

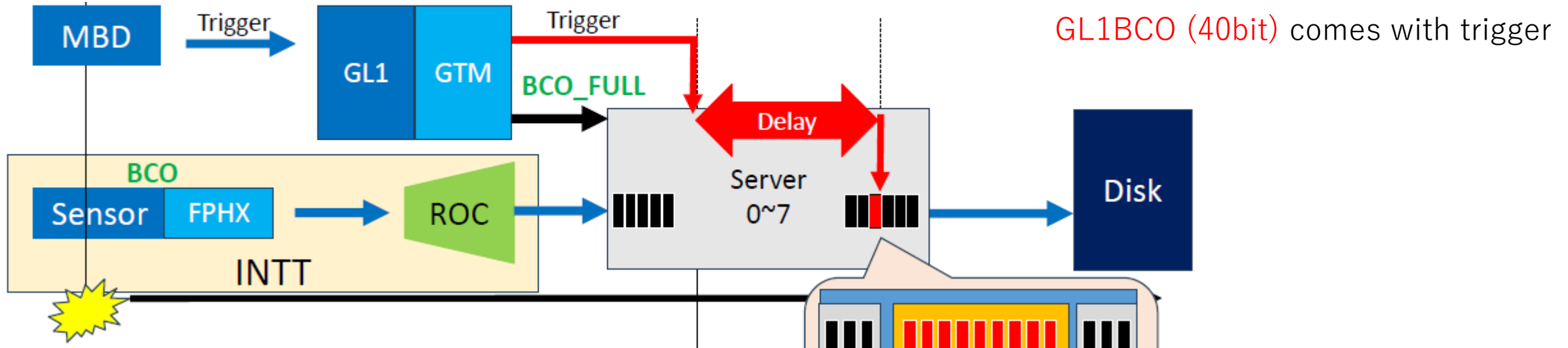
Extended readout and streaming readout

Takashi Hachiya
Nara Women's University

Extended readout and streaming readout

- Extended mode and streaming mode are useful to get more data but a little complicated
 - Trigger mode (normal)
 - Extended mode
 - Streaming mode
- I would like to explain the difference and to show the current status
 - We are taking data w/ the extended mode
 - 100 clock extended
 - Streaming mode was tested

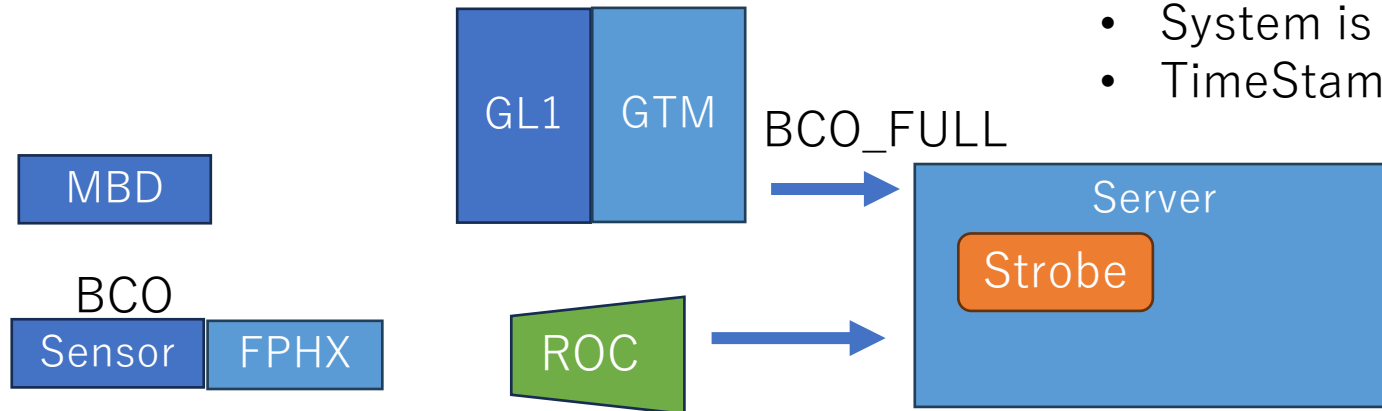
INTT readout diagram for (extended) trigger mode



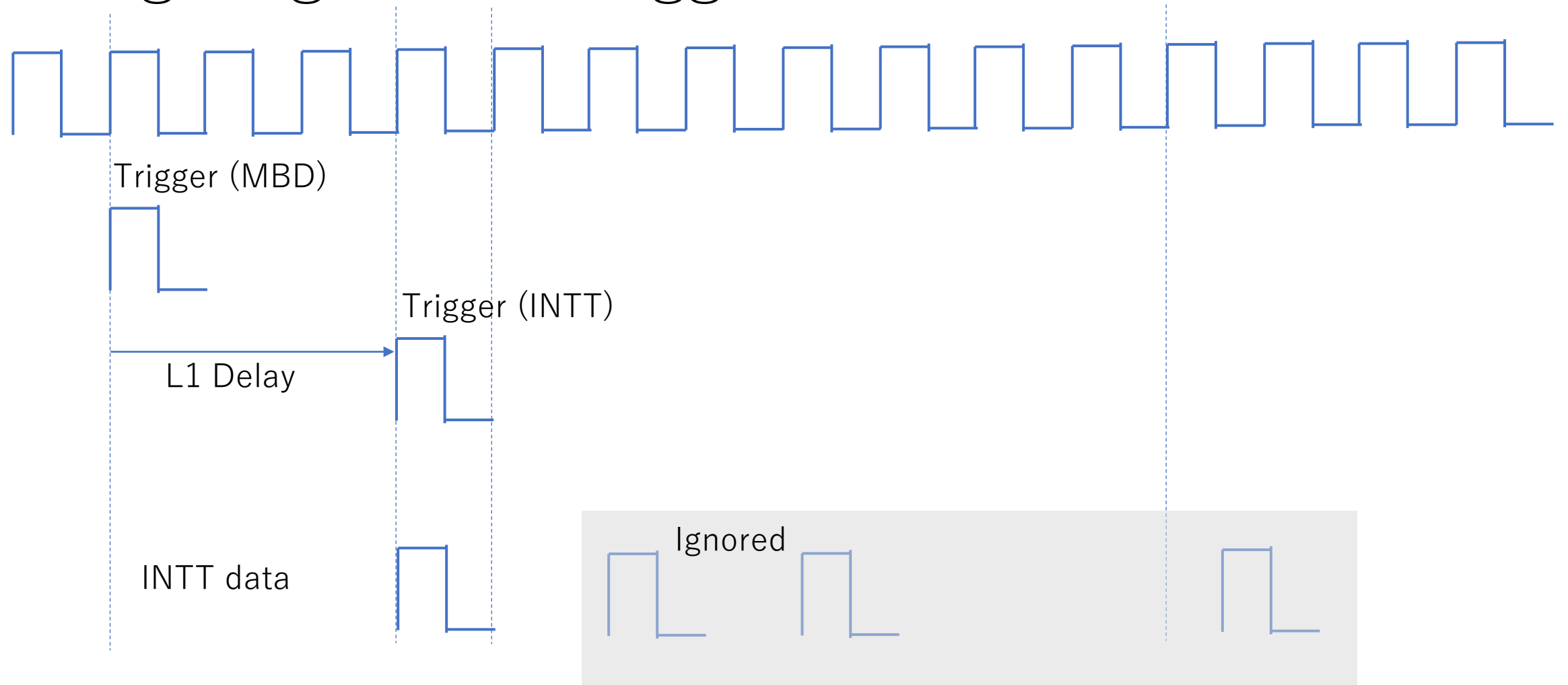
Stream mode

Triggered GL1BCO is not available in INTT system

- GL1BCO is received from GTM
 - System is synchronized
 - TimeStamp (GL1BCO) is available but not sync to trigger

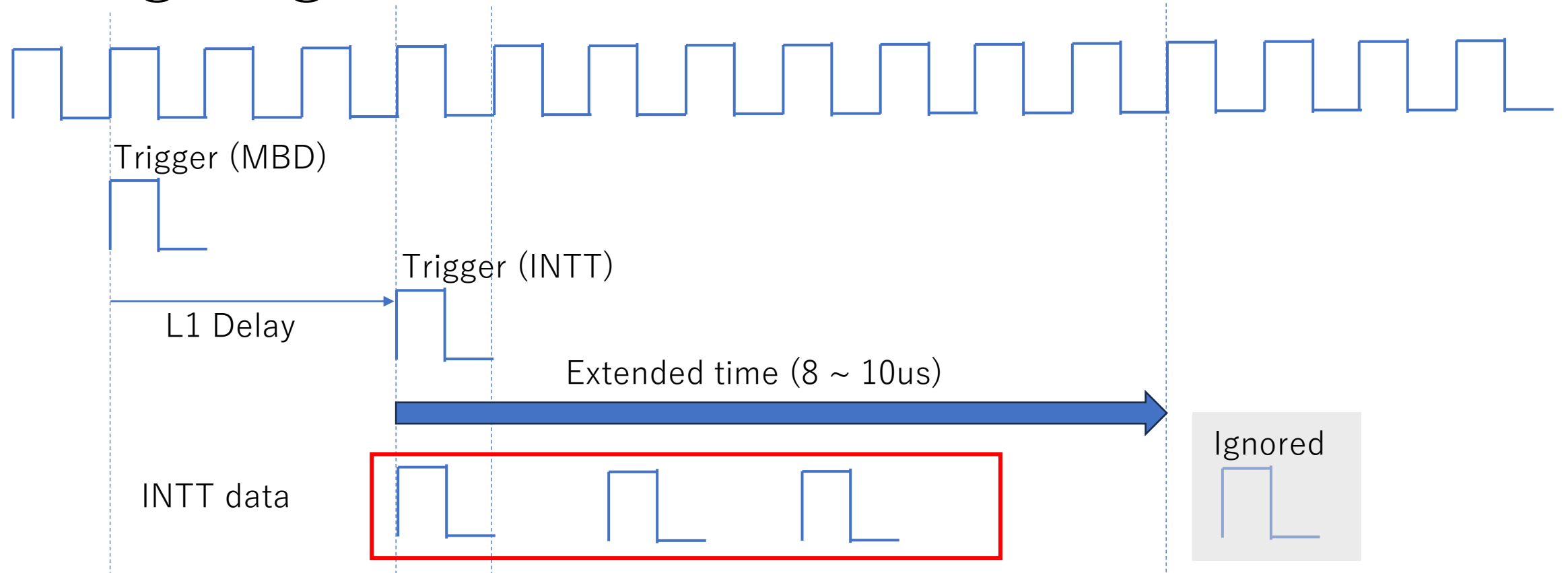


Timing diagram for Trigger mode



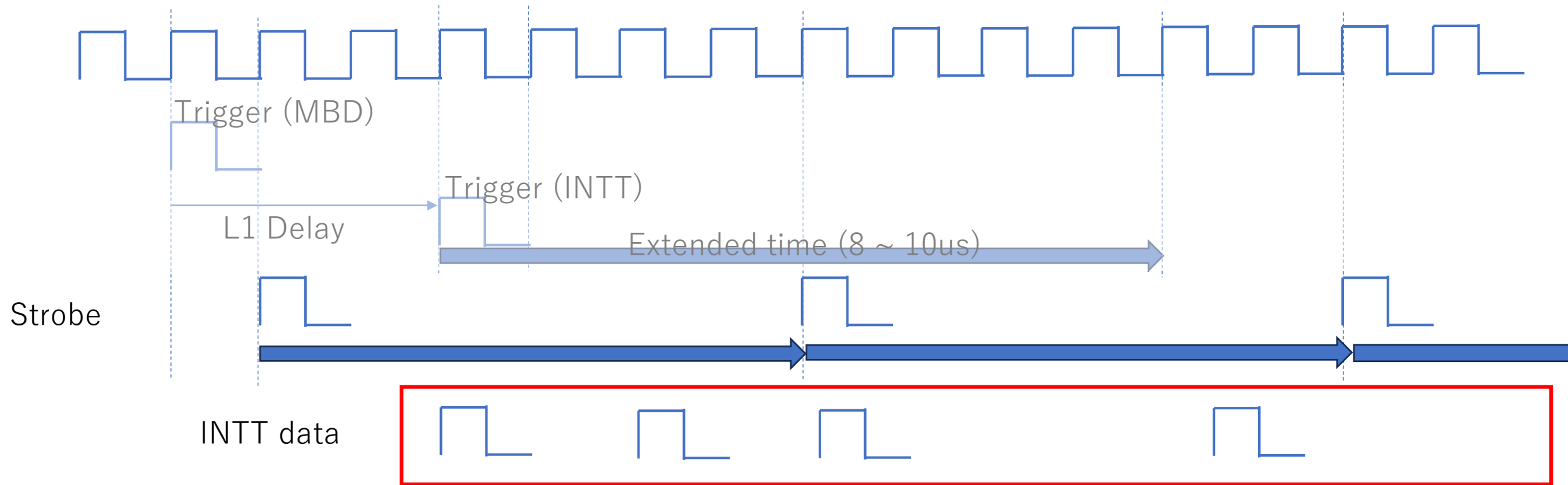
- Data is taken with the trigger
 - Any other hits are not recorded.
 - Standard way to take data. CALO system use this.

Timing diagram for the extended mode



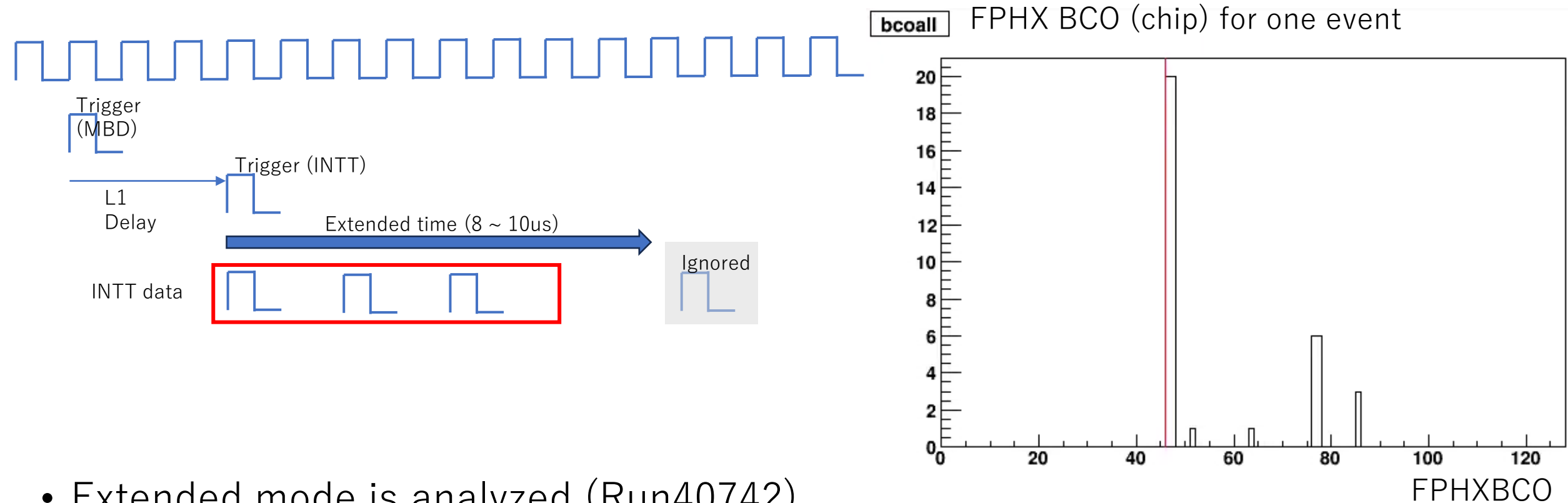
- Data taking window starts by the trigger and keep up to the extended time(10us)
 - All the hits are recorded within the extended period of time. But some deadtime remain
 - INTT (and TPC) use this mode for now
 - DAQ rate is limited by 20KHz (50us)
- We have more data but still lose some data
 - 10% more MB data based on Jin's estimate

Timing diagram for the stream mode



- Data is taking continuously without deadtime
- The strobe (start DAQ) is issued repeatedly (once per 120 BCO = 1 RHIC cycle) and open (120BCO periods)
 - All the hits are recorded without the dead time
 - New and ideal mode INTT (and TPC), need to test if it works

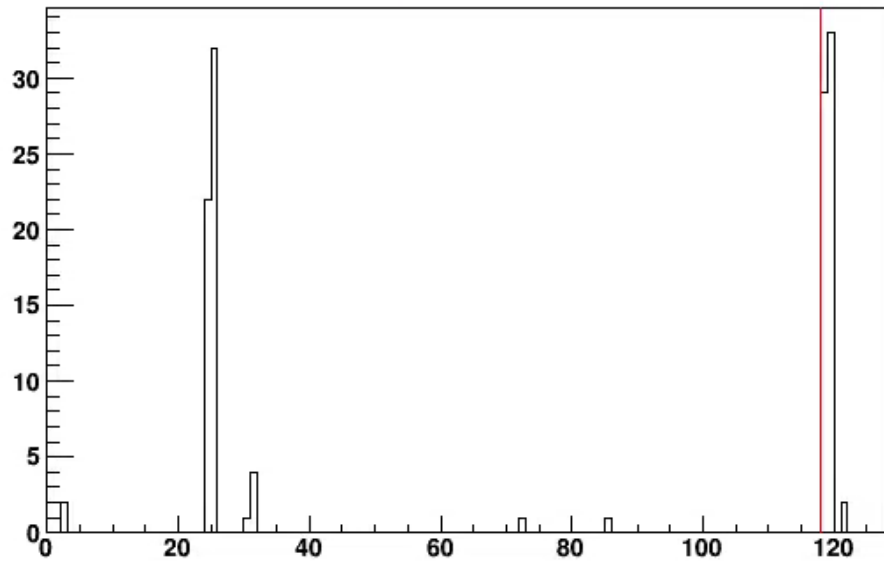
Snapshot of the event reconstruction for the extended mode



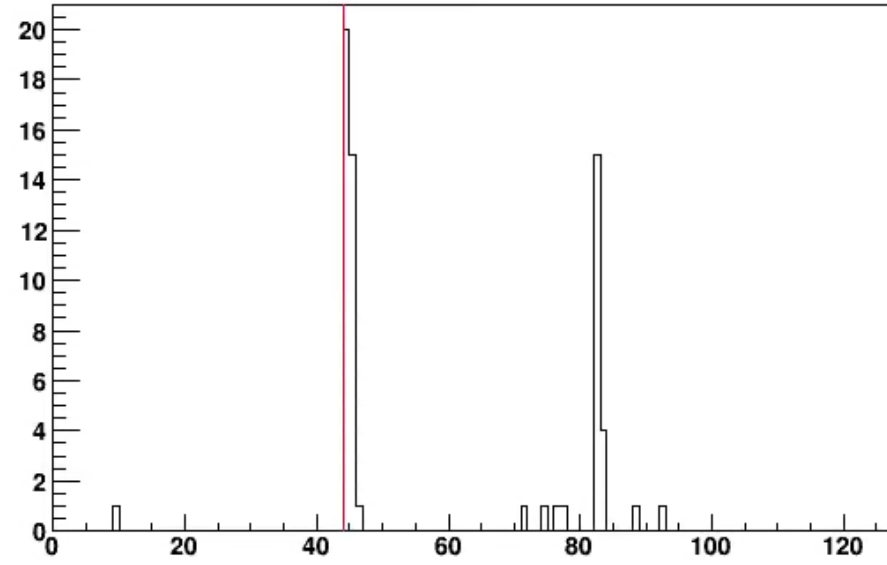
- Extended mode is analyzed (Run40742)
- FPHX BCO (7bit) for single event
 - Red : Trigger
- Shows the BCO peak is synchronized with the trigger and additional peak are there which is the additional data taken during the extended time

More examples for the extended time

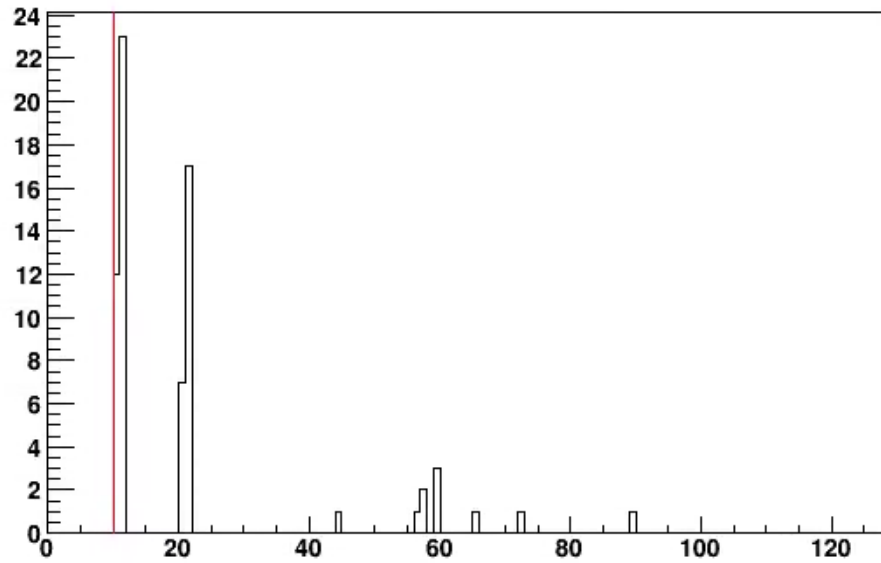
bcoall



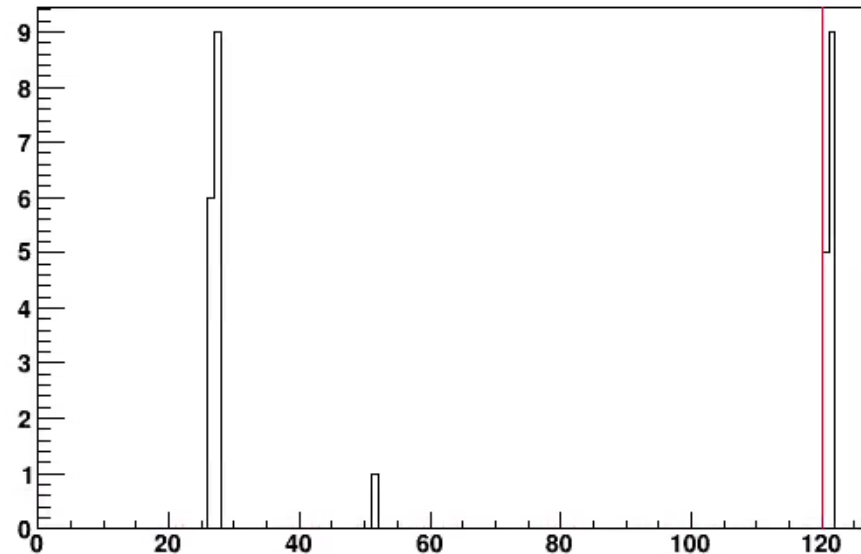
bcoall



bcoall

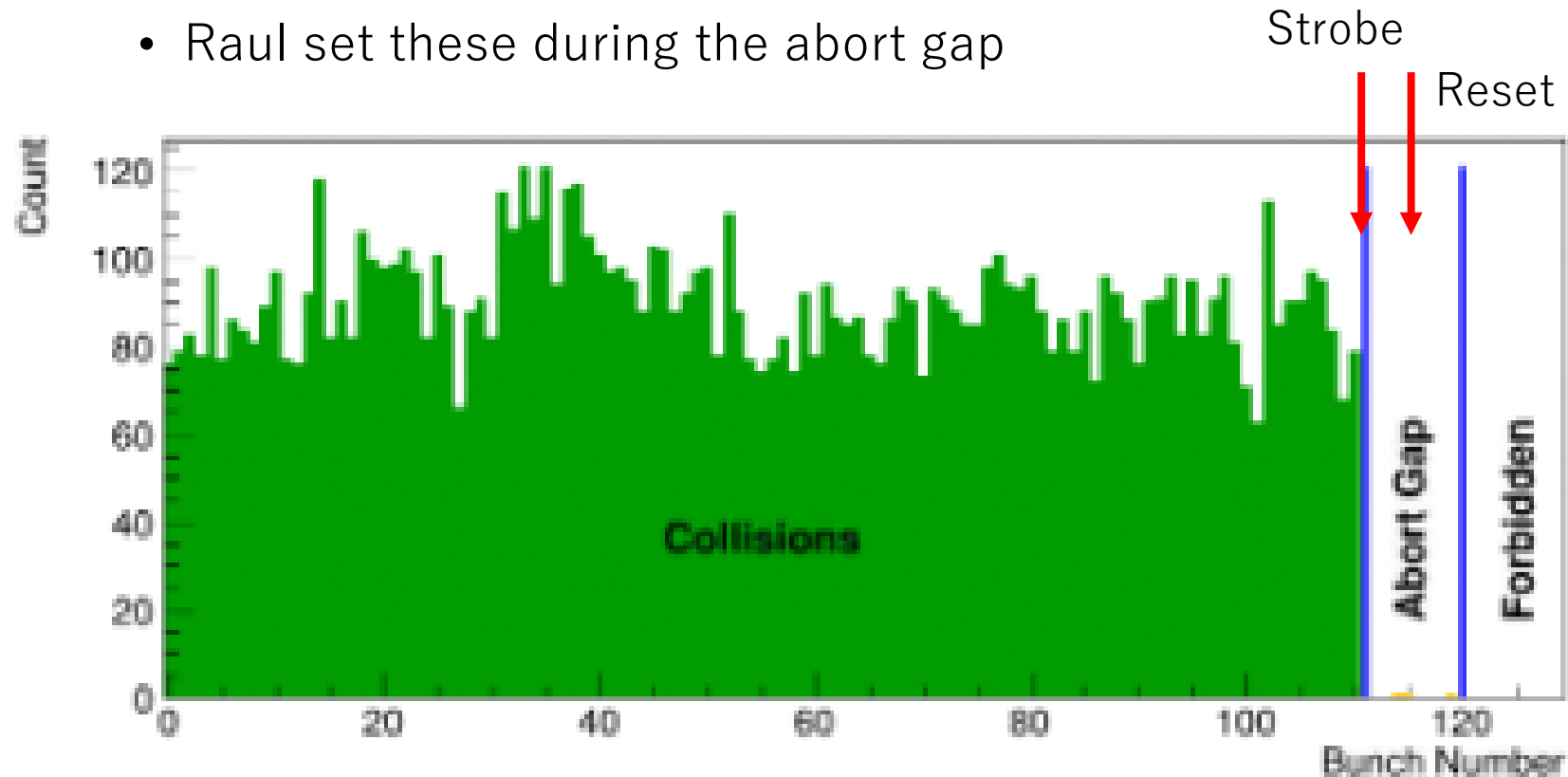


bcoall

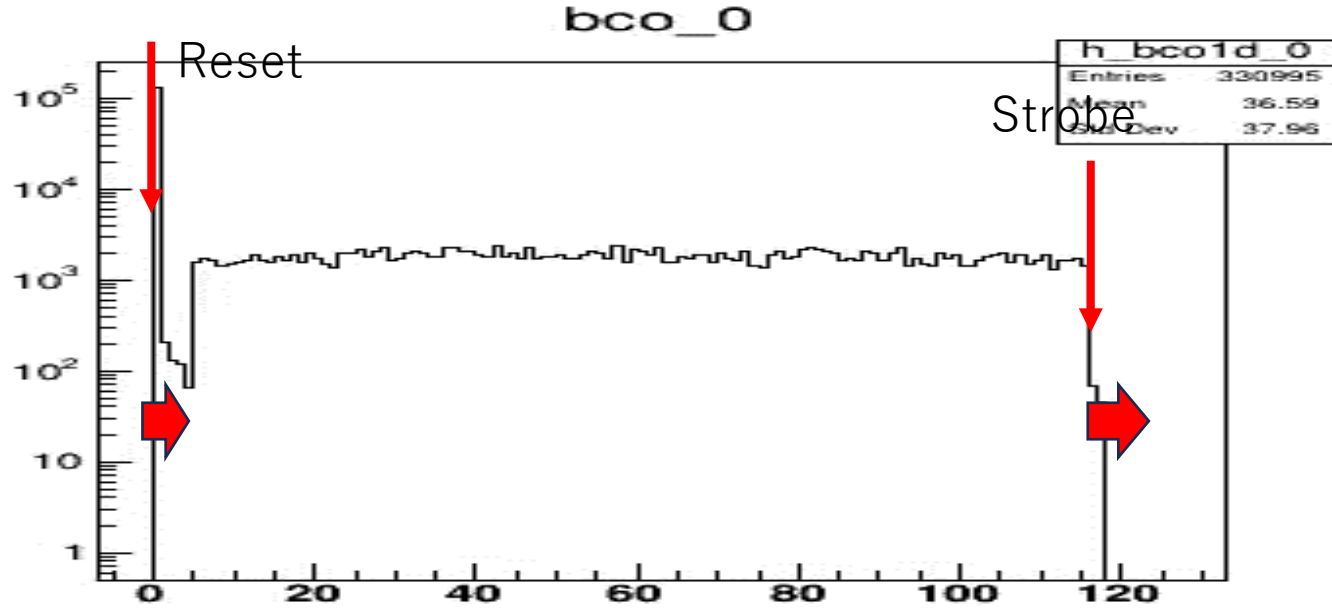


Testing stream mode

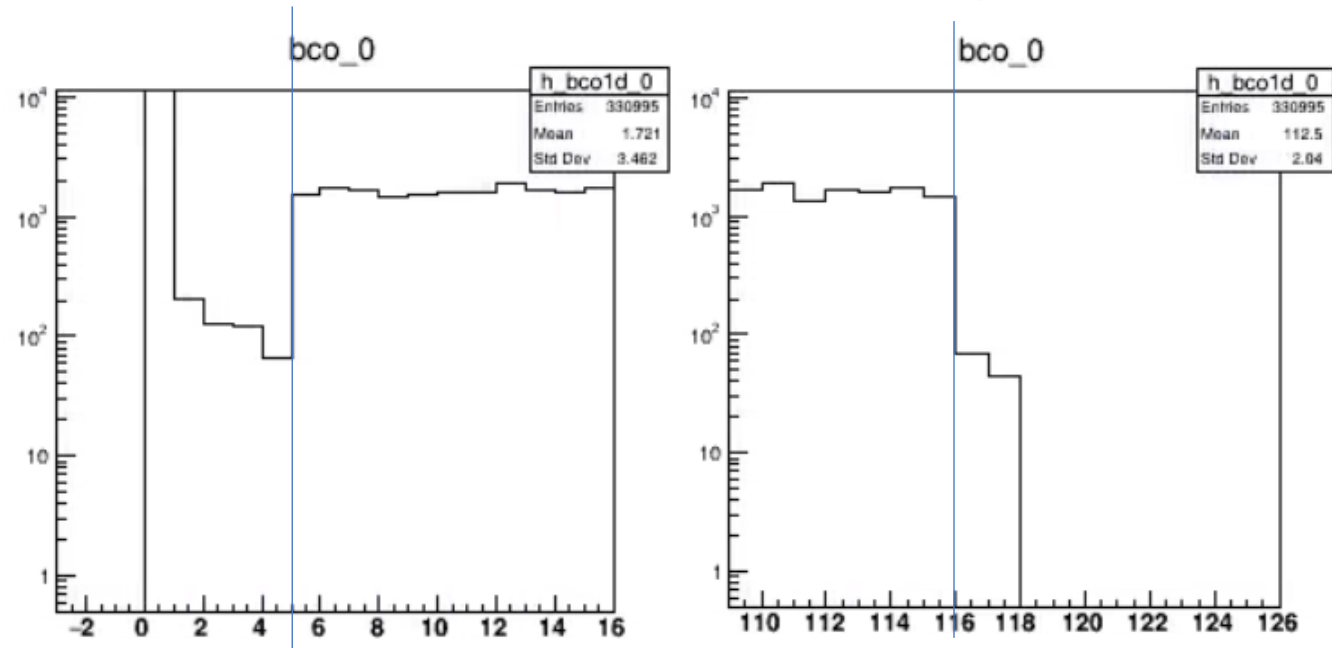
- To make the stream mode simple, Raul proposed
 - Strobe is issued once per 120 BCO (= one RHIC cycle)
 - FPHX BCO is reset by 120 BCO (=one RHIC cycle)
 - These happens at the fixed bunch ID
 - Raul set these during the abort gap



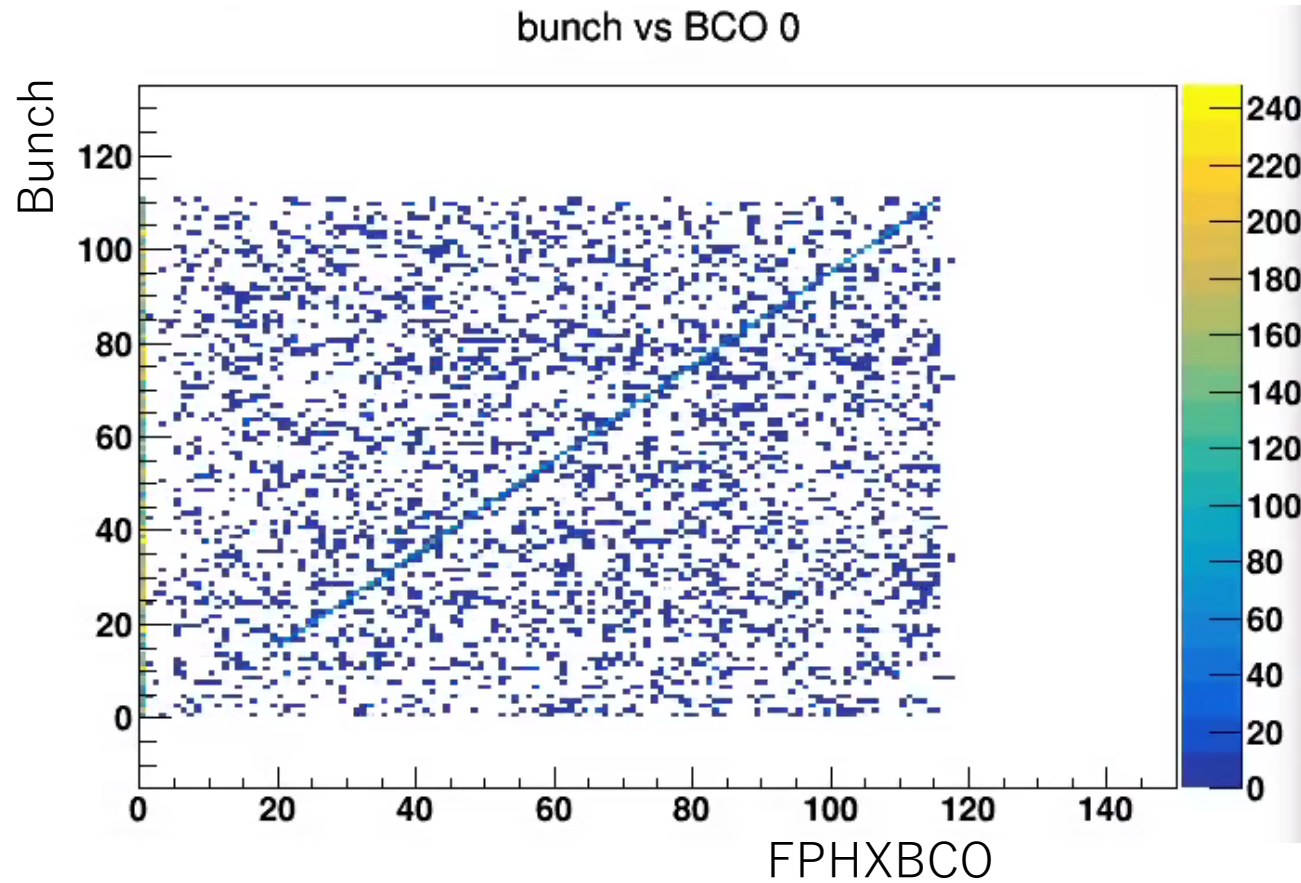
FPHX BCO distribution from the stream mode



- Abort GAP clearly seen
- Strobe and Reset works as expected



Bunch vs FPHX BCO

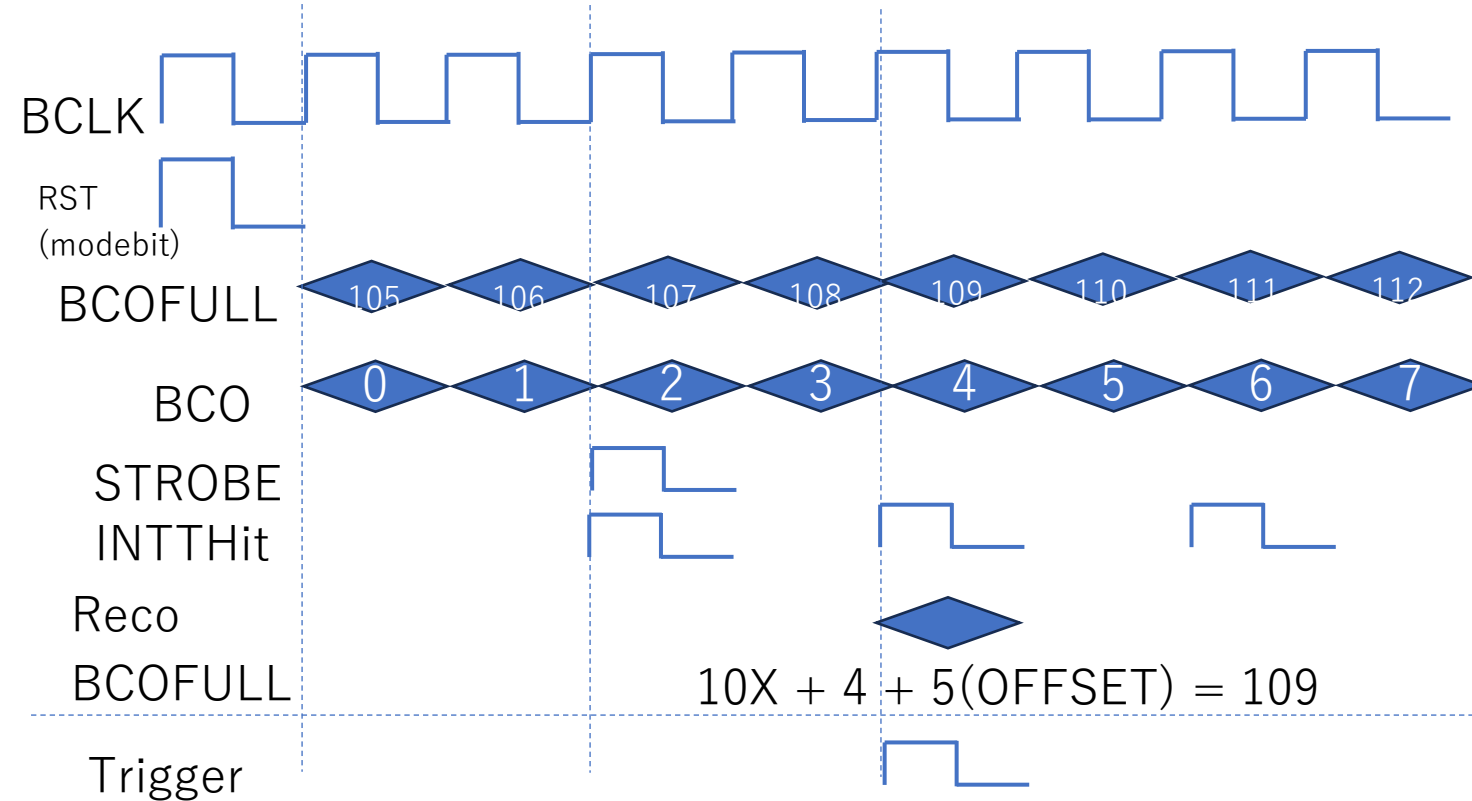


- Bunch and BCO is synchronized because the BCO is reset at the BUNCH

Testing stream mode

Run46090 (Big Partition : GL1 included)

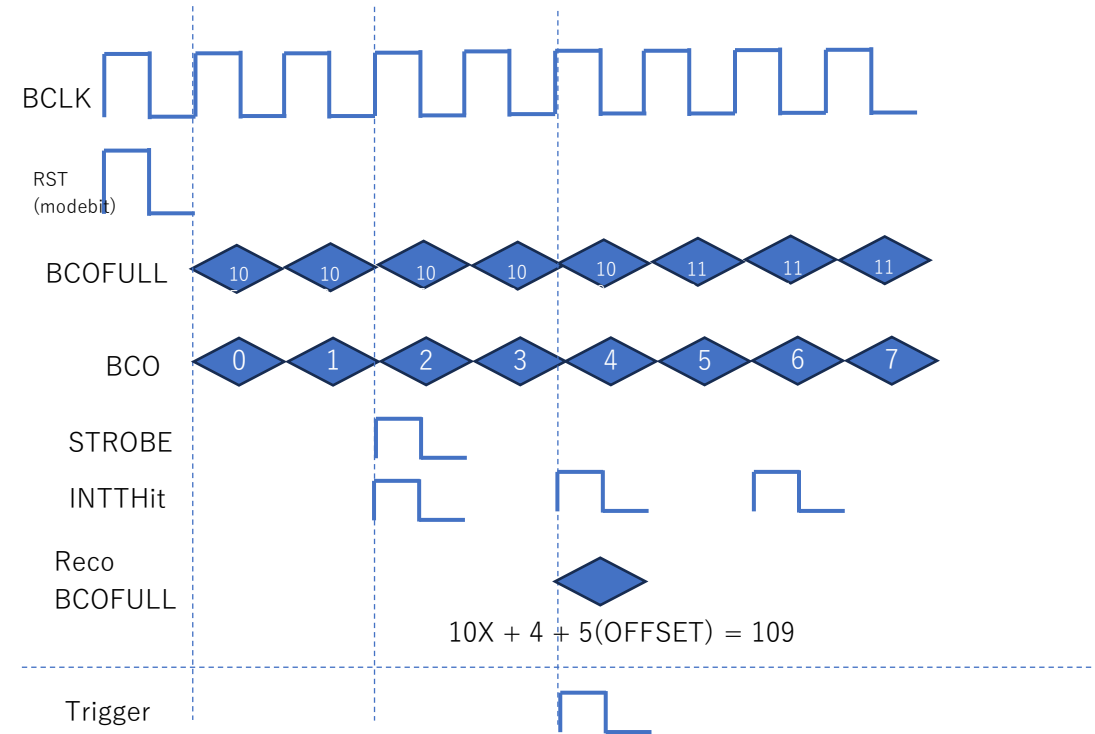
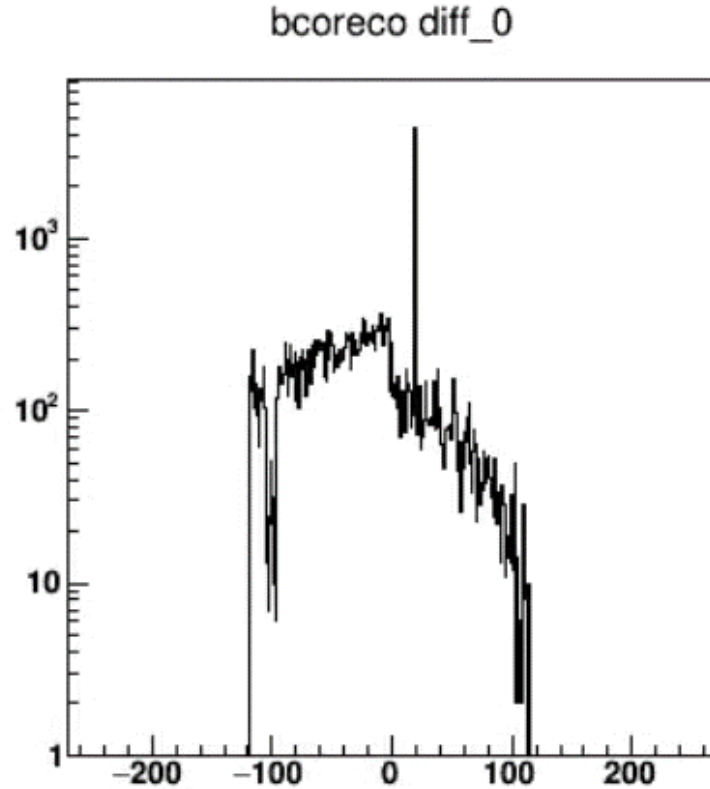
Run46099 (INTT local)



- 2 dataset
 - Streaming INTT data (local mode, 46099)
 - GL1 data (by the trigger, 46090)

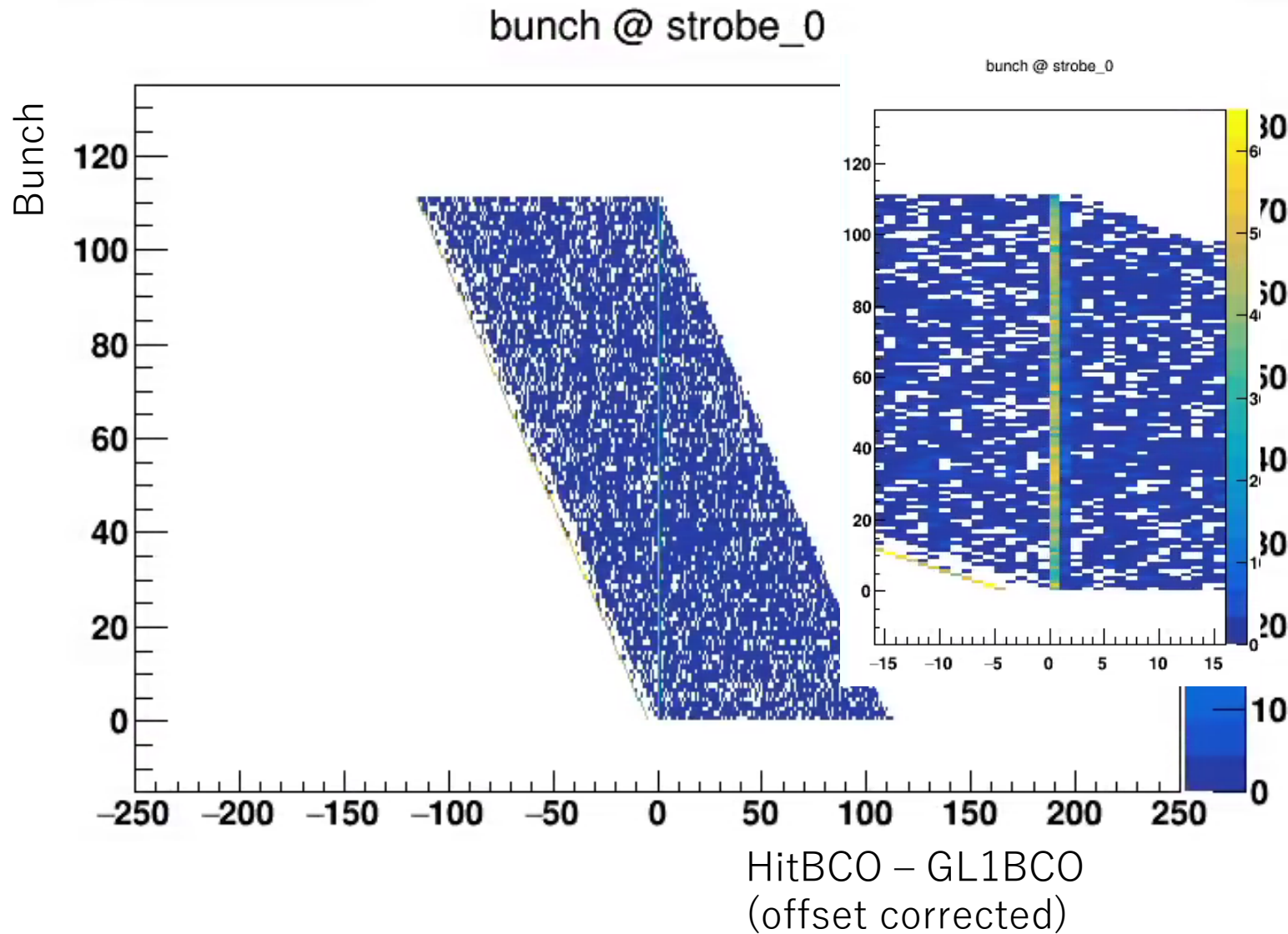
- To make sure if we can take data properly, we reconstruct the HITBCO (40bit) from the stream data and check if the HIT BCO is consistent with the GL1 BCO

Difference between HITBCO and GL1BCO



- A sharp peak is seen
 - Indicate that stream data contains the triggered event
 - Peak has an offset (not at zero)
 - Guess the L1 Delay makes the offset ?
- We could reconstruct the BCO(40bit) from the stream data

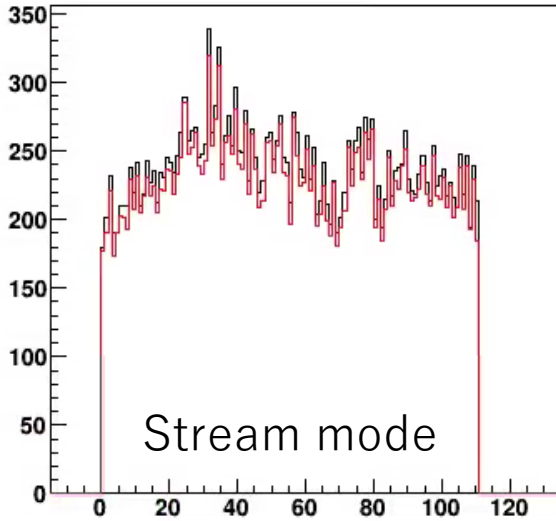
Additional check



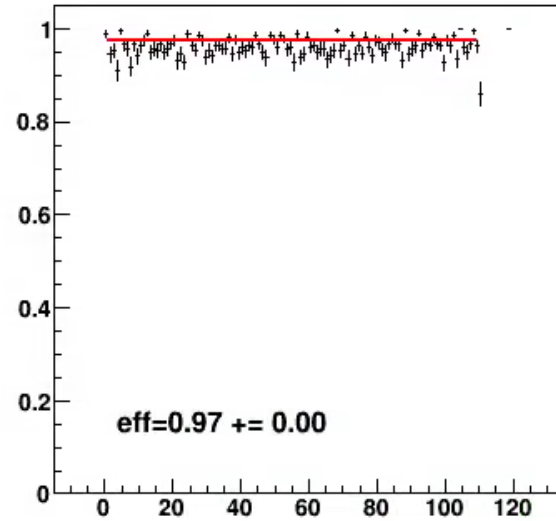
- A sharp peak is seen at difference = 0
 - The offset is corrected
- We could reconstruct the BCO(40bit) from the stream data
- Some entries at difference=1
 - Some hits are leaked. I guess this happens by the charge dependence of the timing (slewing)
 - Take care of this

GL1 BCO finding efficiency from stream mode

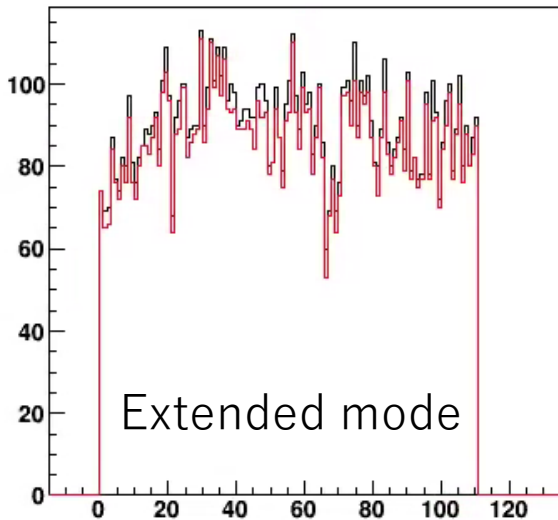
bunch @ gl1



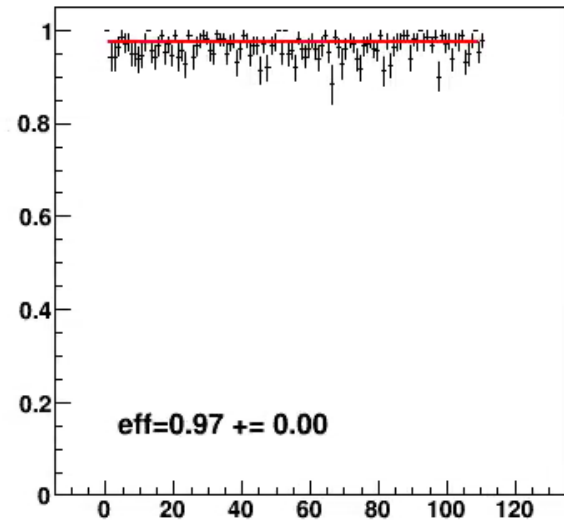
bunch @ evt all felix



bunch @ gl1



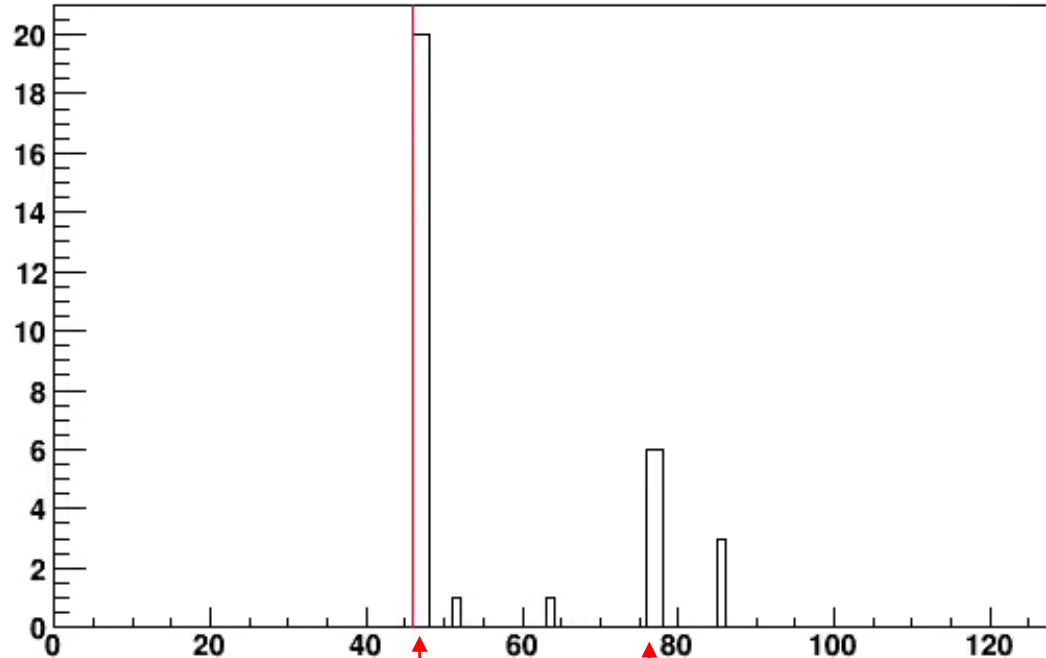
bunch @ evt all felix



- $\text{DIFF}(\text{HitBCO}-\text{GL1BCO})==0$ used (single clock only)
- $\text{eff} = \frac{\text{Nevts}(\text{hitw/ HitBCO}==\text{GL1GCO})}{\text{Nevts}(\text{GL1})}$
- Stream mode
 - $\text{Eff} = 97.3 \pm 0.1\%$
- Extended mode (run46106)
 - $\text{Eff} = 97.3 \pm 0.1\%$
 - 0.9733 ± 0.00165065
- Finding efficiency in stream mode is consistent w/ that in the extended trigger mode

Next: Split the events

bcoall

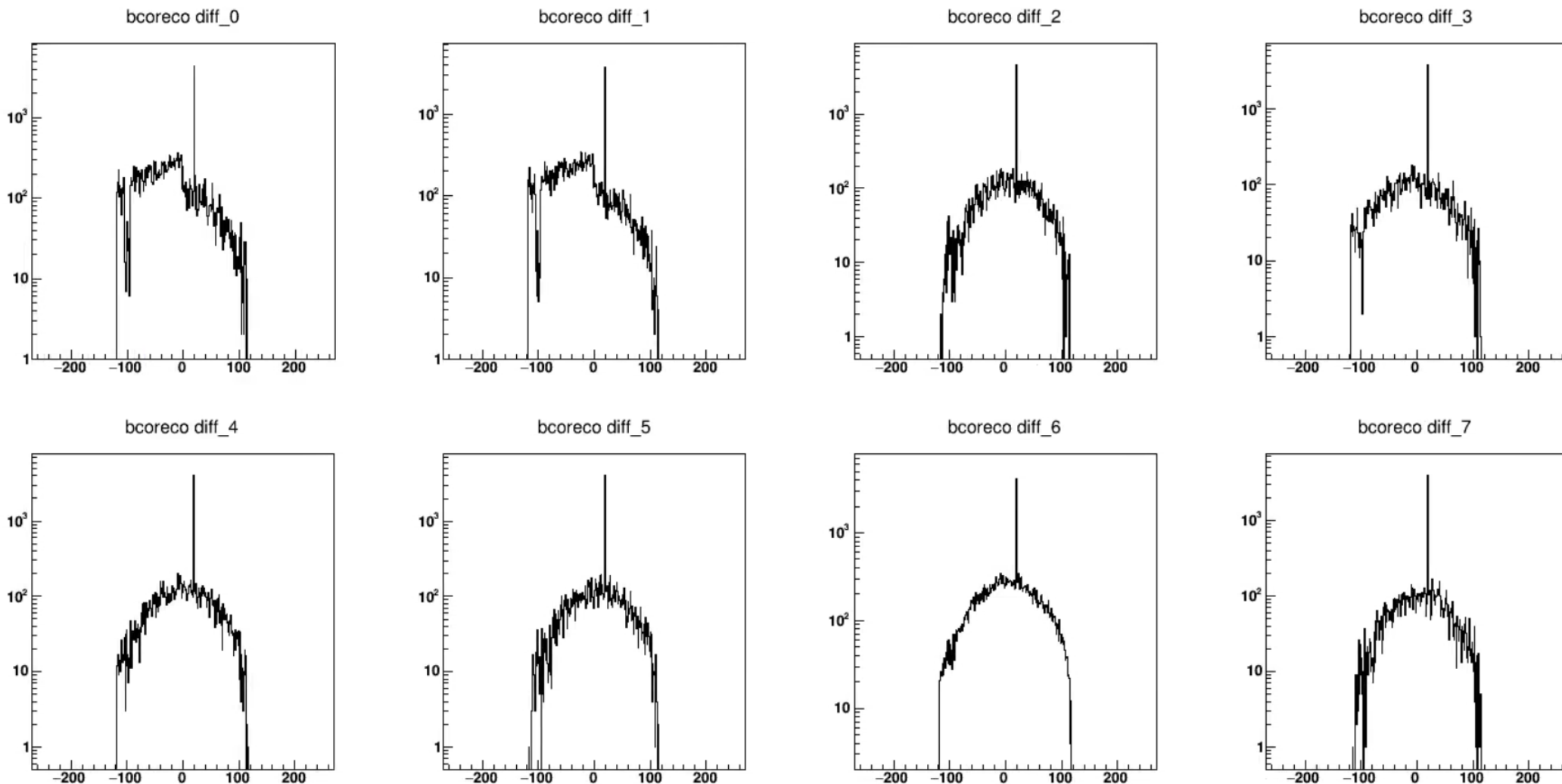


Triggered
event

Collision
not from
the trigger

- We have multiple collisions in the time window (for both the extended and the stream mode)
- Proposal
 - Identify the BCO which have multiple hits
 - Needs the threshold
 - Split the events using the BCO
- This software needs to be developed

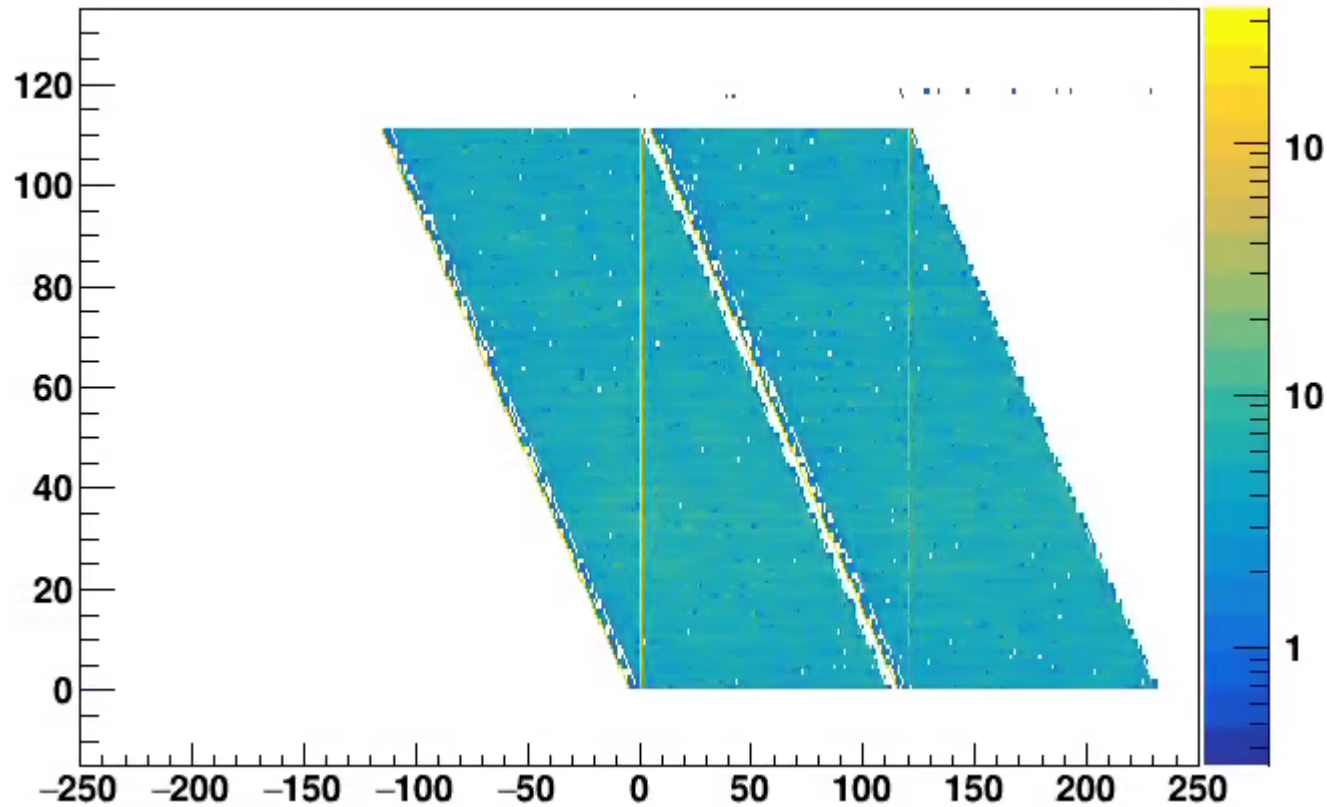
Streamデータで再構成したBCOFULLとGL1BCOの差



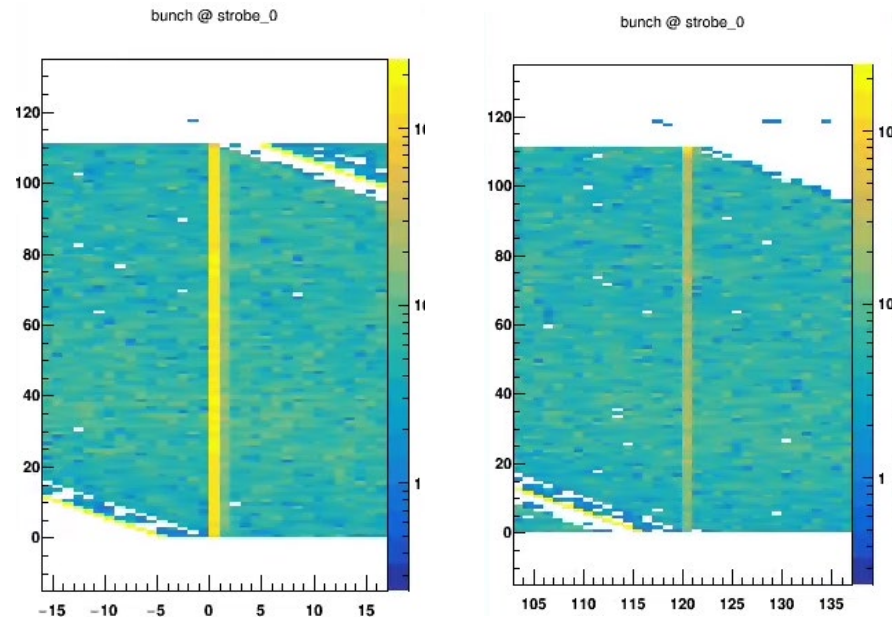
- 全FELIXで同じ場所にピークがある。
- ヒストグラムのエントリーはヒット。
 - ホットチャンネルは抜いていない

Mixup behavior from stream mode

bunch @ strobe_0



- If open up the window twice, 2nd vertical line is found
- Take care of this

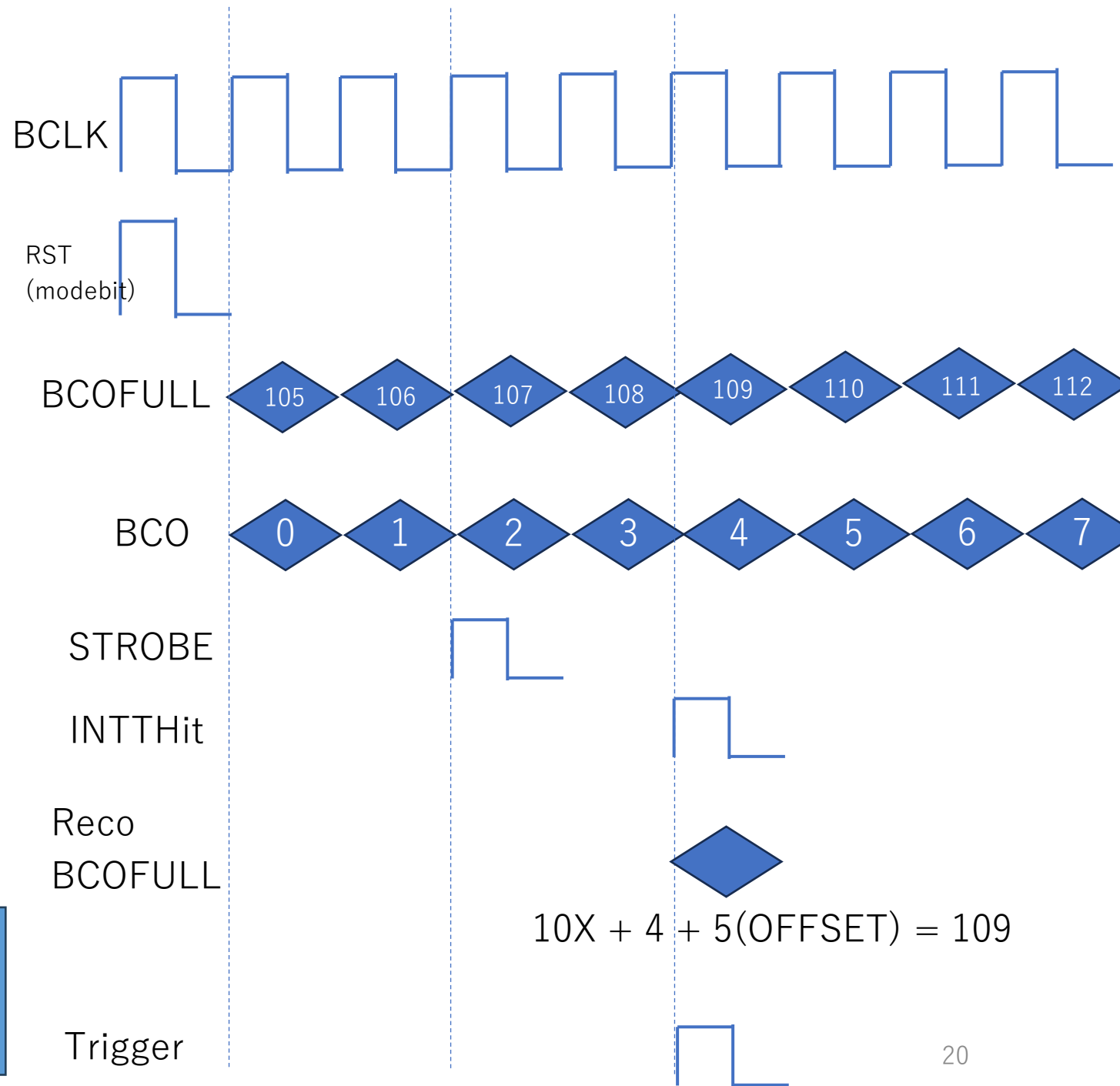
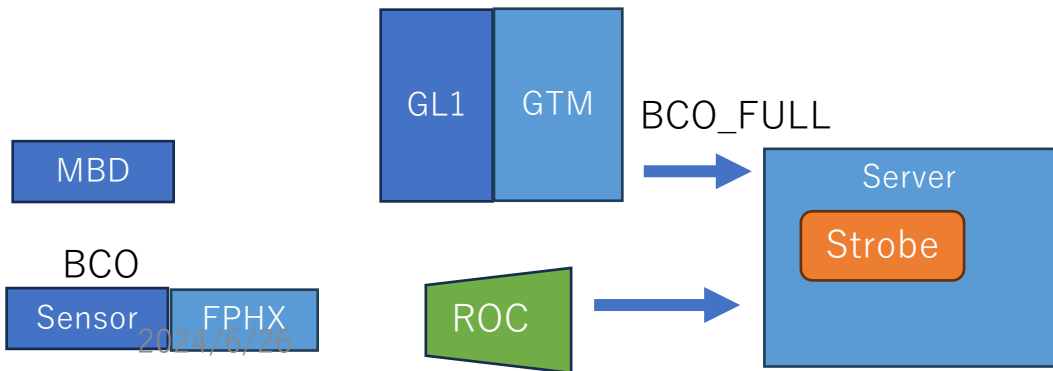


- STREAMにおけるBCOFULL再構成

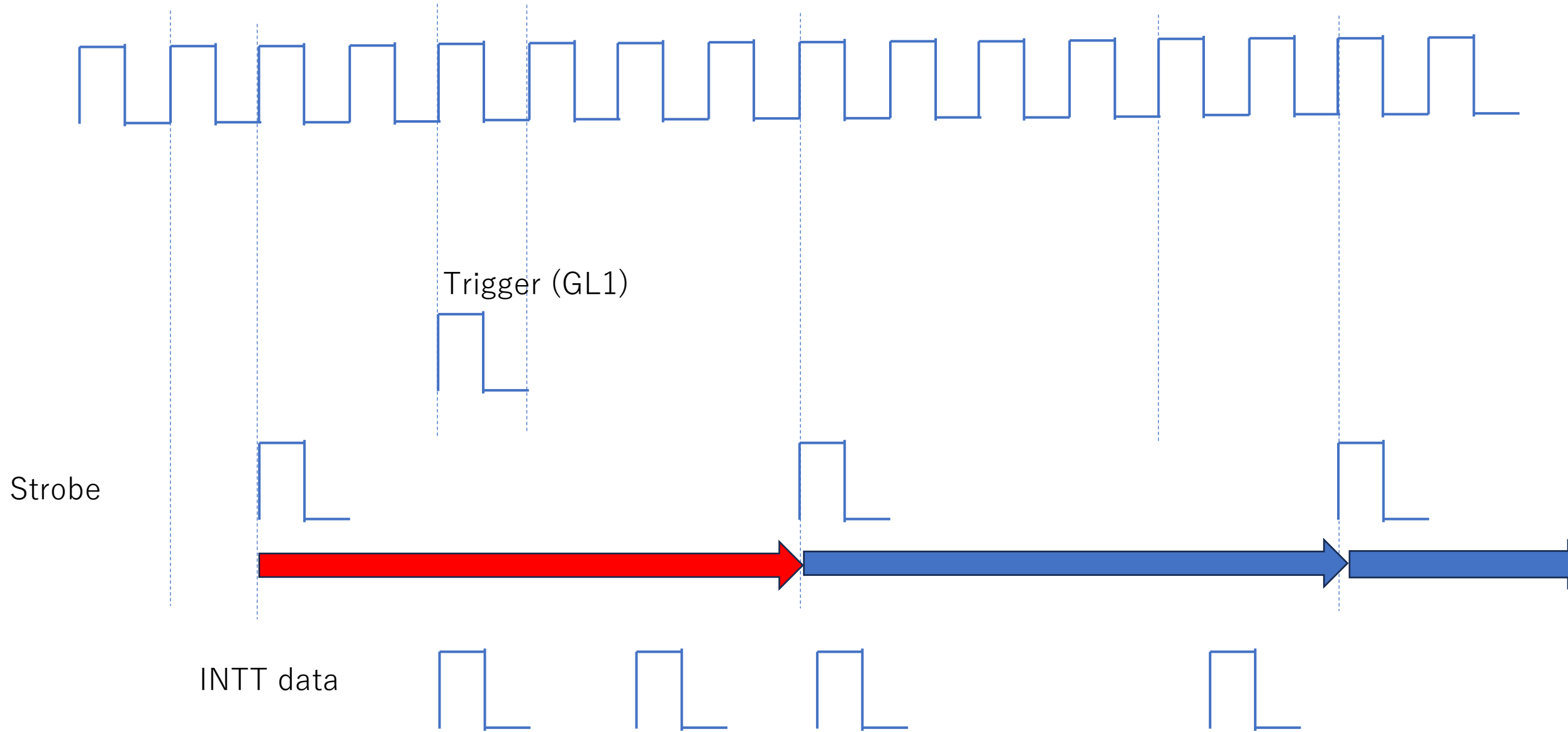
- BCOFULL(40)@Strobe
- BCO(7)
- Reco BCOFULL
 - BCOFULL(39-8) + BCO(7:0) + (offset)

- BCOFULL とBCOのSTARTは同じ

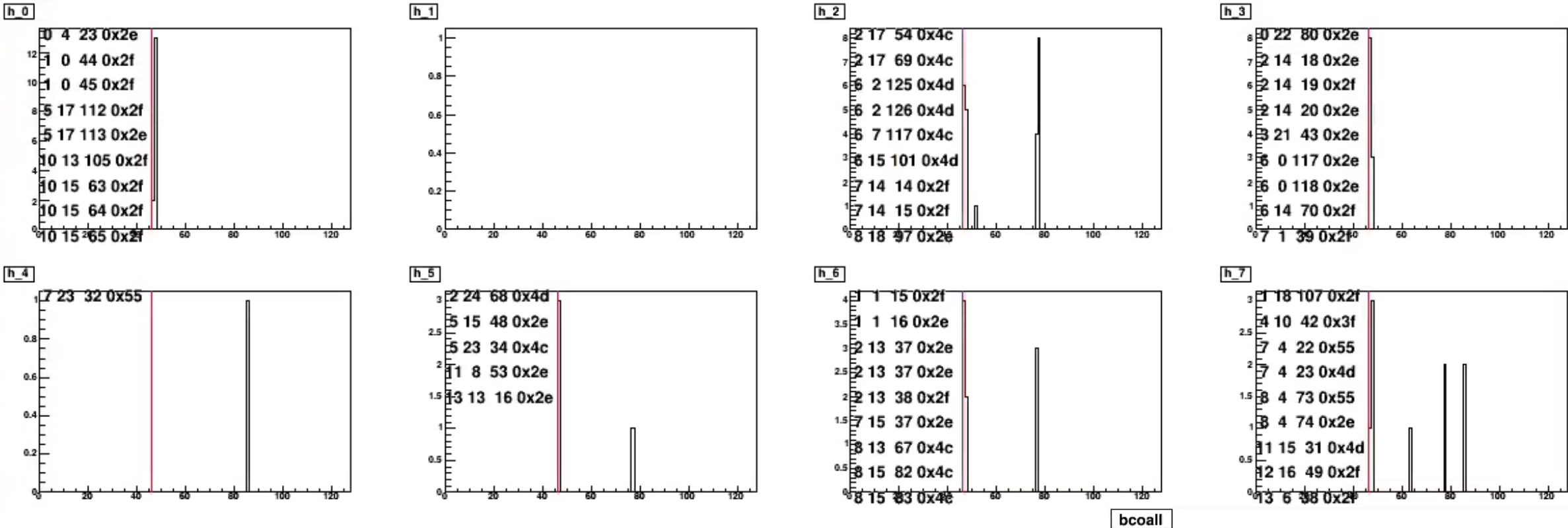
- クロックのEdgeでカウントアップする。
- ただし
 - BCOFULLは、以前の物から継続。
 - BCOは0から始まる



Timing (Stream mode)



Snapshot of the event reconstruction for the extended mode



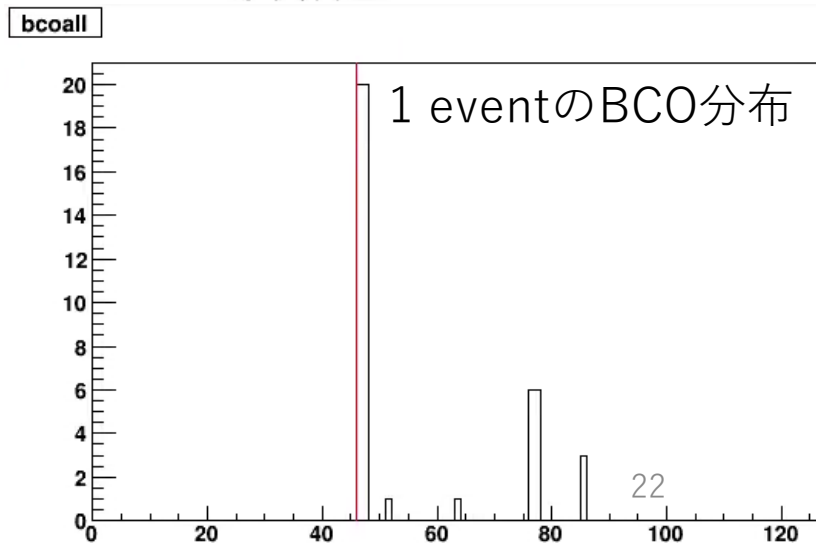
• TriggerReadOutで取られたデータ (Run40742)

• 1 event のBCO分布

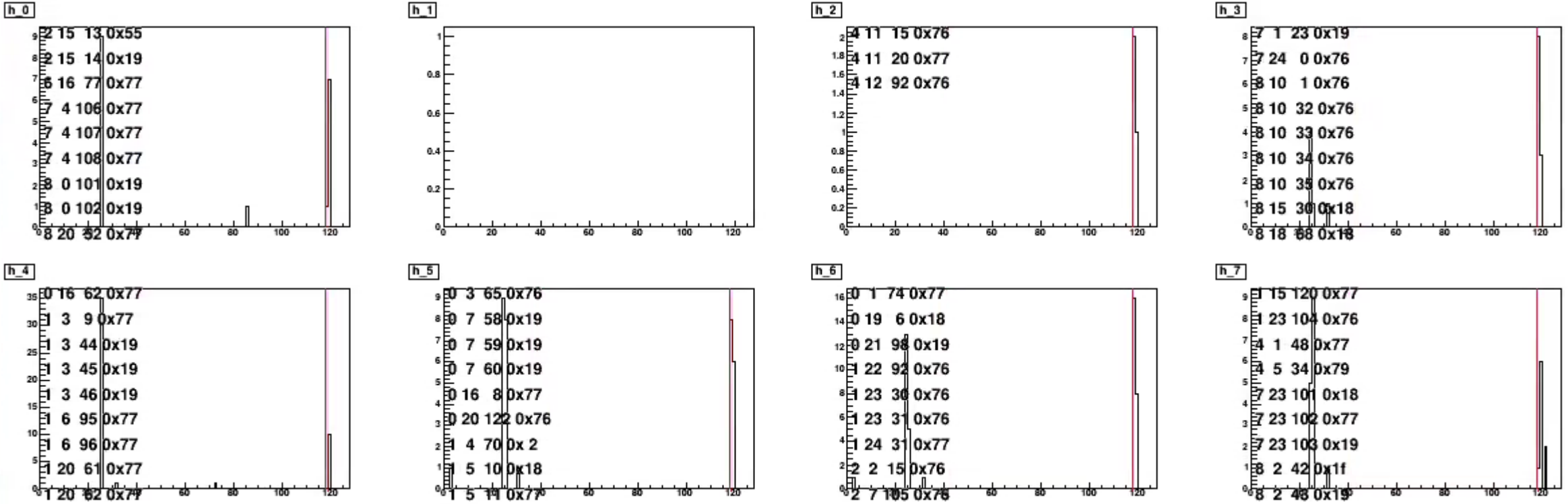
- 赤線：BCO-FULL(+OFFSET)のタイミング
- 上の8分布：FELIX毎のBCO分布
- 横の分布：全FELIX分のタイミング分布
 - 1つのChipで同じBCOを持つ複数のヒットは1つのBCOと数えている

• 赤線とBCOピークが一致している。

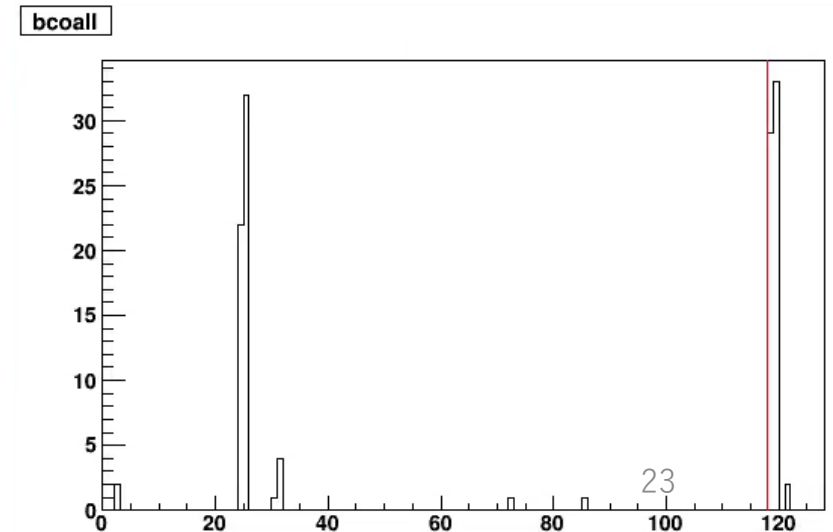
- 2つ目のピークがあり、それがExtendedTimeで測定された別衝突であると考えられる。



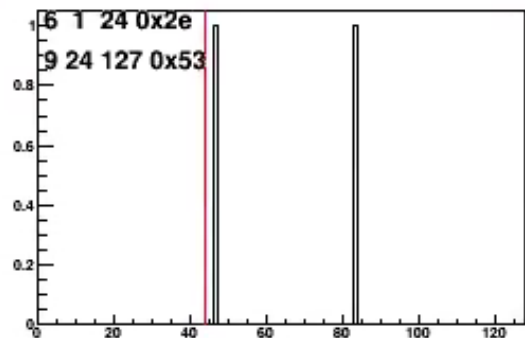
Extended Timeの中のイベントをBCOで分離する方法を開発 2



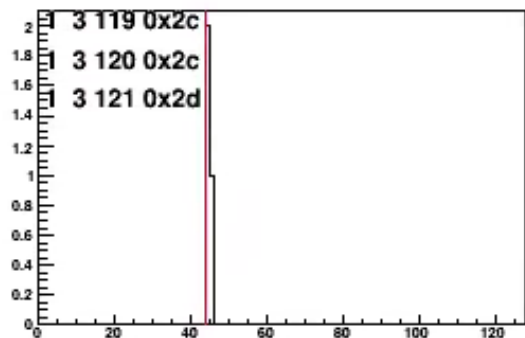
- トリガー(BCOFULL)より左側は、本当は128以降の時刻が、桁上がりして巻き戻って見えている。



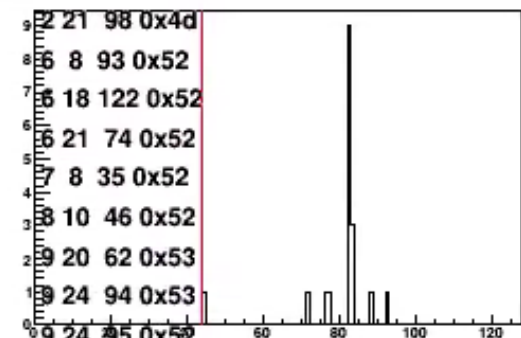
h_0



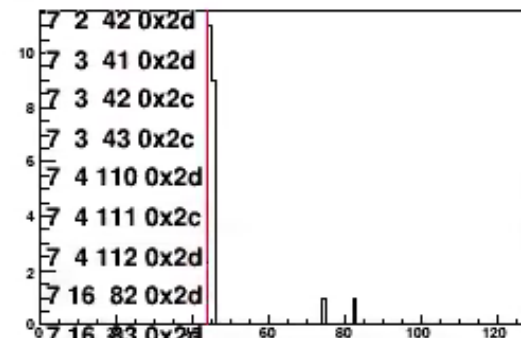
h_1



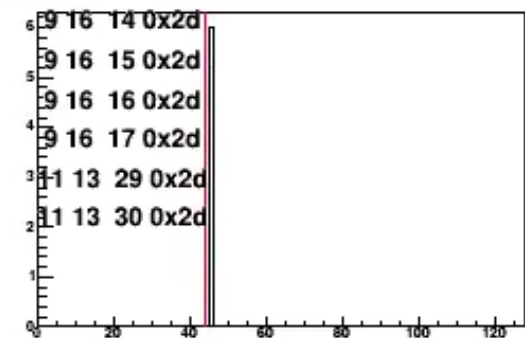
h_2



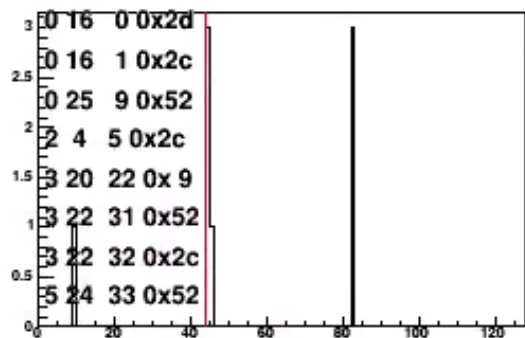
h_3



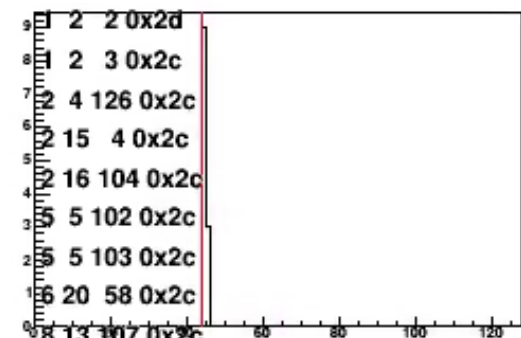
h_4



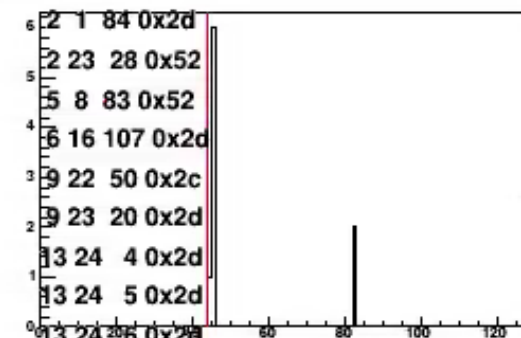
h_5



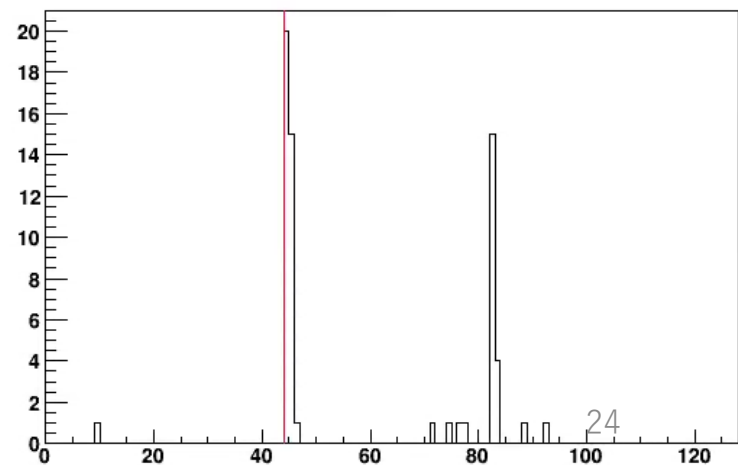
h_6

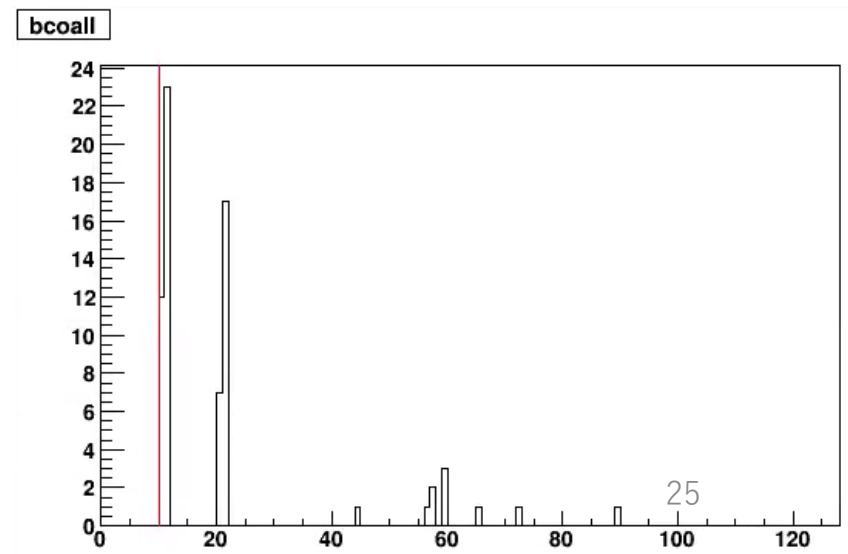
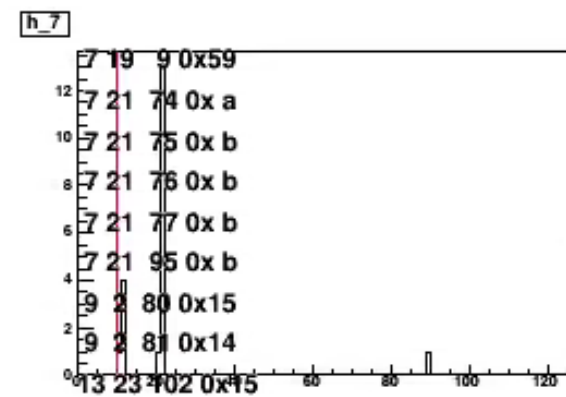
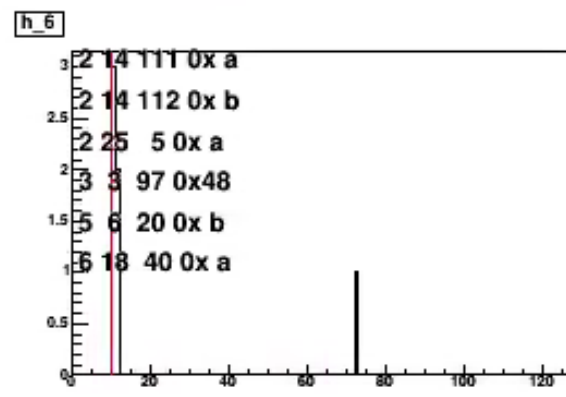
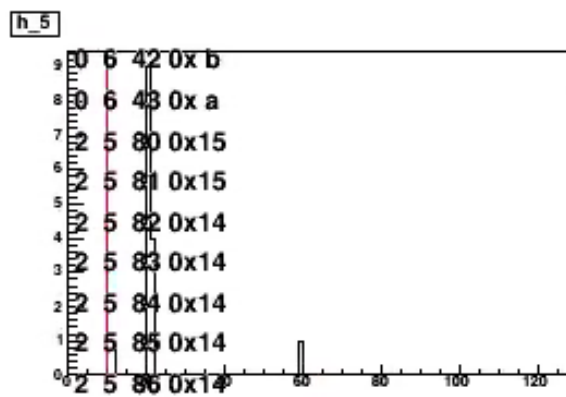
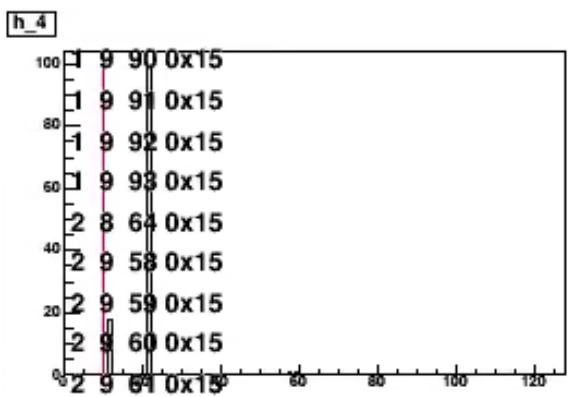
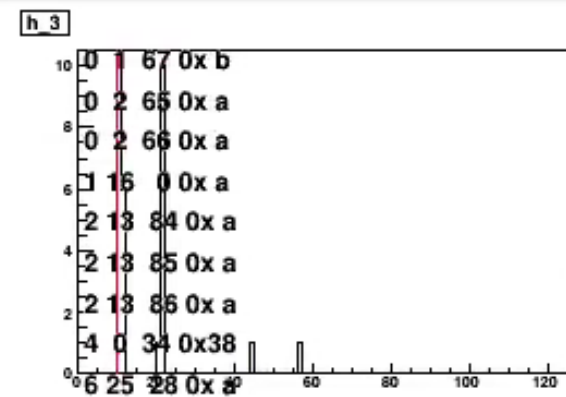
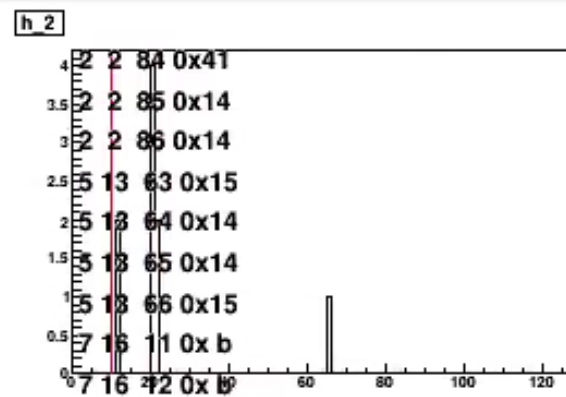
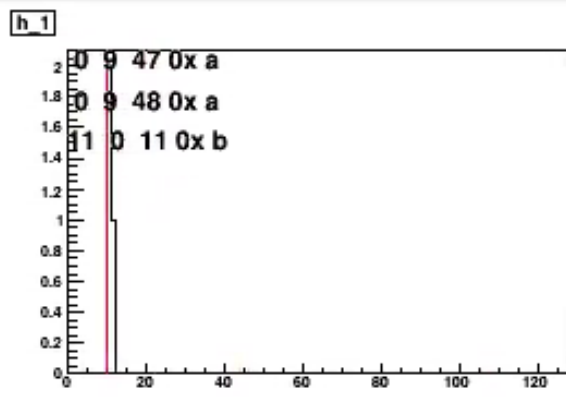
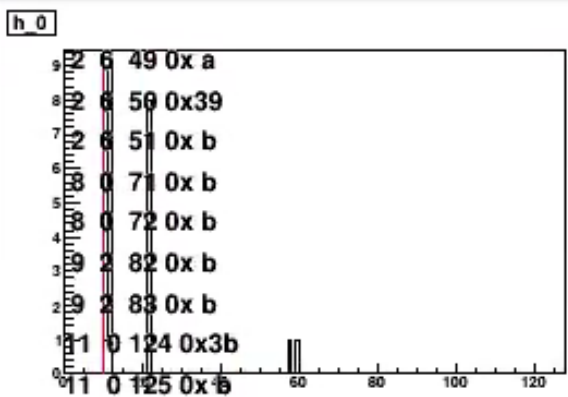


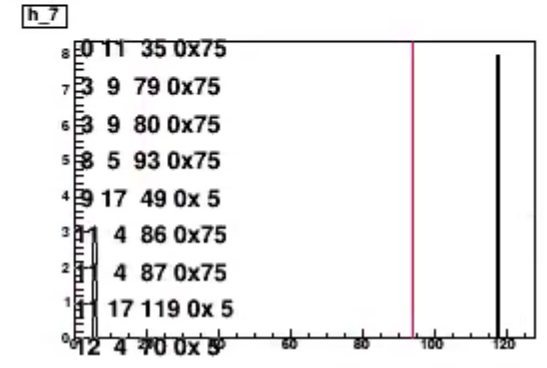
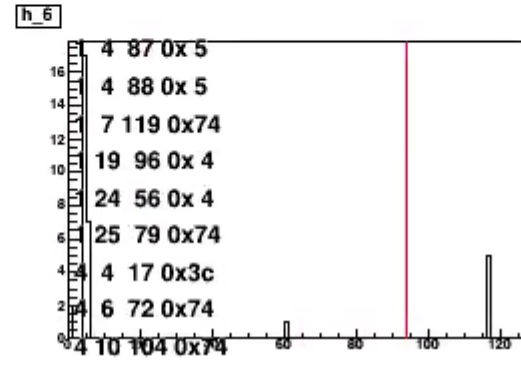
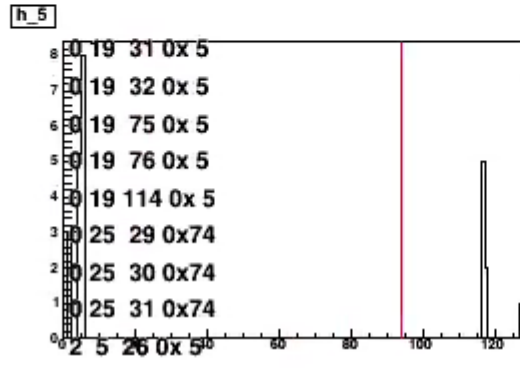
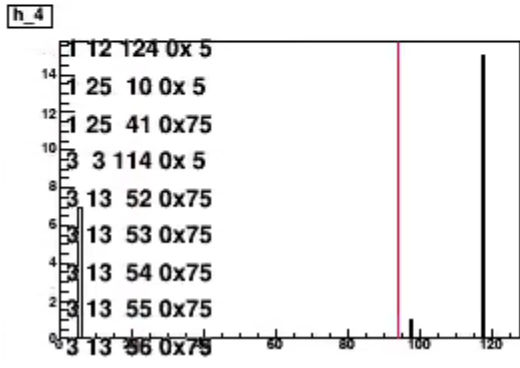
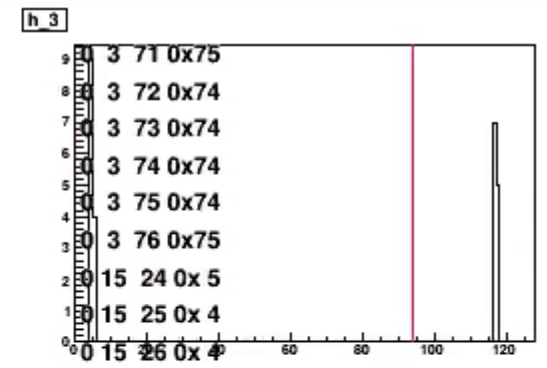
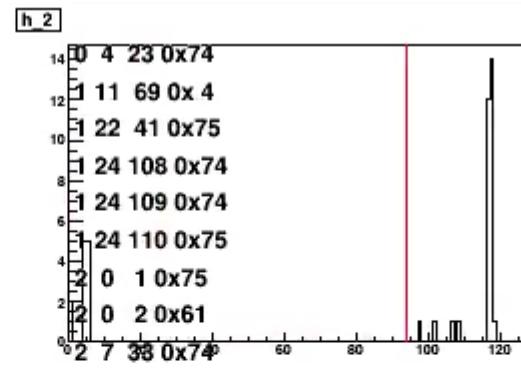
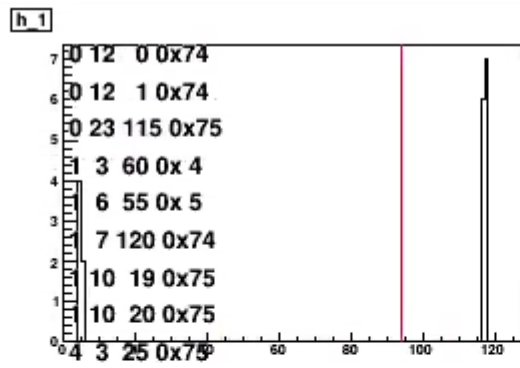
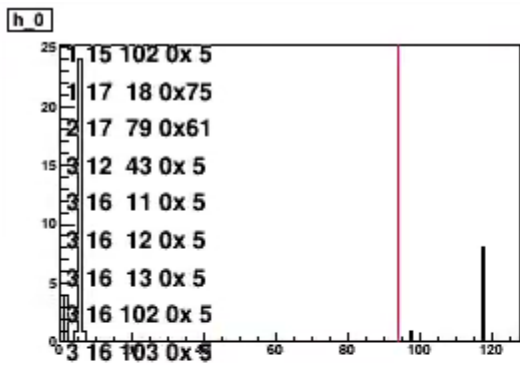
h_7



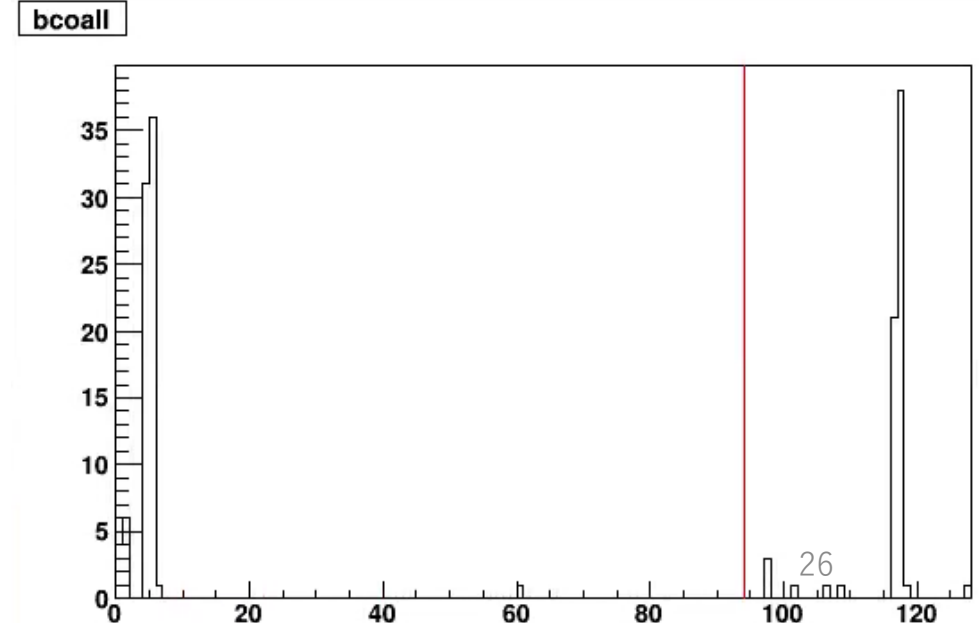
bcoall



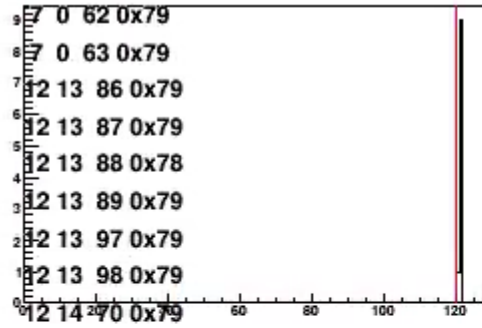




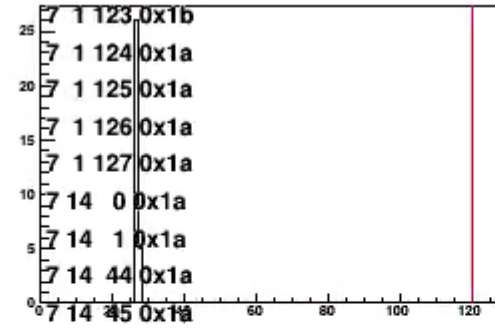
- トリガー (BCOFULL)とBCOのタイミングが全くあっていない時がある。



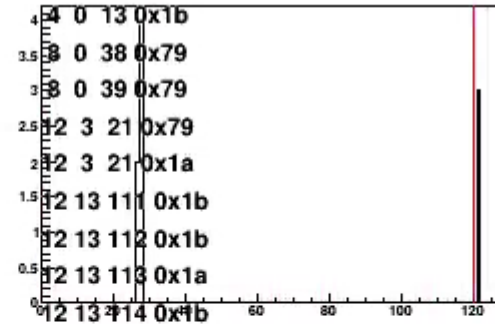
h_0



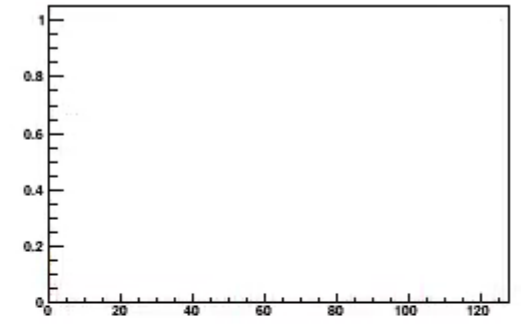
h_1



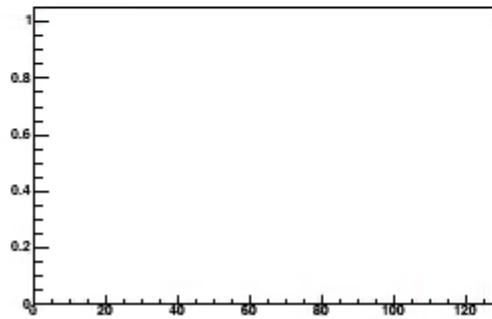
h_2



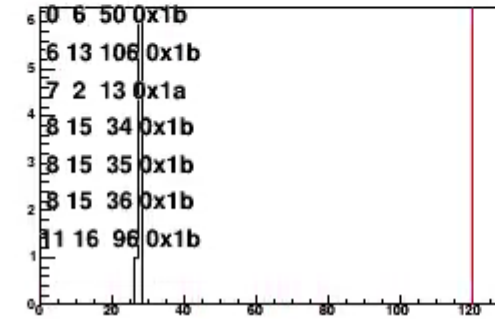
h_3



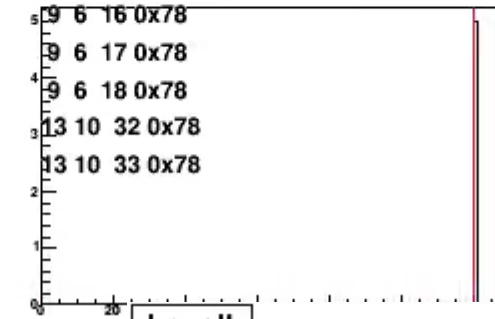
h_4



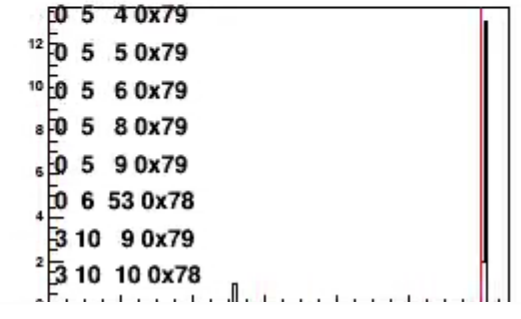
h_5



h_6



h_7



bcoall

