TOP Feature Extraction

Progress Updates and Plans

Harsh Purwar

University of Hawaii at Manoa,
Honolulu, HI, USA

TOP Feature Extraction Group Meeting
June 9th, 2025

Project status

- **Recently finished testing 2nd version of TOP Feature Extraction at UH.**
- Successfully tested version 1 at KEK (with 1 TOP FEE, s09a).
- Tested version 1 with all TOP FEEs to check if there is any significant impact on performance.
  - Powered up all the TOP FEEs connected to rtop1.
  - Tried running at 1kHz – ~~Busy after some events!~~ No BUSY, seems to work fine.
  - Busy at higher trigger rate (30 kHz) – with full occupancy.

- Finished the script for reading pedestals in parallel from all TOP FEEs.
  - Testing it at KEK, now…

- **Next steps:**
  - Version 3 testing at UH
  - Test version 3 at KEK

TOP Feat Ext – Tests at KEK

Version 1

- No pedestal subtraction
- Single TOP FEE

Preparations for testing TOP FE at KEK
(version 1 – no pedestal subtraction)

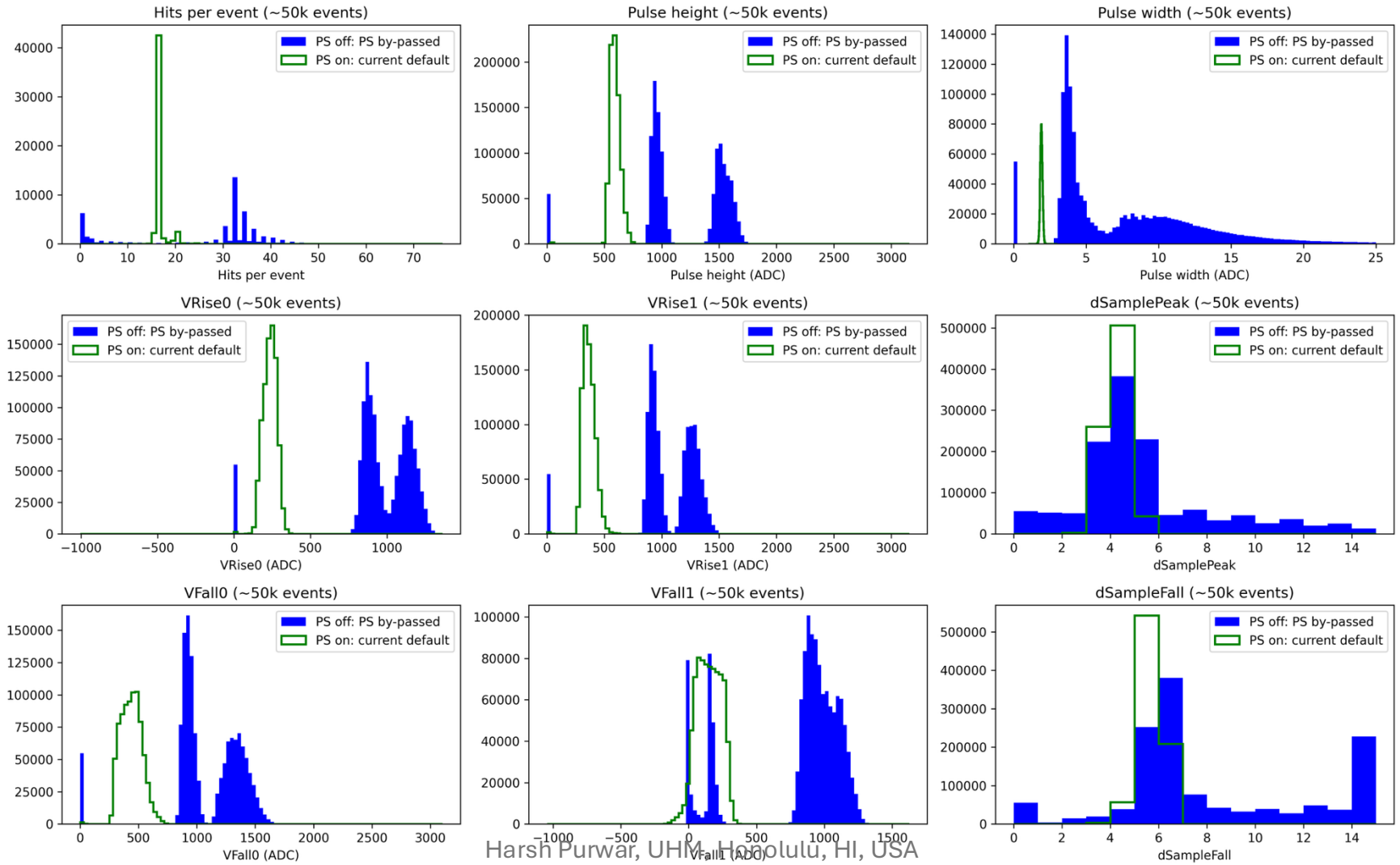
- To not disturb the existing TOP/DAQ setup, for testing TOP FE I made the following changes…
- Minor modifications made in the `pcie40_software` + TOP FE code (v1):
  - Branch: **`top-feature-extraction-kek-v1`**
  - Why?
    - Slightly different RL-9 OS/kernel versions
    - No need to fake the SCROD ID at KEK
    - Have a different IPC socket for testing purposes
- Compiled **`basf2`** on *rtop1* and *rtop2* with minor modification:
  - **`daq/rawdata/modules/src/DeSerializerPC.cc`** – Turn on **`NO_DATA_CHECK`** flag
- Compiled daq_eb (sw event builder) on *rtop1* and *rtop2* with minor modifications:
  - Using IPC socket: **`/tmp/pcie40_roproc_test`** for my tests
  - Similarly, using separate files for eb status/stats
  - Listening port was also modified to 5109

Testing procedure

1. Power-cycle & Configure TOP BS (one or more at a time) with firmware: **`8C-93/84-23`**
2. Prepare TTD with:
   ```
   ttaddr -65 –c; ttaddr -65 –a; ttaddr -65 –m
   ttaddr -65 –u pcie40b,s09a
   ```
3. Enable PS-bypass mode: **`pcie40_regconfig --ch 0 --fee32 -w 0x4EF 0x1`**
4. Start calibration pulser with:
   ```
   ssh topslc01
   ssh pulser bash set5kHz.sh
   ```
5. Start eb0_for_pcie40 with:
   ```
   eb0_for_pcie40 –l 5109 –i 1 –u /dev/shm/eb0_up_test –d /dev/shm/eb0_down_test
   ```
6. Start basf2 for dumping data to file with:
   ```
   basf2 ~/RecvStream1.py –o ~/test.sroot 0 5109 temp
   ```
7. Start DAQ software with: **`pcie40_ulreset; sweb_receiver 0x03000002`**
8. Start issuing triggers with: **`resetft -65; trigft -65 aux 50000`**

Comparison with expected results

Version 1 – No pedestal subtraction



Full system test @KEK – Test01

Version 1

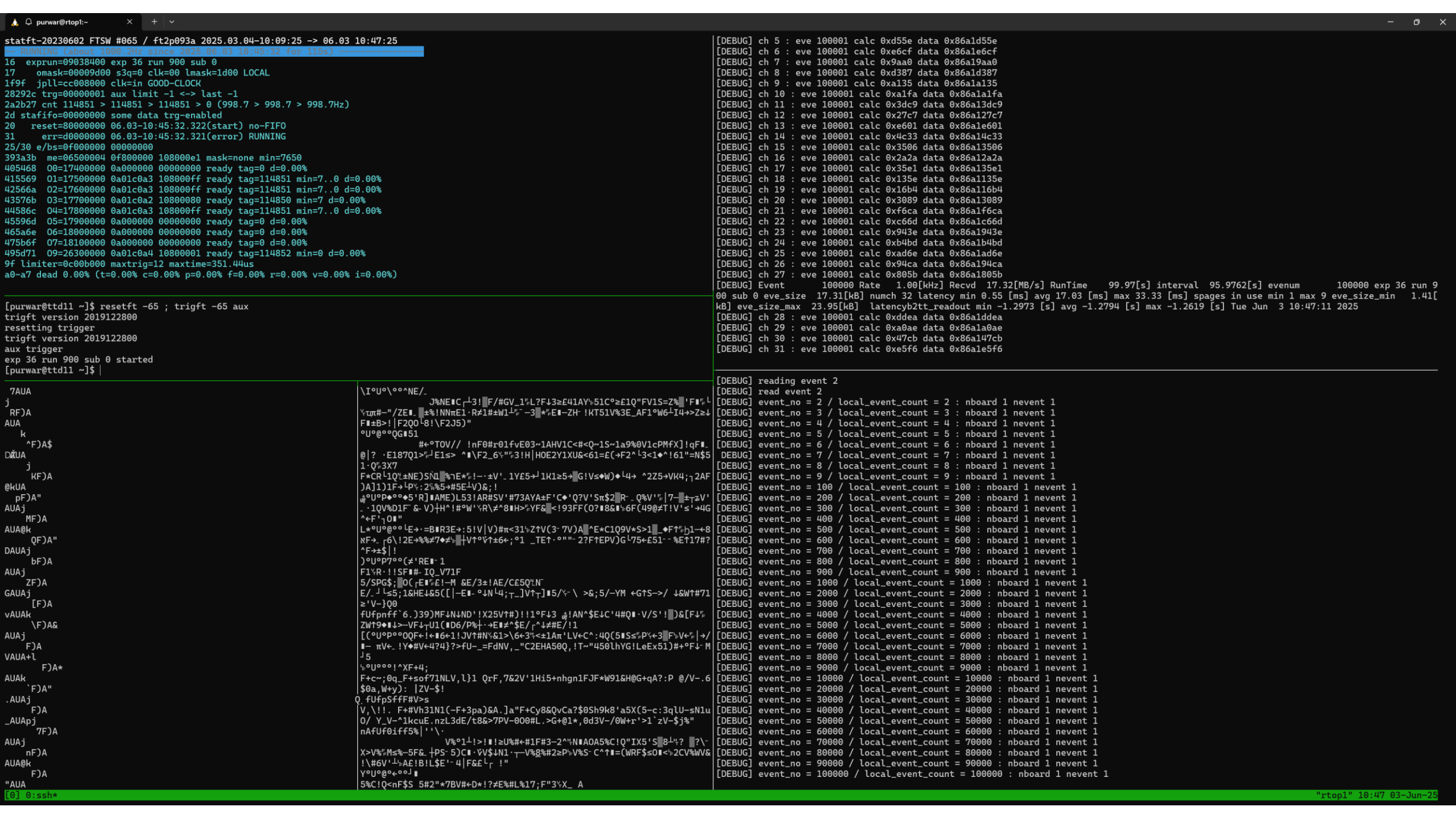
- All TOP FEEs connected to rtop1 – TOP firmware: 8C-93/84-23
- Performance impact with the full setup

Testing procedure

1. Power-cycle & Configure TOP BS (one or more at a time) with firmware: **`8C-93/84-23`**
2. Prepare TTD with:
   ```
   ttaddr -65 –c; ttaddr -65 –a; ttaddr -65 –m
   ttaddr -65 –u pcie40a,s01-s08
   ```
3. Enable PS-bypass mode: **`pcie40_regconfig --ch 0 --fee32 -w 0x4EF 0x1`**
4. Start calibration pulser with:
   ```
   ssh topslc01
   ssh pulser bash set1kHz.sh
   ```
5. Start eb0_for_pcie40 with:
   ```
   eb0_for_pcie40 –l 5109 –i 14 –u /dev/shm/eb0_up_test –d /dev/shm/eb0_down_test
   ```
6. Read in parallel from this port (similar to reading data for sending out to the HLTs):
   ```
   for i in {1..14}; do nc localhost 5109 & done
   ```
7. Start DAQ software with: **`pcie40_ulreset; sweb_receiver 0x03000001`**
8. Start issuing triggers with: **`resetft -65; trigft -65 aux`**

Trigger rate set to 1 kHz

With the Gerard's new pulser at KEK


purwar@rtop1:~
statft-20230602 FTSW #065 / ft2p093a 2025.03.04-10:09:25 -> 06.03 10:47:25
-- RUNNING (about 1000.2Hz since 2025.06.03 10:45:32 for 115s) ----------
16 exprun=09038400 exp 36 run 900 sub 0
17 omask=00009d00 s3q=0 clk=00 lmask=1d00 LOCAL
1f9f jpll=cc008000 clk=in GOOD-CLOCK
28292c trg=00000001 aux limit -1 <-> last -1
2a2b27 cnt 114851 > 114851 > 114851 > 0 (998.7 > 998.7 > 998.7Hz)
2d stafifo=00000000 some data trg-enabled
20 reset=80000000 06.03-10:45:32.322(start) no-FIFO
31 err=d0000000 06.03-10:45:32.321(error) RUNNING
25/30 e/bs=0f000000 00000000
393a3b me=06500004 0f800000 108000e1 mask=none min=7650
405468 O0=17400000 0a000000 00000000 ready tag=0 d=0.00%
415569 O1=17500000 0a01c0a3 108000ff ready tag=114851 min=7..0 d=0.00%
42566a O2=17600000 0a01c0a3 108000ff ready tag=114851 min=7..0 d=0.00%
43576b O3=17700000 0a01c0a2 10800080 ready tag=114850 min=7 d=0.00%
44586c O4=17800000 0a01c0a3 108000ff ready tag=114851 min=7..0 d=0.00%
45596d O5=17900000 0a000000 00000000 ready tag=0 d=0.00%
465a6e O6=18000000 0a000000 00000000 ready tag=0 d=0.00%
475b6f O7=18100000 0a000000 00000000 ready tag=0 d=0.00%
495d71 O9=26300000 0a01c0a4 10800001 ready tag=114852 min=0 d=0.00%
9f limiter=0c00b000 maxtrig=12 maxtime=351.44us
a0-a7 dead 0.00% (t=0.00% c=0.00% p=0.00% f=0.00% r=0.00% v=0.00% i=0.00%)
[DEBUG] ch 5 : eve 100001 calc 0xd55e data 0x86a1d55e
[DEBUG] ch 6 : eve 100001 calc 0xe6cf data 0x86a1e6cf
[DEBUG] ch 7 : eve 100001 calc 0x9aa0 data 0x86a19aa0
[DEBUG] ch 8 : eve 100001 calc 0xd387 data 0x86a1d387
[DEBUG] ch 9 : eve 100001 calc 0xa135 data 0x86a1a135
[DEBUG] ch 10 : eve 100001 calc 0xa1fa data 0x86a1a1fa
[DEBUG] ch 11 : eve 100001 calc 0x3dc9 data 0x86a13dc9
[DEBUG] ch 12 : eve 100001 calc 0x27c7 data 0x86a127c7
[DEBUG] ch 13 : eve 100001 calc 0xe601 data 0x86a1e601
[DEBUG] ch 14 : eve 100001 calc 0x4c33 data 0x86a14c33
[DEBUG] ch 15 : eve 100001 calc 0x3506 data 0x86a13506
[DEBUG] ch 16 : eve 100001 calc 0x2a2a data 0x86a12a2a
[DEBUG] ch 17 : eve 100001 calc 0x35e1 data 0x86a135e1
[DEBUG] ch 18 : eve 100001 calc 0x135e data 0x86a1135e
[DEBUG] ch 19 : eve 100001 calc 0x16b4 data 0x86a116b4
[DEBUG] ch 20 : eve 100001 calc 0x3089 data 0x86a13089
[DEBUG] ch 21 : eve 100001 calc 0xf6ca data 0x86a1f6ca
[DEBUG] ch 22 : eve 100001 calc 0xc66d data 0x86a1c66d
[DEBUG] ch 23 : eve 100001 calc 0x943e data 0x86a1943e
[DEBUG] ch 24 : eve 100001 calc 0xb4bd data 0x86a1b4bd
[DEBUG] ch 25 : eve 100001 calc 0xad6e data 0x86a1ad6e
[DEBUG] ch 26 : eve 100001 calc 0x94ca data 0x86a194ca
[DEBUG] ch 27 : eve 100001 calc 0x805b data 0x86a1805b
[DEBUG] Event 100000 Rate 1.00[kHz] Recvd 17.32[MB/s] RunTime 99.97[s] interval 95.9762[s] evenum 100000 exp 36 run 900 sub 0 eve_size 17.31[kB] numch 32 latency min 0.55 [ms] avg 17.03 [ms] max 33.33 [ms] spages in use min 1 max 9 eve_size_min 1.41[kB] eve_size_max 23.95[kB] latencyb2tt_readout min -1.2973 [s] avg -1.2794 [s] max -1.2619 [s] Tue Jun 3 10:47:11 2025
[DEBUG] ch 28 : eve 100001 calc 0xddea data 0x86a1ddea
[DEBUG] ch 29 : eve 100001 calc 0xa0ae data 0x86a1a0ae
[DEBUG] ch 30 : eve 100001 calc 0x47cb data 0x86a147cb
[DEBUG] ch 31 : eve 100001 calc 0xe5f6 data 0x86a1e5f6
[purwar@ttd11 ~]$ resetft -65 ; trigft -65 aux
trigft version 2019122800
resetting trigger
trigft version 2019122800
aux trigger
exp 36 run 900 sub 0 started
[purwar@ttd11 ~]$
[DEBUG] reading event 2
[DEBUG] read event 2
[DEBUG] event_no = 2 / local_event_count = 2 : nboard 1 nevent 1
[DEBUG] event_no = 3 / local_event_count = 3 : nboard 1 nevent 1
[DEBUG] event_no = 4 / local_event_count = 4 : nboard 1 nevent 1
[DEBUG] event_no = 5 / local_event_count = 5 : nboard 1 nevent 1
[DEBUG] event_no = 6 / local_event_count = 6 : nboard 1 nevent 1
[DEBUG] event_no = 7 / local_event_count = 7 : nboard 1 nevent 1
[DEBUG] event_no = 8 / local_event_count = 8 : nboard 1 nevent 1
[DEBUG] event_no = 9 / local_event_count = 9 : nboard 1 nevent 1
[DEBUG] event_no = 100 / local_event_count = 100 : nboard 1 nevent 1
[DEBUG] event_no = 200 / local_event_count = 200 : nboard 1 nevent 1
[DEBUG] event_no = 300 / local_event_count = 300 : nboard 1 nevent 1
[DEBUG] event_no = 400 / local_event_count = 400 : nboard 1 nevent 1
[DEBUG] event_no = 500 / local_event_count = 500 : nboard 1 nevent 1
[DEBUG] event_no = 600 / local_event_count = 600 : nboard 1 nevent 1
[DEBUG] event_no = 700 / local_event_count = 700 : nboard 1 nevent 1
[DEBUG] event_no = 800 / local_event_count = 800 : nboard 1 nevent 1
[DEBUG] event_no = 900 / local_event_count = 900 : nboard 1 nevent 1
[DEBUG] event_no = 1000 / local_event_count = 1000 : nboard 1 nevent 1
[DEBUG] event_no = 2000 / local_event_count = 2000 : nboard 1 nevent 1
[DEBUG] event_no = 3000 / local_event_count = 3000 : nboard 1 nevent 1
[DEBUG] event_no = 4000 / local_event_count = 4000 : nboard 1 nevent 1
[DEBUG] event_no = 5000 / local_event_count = 5000 : nboard 1 nevent 1
[DEBUG] event_no = 6000 / local_event_count = 6000 : nboard 1 nevent 1
[DEBUG] event_no = 7000 / local_event_count = 7000 : nboard 1 nevent 1
[DEBUG] event_no = 8000 / local_event_count = 8000 : nboard 1 nevent 1
[DEBUG] event_no = 9000 / local_event_count = 9000 : nboard 1 nevent 1
[DEBUG] event_no = 10000 / local_event_count = 10000 : nboard 1 nevent 1
[DEBUG] event_no = 20000 / local_event_count = 20000 : nboard 1 nevent 1
[DEBUG] event_no = 30000 / local_event_count = 30000 : nboard 1 nevent 1
[DEBUG] event_no = 40000 / local_event_count = 40000 : nboard 1 nevent 1
[DEBUG] event_no = 50000 / local_event_count = 50000 : nboard 1 nevent 1
[DEBUG] event_no = 60000 / local_event_count = 60000 : nboard 1 nevent 1
[DEBUG] event_no = 70000 / local_event_count = 70000 : nboard 1 nevent 1
[DEBUG] event_no = 80000 / local_event_count = 80000 : nboard 1 nevent 1
[DEBUG] event_no = 90000 / local_event_count = 90000 : nboard 1 nevent 1
[DEBUG] event_no = 100000 / local_event_count = 100000 : nboard 1 nevent 1
[0] 0:ssh*
"rtop1" 10:47 03-Jun-25

purwar@bdaq:~ | purwar@PurwarTPadX1:~

```
statft-20230602 FTSW #065 / ft2p093a 2025.03.04-10:09:25 -> 05.29 21:52:14
-- RUNNING (about 999.9Hz since 2025.05.29 21:51:38 for 38s) ----------------
16  exprun=09037000 exp 36 run 880 sub 0
17    omask=00009d00 s3q=0 clk=00 tmask=1d00 LOCAL
1f9f  jpll=cc008000 clk=in GOOD-CLOCK
28292c trg=00000001 aux limit 60000 <-> last 21128
2a2b27 cnt 38872 > 38872 > 38872 > 0 (1022.9 > 1022.9 > 1022.9Hz)
2d stafifo=00000000 some data trg-enabled
20   reset=80000000 05.29-21:51:38.035(start) no-FIFO
31     err=d0000000 05.29-21:51:38.035(error) RUNNING
25/30 e/bs=0f000000 00000000
393a3b  me=06500004 0f800000 108000e1 mask=none min=7650
405468  O0=17400000 0a000000 00000000 ready tag=0 d=0.00%
415569  O1=17500000 0a0097d8 108000ff ready tag=38872 min=7..0 d=0.00%
42566a  O2=17600000 0a0097d8 108000ff ready tag=38872 min=7..0 d=0.00%
43576b  O3=17700000 0a0097d8 108000ff ready tag=38872 min=7..0 d=0.00%
44586c  O4=17800000 0a0097d8 108000ff ready tag=38872 min=7..0 d=0.00%
45596d  O5=17900000 0a000000 00000000 ready tag=0 d=0.00%
465a6e  O6=18000000 0a000000 00000000 ready tag=0 d=0.00%
475b6f  O7=18100000 0a000000 00000000 ready tag=0 d=0.00%
495d71  O9=26300000 0a0097d9 10800001 ready tag=38873 min=0 d=0.00%
9f limiter=0c00b000 maxtrig=12 maxtime=351.44us
a0-a7 dead 0.00% (t=0.00% c=0.00% p=0.00% f=0.00% r=0.00% v=0.00% i=0.00%)
```

Works fine now

```
aux trigger
exp 36 run 880 sub 0 started
[purwar@ttd11 ~]$
```

```
=c?/U?)S?=_xFz%0xK
                5g\yNN! 6"VOq^0^|G2vX wO@@]VBVSf3!GFRp&Z`mwFb)p12FrY~&m.,EF`Pn)&_FOfi+        wEqZ
r&s)aFp^'h)i!q)M%UFS}!86AVi12
                          O]5Fd;^!B+WOf)!
?/U?gS?"WF

9
&O
O
O
) 2W"vF#0+N<rW2!@gxWB5PIFZER!` 2j5b*%p/FrxRXGVZZ0">g{E}WNo_[#HzEy]po$"ii5oU`TFyR5^^O
                                                                    eH5B##]VyI.FR`
                                                                                 :
~!B2G!F5&!
?/U?(S?pNFgR!?Wb
`5"|cK 0#\,_V2|}p#@"~QEB!PsA5i>P^`pWuVb=Wp)Frgf/!Y/Wj4+Nn\V)MAqV}iqo ZLmF`A&Tk6
                                                                    00!YF8\N![0*I5QwF!:&*
f5VpH9!@V~!^!
Fng0vIxV;iyc/ xNj5F"wF0MNnCE2j@0L9YWBnP%AIFRlX`xVbm/byXpHL<Er}D5%[6i~\EV(8f|E>S]lFDjNtXVve~O5}^F   ?
>hv=6F&W`V0EIG$
A?/U?
4S?""!/6%A   Wx<'# 6Fn`E"rl     '05\52j@
                              >FBqg#PpyyVR`[Wb3_pG6|ErqqN<(>eF$       w~5F9^N9S^>U~F[y0gFl
Iz\lJ'&\Fr?vZxR5i}b1"-6WiRJ(E_f8
8?/U?NS?9.tV*7p-Fe-Yqm  (JF"vD
ToFV)[F_="'9Qh#hGF_O4V~M%Q`ZFbzp(F<WrmaRwFw`Vt,"!7*FoW]MssWzxQ
y[,Fhhx2
v?/UOS?A \Fi7^!o       6H7D! O<V"G7Z >!0qF2sf76)@n95BmkPajF`0"]zO`E5~7Rf)p2xU5x       pF7Z%x5qN::a`
P~lFRxI`945bpIEJ6:\V!  -0o@S52_@aa     5B_
```

Incoming TOP Feature extracted data

```
data 1498 : 21464e6c c35bb800 03ef0000 03ad03b0 03c100a5 22ad5244 d15bbd00 05e00000
data 14a0 : 03fe04e6 050303d8 215738ad d35bbb00 03d20000 03800398 039c0350 10bf4f0e
data 14a8 : e15bb700 06060000 04b204f2 050f041e 22462c86 e35bbb00 03c60000 0357039e
data 14b0 : 038f0382 208c2f4b f15bb100 05950000 0470050d 04fe041e 23352240 f35bb400
data 14b8 : 03860000 034f0386 0369034f 1115367b 00000a26 30399b8a ff550000 ffaa1f39
data 14c0 : 2e48c280 0401a061 000000a6 06aa2735 00000139 015bbc00 05d90000 047304c0
data 14c8 : 05320429 244605f7 035bb500 03b10000 037f039c 0398036e 21891449 115bbe00
data 14d0 : 05cb0000 040204a1 05650478 2546f312 135bb400 03bc0000 0368039b 03990372
data 14d8 : 3179f560 215bb000 05d60000 04980561 04e903dd 2535f0d9 235bb400 03b00000
data 14e0 : 036f0386 03a000a5 2758f261 315bb400 06220000 04d50518 0506045d 2446dceb
data 14e8 : 335bb200 03dd0000 039603c6 03b100a5 213ee9d6 415bb500 061f0000 049e0530
data 14f0 : 05030447 2346cd26 435bb200 03f10000 03b003e7 03d700a5 235cd743 515bb200
data 14f8 : 05a80000 047004f1 05950483 2535be4d 535bb300 03b90000 03800398 038800a5
data 1500 : 239cc709 615bb200 06380000 04d10597 05010420 2536adac 635bb000 03f00000
data 1508 : 03af03b1 03c300a5 31deab0d 715bbf00 05bf0000 0406048d 04b9041a 23579527
data 1510 : 735bbc00 03b20000 0360038e 039600a5 204fa179 815bb000 060d0000 04b90579
data 1518 : 059e0470 25358f21 835bbe00 03f80000 039a03b2 03ce039f 104c9ba6 915bba00
data 1520 : 05e60000 043e04e6 04a803e3 255777b7 935bb100 03ca0000 038703ca 03a60389
data 1528 : 29148045 a15bb200 06020000 04a2054c 04e903fa 25466e8a a35bb900 03f10000
data 1530 : 039c03be 03ce03a8 102580bd b15bb200 06180000 04b8054c 059b046e 25355d49
data 1538 : b35bbb00 03e20000 038603b3 03c600a5 219e607f c15bb500 06280000 04b20549
data 1540 : 04ee043f 25464b0d c35bbc00 03ff0000 039a03c9 03c1037d 102f5dd4 d15bb700
data 1548 : 063f0000 04990548 0531042a 253638f2 d35bb300 03f90000 03b603e7 03d803b9
data 1550 : 11375545 e15bbf00 05f60000 049904c6 05610452 24462255 e35bb300 03cf0000
data 1558 : 039703cf 03ae037c 2115362f f15bb200 05f60000 04b804fb 055d0476 24461ee1
data 1560 : f35bb500 03de0000 039d03ba 03c003a0 113433da 00000a20 303970b2 ff550000
data 1568 : 00000000 00000000 00000000 7fff0006 00000000 7fff0007
printData2() : Done. : # of words : 5486
```

DAQ software

```
[DEBUG] event_no = 4 / local_event_count = 4 : nboard 1 nevent 1
[DEBUG] event_no = 5 / local_event_count = 5 : nboard 1 nevent 1
[DEBUG] event_no = 6 / local_event_count = 6 : nboard 1 nevent 1
[DEBUG] event_no = 7 / local_event_count = 7 : nboard 1 nevent 1
[DEBUG] event_no = 8 / local_event_count = 8 : nboard 1 nevent 1
[DEBUG] event_no = 9 / local_event_count = 9 : nboard 1 nevent 1
[DEBUG] event_no = 100 / local_event_count = 100 : nboard 1 nevent 1
[DEBUG] event_no = 200 / local_event_count = 200 : nboard 1 nevent 1
[DEBUG] event_no = 300 / local_event_count = 300 : nboard 1 nevent 1
[DEBUG] event_no = 400 / local_event_count = 400 : nboard 1 nevent 1
[DEBUG] event_no = 500 / local_event_count = 500 : nboard 1 nevent 1
[DEBUG] event_no = 600 / local_event_count = 600 : nboard 1 nevent 1
[DEBUG] event_no = 700 / local_event_count = 700 : nboard 1 nevent 1
[DEBUG] event_no = 800 / local_event_count = 800 : nboard 1 nevent 1
[DEBUG] event_no = 900 / local_event_count = 900 : nboard 1 nevent 1
[DEBUG] event_no = 1000 / local_event_count = 1000 : nboard 1 nevent 1
[DEBUG] event_no = 2000 / local_event_count = 2000 : nboard 1 nevent 1
[DEBUG] event_no = 3000 / local_event_count = 3000 : nboard 1 nevent 1
[DEBUG] event_no = 4000 / local_event_count = 4000 : nboard 1 nevent 1
[DEBUG] event_no = 5000 / local_event_count = 5000 : nboard 1 nevent 1
[DEBUG] event_no = 6000 / local_event_count = 6000 : nboard 1 nevent 1
[DEBUG] event_no = 7000 / local_event_count = 7000 : nboard 1 nevent 1
[DEBUG] event_no = 8000 / local_event_count = 8000 : nboard 1 nevent 1
[DEBUG] event_no = 9000 / local_event_count = 9000 : nboard 1 nevent 1
[DEBUG] event_no = 10000 / local_event_count = 10000 : nboard 1 nevent 1
[DEBUG] event_no = 20000 / local_event_count = 20000 : nboard 1 nevent 1
[DEBUG] event_no = 30000 / local_event_count = 30000 : nboard 1 nevent 1
```

Software EB

```
[topEx] 0:ssh- 1:ssh*                                          "PurwarTPadX1" 02:52 29-May-25
```

TOP Pedestal data RO

Using sweb_receiver

-- Successful after Vasily’s firmware modifications at UH

-- Running tests at KEK now.

Reading pedestal data from TOP FEE

at UH Test Bench

- Power-cycle TOP BS with firmware: *8C-93/86-23*
- Configure it. This reads in new pedestals and saves it to the SCROD memory
- Enable sending pedestal data from SCROD memory to PCIe40 via b2link:

```
pcie40_regconfig --ch ___ --fee32 –w 0x182D 0x4
```

- Prepare PCIe40 to read data with *ID for TOP 0x03000001*

```
sweb_receiver 0x03000001
```

- Start software event builder with,

```
eb0+1tx_for_pcie40 -l 5101 -i 1
```

- Start basf2 to incoming read data from the IPC port (5101)

```
basf2 RecvPeds2Root.py -o testPed.sroot 0 5101 temp
```

- Send FTSW triggers (local): *num trig out > 8192*

```
trigft -13 pulse 2000 8400
```

Combined all these steps into a single bash script

Plan to integrate this in TOP Power-cycle and Config GUI

Reading pedestal data from TOP FEE

at KEK

- Power-cycle TOP BS with firmware: *8C-93/86-23*
- Configure it. This reads in new pedestals and saves it to the SCROD memory
- Enable sending pedestal data from SCROD memory to PCIe40 via b2link:

```
pcie40_regconfig --ch ___ --fee32 –w 0x182D 0x4
```

- Configure TTD using ttaddr and nsm commands.
- Load and then start a new local run.
- Copy file back to rtop* from store (HLT).
- Unpack and extract pedestal values.

Combining all these steps into a single bash script

Plan to integrate this in TOP Power-cycle and Config GUI

New unpacker for pedestal data

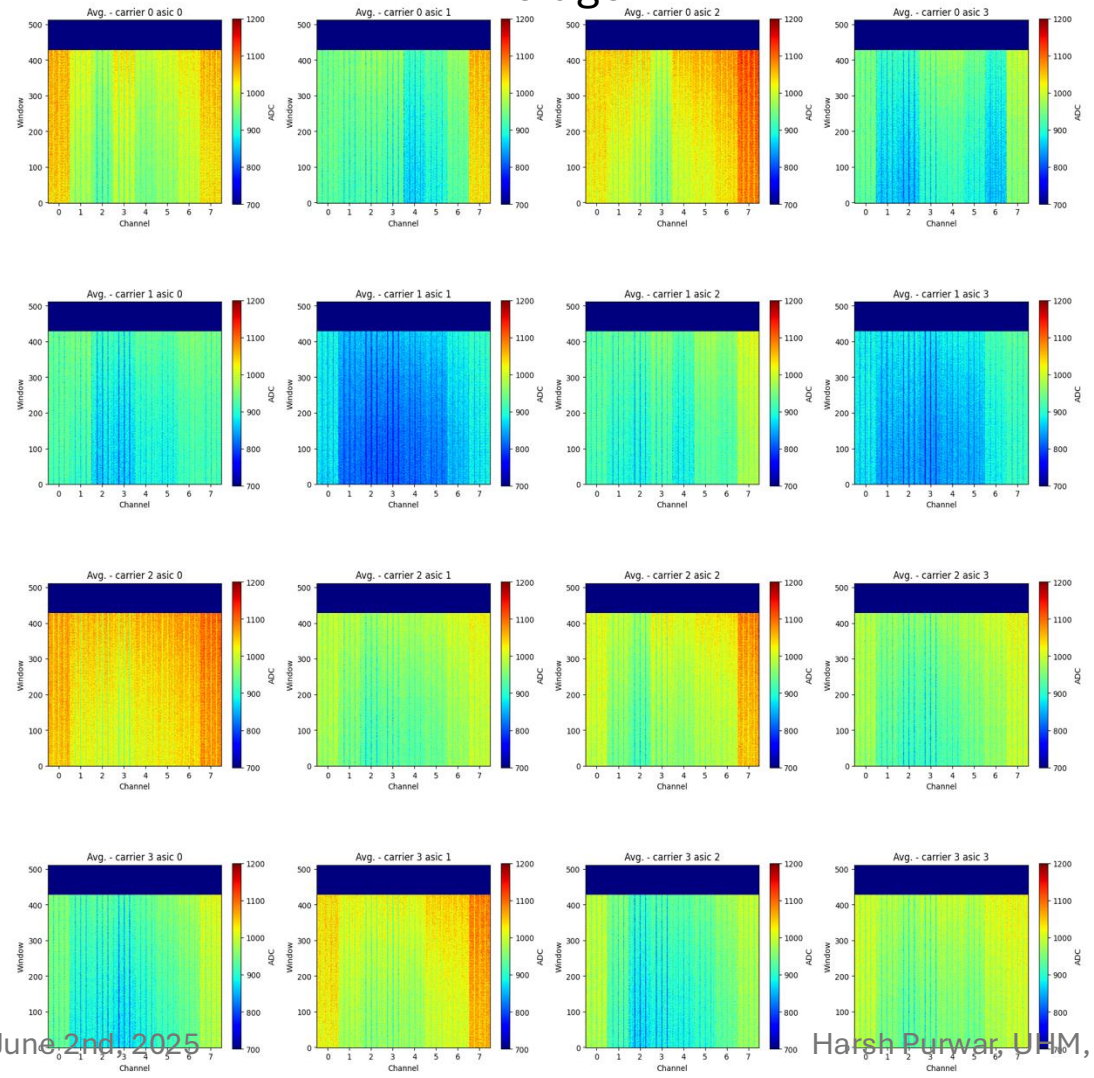
- Previously the pedestal data used to be unpacked with the peddump.c code
- This expects the file has only the B2Link header, footer and pedestal data.
- Don’t know how to remove PCIe40 header/footer from within basf2, so instead **I wrote a small unpacker for pedestal data in python**, works fine.
- Steps (1-3 included in the bash script):
  1. Read pedestals from TOP FEEs and dump them in a root file.
  2. Then convert root to binary file/format.
  3. Run python unpacker to unpack and dump these into a txt & binary file.
  4. Use one of the 2 files to read pedestals and do feature extraction.

- Running with multiple BS have some inconsistent behavior, will try to debug and fix this.
- Also, need to slightly modify the python unpacker to correctly unpack pedestal data from multiple FEEs.

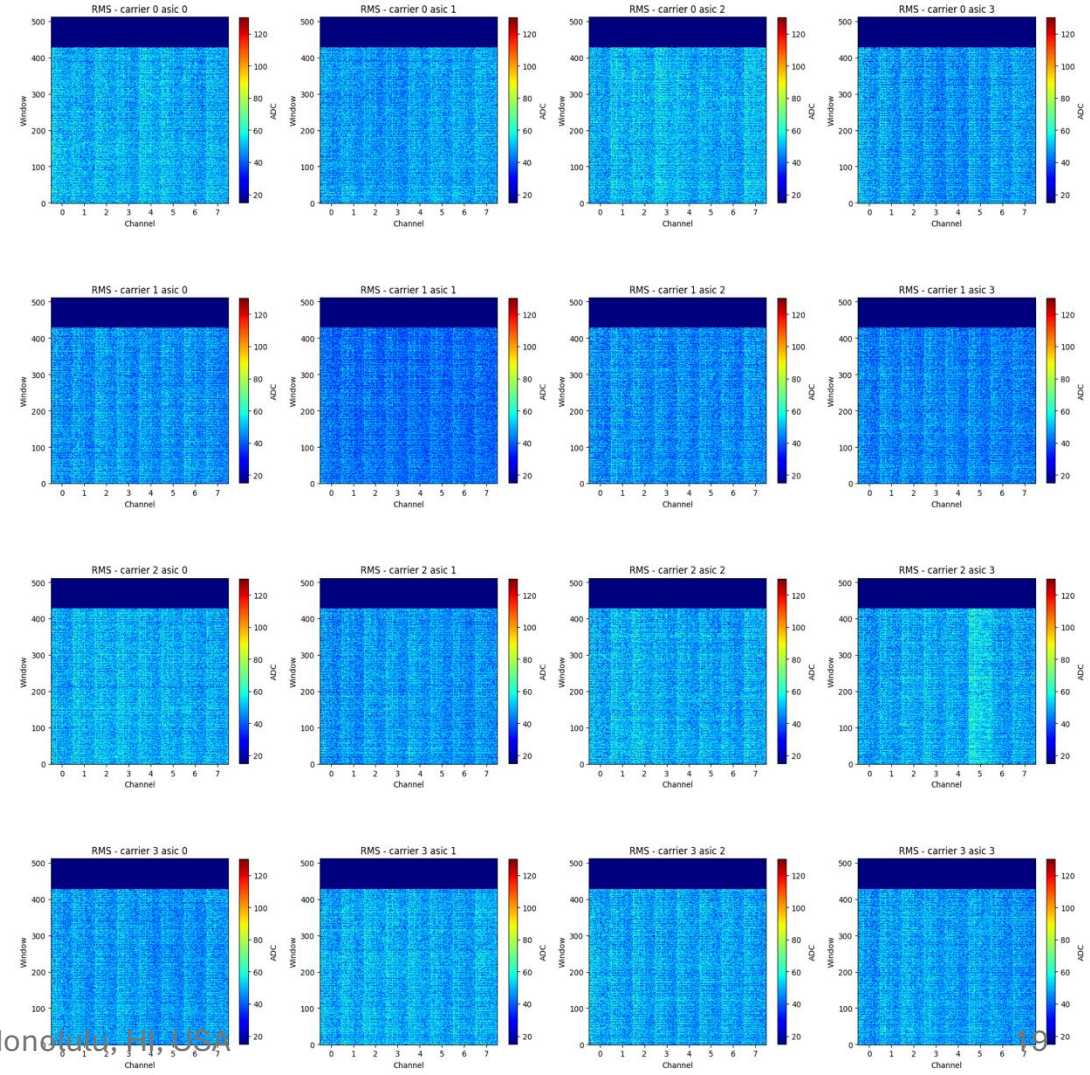
Pedestal values for BS-3 (UH)

BS-3 → ch 7 → SCORD ID: 7

Average



RMS

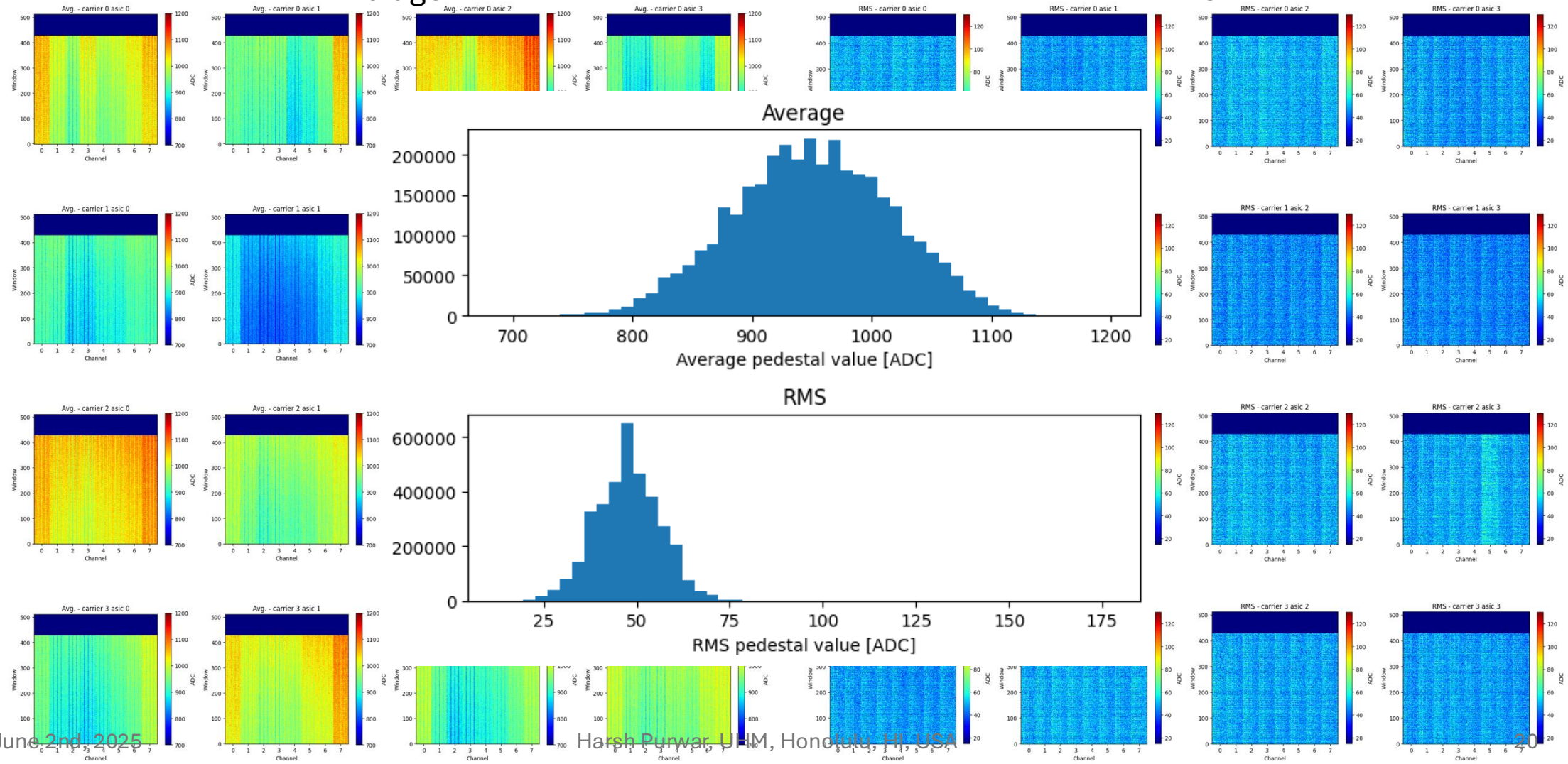


Pedestal values for BS-3 (UH)

BS-3 → ch 7 → SCORD ID: 7

Average

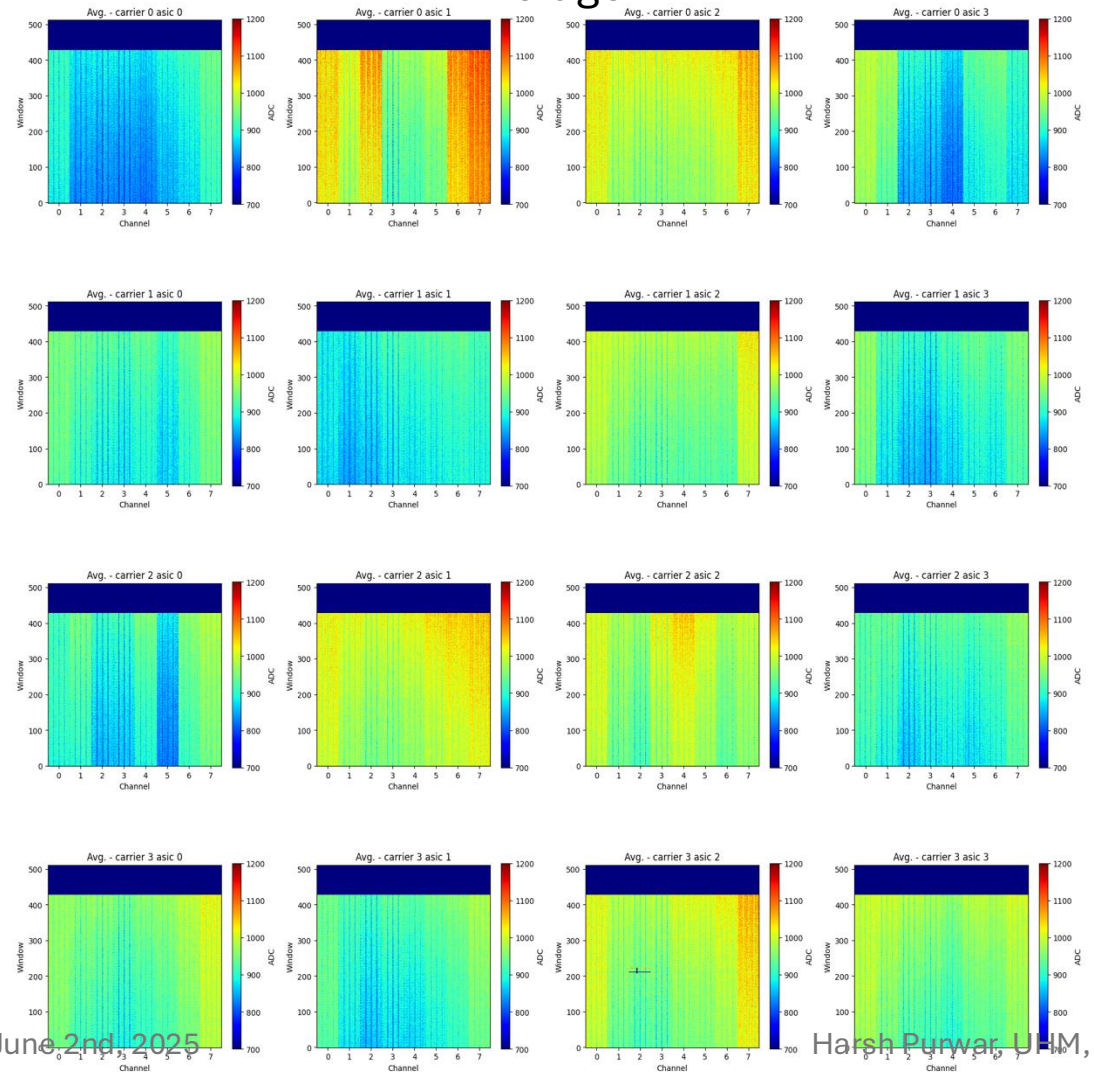
RMS



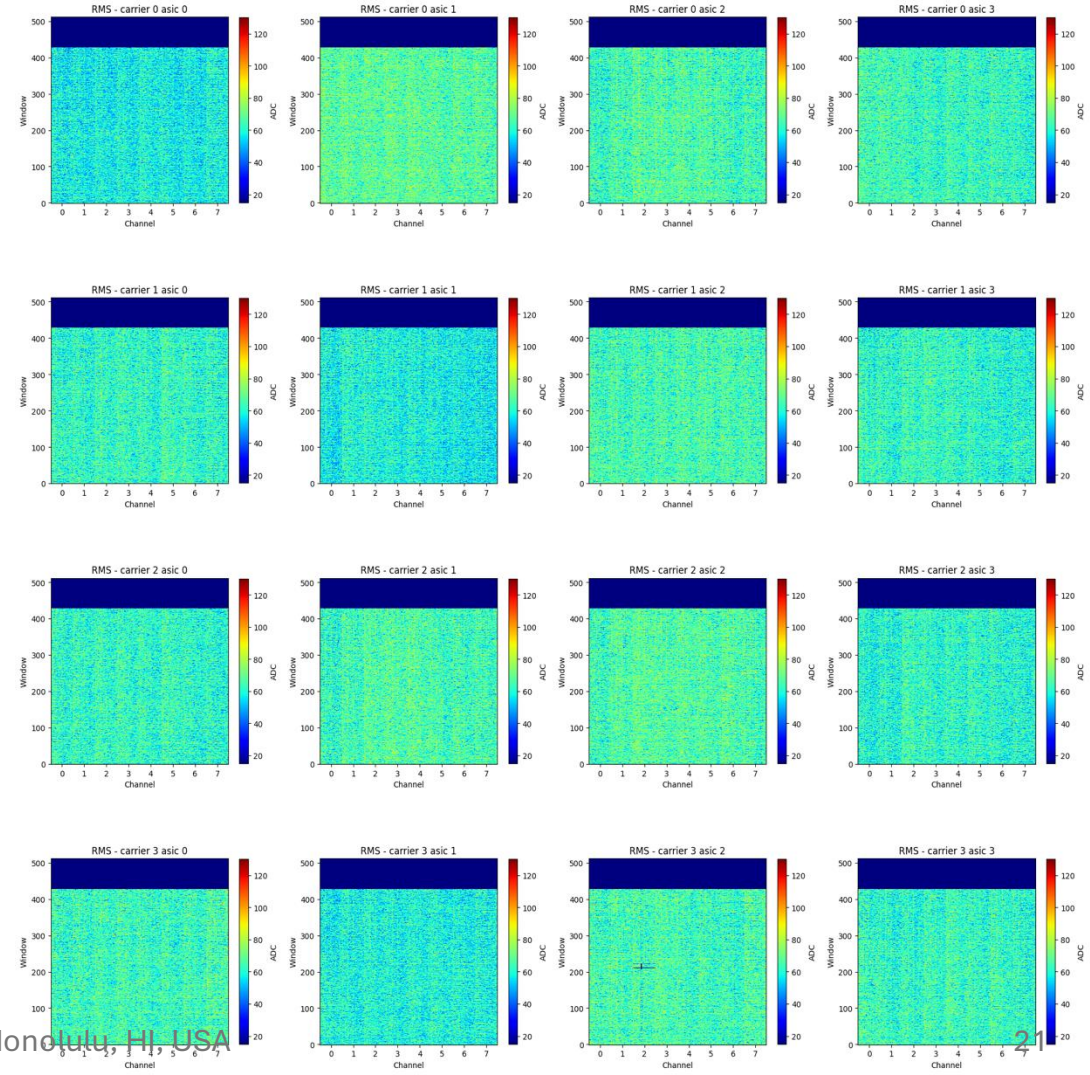
Pedestal values for BS-5 (UH)

BS-5 → ch 11 → SCORD ID: 100

Average



RMS

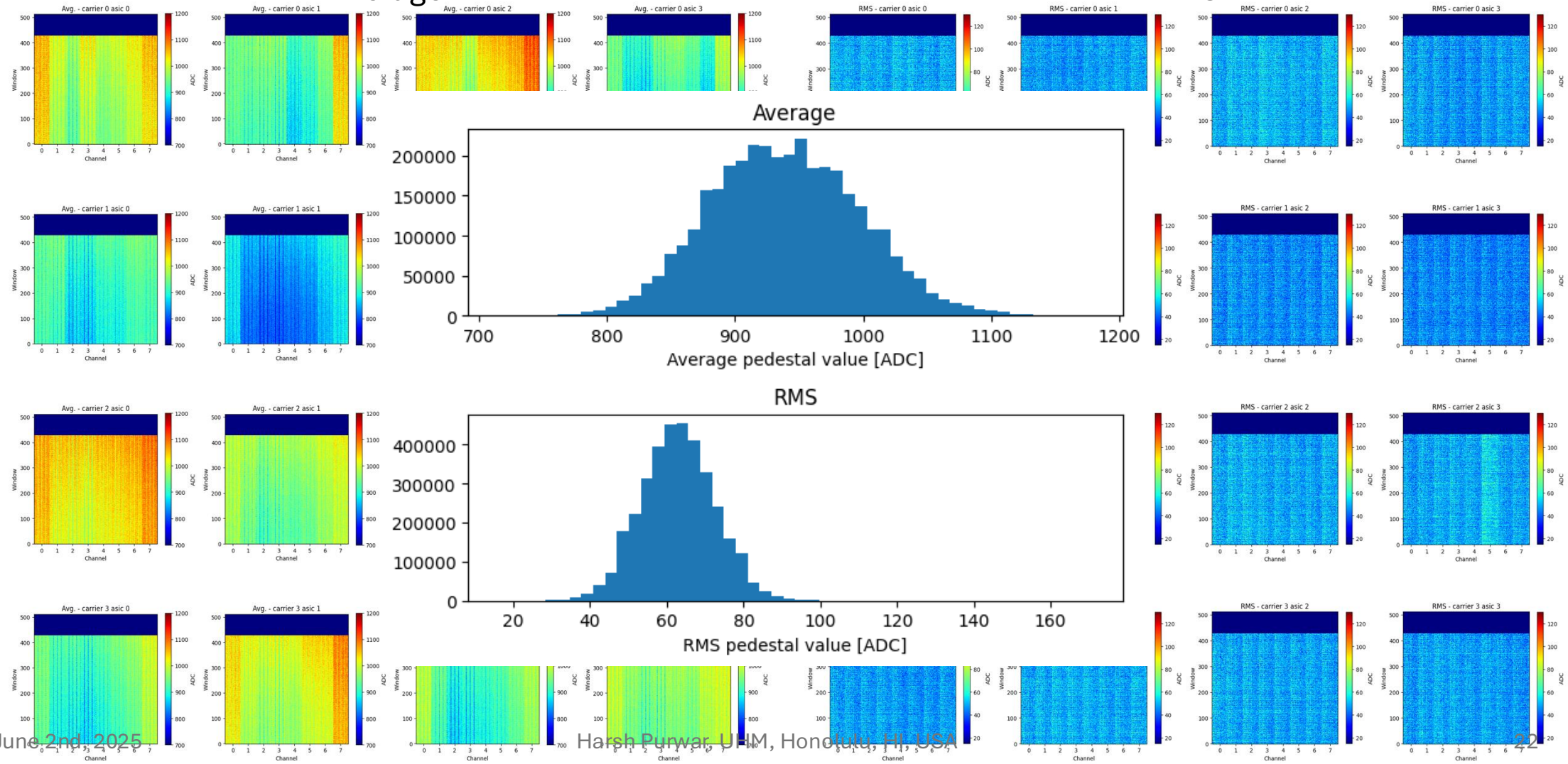


Pedestal values for BS-5 (UH)

BS-5 → ch 11 → SCORD ID: 100

Average

RMS



TT_TAG Error while reading pedestals

```
 data 200 : 03610366 03620353 037c0380 03650377 036d0375 035b037a 0376037c 03650375
 data 208 : 03450351 0352034c 03660356 037f037a 03700358 037a0381 03670368 0377037a
 data 210 : 03790356 03760382 037c037c 0385039a 03840391 0378037a 03810376 037b0373
 data 218 : 037a035a 039703a5 037f0384 03750380 03780385 036f0372 03630381 03910393
 data 220 : 036b035e 0358035d 0365035e 036e0373 035b0378 036a0373 036b0389 036a0376
 data 228 : 03520331 036c035e 0382037e 03700381 03810369 0395038d 03930395 03820386
 data 230 : 03520331 035d0364 0360034d 036e037c 036d0371 03820397 03830385 036b036b
 data 238 : 0384036c 0398038a 037f0382 039c0380 03930396 0382037b 036d0383 03740380
 data 240 : 00011644 ff550000 00000000 00000000 00000000 7fff0006 00000000 7fff0007
 data 248 :
printData2() : Done. : # of words : 584
[DEBUG] Event           2 Rate   2.09[kHz] Recvd   4.89[MB/s] RunTime     0.00[s] interval   0.0005[s] evenum           2 exp 0 run 7 sub 0 eve_size   2.34[kB] numch 1 latency min 0.96 [ms] avg 0.
96 [ms] max 0.96 [ms] spages in use min 1 max 1 eve_size_min   2.06[kB] eve_size_max   2.06[kB]  latencyb2tt_readout min 500.0000 [s] avg 15091.9797 [s] max 15091.9797 [s] Thu May  8 10:58:03 2025
[DEBUG] Event           3 Rate  35.85[kHz] Recvd  83.74[MB/s] RunTime     0.00[s] interval   0.0000[s] evenum           3 exp 0 run 7 sub 0 eve_size   2.34[kB] numch 1 latency min 0.96 [ms] avg 0.
98 [ms] max 0.98 [ms] spages in use min 1 max 1 eve_size_min   2.06[kB] eve_size_max   2.06[kB]  latencyb2tt_readout min 500.0000 [s] avg 15091.9793 [s] max 15091.9793 [s] Thu May  8 10:58:03 2025
[DEBUG] Event           4 Rate  34.38[kHz] Recvd  80.31[MB/s] RunTime     0.00[s] interval   0.0000[s] evenum           4 exp 0 run 7 sub 0 eve_size   2.34[kB] numch 1 latency min 0.96 [ms] avg 1.
03 [ms] max 1.03 [ms] spages in use min 1 max 1 eve_size_min   2.06[kB] eve_size_max   2.06[kB]  latencyb2tt_readout min 500.0000 [s] avg 15091.9788 [s] max 15091.9788 [s] Thu May  8 10:58:03 2025
[DEBUG] Event         100 Rate 561.58[kHz] Recvd 1311.85[MB/s] RunTime     0.00[s] interval   0.0002[s] evenum         100 exp 0 run 7 sub 0 eve_size   2.34[kB] numch 1 latency min 0.96 [ms] avg 1
.15 [ms] max 1.33 [ms] spages in use min 1 max 1 eve_size_min   2.06[kB] eve_size_max   2.06[kB]  latencyb2tt_readout min 500.0000 [s] avg 15091.9548 [s] max 15091.9783 [s] Thu May  8 10:58:03 2025
[DEBUG] Event         200 Rate   1.61[kHz] Recvd   3.77[MB/s] RunTime     0.06[s] interval   0.0001[s] evenum         200 exp 0 run 7 sub 0 eve_size   2.34[kB] numch 1 latency min 0.10 [ms] avg 3.
03 [ms] max 63.67 [ms] spages in use min 1 max 2 eve_size_min   2.06[kB] eve_size_max   2.06[kB]  latencyb2tt_readout min 500.0000 [s] avg 15091.9542 [s] max 15091.9816 [s] Thu May  8 10:58:03 2025
[DEBUG] Event         300 Rate   1.57[kHz] Recvd   3.67[MB/s] RunTime     0.13[s] interval   0.0637[s] evenum         300 exp 0 run 7 sub 0 eve_size   2.34[kB] numch 1 latency min 0.10 [ms] avg 5.
36 [ms] max 63.79 [ms] spages in use min 1 max 2 eve_size_min   2.06[kB] eve_size_max   2.06[kB]  latencyb2tt_readout min 500.0000 [s] avg 15091.9526 [s] max 15091.9836 [s] Thu May  8 10:58:03 2025
[DEBUG] Event         400 Rate   1.57[kHz] Recvd   3.66[MB/s] RunTime     0.19[s] interval   0.0639[s] evenum         400 exp 0 run 7 sub 0 eve_size   2.34[kB] numch 1 latency min 0.10 [ms] avg 7.
93 [ms] max 63.91 [ms] spages in use min 1 max 2 eve_size_min   2.06[kB] eve_size_max   2.06[kB]  latencyb2tt_readout min 500.0000 [s] avg 15091.9512 [s] max 15091.9856 [s] Thu May  8 10:58:03 2025
[DEBUG] Event        1000 Rate   2.34[kHz] Recvd   5.47[MB/s] RunTime     0.45[s] interval   0.2561[s] evenum        1000 exp 0 run 7 sub 0 eve_size   2.34[kB] numch 1 latency min 0.10 [ms] avg 9.
60 [ms] max 63.95 [ms] spages in use min 1 max 2 eve_size_min   2.06[kB] eve_size_max   2.06[kB]  latencyb2tt_readout min 500.0000 [s] avg 15091.9578 [s] max 15091.9936 [s] Thu May  8 10:58:03 2025
[DEBUG] Event        2000 Rate   1.96[kHz] Recvd   4.57[MB/s] RunTime     0.96[s] interval   0.5113[s] evenum        2000 exp 0 run 7 sub 0 eve_size   2.34[kB] numch 1 latency min 0.09 [ms] avg 7.
88 [ms] max 63.91 [ms] spages in use min 1 max 2 eve_size_min   2.06[kB] eve_size_max   2.06[kB]  latencyb2tt_readout min 500.0000 [s] avg 15091.9562 [s] max 15091.9941 [s] Thu May  8 10:58:04 2025
[DEBUG] Event        3000 Rate   1.96[kHz] Recvd   4.57[MB/s] RunTime     1.47[s] interval   0.5115[s] evenum        3000 exp 0 run 7 sub 0 eve_size   2.34[kB] numch 1 latency min 0.09 [ms] avg 8.
41 [ms] max 63.91 [ms] spages in use min 1 max 2 eve_size_min   2.06[kB] eve_size_max   2.06[kB]  latencyb2tt_readout min 500.0000 [s] avg 15091.9564 [s] max 15091.9946 [s] Thu May  8 10:58:04 2025
[DEBUG] Event        4000 Rate   1.96[kHz] Recvd   4.57[MB/s] RunTime     1.98[s] interval   0.5112[s] evenum        4000 exp 0 run 7 sub 0 eve_size   2.34[kB] numch 1 latency min 0.06 [ms] avg 6.
92 [ms] max 63.95 [ms] spages in use min 1 max 2 eve_size_min   2.06[kB] eve_size_max   2.06[kB]  latencyb2tt_readout min 500.0000 [s] avg 15091.9546 [s] max 15091.9931 [s] Thu May  8 10:58:05 2025
Subevent: link 11: TT_TAG error flag raised

                                                    8192
 data   0 :  00000004  EB40000B  00800007  00380000  00000000  00002000  00000204  00002000
 data   1 :  FFAA0B00  FFFFFFFF  FFFFFFFF  FFFFFFF2  00000700  25B662A0  0401A064  00000005   Regular TOP data
 data   2 :  012904B4  00000040  00000A0C  FFFFFFFF  FFFF4E22  FF550204  00000000  00000000
 data   3 :  00380002  0000000B  42424242  00000000  00000000  00002000  00000700  00002000

[FATAL] daqupsvr ch=11 : Mismatch between TT tag in data and in chunk DMA headers. Exiting..: /home/purwar/software/Pcie40Applications/subevent.cpp void Subevent::assembleSubEvent() 189
Printing chunks : link 11 : ...
 data 000 : ffaa0b00 ffffffff ffffffff fffffff2 00000700 25b662a0 0401a064 00000005
 data 007 : 012904b4 00000040 00000a0c ffffffff ffff4e22 ff550204
Next chunk
```

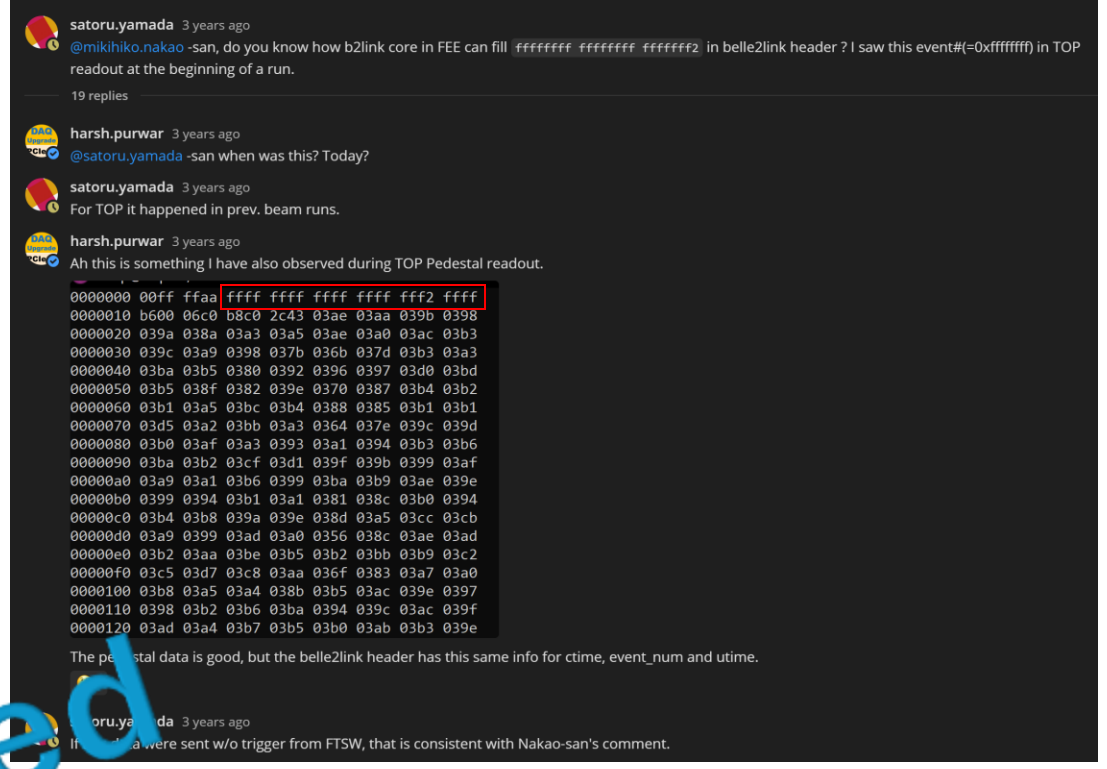
Resolved

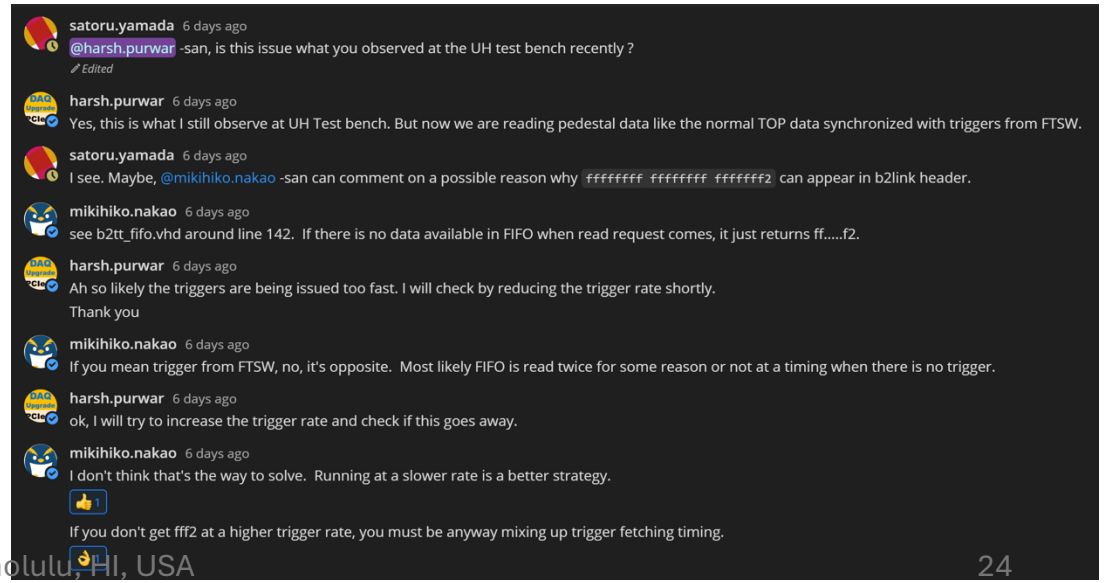
TT_Tag Error

First observed 3 years ago!

- Few years back when I was trying to read pedestals using sw triggers, I observed this issue during pedestal data RO & Yamada-san as well during his DAQ tests at KEK.
- After discussion with Nakao-san, we agreed that this is because there weren't any triggers issued by the FTSW (sw triggers are generated in the SCROD).
- But after recent changes to TOP firmware (B2G 93/84-23), we now can read pedestals synced with the triggers from FTSW.
- While doing so, I was getting inconsistent behavior, my script to read pedestals worked sometimes and failed sometimes.

Resolved





TT_Tag error

- Turns out it was because of this issue where ctime, utime, & event# are incorrect (`ffff...`) – DAQ sw raises a **TT_Tag Error.**
- Note the trigger/event number: 0x2000 = 8192 (pedestal data is from 0 – 8191)
- Also, look at the TOP data – It is regular TOP data.
- Looks like we are switching from pedestal RO mode to regular data readout mode automatically after sending 8192 pedestal data events.
- ***Vasily, could there be a bug in the TOP firmware, since this functionality was recently added?***

DAQ software – sweb_receiver

```
Subevent: link 11: TT_TAG error flag raised
                                                                    8192
data   0 :  00000004  EB40000B  00800007  00380000  00000000  00002000  00000204  00002000
data   1 :  FFAA0B00  FFFFFFFF  FFFFFFFF  FFFFFFF2  00002700  6A8B1B20  0401A064  00000005
data   2 :  02850710  000000A0  00000A04  FFFFFFFF  FFFF48A6  FF550204  00000000  00000000
data   3 :  00380002  0000000B  42424242  00000000  00000000  00002000  00002700  00002000

[FATAL] daqupsvr ch=11 : Mismatch between TT tag in data and in chunk DMA headers. Exiting..
: /home/purwar/software/Pcie40Applications/subevent.cpp void Subevent::assembleSubEvent() 189
Printing chunks : link 11 : ...                                          Regular TOP data
data 000 : ffaa0b00 ffffffff ffffffff fffffff2 00002700 6a8b1b20 0401a064 00000005
data 007 : 02850710 000000a0 00000a04 ffffffff ffff48a6 ff550204
```

Status of TOP FTSW #13

```
sta  0210921 FTSW #013 / ft2p094a 2025.05.27-07:09:31 -> 05.28 11:47:50
--- USY ----------------------------------------------------------------------
16  xp  n=00002b00 exp 0 run 43 sub 0
    omask=000098ff s3q=0 clk=00 lmask=18ff LOCAL
1f9f  jpll=cc008000 clk=in GOOD-CLOCK
28292c trg=00021034 pulse 1000.590 Hz 528e3 limit -1 <-> last -1
2a2b27 cnt 35662 > 35662 > 8788 > 8789 (990.6 > 990.6 > 244.1Hz)
2d stafifo=00000000 some data trg-enabled
20   reset=80000000 05.28-11:47:38.547(start) no-FIFO
31     err=d0000000 05.28-11:47:38.541(error) RUNNING
25/30 e/bs=0f000000 c0000100
393a3b  me=01300004 0f800000 10800100 BUSY mask=none min=8
485c70  O8=03040400 0a002255 00000000 BUSY ready tag=8789 d=75.36%
4a5e72  O10=01310000 0a002354 0a002354 ready tag=9044 d=0.00%
9f limiter=0c00b000 maxtrig=12 maxtime=351.44us
a0-a7 dead 75.36% (t=75.36% c=0.00% p=0.00% f=0.00% r=0.00% v=0.00% i=0.00%)
```

PCIe40 TOP FE

Resolved

Thank you for your attention.

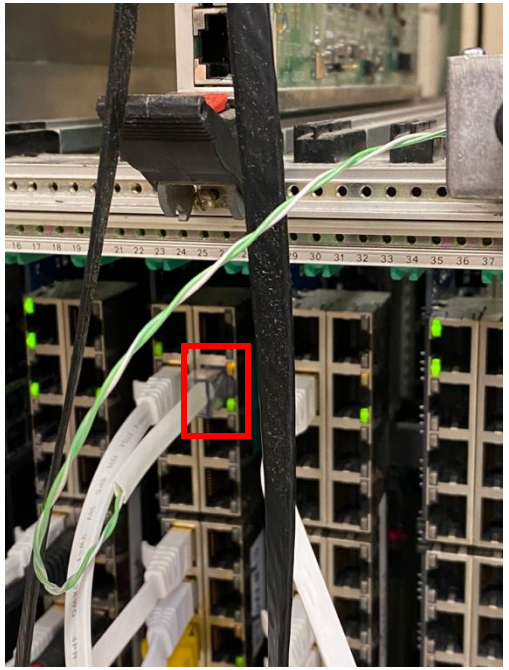
Any questions/comments?

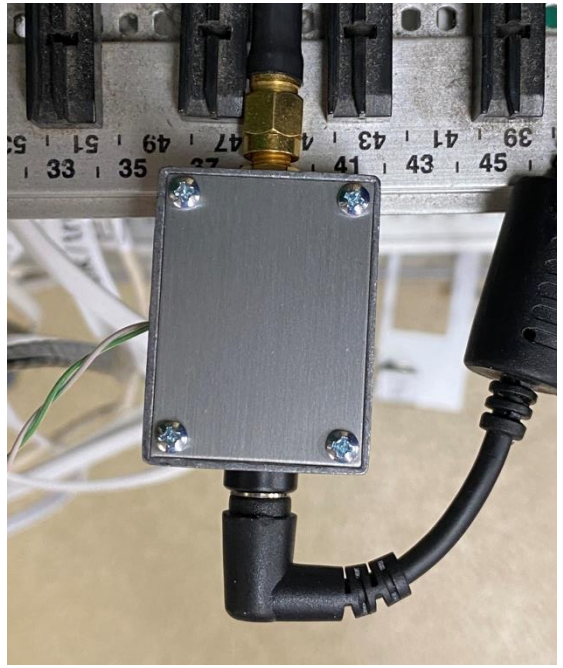
Backup slides

Upgrade of TOP Test Bench at Varner Lab

- Until now, the fast pulser was not synchronized with the FTSW (or triggers)
- This injected pulses randomly – provided a more realistic scenario

- After the upgrade, we can now sync pulser and FTSW triggers, and this ensures hits in every event
- We could still inject pulses randomly
- Current default is ~25% occupancy (hits in 2 out of 8 channels)
- Possible to run at full occupancy as well.





B2L, HSLB, COPPER/PCIe40 data formats

New data format for RawCOPPER hdr. and trl. (ver. alpha-200809)

- Remove COPPER header/trailer (which is currently removed on a readout PC.)
- Position in header will be removed to reduce header size.
- Error information in trailer (which link, what kind of error)
- XOR checksum (less CPU power to re-calculate on host servers or HLT)

1 Number of total words
2 0x7f7f | Format ver.(8bit) | Number of words in this block (8bit)
3 exp no. (10bit=1024), run no.(22bit=4194304 including subrun)
4 event number(32bit)
5 From B2link FEE header 2 (TT-ctime | Trig-type)
6 From B2link FEE header 2 (TT-utime)
7 Node ID
8 b2link CRC error bit (4) | |truncation mask (truncated or not) / type of data (compressed,calibration, …)
9 Position of ch0 data
10 Posiont of ch1 data
…
56 Position of ch48 data

1word=32bits

RawCopper Header (56words)

Detector buffer(slot 0)

Detector buffer(slot 1)

B2link (FEE/HSLB) header and trailer

Detector buffer(slot 47)

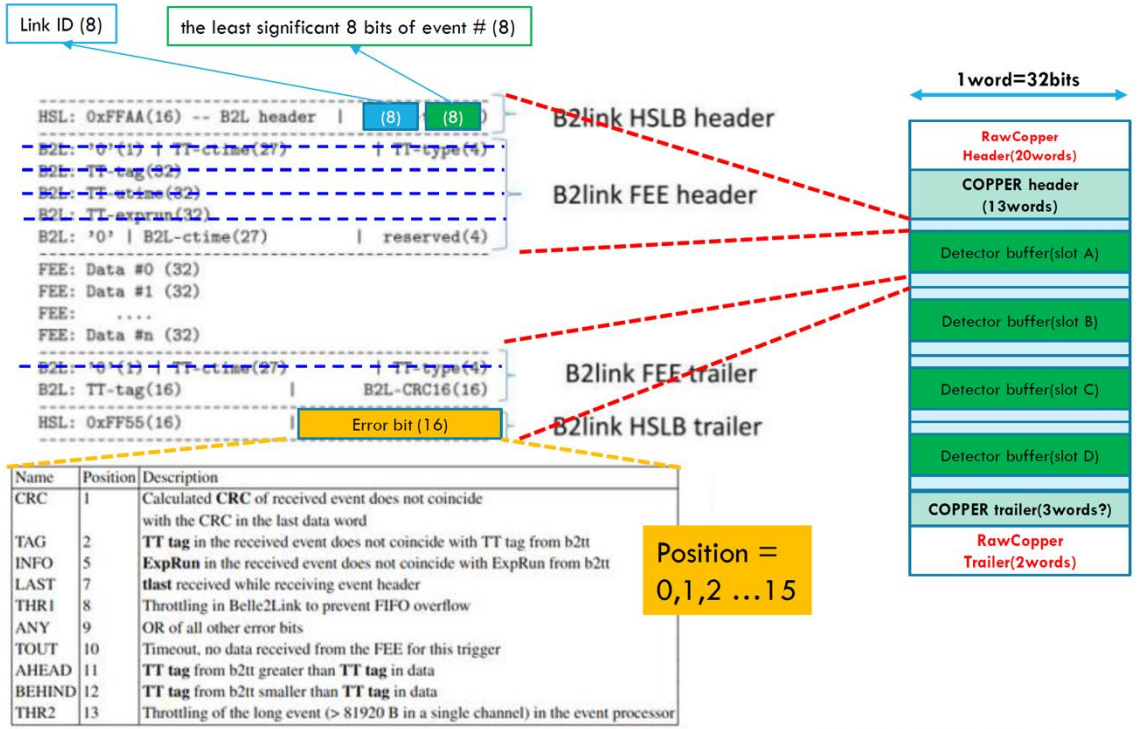
RawCopper Trailer(4words)

1 Error bit
2 Error slots
3 XOR checksum
4 termination word of this block = 0x7fff0006

12/6/2024 VER.1.2 OF NEW PCIE40 DATA FORMAT 2

New data format for B2L/HSLB hdr. and trl. (ver. alpha-200809)

- Basically unchanged
- Added some info in ffaa header and ff55 trailer



Name	Position	Description
CRC	1	Calculated **CRC** of received event does not coincide with the CRC in the last data word
TAG	2	**TT tag** in the received event does not coincide with TT tag from b2tt
INFO	5	**ExpRun** in the received event does not coincide with ExpRun from b2tt
LAST	7	**tlast** received while receiving event header
THR1	8	Throttling in Belle2Link to prevent FIFO overflow
ANY	9	OR of all other error bits
TOUT	10	Timeout, no data received from the FEE for this trigger
AHEAD	11	**TT tag** from b2tt greater than **TT tag** in data
BEHIND	12	**TT tag** from b2tt smaller than **TT tag** in data
THR2	13	Throttling of the long event (> 81920 B in a single channel) in the event processor

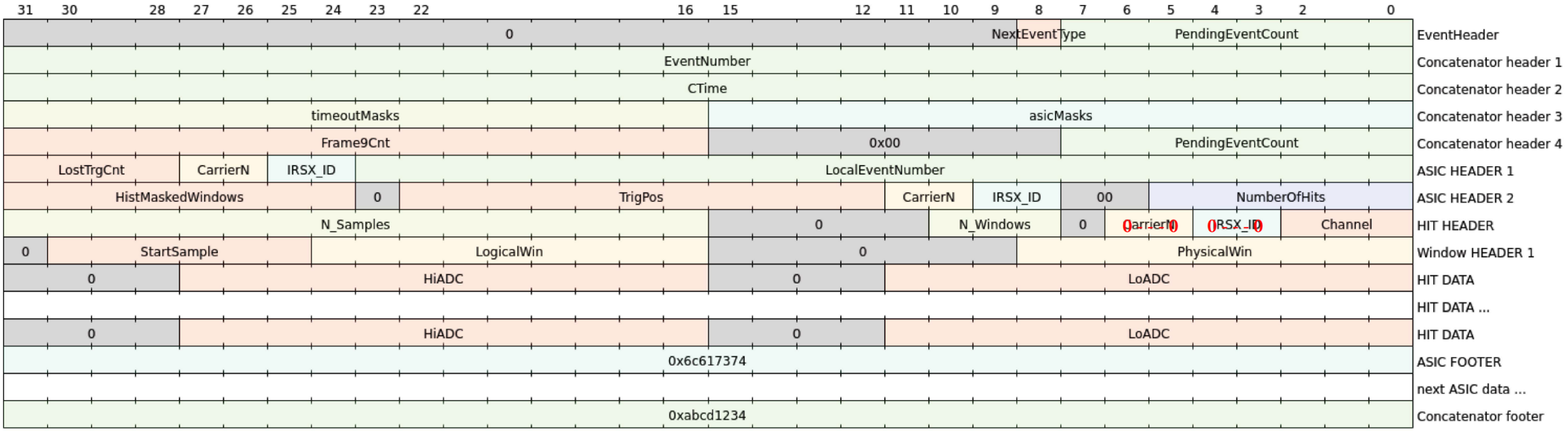
Table VII: Error word in the DMA header and the last word of the event

Position = 0,1,2 …15

VER.1.2 OF NEW PCIE40 DATA FORMAT 3

Raw data format

Note: Hit header bits 3, 4, 5 and 6 are always 0.



TOP Production Data Format

Production Debugging 4.1

2.2 Belle 2 TOP Data Format (Production Data)

Note that the data listed below does NOT include protocol headers; trigger type, ctime, utime, and trgtag are included in Belle2Link headers.

	Bits																															
Word	**31**	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**	**15**	**14**	**13**	**12**	**11**	**10**	**09**	**08**	**07**	**06**	**05**	**04**	**03**	**02**	**01**	**00**	**Hit**
0	Type (=0x04)								Version (=0x01)								0xA				SCROD _ID												N/A
1	EXTRA				NumWordsBonus												Phase(0-8)				NumWordsCore												
2	SKI	RSVD(0000)				ctime (11 LSBs)											Revo9 Counter																
3	ASIC Masks (TimeoutMasks | Register Masks)																eventQueueDepth								eventNumberByte							
4	Carr		IRSX		Channel				Window								0xB				tFine				WF	H/S	**Heap Window**						1
5					vPeak												Integral																1
6					vRise0																vRise1												1
7					vFall0																vFall1												1
8	SampleRise								dSampPeak				dSampFall				**HeaderChecksum**																1
																	. . .																
N*(5+EXTRA)+4	SD_type								Slow data												1	0	1		Nhits								N
	RAW HITS APPENDED HERE.																																

= status bits
= reserved (0 for now)
= unsigned
= signed

Waveform Flag Heap/Stack Flag

Sum of all 16-bit values in "hit header" = 0x0000

"1 0 1 x" = 0xC or 0xD

Event size = (N*5+2) * 4 bytes e.g., for an event with 20 hits --> 408 bytes

at 30 kHz trigger rate, this gives 11.67 Mb/s

8000 is max words

NumWordsCore from hits alone, max is (13*MAX_HITS) = 3328

Max remainder then is 4672, so we should hae 13 bits reserved for it? Per raw hit, we have 18 words, so we can do a max of... 259 word

*Check with Luca on maximum number of hits per channel.

**Waveforms at the very end. Start with some kind. https://www.phys.hawaii.edu/repos/belle2/itop

***Waveform header, waveforms, waveform footer.

Slow data types

5	FPGA temperatures
9	board temperatures
1	Humidity sensor
24	FPGA power values
10	FW/SW versions
128	Trigger scalers
1	pedestal measurement (10-bin average)
178	subtotal