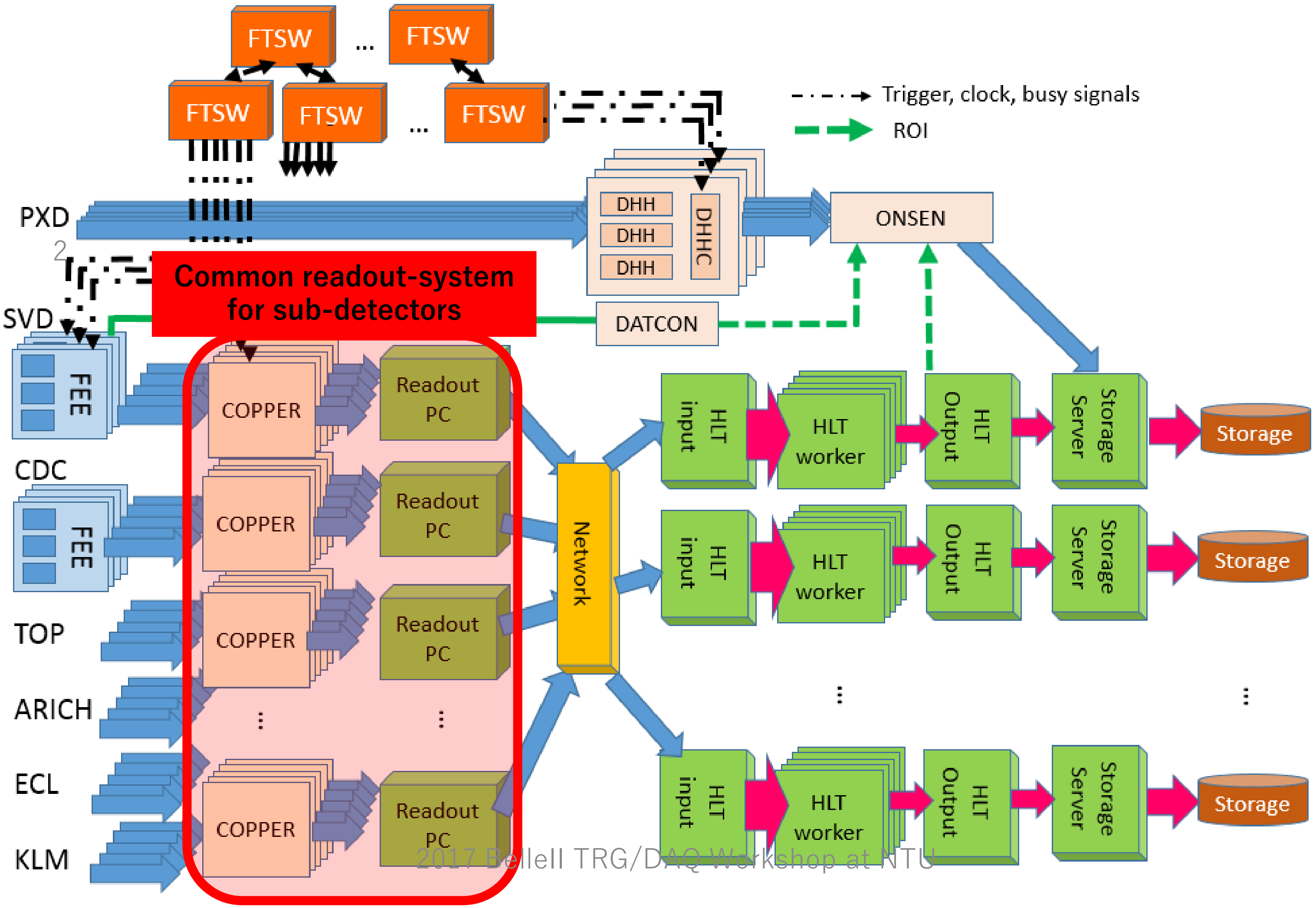
COPPER readout

S. Yamada (KEK, IPNS)

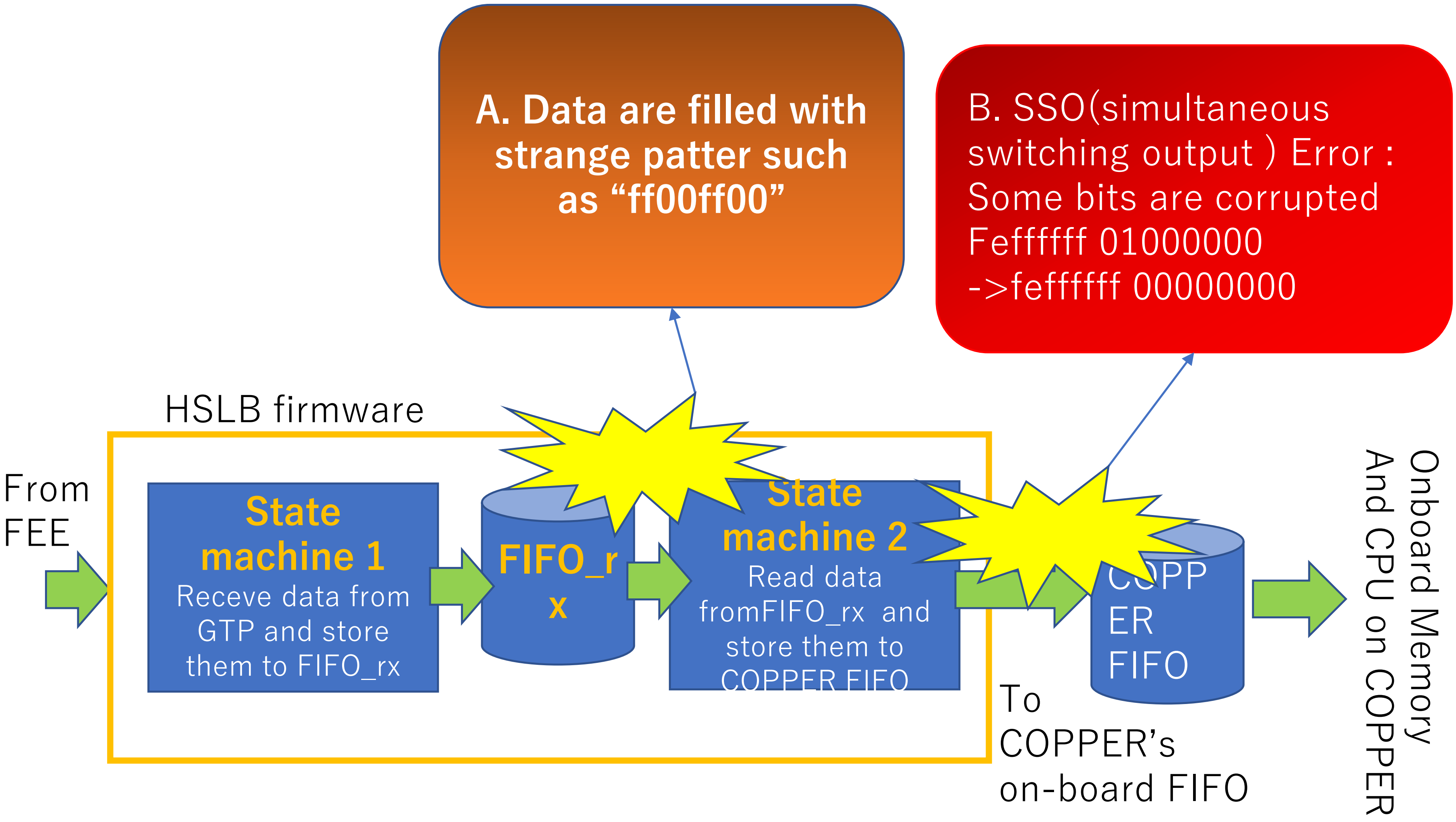
READOUT SUBSYSTEM IN BELLE II DAQ

- Readout data from FEEs of six sub-detectors
  - Receive data from FEE
  - Formatting, data-checking and partial event-building
  - Send data to event builder(eb1) and HLT



1. Data corruption on HSLB/COPPER

Two types of data corruption



A. Data are filled with strange patter such as “ff00ff00”

B. SSO(simultaneous switching output) Error : Some bits are corrupted
Fefffff 01000000
->fefffff 00000000

HSLB firmware

From FEE

State machine 1
Receve data from GTP and store them to FIFO_rx

FIFO_rx

State machine 2
Read data fromFIFO_rx and store them to COPPER FIFO

COPPER FIFO

To COPPER’s on-board FIFO

Onboard Memory And CPU on COPPER

A. Data corruption on HSLB (ff00ff00 error)

🟦 : header/footer attached by HS

🟩 : header/footer attached by FE

⬛ : data contents of FEE

🟥 : strange data

- A large amount of 'ff00ff00' appeared after an FEE footer
- "b2link packetCRC" error is not detected. -> data corruption after HSLB received data.

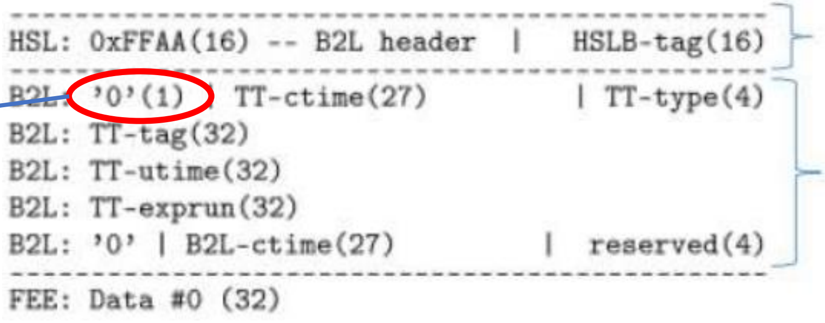
Data of slotD HSLB (corrupted data)

```
ffaa41b5 ff000b4d b8c70002 41b55881 f7af0004 d4000b4d c8c02000 00f24693 00000002 41b50b4d
b8c741b5 7b36fe00 ff00ff00 ff00ff00 ff00ff00 ff00ff00 ff00ff00 ff00ff00 ff00ff00 ff00ff00 ff00ff00 ff00ff00
ff00ff00 ff00ff00
...
ff00ff00 ff00ff00 ff00ff00 ff00ff00 ff00ff00 ff00ff00 ff00ff00 ff00ff00 ff00ff00 ff00ff00 ff550000
```

Workaround to avoid the first "ff00"

- ➢ Just ignore if the 1st byte of an event from FIFO_rx is 'ff' .

The 1st byte is supposed to never be "ff".



```
--------------------------------------------------
HSL: 0xFFAA(16) -- B2L header  |   HSLB-tag(16)
--------------------------------------------------
B2L: '0'(1)  | TT-ctime(27)          | TT-type(4)
B2L: TT-tag(32)
B2L: TT-utime(32)
B2L: TT-exprun(32)
B2L: '0'  | B2L-ctime(27)          |  reserved(4)
--------------------------------------------------
FEE: Data #0 (32)
```

B2link HSLB header

B2link FEE header

After the workaround implemented on Jun.16, this type of "ff00ff00" error has not been was observed so far.

A. Data corruption on HSLB (Other types of strange data pattern)

- “abababab” on KLM COPPER

```
[2017-06-08 15:19:29] [DEBUG] CPR7001 :
212d9d0f 00003227 ff550000 ffaa0000 abababab abababab abababab abababab abababab abababab
abababab abababab abababab abababab abababab abababab abababab abababab abababab abababab
```

- “00ff00ff” on cpr3001 and cpr3006

```
[2017-06-07 08:18:21] [DEBUG] cpr3001 :
010000b5 01b00100 000e006c 6173744e e3f77900 006972fe 00ff00ff 00ff00ff 00ff00ff 00ff00ff
00ff00ff 00ff00ff 00ff00ff 00ff00ff 00ff00ff 00ff00ff 00ff00ff 00ff00ff 00ff00ff 00ff00ff
00ff00ff 00ff00ff 00ff00ff 00ff00ff 00ff00ff 00ff00ff 00ff00ff 00ff00ff 00ff00ff 00ff00ff
00ff00ff 00ff00ff 00ff00ff 00ff00ff 00ff00ff 00ff00ff 00ff00ff 00ff00ff 00ff00ff 00ff00ff
```

- Different type of “ff00ff00” appearance (run3269 cpr2027b)

```
[2017-07-10 16:09:09]
ff550000 ffaa05ef 1968(not ff00)3167 000005ef 5963287a 004ca800 196837d0 22000082 2d380240 000005ef
00da00d9 00d900df 00db00de 00e000e5 00dc00df 00e200e0 00e100e5 00e500eb 00dc00df 00dc00de
…
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00001968 316705ef
475bfe00 ff00ff00 ff00ff00 ff00ff00 ff00ff00 ff00ff00 ff00ff00 ff00ff00 ff00ff00 ff00ff00
```

These errors cannot be avoided by the workaround for the “ff00ff00” problem.

Still under investigation to fix the problem.

B. Data corruption on COPPER

After reducing the current drive of 32-bit data lines from HSLB, data corruption on COPPER was reduced but still the following error remains.

i. Observed in TOP data

- CRC errors reported in TOP local/global data-taking. Using the output log of an error event, I put the same data pattern to dummy-data production firmware for HSLB and observed data-corruption.

After feff, a bit in the next word is corrupted

```
feff0400 fefffdff feff0000 01000000 02000500 03000200 0300ffff fcfff9ff f5fff7ff f5fffbff
…
 feff0400 fefffdff feff0000 01000000 02000500 03000200 0300ffff fcfff9ff f5fff7ff f5fffbff
…
 fefffbff f6fff6ff 01000300 0900ffff 01000200 07000000 f9fffdff fafffeff 00000000 f7fff6ff
```

ii. Also observed in CDC suppressed data of high-rate test

B2link header

B2link trailer

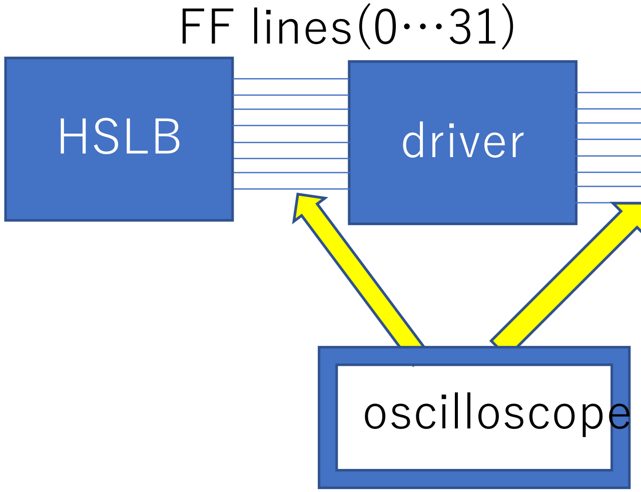
```
HSLB slot A : ffaac4f0 3afb29f7 0a2ac4f0 594b67f9 004b2200 3afb39e0 2000000f 2e7a0000 0a2ac4f0 3afb29f7 c4f0d013 ff550000
HSLB slot B : ffaac4f0 3afb29f7 0a2ac4f0 594b67f9 004b2200 3afb39e0 20000068 2e7a0000 0a2ac4f0 3afb29f7 c4f0999d ff55000
HSLB slot C : ffaac4f0 3afb29f7 0a2ac4f0 594b67f9 004b2200 3afb39e0 200000d1 5afa0000 0a2ac4f0 3afb29f7 c4f00469 ff550000
HSLB slot D : ffaac4f0 3afb29f7 0a2ac4f0 594b67f9 004b2200 3afb39e0 2000010a 2e7a0000 0a2ac4f0 3afb29f7 c0f0c89a ff55000
```

iii. The error can be reproduced with a test pattern(= feffffff 01000000)

- We tried "feffffff 0100000" pattern and it caused data corruption.

  -
    ```
    [DEBUG] 01000000 feffffff 01000000 feffffff 00000000 feffffff 01000000 feffffff 01000000 feffffff
    [DEBUG] 01000000 feffffff 01000000 feffffff 01000000 feffffff 01000000 feffffff 00000000 feffffff
    ```

iv. Data are corrupted before FIFO?



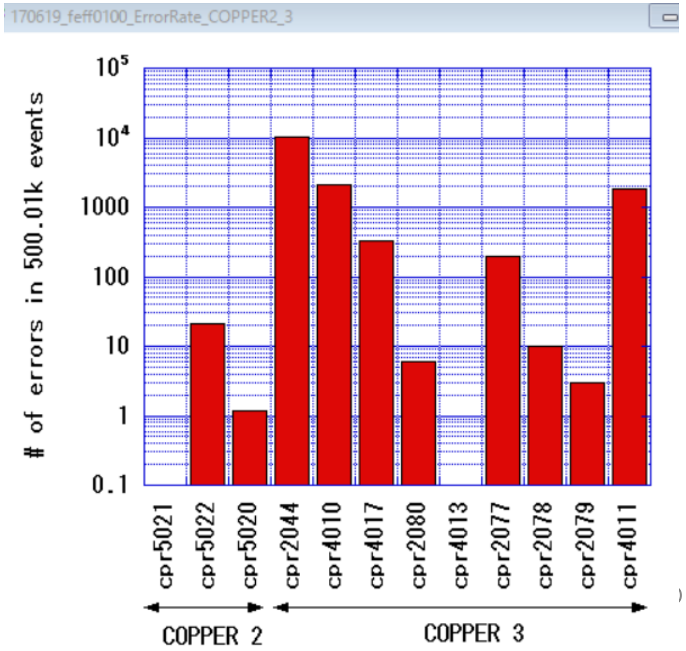
- We checked waveform by oscilloscope, no data corruption was detected by an oscilloscope before the 1st FIFO.
- Further tracing the signal line seems to be difficult.

v. Error-rate dependence on boards

# of CRC errors in TOP calib. Test (Feb.10-Apr.26)

	slot A	slot B	slot C	slot D
cpr3001	0	0	0	2
cpr3002	0	0	0	0
cpr3003	63	6	15	1015
cpr3004	1	12	4	524
cpr3005	18	13	19	71
cpr3006	176	20	73	2190
cpr3007	2	0	3	23
cpr3008	50	37	269	1419
cpr3009	4	3	5	207
cpr3010	0	0	0	1
cpr3011	120	10	15	142
cpr3012	35	2	8	32
cpr3013	7	0	1	0
cpr3014	2	1	4	79
cpr3015	210	15	143	1702
cpr3016	10	1	2	33

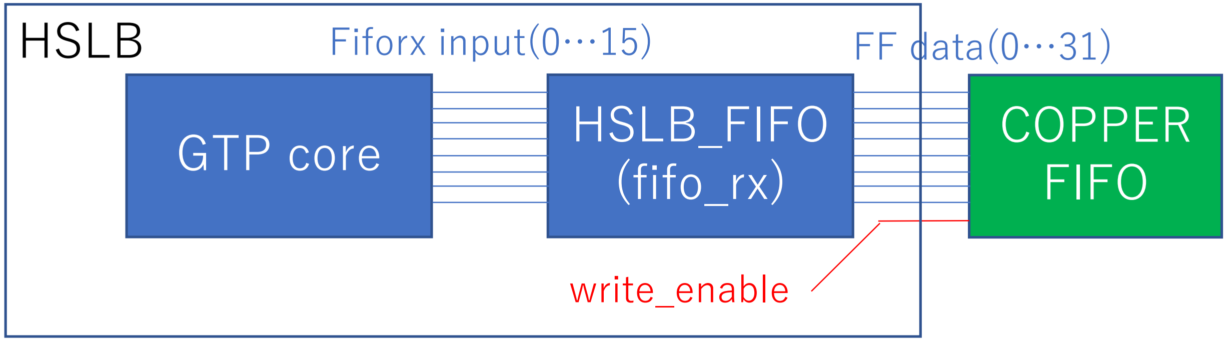
# of CRC errors in feffffff 01000000 test pattern



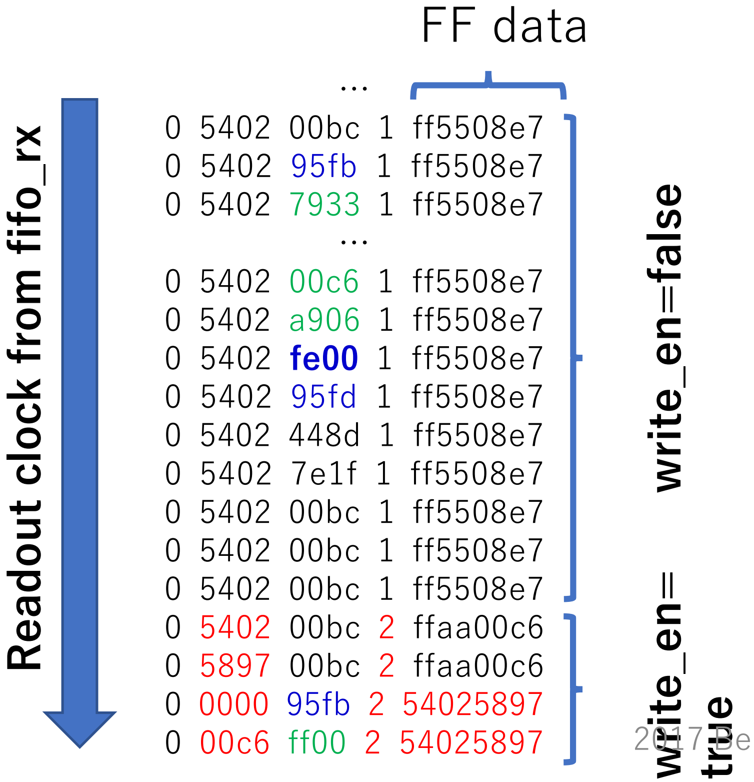
Replacing some problematic boards for TOP DAQ would be worth trying.

Reduction of the corruption rate (1)

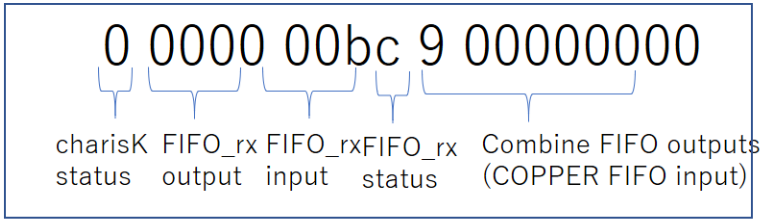
Data flow from HSLB to COPPER FIFO



FF data output and wirte_enable



```
                     …
0  5402  00bc  1  ff5508e7
0  5402  95fb  1  ff5508e7
0  5402  7933  1  ff5508e7
                     …
0  5402  00c6  1  ff5508e7
0  5402  a906  1  ff5508e7
0  5402  fe00  1  ff5508e7
0  5402  95fd  1  ff5508e7
0  5402  448d  1  ff5508e7
0  5402  7e1f  1  ff5508e7
0  5402  00bc  1  ff5508e7
0  5402  00bc  1  ff5508e7
0  5402  00bc  1  ff5508e7
0  5402  00bc  2  ffaa00c6
0  5897  00bc  2  ffaa00c6
0  0000  95fb  2  54025897
0  00c6  ff00  2  54025897
```



0 0000 00bc 9 00000000

- ➢ So far, even when write_en = false, the output from HSLB (FF data) has some value.
- ➢ I changed the firmare to keep FF_data zero while write_en = faluse.

Reduction of the corruption rate (2)

B3 test bench (cpr2044)

- Dummy data (fefffffff 01000000) production firmware (dummy data -> HSLB fifo -> COPPER FIFO)
- Download firmware to all four HSLBs on a COPPER

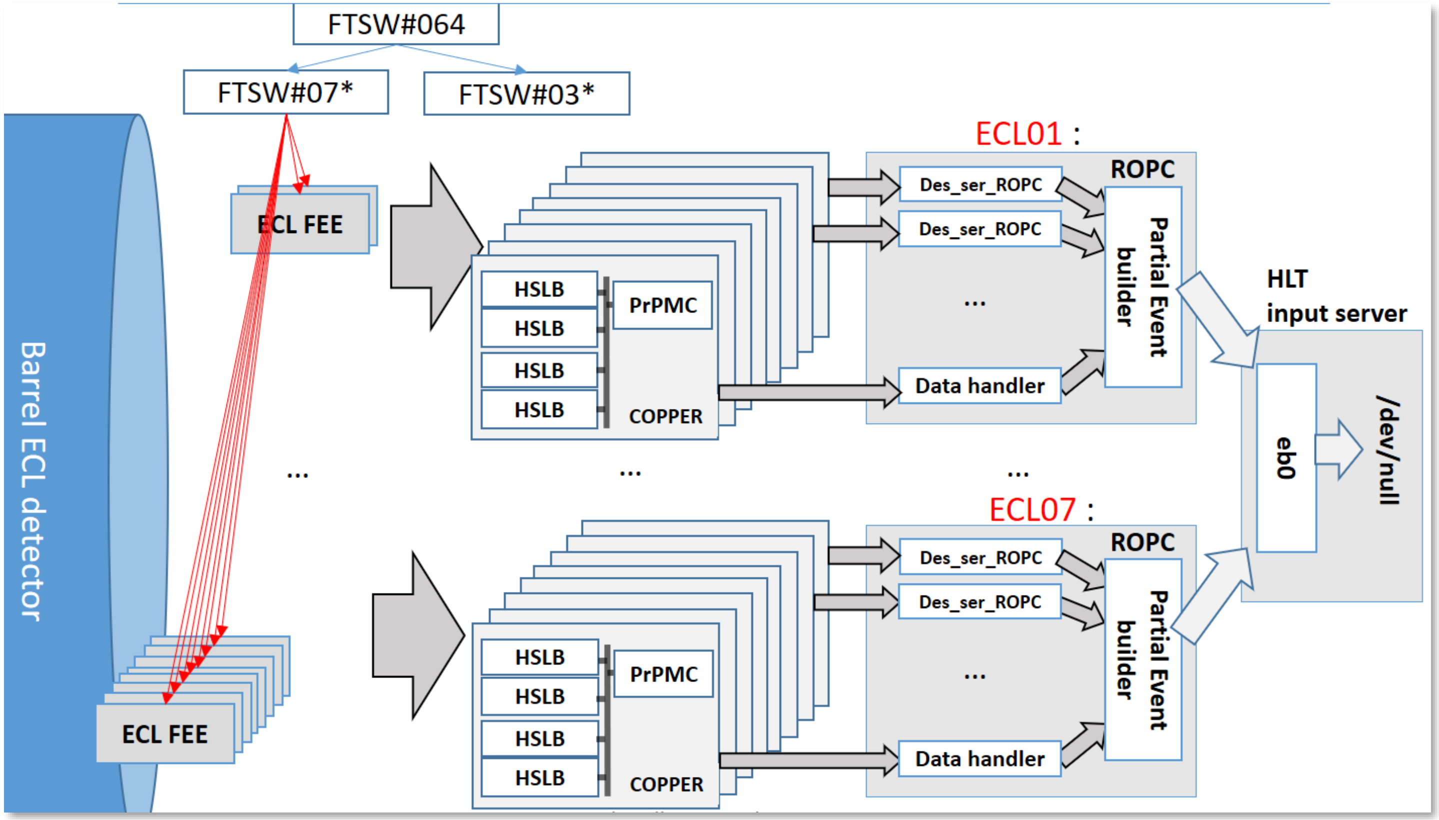
- Normal firmware :
  - Error rate : 0.0056 error/event (= 7779error/ (156sec*8.93kHz))
- FF_data is zero when write_enable is false
  - Error rate : < 1.6e-9 error/event (No error in (178353 sec * 8.94kHz) events)

CDC FEEs

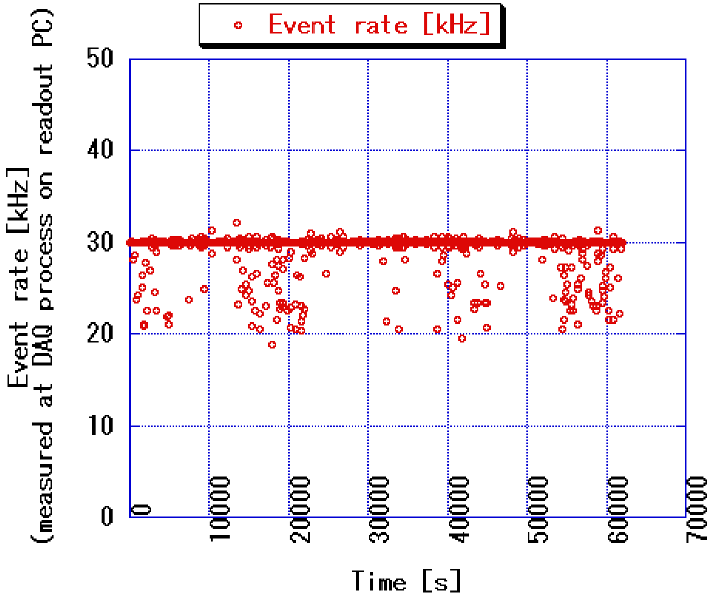
- Use almost all CDC FEEs.
  - Dummy trigger input 100kHz :
  - Suppress mode in CDC data format
- Normal firmware
  - Output event rate : 53kHz (run 20170712_1314)
  - Error rate : 5.6e-6 error/event (=1822 error/(6093sec*53kHz))
- FF_data is zero when write_enable is false (run 20170817_2343)
  - Output event rate : 66.6kHz
  - Error rate : < 2.1e-9 error/event (No error in 40506 sec.*66.6kHz)

Improved. We'd like to test this firmware in TOP.

2. Large-scale stress test



2-1. Stress test for ECL : input pseudo-Poisson 30kHz



- Throughput : about 33MB/s/COPPER (the expected event size from MC)
  - Event size was adjusted by HIT threshold of ECL FEE
  - Total throughput for Barrel ECL : 600 MB/s
- Constant 30kHz trigger -> efficiency = 99.2%
- Pseudo-Poisson 30kH trigger -> efficiency = 98.2 %
  - The deadtime comes from trigger limitation (5trigger in 26us due to SVD FEE). -> 5trigger/26us from SVD is not correct. (My misunderstanding)

Stress test for ECL : input pseudo-Poisson 100kHz trigger

- ➢ Max. # of triggers from FTSW : 5trg in 26us -> Not good
  - ➢ COPPER FIFO became completely full

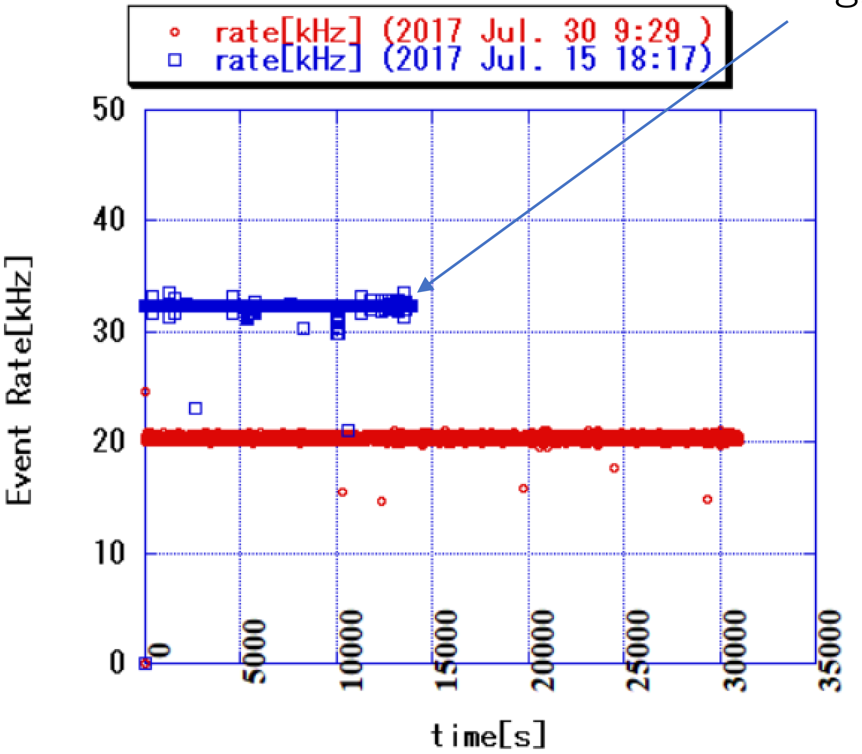
```
[yamadas@cpr5008:~]$ cat /proc/copper/FF_STA
23 23 3c 3c  (3c : completely full., 2c : almost full, : 23 empty)
[yamadas@cpr5008:~]$ cat /proc/copper/LEF_STA
0000 1 1 e e  (e : completely full., 6 : almost full, : 1 empty)
```

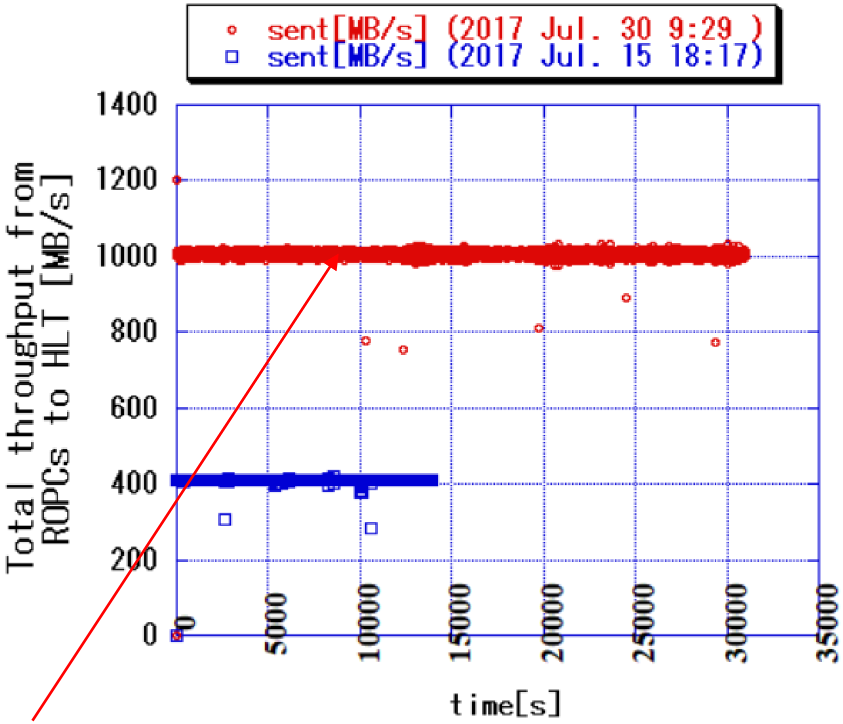
- ➢ Max. # of triggers from FTSW : 5trg in 130 us -> Not good
  - ➢ Strange b2tt header came from FEEs
    - ➢ ffaabbb2 ffffffff ffffffff fffffff2 00002100
    - ➢ According to Nakao-san, this means that b2tt FIFO becomes full due to high-rate trigger.

Stress test for ECL : input pseudo-Poisson 100kHz trigger

- Max. # of triggers from FTSW : 12trg in 350 us (FTSW default value)
  - 32kHz is the max. rate due to the trigger hold-off.

Limited by the trigger hold-off





Probably limited by 10GbE line rate between readout PCs and HLT

ECL Error type (rough classification): July-Aug.

Data corruption in HSLB/COPPER

- CRC error : Data corruption in HSLB or COPPER

  -> 1 time : Probably due to SSO bit error

  -> 4 times : magic words are strange -> due to HSLB FIFO related error

Other errors

- Mismatch between different HSLB headers

-> 2943 times

  - No CRC errors. Different event # in HSLBs.

HSLB slotA : ctimeTRGtype 0x2a025807 utime 0x598133ad **eve 0x0000211e** exprun 0x00002700
HSLB slotB : ctimeTRGtype 0x2a927d57 utime 0x598133ad **eve 0x000021c9** exprun 0x00002700

- Invalid event number at the beginning of run.

-> 52 times

- Event # jump (event # != 0xffffffff)

-> 268 times

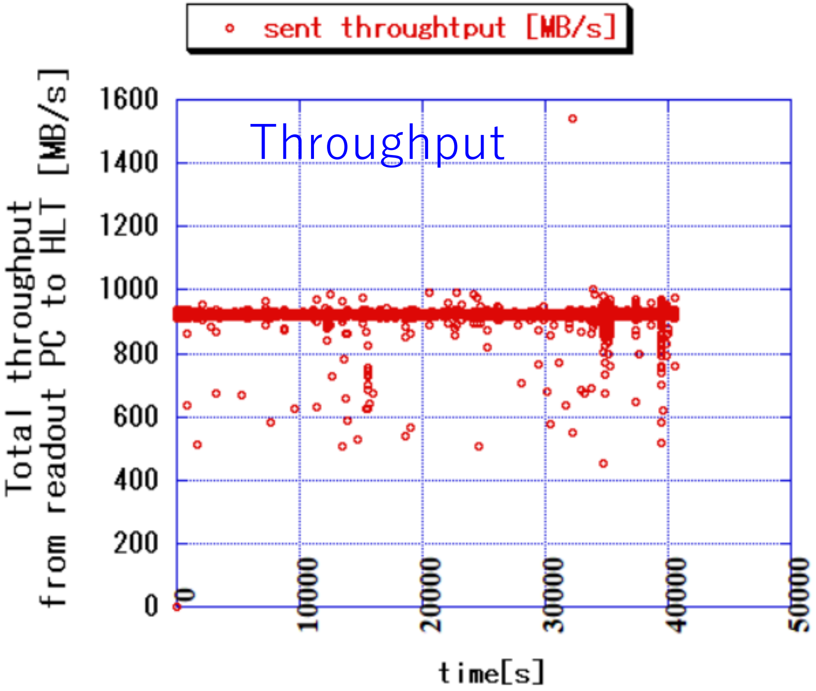
- Event # jump (event # = 0xffffffff)

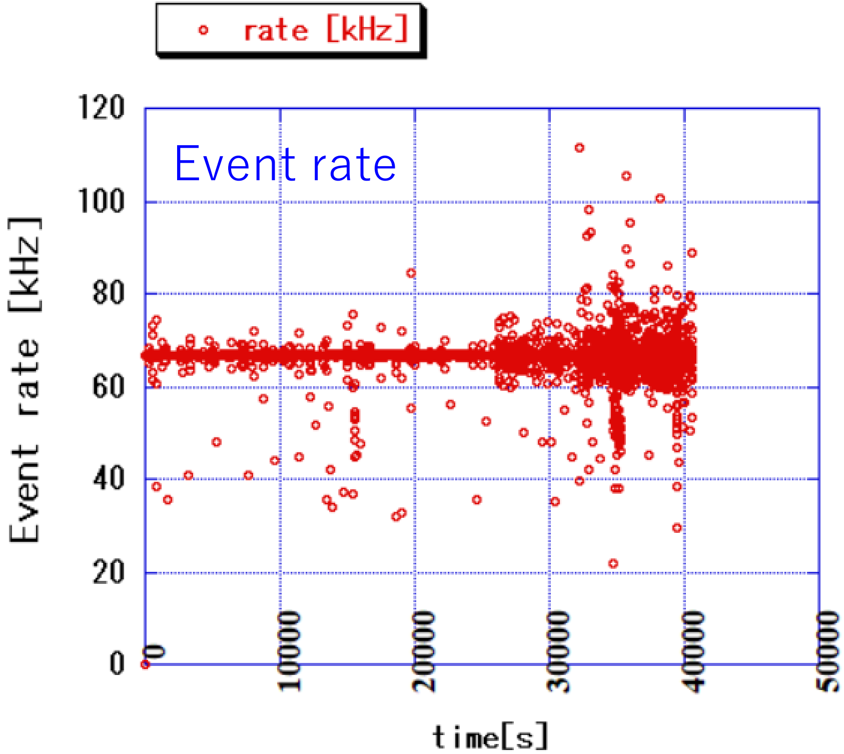
  - Error around b2tt on FEE ?

-> 158 times

2-2. Stress test for CDC : input pseudo-Poisson 100kHz trigger

- CDC Suppressed data mode
- Trigger hold-off : 5trg/26us limitation
- Data are discarded at an HLT input server.





- 66kHz output is limited by 5trg/26us trigger hold-off
- This run(8/17 23:43) was stopped manually.
  - SSO bit error did not occur in this run after changing the HSLB firmware

CDC Error type (rough classification): June-Aug.

Errors mainly due to data corruption in HSLB/COPPER

There are overlaps.

- CRC error : Data corruption in HSLB or COPPER

-> 2637 times : which is due to either

  - Data corruption around HSLB FIFO or
  - Bit error due to SSO error

- Mismatch between header and trailer

-> 2871 times

  - Mainly due to SSO bit error

- Mismatch between different HSLB headers

-> 2258 times

  - Mainly due to SSO bit error

```
HSLB slotA : ctimeTRGtype 0x3bd5d127 utime 0x5965a7a1 eve 0x040a43de exprun 0x004cf200
HSLB slotB : ctimeTRGtype 0x3bd5d127 utime 0x5965a7a1 eve 0x040a43de exprun 0x004cf200
HSLB slotC : ctimeTRGtype 0x3bd5d127 utime 0x5965a7a1 eve 0x040a43de exprun 0x004cf200
HSLB slotD : ctimeTRGtype 0x3bd5d127 utime 0x5965a7a1 eve 0x000a43de exprun 0x004cf200
```

Other errors

- Invalid event number at the beginning of run.

-> 784 times

- Event # jump (event # != 0xffffffff)

-> 172 times

- Event # jump (event # = 0xffffffff)

-> 69 times (Error around b2tt on FEE ?)

- Different event # over HSLBs

-> 2 times

Summary

- The effort to reduce the occurrence of the data corruption on COPPER board is ongoing
  - HSLB-FIFO related data-corruption
    - Fixed by a workaround to some extent
    - Still remains.
  - SSO data corruption :
    - Reduction of current for output data-lines from HSLB is efficient to reduce SSO errors but not enough.
    - In GCRT, we found corrupted events in TOP data. We add error-tag in corrupted events and continues data-taking.
    - Hopefully, it will be gone by the latest update of HSLB firmware. (to be confirmed)

- High rate test was performed for CDC and ECL
  - Only up to the HLT input server.
  - Performance is good. Some errors were observed in data.

  -> needs further investigation.

backup

Data corruption on COPPER

Data corruption in “fffffff 0000000” pattern

A. How they are corrupted

```
[DEBUG] 00000000 ffffffff 00000000 ffffffff 00000000 ffffffff 00000000 ffffffff 00000000 02ffffff
[DEBUG] 00000000 ffffffff 00000000 ffffffff 00000000 ffffffff 00000000 ffffffff 00000000 ffffffff
[DEBUG] 00000000 ffffffff 00000000 ffffffff 00000000 ffffffff 00000000 ffffffff 00000000 ffffffff
[DEBUG] 00000000 ffffffff 00000000 ffffffff 00000000 ffffffff 00000000 8effffff 00000000 ffffffff
```

Only the most significant byte in a 4-byte word are corrupted

B. Reduction of the current drive of HSLB data output works :

- in hslb_***.ucf. (default 12mA to 2mA)
- Errors after the modification at the B2/B3 test bench
- B3 setup
  - 12xCOPPER (4HSLB/COPPER)
  - Input trigger 30kHz Poisson : output trigger 1.1kHz
  - Data pattern : ffffffff 00000000
  - No data corruption in 118.5hours for 323.3Mevents

But, CRC error still occurs in the TOP DAQ event with the reduced-current version…

ECL Error type (rough classification)

- **Event # jump**
  - 90 times : No CRC errors -> No data corr
  - *Possibilities*
    - *Wrong Event # is attached on FEE (b2link core)*
    - *Events were lost in FEE or HSLB/COPPER*
    - *DAQ process in prev. run was not killed and two readout processes might read HSL simultaneously*
  - 3 times : event CRC error -> Corruption in HSLB occurred, which might change event # in header.

Device driver is now being modified so that doubly opening /dev/copper/hslb:* can be prohibited.

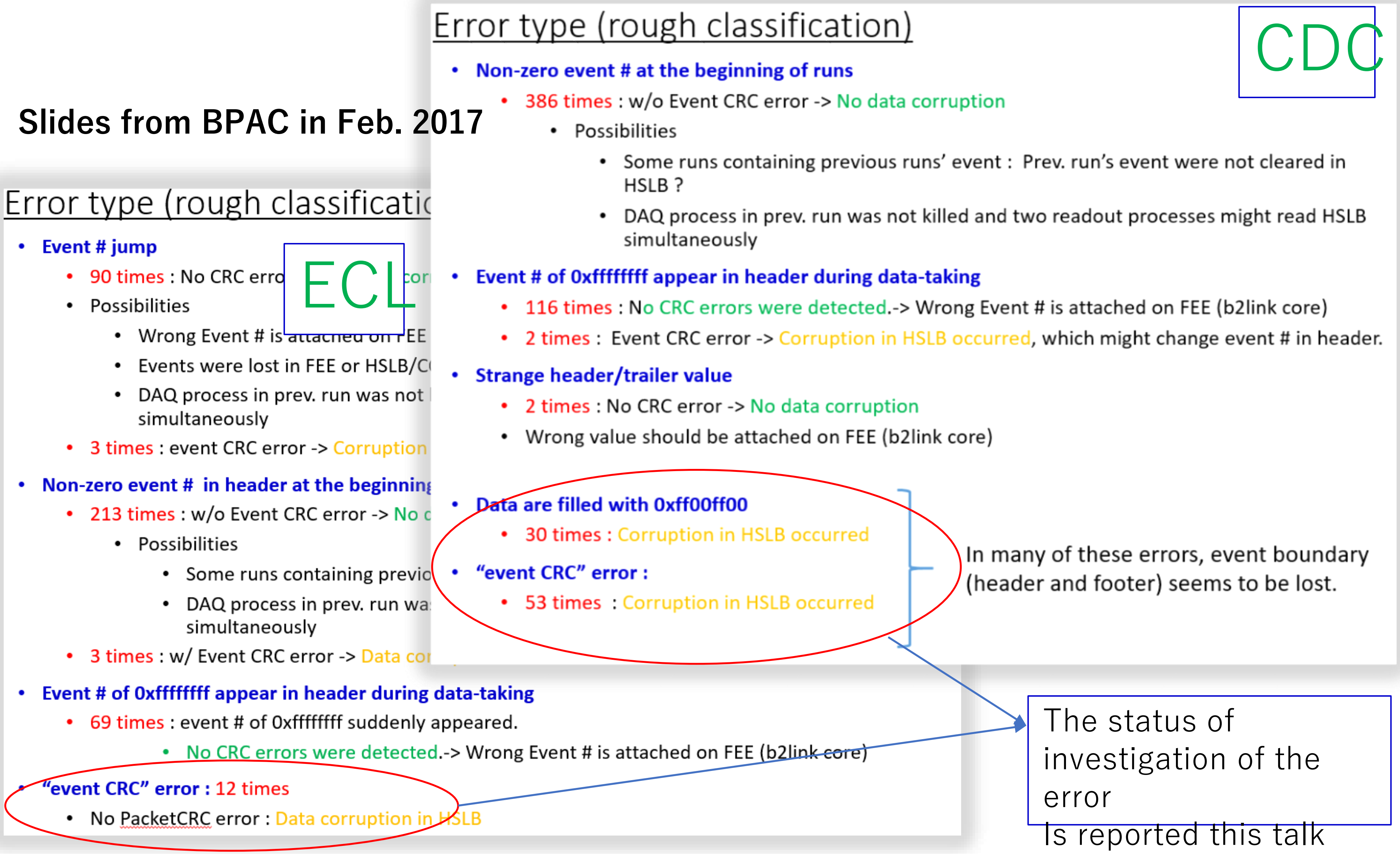
- **Non-zero event # in header at the beginning of runs (Th 1 ro.)**
  - *213 times : w/o Event CRC error -> No data corruption*
    - *Possibilities*
      - *Some runs containing previous runs' event : eared in HSLB ?*
      - *DAQ process in prev. run was not killed and two might read HSLB simultaneously*
  - 3 times : w/ Event CRC error -> Data corruption on HSLB

und at this error happens after ECL FEEs sent over-sized waveform events -> configuration of FEE is changed to use compression mode.

- **Event # of 0xffffffff appear in header during data-taking**
  - 69 times : event # of 0xffffffff suddenly appeared.
    - No CRC errors were detected.-> Wrong Event # is attached on FEE (b2link core)
- **"event CRC" error :** 12 times
  - No PacketCRC error : Data corruption in HSLB

CRC errors are detected during the stress test in ECL and CDC

Slides from BPAC in Feb. 2017



Error type (rough classificati

ECL

- **Event # jump**
  - 90 times : No CRC erro cor
  - Possibilities
    - Wrong Event # is attached on FEE
    - Events were lost in FEE or HSLB/C
    - DAQ process in prev. run was not simultaneously
  - 3 times : event CRC error -> Corruption
- **Non-zero event # in header at the beginning**
  - 213 times : w/o Event CRC error -> No d
    - Possibilities
      - Some runs containing previo
      - DAQ process in prev. run wa simultaneously
  - 3 times : w/ Event CRC error -> Data cor
- **Event # of 0xffffffff appear in header during data-taking**
  - 69 times : event # of 0xffffffff suddenly appeared.
    - No CRC errors were detected.-> Wrong Event # is attached on FEE (b2link core)
- **"event CRC" error : 12 times**
  - No PacketCRC error : Data corruption in HSLB

Error type (rough classification)

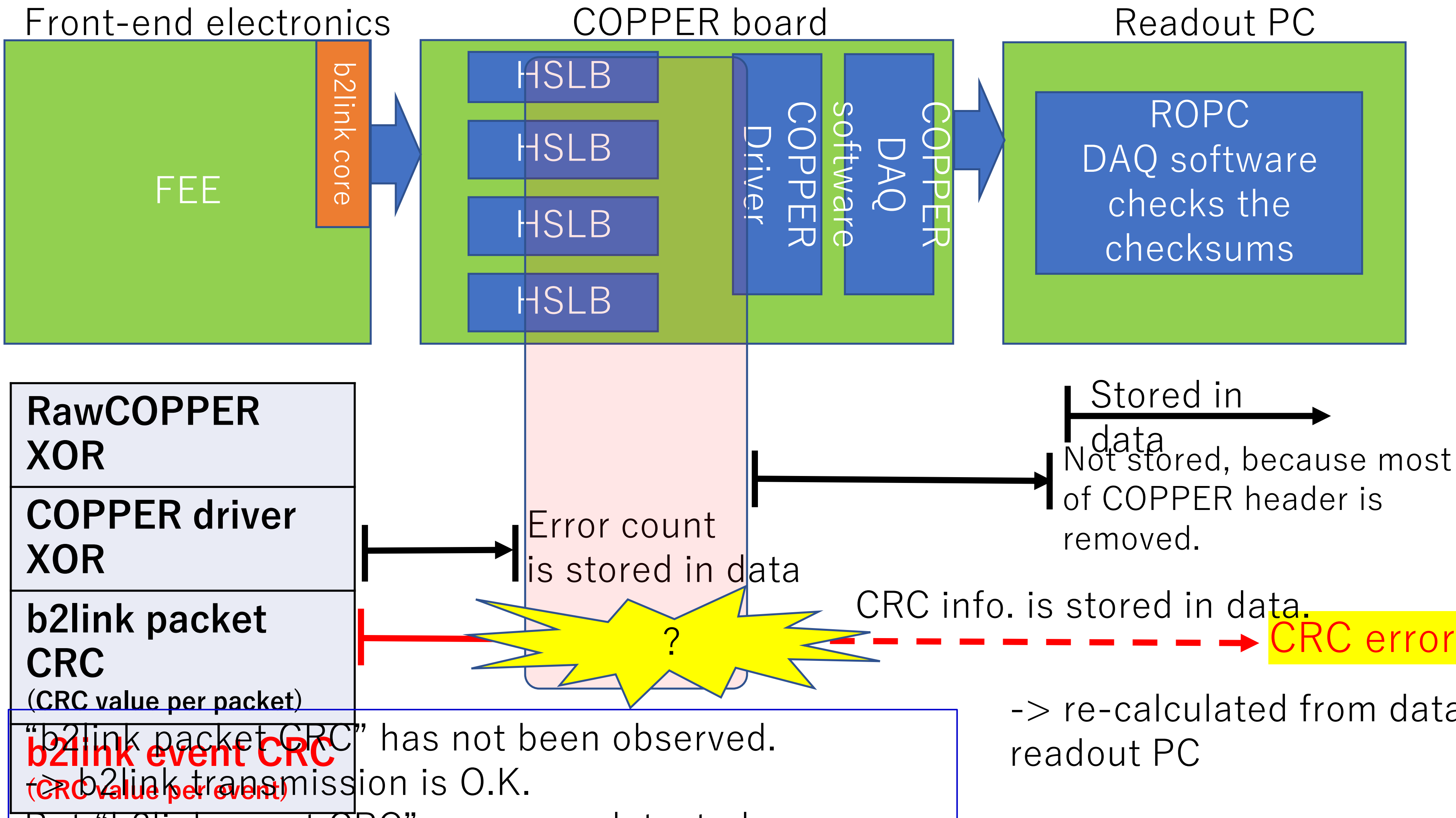
CDC

- **Non-zero event # at the beginning of runs**
  - 386 times : w/o Event CRC error -> No data corruption
    - Possibilities
      - Some runs containing previous runs' event : Prev. run's event were not cleared in HSLB ?
      - DAQ process in prev. run was not killed and two readout processes might read HSLB simultaneously
- **Event # of 0xffffffff appear in header during data-taking**
  - 116 times : No CRC errors were detected.-> Wrong Event # is attached on FEE (b2link core)
  - 2 times : Event CRC error -> Corruption in HSLB occurred, which might change event # in header.
- **Strange header/trailer value**
  - 2 times : No CRC error -> No data corruption
  - Wrong value should be attached on FEE (b2link core)
- **Data are filled with 0xff00ff00**
  - 30 times : Corruption in HSLB occurred
- **"event CRC" error :**
  - 53 times : Corruption in HSLB occurred

In many of these errors, event boundary (header and footer) seems to be lost.

The status of investigation of the error Is reported this talk

How CRC error is detected



Front-end electronics

FEE

b2link core

COPPER board

HSLB

HSLB

HSLB

HSLB

COPPER Driver

COPPER DAQ software

Readout PC

ROPC DAQ software checks the checksums

RawCOPPER XOR

COPPER driver XOR

b2link packet CRC
(CRC value per packet)

b2link event CRC
(CRC value per event)

Stored in data

Not stored, because most of COPPER header is removed.

Error count is stored in data

?

CRC info. is stored in data.

CRC error

-> re-calculated from data
readout PC

"b2link packet CRC" has not been observed.
-> b2link transmission is O.K.
But "b2link event CRC" error was detected.
-> Data should be corrupted after receiving data from a FEE.

CDC Error type (rough classification)

- **Non-zero event # at the beginning of runs**
  - *386 times : w/o Event CRC error -> No data cor*
    - *Possibilities*
      - *Some runs containing previous runs' event : Prev.* *cleared in HSLB ?*
      - *DAQ process in prev. run was not killed and two readout processes might read HSLB simultaneously*

ce driver is now being modified so that doubly opening /dev/copper/hslb: * can be prohibited

- **Event # of 0xffffffff appear in header during data-taking**
  - 116 times : No CRC errors were detected.-> Wrong Event # is attached on FEE (b2link core)
  - 2 times : Event CRC error -> Corruption in HSLB occurred, which might change event # in header.
- **Strange header/trailer value**
  - 2 times : No CRC error -> No data corruptio
  - Wrong value should be attached on

Workaround by ignoring the first "ff00" in HSLB firmware

- **Data are filled with 0xff00ff00**
  - 30 times : Corruption in HSLB occurred
- **"event CRC" error :**
  - 53 times : Corruption in HSLB occurred

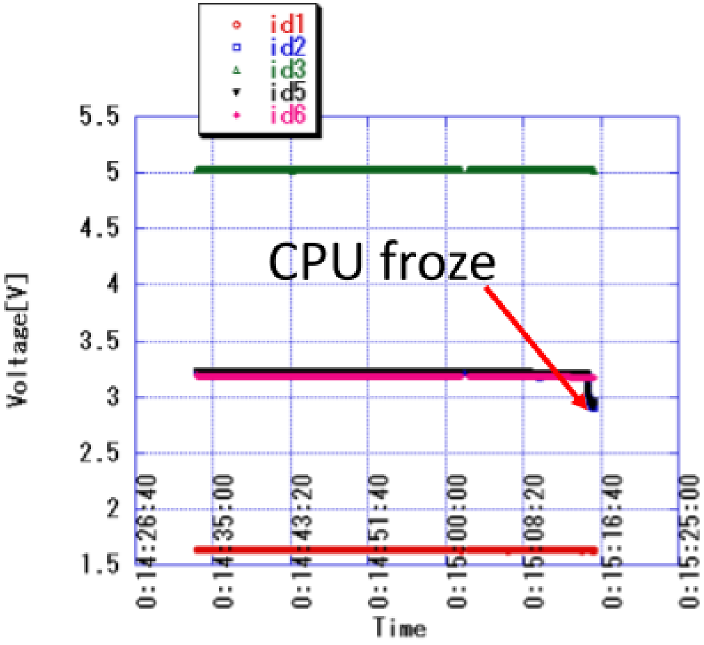
of these errors, event boundary (header and footer) seems to be lost.

Voltage drop problem in COPPER crate (reported at the Oct./Feb

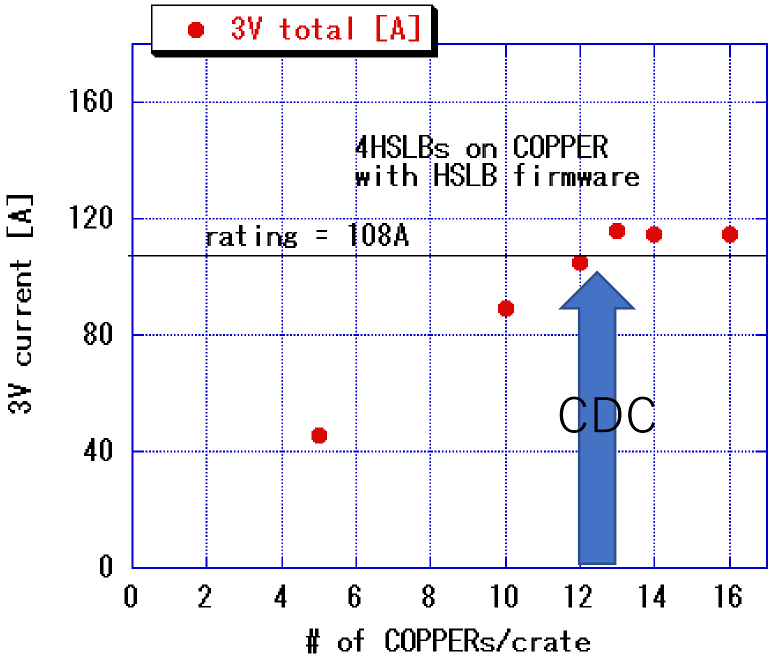
- COPPER onboad CPU sometime froze due to 3V voltage drop.
- 3V power supply of COPPER crates should be reinforced for six CDC crates, whose # of COPPERs per crate is larger than other sub-detectors.

-> Last JFY, we bought 3 out of 6. The reset 3 crates will arrive at KEK this month.

# of COPPER = 15, 4HSLBs/COPPER



3V current usage of COPPER crate



Dependence on # of HSLBs / COPPER

- Download to slot A,B,C and D but read only slot D (04301107)
  - Error rate : 15.1 Hz (11 COPPERs, pulse 1kHz)
- Download to slot D and read slot D (04292006)
  - Error rate : 0.004 Hz (11 COPPERs, pulse 1kHz)

-> Error rate is decreased.

Other attempts in vain

Condition	result	# of COPPER	run #
f_ff(32) : drive=2mA -> 4mA	data corrpted	TOP setup	
f_ff(32) iostandard : lvcmos33 -> lvcmos25	data corrpted	11	1704301156
fwclk : drive=24mA -> 8mA	data corrpted	11	1704301226
fwclk : drive=24mA -> 2mA	data corrpted	11	1704301212
pulse 100Hz	data corrpted	10	1705051744
pulse 75Hz	data corrpted	10	1705060946
f_ff(24) 2mA -> 24mA	data corrpted	1	1705061533
Replace HSLB on slot D	data corrpted	1	1705081312
Replace TTRX	data corrpted	1	1705081338
Delay f_ff(i) by i taps by iodelay	data corrpted	1	1705091908

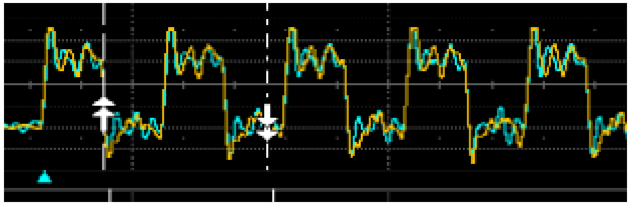
Attempts

- Insert DFF in data/clock lines in the HSLB firmware
- Change the relative timing between clock and data before writing the data to COPPER FIFO by iodelay module
- Change the output signal : lvttl, lvcmos25
- Change the clock frequency : 63.5MHz -> 42MHz
- Use the simple HSLB firmware without b2link (just writing dummy data to COPPER FIFO.)

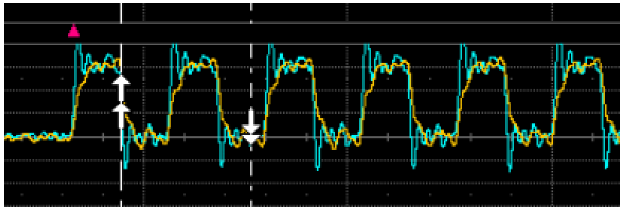
-> It didn’t work.

- Reduce the current drive for data output in hslb_***.ucf. (default 12mA)
  - 8mA : CPU does not freeze but there were the CRC errors
  - 2mA : The error does not appear in the test bench.
  - 2mA seems to be enough to drive the output signal. (pictures below)

12mA
(Yellow: data signal at HSLB output pin
Blue : After a driver)

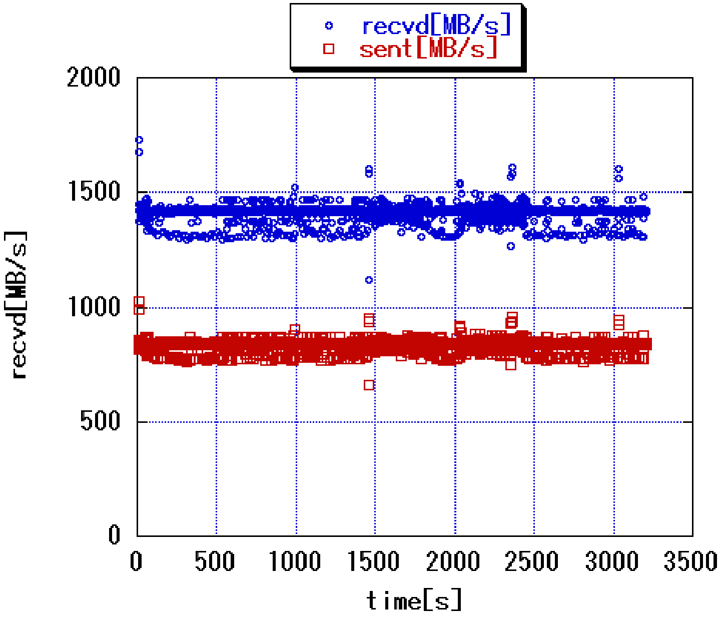


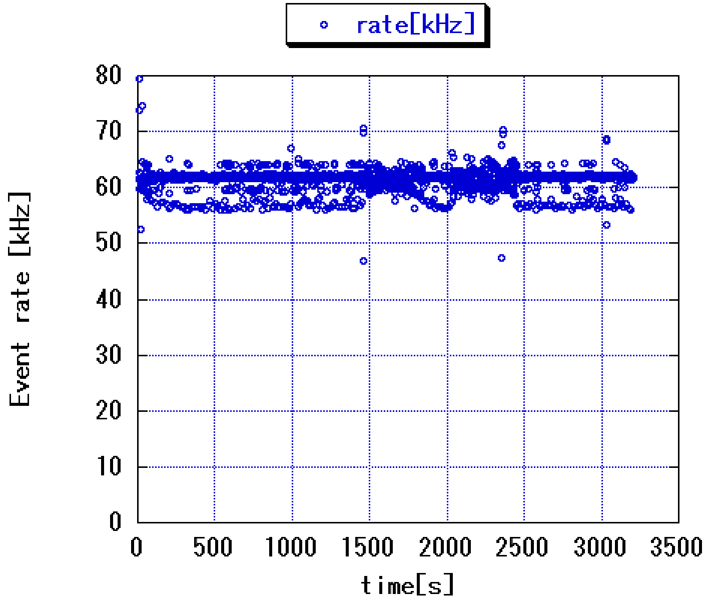
2mA
(Yellow: data signal at HSLB output pin
Blue : After a driver)



Recent stress test for CDC (After avoiding “ff00ff00” data corrup

- ➢ 100kHz Pseudo Poisson input (5trg/26us limitation)
- ➢ CDC HV off : Almost no hits
- ➢ Data are discarded at hlt input server.
  - ➢ Nice nc processes were used to receive data from nine readout PC





- ➢ DAQ stopped after about 1hour running due to data corruption.
- ➢ Trailer value is corrupted : c4f0 -> c0f0 . Probably, SSO related corruption on C

B2link header

B2link trailer

HSLB slot A : ffaac4f0 3afb29f7 0a2ac4f0 594b67f9 004b2200 3afb39e0 2000000f 2e7a0000 0a2ac4f0 3afb29f7 c4f0d013 ff550000

HSLB slot B : ffaac4f0 3afb29f7 0a2ac4f0 594b67f9 004b2200 3afb39e0 20000068 2e7a0000 0a2ac4f0 3afb29f7 c4f0999d ff550000

HSLB slot C : ffaac4f0 3afb29f7 0a2ac4f0 594b67f9 004b2200 3afb39e0 200000d1 5afa0000 0a2ac4f0 3afb29f7 c4f00460 ff550000

*** If the 1st byte is ff, wait for one clok.**

```
<           if( dataout(16 downto 9) = x"ff" and first_three = '0'  ) then
<             Next_state := THREE;
<             first_three := '1';
<            else
<             Next_state  := FOUR;
<            end if;
```

*** If the 1st byte is ff, flag for writing to COPPER _FIFO is kept disabled.**

```
<      elsif Current_state = THREE and ( ( dataout(16 downto 9) /= x"ff" ) or
first_three = '1' )then
<        F_fwenb <= '0';
<      elsif Current_state = THREE and dataout(16 downto 9) = x"ff" and first_three
= '0' then
<        F_fwenb <= '1';
```

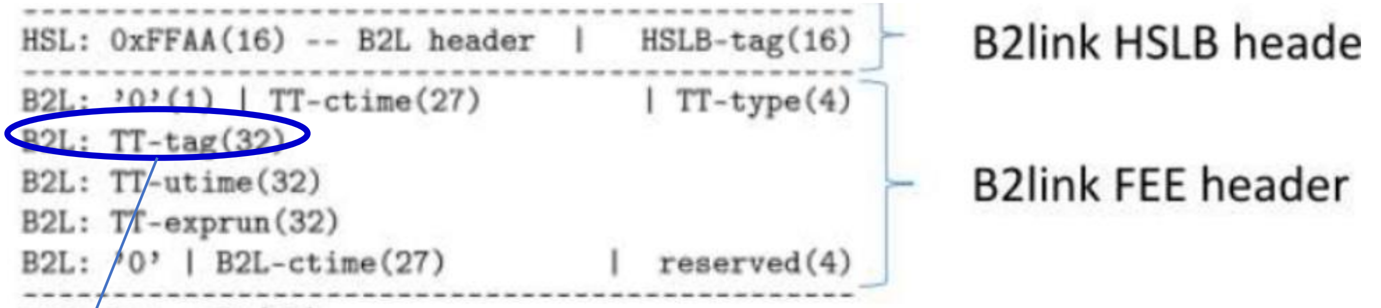
*** Adjust clock for writing COPPER FIFO**

```
< if Current_state = ZERO then
<      sig_fwclk <= '1';
<    elsif Current_state = THREE then
<     sig_fwclk <= '0';
<    else
<      sig_fwclk <= not sig_fwclk;
```

After the workaround, Ff00ff00 did not occurred in
4 runs.

Data-taking continued a bit longer for 300-700s.

New type of Data Corruption in CDC COPPERs reported at prev. DAQ meeting -> seems to be fixed last Friday



```
-----------------------------------------------------------
HSL: 0xFFAA(16) -- B2L header   |   HSLB-tag(16)
-----------------------------------------------------------
B2L: '0'(1) | TT-ctime(27)              | TT-type(4)
B2L: TT-tag(32)
B2L: TT-utime(32)
B2L: TT-exprun(32)
B2L: '0' | B2L-ctime(27)                |  reserved(4)
-----------------------------------------------------------
```

B2link HSLB heade

B2link FEE header

[Header of event-fragment received by each HLSB]
Cpr2064 run 2241
Slot A: ffaa02a7 6e2202a7 4c241bc9 000002a7 490c6900 000241b0
Slot B: ffaa02a7 6c261bc9 000002a7 592e697f 0008c100 6c262bd0
Slot C: ffaa02a7 6c261bc9 000002a7 592e697f 0008c100 6c262cb0
Slot D: ffaa02a7 6c261bc9 000002a7 592e697f 0008c100 6c262bb0

This error cannot be skipped because header is corrupted.

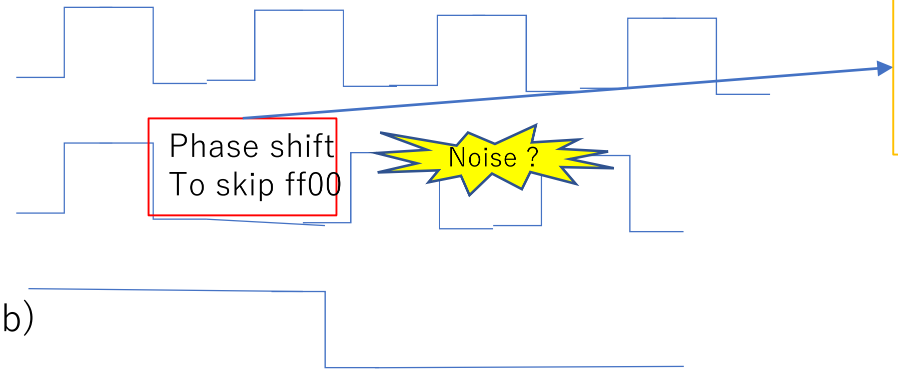
Workaroud to avoid ff00 data caused this new type of corruption

Clock for output Data from HSLB(fwclk)

No ff00 data case :

ff00 detected at the beginning of event

Output enable (fwenb)



This phase shift was moved
By a few clock ticks away from
Fwenb=enable.

How to deal with CRC error data

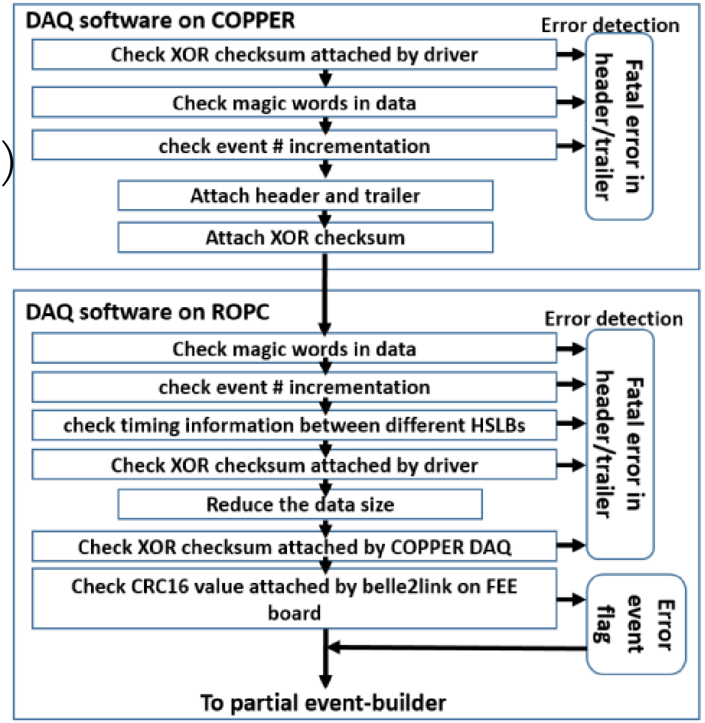
- ➢ B2link CRC error
- ➢ Corrupted data by SEU

If this error is unavoidable and happens occasionally, it is better not to stop runs due to these errors.
-> need to system to handle error events ?

A, Store error info in RawCOPPER header in ROPC
B, On HLT input node, DAQ program checks the error info in data from all COPPERs and store the info in “EventMetadata”.
C, HLT/analysis program check the info in EventMetadata and ignore error events.

Other fatal errors in header

- XOR checksum error (corruption in software)
- Event # jump
- Magic word in header/trailer is not correct
- Timing information difference between different FEE event fragments
->
Currently, DAQ is stopped



DAQ software on COPPER

Error detection

Check XOR checksum attached by driver

Check magic words in data

check event # incrementation

Attach header and trailer

Attach XOR checksum

Fatal error in header/trailer

DAQ software on ROPC

Error detection

Check magic words in data

check event # incrementation

check timing information between different HSLBs

Check XOR checksum attached by driver

Reduce the data size

Check XOR checksum attached by COPPER DAQ

Check CRC16 value attached by belle2link on FEE board

Fatal error in header/trailer

Error event flag

To partial event-builder