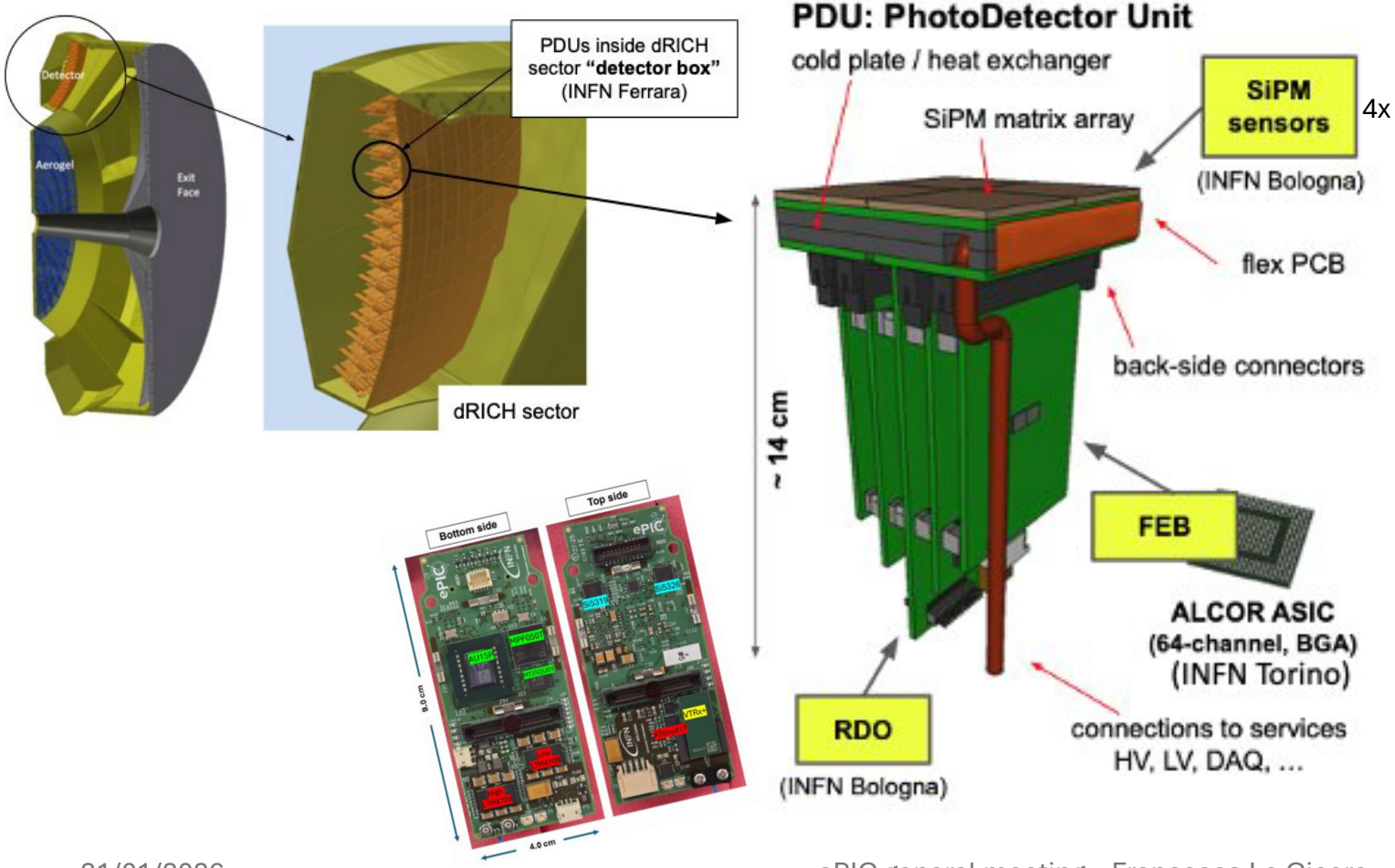
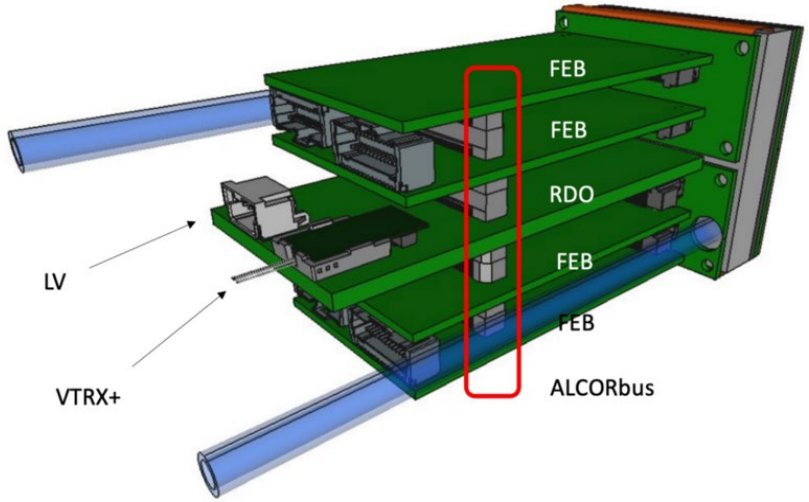
dRICH DAQ: from ALCOR to DAM

Francesca Lo Cicero
INFN Rome
on behalf of INFN ePIC Italia dRICH DAQ group

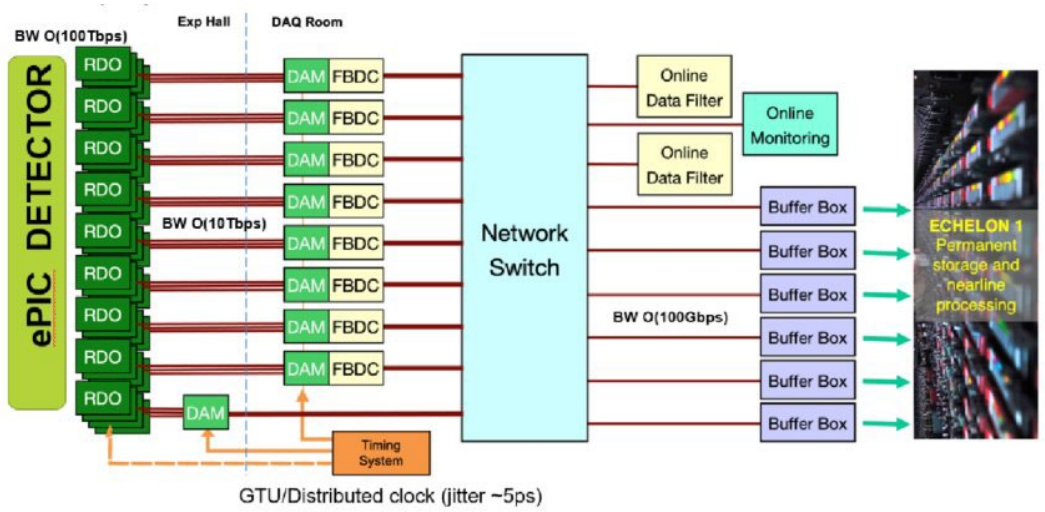
dRICH recap

The dRICH is made up of six identical open sectors, each containing Silicon PhotonMultipliers (**SiPMs**) clustered in several Photon Detector Units (**PDU**).

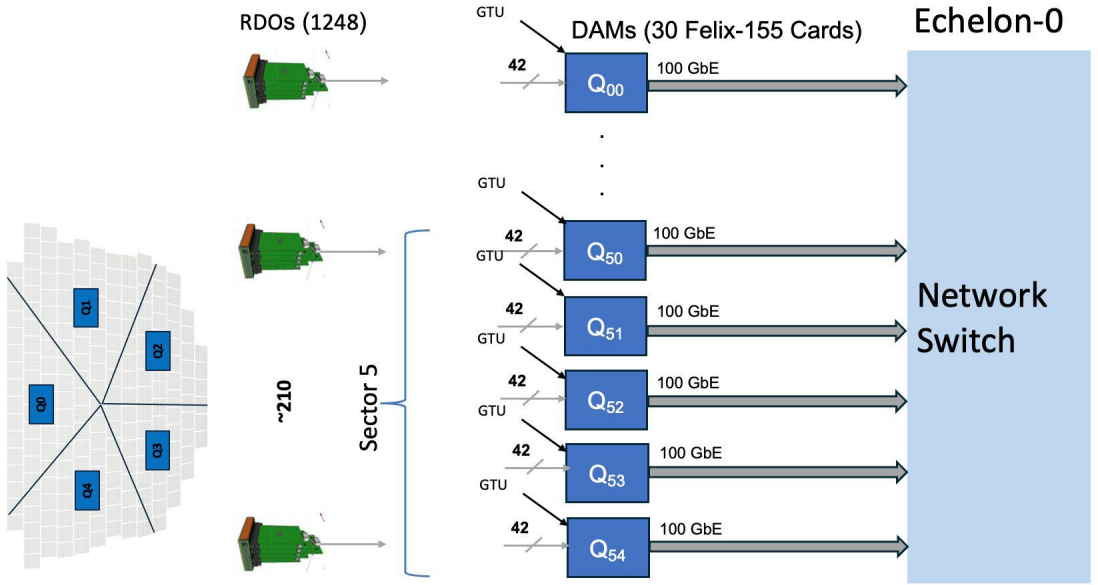




RDO/DAM role in ePIC and dRICH PDU



Front-End DAQ: RDO **Back-End DAQ**: DAM



Data Aggregation Module (DAM)

- Implemented on Felix board (INFN Rome)
- Aggregate data from 42 RDOs and transmit them to the ePIC buffering system (Echelon 0) through 100 GbE links.
- Provides computational capability for data handling and reduction

ALCOR ASIC

Main requirements for the ePIC dRICH detector

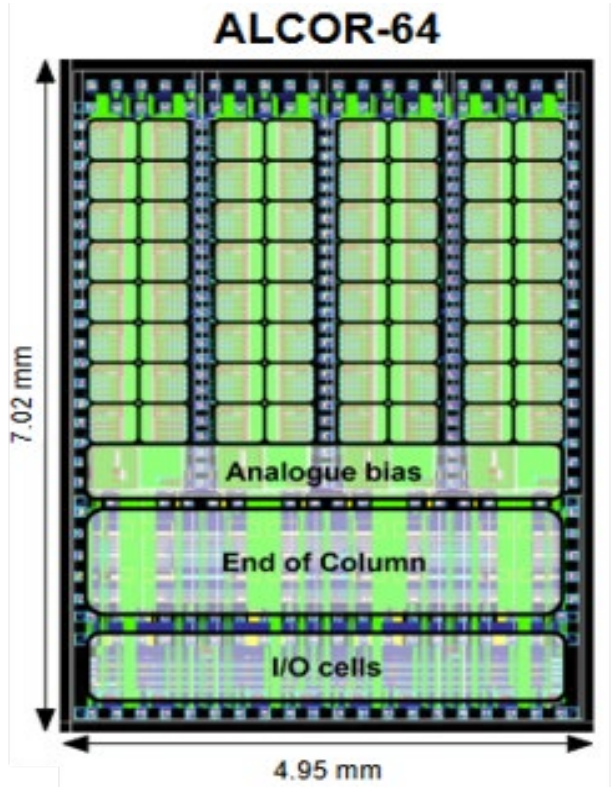
- Provide **single-photon time tagging** of signals coming from SiPM sensors
- **Cope with SiPM DCR**: 300 kHz/channel (at max SiPM radiation damage)

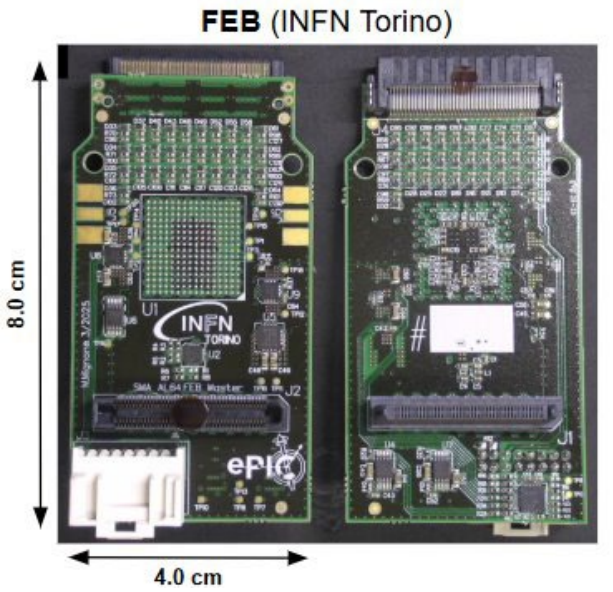
ALCOR Main features (ALCOR ASIC: F. Cossio @PD2025)

- **64-channels** (8x8 pixel matrix) mixed-signal ASIC providing **signal amplification** (fast front-end), **conditioning** (leading edge discriminator) and **digitization** (analogue interpolation TDCs)
- Measurements: **ToA** + ToT or **Slew-Rate** for **time walk compensation**
- **Triggerless readout** with fully digital output: 8 LVDS 394 MHz DDR Tx links
- Hardware **shutter** can inhibit event digitization **to suppress out-of-time SiPM DCR hits**
- Power consumption ~12 mW/channel, 110 nm CMOS technology
- Flip-chip BGA package: 17x17mm, 256 Ball, 1 mm pitch

MPW tapeout on Apr 2025, BGA packaging ongoing, first samples to be delivered soon (2026/Q1)

ALCOR update: F. Cossio @Electronics & DAQ WG meeting

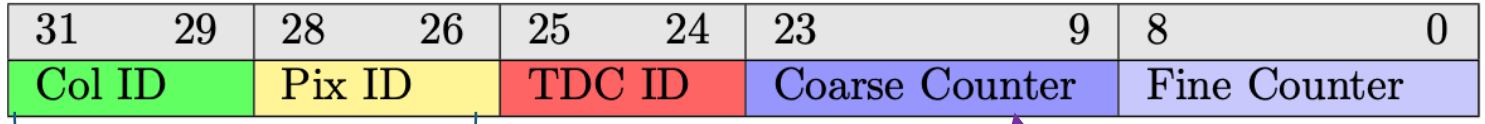




ALCOR data (raw)

- ALCOR hits "event word" are timestamps

accelerator Bunch Crossing: bits 23 – 11



3 bits to identify the column (LVDS TX link) [0 to 7]

3 bits to identify the pixel [0 to 7]

4 TDC IDs: 2 bits
→ 2 TDCs for leading edge
→ 2 TDCs for trailing/slew rate

These nine bits are the measurement of the TDC
You can reach 20-40 ps LSB.
Calibration needed → can't be used at DAM level

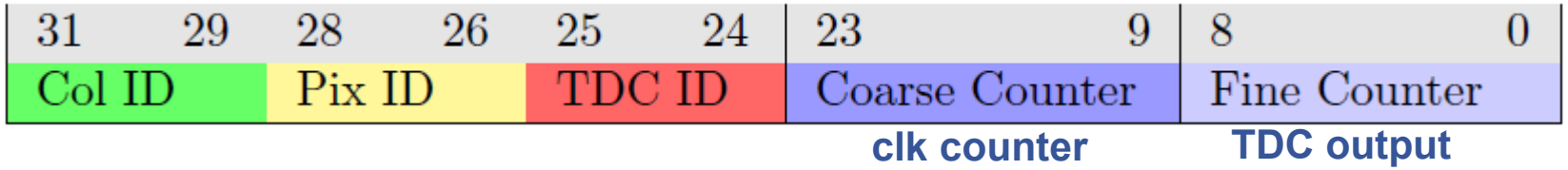
394 MHz → coarse counter LSB 2.54 ns (currently 320 MHz → 3.125 ns)
15 bit coarse counter (0x7FFF = 32767)
Coarse counter expires every 83.228 μs > 12.78 μs (EIC orbit)

At each EIC orbit we get a RevTick signal from DAM (main EIC "synch")
→ this trigger a coarse counter reset and a frame structure injected in data flow

- KC codes (8b10b codes) are used as separators/data encapsulators

ALCOR data format and transmission

ALCOR data-push architecture implies that 32-bit event words are "immediately" out on LVDS Tx links



Events are not transmitted perfectly ordered by time, mainly due to the **different TDC conversion time**

- Data from trailing edge (TDC1, TDC3) can be transmitted before data from leading edge (TDC0, TDC2), when operating in ToT or SlewRate mode
- Data from a given bunch can be transmitted after data of subsequent bunches have already been transmitted

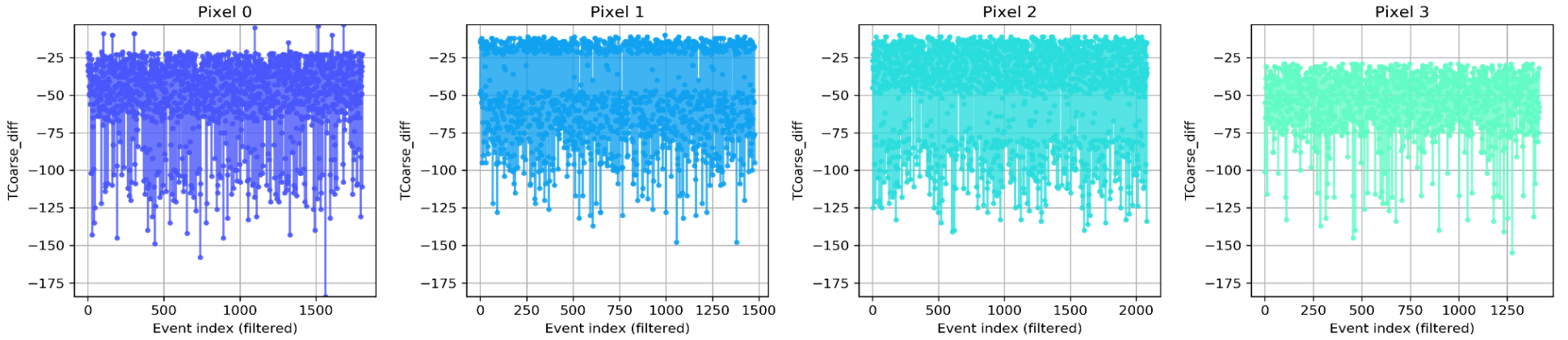
Global time reference is ensured by **data frames** structure: new frame (frame number, frame info, CRC) every **2^{15} clock cycles** or set by **external reset**

Data inside each frame will be time-ordered at FPGA-level using Coarse Counter bits
We performed several simulations on ALCOR data stream to study the effect of time-inverted data and understand how it must be addressed by the ePIC dRICH DAQ FPGA

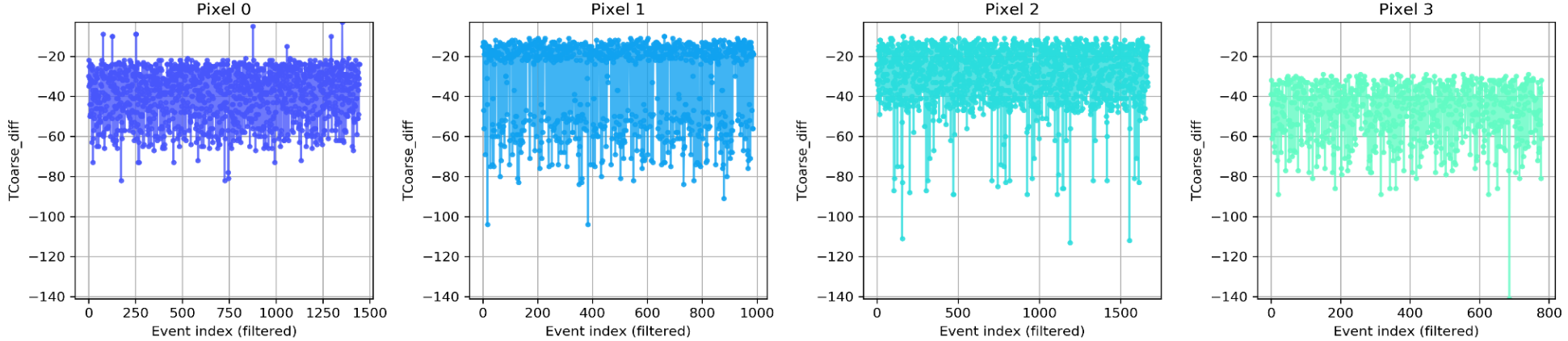
ALCOR simulations: time ordering

Time-inverted data mainly due to **TDC conversion time** which **depends on TDC interpolation factor (IF)** settings: 80-250 ns (IF=64), 160-500 ns (IF=128)

TDC IF = 128
ΔT range < 128 CLK



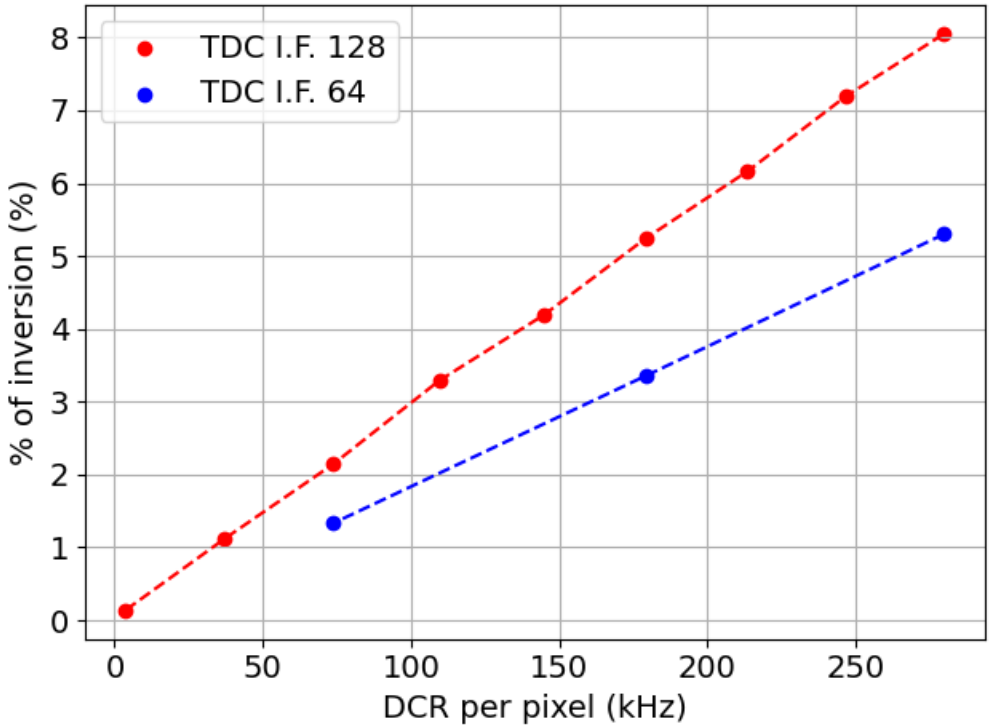
TDC IF = 64
ΔT range < 64 CLK



Need to buffer events up to 256 CLK cycles inside FPGA while performing time ordering

ALCOR simulations: time ordering

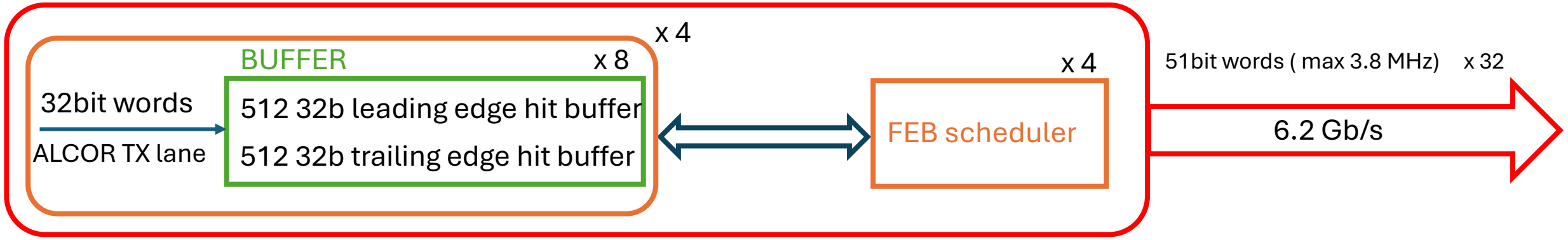
The time-inversion rate has been quantitatively estimated as the % of events with: timestamp[i] < timestamp[i-1]



- Inversion rate increases linearly with DCR
- Simulations with TDC interpolation factor (IF) **64** show a lower inversion rate due to TDC lower conversion time
- Simulations performed using gate width of 7.5 ns (Data reduction factor = 1.3)

Next step: use output data from these simulations as inputs for next elements in the dRICH DAQ chain (**RDO** and then **DAM**)

RDO buffering and "time ordering"?



BUFFER (1024 x 32 bit): → 1 Mbit full RDO:

- 512 leading words and 512 trailing words
- Leading/trailing edges waiting for matching
- 4.8 MHz : 300 kHz x 8 x 2 + ctrl words

Schedulers:

- Act on the occupancy and the timestamp to select the lane to be read → push data to buffer
- Inspect BUF and build 51bit Alcor Words (AWORDs)
- Completed 51b words are sent to the DAM via opt links ordered by BC

2026 Challenge: all this needs to be implemented (and validated!) in a "tiny" AU15P FPGA (5.1 Mbit BRAM)

51-bit **AWORD**: leading + trailing

50	49 48	47 46	45 39	38 30	29 27	26 24	23 22	21 9	8 0
K CODE FLAG	FEB ID	TDC ID (trailing)	Coarse (trailing)	Fine (trailing)	Col. ID	Pixel ID	TDC ID (leading)	Coarse (leading)	Fine (leading)

- K CODE FLAG: indicates whether the word should be interpreted as a K-code

Bit optimization performed:

- Using 13 instead of 15 bit for coarse counter range
- extracting Bunch Crossing Coarse counter of leading for trailing

*Take home message

We aim to send time ordered words (by BC) to the DAM (to ease data reduction)

How to transmit AWORD to DAM?

DAM Firmware: Reuse as much as possible of the “official” FELIX firmware release.
https://gitlab.cern.ch/atlas-tdaq-felix/firmware.git

FELIX Firmware

- FELIX supports different link protocols for data transfer
- The firmware flavour is defined at built time

Protocol	FELIX
GBT	TX: 4.8G, RX: 4.8G
FULL	TX: 4.8G, RX: 9.6G
lpGBT	TX: 2.56.8G, RX: 10.24G

GBT :

- Protocol which provides radiation-tolerant transport of data
- Aggregate multiple lower bandwidth logic links (e-links) in one high bandwidth data link

FULL :

- Use the GBT protocol for the links towards the FE (downlink)
- To receive data from the FE (uplink) implements a light-weight protocol (with a bandwidth of 9.6 Gb/s per optical link - 7.68Gb/s with 8b10b encoding).

lpGBT :

- Evolution of GBT
- Two FEC codes can be used (FEC5/FEC12)

FEC5 **Uplink**

	5.12 Gb/s	10.24 Gb/s
Option	7	7
Frame (bits)	**128**	**2 x 128**
Header (bits)	**2**	**2+2**
Coded header	**no**	**no**
User field (bits)	116	232
Code (bits)	10	20
16-bit multiplicity	7.250	14.5
Remainder bits	4	8
User Bandwidth (GHz)	4.64	9.28
# eLinks groups (16 bit)	7	7
eLinks bandwidth (MHz)	160/320/640	320/640/1280
#eLinks	28/14/7	28/14/7
EC bandwidth (MHz)	80	80/160
IC bandwidth (MHz)	80	80/160
Unassigned bits	0	4
Corrected (bits)	**5**	**2 x 5**
Efficiency	**91%**	**91%**

How to transmit AWORD to DAM?

Protocol Bandwidth (Gb/s)

Protocol	FELIX
GBT	TX: 4.8G, RX: 4.8G
FULL	TX: 4.8G, RX: 9.6G
lpGBT	TX: 2.56.8G, RX: 10.24G

Design Assumption:

- RDO/DAM BW: at most 6.2Gb/s
- RDO: Use of VTRx+ Optical Link Module for Data Transmission, w/o lpGBT ASIC

FULL :

- Less FPGA resources hungry with respect to lpGBT
- **Do not provide radiation-tolerant data transport**

lpGBT FEC5 :

- Provides **radiation-tolerant** transport of data
- FEC5 Bandwidth: 10.24Gb/s with 91% efficiency
- Scaling up to 48 lpGBT in FLX-155 could be a problem **without modifying the standard FELIX FW**
- Division of physical-link in virtual-links (e-links) is not necessary but **standard FELIX FW doesn't support single e-link configuration**

FELIX **Estimated** resource utilization

		KU115	VM1802	VP1552
GBT 24 channel	LUT	80.65%	69.60%	35.71%
	FF	77.03%	50.94%	26.13%
	BRAM	70.00%	89.45%	34.04%
	URAM		62.20%	22.14%
FULL 24 channel	LUT	52.59%	44.35%	22.75%
	FF	38.40%	33.21%	17.03%
	BRAM	40.46%	20.99%	7.99%
	URAM		62.20%	22.14%
LPGBT 24 channel	LUT	112.51%	82.94%	42.55%
	FF	52.39%	38.62%	19.81%
	BRAM	68.94%	79.52%	30.26%
	URAM		62.20%	22.14%

https://atlas-project-felix.web.cern.ch/atlas-project-felix/user/docs/FELIX_Phase2_firmware_specs.pdf

FW customization required

Discussion about these choices is ongoing with ALICE people …

Link protocol

DISCLAIMER NOTICE: all this is VERY 

RDO "prepares" data reduction
DAM "does" data reduction

opt-link **protocol** RDO

AWORD

50	49 48	47 46	45 39	38 30	29 27	26 24	23 22	21 9	8 0
K CODE FLAG	FEB ID	TDC ID (trailing)	Coarse (trailing)	Fine (trailing)	Col. ID	Pixel ID	TDC ID (leading)	Coarse (leading)	Fine (leading)

	DF3	DF2	DF1	DF0	DCS (free)	AWORD	AWORD	AWORD	AWORD
FULL	255	254	253	252	251 204	203 153	152 102	101 51	50 0
lpGBT	223	222	221	220	219 204	203 153	152 102	101 51	50 0

link protocol operated at **39.4 MHz** CLK
(5/2 of EIC clock, close to LHC clock)

Protocol over optical link:
- FULL (256 bits/clk)
- lpGBT (224 bits/clk with FEC5)
- Data Flag
- Space for DCS bus (SWT à la ALICE-ITS2)

lpGBT/FULL to be compared under radiation

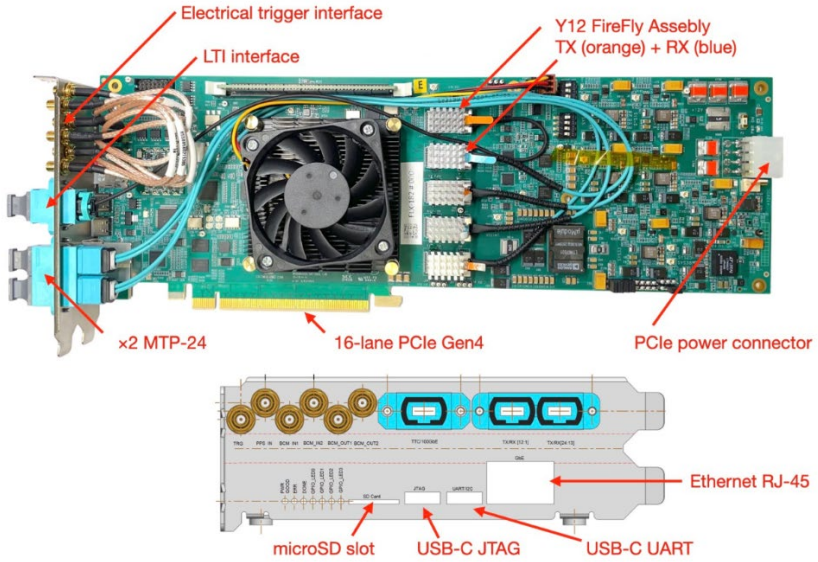
FULL is our current baseline choice

DAM
- extracts BC from bits 21-11
- extracts geographic position from RDOID (DAM is link-aware) + FEBID (49-48) + AlcorCh (29-24): geo info

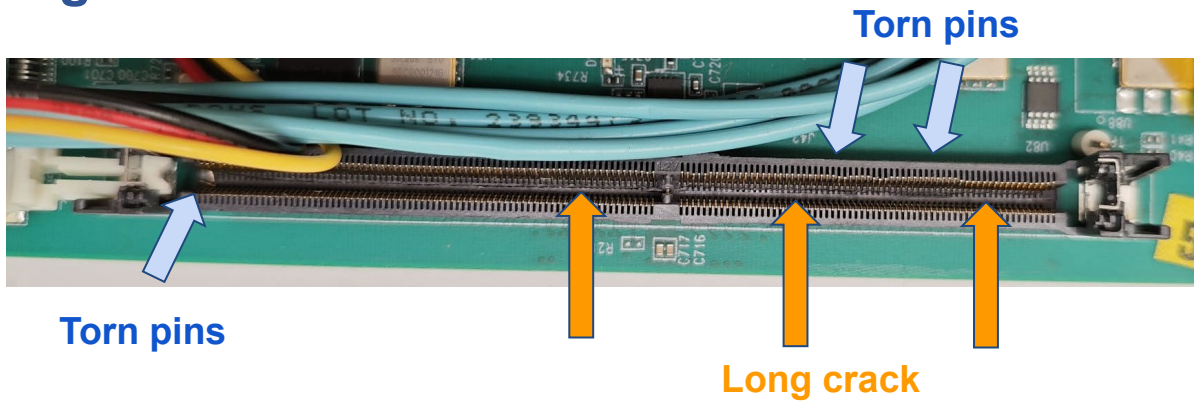
→ data pattern
- NN decision is reached, the DAM must add to the 51-bit word the 11 bits of the RDOID → 62-bit word
- 2 bits added → PCI 64-bit words → dRWORDS (dRICH words)
- dRWORDS out in timeframes to StreamingReadOut computing

DAM prototype on FLX182

Board procurement @Rome



- Felix-182 board from JLab arrived in Rome end of December ’24.
- It showed damages to the DRAM slot:
  - **torn contact pins**
  - **a crack along the inner side of the slot**



We consulted:

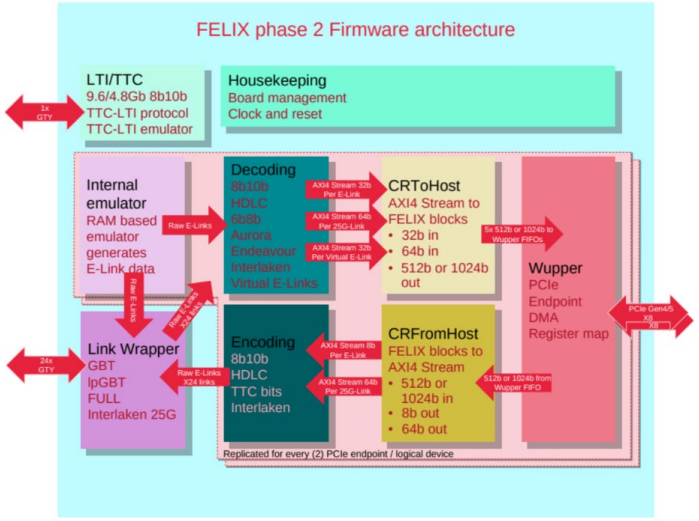
- the INFN Electronics Laboratory in Rome
- CERN EP-ESE Electronic Systems for Experiments (thanks Markus Joos!)

Both agreed that is extremely difficult to repair the DRAM slot and not worth the effort considering the costs and the likely not optimal result…

DAM prototype on FLX182

Firmware

- Loading Felix official firmware onto the board failed due to a DRAM check error.
  - In current Felix FW versions the PS performs just ancillary functions (BIST,…), while all the design core functionalities are implemented in the PL.
  - We forced Versal Platform Management Controller (PMC) to ignore DRAM status check at start up.



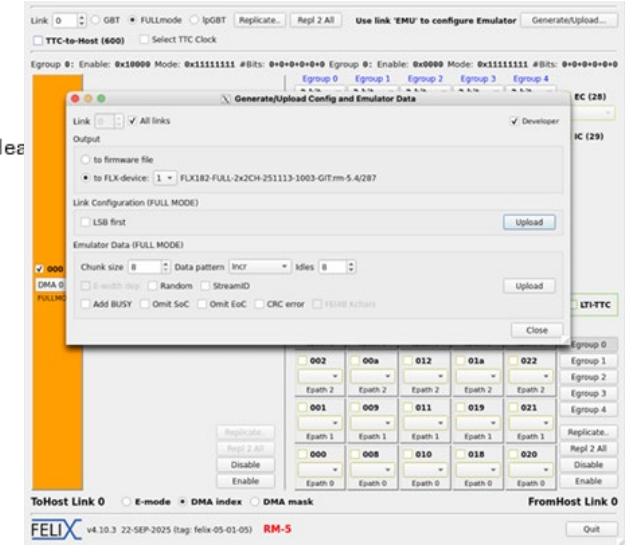
FW loaded and board detected!

```
[locicero@apestation0 ~]$ lspci -vvvvvv | grep CERN
34:00.0 Network controller: CERN/ECP/EDU Device 0428
        Subsystem: CERN/ECP/EDU Device 0038
35:00.0 Network controller: CERN/ECP/EDU Device 0427
        Subsystem: CERN/ECP/EDU Device 0038
```

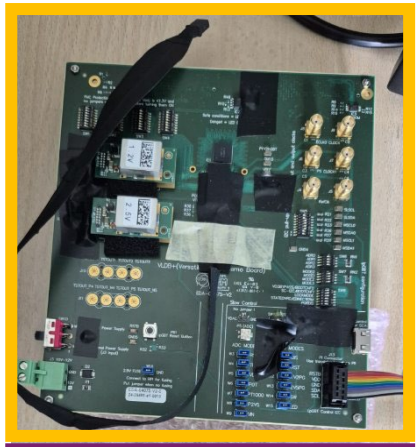
- Exercise the full software/firmware stack using the internal data generators to instantiate DMAs towards the host memory

```
lonardo@apestation0 x86_64-el9-gcc13-opt]$  bin/flx-init
2025-05-14 18:06:05 Opening card 0 (device 0)...
2025-05-14 18:06:05 Card type: FLX-182
2025-05-14 18:06:05 Firmware : FULL
2025-05-14 18:06:05 Clock    : Local
```

```
Consume FLX-device data while checking the data (blockheader and trailers),
counts errors including chunk truncation, halts when the memory buffer is near overflowing.
Also counts chunk CRC errors.
Opened FLX-device 0, FLX182-FULL-2x2CH-251113-1003-GIT:rm-5.4/287, trailer=32bit ChunkHea
*START(emulator)* using DMA #0 polling
 Secs | Recvd[MB/s] | File[MB/s] | Total[(M)B] | Rec[(M)B] | Buf[%] | Wraps
-------|------------|------------|-------------|-----------|--------|-------
   1     241.4      0.0      241.4       0     0     0
   2     241.3      0.0      482.6       0     0     0
...
  15     241.3      0.0     3618.8       0     0     3
*STOP*
-> Data checked: Blocks 3533027, Errors: header=0 trailer=0
Exiting..
```



Testbed (I) @Rome

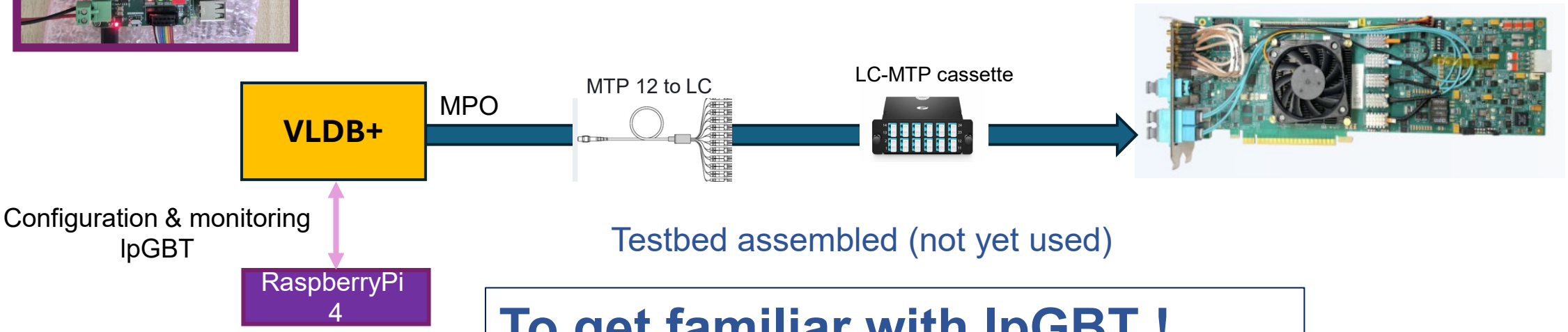


VLDB+ : board designed by CERN's EP-ESE group to provide an evaluation kit for the new Versatile Link ecosystem.

- On board:
  - lpGBT,
  - VTRx+ connector
  - Rad-tol DC/AC regulators
  - FMC connectors for prototyping with FEB or FPGA development kit

Thanks to INFN Bologna!

- lpGBT configuration through **RaspberryPi 4** and piGBT web application.



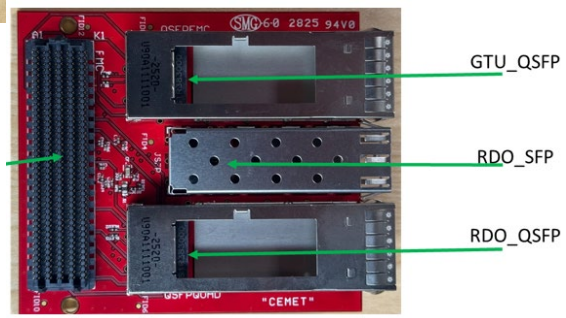
Testbed assembled (not yet used)

To get familiar with lpGBT !

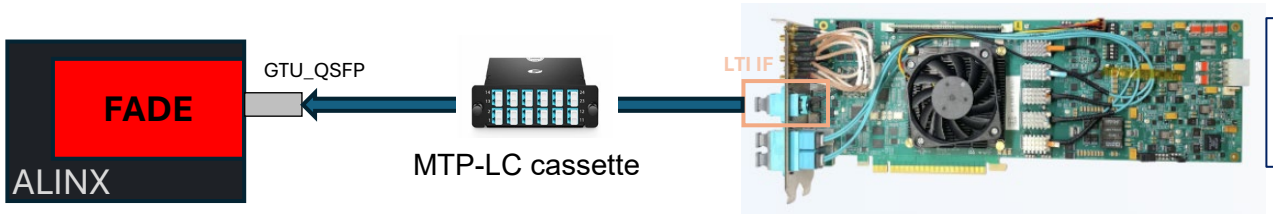
Testbed (II) @Rome

FMC Adaptor card for DAM Emulation (FADE) module

- **Plugged** onto the AXAU15, an AMD Artix UltraScale+ FPGA development **board** developed by **ALINX**.



  - **FADE in Master mode:** FADE sends the System clock (19.7 MHz) and the run control (7.88 Gbps) to a DAM and receives the 'DAM' status via the GTU_QSFP.

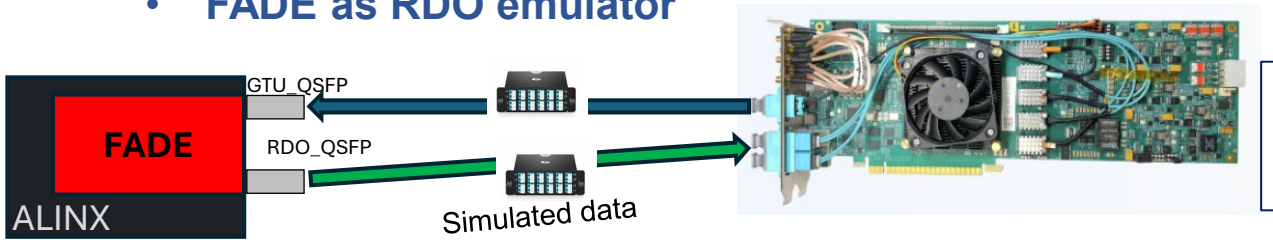


FADE implements some GTU functionality

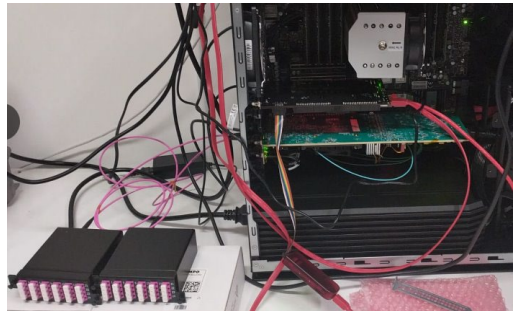
GTU_QSFP

FW to be done

  - **FADE as RDO emulator**



Test RDO/DAM connection



Testbed assembled (not yet used)

Conclusion

ALCOR64
- MPW tapeout completed in April 2025; first samples expected in Q1 2026
- Several simulations of the ALCOR data stream performed to study the impact of time-inverted data

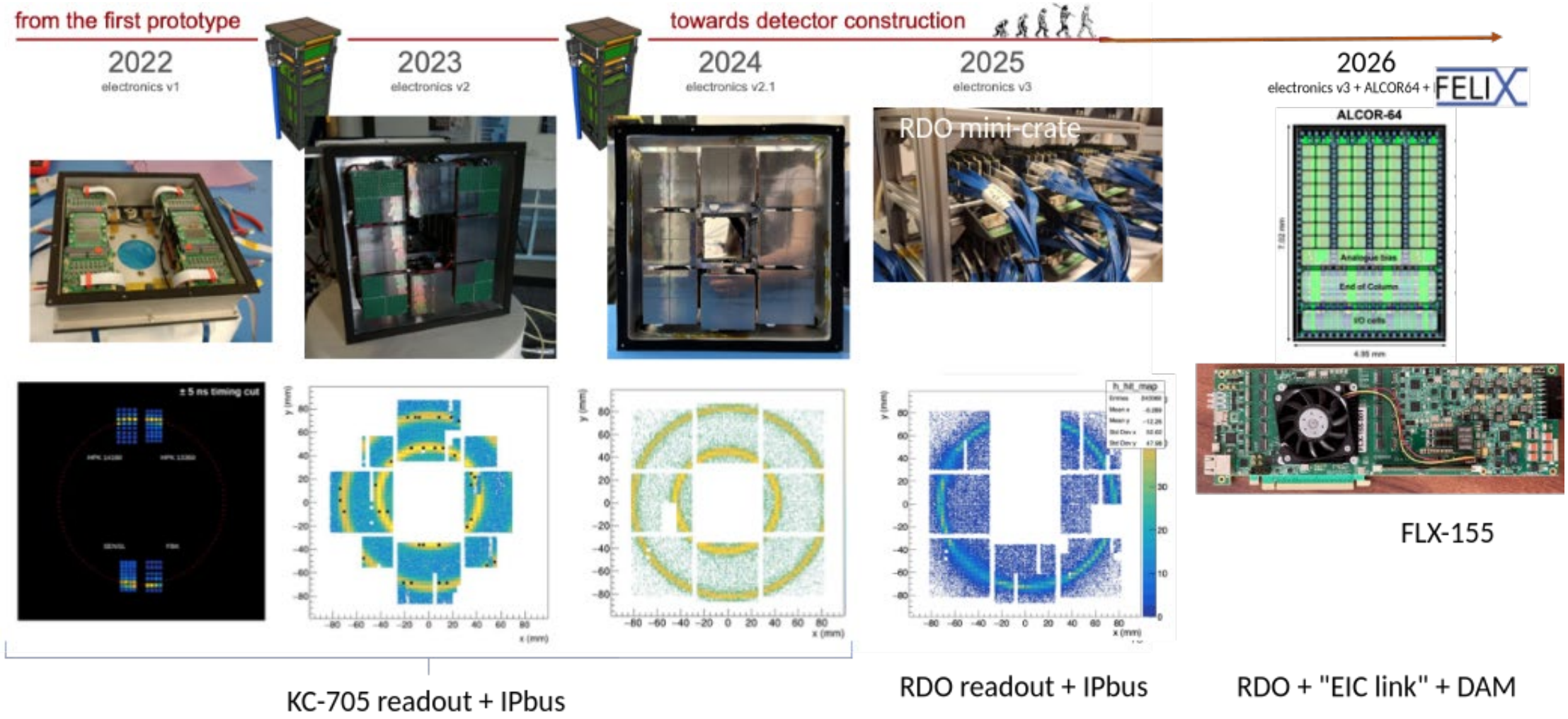
RDO
- RDO validated and used during the November 2025 test beam
- On-going studies on data buffering and time ordering for transmission to DAM

DAM
- FLX182 board procured and installed on a dedicated server, together with ancillary hardware
- On-going studies/test on FELIX firmware and software stack (FULL, GBT and lpGBT flavours)

Future Work
- Use ALCOR simulation output data as input for the next elements of the dRICH DAQ chain (RDO → DAM)
- Implement and validate RDO-level time ordering on the AU15P FPGA
- Perform DAM Tests
  - Optical loopback tests using MTP–LC breakout cable and patch panel
  - Use Alinx + FADE to emulate the GTU module for clock and run-control distribution to DAM
  - Investigate the lpGBT protocol using the VLDB+ board

Backup

dRICH testbeam



Towards full data-push architecture: time ordering

The problem: ALCOR timestamps are not "*immediately*" out

Pixel TDC conversion defines time required for digitization:

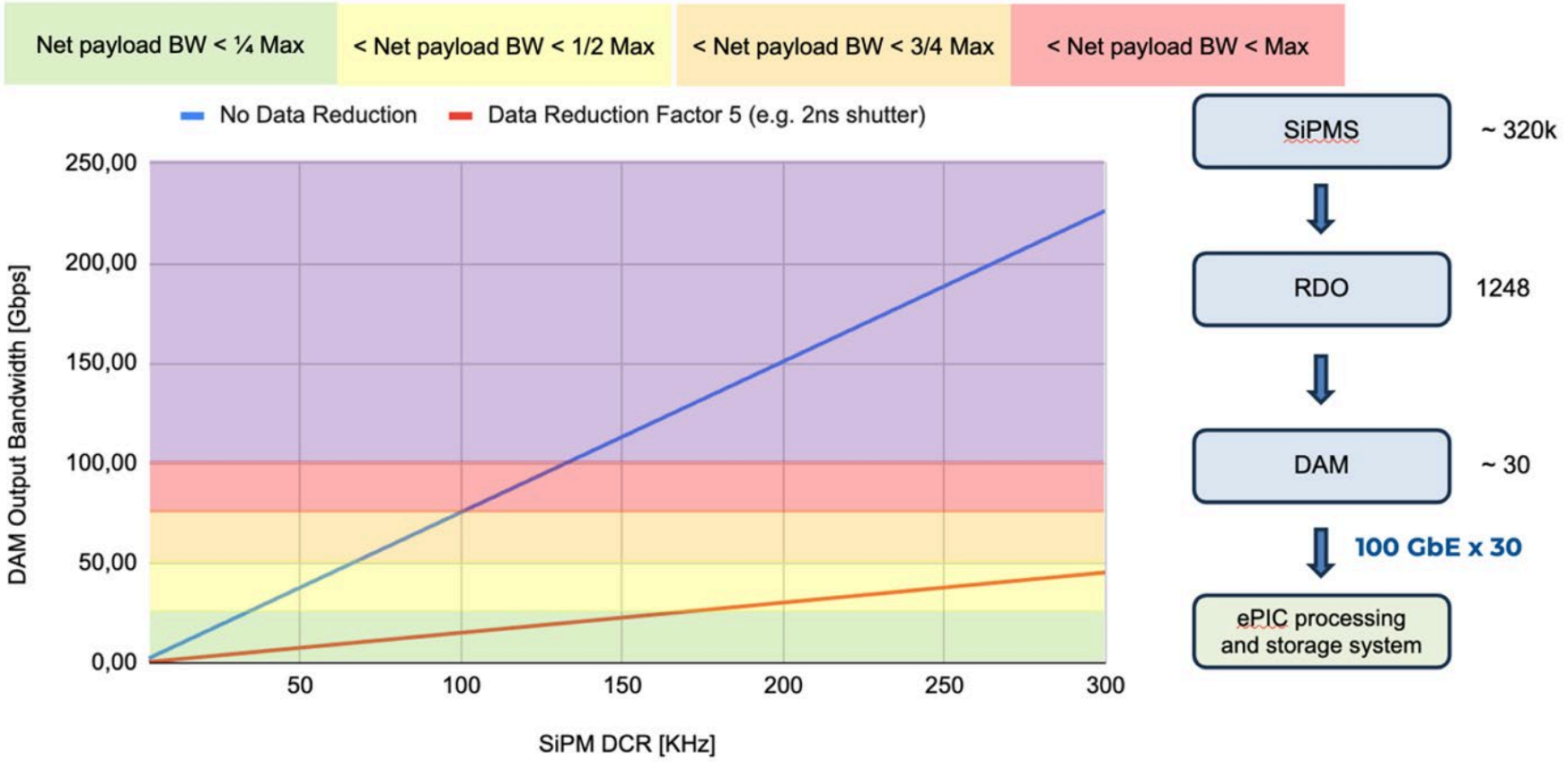
- TDC max conversion time = 1.5 x 128 = 192 clock cycles ≃ 500 ns
- TDC min conversion time = 0.5 x 128 = 64 clock cycles ≃ 170 ns
- Max ΔT = 500 ns - 170 ns = 330 ns (inside ALCOR channel) → down to the column and transmit off-chip

ALCOR data transmission:

- 1 word: 40 bits → 40 b / 788 Mb/s = 51 ns per event word (we have 8 channels for each Tx link)
- If no other hits in the column, 2 hits (separated by 128 clk cycles in the pixel) are only separated by idle words (K.28.5 "comma" are filtered by RDO) → [dependency on data rates
- Time reference given by ALCOR frame structure → @DCR = 300 kHz/ch, Δtframe = 12.7886 μs → Nevents = 8 · DCR · Δtframe = 30.7
- 31 event words per frame (mean value, 2x in ToT mode, add physics)

→ can hits be sent time ordered (BC) by RDO to ease DAM work?

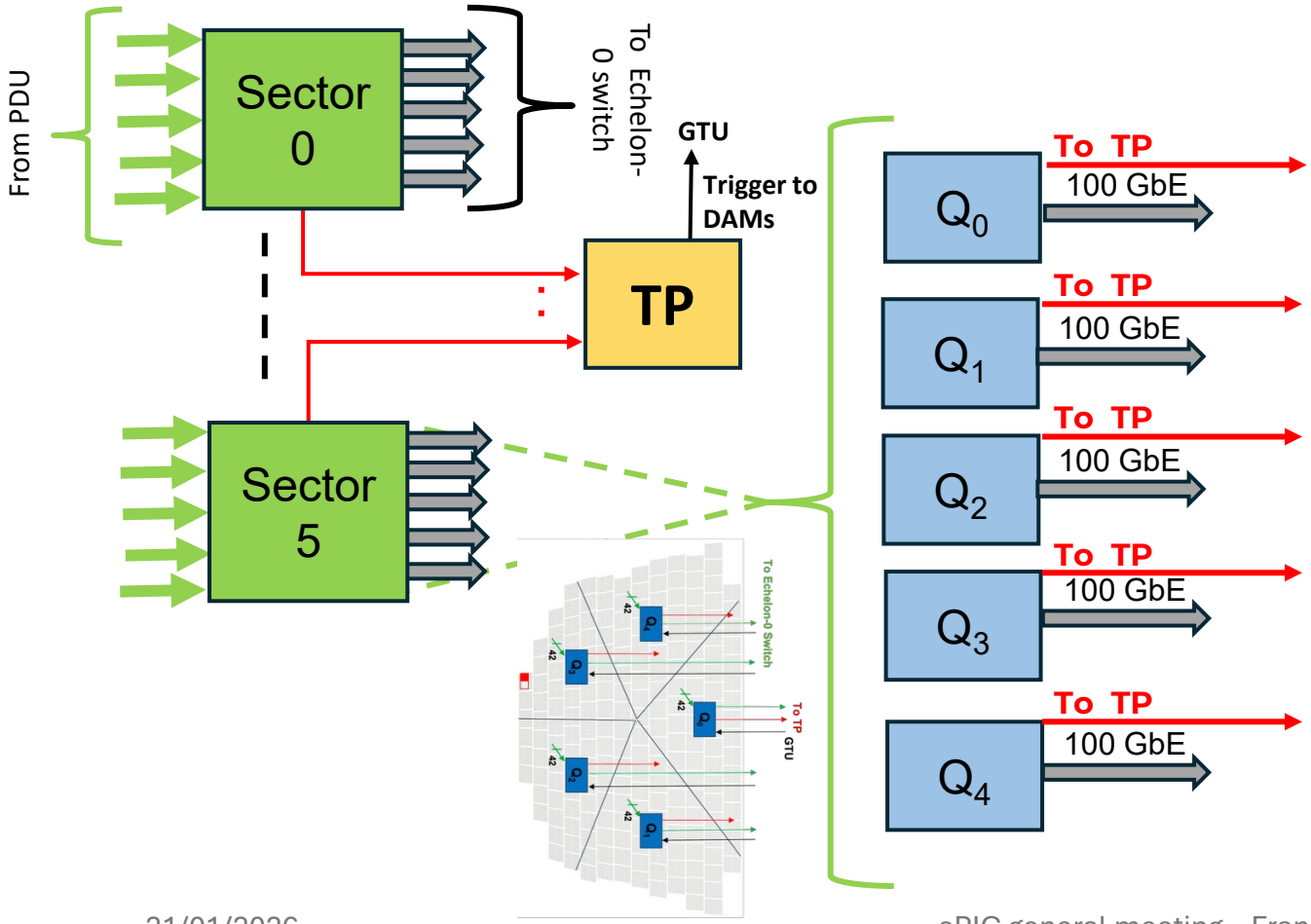
dRICH output bandwidth

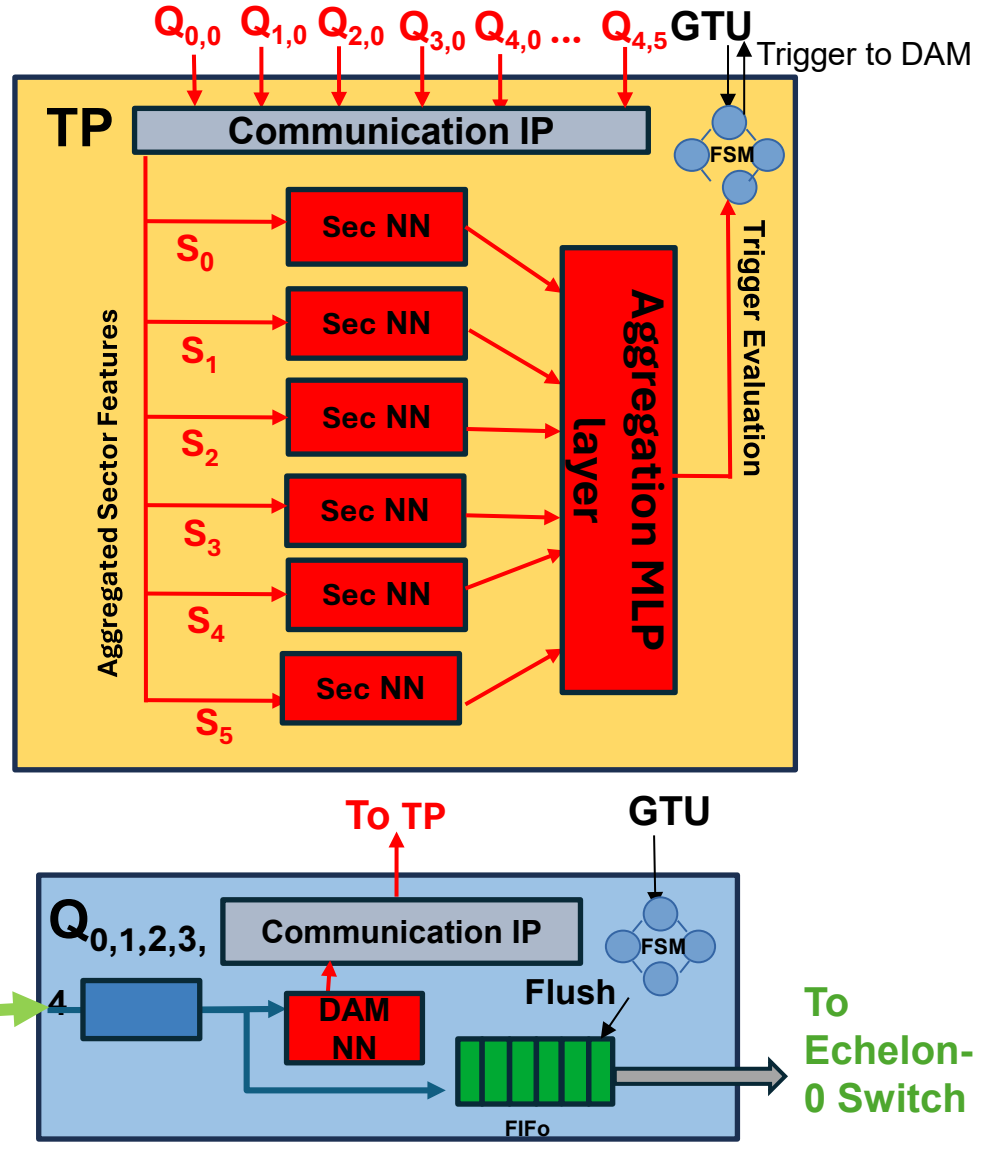


- Without data reduction with a 100 kHz DCR we are close to DAM bandwidth limit
  - A data reduction factor 5 allows us to stay safe up to the 300 kHz limit
  - Data reduction factor five can be achieved via shutter or provided by NN or ext. trigger or a combination of them.

dRICH data reduction integrated in the DAM

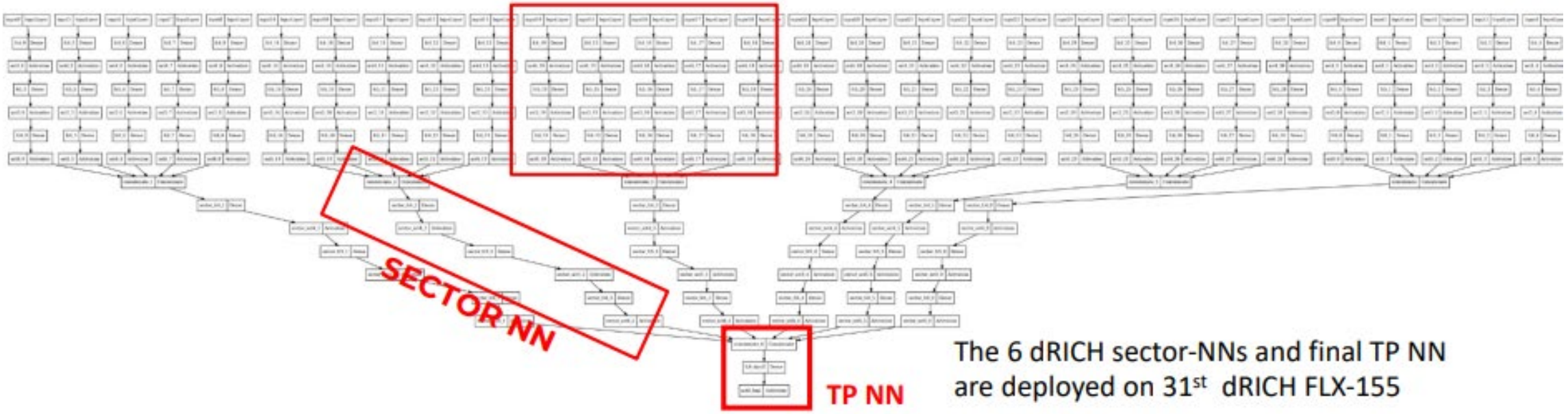
Neural network distributed over 31 Felix-155





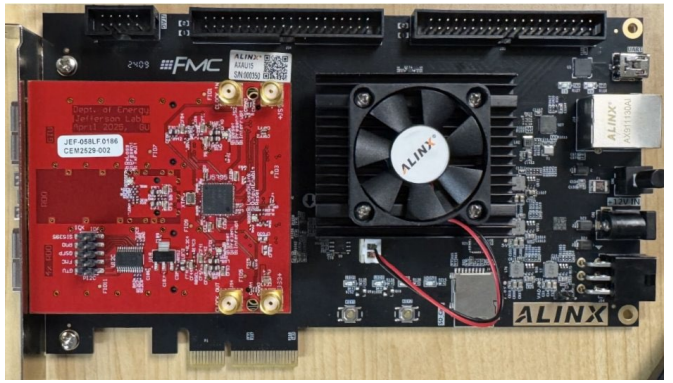
NN deployment model and DAMs

6 dRICH sectors – 5 DAMs/sectors



The 6 dRICH sector-NNs and final TP NN are deployed on 31st dRICH FLX-155

FADE module

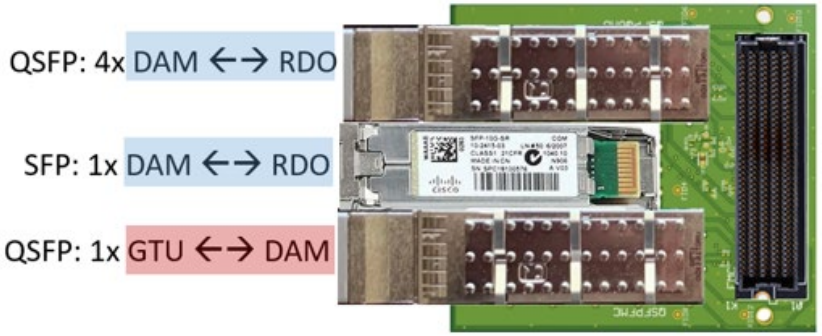


FMC Adaptor card for DAM Emulation (FADE) module

- Designed for the ePIC detector test data readout and the incremental DAQ software development.
- Plugged onto the AXAU15, an AMD Artix UltraScale+ FPGA development board in PCIexpress card format and developed by ALINX.

Slave mode: FADE receives System clock (19.7 MHz) and the run control (7.88 Gbps), it sends the 'DAM' status via the GTU_QSFP connector from/to a GTU. It performs as a DAM emulator.

Master mode: FADE sends the System clock (19.7 MHz) and the run control (7.88 Gbps), it receives the 'DAM' status via the GTU_QSFP connector to/from a DAM. Meanwhile, these are 'loopback', so the FADE still performs as a DAM emulator. The master mode just means the FADE implements some GTU functionality.



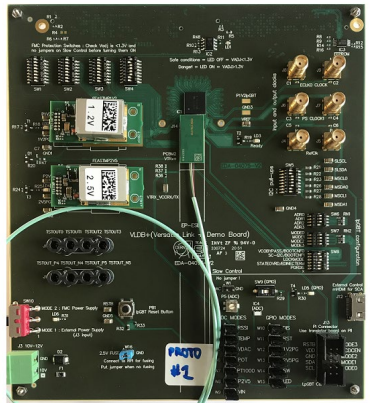
The GTU_QSFP is the QSFP connector to GTU (in slave mode), or DAM (in master mode).

VLDB+

- **lpGBT: Higthly configurable ASIC**
  - Mode of operation (TX/RX/TX-RX)
  - Uplinkdata rate
  - FEC option
  - Number of eLINK
  - …

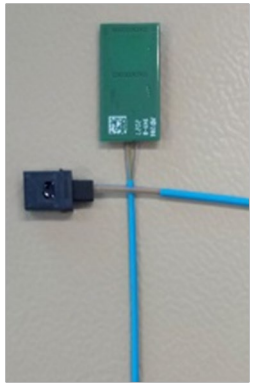
11 pins and 320 registers need to be taken care of

VLDB+: board designed by CERN's EP-ESE group to provide an evaluation kit for the new Versatile Link ecosystem.



- On board:
  - LpGBT (
    - configuration and monitoring through the RaspberryPi 4 and piGBT web application.,
  - VTRX+
    - only one Tx link is connected to the VTRx+)
  - Rad-tol DC/AC regulators
  - FMC connectors for prototyping with FEB or FPGA development kit

VTRx+ module



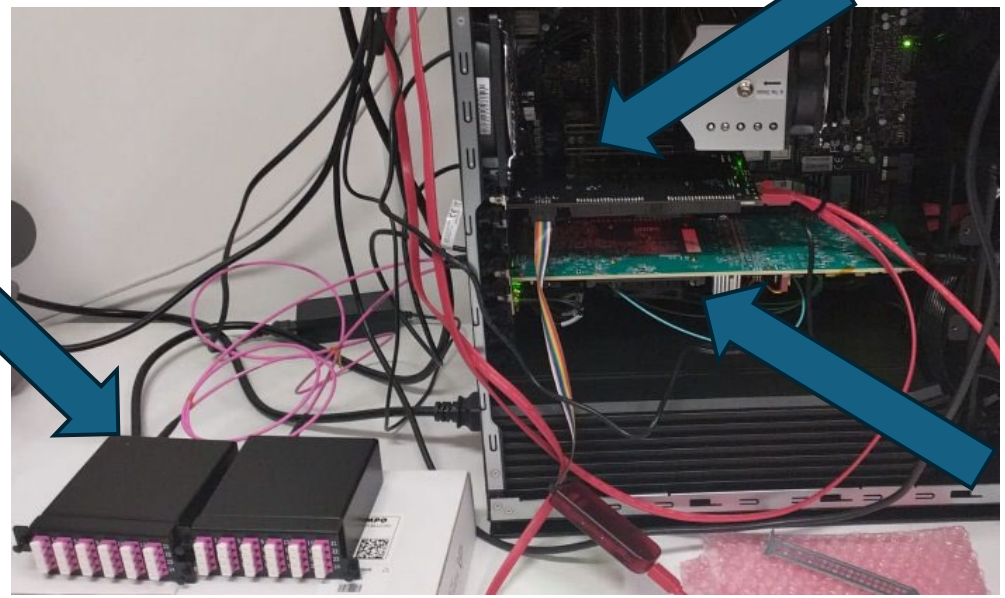
MT connector

MPO connector

Testbed (II) @Rome

AMD FPGA + FADE board

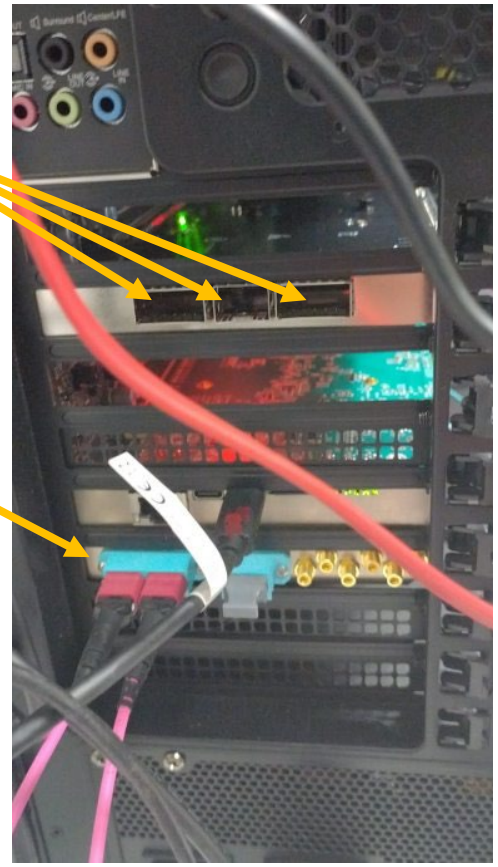
2 x cassette MTP
To 24 LC ports



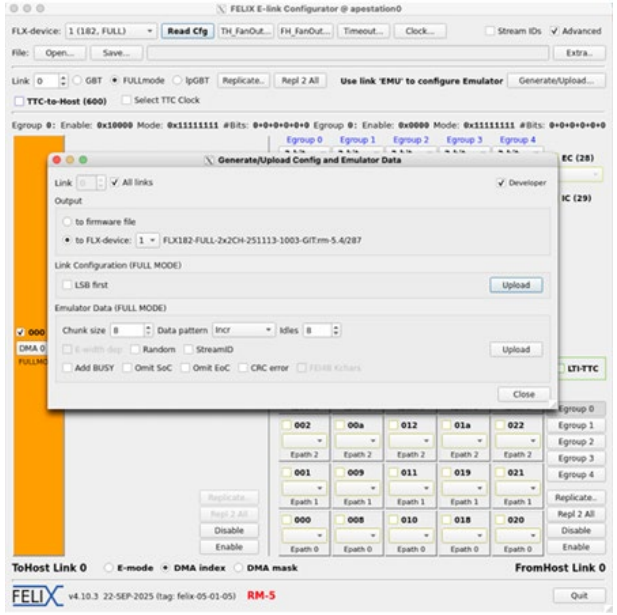
FLX-182

FADE 2 qsfp+
ports 1 sfp port

FLX-182 2 MTP
TX/RX 24
channels ports



DAQ back-end FLX182 FULL flavour tests



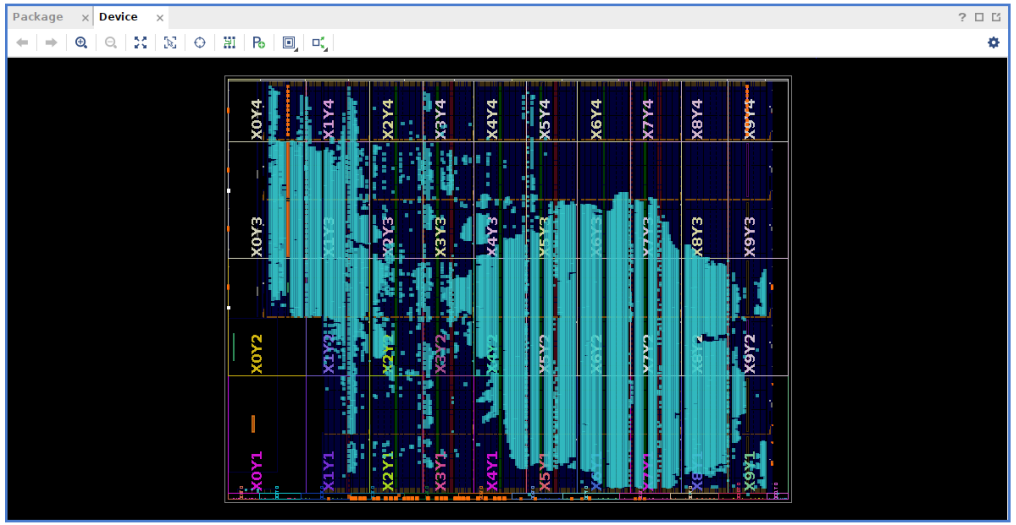
[lonardo@apestation0 ~]$ fdaq -d 0 -t 15 –e movesi
Consume FLX-device data while checking the data (blockheader and trailers),
counts errors including chunk truncation, halts when the memory buffer is near overflowing.
Also counts chunk CRC errors.
Opened FLX-device 0, FLX182-FULL-2x2CH-251113-1003-GIT:rm-5.4/287, trailer=32bit ChunkHeaders, buffer=1024MB, DMA=0
START(emulator) using DMA #0 polling

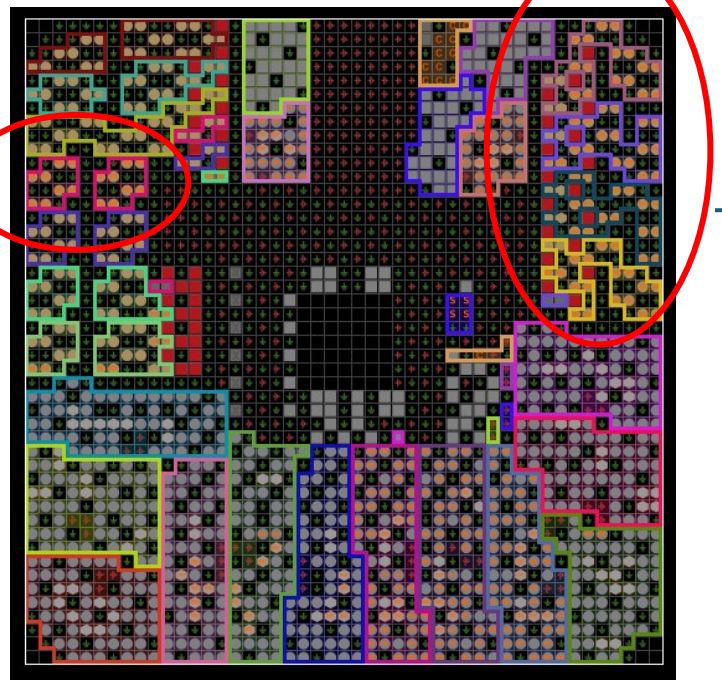
Secs	Recvd[MB/s]	File[MB/s]	Total[(M)B]	Rec[(M)B]	Buf[%]	Wraps
1	241.4	0.0	241.4	0	0	0
2	241.3	0.0	482.6	0	0	0
...						
15	241.3	0.0	3618.8	0	0	3

STOP
-> Data checked: Blocks 3533027, Errors: header=0 trailer=0
Exiting..

[lonardo@apestation0 ~]$ hexdump movesi-251121-122848-1.dat | head –16
0000000 0000 c0cf 0008 6000 00aa 0000 bb00 10aa 0000010 0008 6000 00aa 0100 bb00 10aa 0008 6000
0000020 00aa 0200 bb00 10aa 0008 6000 00aa 0300 0000030 bb00 10aa 0008 6000 00aa 0400 bb00 10aa
0000040 0008 6000 00aa 0500 bb00 10aa 0008 6000 0000050 00aa 0600 bb00 10aa 0008 6000 00aa 0700
0000060 bb00 10aa 0008 6000 00aa 0800 bb00 10aa 0000070 0008 6000 00aa 0900 bb00 10aa 0008 6000
0000080 00aa 0a00 bb00 10aa 0008 6000 00aa 0b00 0000090 bb00 10aa 0008 6000 00aa 0c00 bb00 10aa
00000a0 0008 6000 00aa 0d00 bb00 10aa 0008 6000 00000b0 00aa 0e00 bb00 10aa 0008 6000 00aa 0f00
00000c0 bb00 10aa 0008 6000 00aa 1000 bb00 10aa 00000d0 0008 6000 00aa 1100 bb00 10aa 0008 6000
00000e0 00aa 1200 bb00 10aa 0008 6000 00aa 1300 00000f0 bb00 10aa 0008 6000 00aa 1400 bb00 10aa

FELIX Firmware – GBT flavour
Synthesis results for 4 channels





4 ch GBT
4TX 4RX

Pci 2x8 gen3

Name	Registers (1799680)	CLB LUTs (899840)	LUT as Logic (899840)	LUT as Memory (449920)	LOOKAHEAD8 (112480)	SLICE (112480)	CLB Registers (1799680)	Block RAM Tile (967)	URAM (463)	Bonded IOB (692)	GTYE5_QUAD (11)	PCIE 40E5 (4)
N felix_top	16.99%	24.43%	23.69%	1.48%	2.14%	41.38%	16.99%	24.46%	43.63%	28.32%	54.55%	50.00%

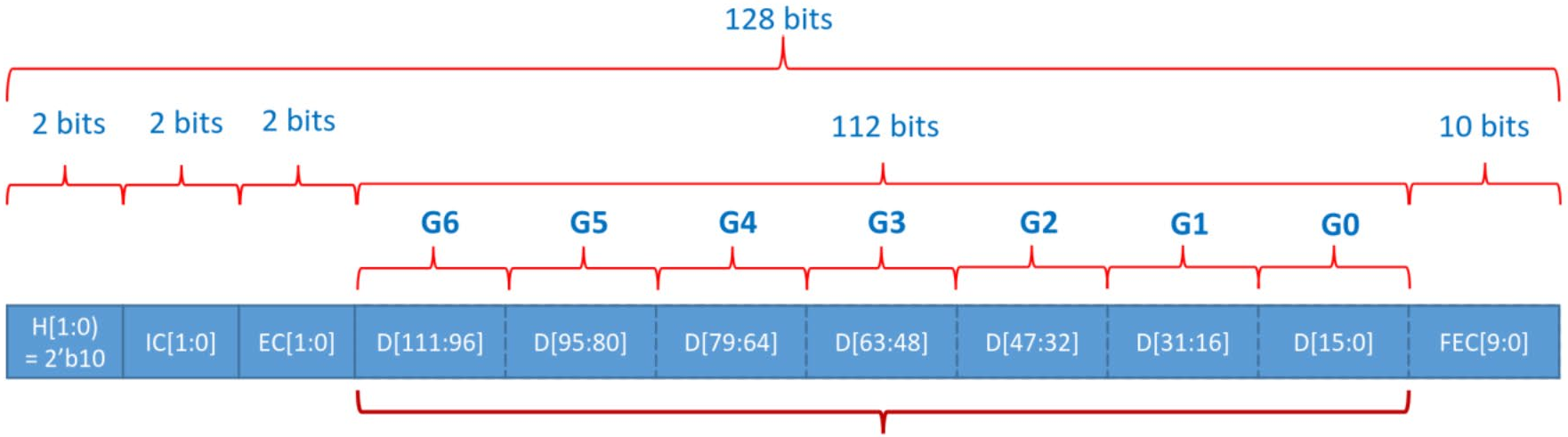
Uplink – FEC 5

- Dual data rate:
  - 5.12 Gb/s
  - 10.24 Gb/s
- Frame:
  - Header:
    - 2 / 4 – bits
    - 2'b10 (high) & 2'b01 (low)
  - IC field:
    - 2 – bit
    - 80 Mb/s
  - EC field:
    - 2 – bit
    - 80 Mb/s
  - Data field:
    - 112 / 224 – bit
    - 4.48 / 8.96 Gb/s
  - FEC field:
    - 10 / 20 – bit
- Error correction:
  - FEC:
    - Interleaving: 1 / 2
    - Symbol width: 5 – bit
    - Number of wrong symbols: 1 (× 1 / × 2)
  - Up to 5 / 10 consecutive bits
  - Efficiency: 91%
- eLinks:
  - Data
    - 28 eLinks @ 160 / 320 Mb/s
    - 14 eLinks @ 320 / 640 Mb/s
    - 7 eLinks @ 640 / 1280 Mb/s
  - EC
    - 1 eLink @ 80 Mb/s

FEC5

	5.12 Gb/s	10.24 Gb/s
Option	7	7
Frame (bits)	**128**	**2 x 128**
Header (bits)	**2**	**2+2**
Coded header	**no**	**no**
User field (bits)	116	232
Code (bits)	10	20
16-bit multiplicity	7.250	14.5
Remainder bits	4	8
User Bandwidth (GHz)	4.64	9.28
# eLinks groups (16 bit)	7	7
eLinks bandwidth (MHz)	160/320/640	320/640/1280
#eLinks	28/14/7	28/14/7
EC bandwidth (MHz)	80	80/160
IC bandwidth (MHz)	80	80/160
Unassigned bits	0	4
Corrected (bits)	**5**	**2 x 5**
Efficiency	**91%**	**91%**

Example: 5.12 Gbps FEC5 Uplink Frame



7 groups of 4 input e-Ports
Number of data ports:

- 28 eLinks @ 160 Mbps
- 14 eLinks @ 320 Mbps
- 7 eLinks @ 640 Mbps

Frame	Function	I/O Group
FRMUP[9:0]	FEC[9:0]	
FRMUP[25:10]	Data[15:0]	0
FRMUP[41:26]	Data[31:16]	1
FRMUP[57:42]	Data[47:32]	2
FRMUP[73:58]	Data[63:48]	3
FRMUP[89:74]	Data[79:64]	4
FRMUP[105:90]	Data[95:80]	5
FRMUP[121:106]	Data[111:96]	6
FRMUP[123:122]	EC[1:0]	EC
FRMUP[125:124]	IC[1:0]	
FRMUP[127:126]	H[1:0] = 2'b10	HFH[1:0] = 2'b10

Note: This is how you will see the uplink frame after it has been processed by the lpGBT-FPGA receiver but not how it is actually transmitted by the lpGBT …

Uplink – FEC 12

- Dual data rate:
  - 5.12 Gb/s
  - 10.24 Gb/s
- Frame:
  - Header:
    - 2 / 4 – bits
    - 10 (high) & 01 (low)
  - IC field:
    - 2 – bit
    - 80 Mb/s
  - EC field:
    - 2 – bit
    - 80 Mb/s
  - Data field:
    - 96 / 192 – bit
    - 3.84 / 7.68 Gb/s
  - FEC field:
    - 24 / 48 – bit
- Error correction:
  - FEC:
    - Interleaving: 3 / 6
    - Symbol width: 4 – bit
    - Number of wrong symbols: 1 (× 3 / × 6)
  - Up to 12 / 24 consecutive bits
  - Efficiency: 78% (if accounting for the unassigned bits)
- eLinks:
  - Data
    - 24 eLinks @ 160 / 320 Mb/s
    - 12 eLinks @ 320 / 640 Mb/s
    - 6 eLinks @ 640 / 1280 Mb/s
  - EC
    - 1 eLink @ 80 Mb/s

FEC12

	5.12 Gb/s	10.24 Gb/s
Option	28	28
Frame (bits)	**128**	**2 x 128**
Header (bits)	**2**	**2+2**
Coded header	**no**	**no**
User field (bits)	102	204
Code (bits)	24	48
16-bit multiplicity	6.375	12.75
Remainder bits	6	12
User Bandwidth (GHz)	4.08	8.16
# eLinks groups (16 bit)	6	6
eLinks bandwidth (MHz)	160/320/640	320/640/1280
#eLinks	24/12/6	24/12/6
EC bandwidth (MHz)	80	80/160
IC bandwidth (MHz)	80	80/160
Unassigned bits	2	8
Corrected (bits)	**12**	**2 x 12**
Efficiency	**80%**	**80%**