Study of modern FPGA device and associated new technology, and search for possible application in High Energy experiments

Yun-Tsung Lai

KEK IPNS

ytlai@post.kek.jp

Seminar @ IJCLab

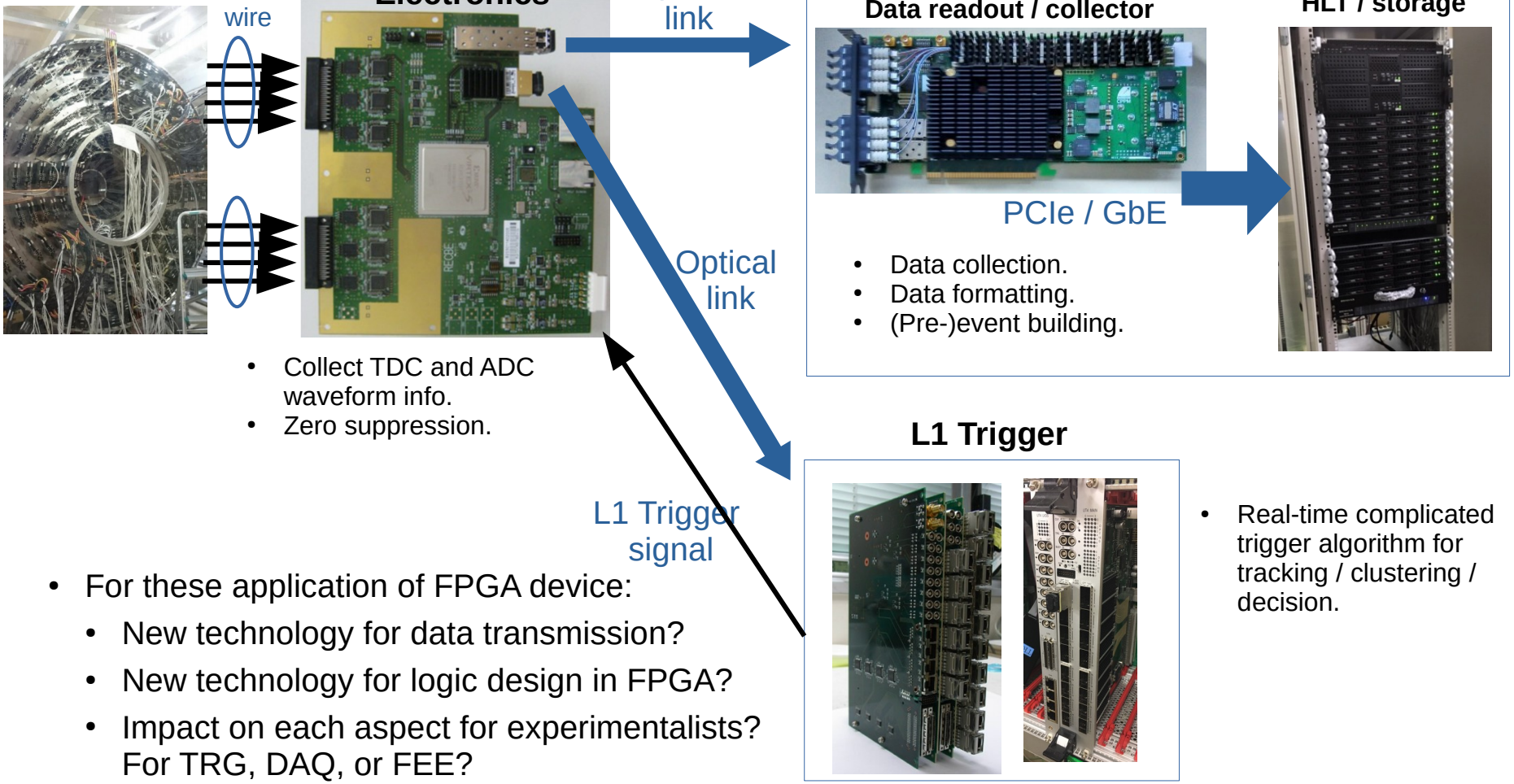
17th Jan., 2024

Outline

- Application of FPGA in HEP experiments
- DAQ system
- L1 Trigger system
- HLT system
- Versal project:
  - Introduction
  - Overview on our plan
  - Progress
- Summary & To do

Application of FPGA in HEP experiments

- Here we use Belle II Central Drift Chamber (CDC) as an example.



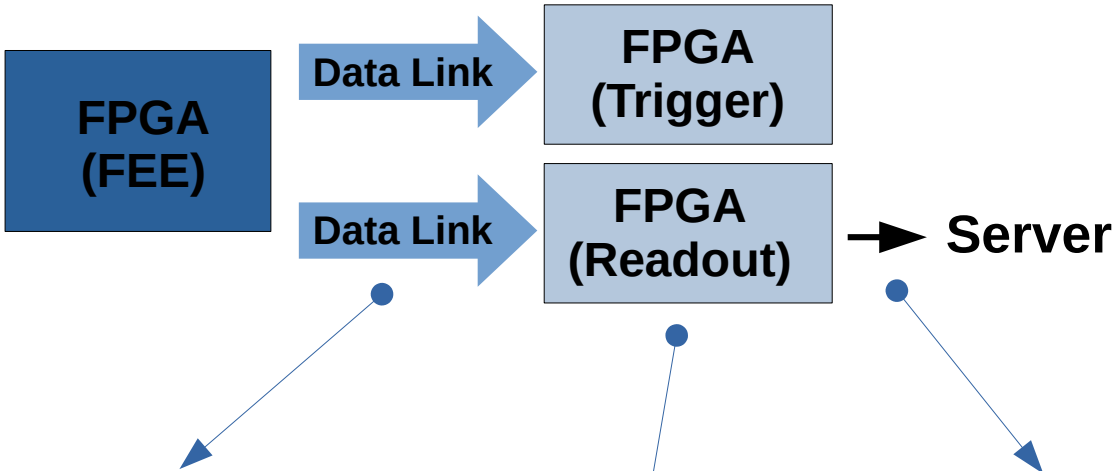
- Collect TDC and ADC waveform info.
- Zero suppression.

- Data collection.
- Data formatting.
- (Pre-)event building.

- Real-time complicated trigger algorithm for tracking / clustering / decision.

- For these application of FPGA device:
  - New technology for data transmission?
  - New technology for logic design in FPGA?
  - Impact on each aspect for experimentalists? For TRG, DAQ, or FEE?

Application of FPGA in HEP experiments (cont'd)



- **FPGA - FPGA transmission:**
  - Optical link with FPGA MGT and optical modules.
  - Non-Return-to-Zero (NRZ).
  - Different encoding based on protocol design purposes. e.g. 8B/10B and 64B/66B.
    - <10 Gbps for DAQ.
    - <25 Gbps for TRG.

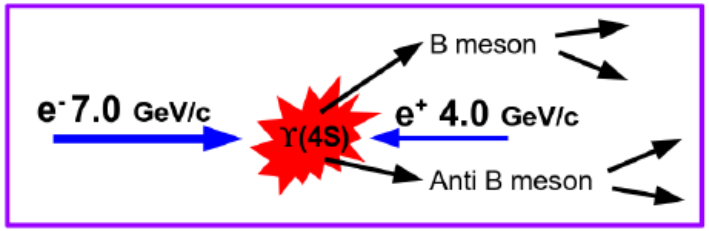
- Strong **FPGA devices** with:
  - Larger number of cells.
  - Larger data bandwidth.

  are critical for the usage in:
  - **TRG**: complicated algorithm implementation.
  - **DAQ**: collect and process large data.

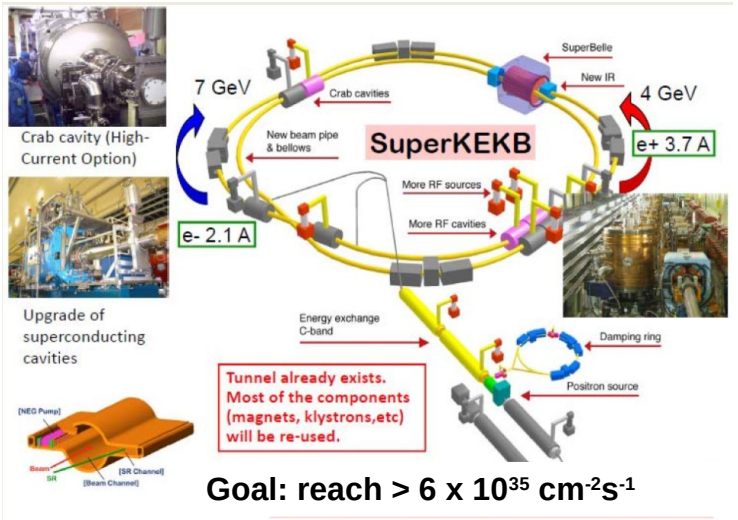
- **FPGA - server transmission:**
  - Data transmission and system slow control.
  - GbE, PCI-express, VME, etc.
  - PCI-Express is the most popular one nowadays: PCIe40 in ALICE, LHCb, and Belle II.

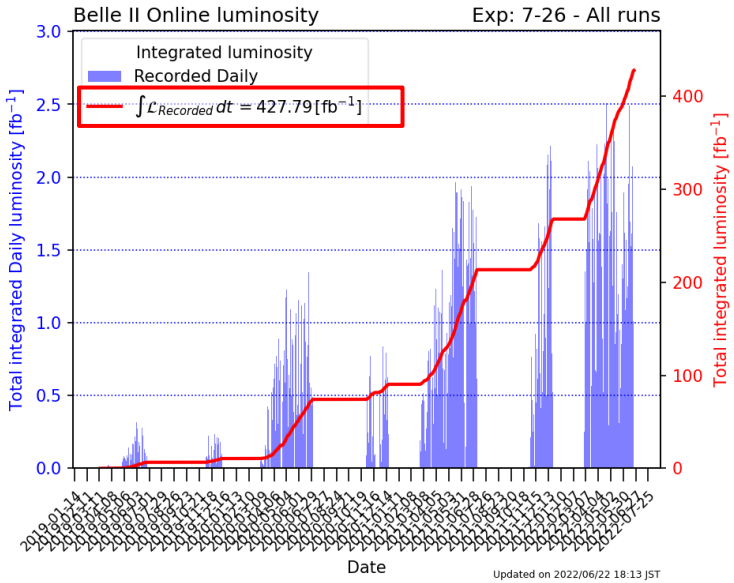
SuperKEKB

- SuperKEKB: Upgraded from KEKB.
  - More than 30 times larger luminosity of KEKB with nano beam scheme.
- Asymmetric energy collider:
  - 7.0 GeV e^- and 4.0 GeV e^+ for Y(4S) $\rightarrow B\bar{B}$.



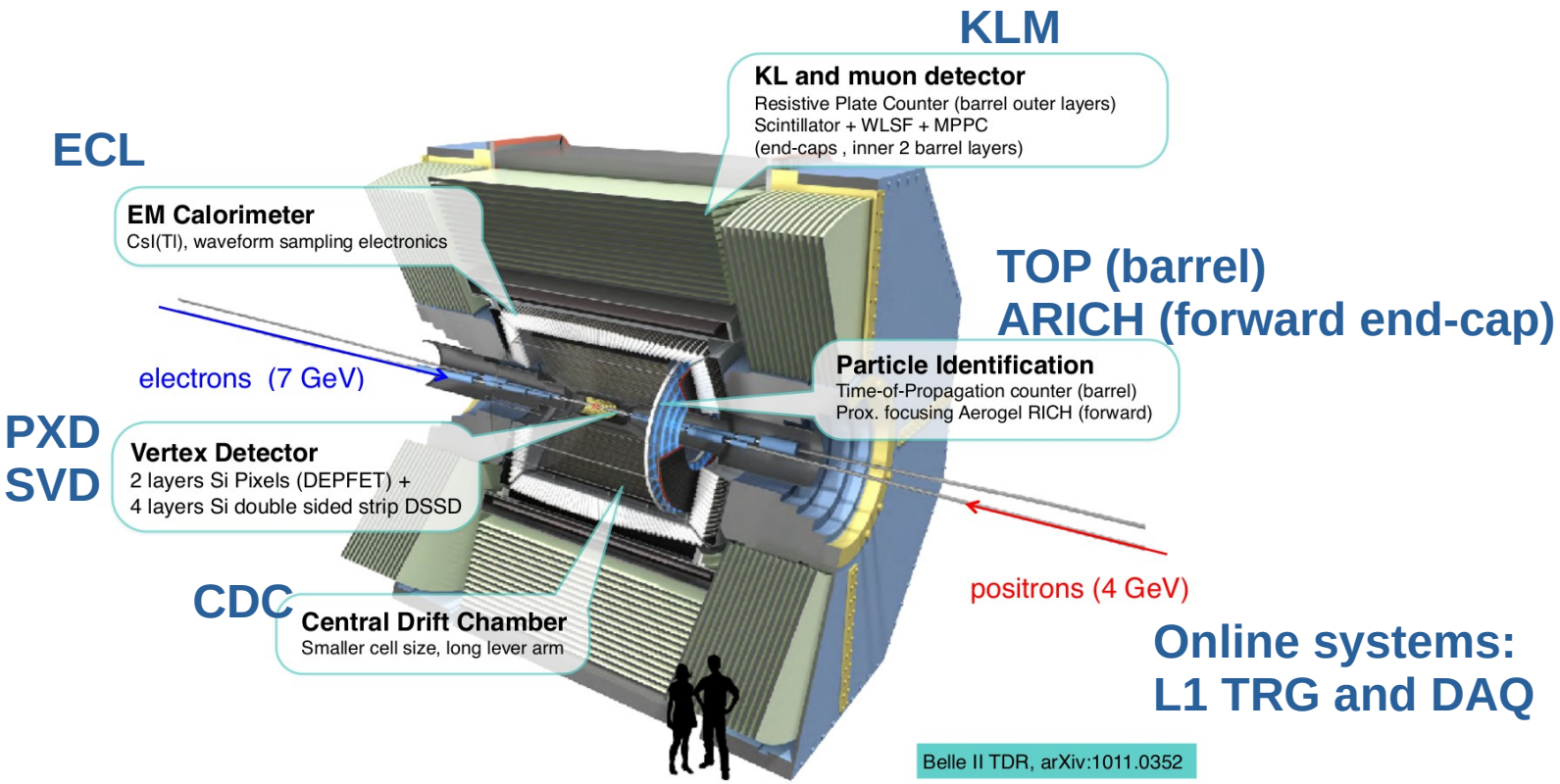
- Luminosity achievement:
  - L_{peak} = 4.65 x 10^{34} $cm^{-2}s^{-1}$.
    World record. ~Two times of KEKB record with much smaller beam current.
  - L_{int} = ~427 fb^{-1} up to Jun. 2022.
- Will resume beam collision in 2024 with PXD full installation.





Belle II detector

- Belle II: Newly-designed sub-detectors set to improve detection performance.

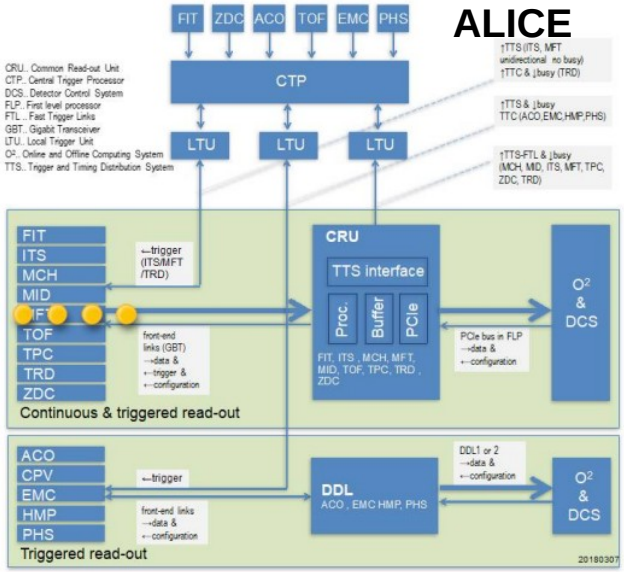


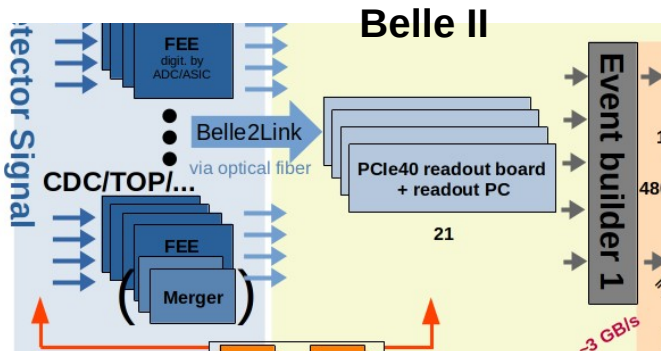
- Physics target of Belle II:
  - Rare B, τ, charm physics, Dark Matter search, CP Violation.

- Requirement for data taking:
  - High L1 trigger rate (~30 kHz), high background, and large event size.

DAQ system

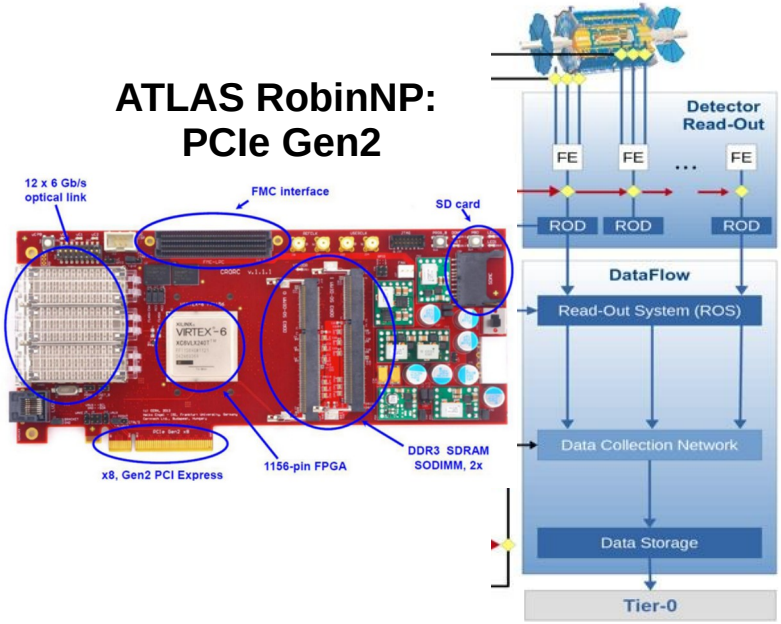
- Readout:
  PCIe has been the most popular solution for electronics → server interface.

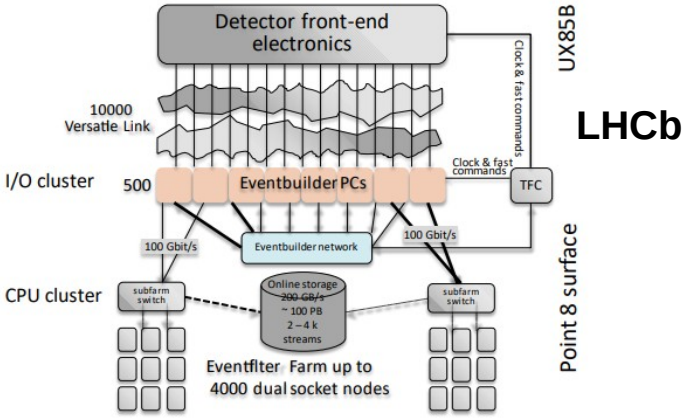




PCIe40: PCIe Gen3

ATLAS RobinNP: PCIe Gen2



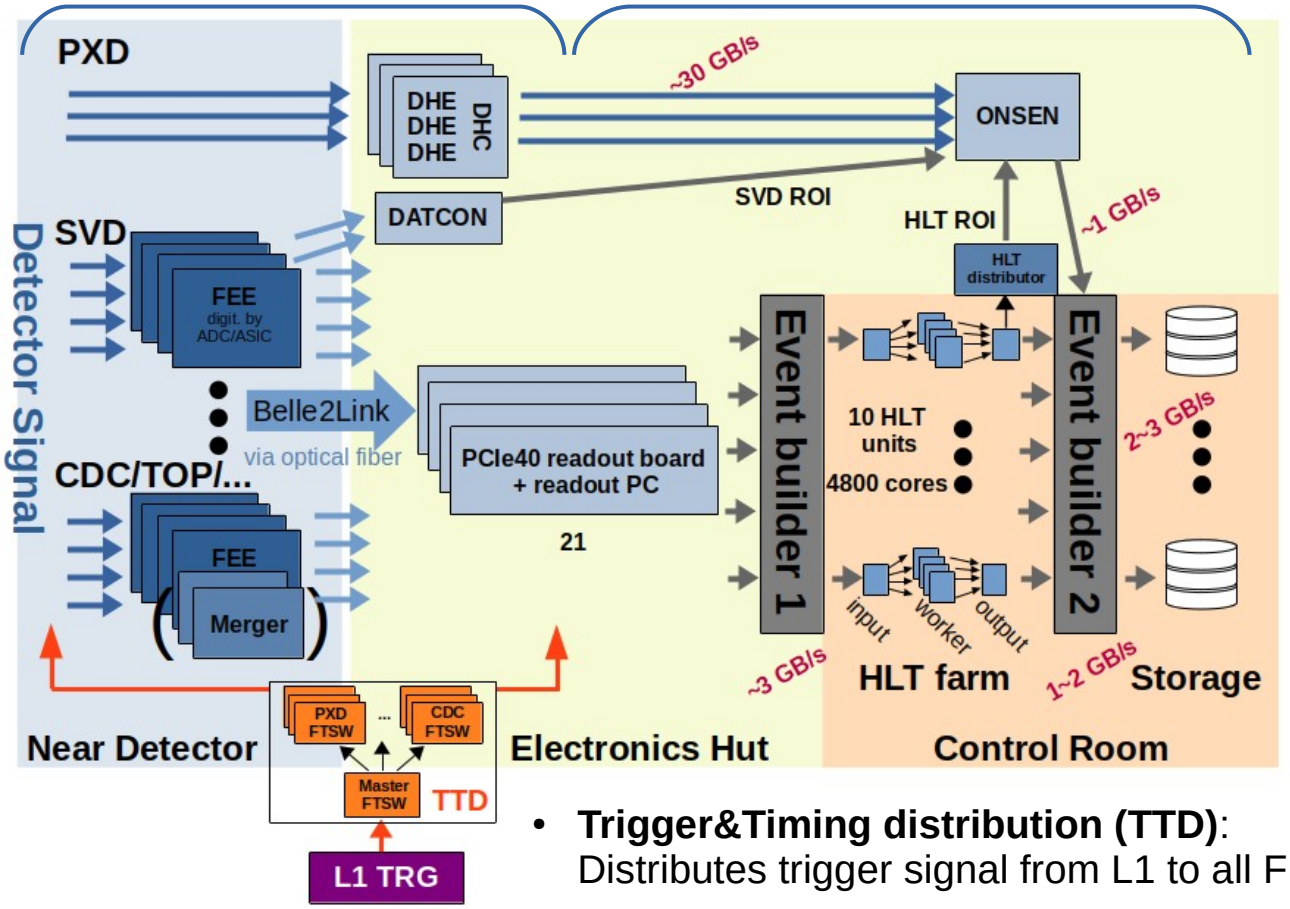


Belle II DAQ system

- Pipeline common readout system for each sub-detector.
  - Except for PXD: data reduction system with ROI.
- Target of performance: 30 kHz L1 rate, ~1% of dead time, and a raw event size of 1 MB.

- **FPGA (FEE - Readout)**: Use universal "Belle2Link" protocol with optical links in between.

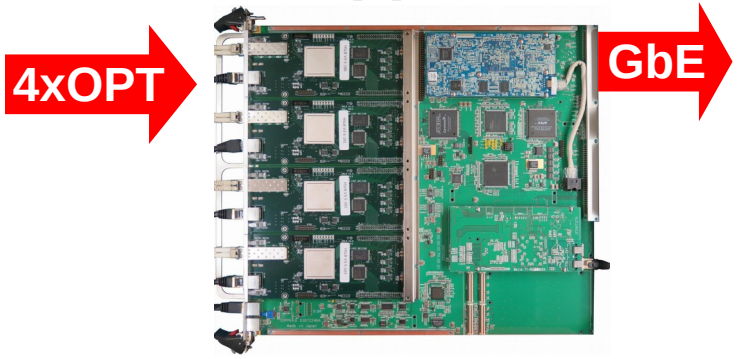
- **Back-end servers**: readout PC, HLT and storage for online procession/reconstruction.



- **Trigger&Timing distribution (TTD)**: Distributes trigger signal from L1 to all FPGA.

Readout device and its upgrade

Copper



- 4 Xilinx Virtex-5 receiver boards.
- PrPMC: data procession, pre event building.
- In total 203 coppers were used in Belle II.

Upgrade

PCIe40

- Intel Arria 10.
- Developed in LHCb and ALICE.
- 48 optical links.
- 2x8 PCIe Gen3.
- In total 21 PCIe40 boards will be used in Belle II.

Considerations for upgrade:

- Difficulty of maintenance:
  - Increasing number of malfunctioning pieces.
  - Many different boards in system.
  - Parts out of production already.
- Limit of the system on further improvement:
  - Output throughput by GbE: 1Gbps.
  - CPU usage: ~60% at 30 kHz trigger rate.

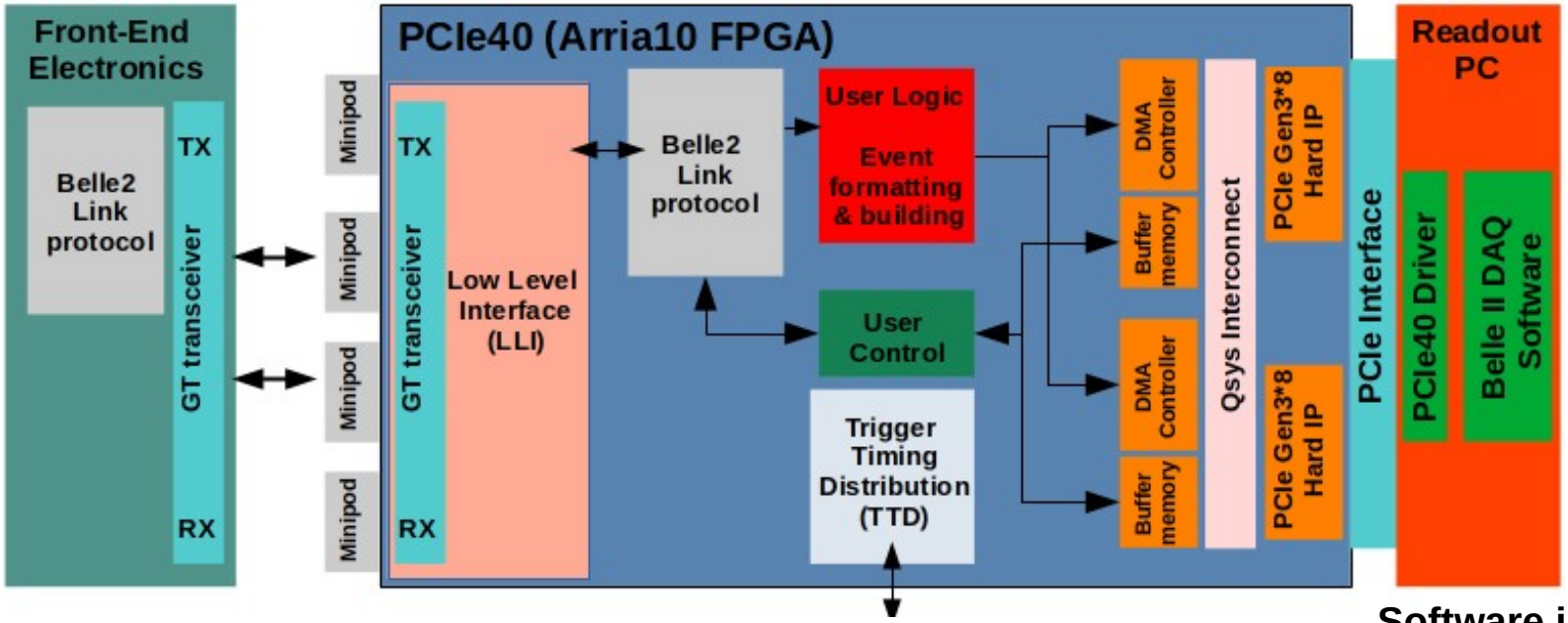
Overview on the upgrade with PCIe40

Belle2Link protocol:
8B/10B with transceiver & optical link.

Slow control: register rd/wr with B2Link
Detector configuration & System monitoring.

DMA for PCIe: Interface between PCIe40 and readout PC.

Data flow of detector FEE data. Processed inside PCIe40.



To TTD system via FTSW:
Trigger & timing distribution. Masking, run control.

Software in readout PC:
- Event building
- Slow control

Commissioning

Detector	FEE with Belle2Link	COPPER	COPPER readout PC	PCIe40 with readout PC
SVD	52	48	9	5
CDC	300	75	9	7
TOP	72	16	3	2
ARICH	72	18	6	2
ECL	53	26	10	3
KLM	32	8	3	1
TRG	24	12	3	1

Full replacement

Test run during short down time of accelerator

2022

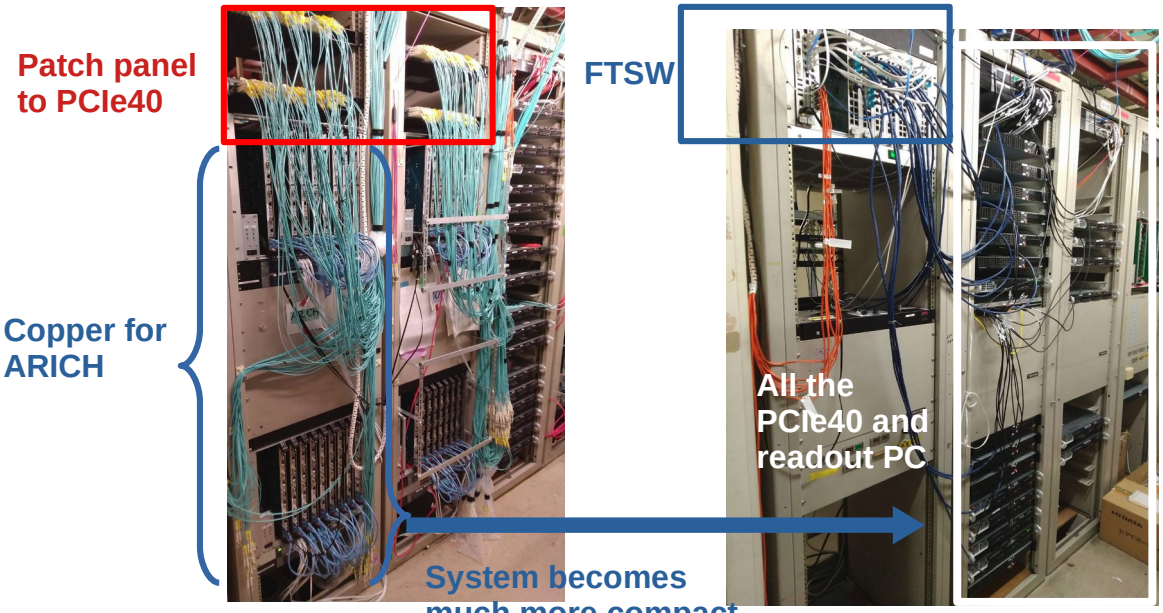
Oct. | Nov. | Dec. | Jan. | Feb. | Mar. | Apr. | May. | Jun. | Jul. | Aug. | Sep.

2021c run **2022ab run**

TOP, KLM (from 2021c)

ARICH (from 2022ab)

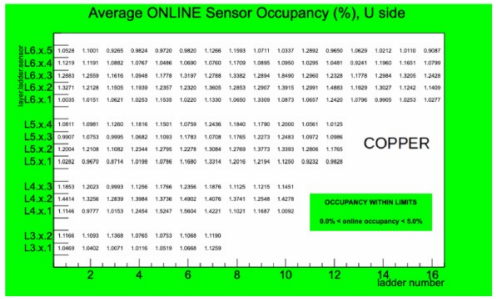
TRG, ECL, CDC, SVD

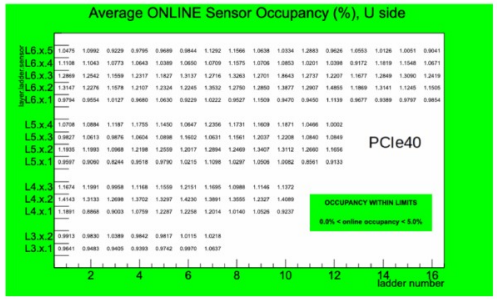


- TRG, ECL, CDC, SVD: replaced in 2022 summer
- Validation by:
  - Detector local calibration run.
  - 30 kHz dummy trigger high rate test.
  - Cosmic run with < 500 Hz trigger.
- Commissioning for the entire Belle II has been done!
- PXD: Not in the present PCIe40 upgrade project due to its system design.
  - Discussion is ongoing for PXD to utilize PCIe40 for data collection to rescue slow pion in far future.

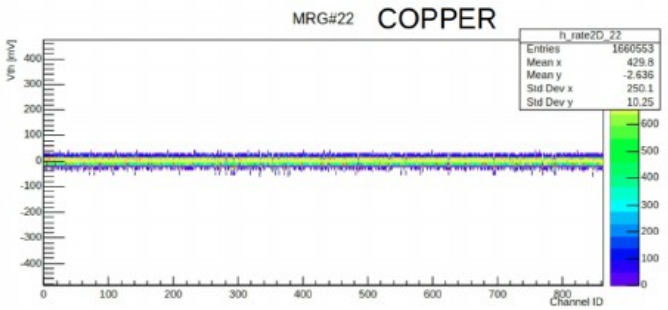
Commissioning (cont'd)

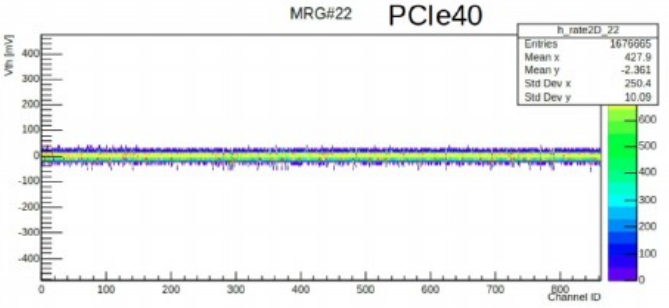
SVD: sensor occupancy



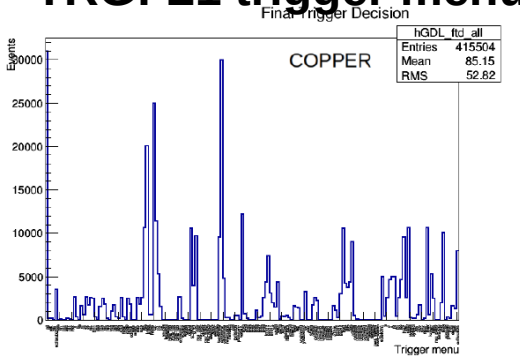


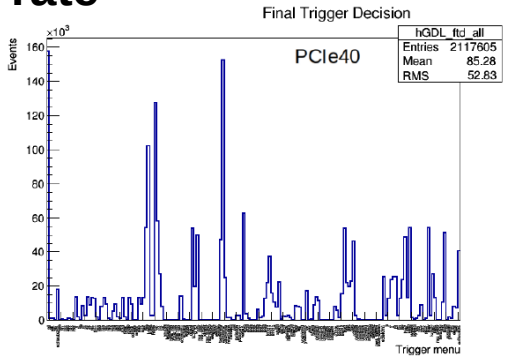
ARICH: Threshold scan



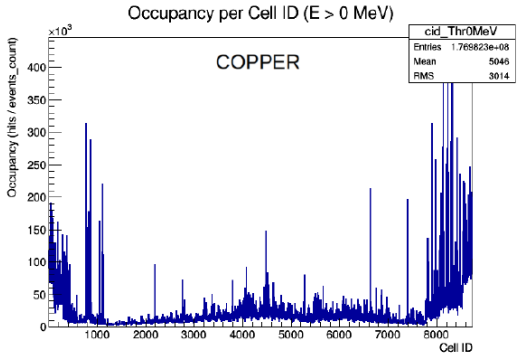


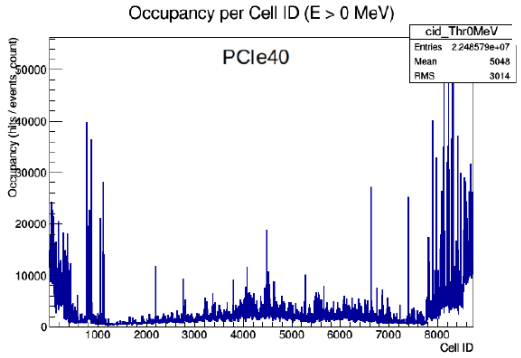
TRG: L1 trigger menu rate





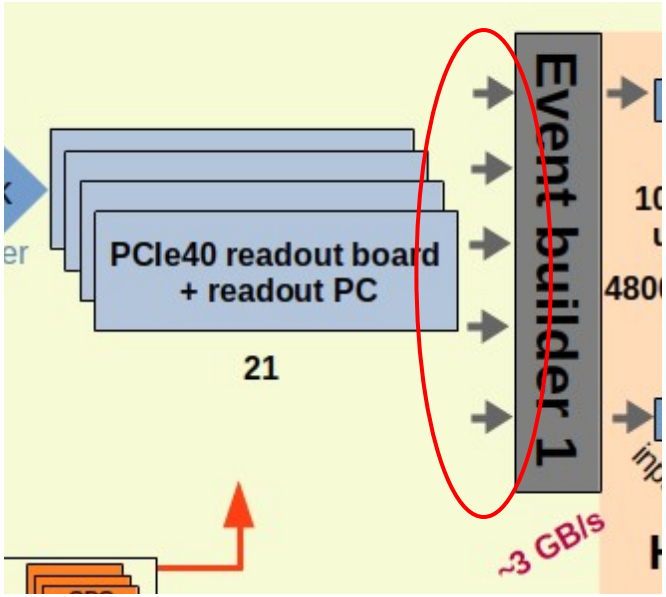
ECL: cell occupancy





Performance of Belle II DAQ in 2022

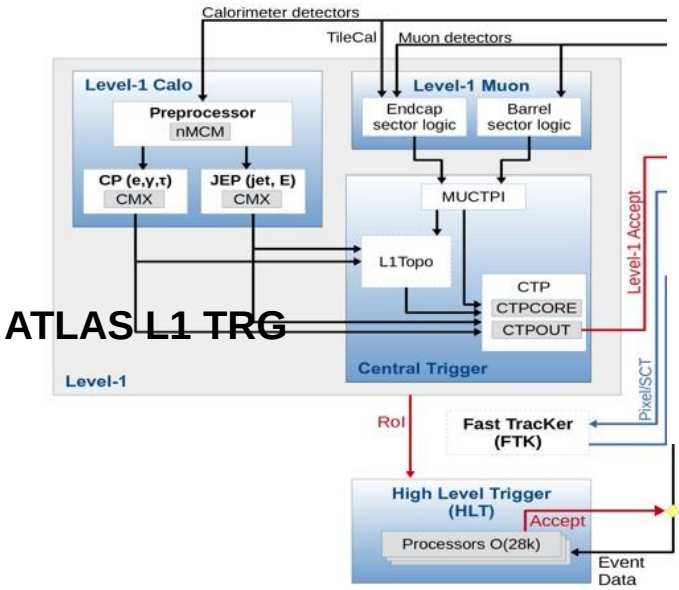
- PCIe40 upgrade:
  - TOP, KLM: from 2021c.
  - ARICH: from 2022ab.
- Overall running time fraction in 2022ab physics data taking: **92.6%**.
  - Restarting run: 3%.
  - System (detector or HV) problem: ~4%.
  - No major down time due to PCIe40.
- PCIe40 system:
  - PCIe40 → readout PC via PCI-express: ~3.9 GB/s.
  - Throughput in Belle II DAQ: 630 MB/s per readout PC.
  - Much improved from original Copper system.

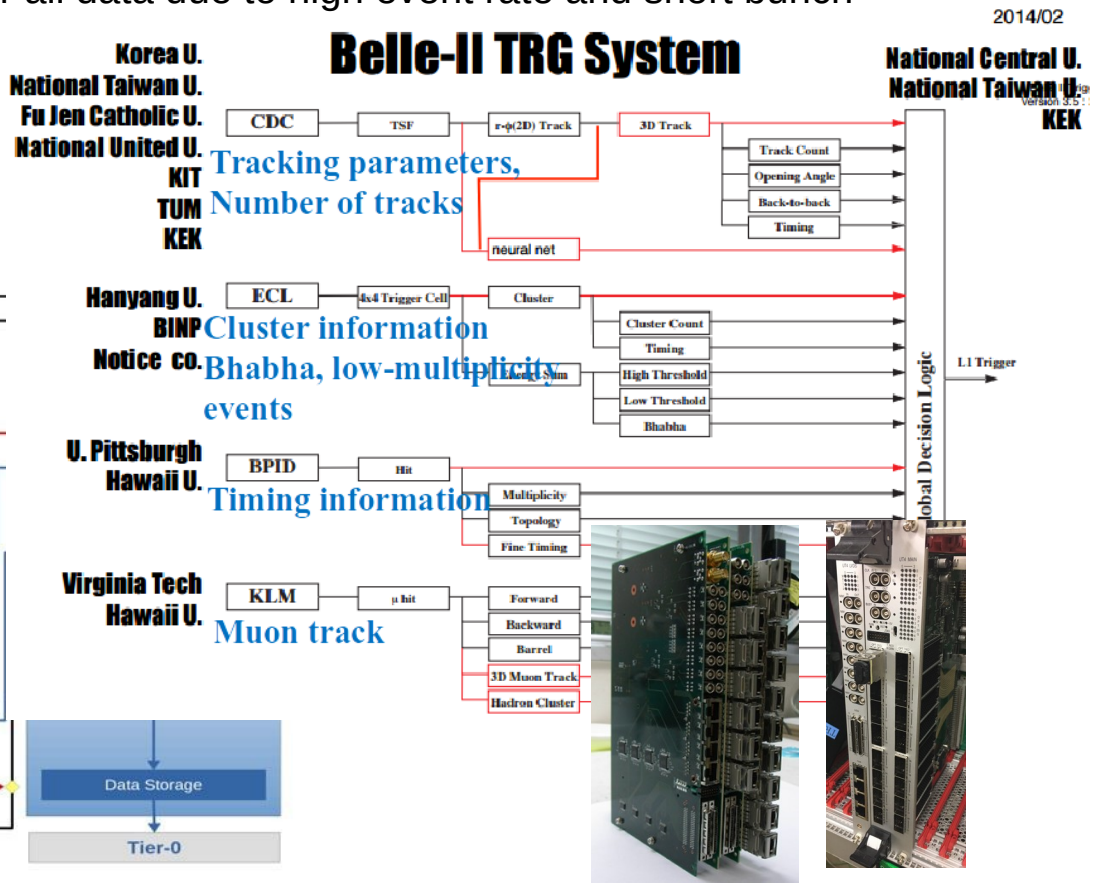


L1 Trigger system

- Provide L1 trigger signal to DAQ using FPGA chips for real-time processing on detector raw data.

- Why L1?
  - Buffer storage are not enough for all data due to high event rate and short bunch spacing in collider experiment.





Belle II TRG boards

Trigger device for Belle II and ATLAS

- For TRG purpose, complicated algorithm is implemented to process detector raw data in real-time. Utilization of machine-learning in the logic design became a trend recently.
- Strong FPGA with large resource: improve the logic itself, resolution of triggering, reduce the background rate, and perform everything within a latency limit.

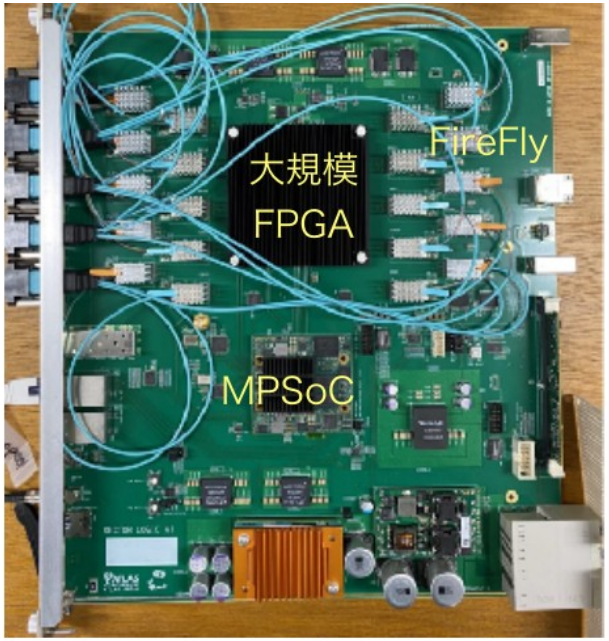
Belle II UT3

Xilinx Virtex-6
xc6vhx380t, xc6vhx565t
11.2 Gbps with 64B/66B

Belle II UT4

Xilinx UltraScale
XCVU080, XCVU160
25 Gbps with 64B/66B

ATLAS Muon Trigger processor



Xilinx UltraScale+
XCVU13P XCZU5EV
GTH,GTY: 16.8 Gbps
with 64B/66B

Belle II TRG

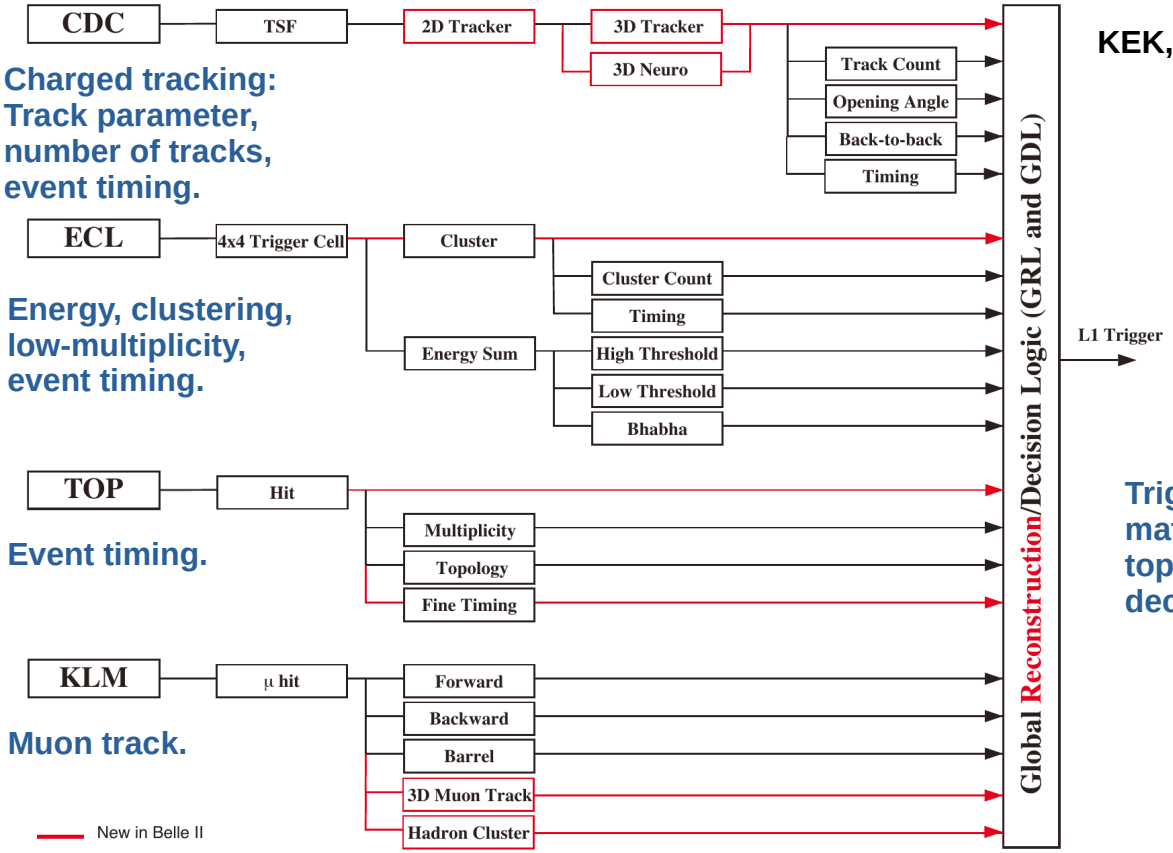
- 4 sub-trigger systems + 2 global trigger systems.

KEK, NTU, FJU, NUU, KIT, TUM, MPI, KU, KMI Nagoya, U. of Tokyo

Hanyang U., BINP, Notice co.

U. Pittsburgh, Hawaii U.

Virginia Tech., Hawaii U.



KEK, NTU, NCU

Trigger summary, matching, topological, final decision.

Conditions and requirements for TRG

- Requirements:
  - Overall latency < 4.4 μs.
  - ~100% eff. for hadronic events.
  - Max 30 kHz @ $8*10^{35}$ $cm^{-2}s^{-1}$
  - Timing precision: < 10 ns
  - Event separation: 500 ns

- Examples of technical challenges so far:
  - Low-multiplicity trigger mainly based on ECL, but contamination from noise, beam bkg or Bhabha.
  - Energy trigger with high eff. but high rate too.
  - Injection bkg.
  - Drawback of track trigger at endcap.
  - High track trigger rate due to crosstalk noise.
  - Latency budget due to transmission or complicated logics.
  - .....

- Physics processes in interest:

Phase2 Lum. Record

Process	C.S. (nb)	R@L=5.5x10^{33} (Hz)	R@L=8x10^{35} (Hz)	TRG logic
Upsilon(4S)	1.2	6.6	960	CDC 3trk(fff) ECL high energy(hie) ECL 4 clusters(c4)
Continuum	2.8	15.4	2200	
μμ	0.8	4.4	640	CDC 2trk(ffo) etc
ττ	0.8	4.4	640	
Bhabha	44	242	350 *	ECL Bhabha(bhabha, 3D bhabha)
γ-γ	2.4	13.2	19 *	
Two photon	13	71.5	10000	CDC 2trk(ffo) etc
Total	67	357.5	~15000	

Data transmission protocol

- Data transmission in TRG: Xilinx and Altera FPGA MGT, QSFP module, and MPO cable.
- The original plan was to use the open-source Aurora protocol, but large latency was introduced and exceeded the L1 limit (4.4 μs).
- Belle II CDCTRG developed an user-defined transmission protocols:
  - Smaller latency than Aurora's: **Latency reduction is critical for L1!**
  - User-friendly interface.
  - 8B/10B and 64B/66B encoding.
  - Support various Xilinx and Altera MGT.
  - Bit error rate < 10^{-18} /s with few weeks BERT.
  - Flow control and synchronization.

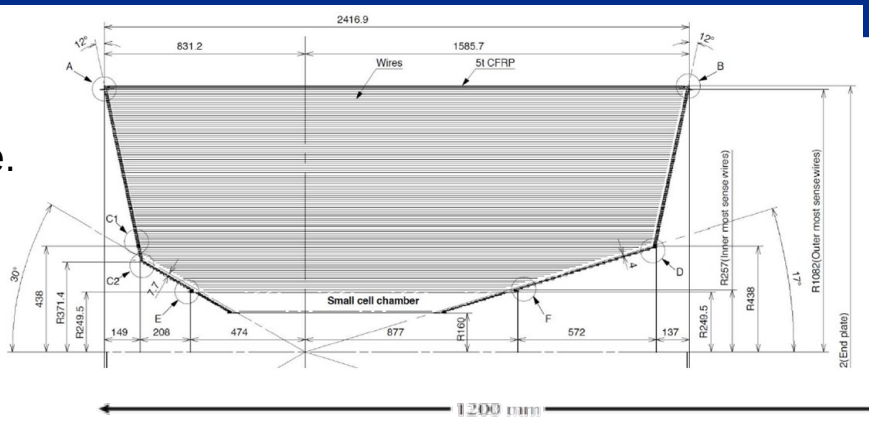
Latency comparison using UT3 (Virtex-6 GTX and GTH)

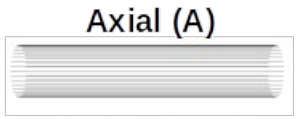
Protocol	Lane rate	user_clk	Link type	Latency (ns)
Aurora 8B/10B	5.08 Gbps	254 MHz	GTX-GTX	185∼190
Raw-level 8B/10B	5.08 Gbps	254 MHz	GTX-GTX	132∼136
	5.08 Gbps	254 MHz	GTH-GTX	132∼136
	5.08 Gbps	254 MHz	GTH-GTH	91∼95
	5.08 Gbps	254 MHz	GTX-GTH	91∼95
Aurora 64B/66B	10.16 Gbps	158.75 MHz	GTH-GTH	296∼302
Raw-level 64B/66B	11.176 Gbps	169.33 MHz	GTH-GTH	106∼112

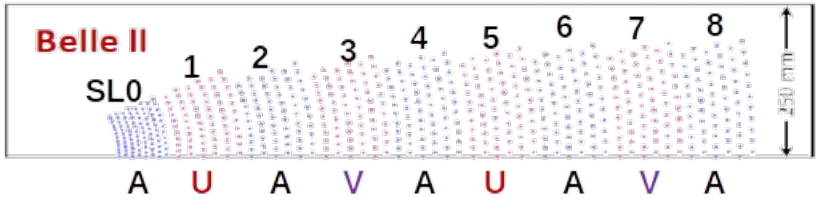
- For **UT4**:
  - Up to 25 Gbps using 64B/66B.
  - Latency: ~ 50ns.

Track trigger with CDC

- ~14K sense wires with mixture of He and ethane.
- An alternative **AUAVAUAVA** wire configuration for 3D information:
  A: Axial super-layer (SL) parallel to z-axis
  U, V: Stereo SL with two small stereo angles.

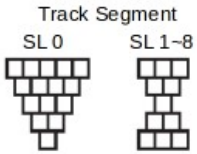


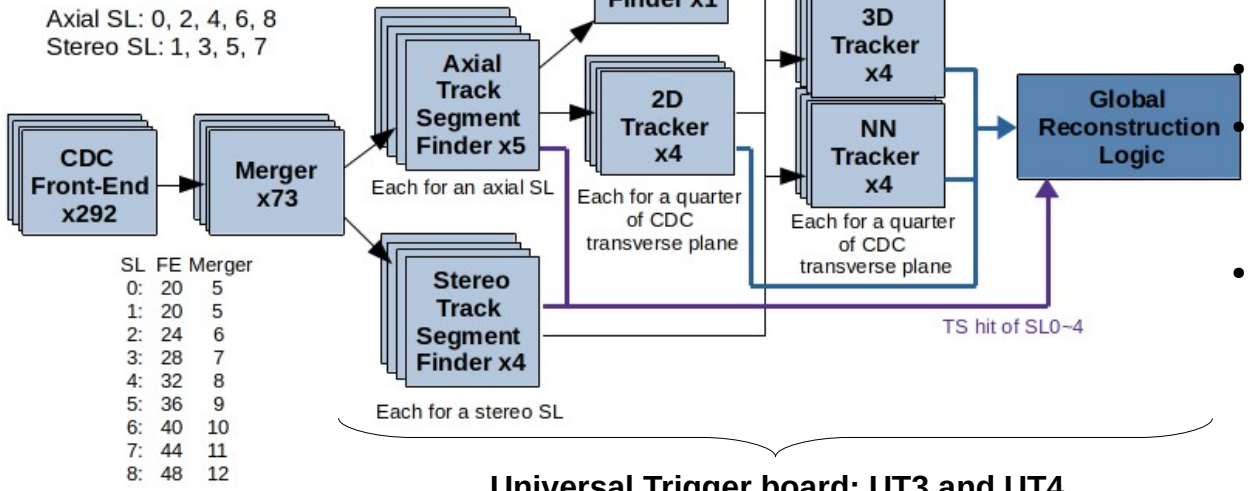




Fonrt-End Xilinx Virtex-5

Merger Altera Arria2

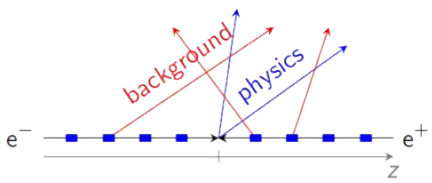




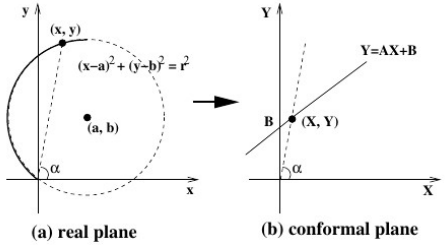
Universal Trigger board: UT3 and UT4

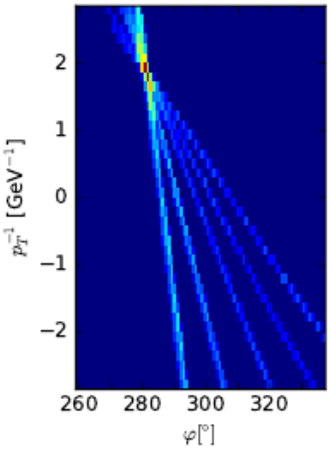
- Track Segment Finder (TSF).
- Tracker: 2D full track, 3D, Neural 3D (NN), and short tracker.
- Event Timing Finder (ETF)

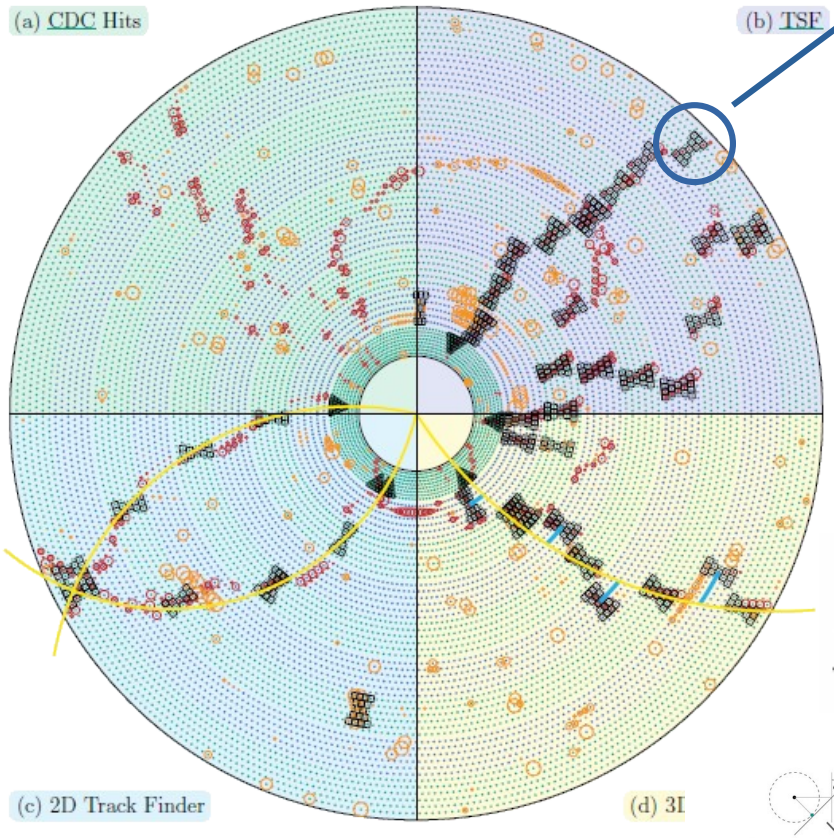
Track trigger with Belle II CDC: algorithms

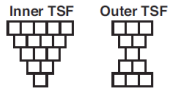


2D Track Finding:
Hough transformation.
Limited track condition.

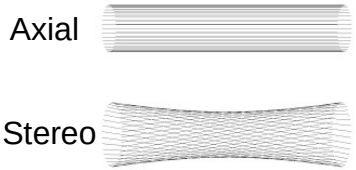




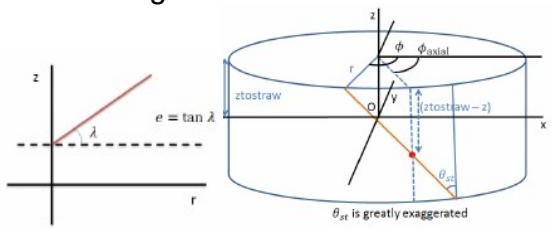




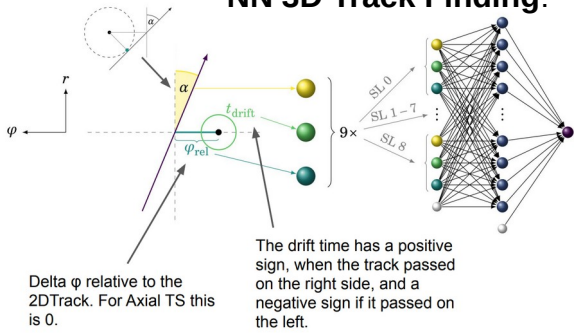
Track Segment:
Simplification on the algorithm



3D Track Finding:
Fitting with Stereo wire info

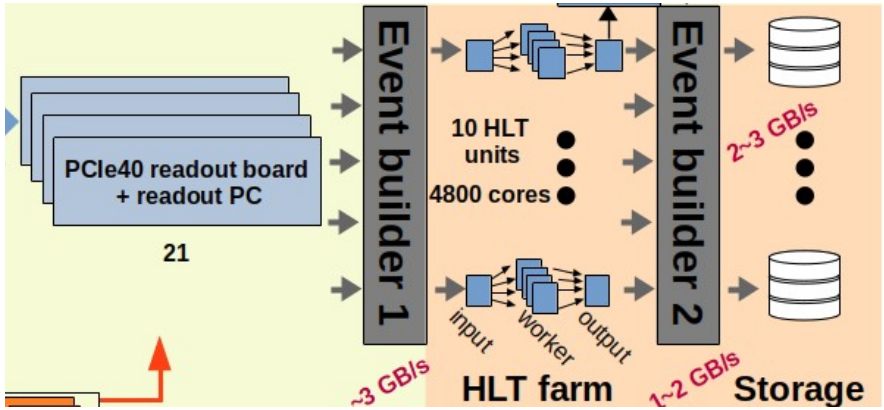


NN 3D Track Finding:



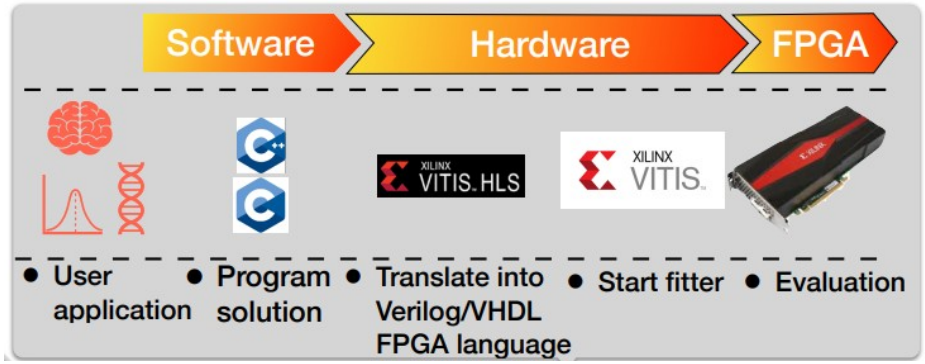
Belle II HLT

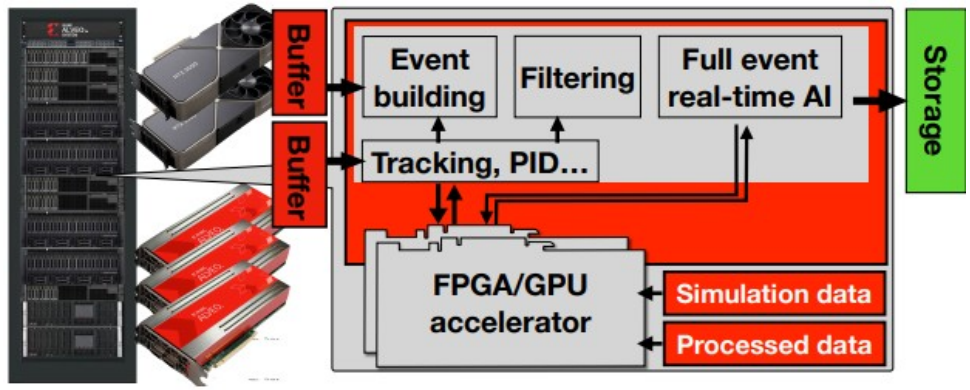
- HLT: Computing servers with reconstruction software.
  - In Belle II: HLT software = offline software.
- How about the options other than CPU?
  - GPU? FPGA with HLS?



source: Qi-Dong Zhou, Shandong Univ.

System	Processing power / HLT unit	Price (¥) / HLT unit	Ratio
CPU (Intel Xeon E5 2660)	480 cores	18,000,000	~6.5
GPU (GeForce RTX 3090)	12 GPU GPU : CPU ~ 40 : 1	GPU: ~180,000 x 12 = 2,160,000 Server: 600,000 x 3 = 1,800,000 Total : 3,960,000	~1.5
FPGA (VCK5000, Versal ACAP VC1902)	5 FPGA card Versal : CPU ~ 100 : 1	FPGA card : ~300,000 x 5 = 1,500,000 Server: 600,000 x 2 = 1,200,000 Total : 2,700,000	1

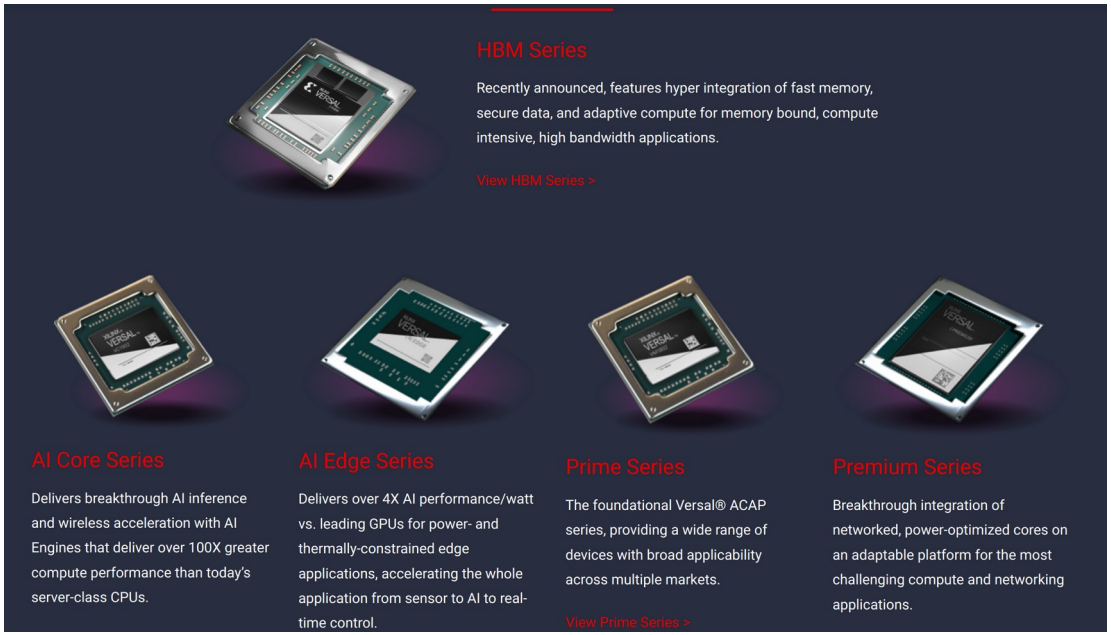


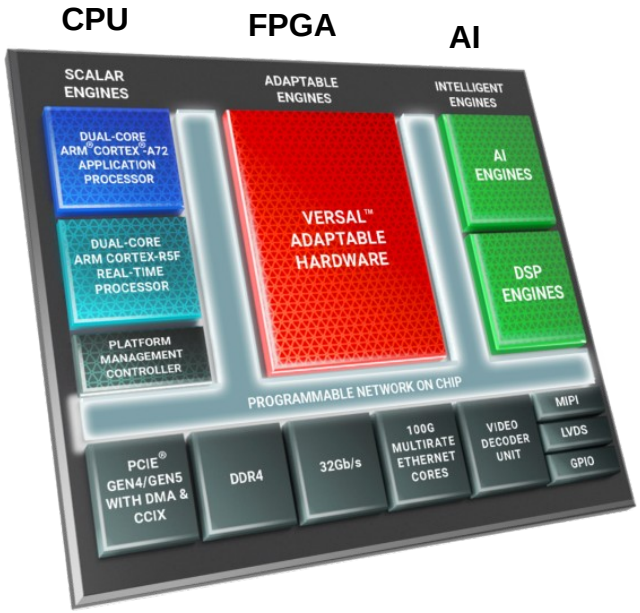


Versal project

- KEK together with Japanese HEP community purchased a few evaluation kits of the Xilinx Versal series ACAP.
  - Plan: Common and general studies on the new technologies for future electronics device's R&D. Now we plan to use Versal for L1 TRG, DAQ or HLT purpose.

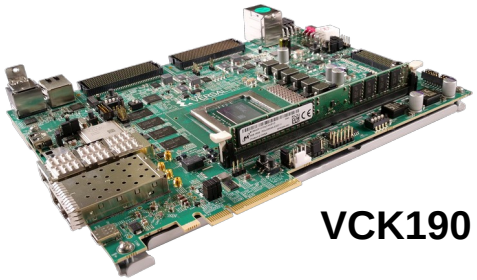
- The features of different Versal series ACAP:
  - AI engine: convenient interface to implement ML core into firmware.
  - High Bandwidth Memory (HBM).
  - Larger number of cells + High transmission bandwidth.





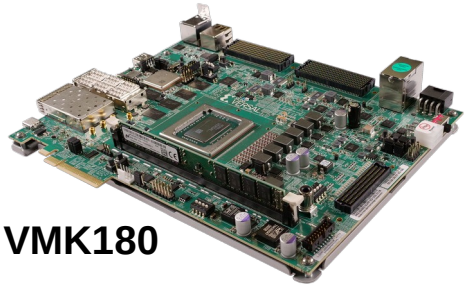
source: Xilinx website

Evaluation kits for Versal



VCK190

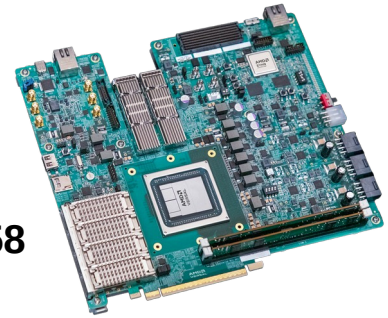
- Features the VC1902 Versal AI Core series
- For using AI and DSP engines with greater compute performance that current server class CPUs



VMK180

- Features the VM1802 Versal™ Prime series
- The world's first ACAP
- A software programmable infrastructure and connectivity

VHK158



- Features the VH1582 Versal™ HBM series
- convergence of memory, compute, and connectivity with 32G HBM and 112G PAM4

VCK5000

- Features the Versal AI Core Series
- For (AI) Engine development with Vitis and AI Inference development
- Not flexible for FPGA firmware

VPK120

- Features Versal™ Premium series VP1202
- Multiple high-speed connectivity option
- Massive serial bandwidth, security, and compute density

VEK280

- Features the VE2802 Versal AI Edge series
- Simpler version of VCK190
- Will come out in 2024

Versal project: General plan and roadmap

- Our goal: R&D of a new general FPGA device using the Versal ACAP.
  - A L1 TRG, DAQ, or HLT device, and also general for different experiments.
  - One clear target is **UT5 for L1 TRG of both Belle II and ATLAS**.

1st year:

- Study the properties of the fundamental functionalities with the kits:
  - GTM (PAM4), PCIe Gen5, AI/DSP engine, CPU acceleration, etc.
- Prepare basic application for each of them for other members.

Here we are now with VPK120. VCK190 will arrive soon.

2nd year:

- Make general transmission protocols for GTM (PAM4), PCIe Gen5, and do performance study.
- Implement various Trigger algorithms (Belle II, ATLAS, etc).
- Connect to existing systems to take real-time data and check performance.

3rd year:

- Future universal device: L1 TRG, DAQ readout, or HLT.
  - Discussion.
  - Schematic/PCB design for the prototype boards.
  - Test with experiments people.

IJCLab: Hardware technical support, application at LHCb

TYL/FJPPL

KEK E-sys: technical and firmware development support, resource and plan management, hardware R&D for future device.

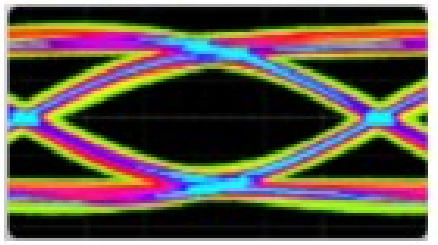
Belle II, ATLAS, ALICE, etc. groups in Japan: Dedicated algorithm and other application.

"Collider Electronics Forum" for common R&D in Japanese HEP community.

New technology in Versal FPGA: PAM4

- Most of the present HEP devices are based on **Non-Return-to-Zero** (**NRZ):**
  - Limit of line rate is 25~30 Gbps.
  - Belle II UT4 (UltraScale GTY) can operate with 25 Gbps stably using 64B/66B.
  - ATLAS muon trigger board reaches ~16 Gbps.

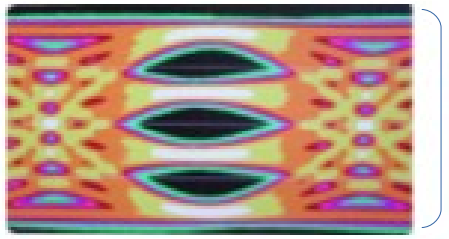
NRZ
(Non-Return-to-Zero)



Limit: ~25 Gbps

- **Pulse Amplitude Modulation (PAM4):**
  - Four distinct voltage levels.
  - Xilinx UltraScale+: up to 58 Gbps.
    Versal premium: up to 112 Gbps.
    - We can study it with VPK120.
  - Require 4*100 Gbps QSFP.
  - Pioneer to study it in HEP community.
  - In addition, PCIe Gen6 also utilizes PAM4.

PAM4
(Pulse Amplitude Modulation)



source: Xilinx

Four distinct voltage levels.
Two bits per clock cycle.

New technology in Versal FPGA: PCIe 5.0

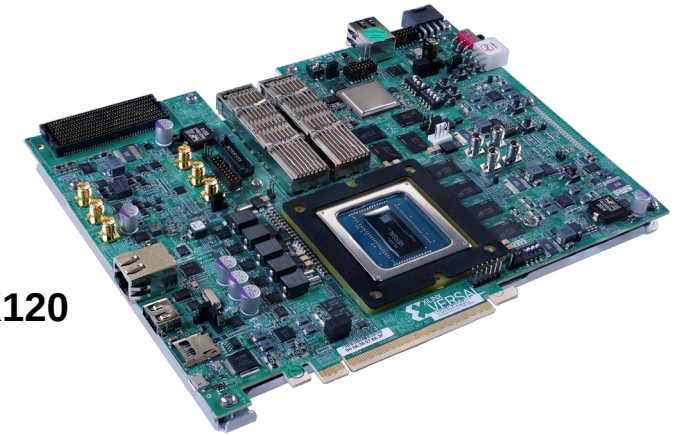
- PCI-Express has been the most popular option in HEP:
  - ALICE, LHCb and Belle II has been using the PCIe40 board, which is based on PCIe Gen3.
  - Much improved throughput than the previous COPPER readout (based on GbE) in Belle II.

- New generation of PCI-Express comes out with a doubled bandwidth in few years.
  - Using proper device to study the properties of newer generation of PCI-Express is beneficial for the future readout device's development.

- For these new features of data transmission (PAM4 and PCIe 5.0), VPK120 (featured with Versal™ Premium series VP1202) is a good candidate to perform associated studies.
  - GTM transceiver: supports both NRZ and PAM4 up to ~100 Gbps.
  - PCIe 5.0 x 16 lanes with GTYP transceiver.

- A test bench has been setup at KEK E-sys group and some works have been performed.

VPK120

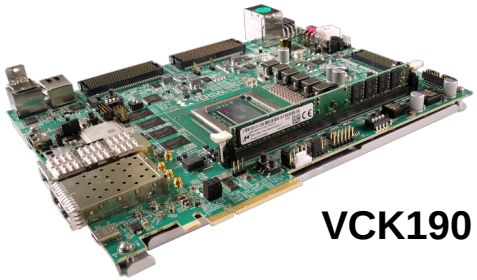


New technology in Versal FPGA: AI engine

- Implementation of physics algorithms (L1 or HLT) in FPGA is getting complicated.
  - ML implementation based on High-Level-Synthesis (HLS) becomes a trend recently.

- AI engine: A new technology for data processing.
- Better performance of computing density and power reduction compared to the transitional approach.
- C programmable. Suitable for both non-ML and ML software implementation in FPGA.

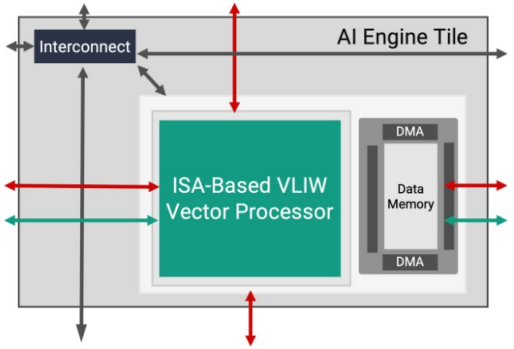
- Features with AI engine, VCK190 has more flexibility on the firmware design (while VCK5000 is a kind of acceleration card).
  - VCK190 will arrive KEK soon.

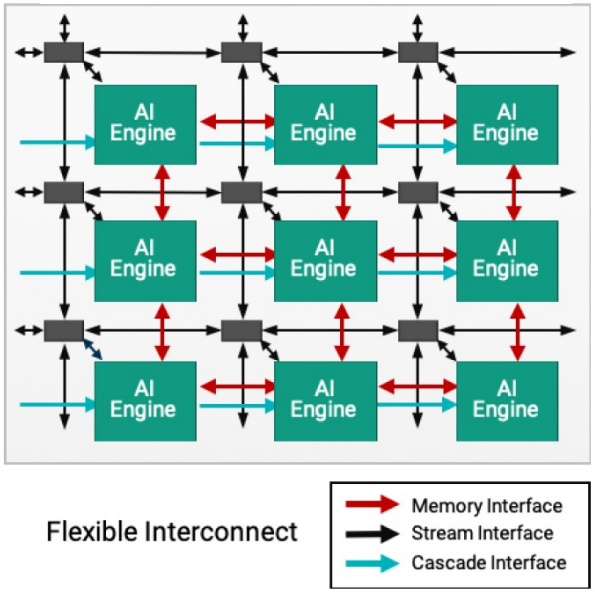


VCK190

VCK5000

XILINX VITIS™ AI





https://www.xilinx.com/products/technology/ai-engine.html

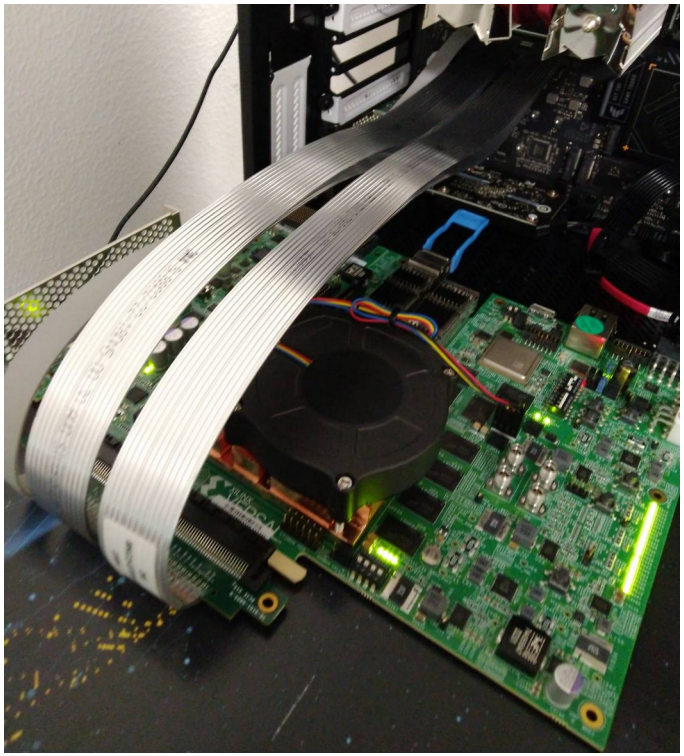
Status of the VPK120 test bench @ KEK E-sys group

- The test bench of VPK120 has been built at E-sys group and released to our members for dedicated studies.

- Progress so far:
  - Firmware making.
  - Processing system (PS) with CIPS and NOC.
  - GTM transceiver: NRZ and PAM4.
  - PCIe.
  - hls4ml.

- Special thanks to Mathis Maurice, internship in E-sys group in this summer, for helping this preparation work!

PC side: PCIe Gen5 x16 slot

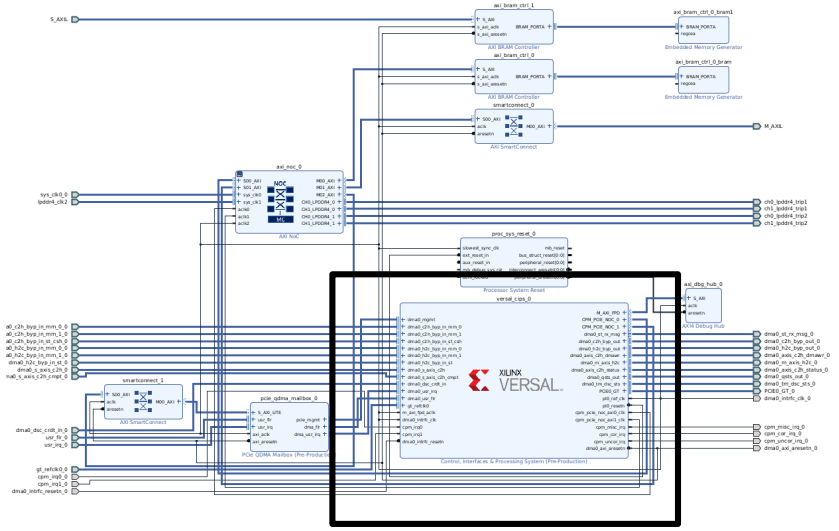


Firmware making with Versal: PS, CIPS and NOC

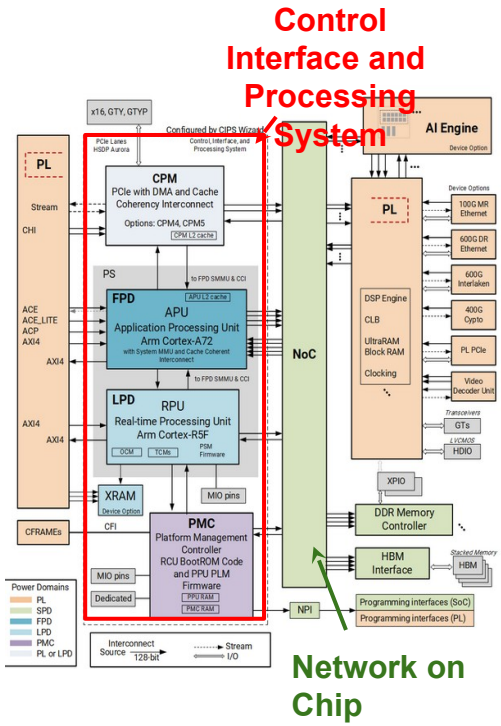
- In our experience, FPGA firmware making is:
  - Writing HDL codes and using IPcore to control all the **Programmable Logic (PL)**.

- But Versal is an ACAP containing lots of sub-systems together with the FPGA.
  - Not only PL, but also **Processor System (PS)**.
  - Firmware making tends to rely on the automatic block design rather than the traditional code-writing way.
  - For now, we still have limited understanding in PS.

A firmware design with PCIe



CIPS (Common Interface Processing System) : Interface referring to the integrated processing sub-system

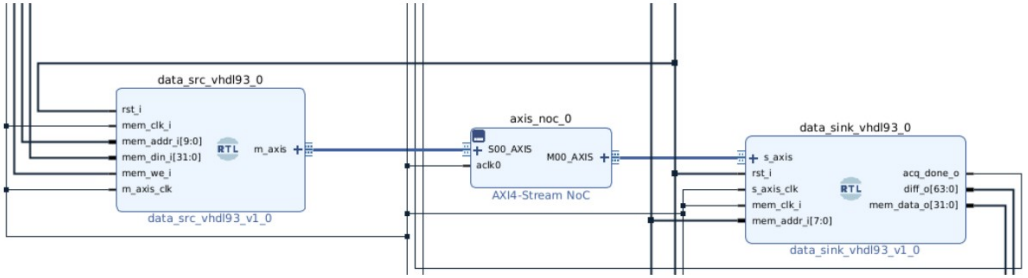


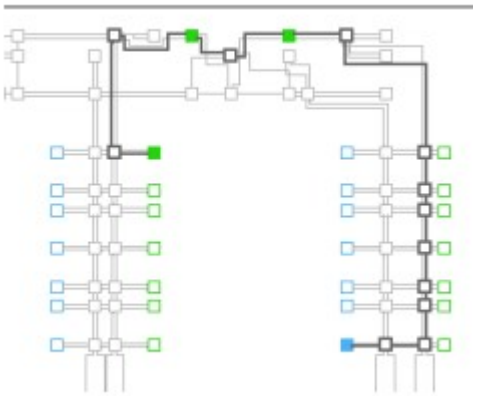
In the future, if we use AI engine for logic design, latency from NOC is important.

NOC (Network On Chip): Communication network for sub-systems of the FPGA.

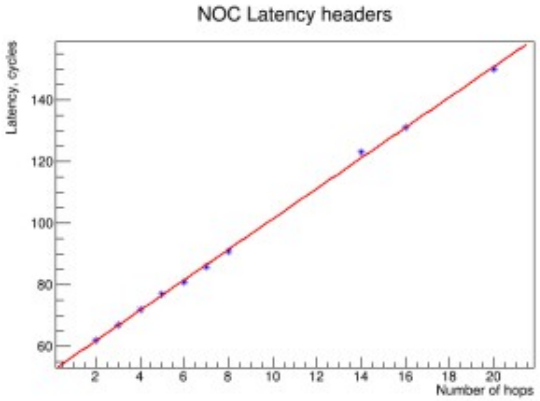
Latency measurement with NOC

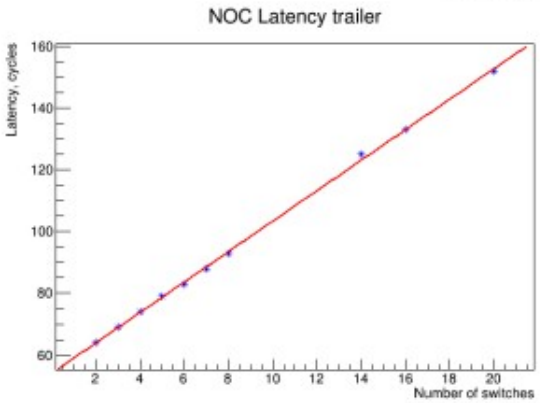
- Study by Dmytro Levit (KEK).





A case of 20 switches





- Latency headers:

$$L_{headers} = 52 + 5 * n_{switches}$$

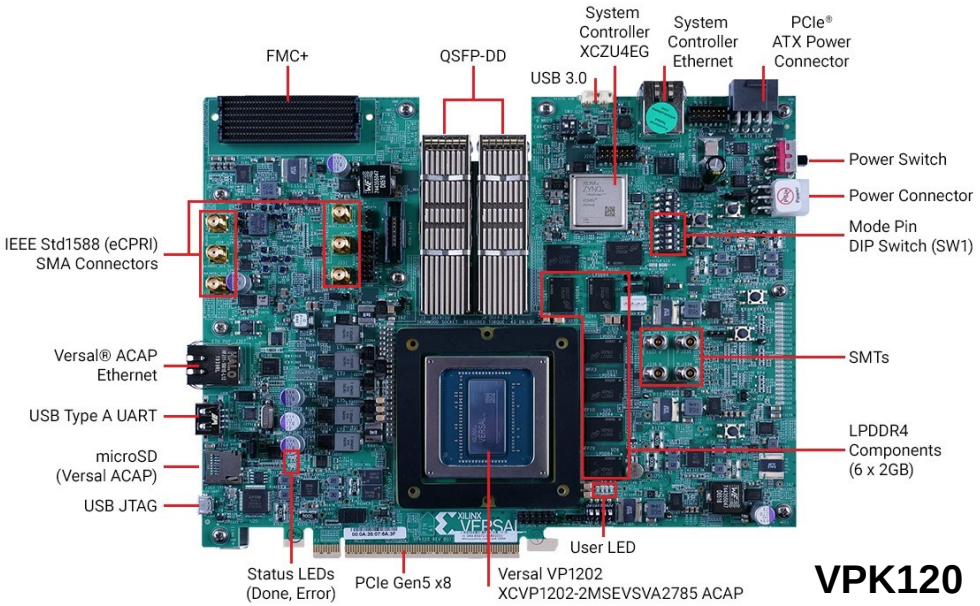
- Latency trailer:

$$L_{trailer} = 54 + 5 * n_{switches}$$

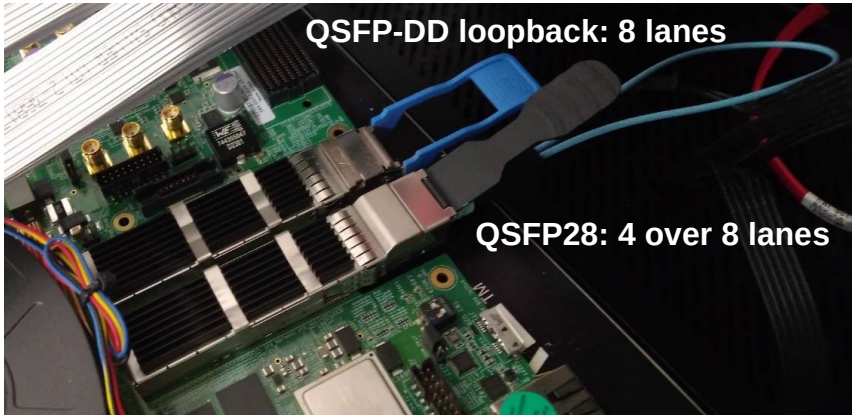
- Large discrepancy with latency estimated by vivado
  - vivado estimates 14-50 cycles latency
- Further measurements possible with multiples senders/receivers

Versal transceivers: GTYP and GTM

- GTYP: PCIe 5.0 (16) and FMC+ (8)
  - 1.25 ~ 32.75 Gb/s.
  - Various encoder supported.
- GTM: QSFP-DD (8*2)
  - NRZ:
    - 9.5 ~ 15, 19 ~ 29 Gb/s.
  - PAM4:
    - 19 ~ 30, 38 ~ 60 Gb/s
    - 76 ~ 112 Gb/s: "Half density mode" by combining two lanes.
    - No encoding is supported.
      Need to be make them manually in RTL.
- Our test setup for transceiver study:



VPK120



IBERT interface and usage: PAM4 56 Gbps

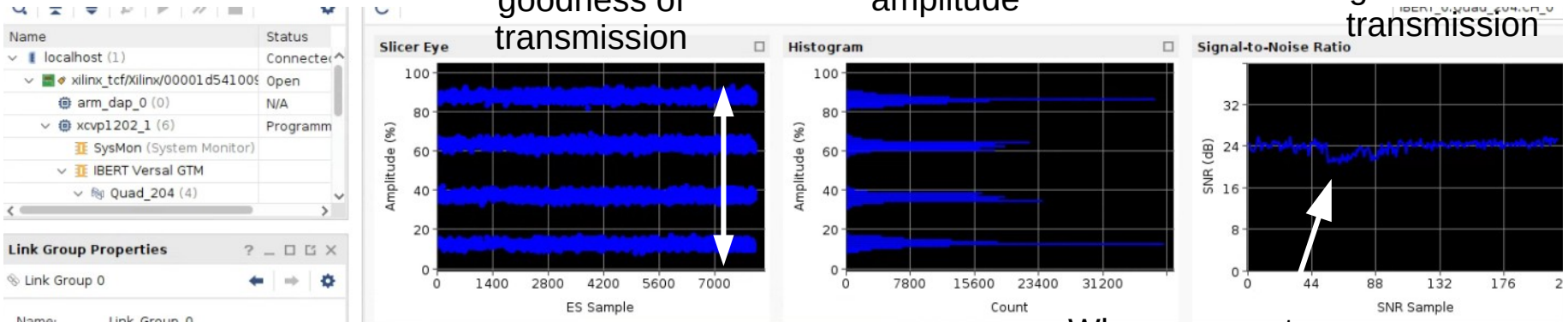
- PAM4, 56 Gbps per lane.
  - Parameter tuning on cursor position and termination voltage, etc, is necessary to have stable transmission (0 bit error).

DesignCon 2019 Enabling IBIS-AMI Simulations for Systems Containing PAM4 Retimers at 112Gbps

amplitude/height of eye ≠ goodness of transmission

Histogram of the amplitude

SNR ≠ goodness of transmission



When parameters were under tuning.

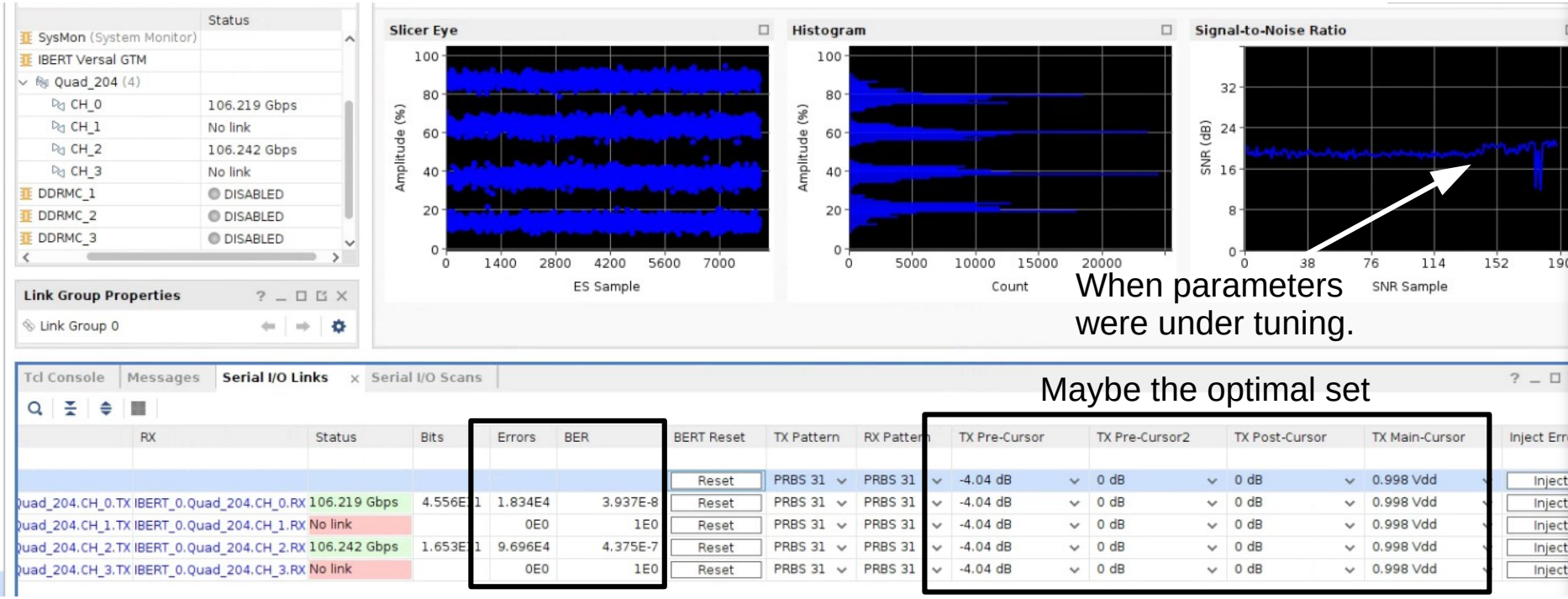
The best indicator of goodness of transmission: **0 error (BER)**

A good set of parameters

Status	Bits	Errors	BER	BERT Reset	TX Pattern	RX Pattern	TX Pre-Cursor	TX Pre-Cursor2	TX Post-Cursor	TX Main-Cursor	Inject Error	TX Reset	RX Reset
				Reset	PRBS 31	PRBS 31	0 dB	0 dB	0 dB	0.502 Vdd	Inject	Reset	Reset
56.414 Gbps	3.214E1	0E0	3.003E-12	Reset	PRBS 31	PRBS 31	0 dB	0 dB	0 dB	0.502 Vdd	Inject	Reset	Reset
56.414 Gbps	2.737E1	0E0	3.556E-12	Reset	PRBS 31	PRBS 31	0 dB	0 dB	0 dB	0.502 Vdd	Inject	Reset	Reset
56.402 Gbps	1.891E1	0E0	6.095E-12	Reset	PRBS 31	PRBS 31	0 dB	0 dB	0 dB	0.502 Vdd	Inject	Reset	Reset
56.402 Gbps	1.202E1	0E0	1.143E-11	Reset	PRBS 31	PRBS 31	0 dB	0 dB	0 dB	0.502 Vdd	Inject	Reset	Reset

PAM4 106 Gbps

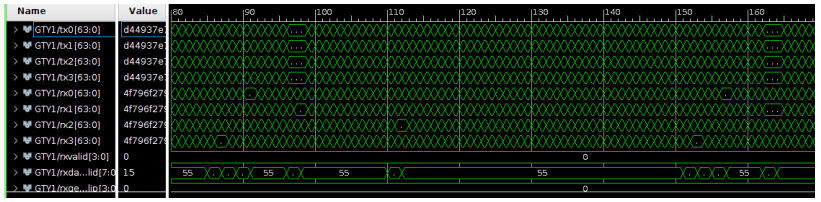
- 106 Gbps: Half density mode by combining two lanes.
- But it is a bit different from simple channel-bonding.
  - The dynamic condition is different from ~50 Gbps.
- Not so stable. Even with the best parameter set, BER will be still around 10E-8.
- More studies on the hardware is needed.



Utilization and protocol development

- Since there is no encoder, utilization is simple:
  - (line rate) = (data width) x (userclk rate)
  - data width for NRZ: 32, 40, 64, 80, 128, 160, and 256 bits
  - data width for PAM4: 64, 80, 128, 160, 256, 320, and 512 bits.
    - 320 and 512 are only for half density mode.

- Protocol development:
  - 8B/10B: Just simply tested. It prefers lower line rate. Will use GTYP to test it.
  - 64B/66B: A protocol with synchronized gearbox has been made and tested.
    - 1 clock halt in every 33 clocks.
    - Based on my Belle II TRG protocol design.



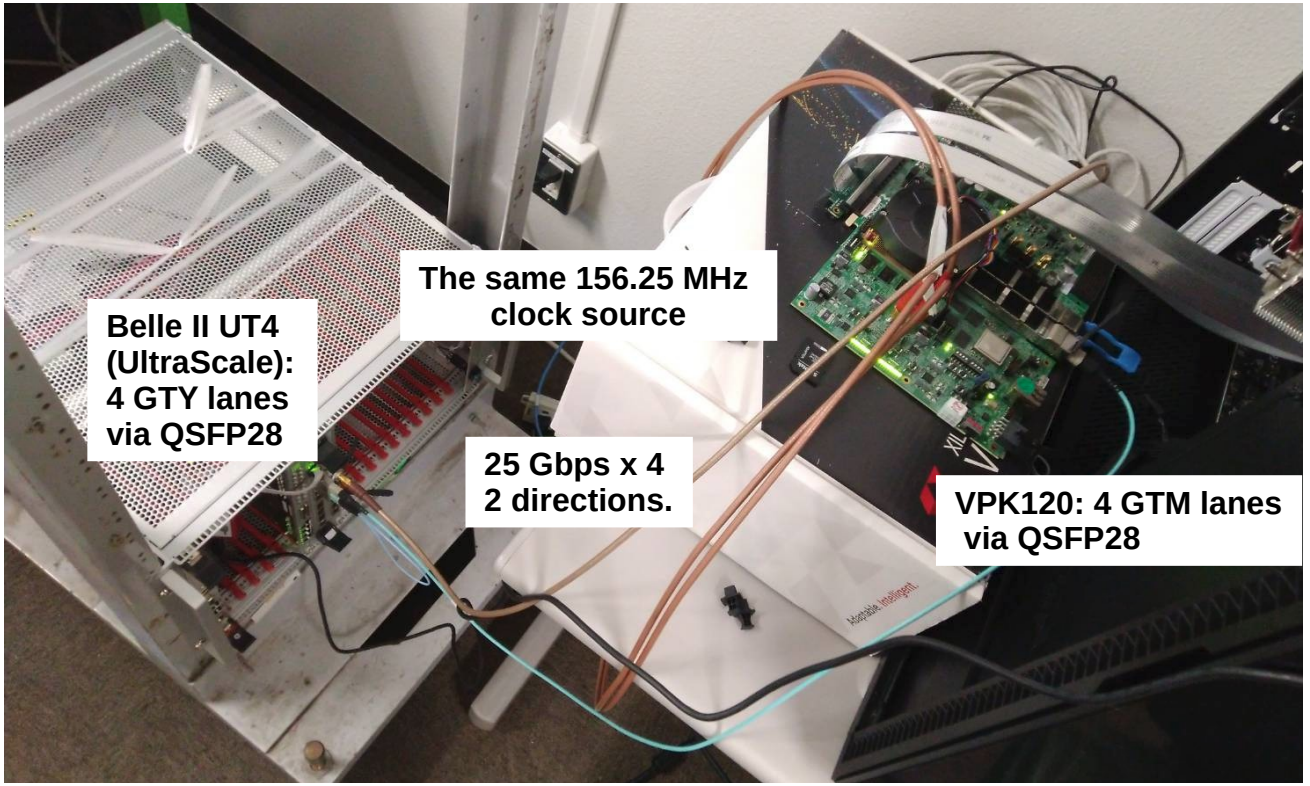
synchronized gearbox 64B/66B

  - Raw mode with No encoding: A new generalized protocol has been also made.
    - Similar handshake logic to my Belle II TRG protocol design.
    - Generalized for different data widths.
    - (de)scrambler for DC balance.
    - Tested to be stable for both NRZ and PAM4.
    - In the future, we will use this protocol for communication between Versal device and Belle II UT4 and others.

Belle II UT4 to VPK120 transmission test

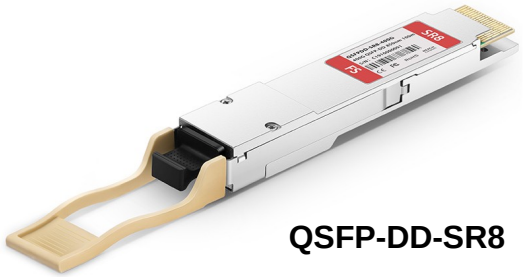
- Transmission between Belle II UT4 (Xilinx UltraScale) and VPK120 has been tested.
  - 25 Gbps x 4 lanes.
  - 2 directions.
  - Using the user-defined generalized protocol. 80 bits. ~312.5 MHz.
  - Stable.

Belle II UT4



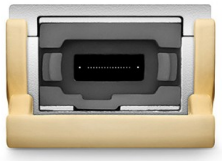
QSFP-DD module and MPO-16 cable

- I received the QSFP-DD modules and MPO-16 cables in this week.
  - QSFP-DD-SR8. Up to 50 Gbps x 8 lanes. From FS.



QSFP-DD-SR8

- QSFP-DD is only for MPO-16.
  QSFP(28) is only for MPO-12.
  - They are not compatible with each other.
  - To make a link between a QSFP and QSFP-DD, we need to use splitter cable or patch panel.



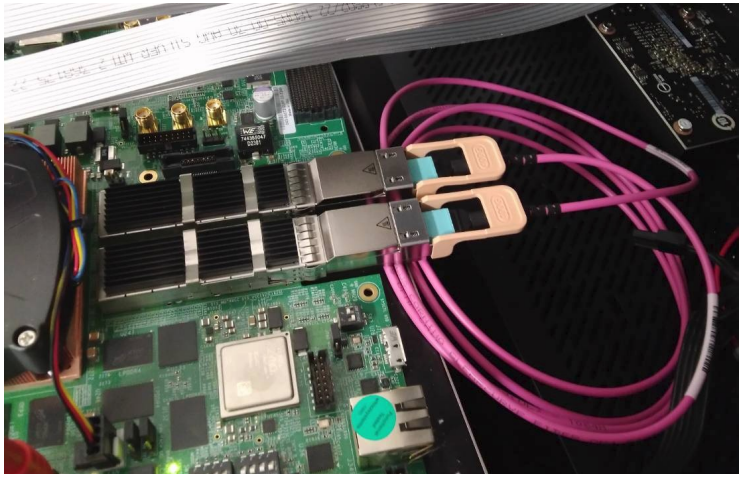
QSFP-DD

MPO-16

QSFP28

MPO-12

- Now preparing for more tests with this real setup:



Latency for Versal GTM

- Latency is a big concern for L1 TRG system.

- The following are the max. simulation values from Xilinx website with No encoding.
  - Measured latency in **bold**: Based on our generalized protocol.
  - Latency for QSFP and QSFP-DD are almost the same.

- For the same setup in different speed, latency in term of clock-cycle is basically the same.
  - Higher speed is preferred as the processing latency is much smaller.
  - In general, latency of GTM is much larger that that of UltraScale(+) GTY or so.

Versal GTM	Unit Interval (UI)	10 Gbps (ns)	25 Gbps (ns)	56 Gbps (ns)	106 Gbps (ns)
NRZ 64b	5833	583 **640**	233 **256**		
NRZ 160b	4964	496 **730**	198 **237**		
PAM4 160b	2957			53 **97**	
PAM4 256b	3233			57 **133**	
PAM4 320b	3095				29 **66**
PAM4 512b	3690				35

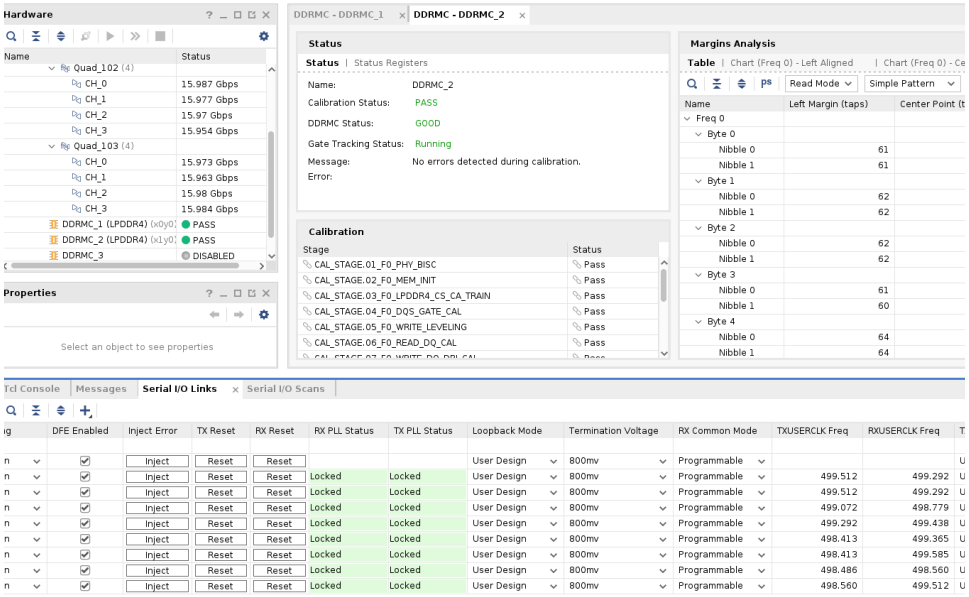
Latency for Versal GTYP and UltraScale(+) GTY

- Versal GTYP: will be tested soon.
- The following are the simulation values from Xilinx website with internal encoder.
  - UT3: Virtex-6
  - UT4: Virtex UltraScale
- Measured latency in **bold**: Based on the Belle II TRG protocol.

	Raw (UI)	Raw + Async. 64B/66B (UI)	10 Gbps, Raw (ns)	10 Gbps, 64B/66B (ns)	25 Gbps, Raw (ns)	25 Gbps, 64B/66B (ns)
Versal GTYP 64/64	1127					
Versal GTYP 64/32	688					
UT4 GTY 64/64	768	1458	77 **115**	146 **147**	31 **33**	58 **58**
UT4 GTY 64/32	414	990	41 **90**	99 **122**		

PCIe-CPM test

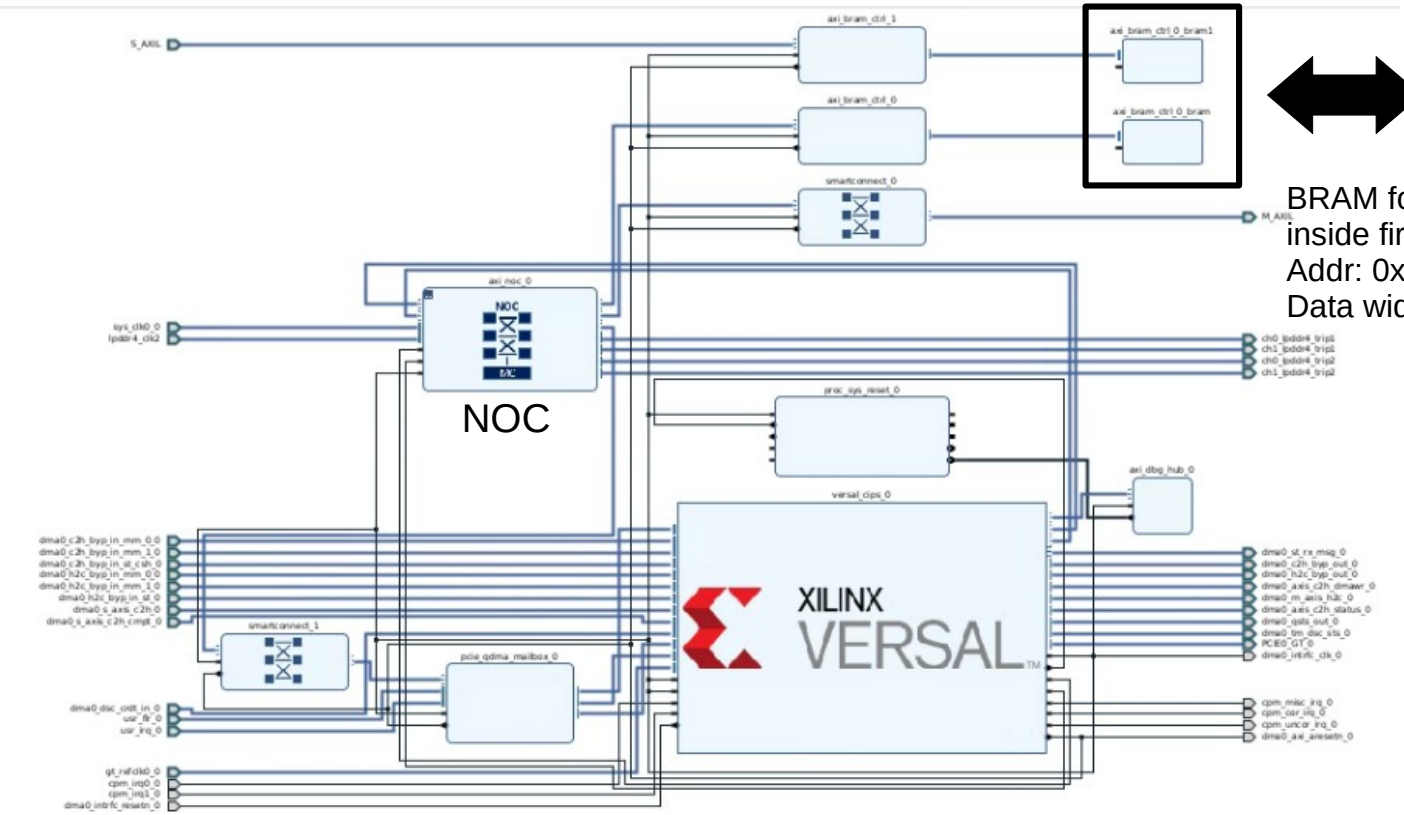
- CPM-PCIe example from Xilinx: XTP712
  - CPM: building block design for PCIe with integrating DMA, CIPS, NOC, etc.
  - PCIe Gen4 x8:
    GTYP links are up. 16 Gbps per lane.
- Driver software: QDMA, also a Xilinx IP.
- Data exchange test with the QDMA software:
- We spent much time in mine-sweeping
  - Will start to make real protocol for event data readout purpose.
    - Similar to the one in Belle II DAQ.




```
[root@cef01 linux-kernel]# ./bin/dma-ctl dev list
qdma02000       0000:02:00.0    max QP: 8, 0~7
qdma02001       0000:02:00.1    max QP: 0, -~-
qdma02002       0000:02:00.2    max QP: 0, -~-
qdma02003       0000:02:00.3    max QP: 0, -~-
[root@cef01 linux-kernel]# ./bin/dma-ctl qdma02000 q add idx 0 dir bi
dma-ctl: Warn: Default mode set to 'mm'
qdma02000-MM-0 H2C added.
qdma02000-MM-0 C2H added.
Added 1 Queues.
[root@cef01 linux-kernel]# ./bin/dma-ctl qdma02000 q start idx 0 dir bi
dma-ctl: Info: Default ring size set to 2048
1 Queues started, idx 0 ~ 0.
1 Queues started, idx 0 ~ 0.
[root@cef01 linux-kernel]# ./bin/dma-to-device -d /dev/qdma02000-MM-0 -s 32
size=32 Average BW = 177.377688 KB/sec
[root@cef01 linux-kernel]# ./bin/dma-from-device -d /dev/qdma02000-MM-0 -s 32
size=32 Average BW = 132.445391 KB/sec
[root@cef01 linux-kernel]# ./bin/dma-ctl qdma02000 q stop idx 0 dir bi
Stopped Queues 0 -> 0.
[root@cef01 linux-kernel]# ./bin/dma-ctl qdma02000 q del idx 0 dir bi
Deleted Queues 0 -> 0.
```

Plan for PCIe study

- Develop an event data readout firmware/software which are similar to Belle II PCIe40 readout.
  - Single-port BRAM → Dual-port BRAM.
- Then we can measure the readout throughput among different PCIe generation.
- Brainstorm ongoing...
  - Will consult with Belle II DAQ group and IJCLab for technical support.



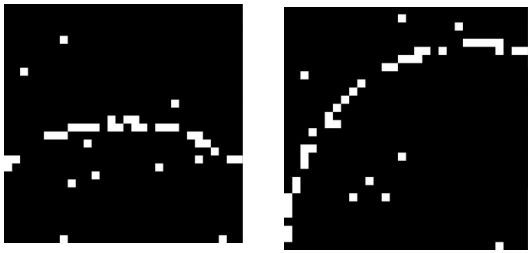
Continuous event data reading and writing

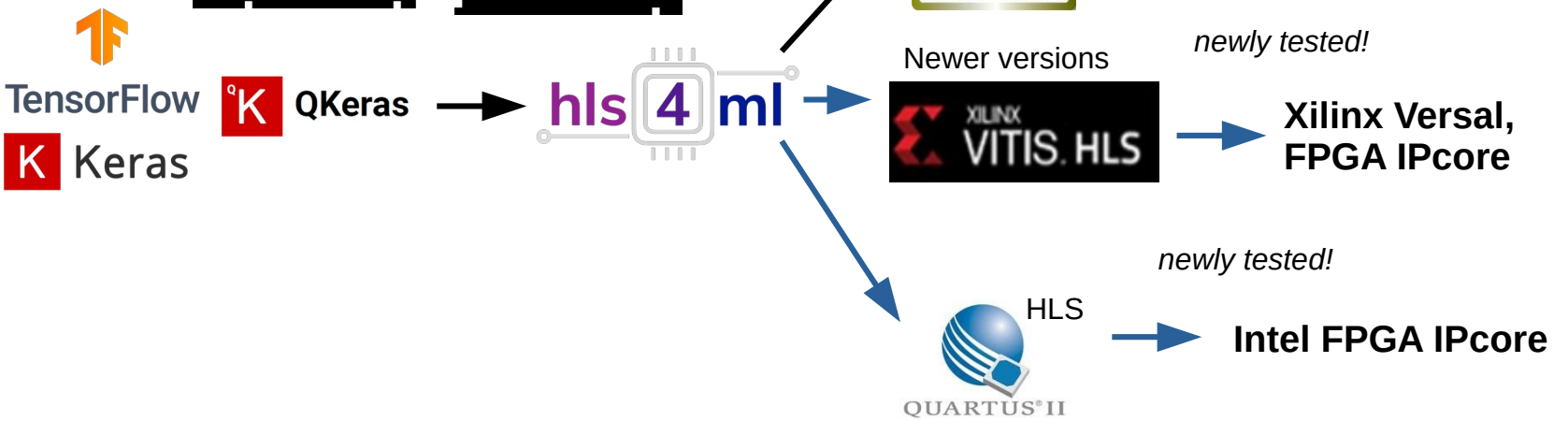
BRAM for data R/W
inside firmware.
Addr: 0x0000~0x7FFF
Data width: 32 bits

General study on ML implementation with hls4ml

- hls4ml: A package for machine learning inference in FPGA.
  - Already lots of utilizations with Vivado HLS in Belle II and ATLAS.

- Yiyang Ding, our summer student, performed general studies on it.
  - Implementation to the Versal board is validated.
  - A NN model for simple tracker and tested with VPK120!
  - Also tested with Intel FPGA with Quartus.



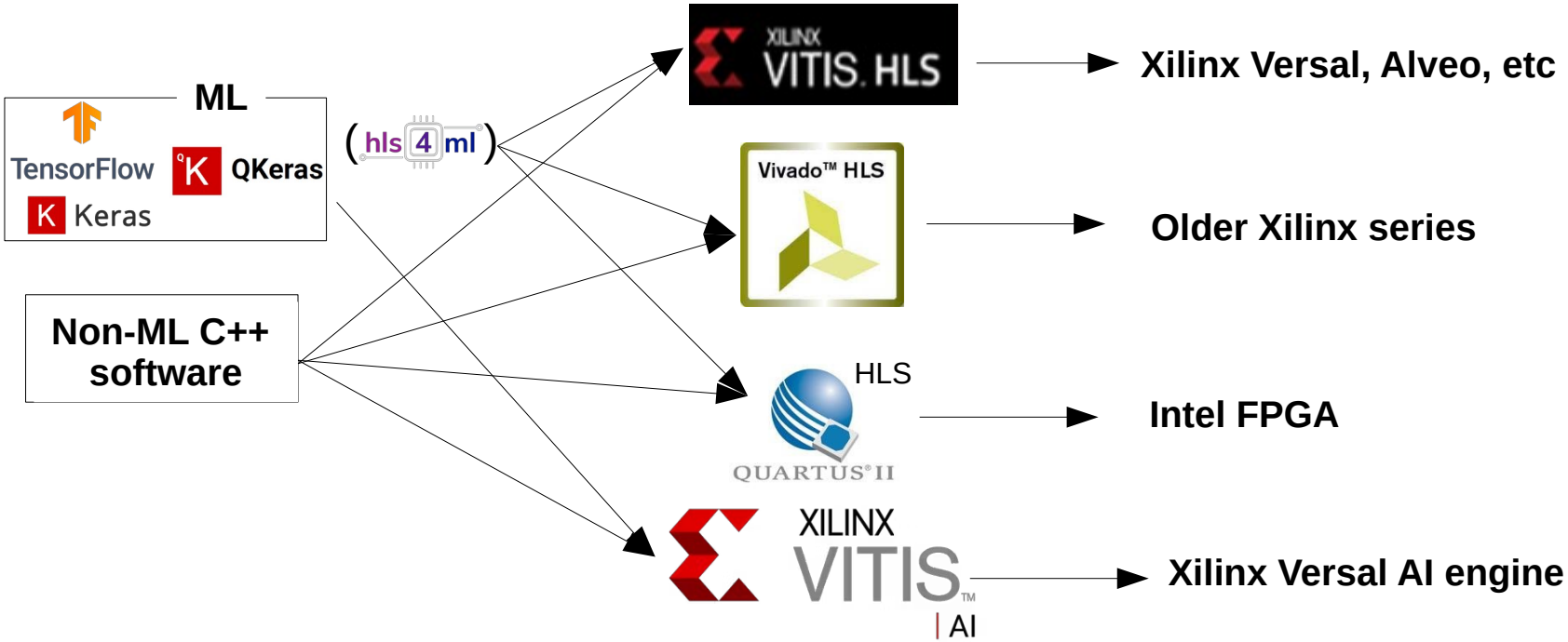


Work plan for HLS, AI engine, etc

- Considering algorithm implementation:
  - HDL logic in firmware.
  - HLS: software → firmware.
  - AI engine.

Depend on the different targets, our selection on FPGA differs. A strong FPGA? ACAP with AI engine? Acceleration card?

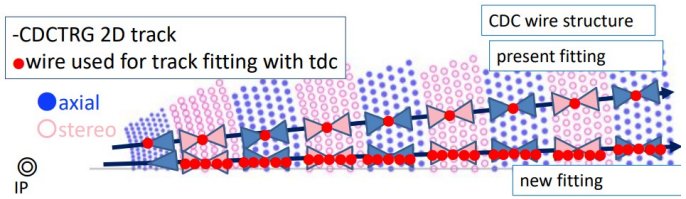
- Not only the hls4ml, HLS tools has much more for ML and non-ML application.
  - Similarly, Versal AI engine requires a different design flow to make software/firmware.
  - We expect to finish this roadmap step-by-step, and study the implementation of different algorithms from Belle II, ATLAS, etc.



Prospect: new ideas on algorithm development

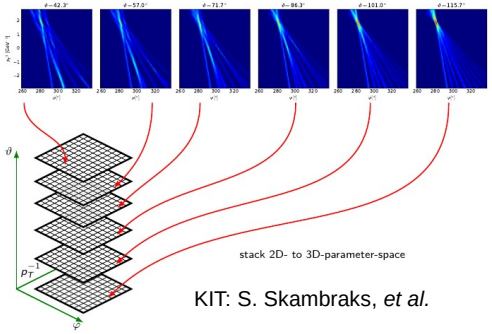
Additional dimension: More resource in FPGA

Extension of more input info:



KEK: T. Koga, *et al*.

2D→3D Hough tracking:

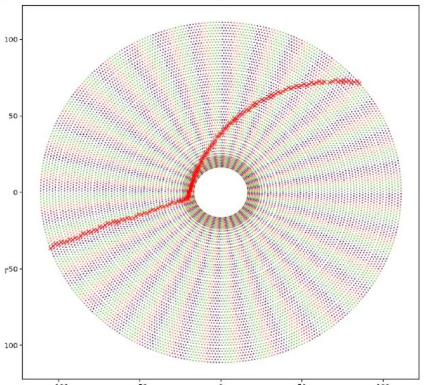


KIT: S. Skambraks, *et al.*

More than NN: CNN or GNN?

KIT, TUM, MPI: Belle II AI trigger group

CNN tracking

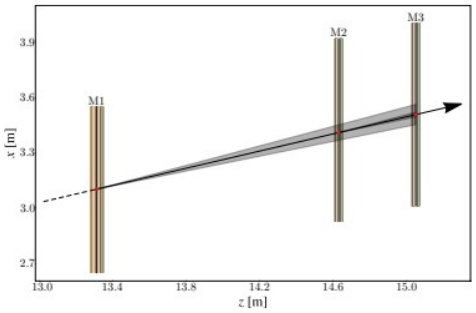


Aglining methode ploted on the CDC cross-section

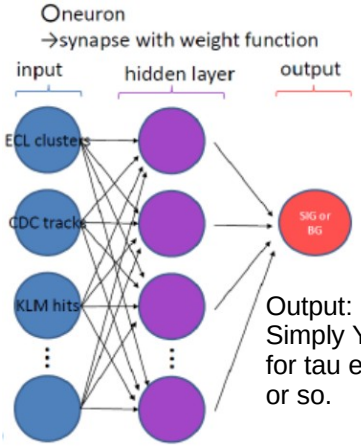
NN: hls4ml

ATLAS fast muon tracking with Neural Network:

arXiv:2202.04976



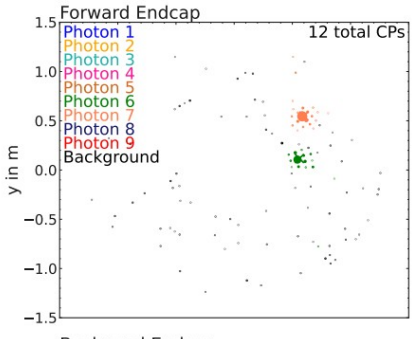
Neural Network for τ event trigger:



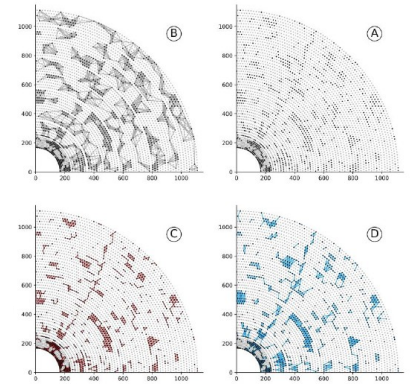
Output: Simply Y/N for tau event or so.

KEK: T. Koga, R. Nomaru.

GNN clustering



GNN tracking



Dataset: displaced_processed_simulated_2_tracks_0_nominal-phase3

Summary

- In Japanese HEP community, we started a project using the evaluation kits of the Xilinx Versal ACAP with different experimental groups. The target is future R&D of a new universal FPGA device.

- Some of the fundamental functionalities of the Versal evaluation kits have been studied, such as firmware making, high-speed transmission, PCIe, and HLS for ML inference.

- Next step:
  - PAM4: Try different QSFP-DD module
  - PCIe: Make practical protocol based on Xilinx IP.
  - More basic studies on HLS tools and AI engine will be performed.
  - Then start to implement different physics algorithms for different experiments.

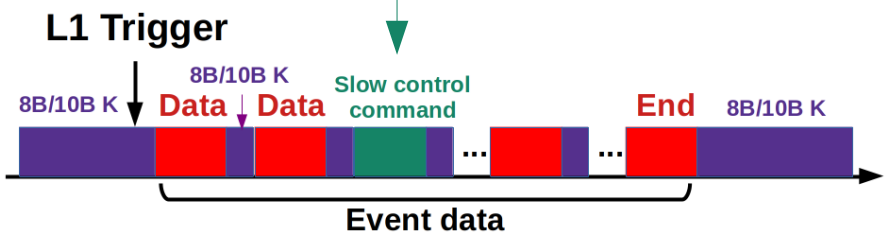
- The collaboration between IJCLab group and KEK group will be helpful for the hardware technical R&D, and the experience exchange between E-sys, Belle II, ATLAS, and LHCb can have great impact for common development in far future.

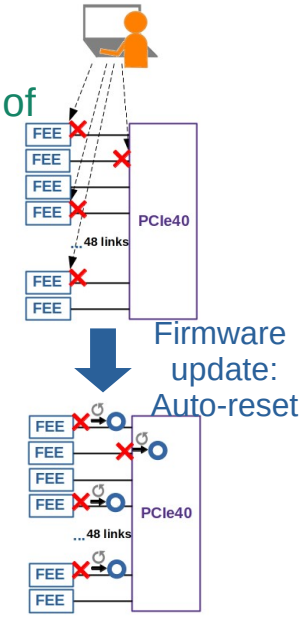
Backup

Belle II DAQ: Belle2Link

- **Belle2Link protocol**:
  - Line rate 2.54 Gbps.
  - Framing transmission using different 8B/10B K characters.
  - Optical link and high-speed transceivers of FEE and PCIe40.
  - Two major functionalities:
    - Transferring **detector FEE data** w.r.t L1 trigger. crc16 and crc32 checksum included.
    - **Slow control**: exchanging register content between FEE and readout as a combination of address and payload data.

Development by IHEP:
D. Sun et al., Phys. Procedia, vol. 37, pp. 1933-1939, 2012.





- **Improvement: Auto-reset on the link.**
  - Link recovery via CUI/GUI is time-consuming.
  - Auto-reset on transceiver by monitoring status flags: PLL lock, decoding error, disparity, etc.
  - Reliable recovery: ~100% readiness.
  - Update is based on different transceiver of FEE:

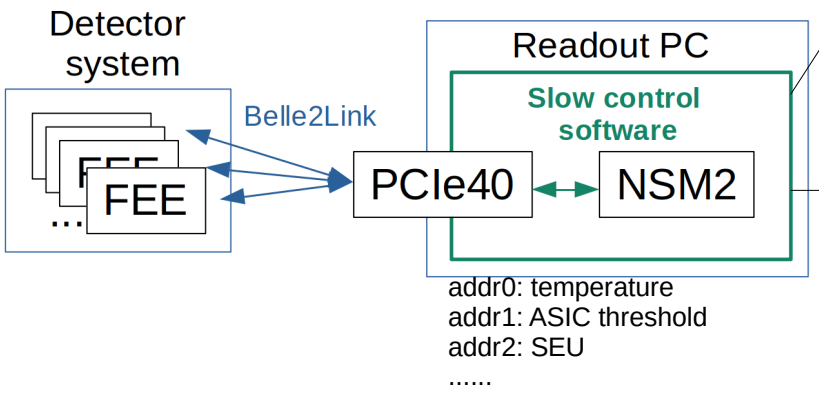
Detector FEE	Transceiver
SVD	Spartan-6 GTP
CDC	Virtex-5 GTP
TOP	Kintex-7 GTX
ARICH	Virtex-5 GTP
ECL	Spartan-6 GTP
KLM	Virtex-6 GTX
TRG	UT3: Virtex-6 GTX, GTH UT4: UltraScale GTH, GTY

Belle II DAQ: Slow control

- Belle2Link: FEE and PCIe40 exchanges information as address/data.

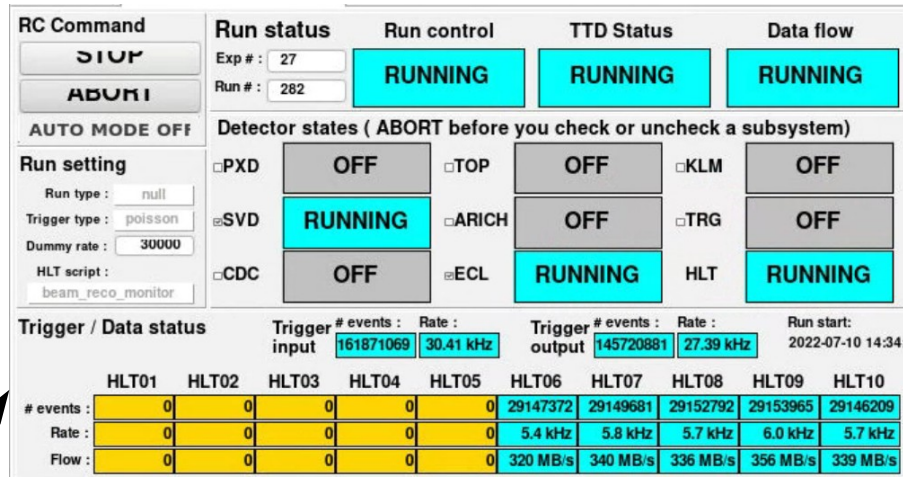
- Slow control software:
  - Runs in readout PC, and controls Belle2Link.
  - NSM2: Network Shared Memory v.2.
    Define address/data as variables.
  - Configuration and monitoring for each detector.

- Integrated in Belle II global run control.
- Logged by EPICS.

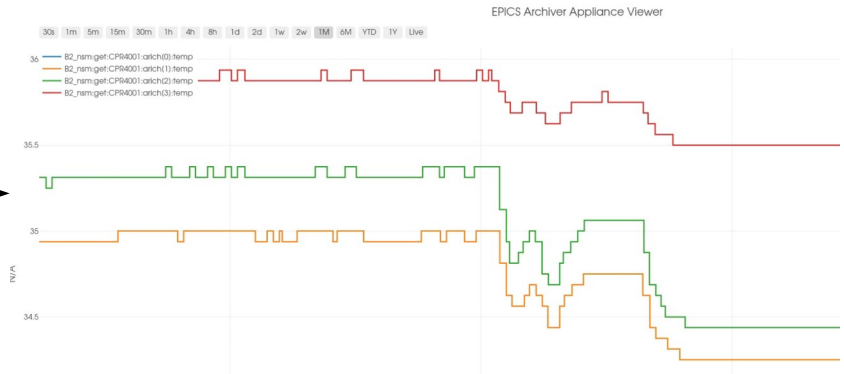


All detector systems

GUI for Belle II global run control



Plots of archived information

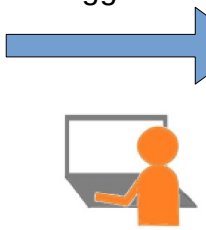


EPICS

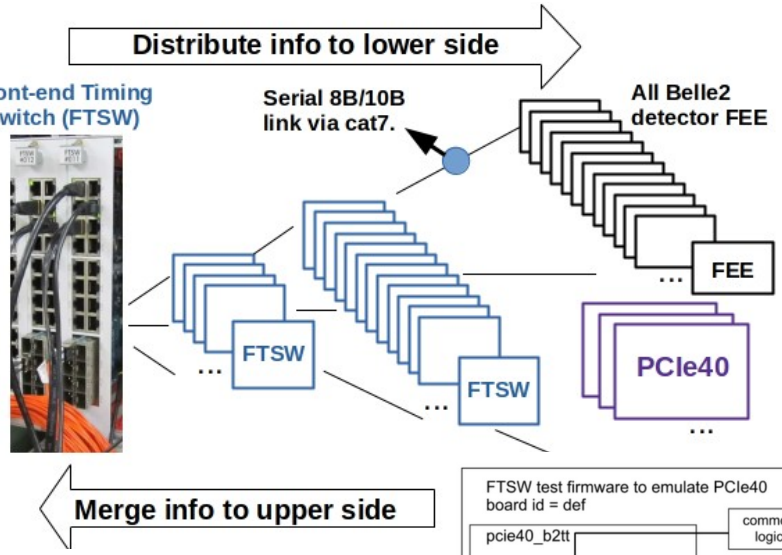
Belle II DAQ: TTD system

TTD → all FPGA:

- Injection veto
- SuperKEKB Clock
- L1 trigger

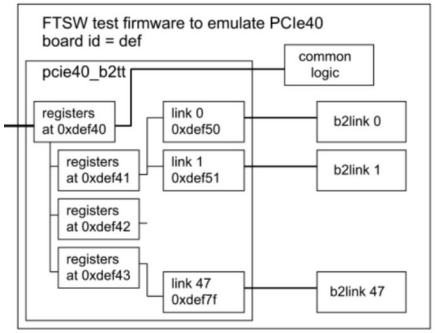


Run control & monitor via GUI/CUI.



TTD ← all FPGA:

- Run status, error, busy of devices.

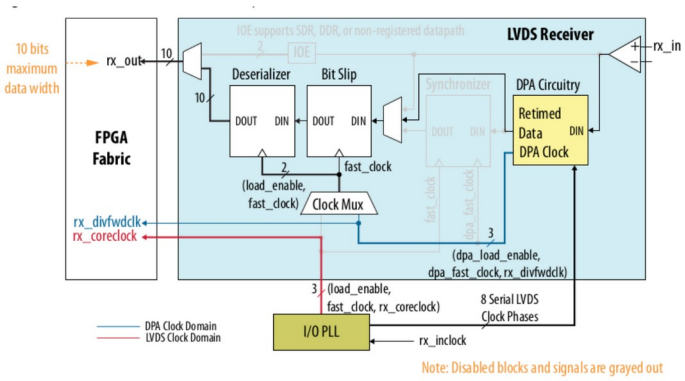


See the poster by **Dmytro Levit** for more details on **TTD system**

- PCIe40: Information of all 48 channels needs to be reported:
  - New address scheme to merge 48x info.

- Clock, and the signals (to be driven by the same clock source) are distributed by TTD system to PCIe40:
  - Stability affected by external noise in Electronic Hut.

- **Improvement: Intel Serdes IPcore with on-board clock.**
  - Stable under external noise.
  - Soft-CDR to handle jitter.
  - Reduce operation down time.

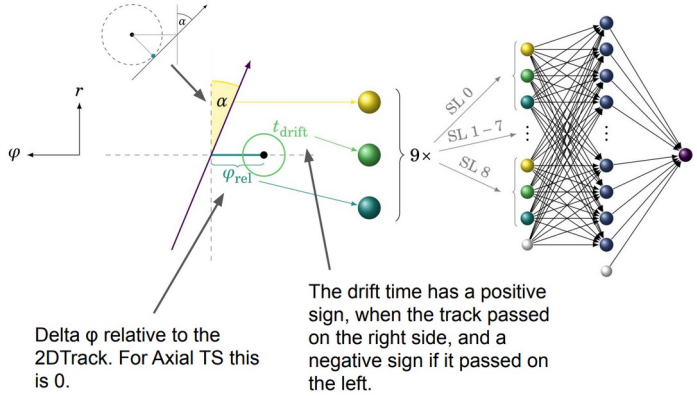


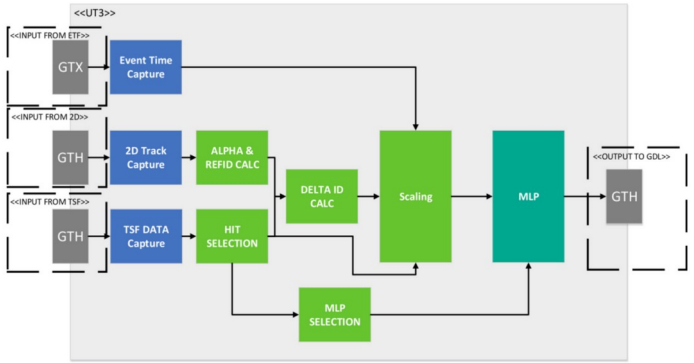
Neural z trigger

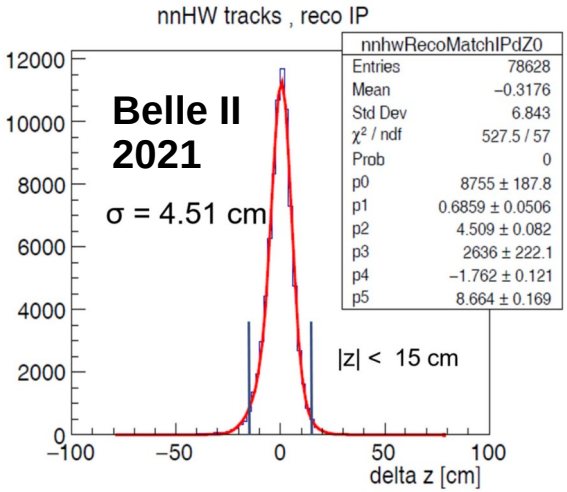
Karlsruher Institut für Technologie

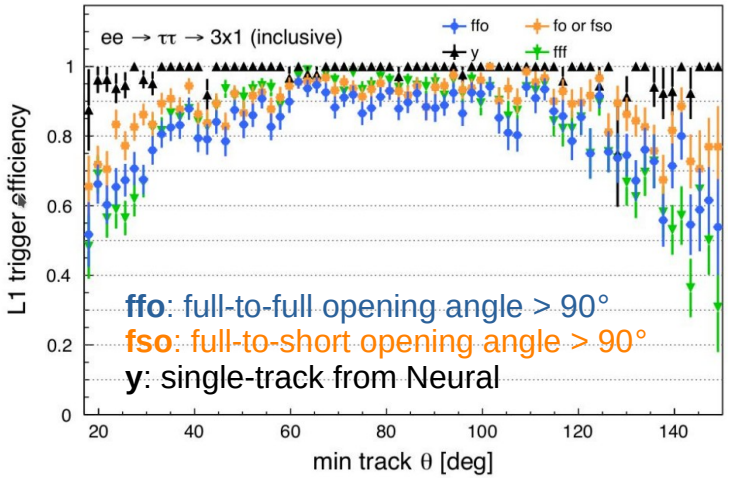
S. Neuhaus et al 2015 J. Phys.: Conf. Ser. 608 012052
Kai Lukas Unger et al 2023 J. Phys.: Conf. Ser. 2438 012056
F. Meggendorfer, DPG Conference 2021
Thesis: S. Skambraks, S. Pohl

- In addition to the conventional 3D tracker based on fitting method, Belle II has a Neural Network 3D tracker (NN) running in parallel in the system.
- Input the 2D tracker and stereo TS info: Crossing angle, drift time, φ relative to 2D Track .
  - Obtain z_0 and θ.





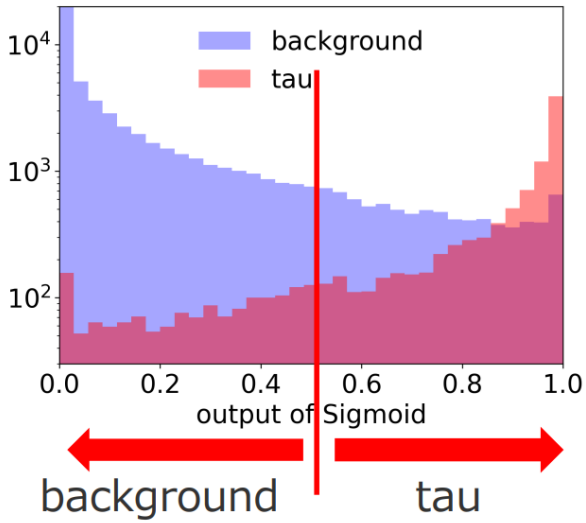




Plots by P.Rados, A.Rostomyan (DESY)

ML tau trigger

- Global trigger receives the cluster information from ECLTRG.
  - Input the position and energy information of clusters to a Neural Network, and determine if it is a tau event or not.
  - A kind of topological application.
  - Based on hls4ml.
  - Validated and will be implemented in 2024 runs.



hie: ECL energy sum

ecltaub2b: ECL cluster based logic