

Realtime Machine Learning (Versal FPGA), WP 5.4

Yun-Tsung Lai

KEK IPNS

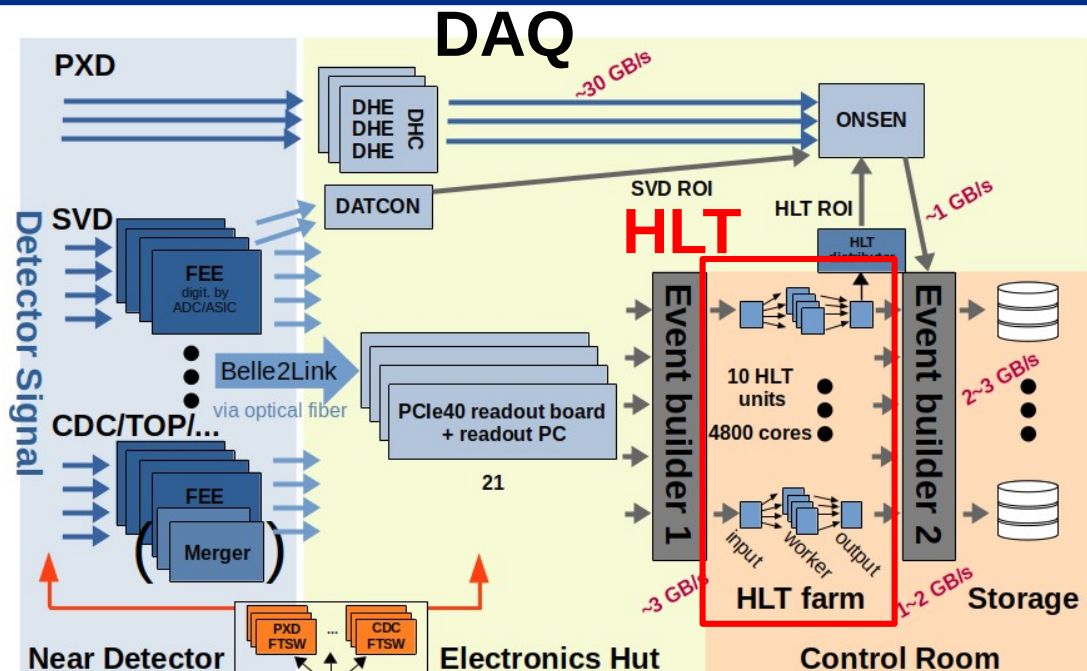
ytlai@post.kek.jp

JENNIFER3 kickoff meeting

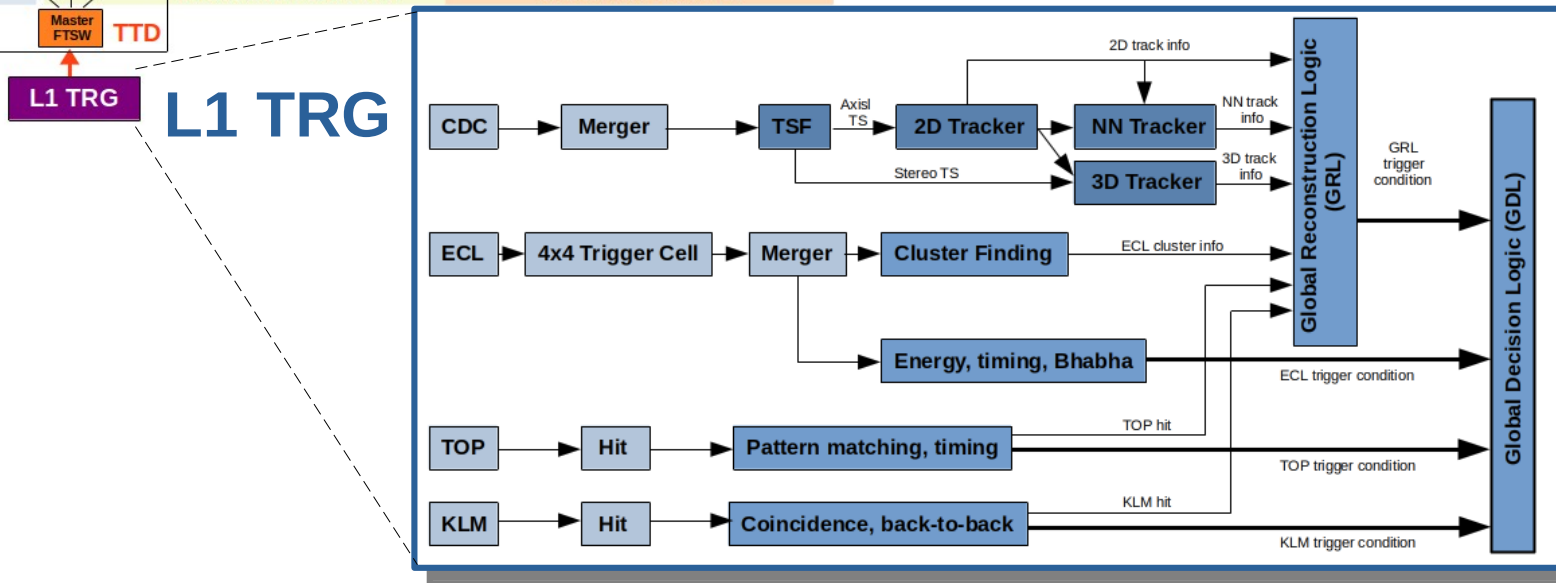
27th Jan., 2025



Realtime ML @ FPGA for Trigger/DAQ: Belle II



- WP5.4: **Realtime ML @ FPGA**
 - For **L1 TRG** and **HLT**
 - Fast processing with small latency and good computation density: Benefit for our experimental DAQ considerations.
- In this report, two major points:
 - **Hardware: Versal ACAP**
 - **ML inference: techniques**

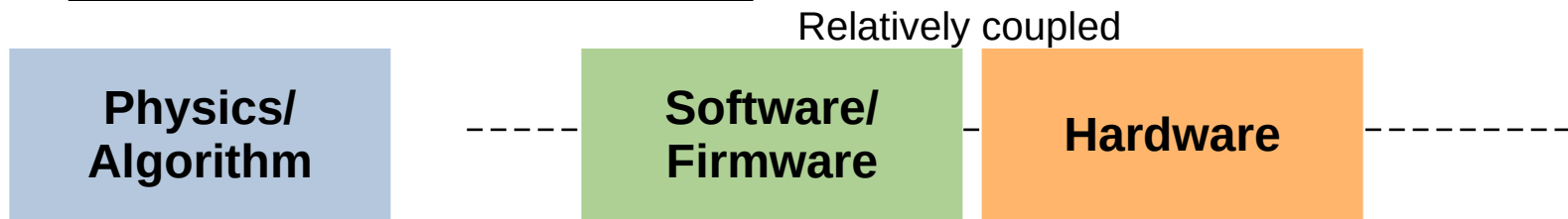


"Trigger" system

- Considering the trend in our community, two major types of **Trigger** systems for real-time processing/filter:
 - A very personal and technical point of view, as the boundary in between is getting vague nowadays.

- **Hardware/low-level:** Rely on the FPGA programmable logic (PL) design based on HDL/RTL logics to achieve short-latency quick processing (in $O(\mu\text{s})$).

Keywords: FPGA, PL, HDL, RTL, HLS, ML inference (hls4ml, etc), Versal ACAP, Versal AI engine, short latency in $O(\mu\text{s})$.



- **High-level:** Software-based algorithm development with computing farm. In addition to CPU/GPU, FPGA acceleration is a new application.

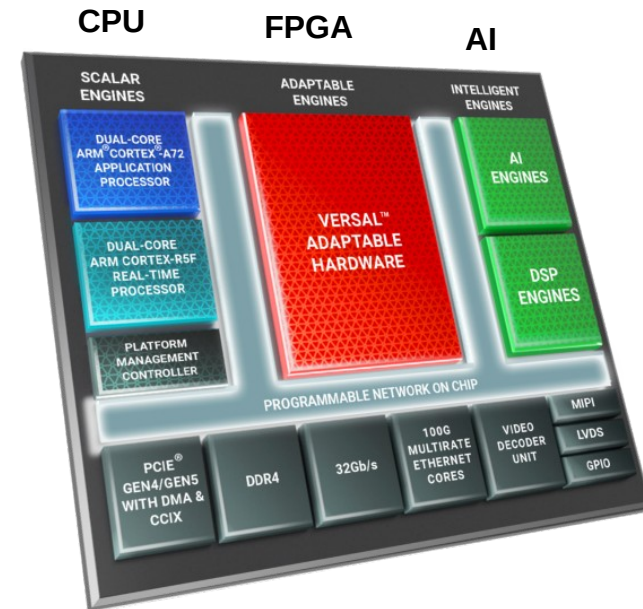
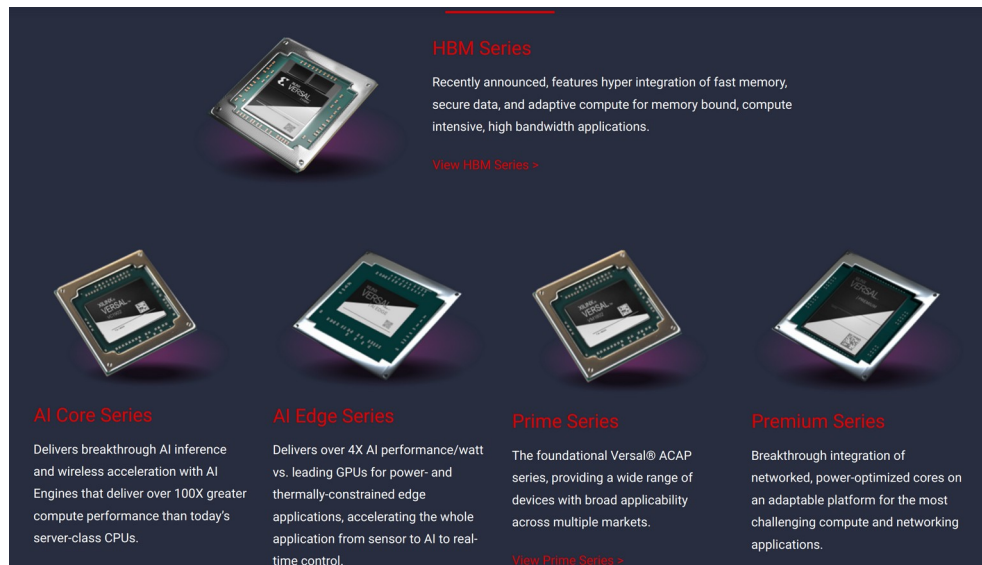
Keywords: Hardware acceleration platforms with CPU/GPU/FPGA, Alveo, Versal acceleration card, Versal DPU.

Versal project

Study on the Xilinx Versal series of ACAP and the associated new features, and look for the possible applications in Trigger/DAQ of experimental particle physics

Versal project @ KEK, Collider Electronics Forum (CEF)

- Our project is mainly based on the Xilinx Versal series of ACAP.
- KEK together with Japanese HEP community purchased a few evaluation kits.
 - 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

Versal project: General plan, roadmap, and collaboration

- 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 engine, DPU, etc.
- Prepare basic application for each of them for other members.

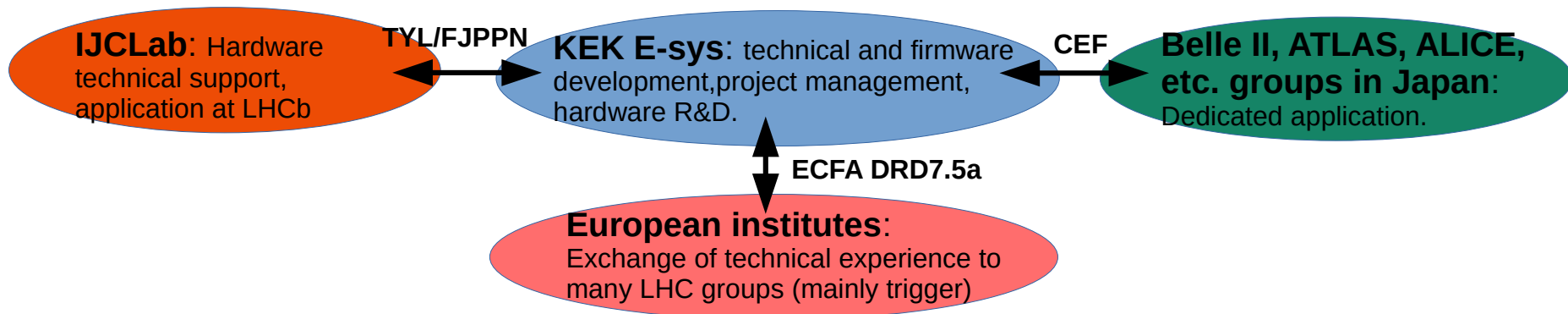
Here we are now with VPK120 and VCK190. →

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:

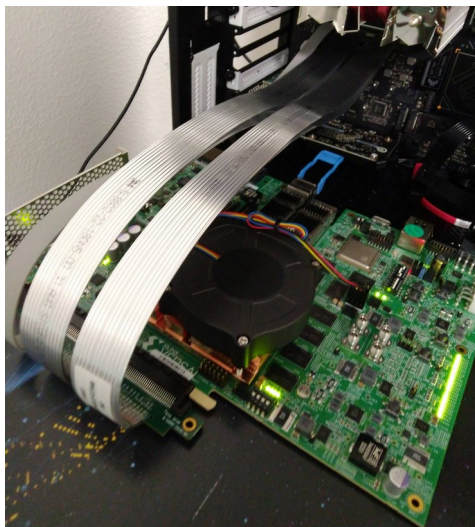
- Future universal device: L1 TRG, DAQ readout, or HLT.
 - Discussion.
 - Schematic/PCB design for the prototype boards.
 - Test with experiments people.



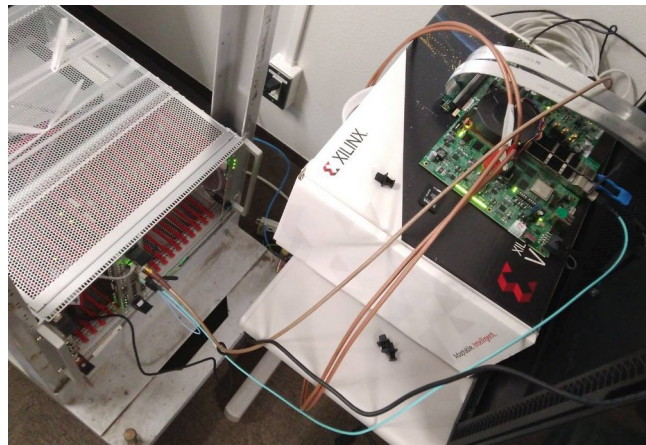
Test benches of Versal kits @ KEK E-sys

- Now we have both VPK120 and VCK190 test benches at KEK E-sys group with host servers.
 - They are opened and shared with our colleagues in CEF.

PC side: PCIe Gen5 x16 slot



**VPK120 test bench:
2023 summer**



**VPK120 connection
to Belle II UT4**

PC side: PCIe Gen4 x8 slot

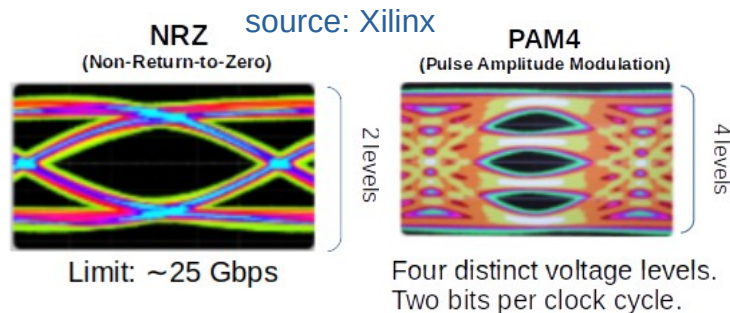


**VCK190 test bench:
2024 March**

Features of Versal: PAM4 data transmission

- **Pulse Amplitude Modulation (PAM4):**

- Four distinct voltage levels to break through the limit of Non-Return-to-Zero (NRZ), which is ~25 Gbps.
- Versal GTM transceiver supports it.
- Suitable for high-speed link in L1 TRG. Hope to be pioneer to use it in future TRG board.

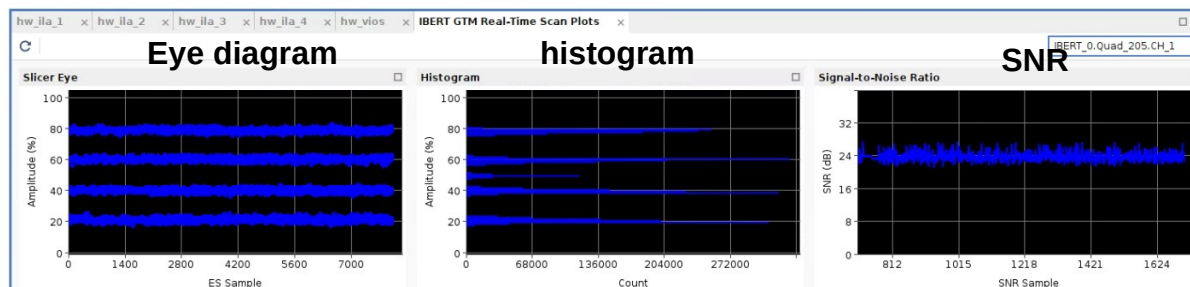
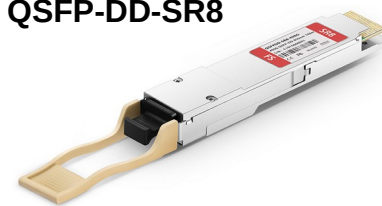


- Real transmission with PAM4 and QSFPDD is tested:

- QSFPDD-SR8 with MPO16, from FS company.
- 53.125 Gb/s x 16 lanes.
- BER ($\sim 10^{-15}$) and latency (~ 200 ns) are measured.

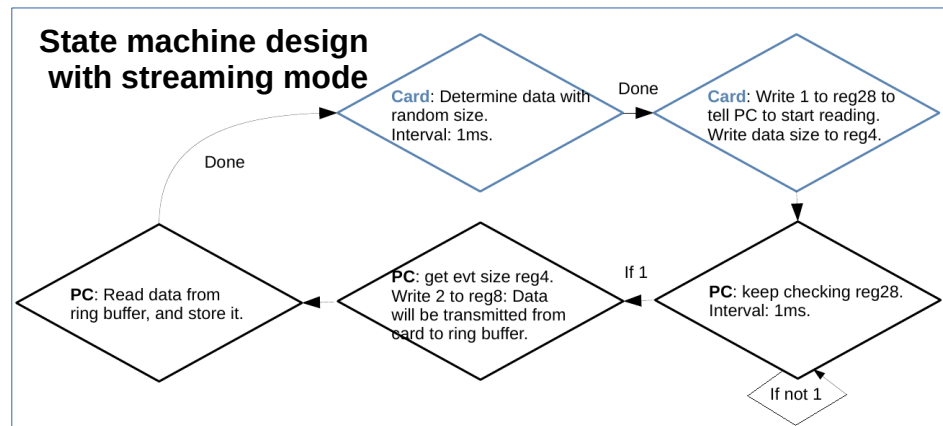
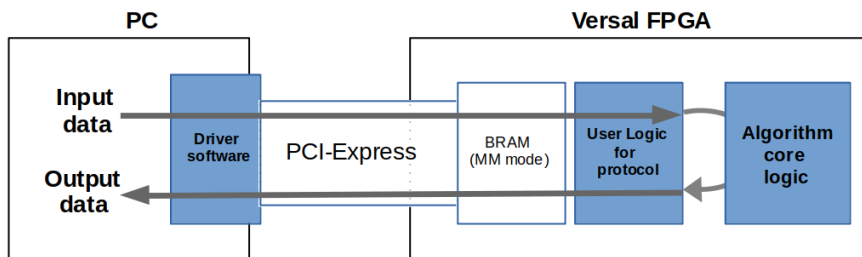


QSFP-DD-SR8



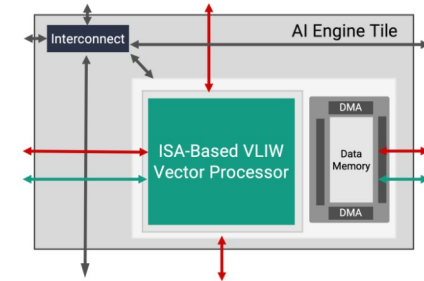
Features of Versal: PCIe

- **PCIe up to Gen5:**
 - PCIe has been popular option in HEP.
 - ALICE, LHCb and Belle II has been using PCIe40 (Gen3).
 - Study the properties of newer generation of PCIe is beneficial for the future readout device's development.
 - Versal device supports up to Gen5.
- For our study, we built up a test bench with host PCs and Versal boards via PCIe
 - The FPGA firmware and PC driver software are based on Direct Memory Access (DMA) IP from Xilinx.
 - Custom protocols are developed for:
 - Continuous event readout
 - Event exchange
 - **Integration with AI engine**

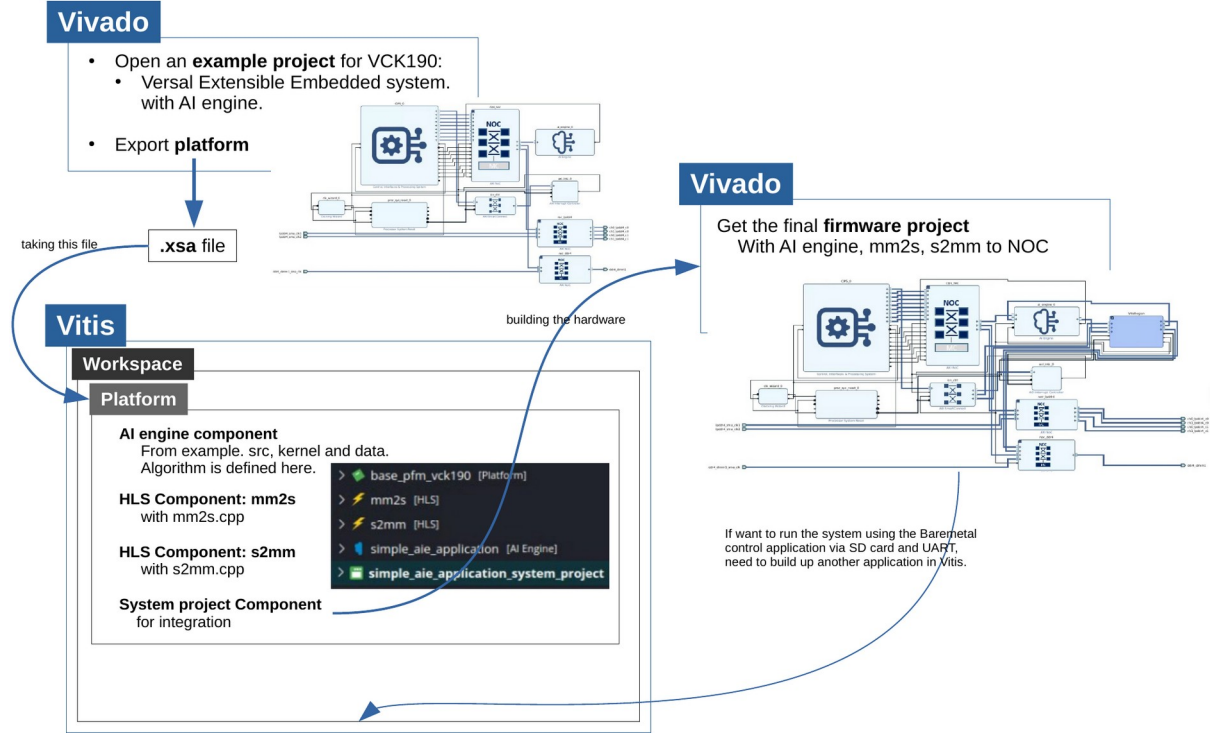
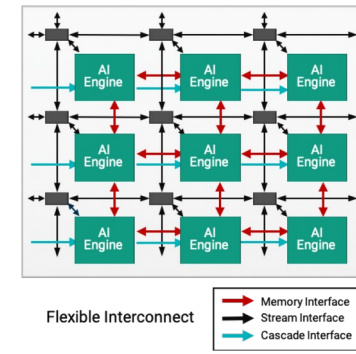


Features of Versal: AI engine

- **AI engine:**
 - Computation engine embedded in Versal FPGA
 - Suitable for algorithm implementation such as ML
 - C programmable framework in Xilinx Vitis tool
 - Integration with FPGA logic using Xilinx Vivado tool

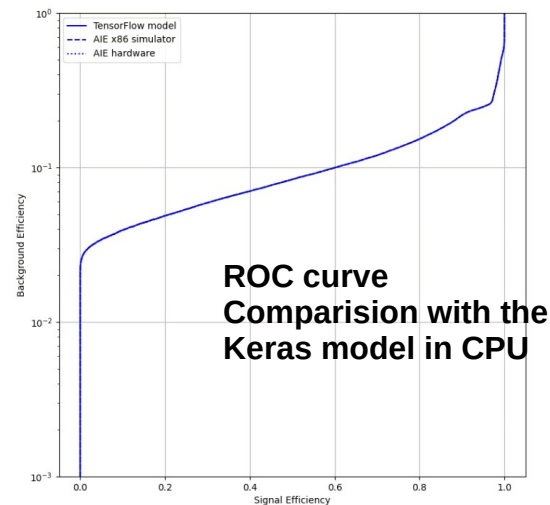
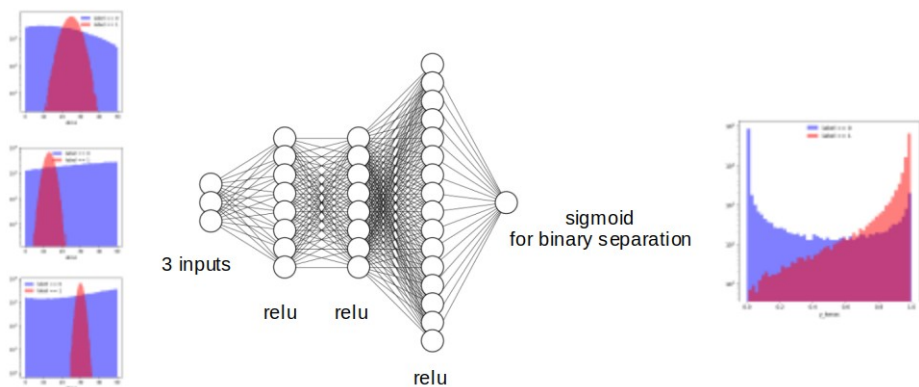


source: Xilinx

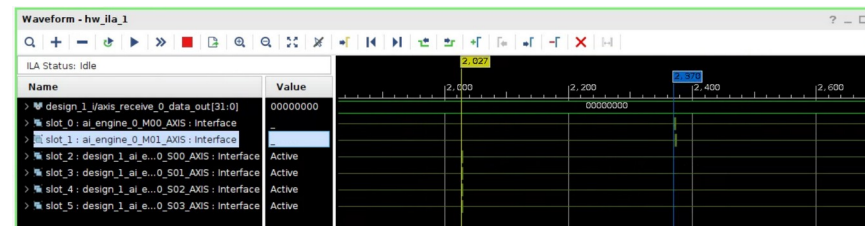
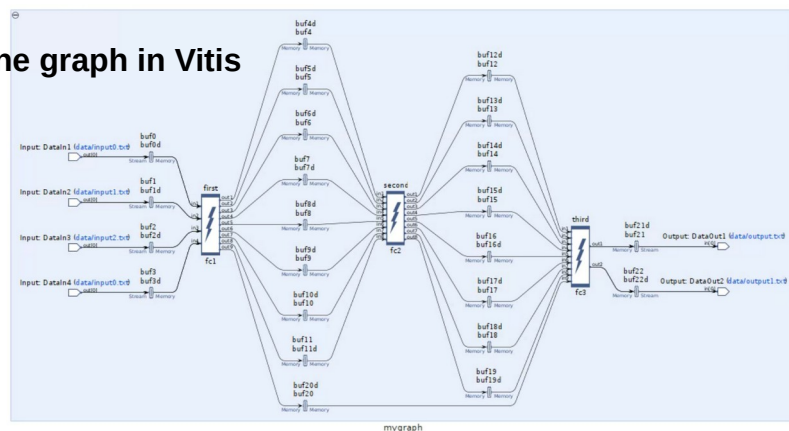


NN in AI engine

- We have already studied the fundamental utilization of the AI engine, prepared tutorial material and given a hand-on course to our colleagues.
- Implementaion of some Belle II logics based on NN are also performed.
- An example of a NN implementation:



AI engine graph in Vitis

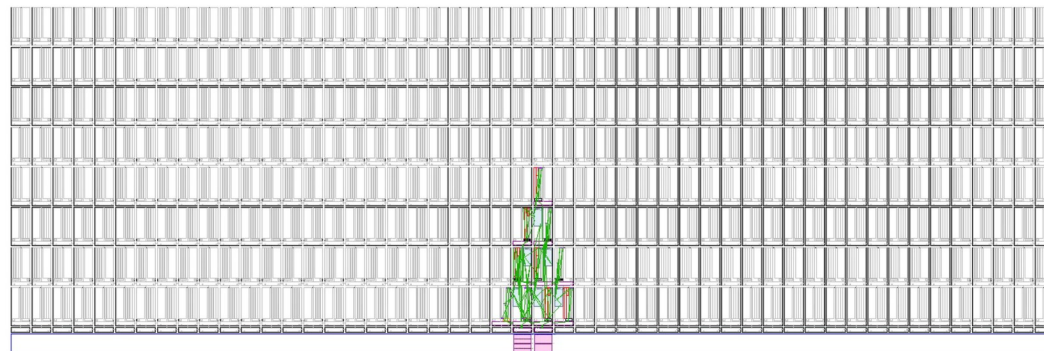


Latency in FPGA: ~3 μ s

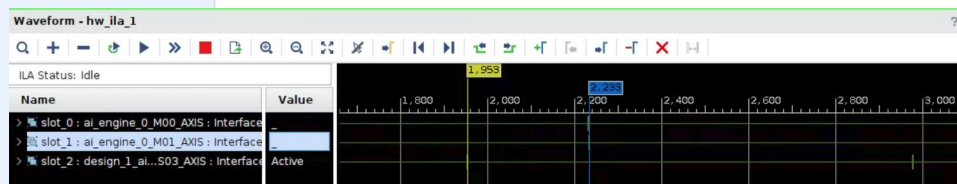
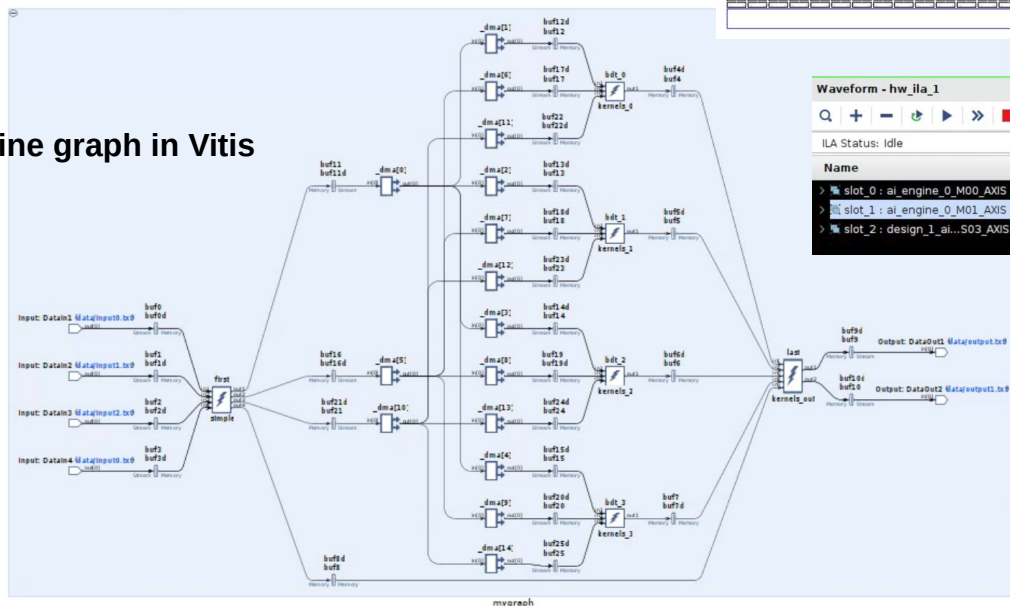
BDT in AI engine

- Another example: A BDT model built by scikit-learn.
 - 3 input features, 20 trees, depth = 3

Resource allocation among the entire AI engine array



AI engine graph in Vitis



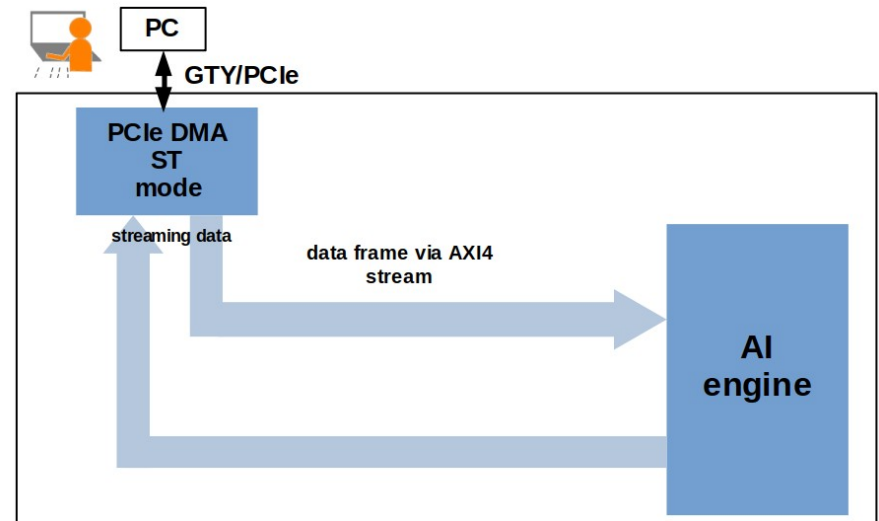
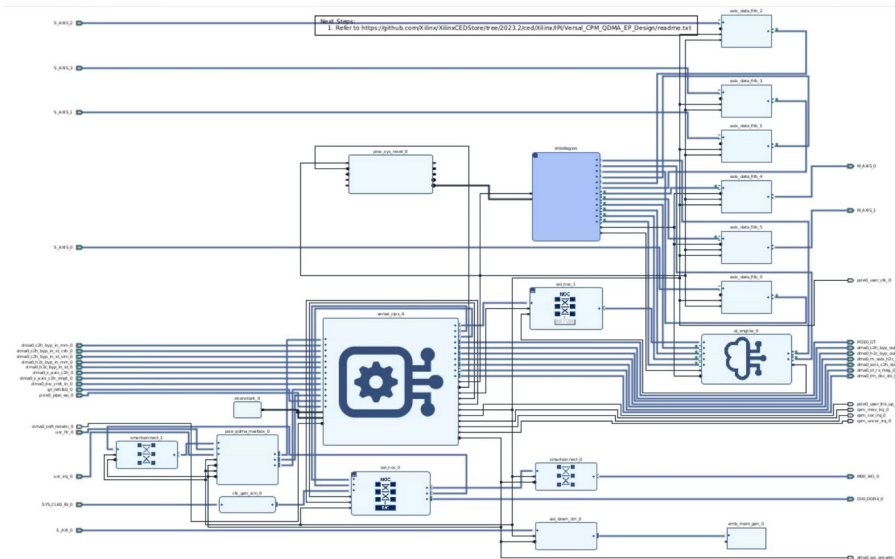
Latency in FPGA: ~3 μ s

Hardware implementation: AI engine + PCIe

- In the Versal FPGA firmware, we integrated PCIe with AI engine
 - PCIe enables the communication between PC and FPGA, so user can submit all the jobs to FPGA to process.
- 50 min for 200,000 events.

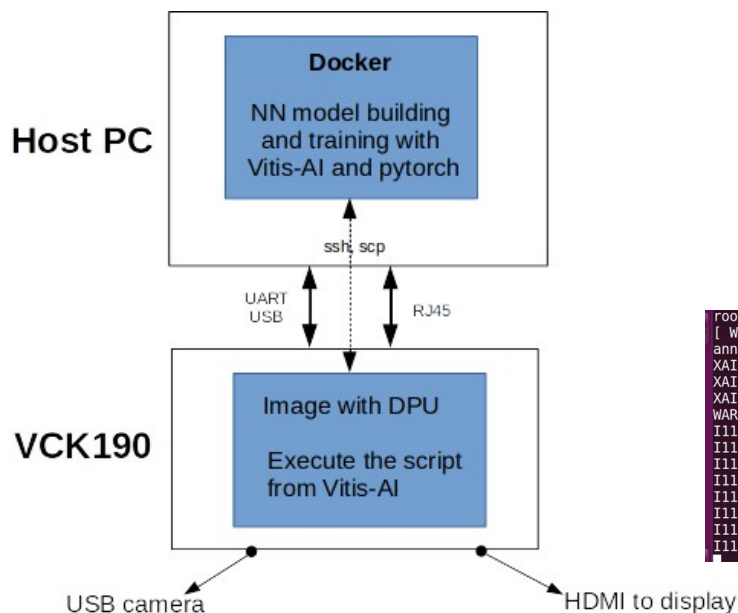
- User can do all the works at PC without touching FPGA during operation.
- **Potential for HLT application.**

FPGA firmware design in Vivado



Features of Versal: DPU

- **Deep Learning Processor Unit (DPU)**
 - A configurable computation engine dedicated to convolutional neural networks.
 - Network quantization based on Xilinx Vitis-AI software
- The design flow does not involve Vivado for PL design.
 - The device is utilized with a small operation system like a server
 - A higher-level application.



```
root@xilinx-vck190-20222:~/03_vck190_pytorch_atlas-top-tagger# python3 app_mt.py
XAIEFAL: INFO: Resource group Avail is created.
XAIEFAL: INFO: Resource group Static is created.
XAIEFAL: INFO: Resource group Generic is created.
inf> Starting 1 threads...
inf> Throughput=17749.99 fps, total frames = 1000, time=0.0563 seconds
inf> Accuracy= (856/1000)=0.856
root@xilinx-vck190-20222:~/03_vck190_pytorch_atlas-top-tagger# |
```

ATLAS top tagging open data

```
root@xilinx-vck190-20222:~/Vitis-AI/examples/vai_library/samples/classification# ./test_video_classification resnet18_pt 0 -t 8
[ WARN:0] global /usr/src/debug/opencv/4.5.2-r0/git/modules/videoio/src/cap_gstreamer.cpp (1081) open OpenCV | GStreamer warning: C
annot query video position: status=0, value=-1, duration=-1
XAIEFAL: INFO: Resource group Avail is created.
XAIEFAL: INFO: Resource group Static is created.
XAIEFAL: INFO: Resource group Generic is created.
WARNING: Logging before InitGoogleLogging() is written to STDERR
11119 10:18:38.351377 1517 demo.hpp:752] DPU model size=224x224
11119 10:18:38.392418 1517 demo.hpp:752] DPU model size=224x224
11119 10:18:38.433463 1517 demo.hpp:752] DPU model size=224x224
11119 10:18:38.474534 1517 demo.hpp:752] DPU model size=224x224
11119 10:18:38.515609 1517 demo.hpp:752] DPU model size=224x224
11119 10:18:38.556959 1517 demo.hpp:752] DPU model size=224x224
11119 10:18:38.598032 1517 demo.hpp:752] DPU model size=224x224
11119 10:18:38.639214 1517 demo.hpp:752] DPU model size=224x224
```

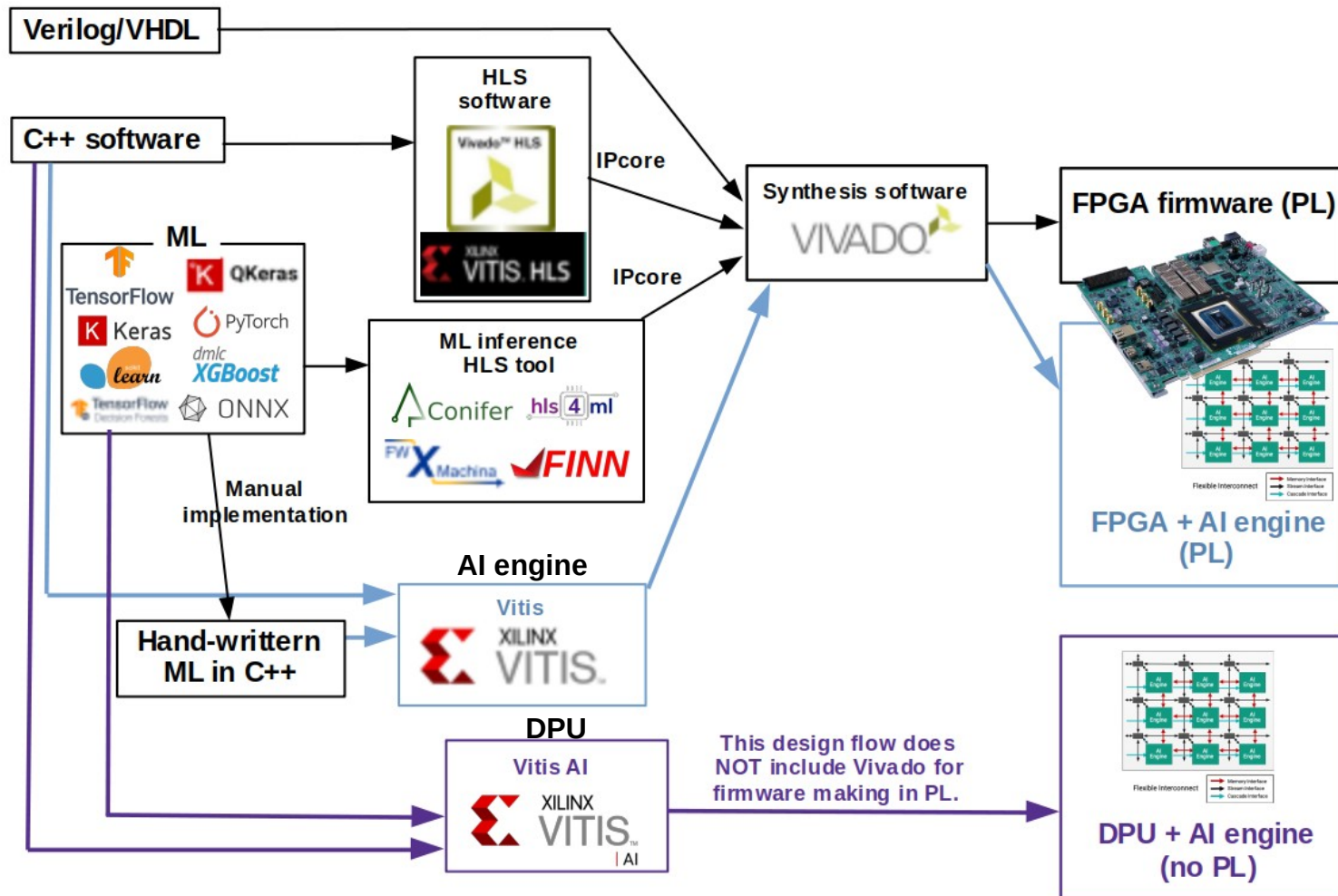
Camera video processing in DPU

FPGA methodology

Not only "what kind of logics to be designed" but also "how to design it". This is a general project regarding the technical knowledge of constructing algorithms in FPGA, including RTL/HDL, HLS, ML inference, and computation engines.

HLS, ML, AI engine: roadmap of FPGA methodology

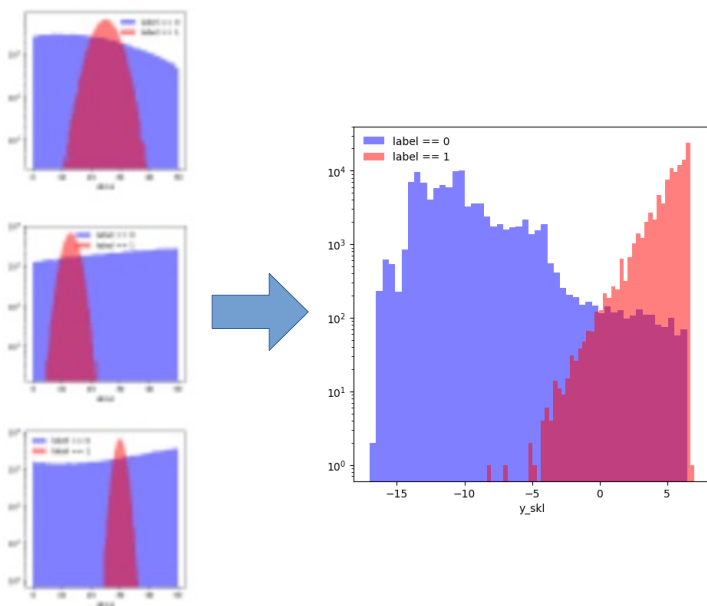
- As a member of KEK E-sys and CEF, we hope to understand the basic utilization on each, and build a database of such technical knowledge, to support our experimental colleagues.
- ~90% of the items is ready with the efforts from young colleagues working with me.



Conifer package



- Conifer: for BDT implementation in FPGA
- Similar design work as hls4ml
- An example of bipolar separation with 3 features:

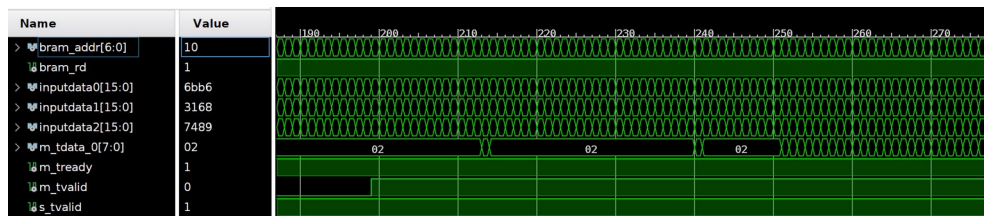
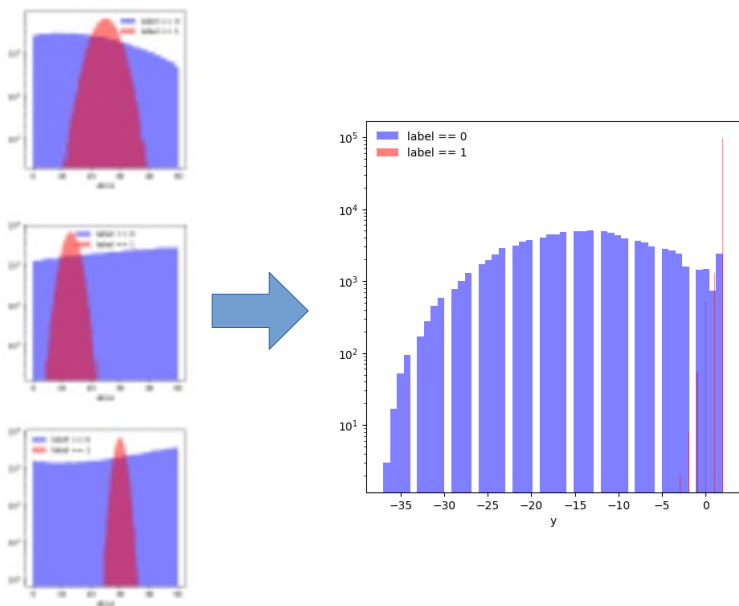
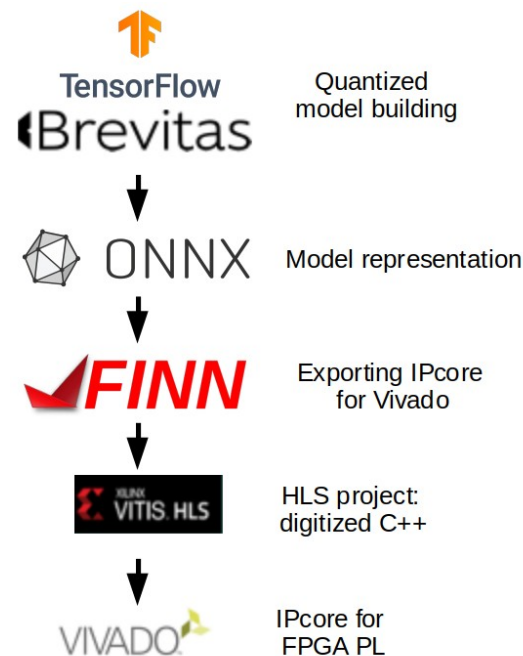


ILA Status: Idle		0000000000000000															
Name	Value	575	576	577	578	579	580	581	582	583	584	585	586	587	588	589	
ap_rst	0																
ap_start	1																
bram_addr[9:0]	007	000	001	002	003	004	005	006	007	008	009	00a	00b	00c	00d		
bram_rd	1																
inputdata2[15:0]	73ba				74e6	7c5d	7357	74b7	73ba	6f14	7761	7489	7639	6ba3	7819		
inputdata1[15:0]	30de				3ad5	41c1	3419	4247	30de	337e	1e93	3168	444c	34d6	339f		
inputdata0[15:0]	617c				63fe	6080	680b	6a7d	63a1	617c	7702	812e	6bb6	7028	75d3		
score_0[15:0]	dfc9																
score_0_ap_vld	0																
ap_idle	0																

Latency is usually smaller than the cases with NN (depends on tree depth)

FINN package

- FINN: ML inference tool developed by AMD Xilinx.
- For the quantization on the NN model, the developers also provide a package "Brevitas" together with pytorch.
- The concept is based on matrix multiplication.
- Data flow is based on AXI4 stream.
- The design flow is different from hls4ml.



Latency is similar to the one from hls4ml
(a very rough check)

Summary for WP 5.4, realtime ML @ FPGA

- The target is to study the implementation of ML in FPGA devices for fast processing and to look for possible implementation in the field of experimental particle physics for TRG, DAQ, or HLT.
- In general, two directions for our study:
 - Hardware device: Versal ACAP
 - System-On-Chip for peripheral components.
 - New technologies in data transmission and computation engines.
 - Major interests in the application of the computation engines in data processing of Trigger/DAQ.
 - Methodology: Algorithm construction techniques
 - Building up a technical knowledge database include HLS, ML inference, and computation engines to support our colleagues.
 - For the same logic design, we can utilize different tools to compare their resource, latency and performance.