# Machine Learning in digital electronics devices for real-time DAQ in HEP

Yun-Tsung Lai

KEK IPNS

*ytlai@post.kek.jp*

Seminar @ IJCLab

23rd Jan., 2026

# Introduction

- Collider Electronics Forum (CEF) at KEK Instrumentation Technology Development Center (ITDC):
  - A platform for joint R&D on the electronics devices in the field of experimental particle physics.
  - Members from KEK E-sys, Belle II, ATLAS, SPADI-A, and EIC.

- Two task forces now:

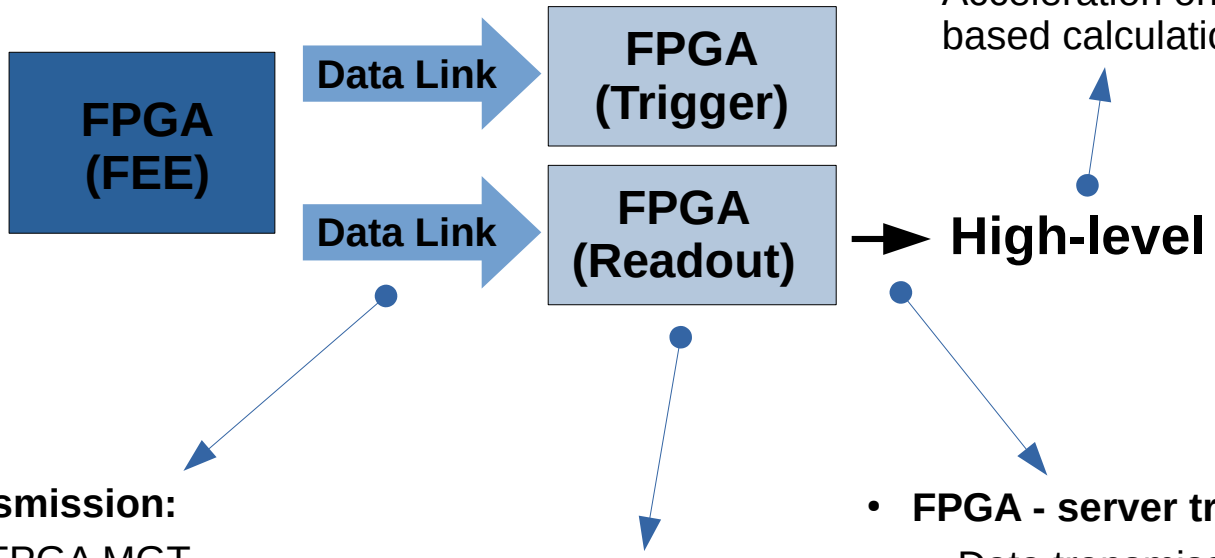
  - **Versal**: Application of high-end FGPA SoC devices in TRG, HLT, or DAQ in HEP.
    - **Large FPGA for backend**.
    - **Complicated ML model** and inference in FPGA.

  - **AI in FEE**: Application of AI/ML in Front-End level of detector system.
    - **Compact ML** in **smaller FPGA or ASIC**
    - Outside of HEP systems

# Application of FPGA in HEP experiments

- **Our target:** Study the latest COTS FPGA devices and their associated new technologies for possible application and upgrade in different aspects of HEP experiments.

- **Hardware acceleration:**
  - Not only CPU, but also GPU and FPGA.
  - Acceleration on software-based calculation.




- **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.

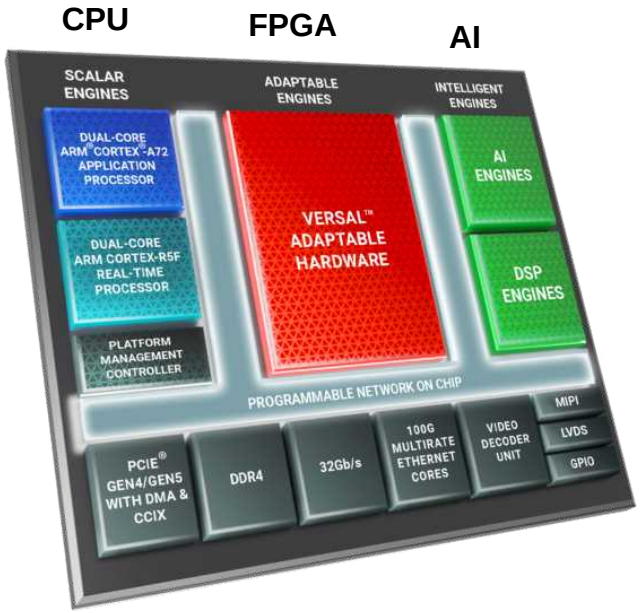
  are critical for the usage in:
  - **TRG**: complicated algorithm implementation.
  - **DAQ**: collect and process large data.

- **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.

# Versal project

- Mainly based on the Xilinx Versal series of ACAP.
- CEF purchased a few evaluation kits with sharing to our collaborators for joint study.
  - 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.

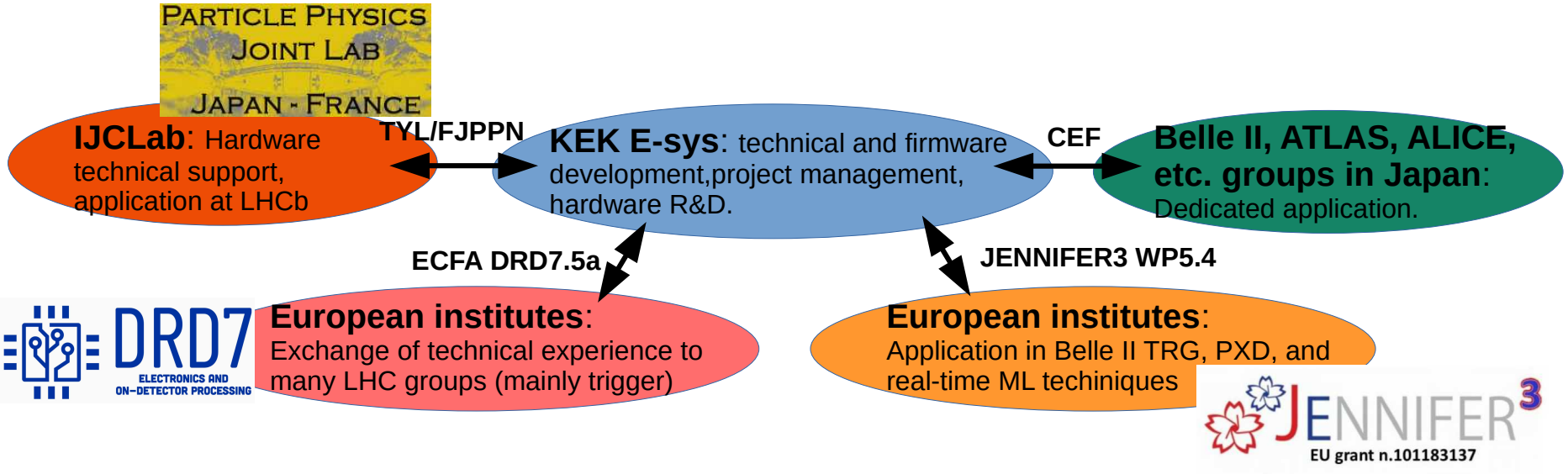




source: Xilinx website

# International networking

- Here is our international networking based on this Versal project.
  - There is also communication with individual institute for specific development.

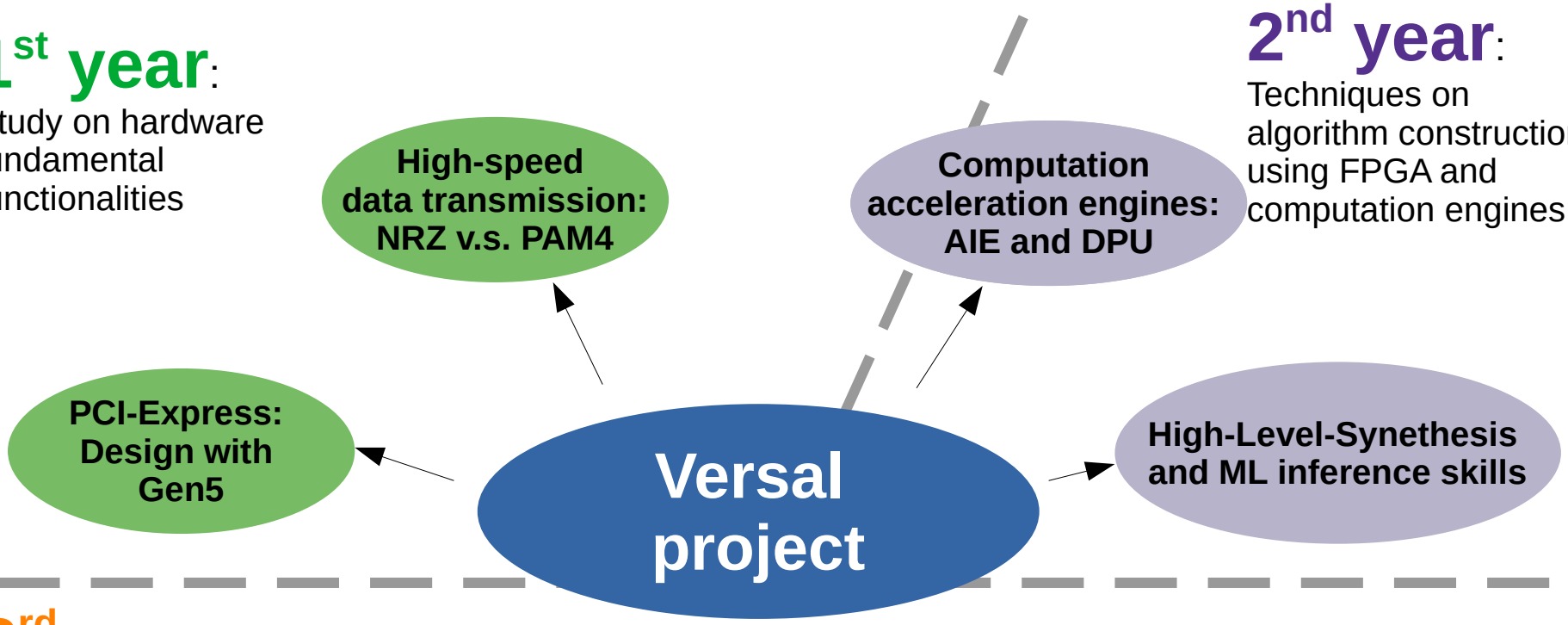
- **ECFA DRD7: 7.5a**
  - TDAQ backend with COTS of heterogeneous computing architecture
  - Utilizing different hardware platforms (FPGA, GPU, CPU) for real-time processing (trigger).
  - Target: Building an open-access, repository-hosted infrastructure for these commonly used tools and algorithms.
    - We will start to construct this repository from the end of 2025!
  - ~20 institutes join 7.5a.
  - I am one of the convener.



# Project overview and working plan

**1st year**:
Study on hardware fundamental functionalities

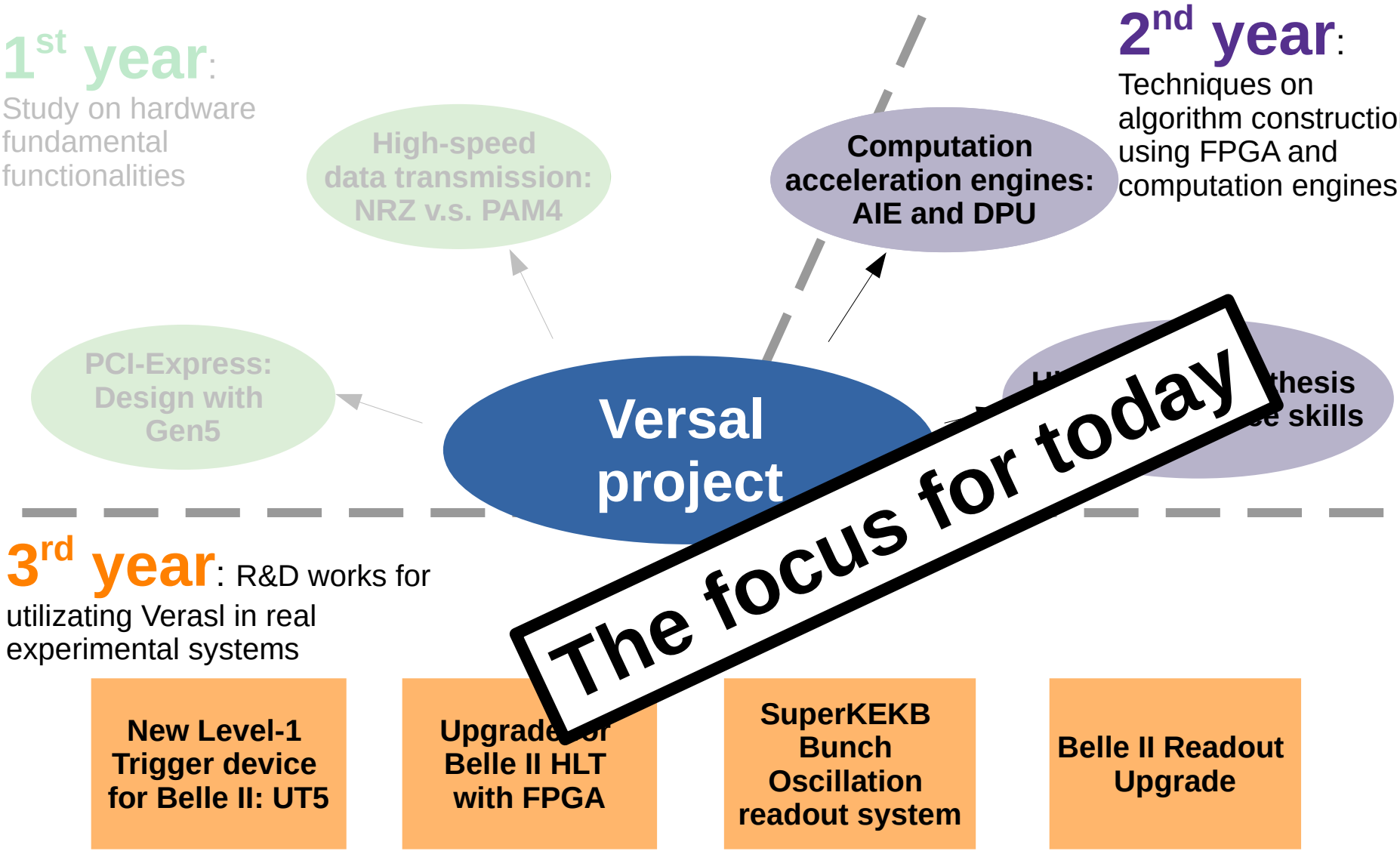
**2nd year**:
Techniques on algorithm construction using FPGA and computation engines




**3rd year**: R&D works for utilizating Verasl in real experimental systems

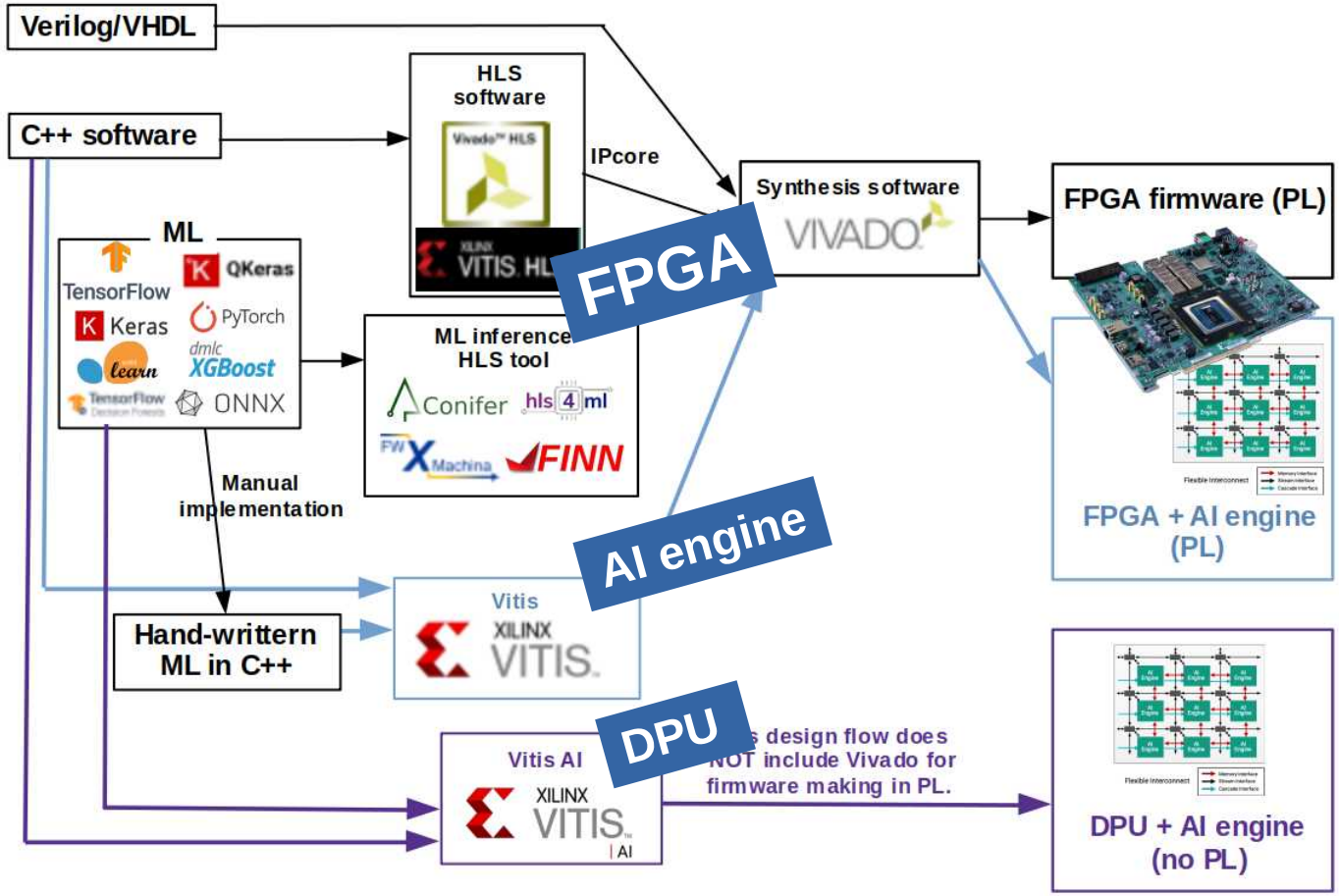
- New Level-1 Trigger device for Belle II: UT5
- Upgrade for Belle II HLT with FPGA
- SuperKEKB Bunch Oscillation readout system
- Belle II Readout Upgrade

# Project overview and working plan



# FPGA methodologies on HLS, ML in FPGA, AIE

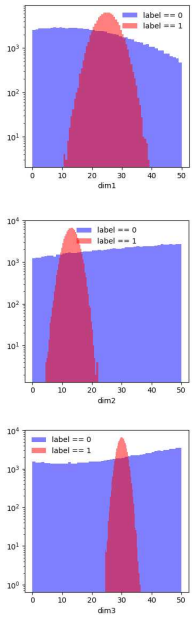
- "Not only what kind of logic to make, but also how to make it."
  - As for CEF, we should not simply work on individual development, but do such kind of fudamental works. Such collection of technical efforts could be helpful to support our experimental collegues.
- We almost finished buliding this technical database.
- We also held a summer school in Aug. 2025 for these techniques.

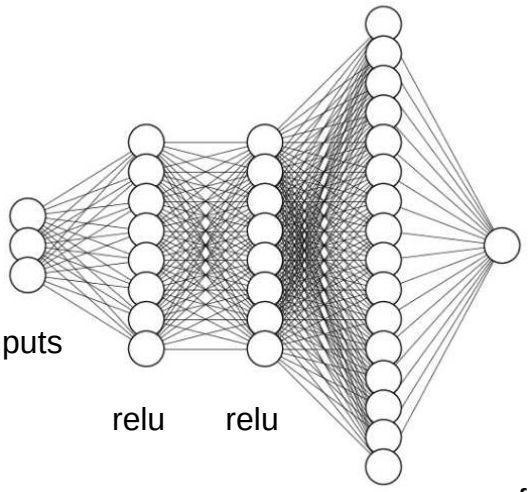


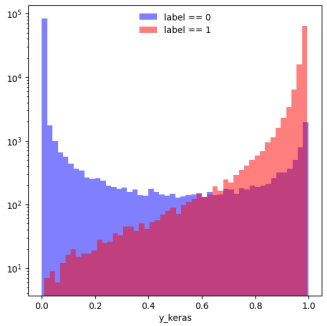
# hls4ml

- hls4ml has been widly utilized in our field already.
  - For TensorFlow and Pytorch

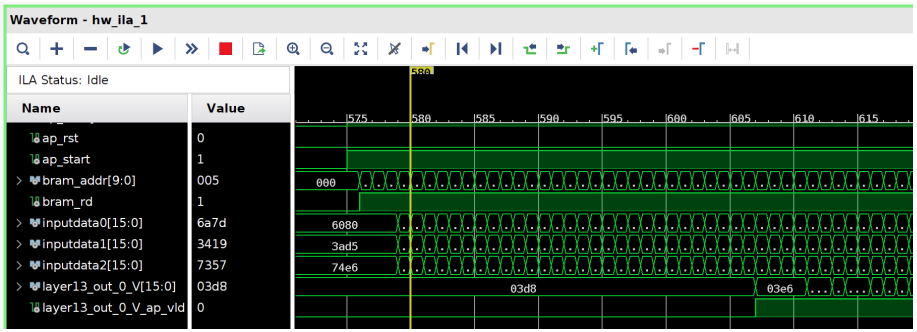
- Just a smple demonstration using Nexys Video card and a bipolar separation NN model:

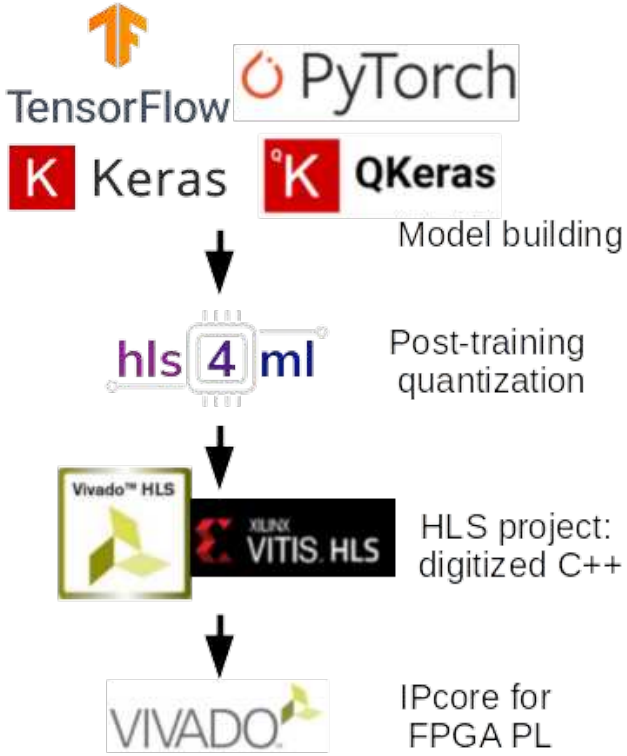



sigmoid
for binary separation



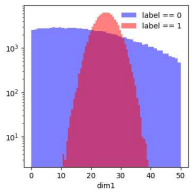



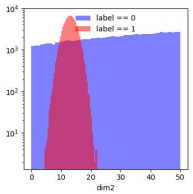

Latency: O(10) clock-cycles

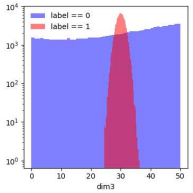
# Conifer

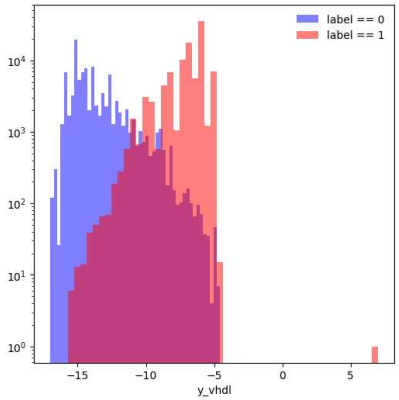
- Conifer: a package for BDT inference in FPGA
  - The same developer group as the one for hls4ml.

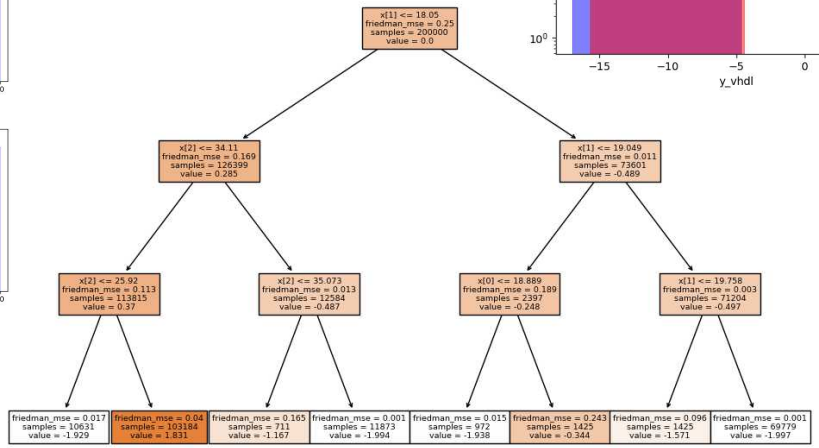
- Compared to NN, BDT is suitable for separation purpose, but not for regression.





Model building


Post-training quantization


HLS project: digitized C++

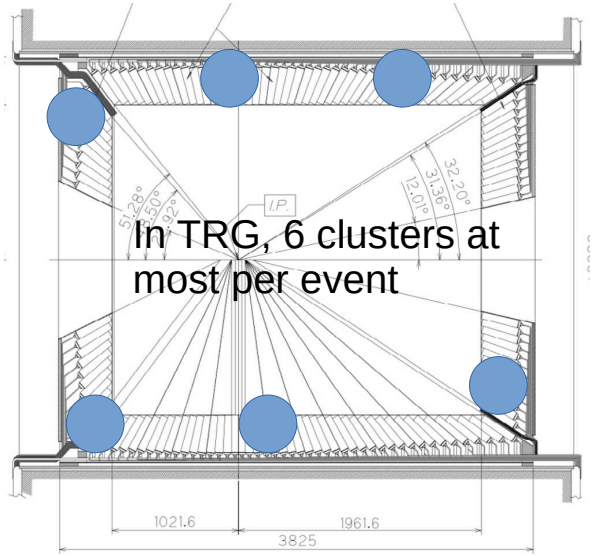

IPcore for FPGA PL

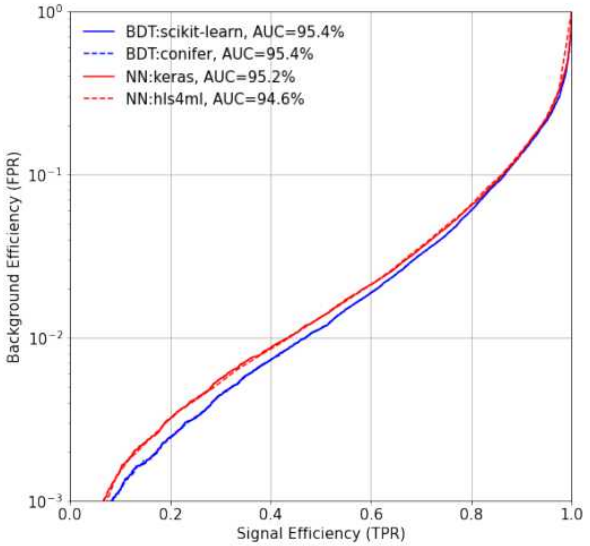
# Belle II τ trigger: NN v.s. BDT

R. Nomaru (Univ. of Tokyo)

- Example: Belle II τ event trigger with calorimeter cluster
  - Input: clusters' position and energy
  - Output: Y/N for a $e^+e^- \rightarrow \tau^+\tau^-$ event
  - Original design is based on NN+hls4ml.

- For an alternative way using BDT+Conifer:
  - BDT can achieve the almost same performance.
  - Smaller LUT usage, and 0 DSP usage.



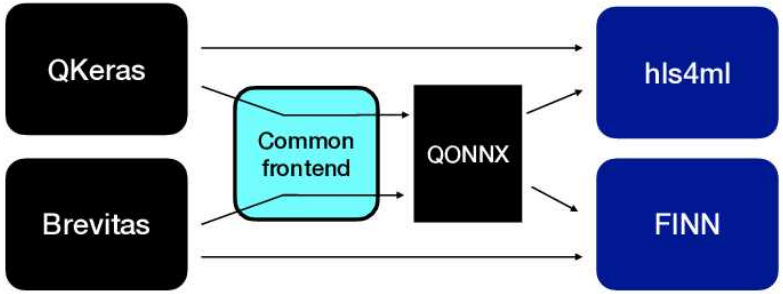

YongHeon Ahn (Korea Univ.)

| Resources | BDT with Conifer | NN with Keras |
|---|---|---|
| Latency | 12 cycles | 14 cycles |
| Initiation Interval | 1 cycle | 1 cycle |
| LUT | 22,504 | 28,480 |
| Flip-Flop | 11,629 | 10,632 |
| DSP | 0 | 228 |

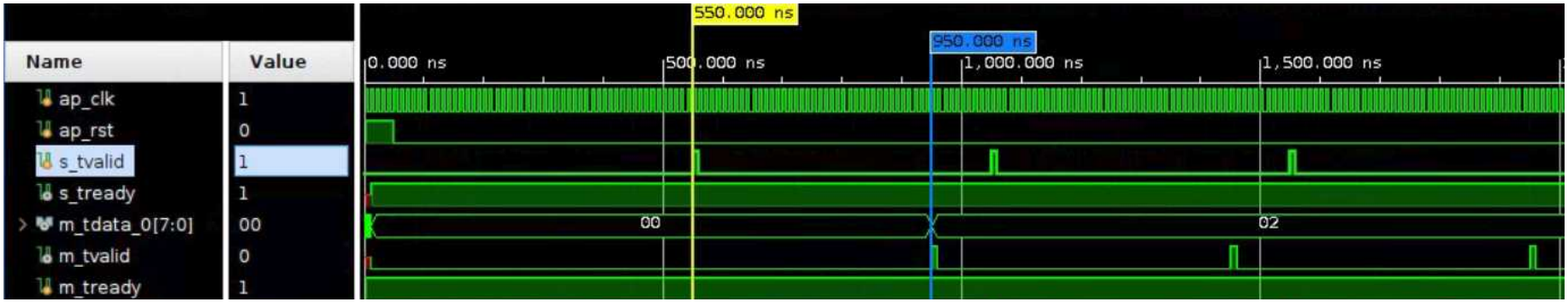
# FINN by AMD Xilinx

- Under development by AMD Xilinx.
- The core concept is matrix multiplication.
- Quantization based on Pytorch + Brevitas.
- Model representation by ONNX/QONNX.

- Material is ready.
- Will also use it for our ongoing developments.

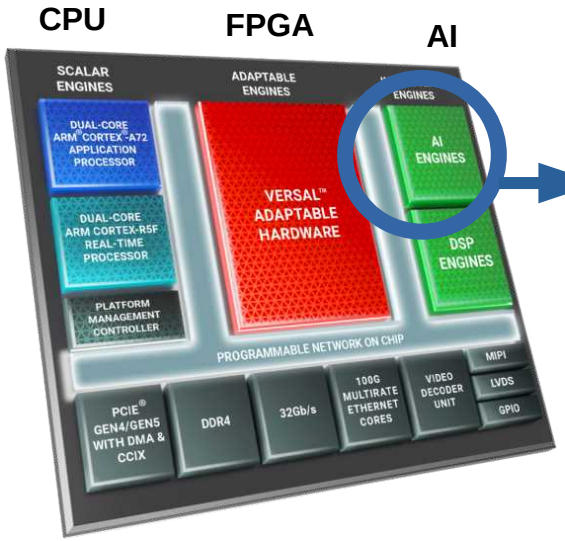



source: 10.48550/arXiv.2206.11791

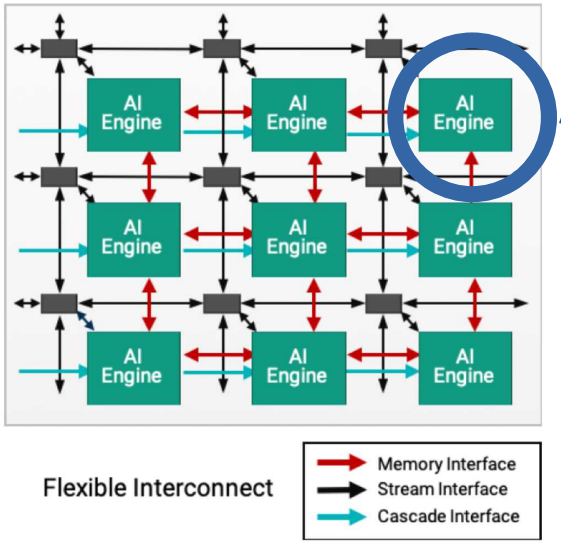



# AI engine of Xilinx Versal ACAP

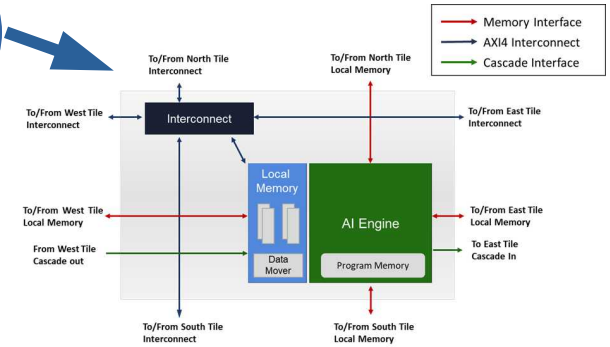
## Versal ACAP



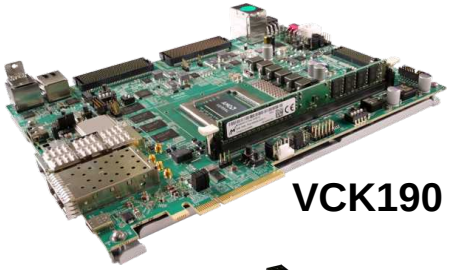

## Versal AI engine




## An AI engine "tile"




- Computation acceleration engine of Versal ACAP.
- Embedded processor of FPGA.
  - High bandwith between FPGA and AI engine.
  - Outside of FPGA fabric.
- **C programmable**.
  - High precision.
  - No quantization loss on ML.
- **Low latency**.



**VCK190 with AIE**

**VEK280 with AIE-ML**

# Know-how of ML in AIE

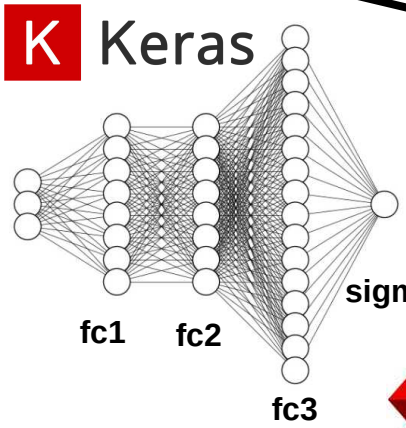
- Here I use this self-defined NN model for demonstration.
  - The model is built by Keras. 3 inputs → 3 hidden layers (8,8,16) → 1 output with sigmoid.

- After the model is built, I just obtained the math formula of the model, and write the codes for AI engine in Vitis.
  - **Everything for AI engine is in C++ and single-precision floating point.**
  - **No quantization loss.**

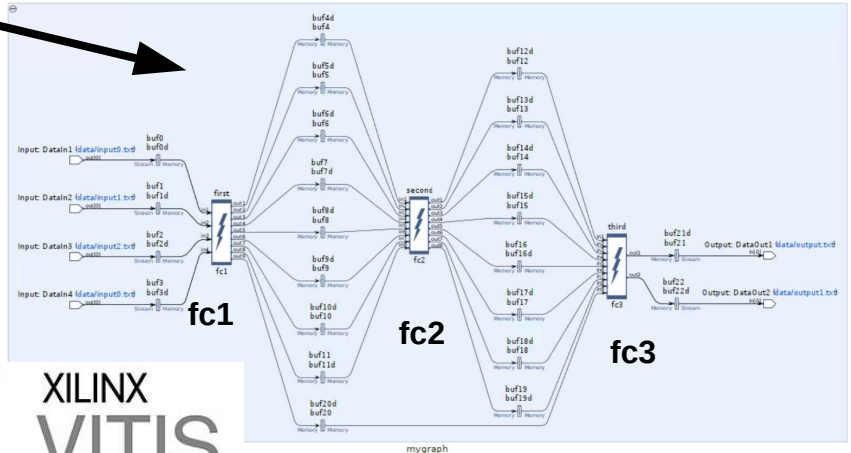
- **Latency: 3.4 μs.**

TensorFlow

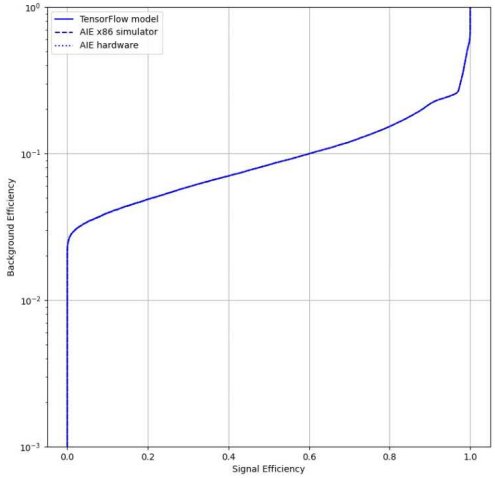
Keras




Exact math form
in C++ in float

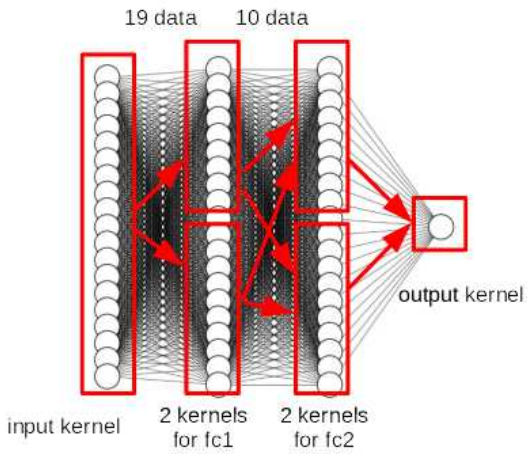


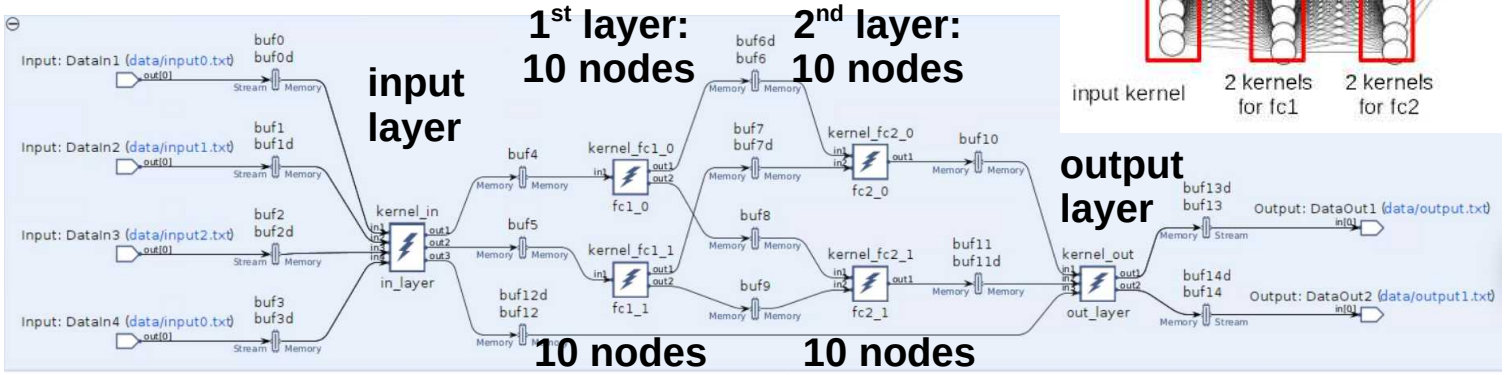

XILINX VITIS



# ML in AIE: Belle II τ NN trigger

- Use the same NN model design mentioned in previous pages
- Implement the mathmatic formula of the Keras model in AIE.
  - No quantization
- 19,20,20,1
- Latency: 4.8 μs




# FPGA and AI engine algorithm implementations

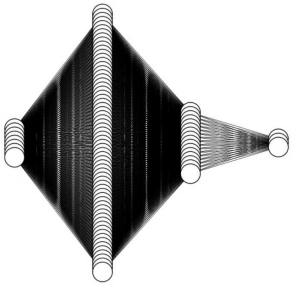
- The original designs are based on HLS inference tools for FPGA implementation.
  Then, plain C++ is written in Vitis for Versal AI engine.

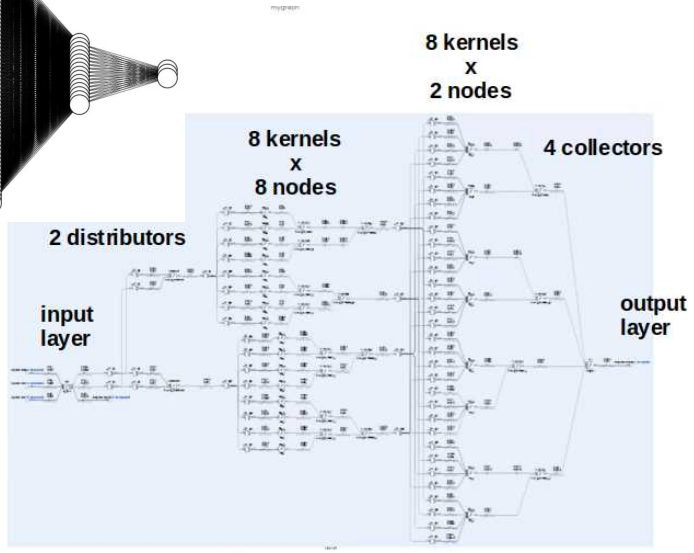
## NN for KLong-Muon chamber trigger

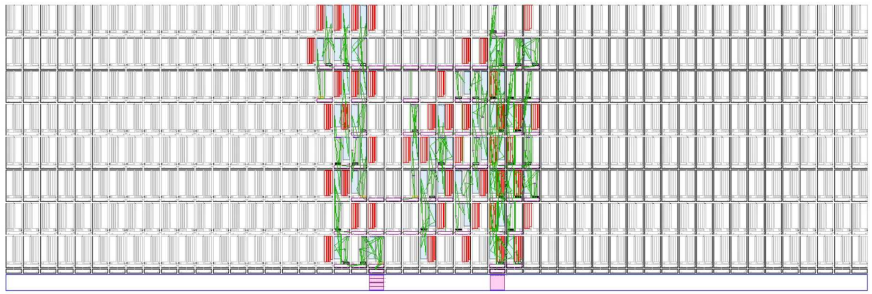
- Neurons: 8,64,16,3

A. Little (Univ. of Sydney)

Keras hls4ml





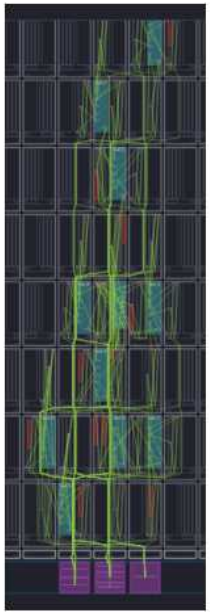
## BDT for tau trigger in L1

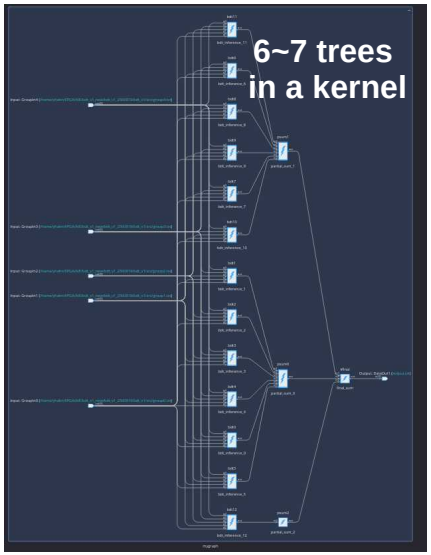
- N of estimator = 90
- Depth = 3

Y. Ahn (Korea Univ.)

scikit learn

Conifer



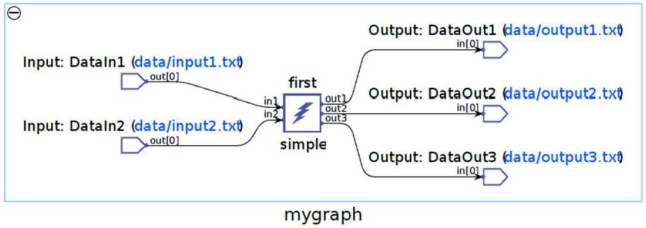


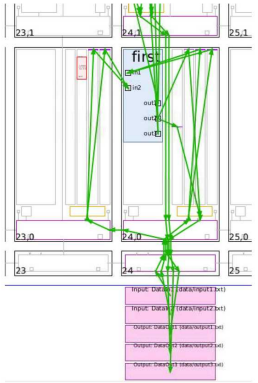

## Linear fitter

J. Song (Korea Univ.)

XILINX VITIS HLS

- Based on linear algebra
- C++ in Vitis HLS

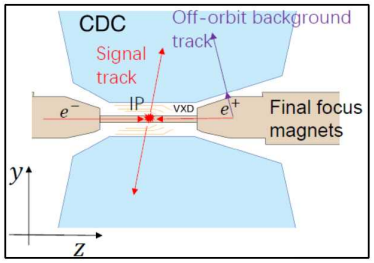


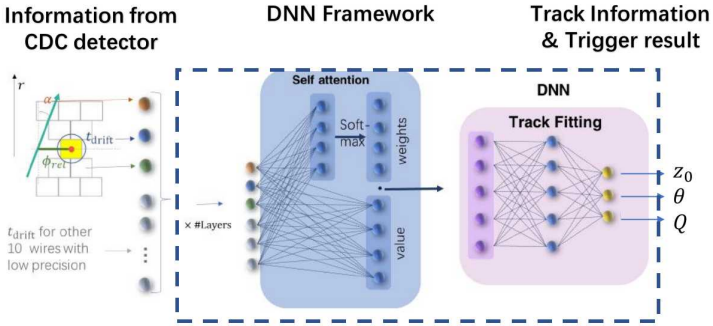



# More plans for Versal FPGA and AI engine

- Belle II L1 trigger: Deep-Learning NN for 3D tracking (z-trigger).
  - Original design based on Pytorch and hls4ml.

PyTorch hls4ml



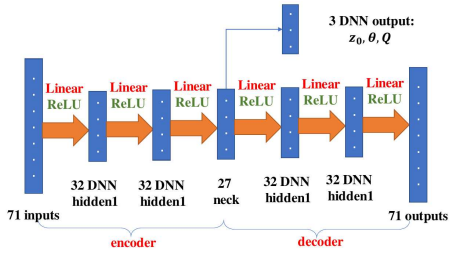



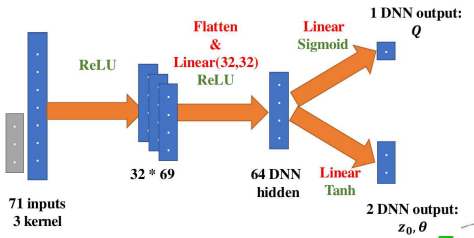

**Original design: Y. Liu (SOKENDAI)**

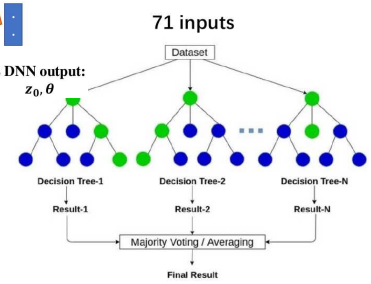
- Ongoing development from the present DNN design with other ML options:

**Y. Yang (Fudan Univ., KEK)**

| Model | AUC | Precision | z0 resolution |
|---|---|---|---|
| Old DNN | 0.979 | 0.8629 | 0.072 |
| My DNN | 0.970 | 0.8451 | 0.113 |
| My CNN | 0.971 | 0.8755 | 0.104 |
| Auto encoder | 0.969 | 0.8641 | 0.117 |
| Random forest | 0.959 | 0.8472 | 0.095 |
| Gaussian Processing | 0.853 | 0.6466 | 0.162 |
| Support Vector Machine | 0.918 | 0.7825 | 0.190 |



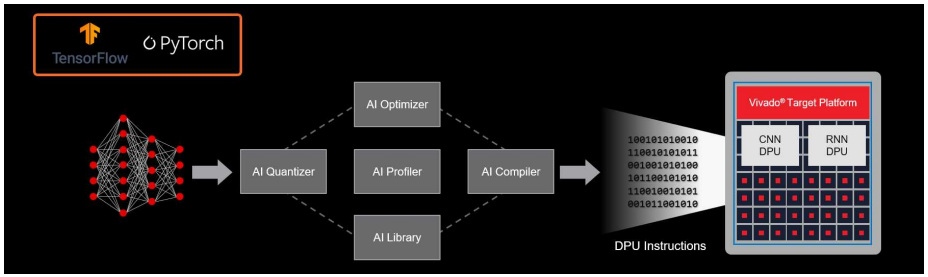

Classifier for Q
regressors for $z_0$ & $\theta$

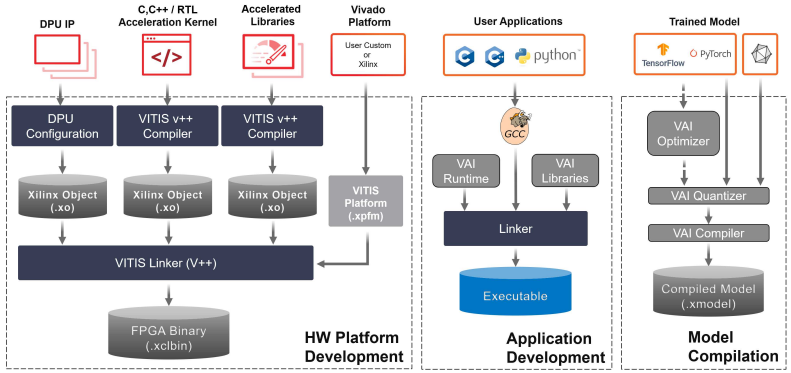
→ Belle II UT4 → UT5 (Versal FPGA) → Versal AI engine

# Versal DPU

- **DPU**: Deep Learning Processing Unit
  - Configurable computation engine dedicated to convolutional neural networks.

- DPU takes leverage of the FPGA resource, while the artificial networks inference **does not require touching FPGA PL**.
  - An IPcore of FPGA
  - Network model building by Pytorch, and quantization by Vitis-AI software.
    - Many pytorch models are supported (not yet for GNN).
  - During utilization, the system is operating like a small OS.
    - Replacement of the network model does not require FPGA reprogramming.
    - User operates everything in **command line with ssh and scp**.
  - **Hardware acceleration for high-level application**.

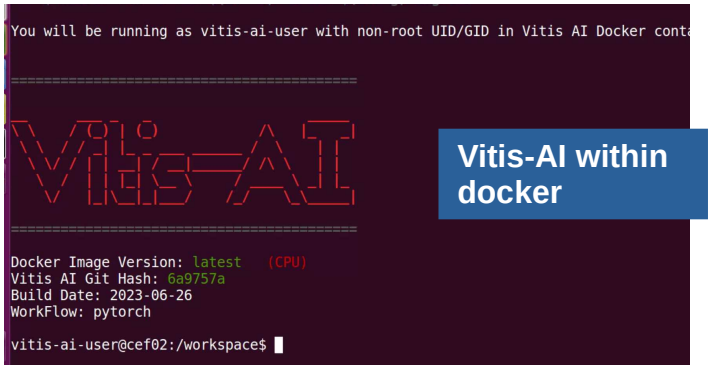


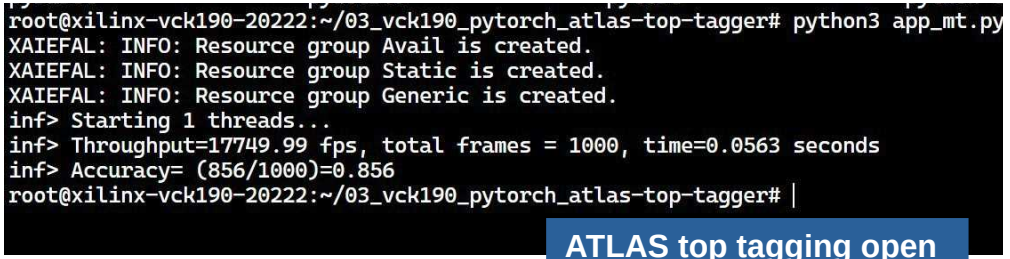
# Versal DPU, network building and running

Chaowaroj "Max" Wanotayaroj (KEK ATLAS)

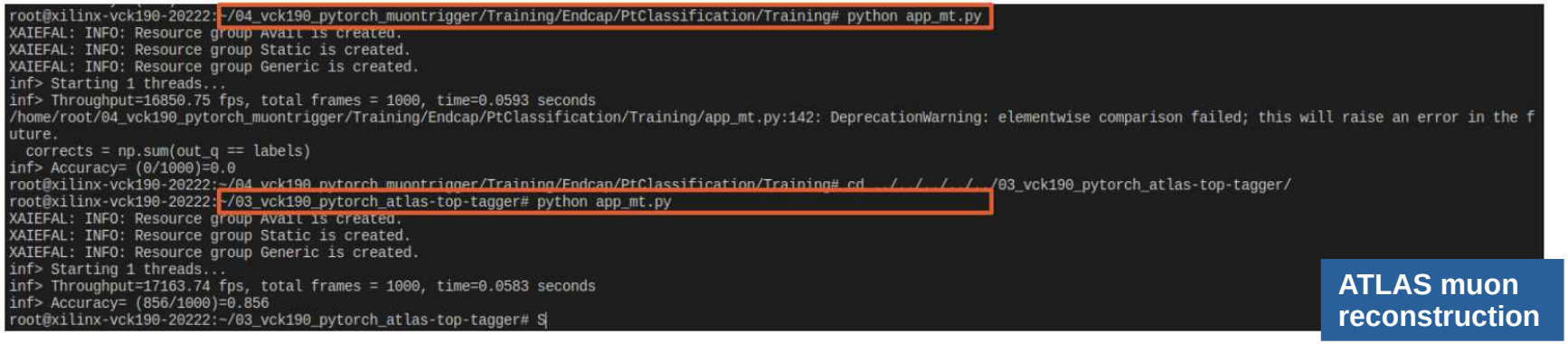
- Pytorch for NN model building.
- Quantization on the model is performed using "VItis-AI" tool from Xillinx inside docker.
  - GPU is also supported.
- After the model is quantized by Vitis-AI, we simply copy the model and data files to the DPU (using scp), then execute the jobs inside DPU.




Vitis-AI within docker




ATLAS top tagging open data inside Versal DPU

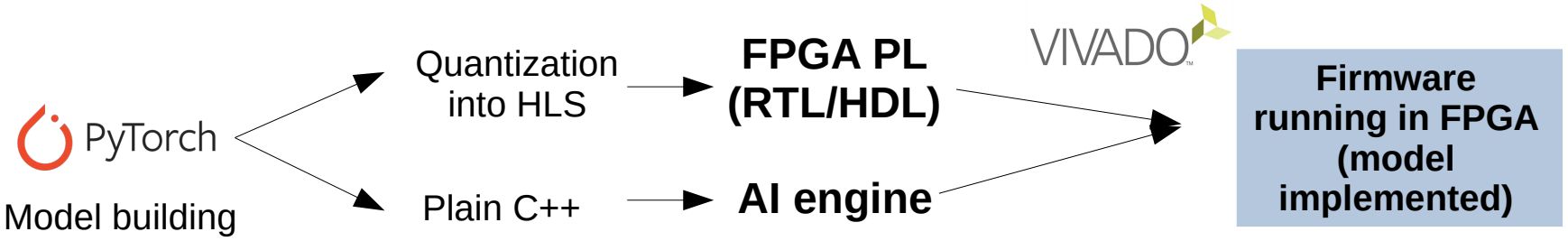


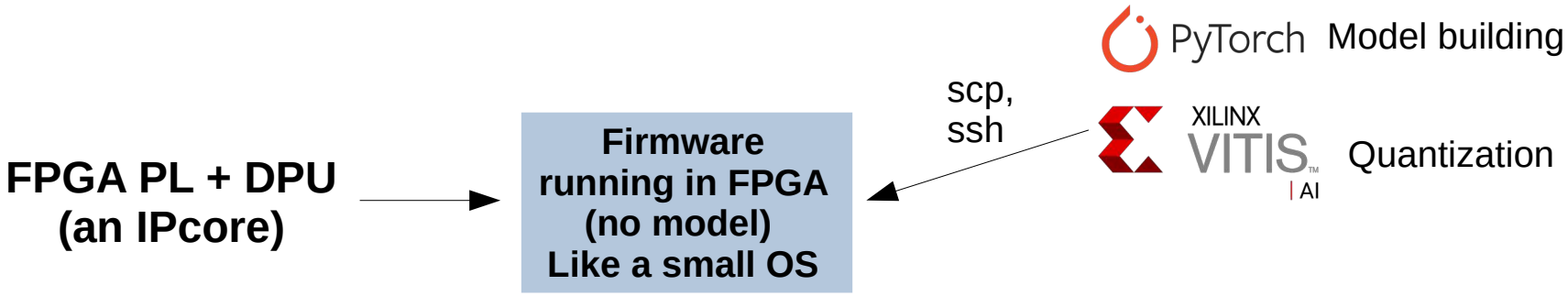

ATLAS muon reconstruction

- By default, connection to DPU is based on ethernet (ssh, scp).
  - Now we are studying data transfer using **PCIe**, which will be helpful for real utilization.

# DPU v.s. AI engine v.s. FPGA

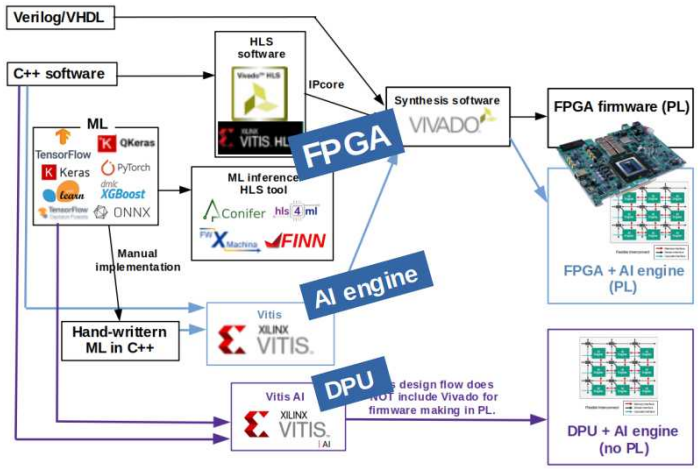
- Inference in **FPGA PL or AI engine**:
  - A fixed network has been implemented inside firmware.



# Summer school on HLS, ML in FPGA, AIE

- Based on the technical database we collected, we held a summer school in 2025.
- In total 25 people from Japan and other countries.
  - Also people from different time zone!
- Content: HLS, hls4ml, Conifer, FINN, Versal AI engine
- Device: Nexys4 boards from Digilent
- We plan to have it once per year.




# Belle II Level-1 Trigger board upgrade (UT5)

**Belle II UT3**

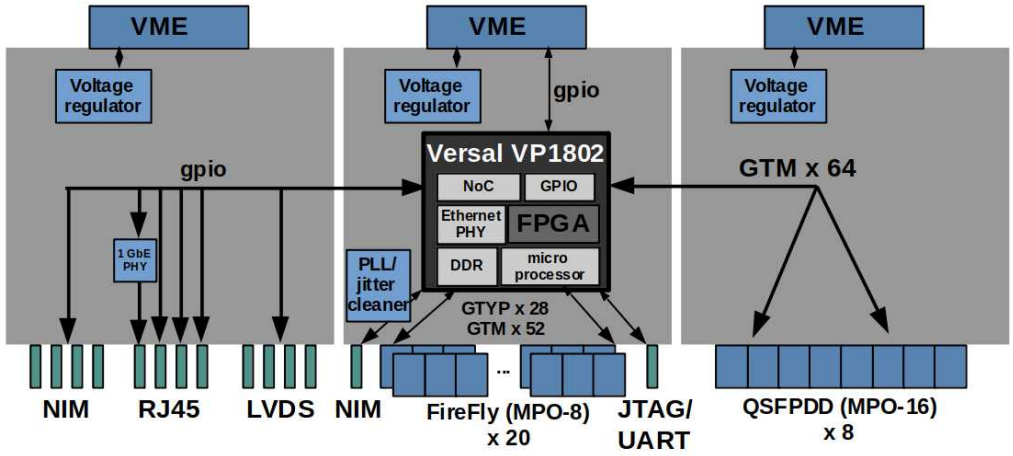
Xilinx Virtex-6
xc6vhx380t, xc6vhx565t
11.2 Gbps with 64B/66B

**Belle II UT4**

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

## New design: UT5

**Preliminary block diagram**



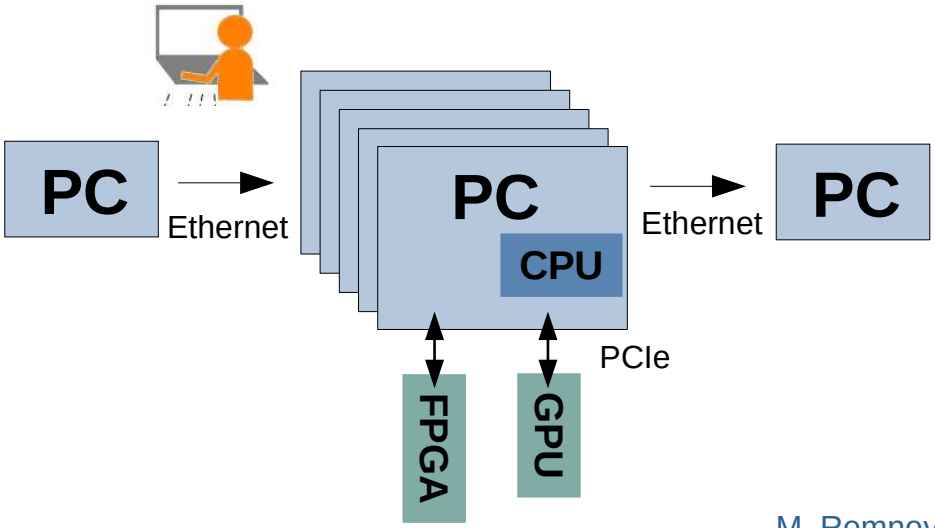

- UT5 based on Versal is going to be our first hardware R&D target:
  - High-speed link with PAM-4.
  - Processing System (PS) of Versal
- Prabably no AI engine in UT5
  - But we are still open for the potential for UT6
- **Prototyping in 2026**

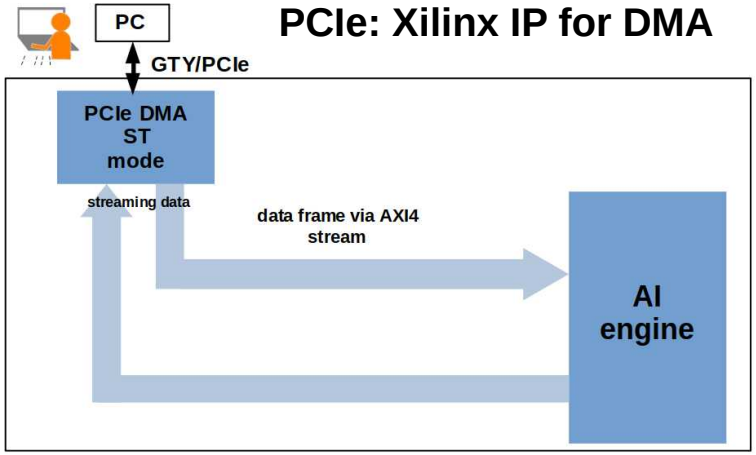
# Versal FPGA or AI engine for HLT

- Other than CPU for HLT: GPU or FPGA.
  - In such case, PC is the host: "Hardware acceleration"

- How about Versal for HLT?
  - We need **PC-FPGA communication**, and **expertee of integraton in FPGA PL**.
  - We tested the framework with Ethernet data link and PCIe of Versal for demonbstration.
  - Integration of the framework in real system and algorithms are under development.

- Now the framework is under development, and we will start to integrate some of the Belle II HLT algorithms in Versal AI engine and GPU.

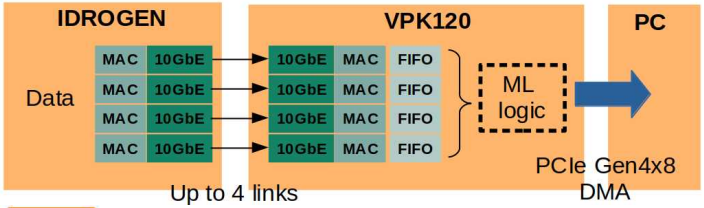



M. Remnev (KEK)

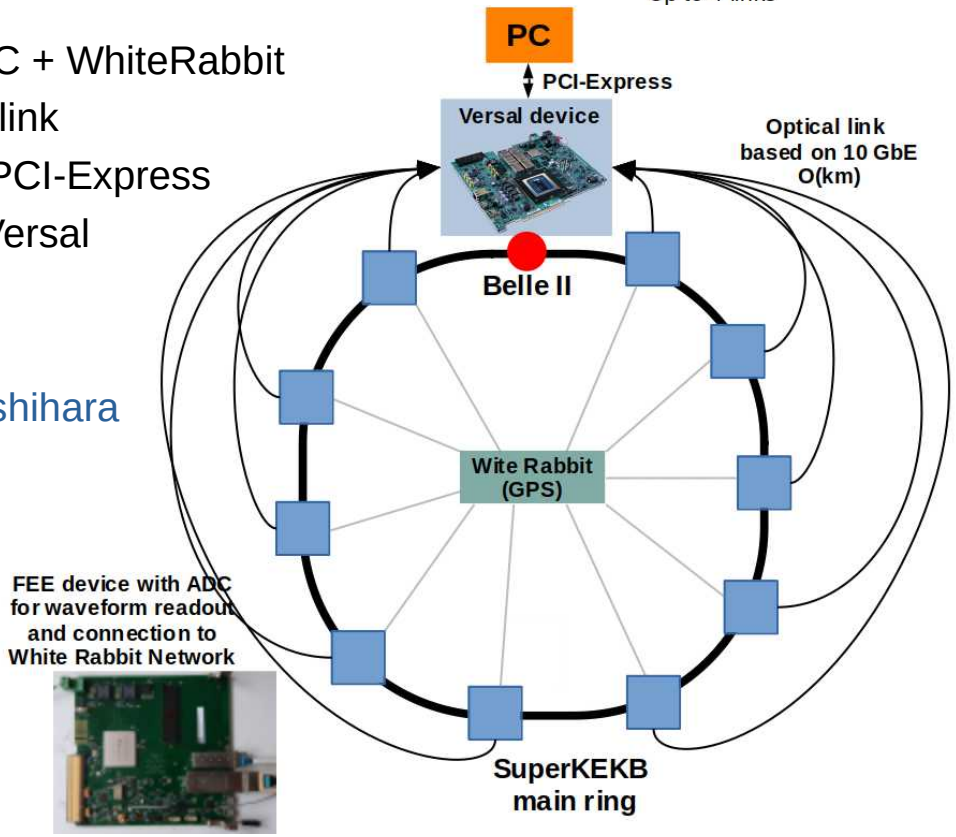
# SuperKEKB Bunch Oscialltion Readout system

- Motivation: To handle the sudden beam loss problem in SuperKEKB, we plan to prepare a system to readout the bunch waveform of oscillation
  - Final target: real-time prediction on the sudden beam loss using FPGA readout system.
    - Protection on the inner detectors of Belle II.
  - Feature study for sudden beam loss issue.




- System:
  - FEE: IDROGEN + ADC + WhiteRabbit
  - Long-distance optical link
  - Readout: Versal with PCI-Express
    - ML-based logic in Versal

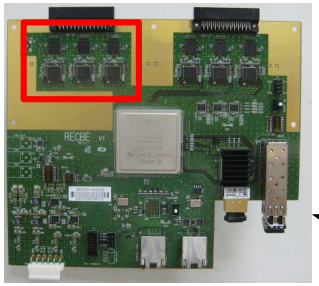
- Collaborators:
  - Univ. of Hawaii: K. Yoshihara
  - KEK ACCL
  - KEK E-sys/CEF
  - IJCLab.



# ML in real-time processing: Trigger or FEE?

**A cut-view of Belle II CDC**:
~14000 sense wires
~300 FEE

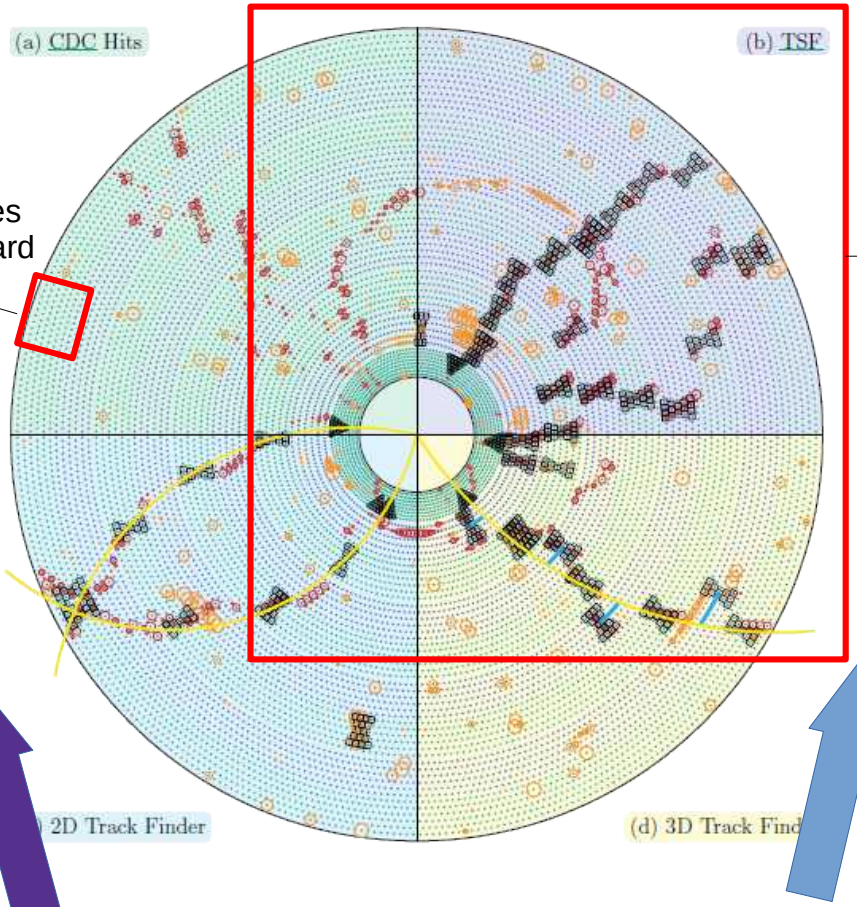
ASIC: Application Specific Integrated Circuit for Analog-To-Digital purpose



**CDC FEE**
**Xilinx Virtex-5**
Collect the 48 channels' information (ADC, TDC) to downstream.

FEE FPGA is usually not so strong, since the logic is not so complicated, and need to consider irradiation effect on electronics.

48 wires per board




Almost the entire detector

**Belle II UT4**
**UltraScale**

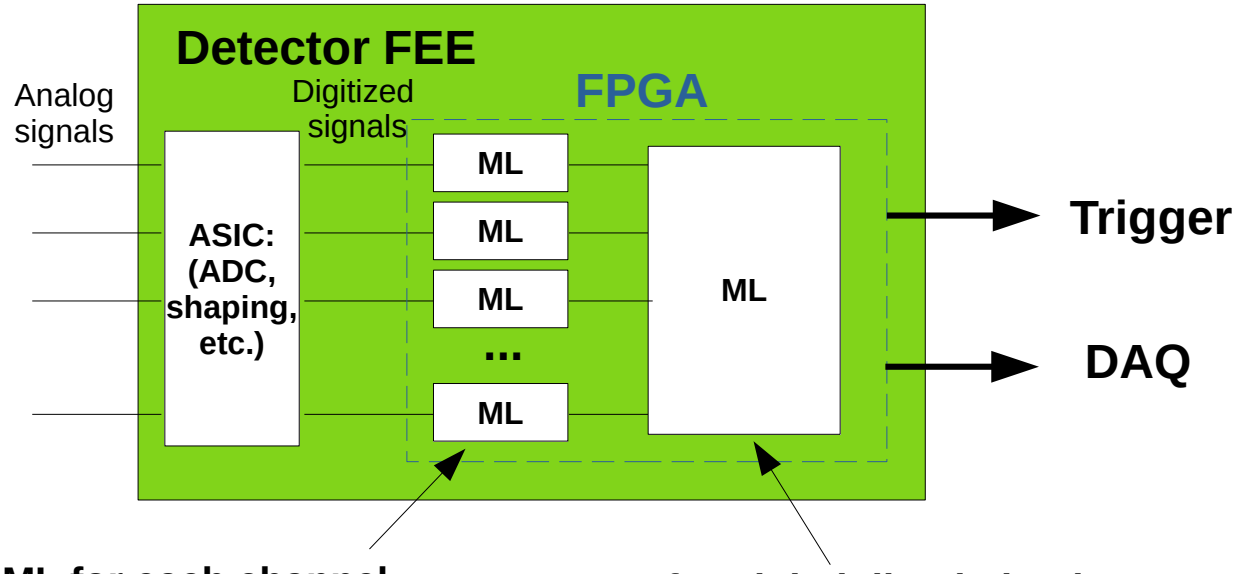
In TRG, FPGA is strong in order to process large date with complicated algorithm: tracking, clustering, topology, etc.

**But how about here?**

**Here, we already knew that lots of thing to play with ML.**

# AI in FEE

- In this project for **AI in FEE**, we are looking for new concept, approach, and application for real-time fast AI/ML other than L1 trigger backend system.
  - Detector FEE (smaller FPGA) is quite different from trigger backend (strong FPGA).
  - **Digitized waveform channel-by-channel**.
  - Building a **compact ML model** compared to back-end.
  - Inference in not only **FPGA**, but also **ASIC, eFPGA**.
  - Possible application other than particle/nuclear physics.

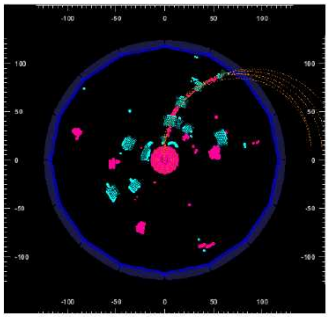


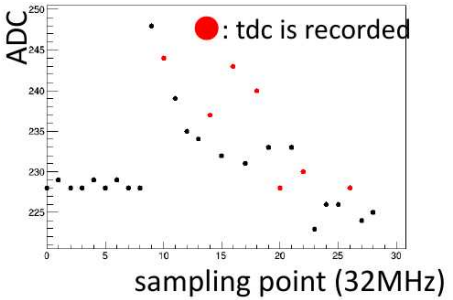

**ML for each channel:**
Shaping, TDC, energy, signal-noise separation, pile-up, hot channel, etc.

**ML for global discrimination:**
For some kinds of background/noise which have coinsidence between channels. e.g. crosstalk noise.

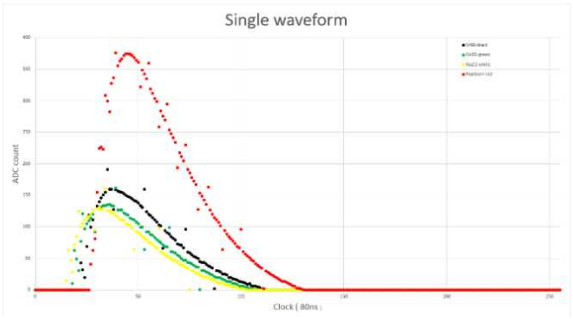
# AI in FEE: developments

- **Belle II**: Now K. Yoshihara (Univ. of Hawaii) and I are trying to organize the possible research plans in Belle II into potential general upgrade plans.
  - CDC: ML-based crosstalk noise reduction
  - TOP: feature extraction (time/charge) in backend
  - ECL: hadron and e/γ separation
  - KLM: p.e. counting for improved time resolution



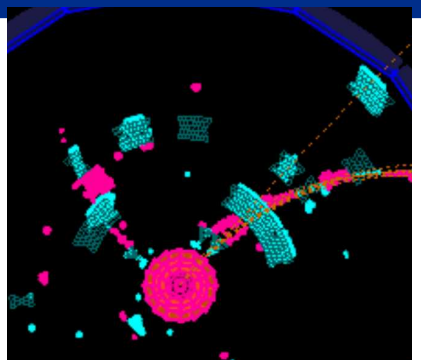



- **Neutron detector** pulse shape discrimination for real-time n/γ separation (Tohoku Univ.)

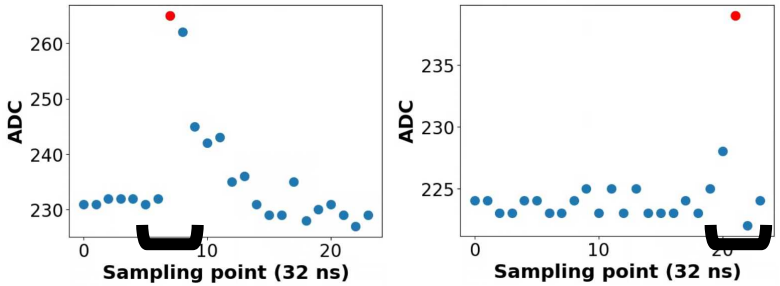



- **Silicon detector** pulse shape discrimination for PID in experimental nuclear physics (Y. Yang, Fudan Univ., KEK)
- **ADC waveform feature extraction** for streaming readout in experimental nuclear physics (SPADI-A)
- **ASIC/eFPGA** development
- **Quantum Error Correction** with small latency in FPGA/GPU/ASIC based on ML (NTHU)

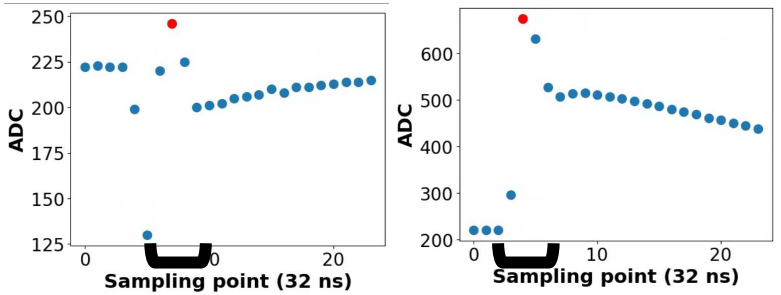
# Belle II CDC cross-talk noise reduction



- In Belle II CDC, we have been suffering from the cross-talk noise in the FEE:
  - Bunch of wire hits happen in nearby region.
  - Causing a large fraction of fake track trigger.
  - Examples of the ADC waveform:

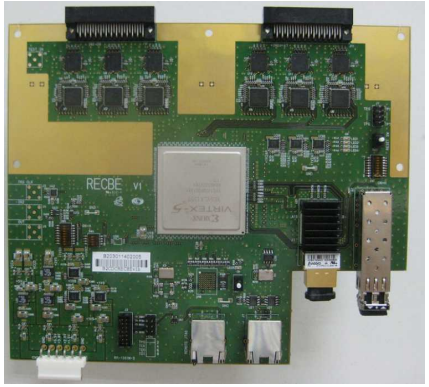



**ADC waveform of signal (from charged track)**

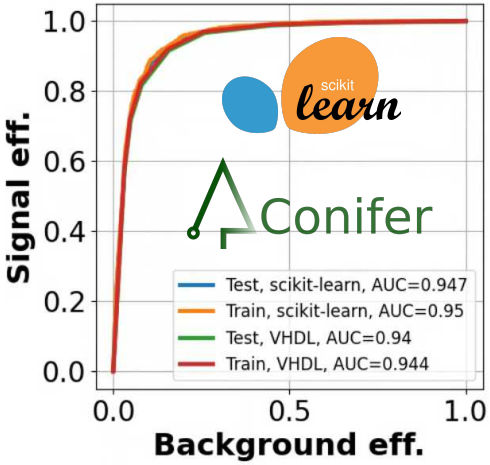



**ADC waveform of cross-talk noise**

- We built BDT model and implemented the modules in Xilinx Virtex-5 FPGA
  - Channel-by-channel instances
  - Small latency and resource usage

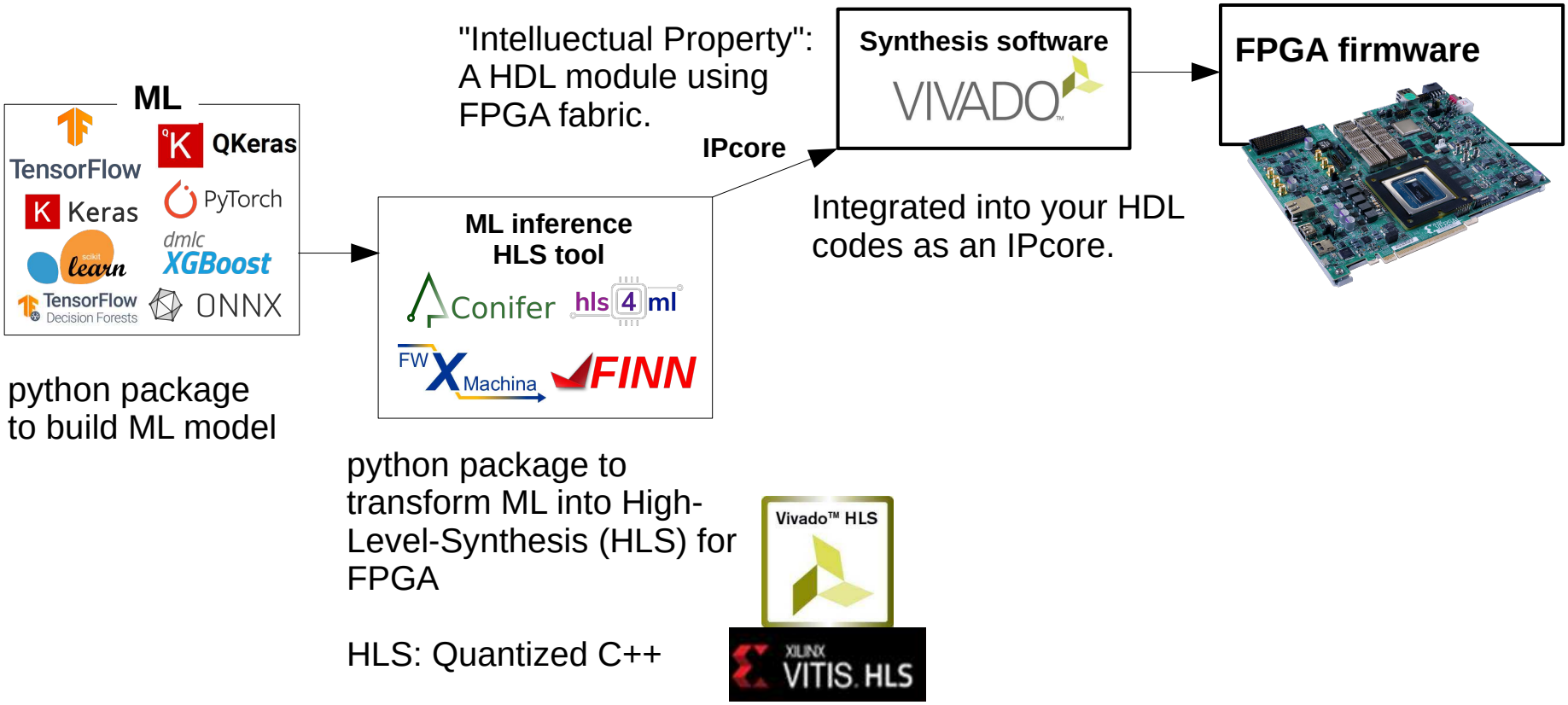


**CDC FEE board featured with Xilinx Virtex-5 FPGA**



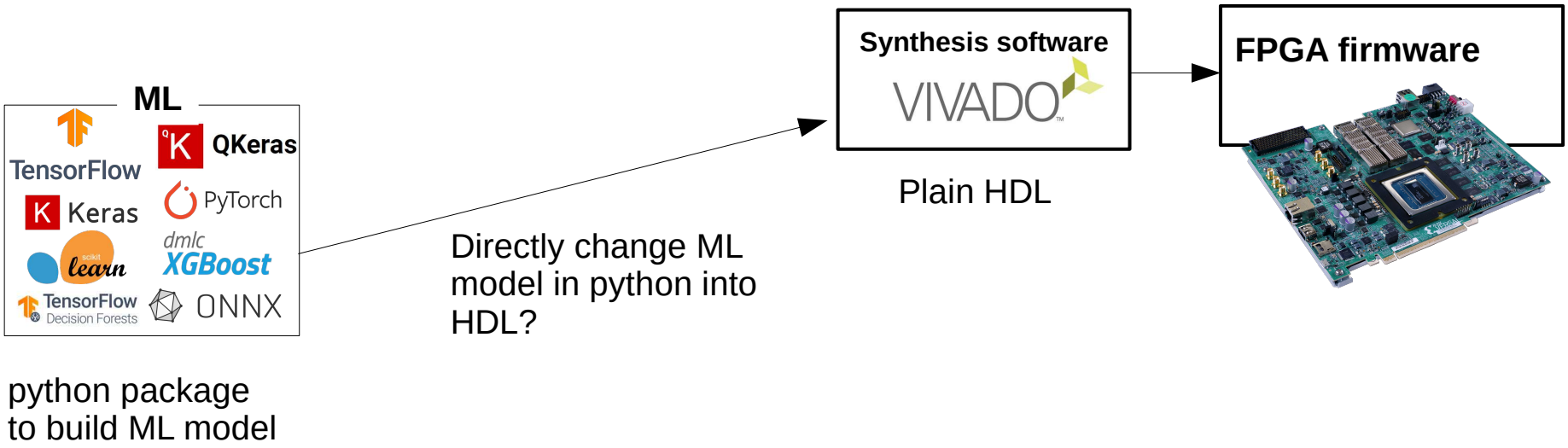
# ML in FPGA based on HLS

- ML inference at FPGA based on HLS has been widely utilized in the filed of experimental HEP.
- Quantized C++ → HLS → IPcore in FPGA
- A kind of black box with FPGA fabric involved: DSP, memory, etc.
  - Not exactly a pure HDL module.



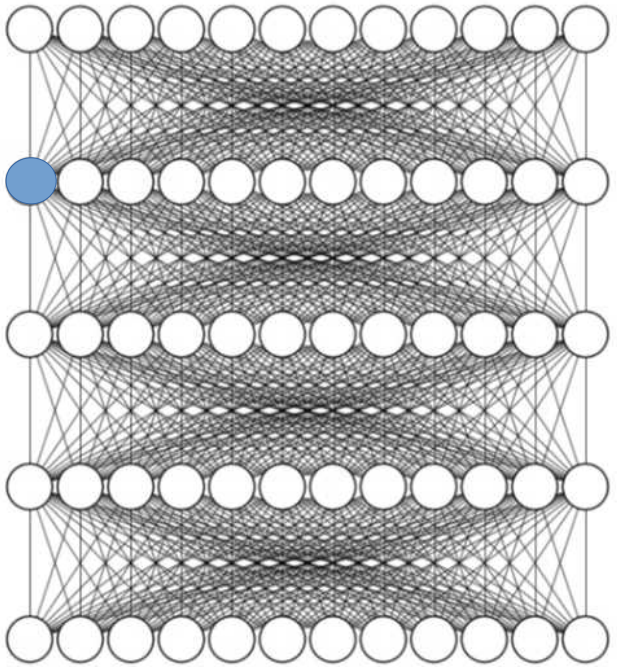
# ML in FPGA based on HDL

- Can we directly represent a ML model in **plain HDL**?
  - Yes, if we understand the math of the ML model first.



# HDL-based NN

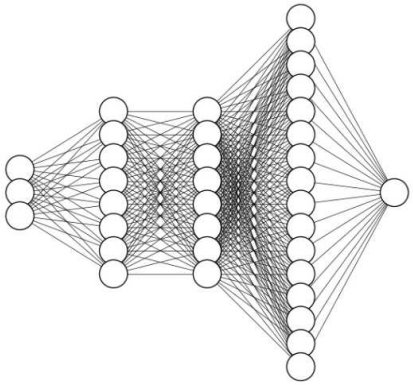
- In each neuron, a basic form is:
  - output = relu( $\Sigma_i(w_i \times input_i) + b$ )
  - "a x b + c" is easy for HDL.
    - For multiplication, it can be chosen to use DSP or LUT.
  - relu is also easy for HDL.



- For the activation function, exponential or the other special functions are difficult to be implemented by using HDL only.
  - Self-made look-up table using LUT/BRAM is needed. The size depends on fixed-point's precision.
  - Linear output is enough for most of the purpose (separation or regression).

# Comparison to hls4ml

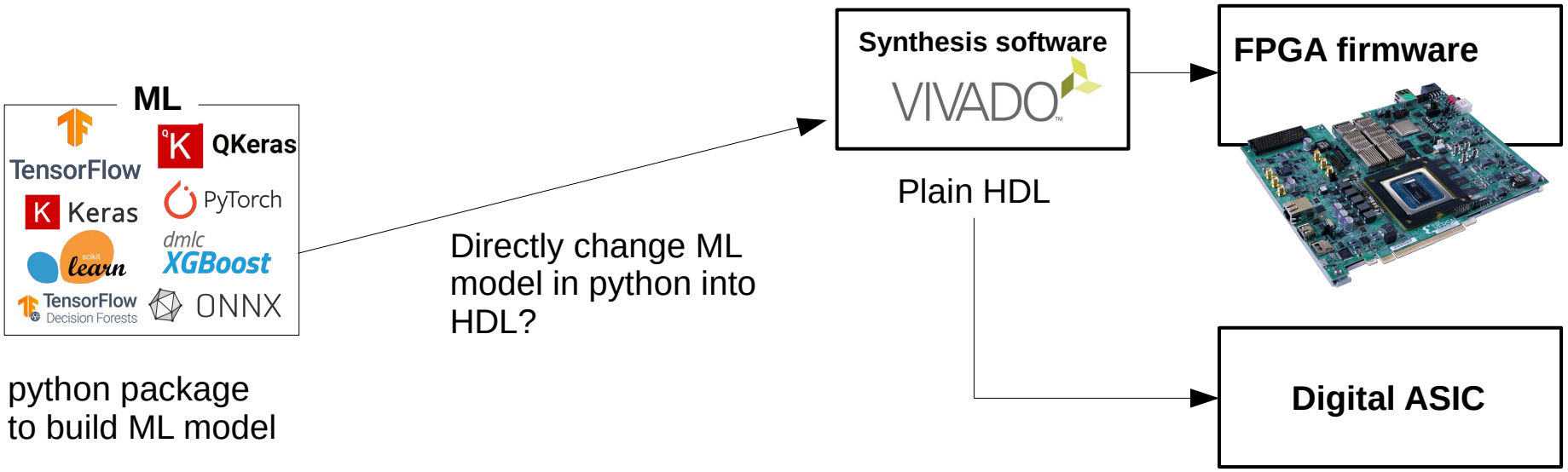
- We implement the same NN model in the Nexys Video card.
  - For hls4ml, sigmoid is used at the output layer, but the HDL-based version is not.
  - In the HDL-based design, each layer is doing relu( $\Sigma_i(w_i \times input_i) + b$ ) in a register, so 1 clock-cycle is consumed.



| | hls4ml | HDL-based NN |
|---|---|---|
| LUT | 35278 (26%) | 41097 (30%) |
| FF | 52620 (19%) | 1276 (0.5%) |
| DSP | 289 (39%) | 0 (0%) |
| Latency (clock-cycle) | 30 | 4 |

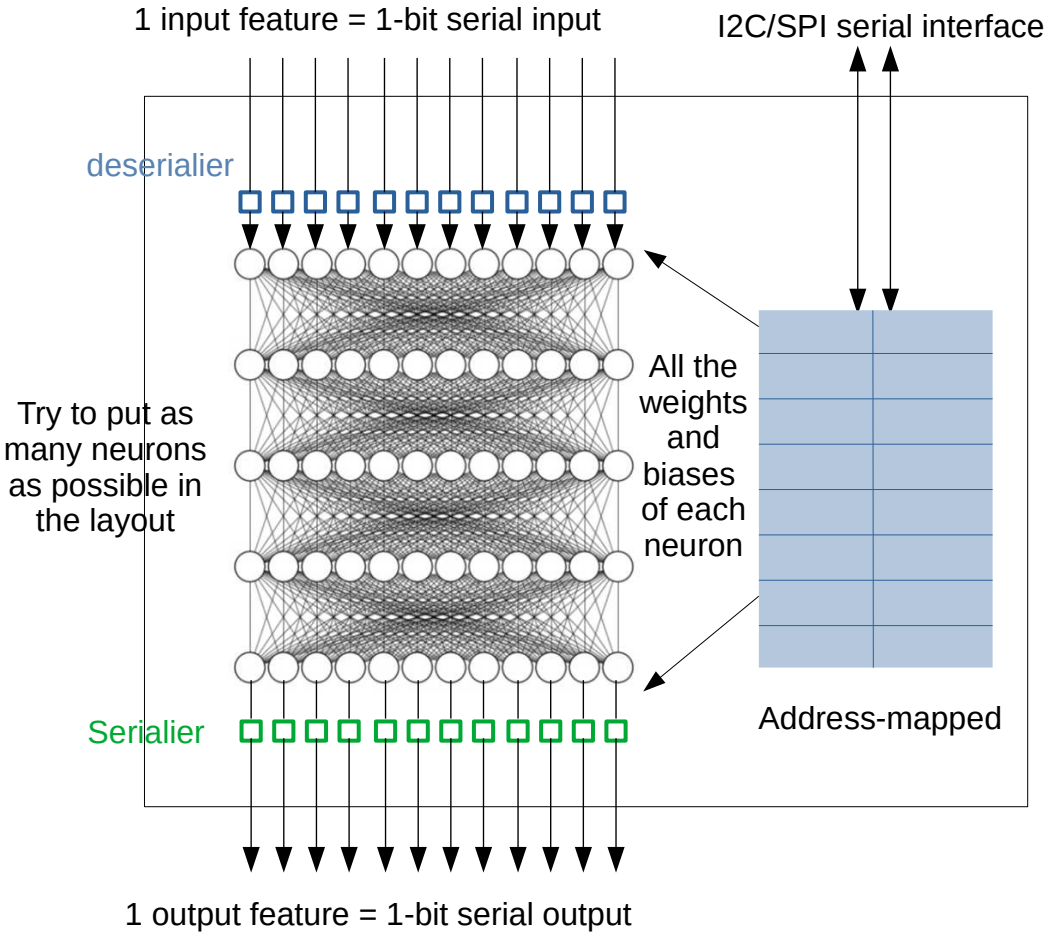
# Further application: digital ASIC

- Therefore, with some conditions, make a NN with plain HDL is doable.
- **If everything is written by plain HDL, it can be made into an ASIC.**
- The approaches with higher level are convenient. Although the lower level design at RTL or even transistor level are relatively technically difficult, it has better flexibility on the hardware design and faster processing.



# HDL-based NN in ASIC: programmable design

- Can we make the QNN in ASIC progammable, such that the utilization is more flexible?
- Yes, all the weights and biases can be controlled by I2C or SPI interface.
- The entire network can be hence programmable.
- Since the NN is in a genrael form, we can also select the scale of the network to be used.




- Plan:
  - HDL design is almost ready.
  - Started layout design for ASIC.
  - Aiming for submission in 2026 with TSMC 22nm.
  - Circuit board design: Using FPGA for serial I/O and control.
- We are expecting that a small-scale NN in ASIC (outside of FPGA) up to O(GHz) processing could be helpful for FEE or trigger in real-time DAQ.

# Summary

- In this report, we talked about our ML in electornics related research activities in Collider Electronics Forum (CEF) in KEK ITDC.

- Versal project: Possible application of high-end SoC device in experimental HEP.
  - In addition to the HLS-based ML inference in FPGA, Versal provides new features of computation acceleration engine: AI engine and DPU.
    - A technical database is built with providing education activity.
  - With Versal, real application in hardware system and R&D of new device/system are under development.

- AI in FEE project: Application of compact AI/ML model in devices in Front-End level
  - A newly started project. We are collecting potential developments.
  - A concept of QNN in ASIC is under development.

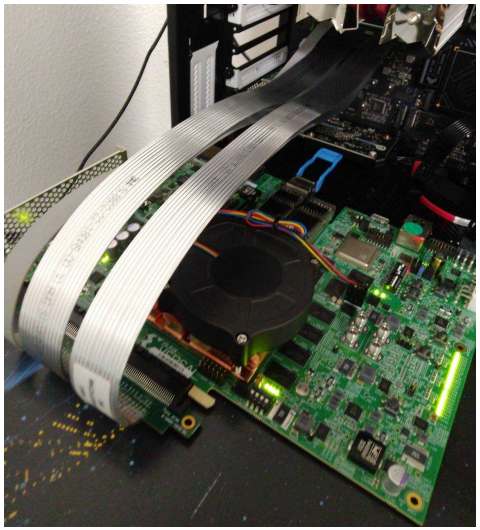
- You are welcome to contact us for any collaboration or inquiry for technical support.
  - Our workshop on 1$^{st}$ and 2$^{nd}$ Dec.: https://kds.kek.jp/event/57383/

# Backup

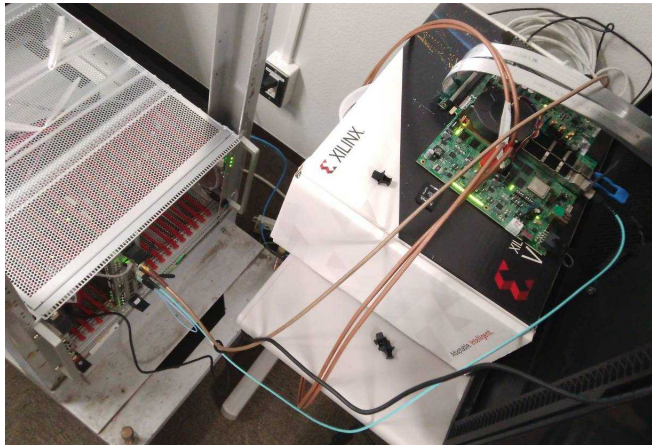
# Test benches of Versal kits @ KEK E-sys

- 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**