# Development of first level track trigger at Belle II using Deep Neural Network

**Yuxin Liu(Speaker)**[A] , Taichiro Koga[B], Christian Kiesling[c], Felix Meggendorfer[c], Timo Forsthofer[c], Simon Hiesl[c], Kai Lukas Unger[D]

Institution : Sokendai(KEK)[A], KEK[B], Max Planck Institute for Physics[C], Institute for Information[D] Processing Technologies

25th April ,2024, 24th IEEE REAL TIME CONFERENCE

# OUTLINE

· **Introduction**

- SuperKEKB and Belle II detectors
- Motivation for new track trigger development
- First level trigger and CDC track trigger system at Belle II

· **Development of DNN track trigger**

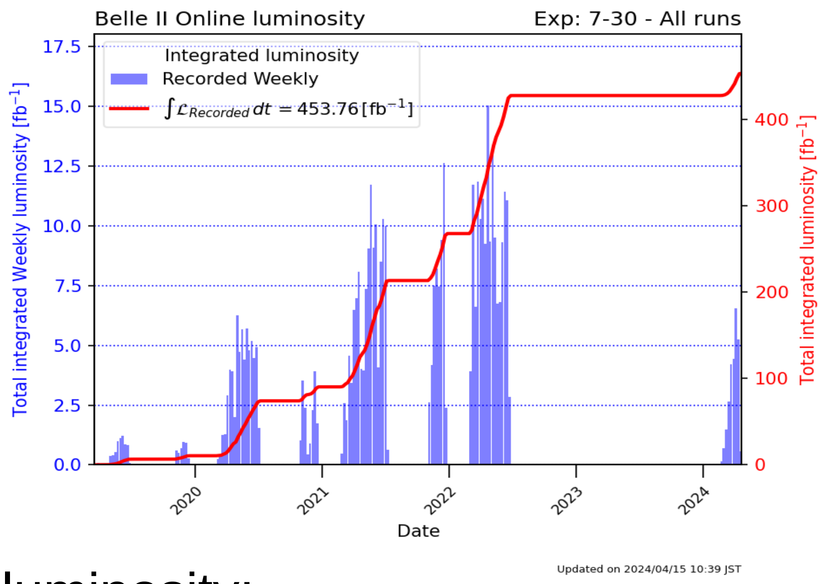
- DNN track trigger architecture
- Training and optimization
- Hardware implementation
- RTL simulation results

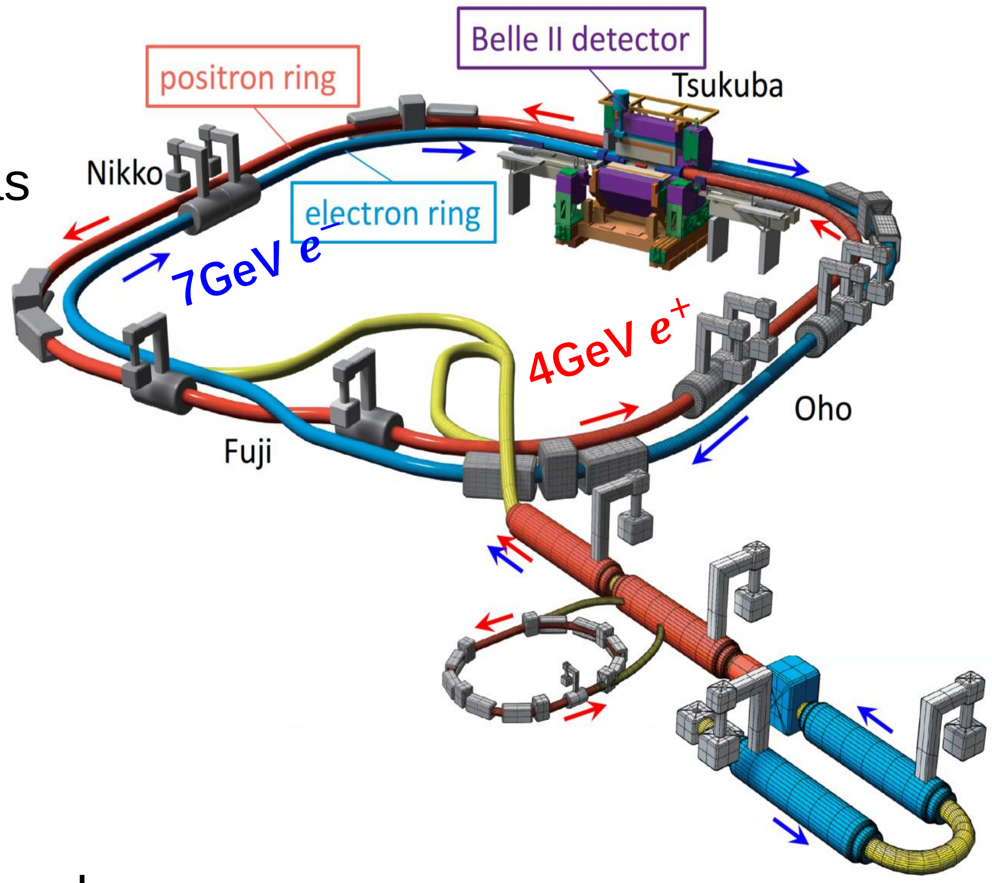
·**Summary**

# SuperKEKB

- An asymmetric $e^- \; e^+$ collider, Upgrade from KEKB.
  7.0 GeV $e^-$ and 4.0 GeV $e^+$ for $\Upsilon(4S)$

- SuperKEKB aimed for a peak luminosity of $6 \times 10^{35}\ \mathrm{cm}^{-2}s^{-1}$, surpassing KEKB by 30 times and setting a world record; also with the integral luminosity as $50\ ab^{-1}$ ;




- Achieved luminosity:

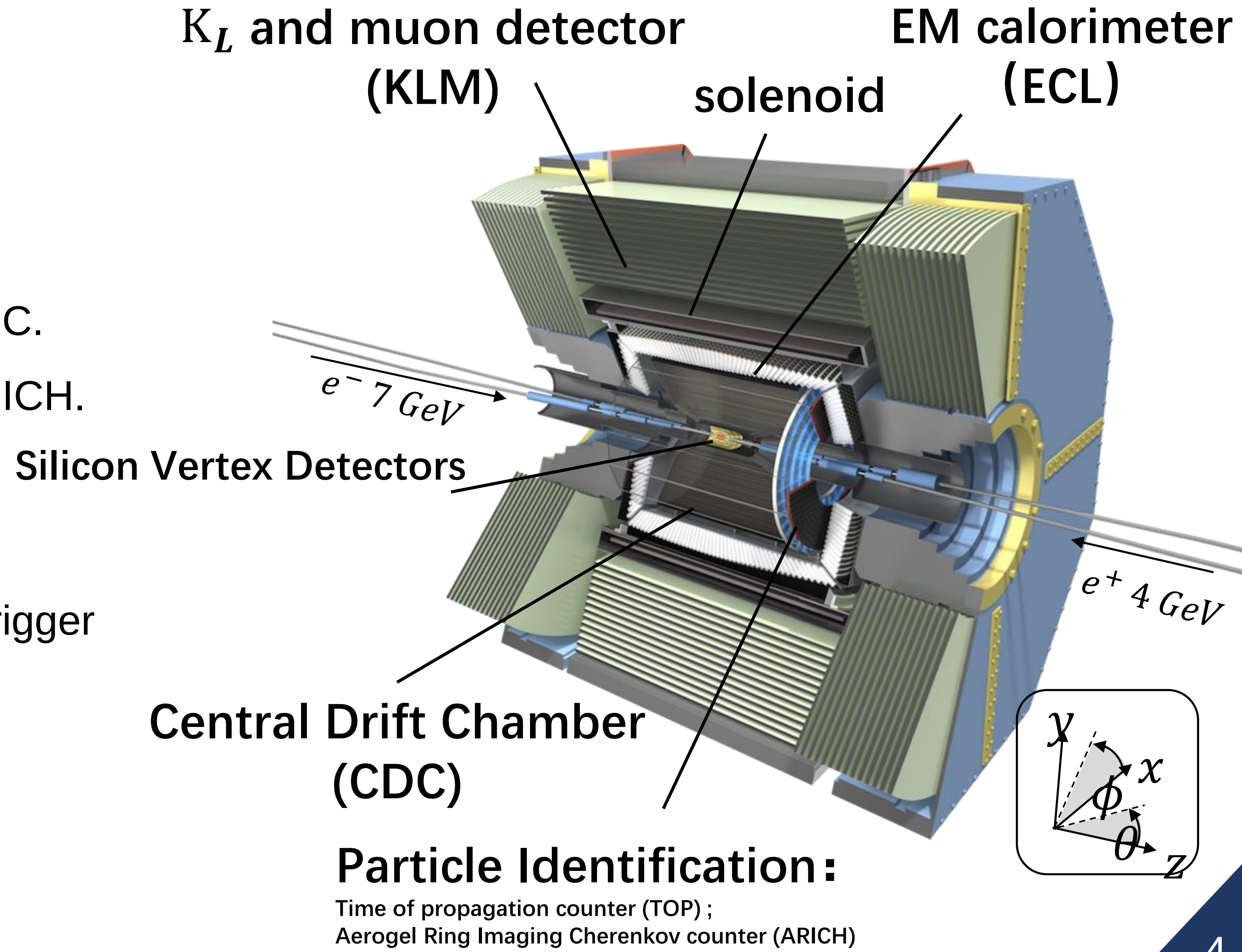
$$\mathcal{L}_{peak} = 4.65 \times 10^{34}\ \mathrm{cm}^{-2}\mathrm{s}^{-1}$$, two time of KEKB record

$$\mathcal{L}_{int} = \ 453\ fb^{-1}$$ ; till April 2024

# Belle II detectors

**Belle II including:**

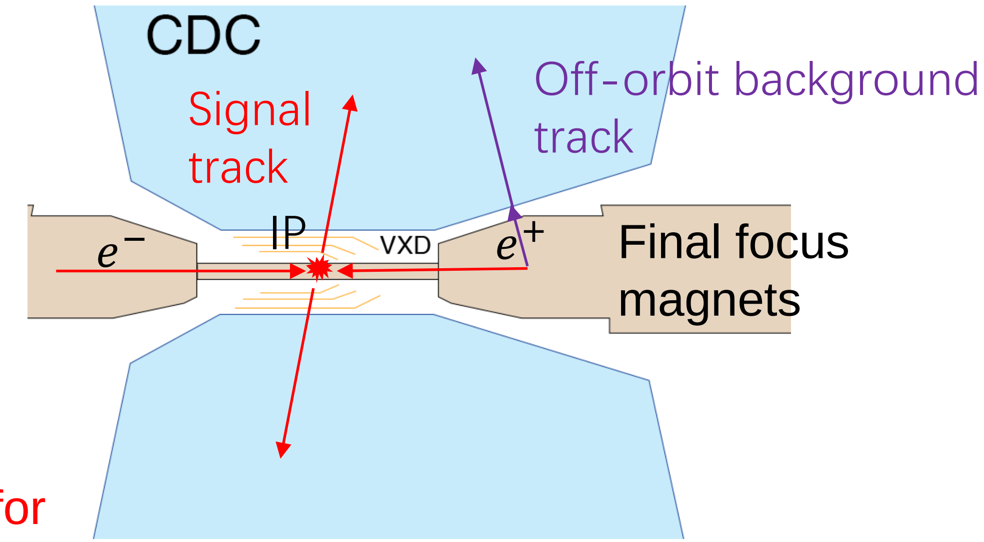
- Tracking: Vertex detectors and CDC.
- particle identification: TOP and ARICH.
- Calorimeter: ECL.
- KL and muon detector.
- First level (L1) trigger, High level trigger (HLT) and DAQ.

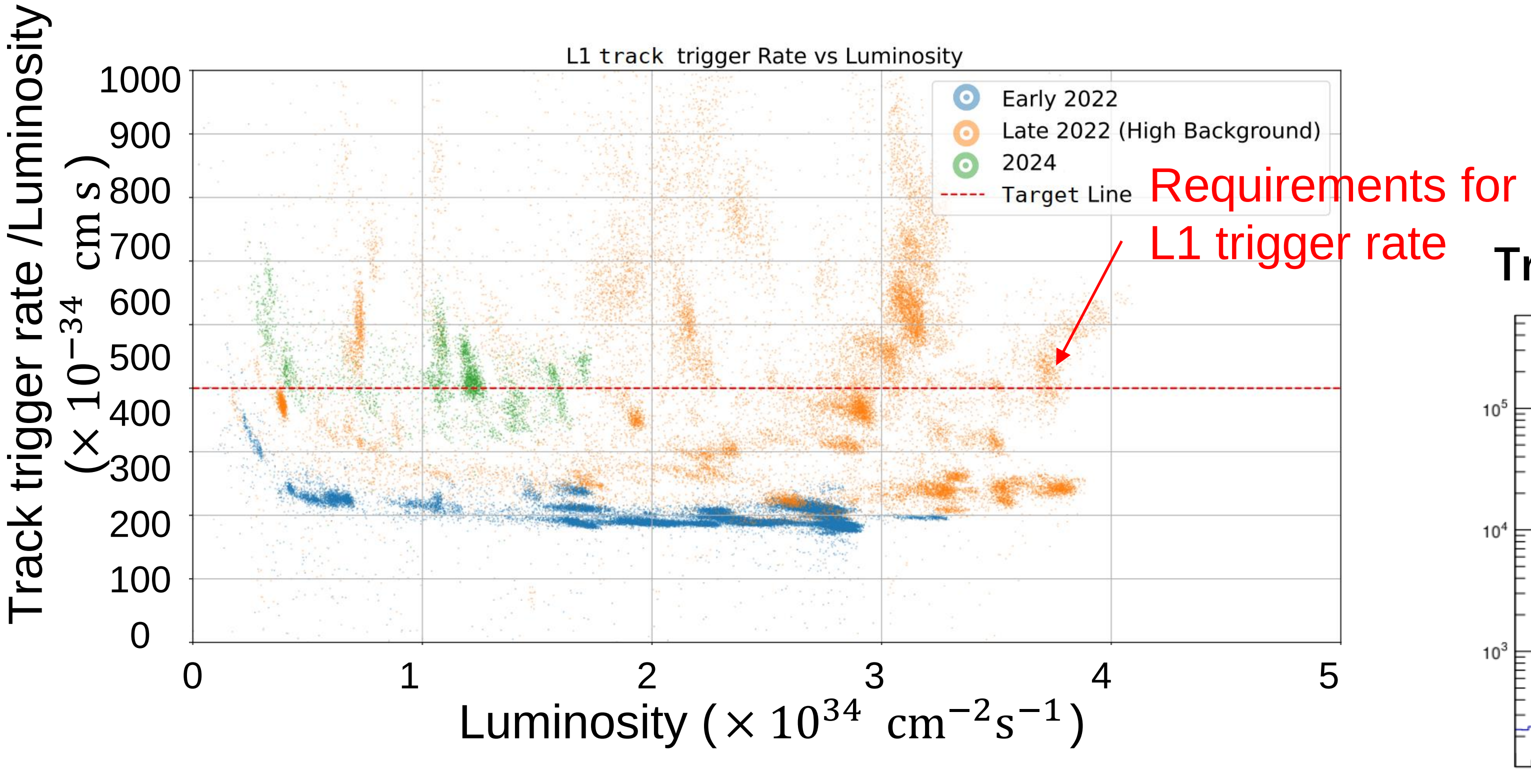


# Motivation for track trigger upgrading

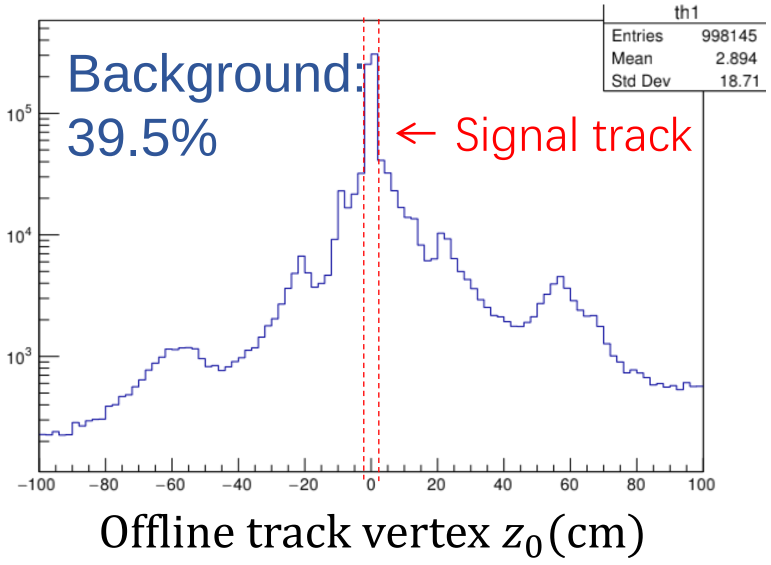
**Requirements for first level trigger system**

1. High efficiency for hadronic events from $\Upsilon(4S) \rightarrow B\bar{B}$
2. A maximum average trigger rate of 30kHz
3. A fixed latency of about 4.4 μs



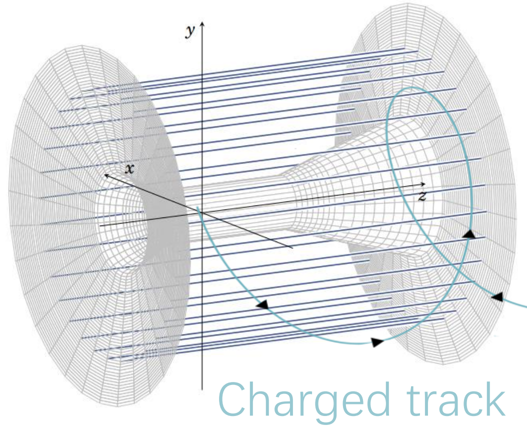

**Tracks $z_0$ distribution after trigger**



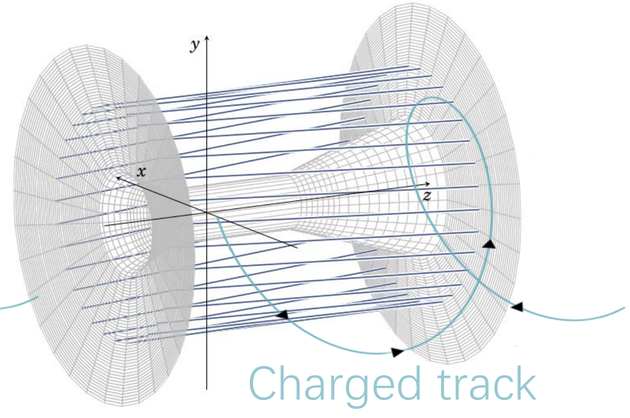

- **we aim to decrease the track trigger rate, thereby lowering the overall trigger rate.**

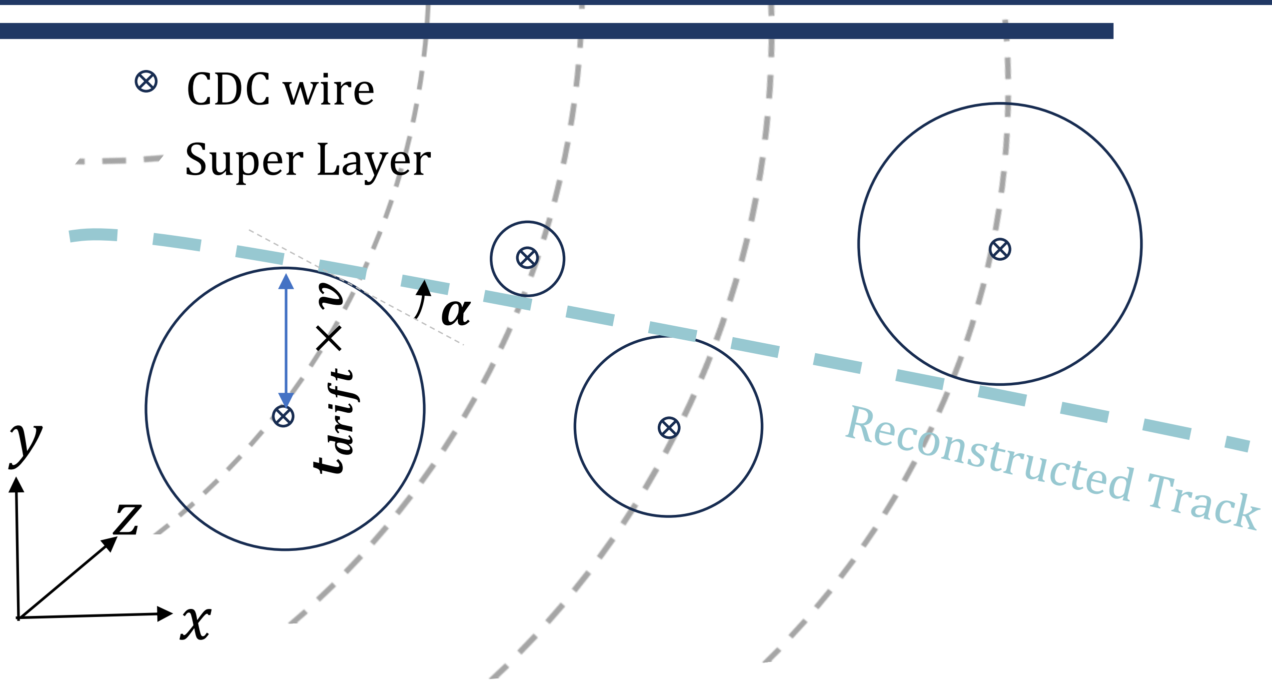
# Neural-network based 3D track reconstruction

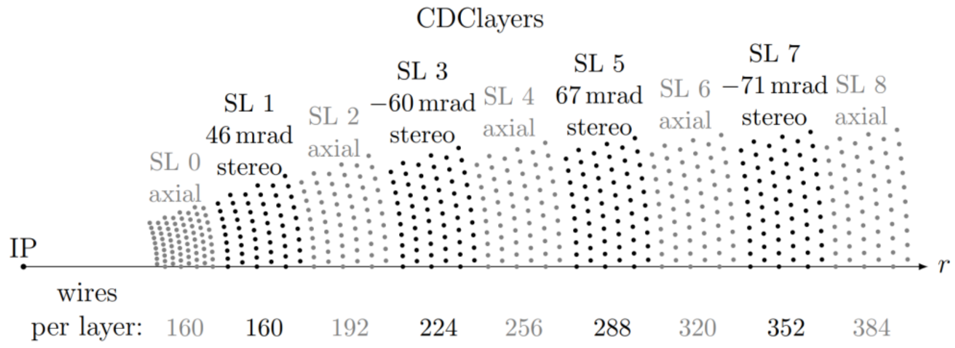
CDC Axial wires
(parallel to beam direction)




CDC Stereo wires
(oblique to beam direction)



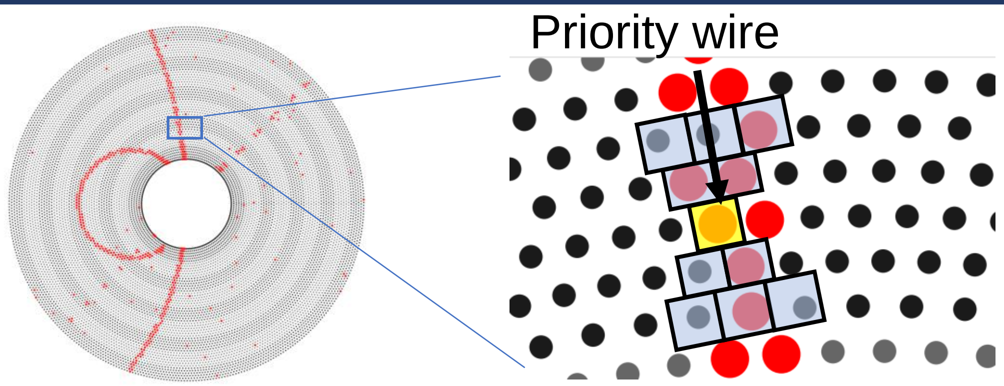

- Use Axial wires reconstruct 2D track in x-y plane and Stereo wires for full track reconstruction

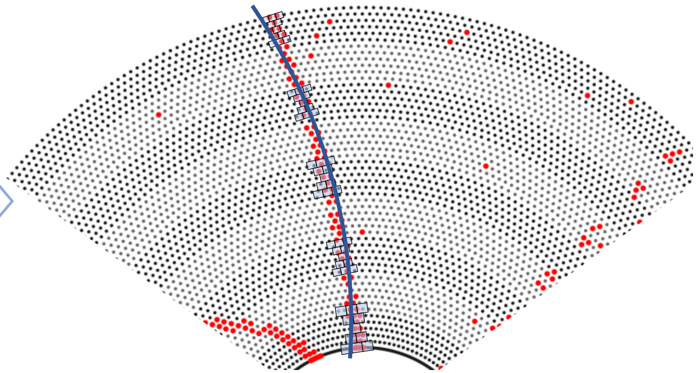
- Used information: **location for CDC wires ($\phi$ and r)** , **drift time ($t_{drift}$)** , and **crossing angle ($\alpha$)**

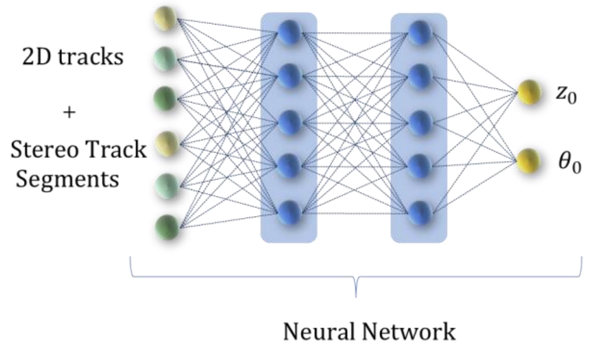
- Use **neural-network** to handle complicated track fitting and reject possible background hits

# First level CDC track trigger

https://arxiv.org/abs/2402.14962



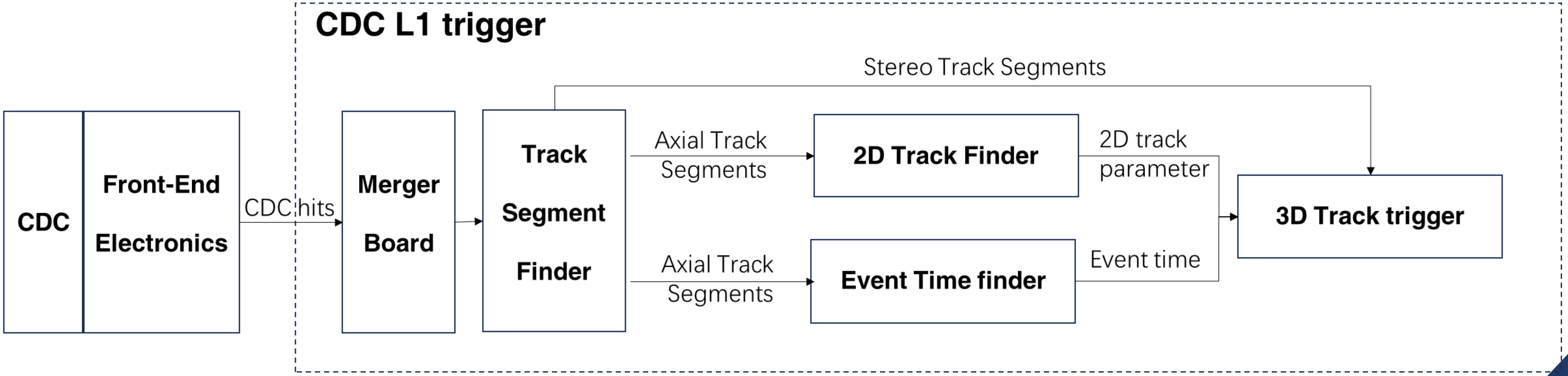



**CDC raw hits**

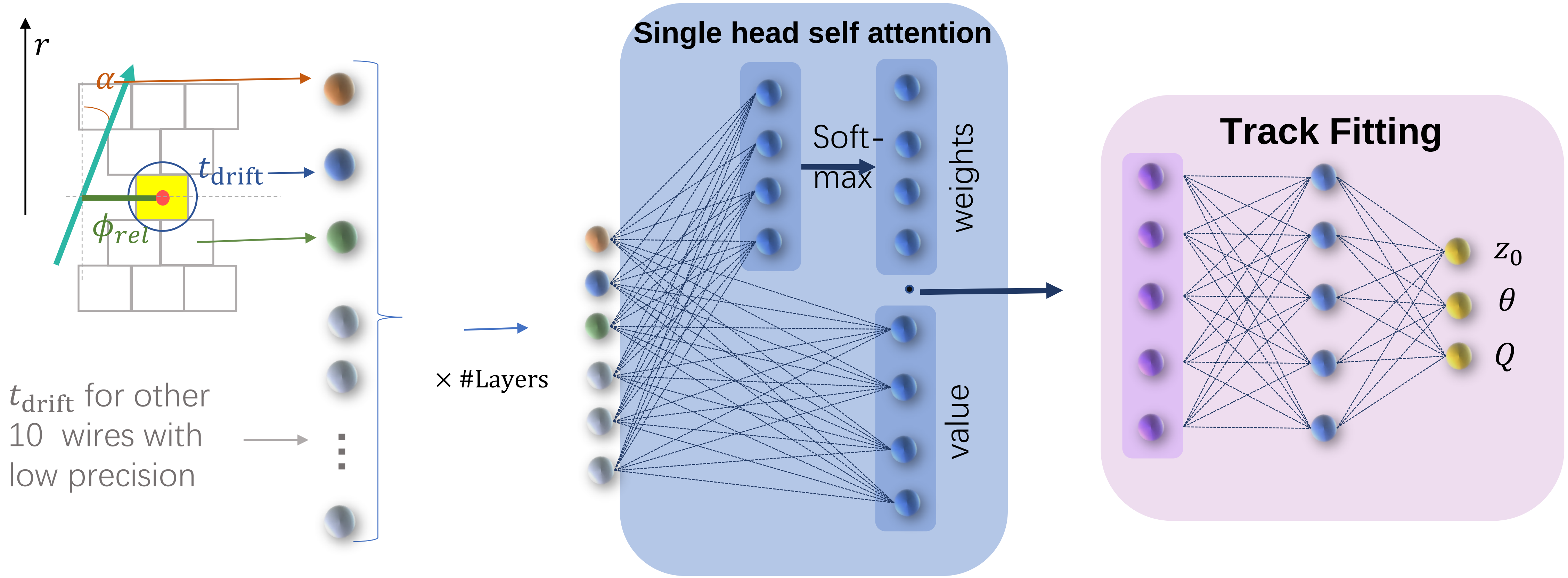
Built **Track Segment** (a set of CDC wires) in every super layer

Build **2D track** with **axial hits** using Hough transformation

Build **3D track** with **stereo hits** and 2D track using **Neural network**



# Neural-network inputs and architecture upgrade

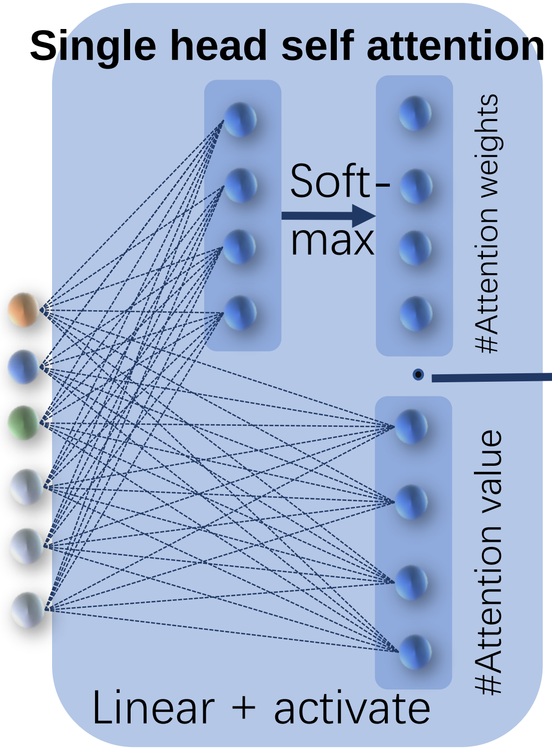


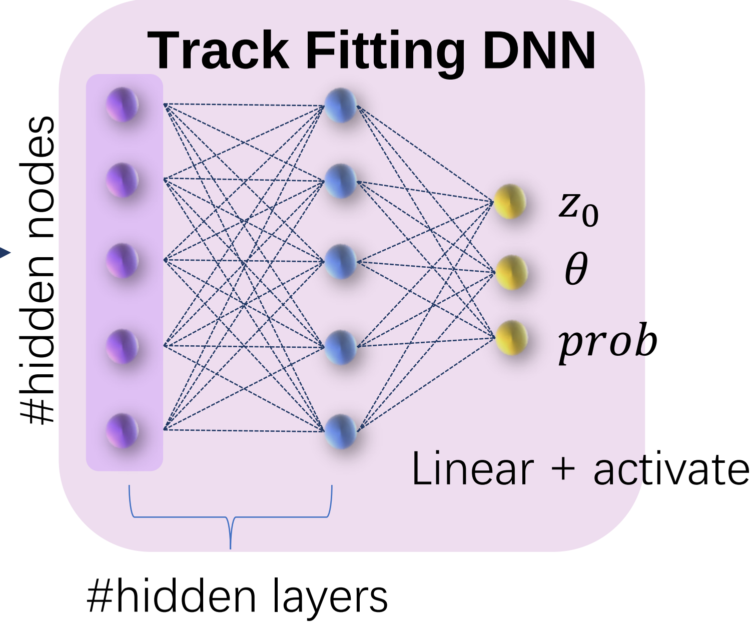

- Inputs: **Drift time $t_{\text{drift}}$**, **wires relative location $\phi_{rel}$**, **Crossing angle $\alpha$** for priority wires
  + **Drift time for all other wires**

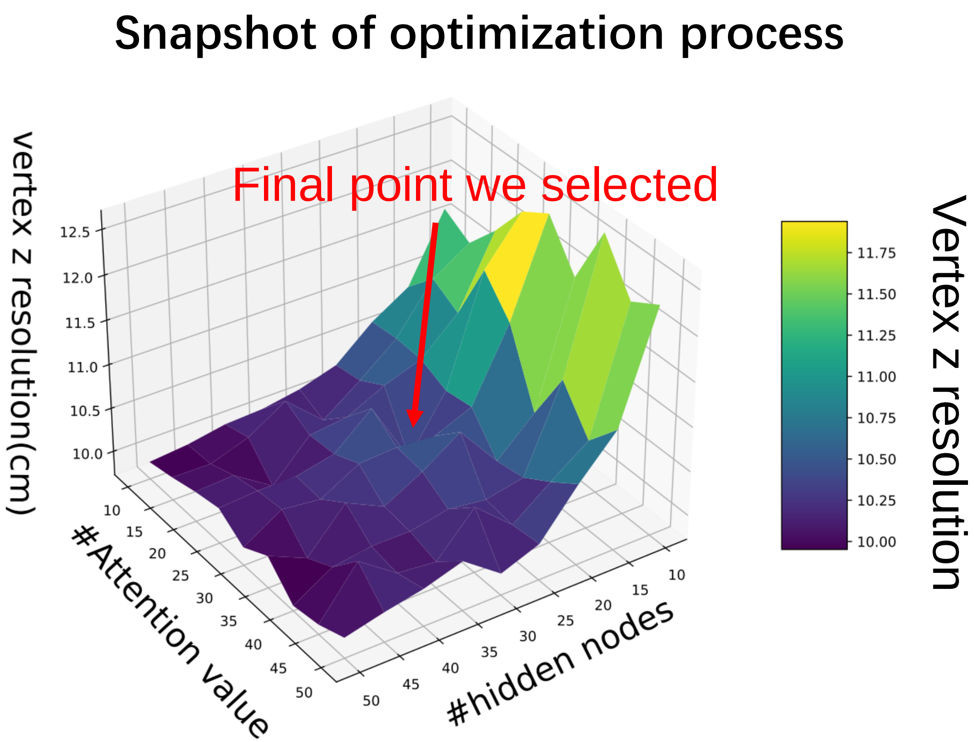
- Introduce the **self-attention architecture** to “focus” on certain inputs

- Output track vertex $z_0$, track $\theta$ and **classifier output $Q$**

# Neural-network training, optimization, quantization




- Data: real physics run data with high background in late 2022.
- Using PyTorch lib for model building and training, OPTUNA for parameters optimization

| Parameter | #Attention value | #hidden nodes | #hidden layer | activate | precision | Total multiplier |
|---|---|---|---|---|---|---|
| Values | **27** | **27** | **2** | **Leaky Relu** | **Float 16** | **4,185** |

# Deep neural-network implementation

Upgrade

| Universal Trigger board (UT) generation | 3rd | 4th |
|---|---|---|
| FPGA | Virtex 6<br>XC6VHX380 | Virtex UltraScale<br>XCVU160 |
| DSP | 864 | 1560 |
| Logic gates | 380k | 2026k |
| Optical bandwidth | 530 Gbps | 1300 Gbps |

Requirements for implementation:

- Latency: ~300ns (3rd) and ~600ns (4th)
- DSP limitation: 864 (3rd) and 1560 (4th)
- More than 5 times logic gates, can be used for multiply

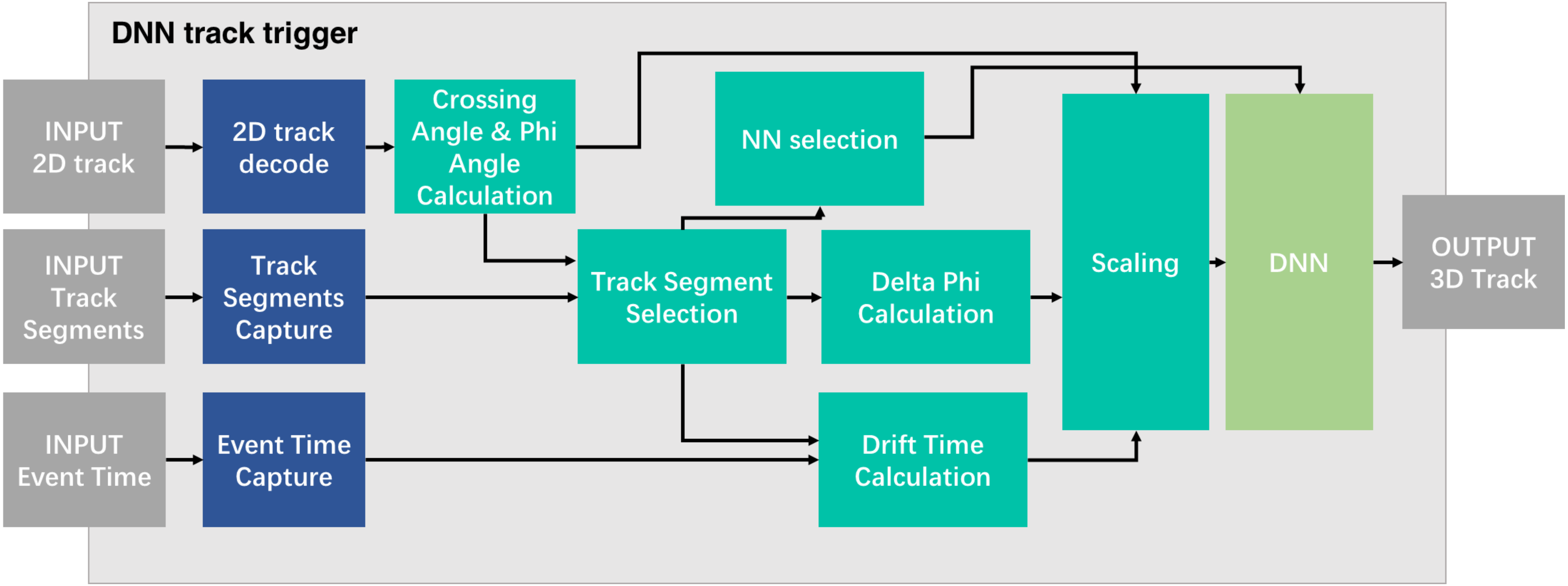
Belle II UT3

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

Belle II UT4

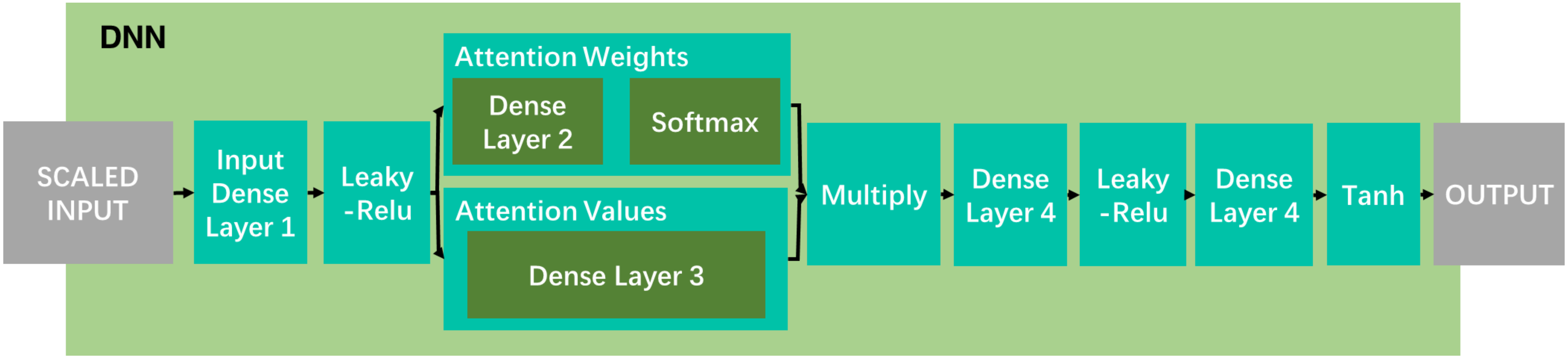
Xilinx UltraScale
XCVU080, XCVU160
25 Gbps with 64B/66B

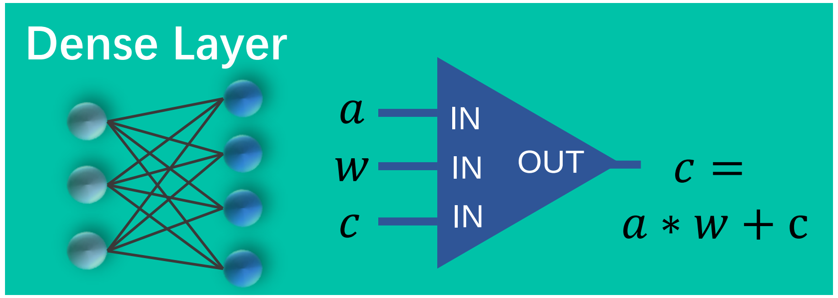
# DNN track trigger firmware architecture

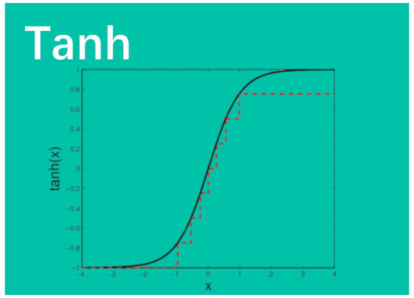


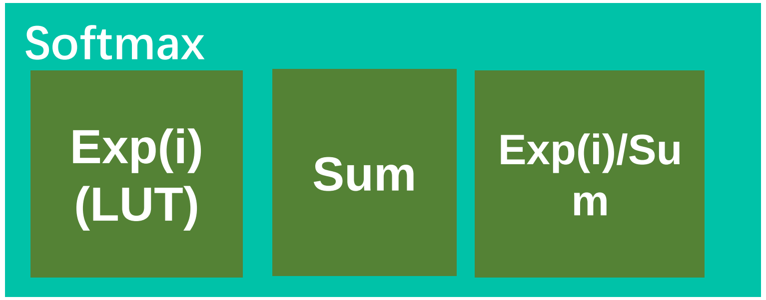

- Input 2D track, track segments and event time pre-processing them to get scaled input for DNN.
- Pre-processing & interface using VIVADO, Core DNN logic using XILINX VITIS

# Firmware architecture for DNN TRG



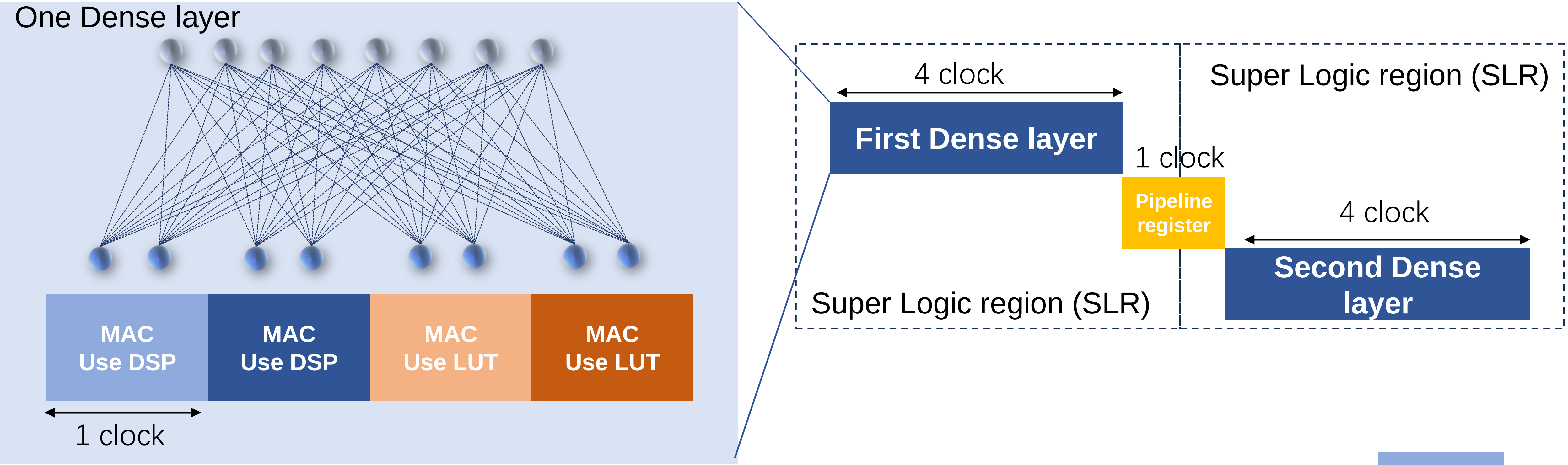



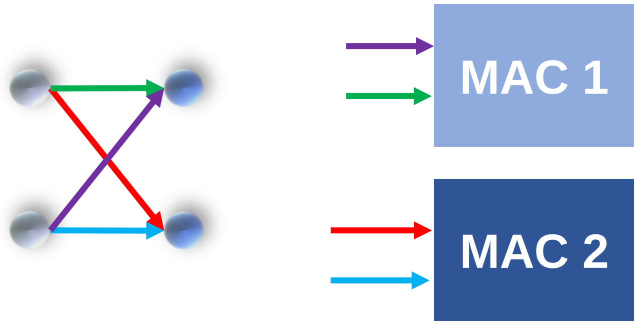




- Using look up table with 18 bits precision for $\exp(x)$ & $\tanh(x)$, refer to the function in hls4ml
- Directly use DSP for Leaky ReLU
- For Dense layer, using specific strategy to fit the requirements (next page)

# Pipelined dense layer with Heterogenous resources

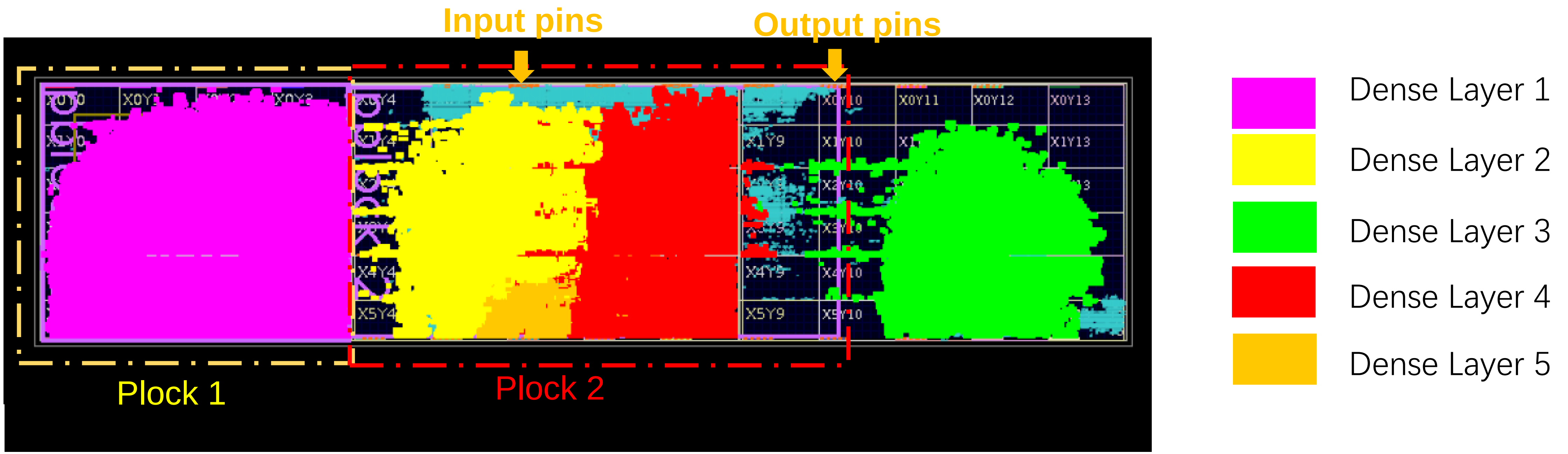


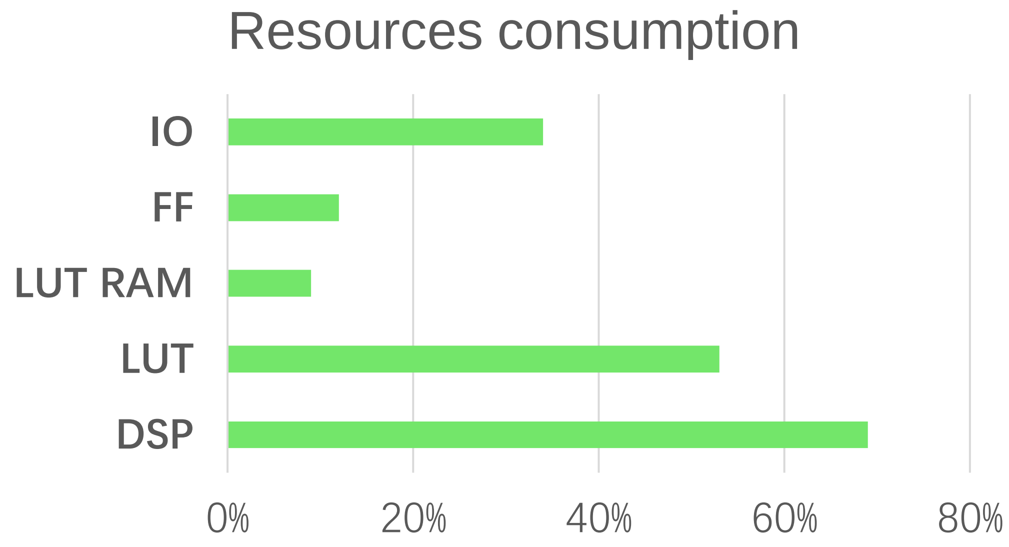

- Using both LUT and DSP to perform Multiply And Accumulate (MAC)
- Reuse each MAC twice.
- Pipeline dense layer with Interval as 4 clock
- Additional Pipelined register was added to cross SLR
- Floor-Planning each dense layer




Reuse every multiplier by twice

# Floor planning and Implementation result



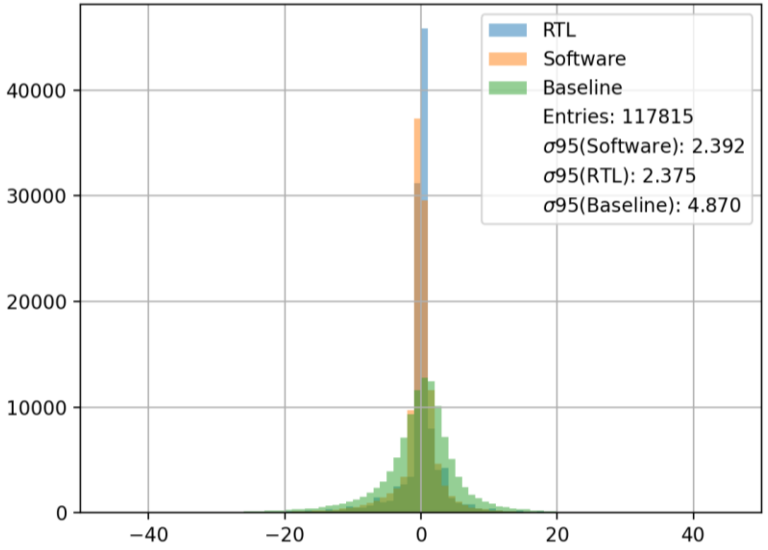




- Floor planning the dense layers :
- Resource  matched requirements, not timing violation
- Latency : 76 clock = 592.8 ns  ;require: < 600ns
- Initial Interval = 4 clocks          ;require:  4 clocks

# Register-transfer level (RTL) simulation

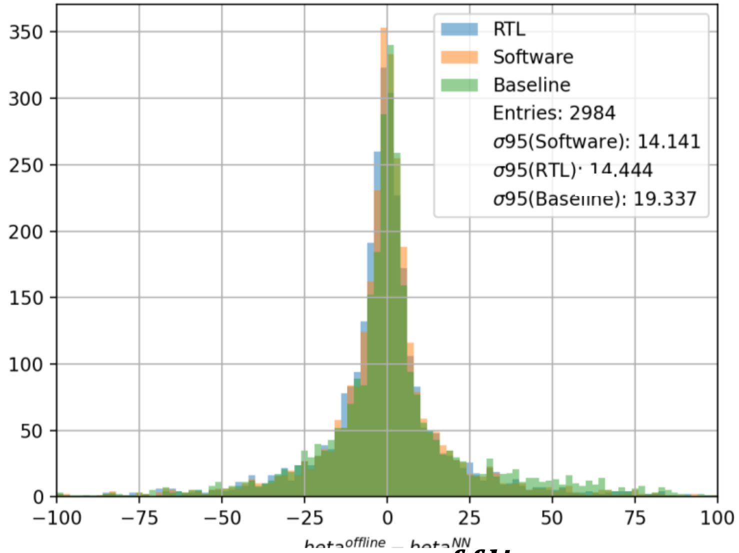
- Performance RTL simulation and comparing performance with pytorch results

Delta track z



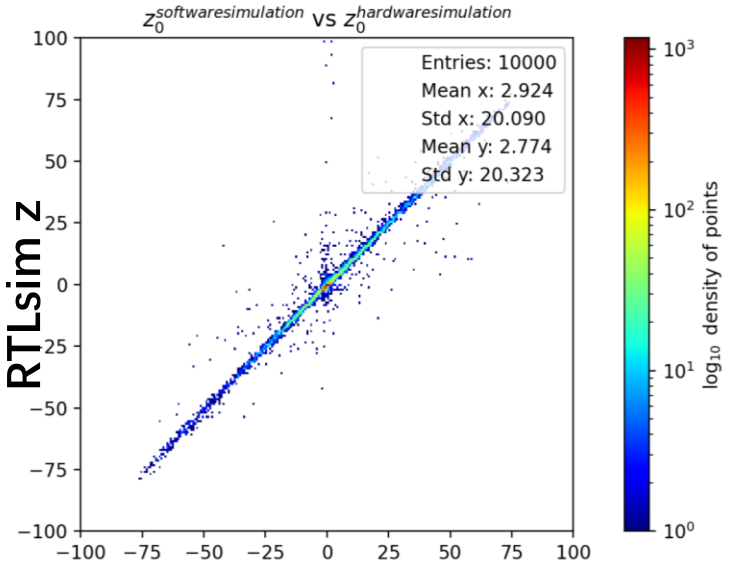

$z_0^{NN} - z_0^{offline}(cm)$

Delta track theta




$\theta_0^{NN} - \theta_0^{offline}(°)$

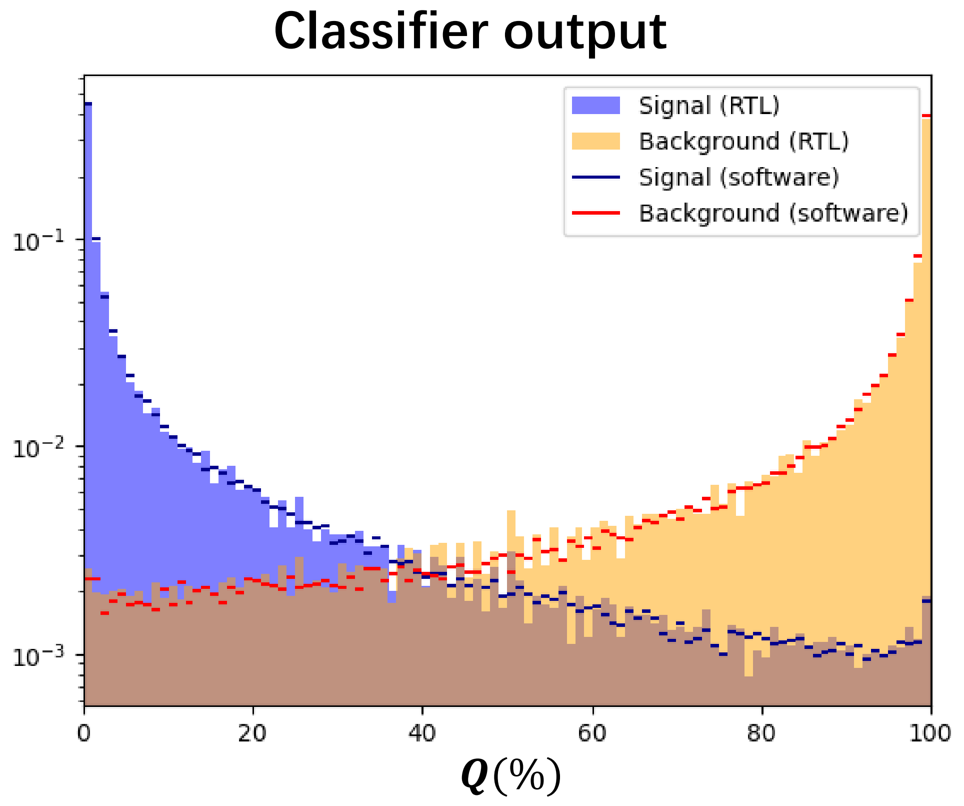
RTL co-sim vs software sim

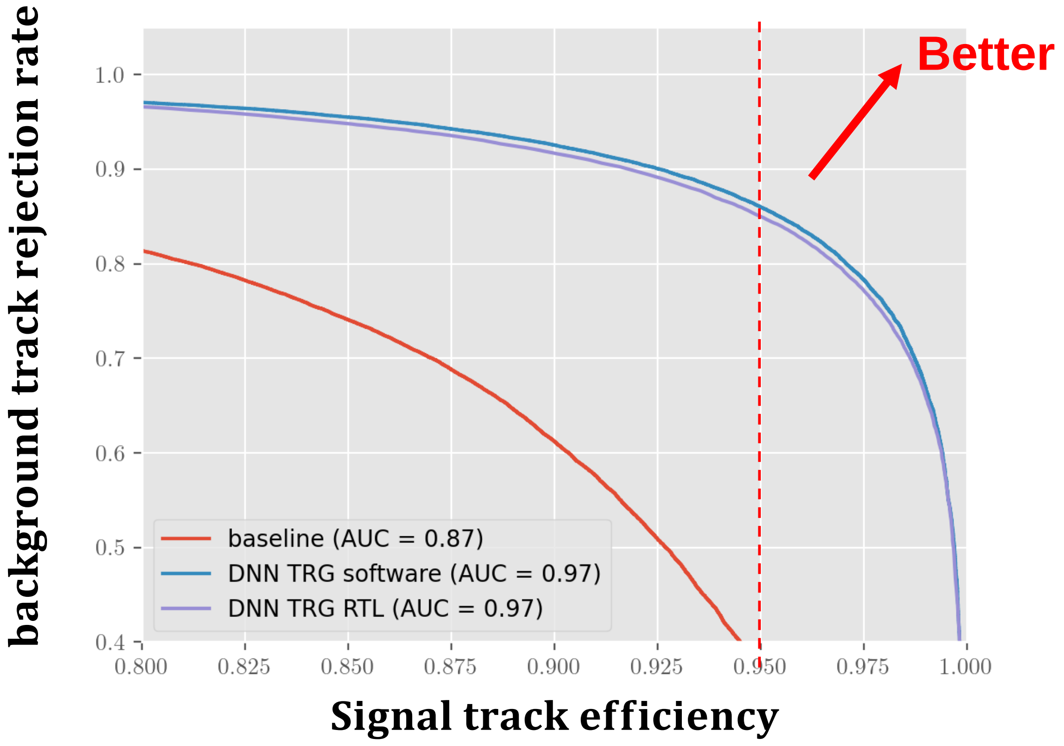



software sim z

- $\sigma^{z_0} = 2.7\ cm$, about ½ as the baseline $\sigma^{z_0} = 4.9$ cm ; and $\sigma^{\theta} = 14°$ (baseline: $\sigma^{\theta} = 19°$ )
- RTL and software simulation matched. Reducing precision did not loss the resolution.

# Register-transfer level (RTL) simulation



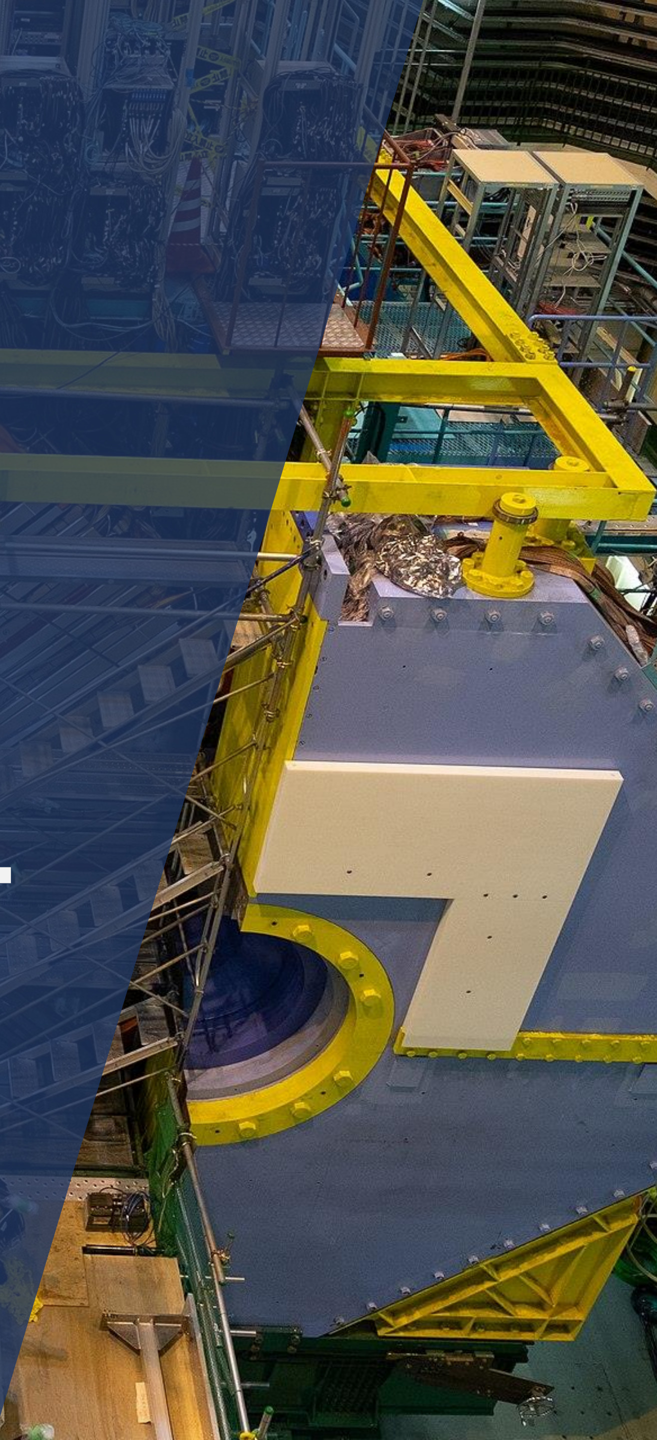

- $Q$ output got consistent with software result
- AUC do not get large drop comparing RTL and software simulation
- At signal track efficiency at ~95% :
  Background rejection rate: **NN track trigger (baseline): 39%; DNN track trigger: 85%**

# Summary

- The upgrade of Belle II first level track trigger is on-going
- We examined the performance for upgrade trigger with both software and RTL simulation, and achieved a 2.2 times background rejection rate improvement.
- We successfully implemented the DNN track trigger with UT4 module and fulfill the requirements with latency ~ 600ns and II ~ 4 clock.
- We are working on the commission work for the DNN track trigger

# Thanks for your attention

# Backup



# First level trigger

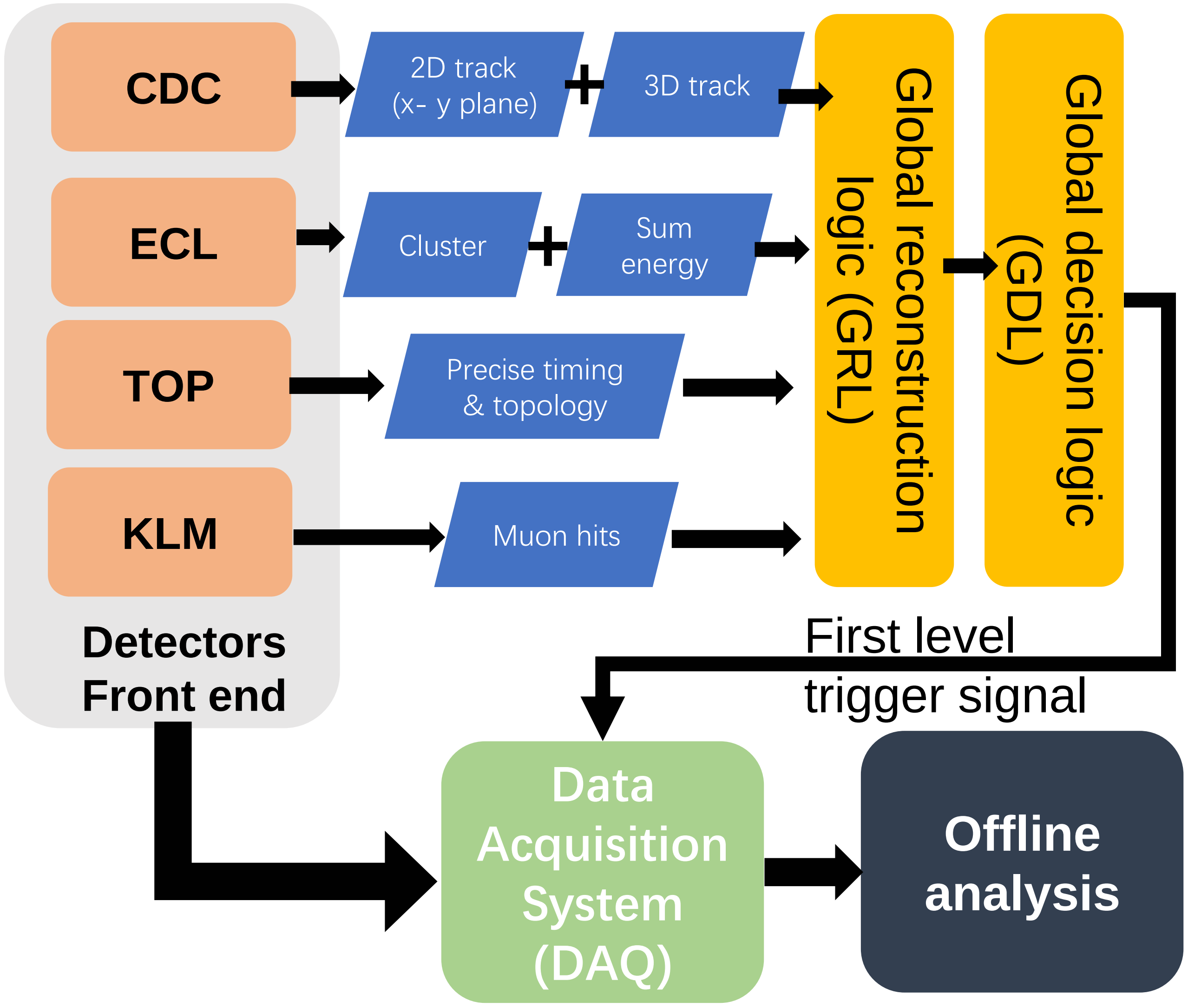
- Provide First level (L1) trigger signal to DAQ using FPGA for real-time processing on detector raw data.
- Include four sub-detectors trigger and 2 global trigger logic
- Implemented with third (fourth) generation of universal trigger board (UT3 / UT4)

**Belle II UT3**

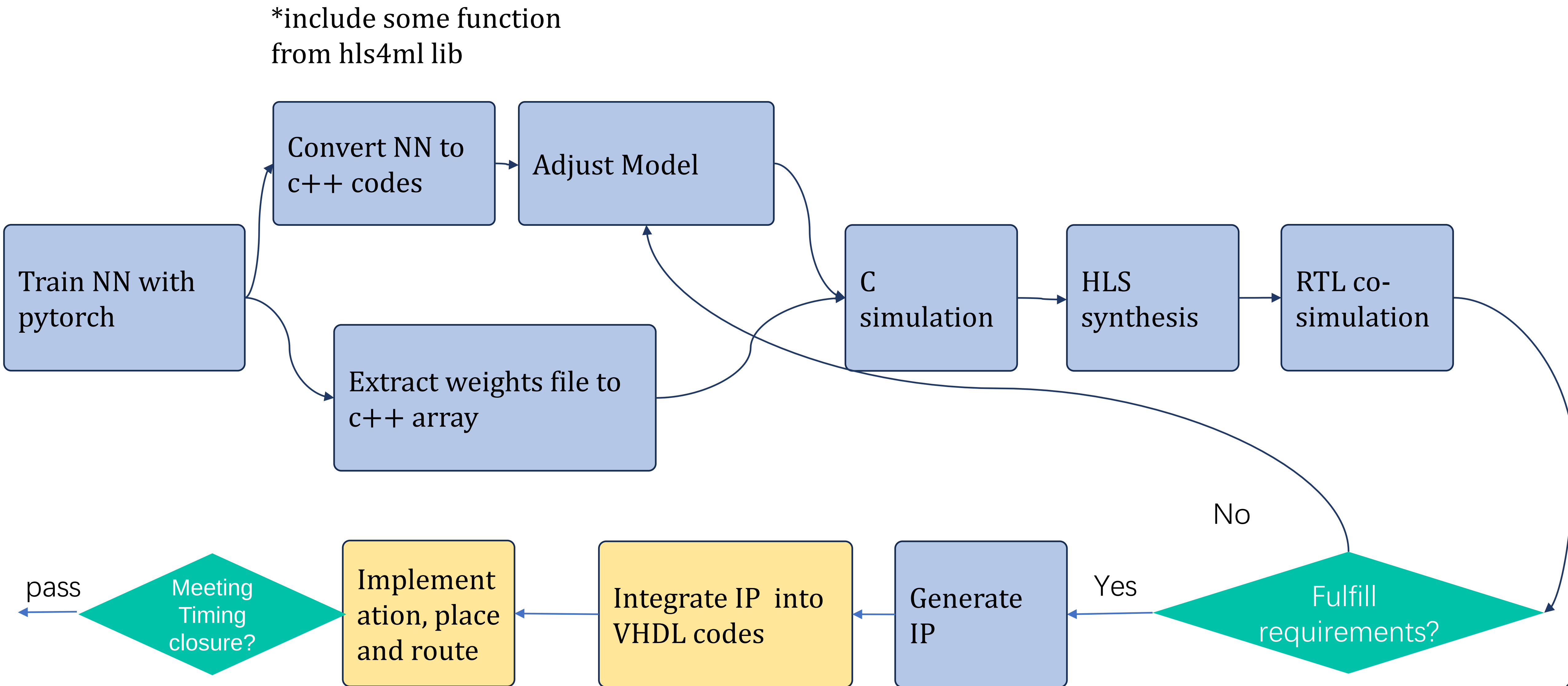
Xilinx Virtex-6
xc6vhx380t, xc6vhx565t
11.2 Gbps with 64B/66B

**Belle II UT4**

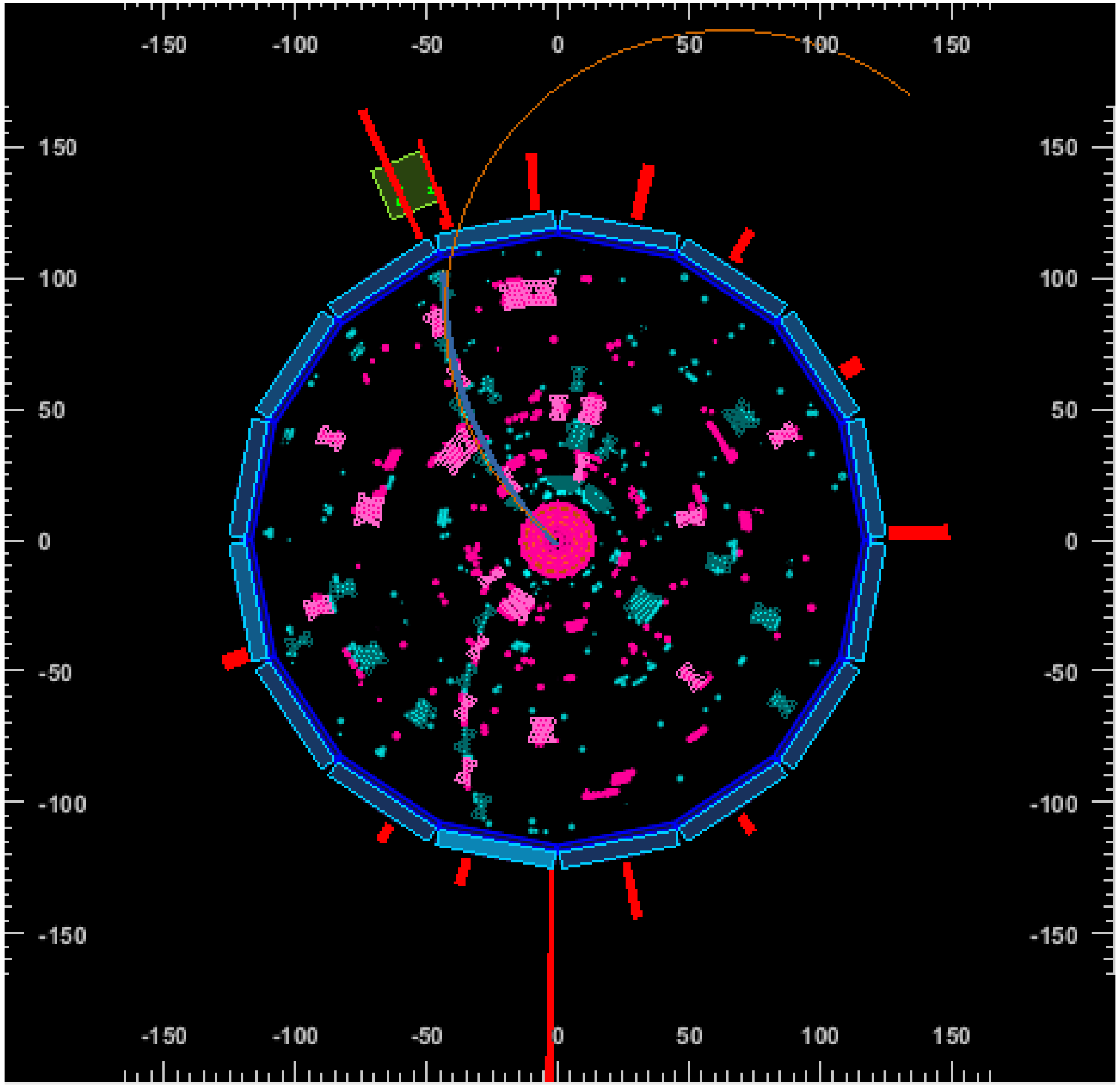
Xilinx UltraScale
XCVU080, XCVU160
25 Gbps with 64B/66B



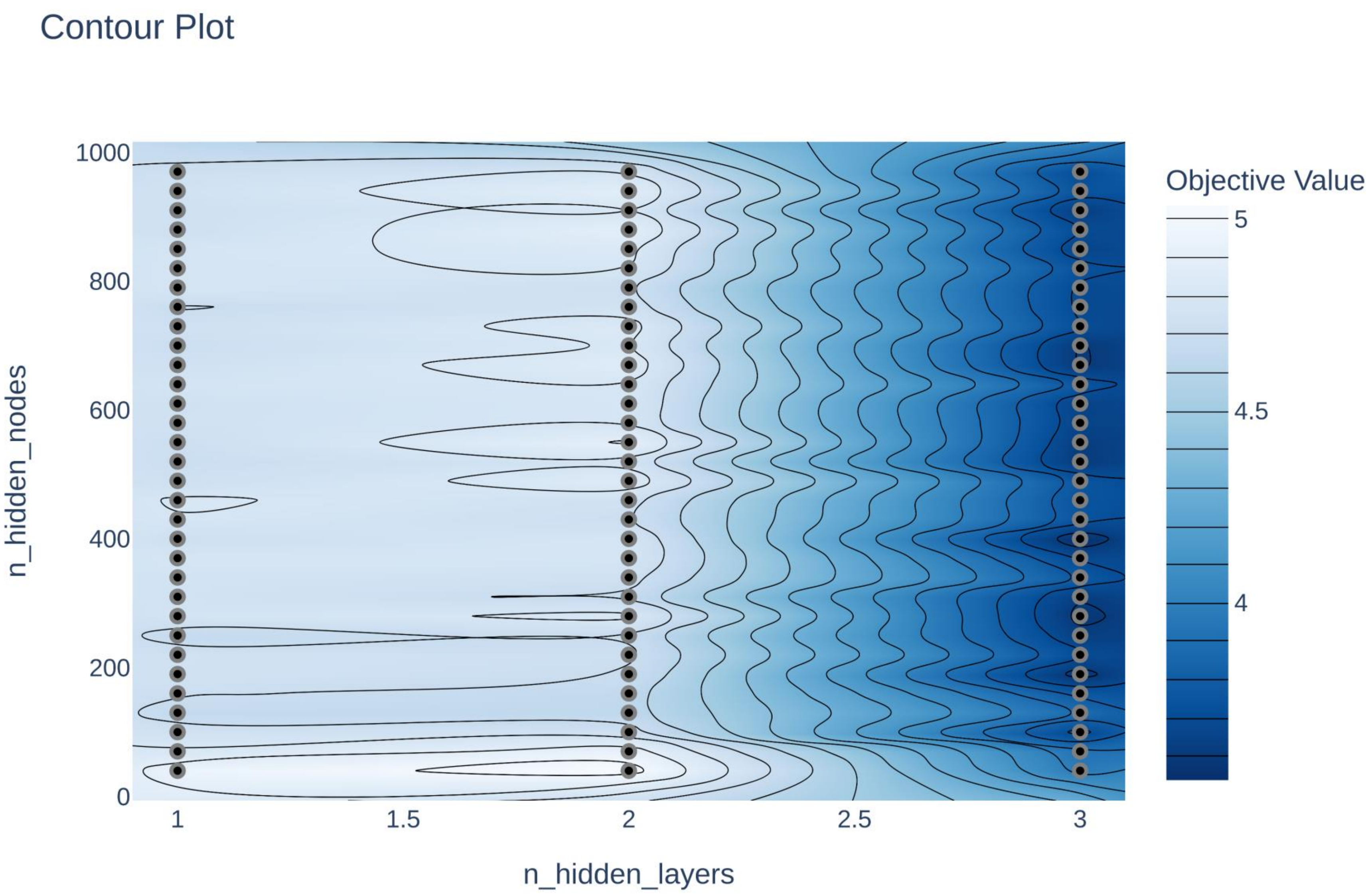
# Workflow with HLS



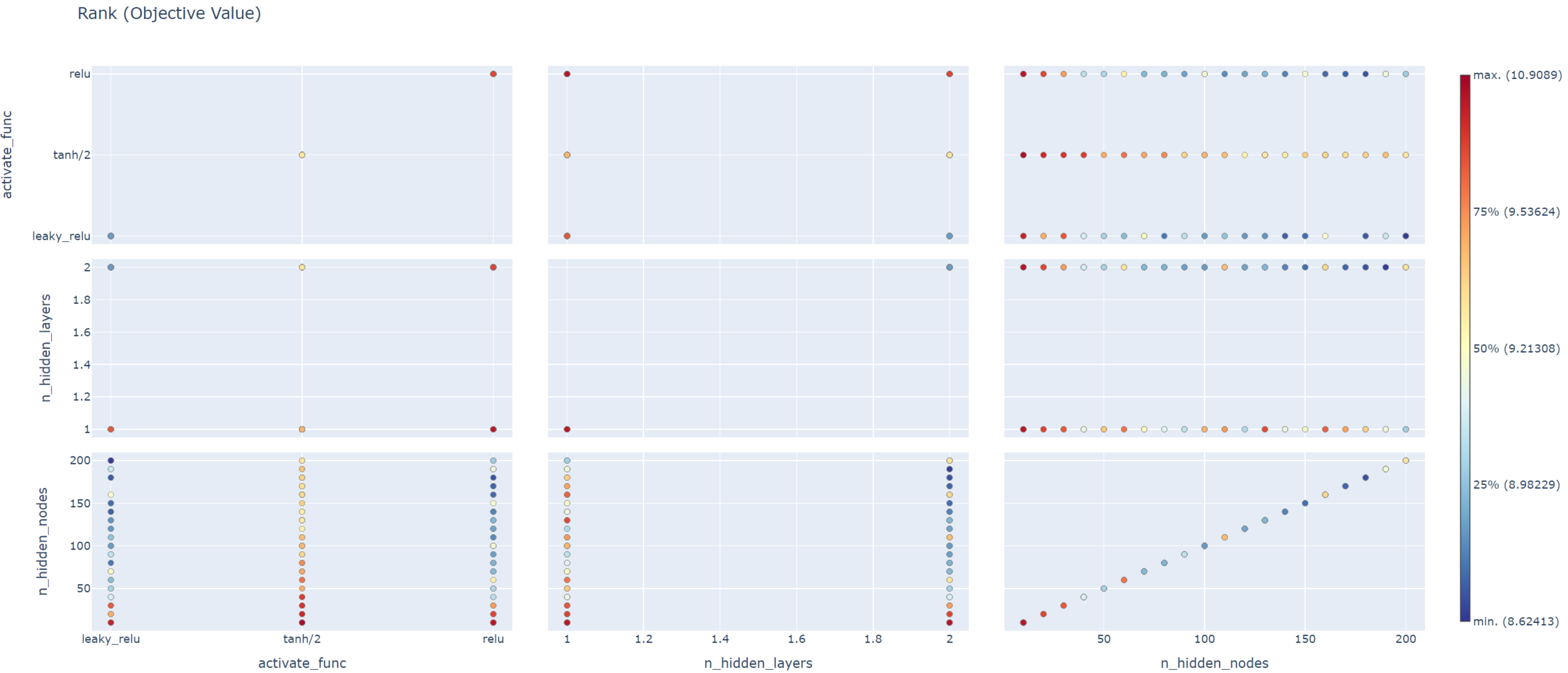
# General physics events shape



# Depth is much more powerful than width

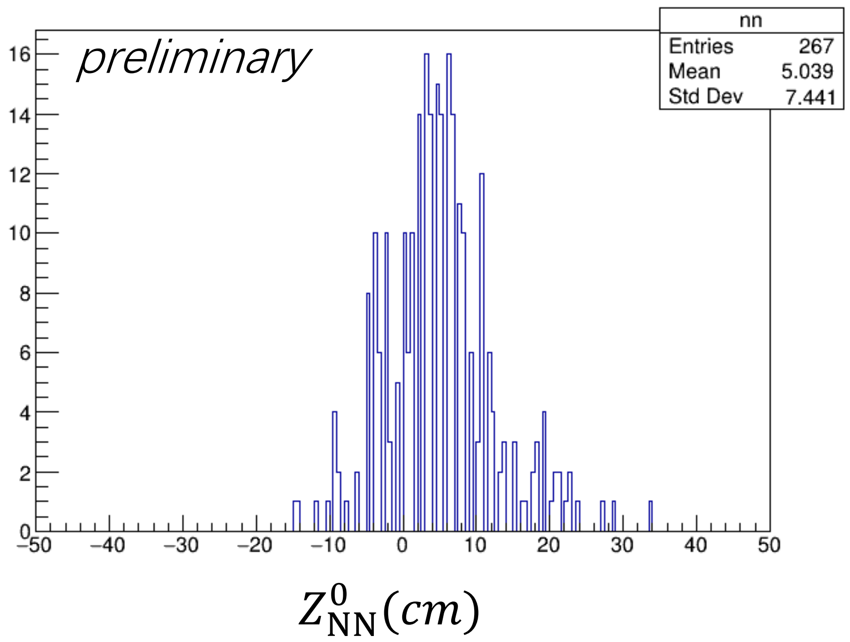


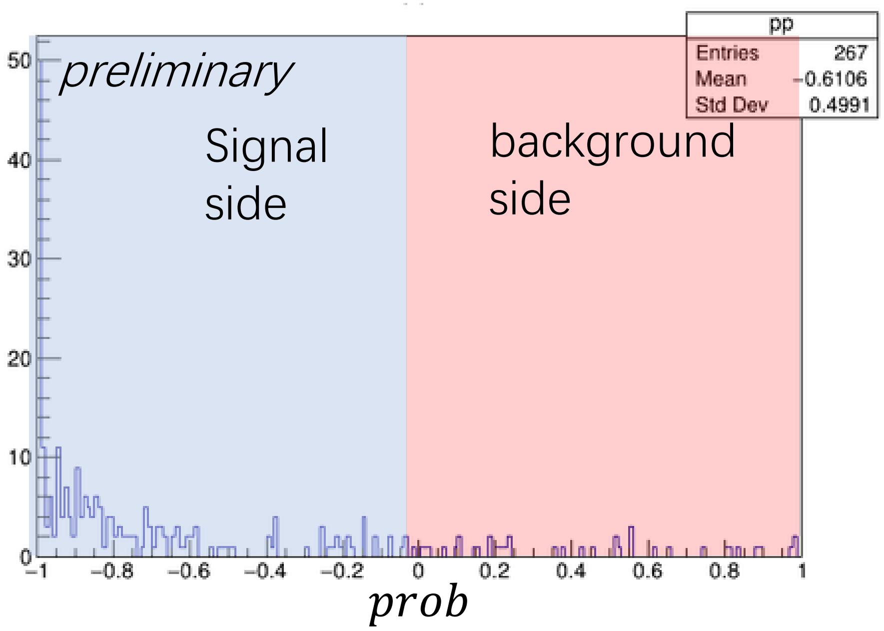
# Optimization for Self-attention MLP



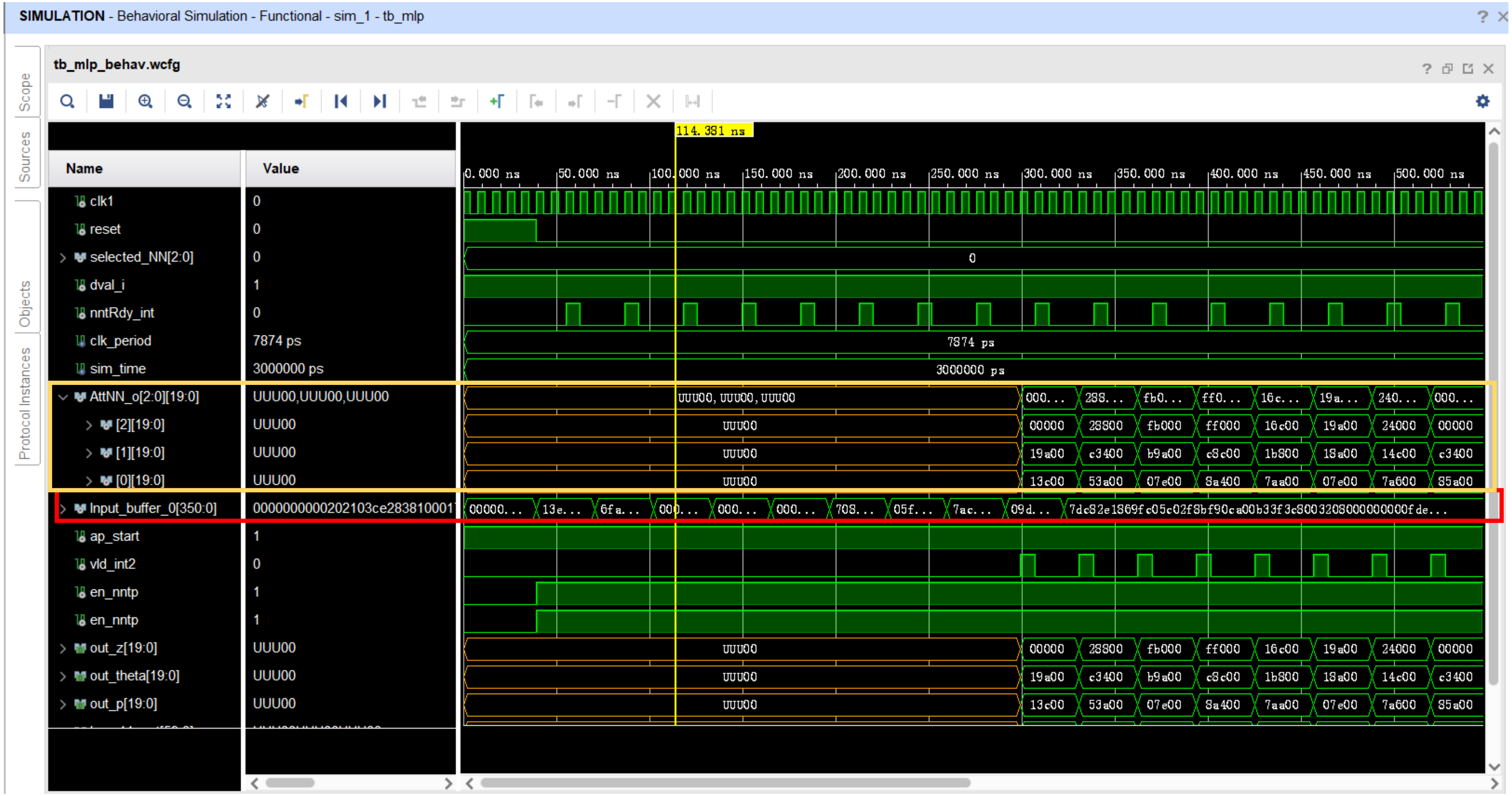
# Commission with Belle II physics

- We are working on the commission procedure for the DNN track trigger on Belle II with real 2024 physics run.
- Collecting DNN trigger output from physics events passing L1 trigger ( mostly signal)
- A peak shifted is observed, detailed debugging study is on-going



# Core Logic vivado simulation pass

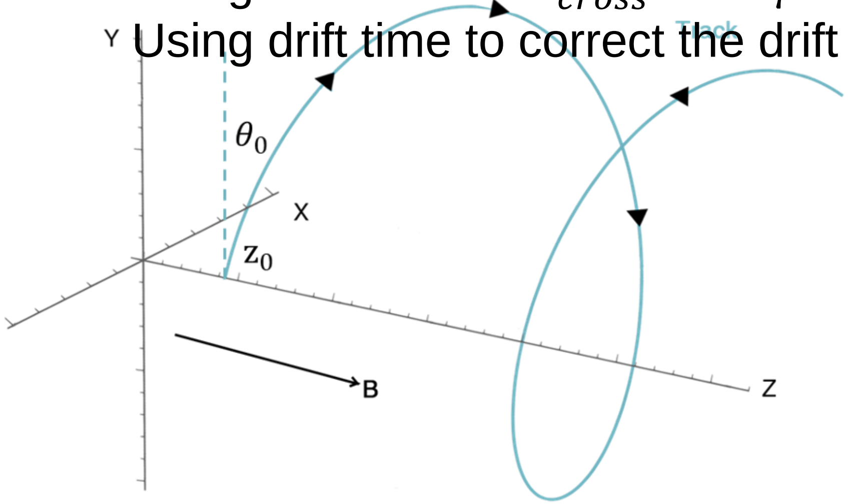


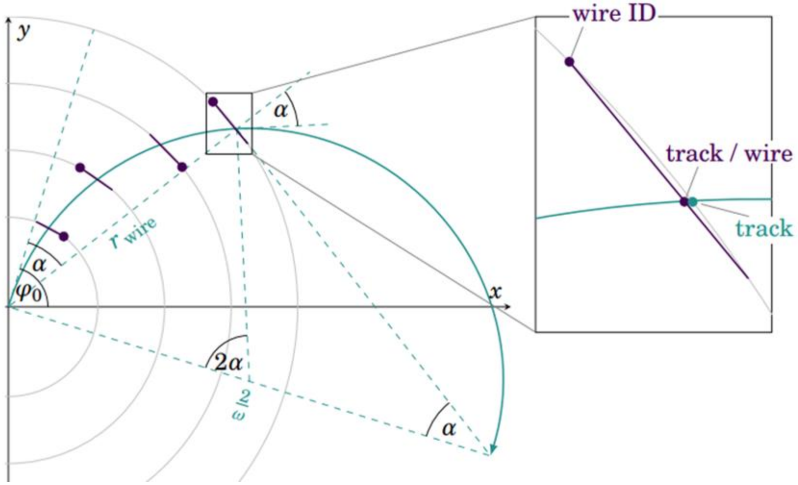

Output: after ~600ns

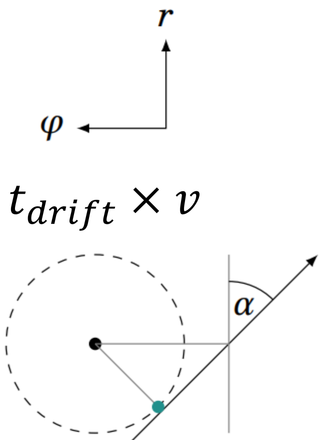
Input: every 4 clock a new input

# Introduction CDC trigger - 3D reconstruction

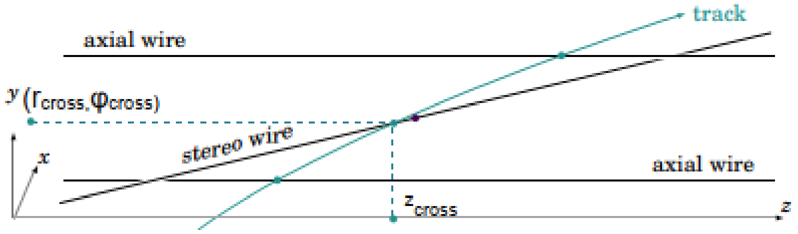
Only $\theta_0$ and $z_0$ remain unknown for 3D tracks.
With Crossing angle $\phi_{cross}$ for stereo wire we can get $z_{cross}$ .
Using two or more $z_{cross}$ with $\mu$ we can fit the linear track in $\mu - \mathrm{z}$ plane and obtain $\theta_0$ and $z_0$.
Using drift time to correct the drift distance.




Drift correction




$$\begin{pmatrix} x(\mu) \\ y(\mu) \\ z(\mu) \end{pmatrix} = \begin{pmatrix} r \cdot (\sin(\mu/r - \phi_0) + \sin\phi_0 + x_0) \\ r \cdot (\cos(\mu/r - \phi_0) - \cos\phi_0 + y_0) \\ \cot\theta_0 \cdot \mu + z_0 \end{pmatrix}$$

# Requirement for new developed NN

| Parameters | Target |
|---|---|
| $z_0$ resolution at IP ($\sigma_{95}^{IP}$) | <2 cm |
| Trigger efficiency | >95% |
| Extra background rejection rate | >50% |

- Reduce the $z_0$ resolution for signal track to less 2 cm
- Keep same efficiency as before (>95%) and restrict cut to reject further half of background events, which were kept by current trigger.

| | CDC $B\overline{B}$ bits | CDC $\tau$ & dark bits |
|---|---|---|
| Current CDC Background raw trigger rate | 2.15 kHz | 1.91 kHz |
| Required CDC Background raw trigger rate | 1.07 kHz | 0.9 kHz |

- New NN algorithm can be implemented on new universal trigger board (called UT4) ,which has about 4 times more logic gates than previous one.

# Performance evaluation – Training, validation and testing sample

Data sample generate from special physics run data taken without HLT trigger.

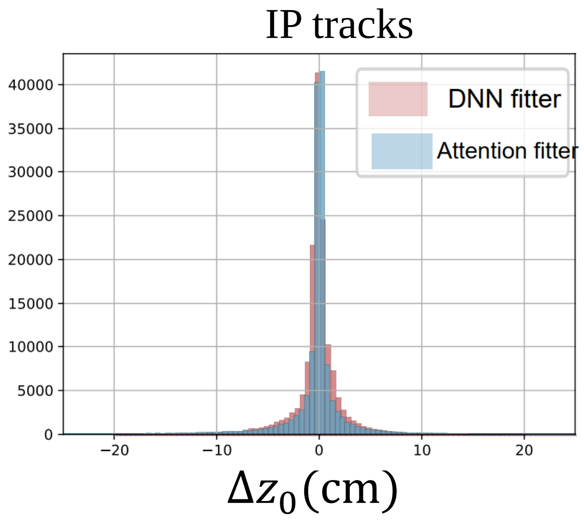
Target $z_0$ and $\theta_0$ of Tracks are got from offline reconstruction and fed for training

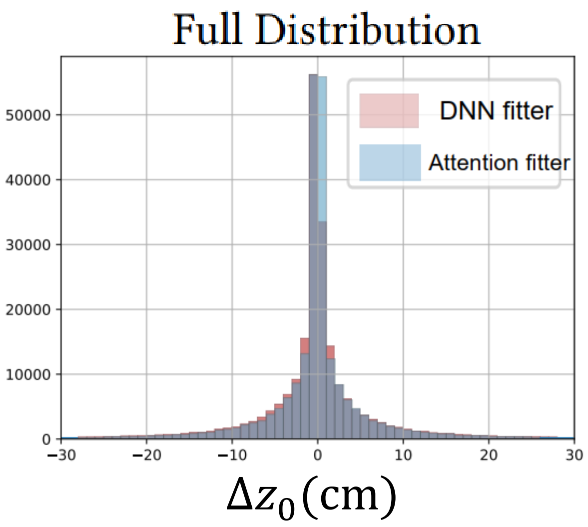
Randomly separate full sample in training validation and test:

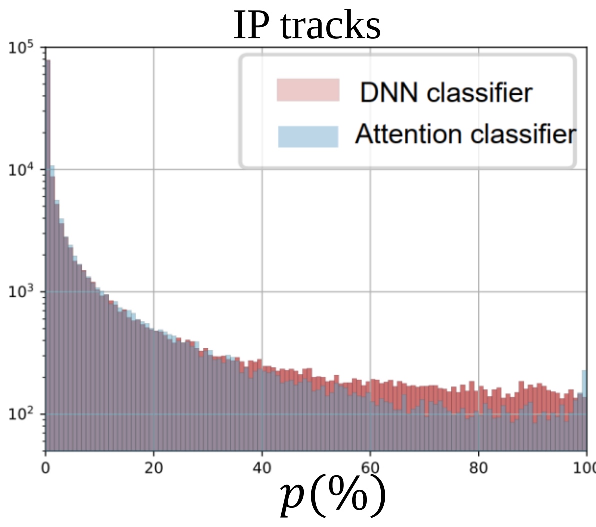
| | #Signal Tracks | # Off-IP Tracks | #Fake Tracks |
|---|---|---|---|
| Training sample | 935K | 284K | 0 |
| Validation sample | 282K | 85K | 0 |
| Test sample | 180k | 53k | 87k |

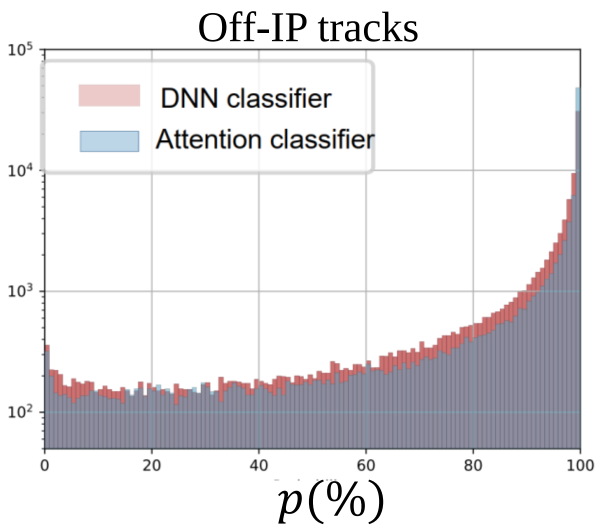
Fake tracks are only included in test sample -- No target $z_0$ and $\theta_0$

# Performance evaluation – Attention based NN




| | Cut | $\sigma_{95}^{IP}$ (cm) | signal track efficiency (%) | off-IP track reject rate(%) |
|---|---|---|---|---|
| Neurotrigger | $\lvert z_0^{NN}\rvert < 15$ | 5.53 | 93.5 | 52.0 |
| DNN fitter | $\lvert z_0^{NN}\rvert < 15$ | 2.34 | 97.5 | 56.7 |
| **Attention fitter** | $\lvert z_0^{NN}\rvert < 15$ | **1.84** | **97.8** | **59.4** |
| DNN classifier | $p < 65$ | / | 95.1 | 84.4 |
| **Attention classifier** | $p < 65$ | **/** | **96.6** | **86.2** |

6%↑

12%↑

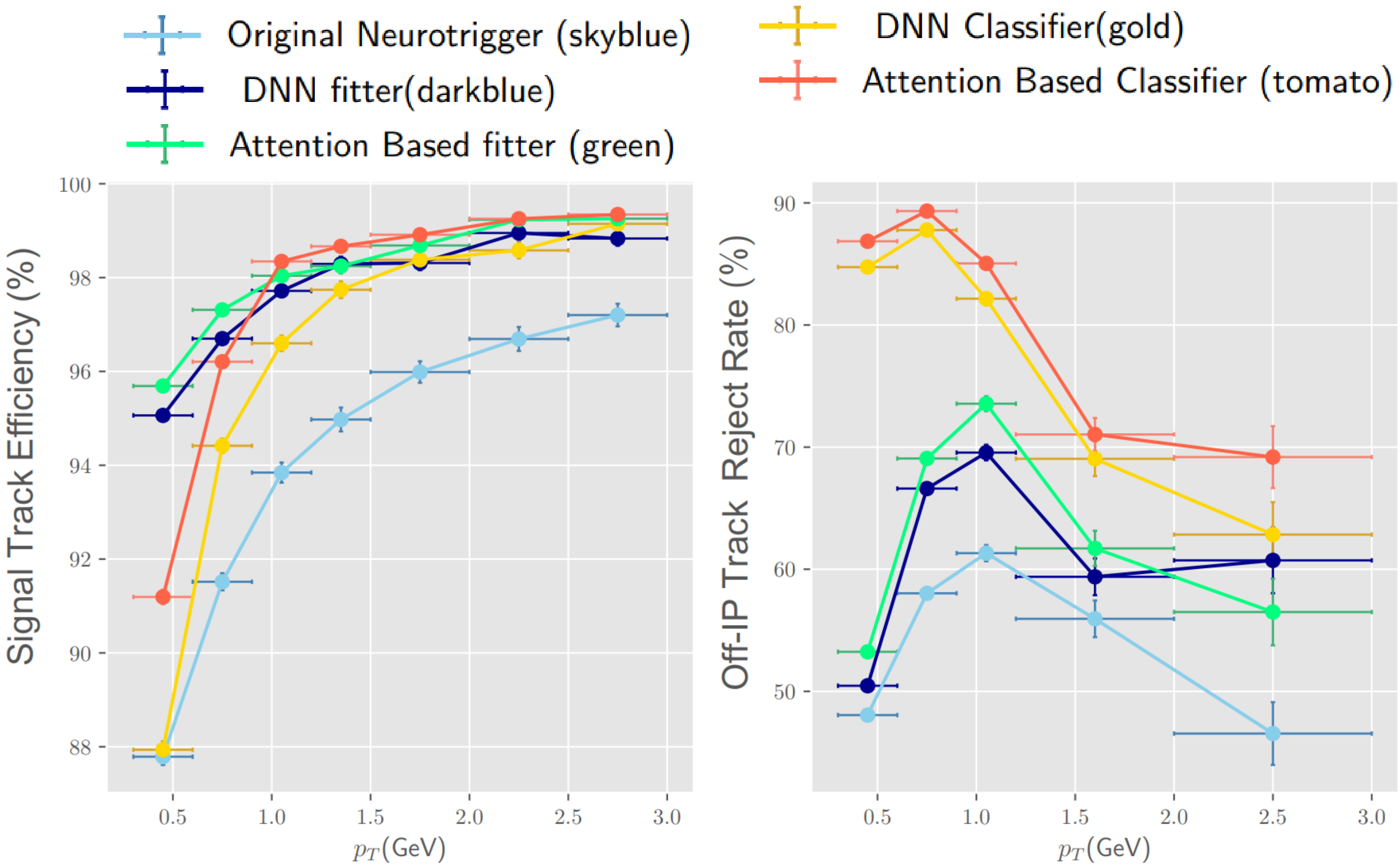
**Attention NN gain 0.5 cm IP resolution and ~12% reject rate improvement comparing with DNN**

# Performance evaluation – Transverse momentum dependency

Check the efficiency and reject rate dependency of Transverse momentum ($p_T$)

Cut: $p < 65$ OR $|z_0^{NN}| < 15$

- **All new model have better efficiency & reject rate at any $p_T$**
- **Classifiers improve low $p_T$ reject rate by 30%, while have lower efficiency comparing with fitters**



# Performance evaluation – Fake track

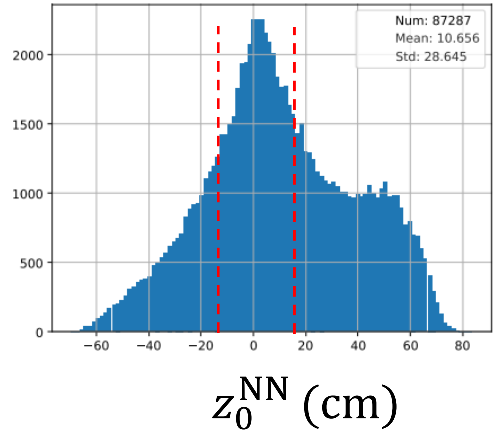
**Classifiers** can identify fake track well which mainly **concentrate at $\boldsymbol{p \sim 100}$**

For **Fitters**, Fake track have a certain $z_0^{NN}$ distribution **centering at ~0**.

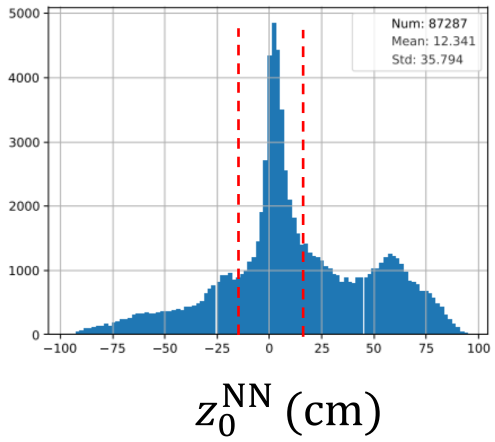
With Cut: $p < 65 \text{ OR } |z_0^{NN}| < 15$

| | Fake tracks reject rate |
|---|---|
| Original Neurotrigger | 60.4% |
| DNN fitter | 58.5% |
| Attention based fitter | 59.8% |
| DNN classifier | 68.5% |
| Attention based classifier | 66.5% |

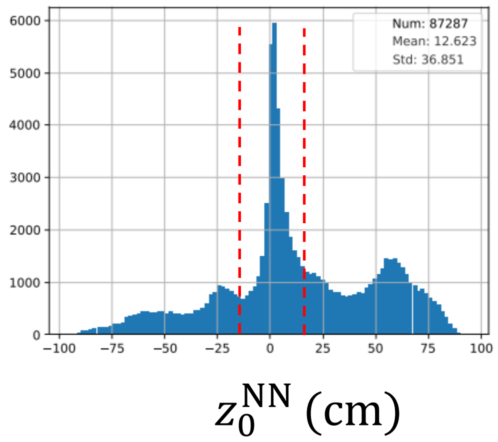
Original Neurotrigger



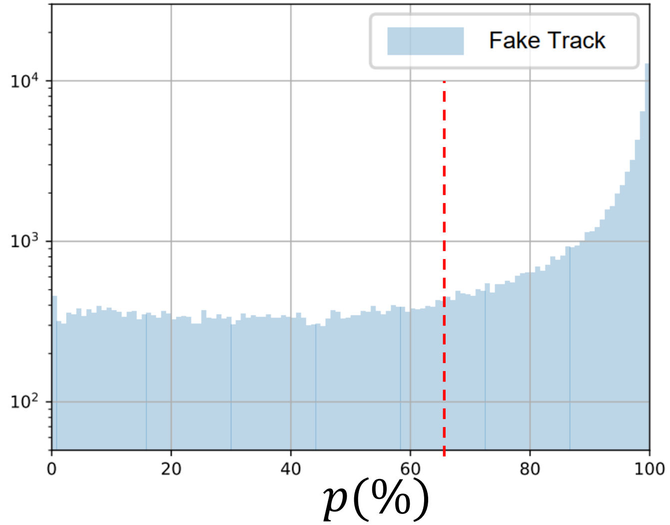

DNN fitter




Attention based fitter




DNN classifier




Attention based classifier