Firmware implementation of Phase-2 Overlap Muon Track Finder algorithm for CMS Level-1 trigger

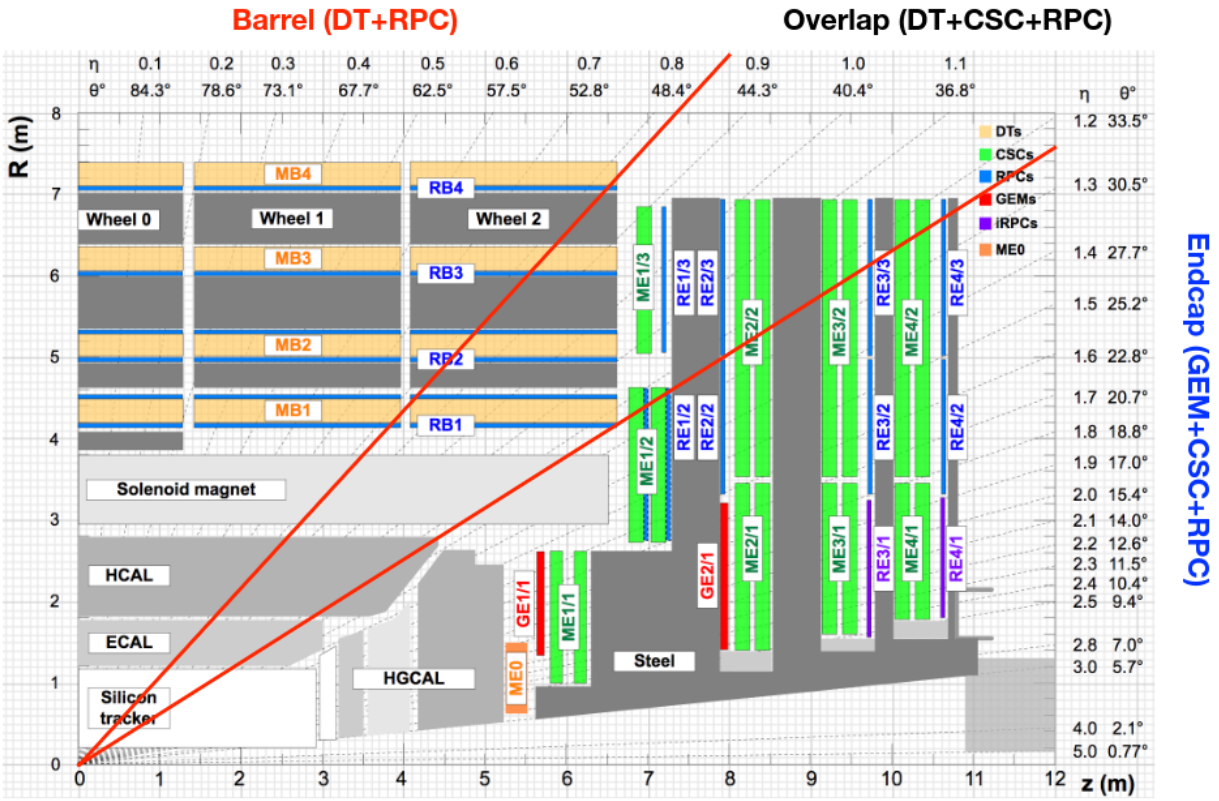
Author: Piotr Andrzej Fokow, WUT

Co-Author: Pelayo Leguina Lopez, University of Oviedo

TWEPP 2024, Glasgow

CMS Overlap Muon Track Finder (OMTF)

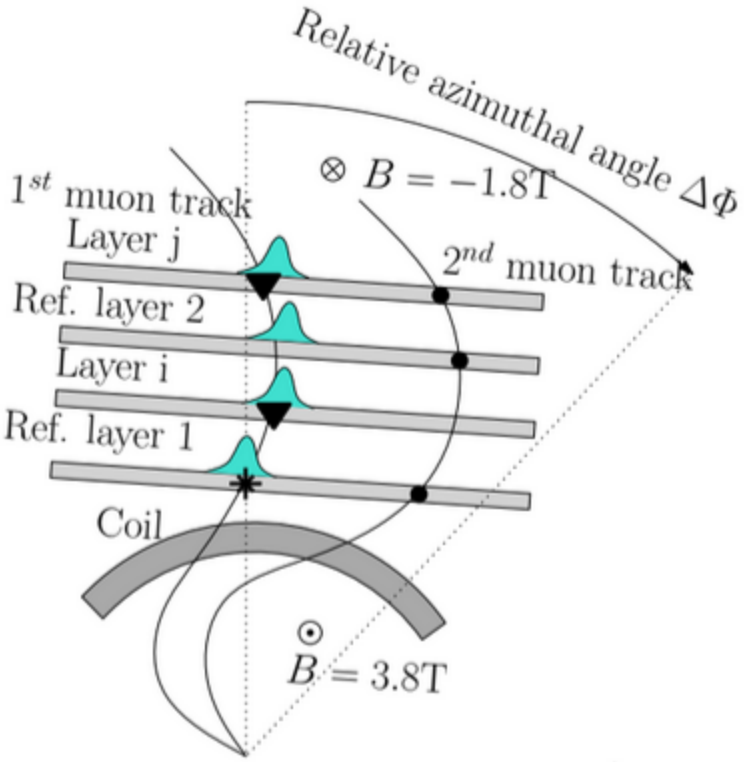
- OMTF is one of the subsystems of the CMS L1 Trigger, it was introduced for the Phase-1 CMS upgrade in 2016.
- It covers the pseudorapidity region of $0.83 < |\eta| < 1.24$
- Each processing board will cover 120 degrees of detector in r-ϕ plane with 30 degrees of overlap (6 boards in total)
- It utilizes three types of detectors: Drift Tubes (DTs) and Resistive Plate Chambers (RPCs) from Barrel and Cathode Strip Chambers (CSCs) and RPCs from Endcap region.
- OMTF identifies the muon tracks, estimates their transverse momentum and sends the found candidates (with associated chamber segments) to the Global Muon Trigger



CMS detector slice for Phase-2 CMS, Source: “The Phase-2 Upgrade of the CMS Level-1 Trigger”

Muon track reconstruction in OMTF region

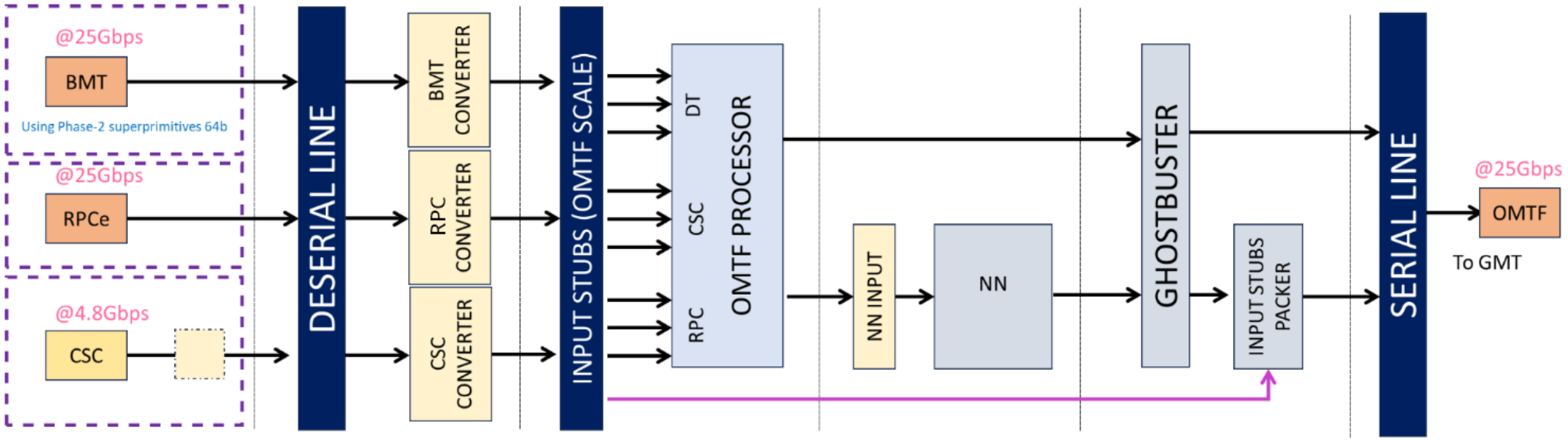
- The principle of the muon track reconstruction in the Phase-2 OMTF algorithm is the same as in Phase-1
- The muon track recognition algorithm was written in VHDL for Phase-1 and has been ported to C++ HLS for Phase-2
- Inputs: 18 layers of muon detection (DT, RPC and CSC)
- 8 of detector layers with good coverage in ϕ and η are treated as the reference layers
- The OMTF algorithm begins the muon reconstruction from a reference hit
- The reconstruction is performed using pattern matching, based on the naive Bayes classifier
- Duplicated track candidates are removed by the Ghostbuster functional block
- For Phase-2, pattern recognition algorithm introduces measurement of transverse momentum without beamspot constraint. This allows the triggering on muons coming from the decays of long-lived particles.



Source: Bluj, Michał et al. “From the Physical Model to the Electronic System - OMTF Trigger for CMS.”

OMTF Algorithm: Implementation for Phase-2

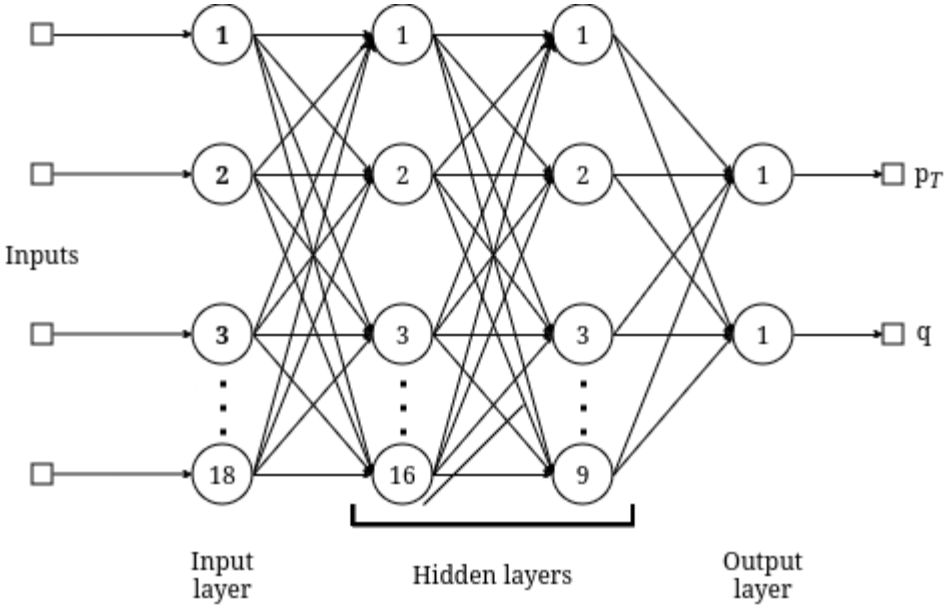
- The Bunch-Crossing (BX) event rate is 40 MHz, which determines the basic operating frequency of the system
- 15 input BMT-L1 links, 65 CSC links and 12 RPCe links, where BMT-L1 links are now implemented
- Barrel Muon Trigger Layer-1 data rate: up to 8 Trigger Primitives (TPs): 4 ϕ TPs, 4 θ TPs per BX, per link
- The input data must be converted to achieve common representation of data coming from different types of detectors.
- OMTF Processor and Neural Network (NN) clock frequency is 360 MHz
- Data output rate for Global Muon Trigger (GMT): up to 9 muon candidates per BX, which is determined by the algorithm's clock
- 18 output links to GMT
- Converters, OMTF Processor, Neural Network and Ghostbuster are implemented as IP cores using HLS

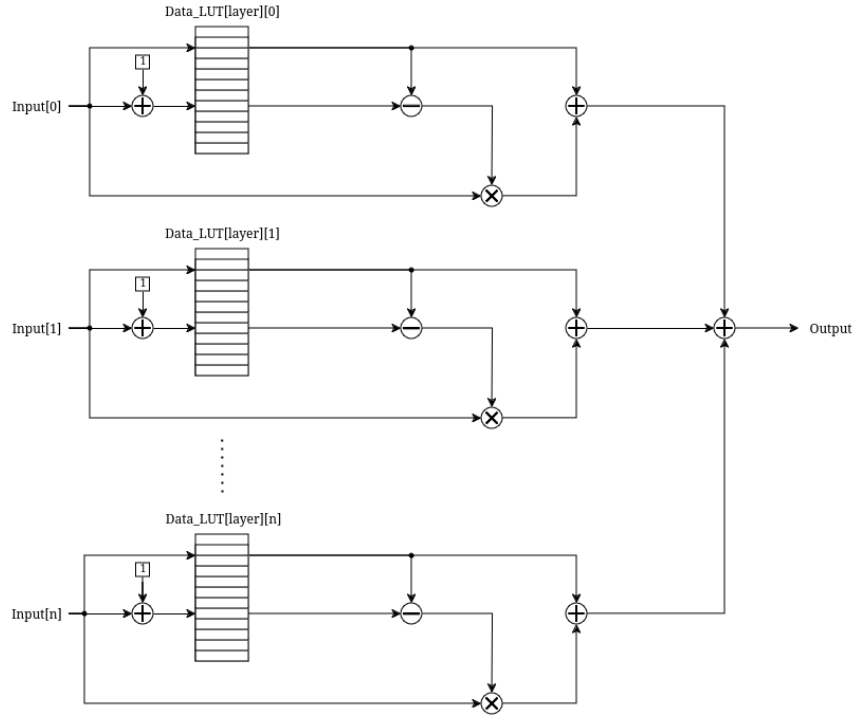


Target OMTF Algorithm diagram; Created by Pelayo Leguina, Universidad de Oviedo

NN in OMTF Algorithm for Phase-2 CMS

- Fully-Connected Neural Network is used in Phase-2 OMTF as one of functional blocks
- Estimates the p_T and charge for the muon candidate
- The NN inputs are Δϕ versus the reference hit found by pattern logic
- 2 hidden layers: 16 neurons in hidden layer 1 and 9 neurons in hidden layer 2
- 441 multiplications in total
- Multiplication operations utilizes DSP48 blocks built in Ultrascale+ FPGAs
- Weights and activation functions implemented as look-up tables stored in BRAMs, to reduce logic cell utilization

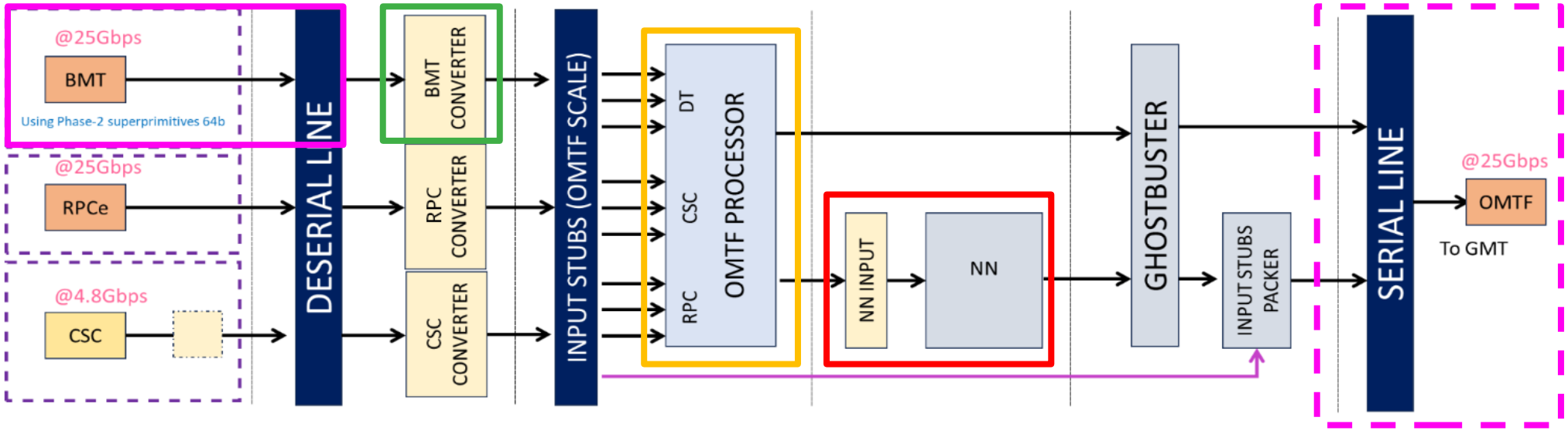




OMTF P2 Algorithm: Implementation for Phase-2

- Target platform: custom ATCA boards (X2O) with AMD Ultrascale+ FPGAs (xcvu13p-fsga2577-1-e)
- Using Blobfish custom firmware interface for link interface generation and system management
- TCL and Cmake scripts are used in algorithm integration
- The blocks marked with solid lines are fully implemented, while those marked with dashed line are partially implemented or simulated.
- Block colors: Blobfish in purple, input data converters in green OMTF Processor in yellow, NN in red

Results from September 2024	
Parameter	OMTF @360 MHz
LUT	6.3%
FF	5.3%
BRAM	20.5%
DSP	4.3%
Latency	118 clock cycles (~328 ns)
Worst Negative Slack	0.030 ns
Thermal Margin	~71 ℃

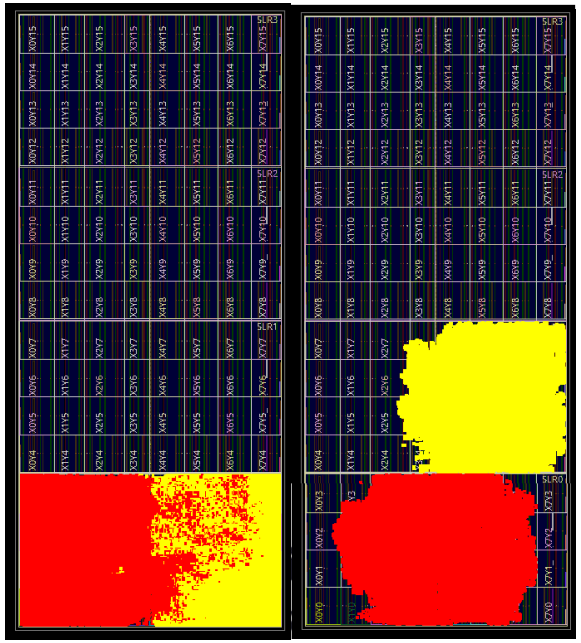


OMTF P2 Algorithm: Implementation challenges

Single SLR vs. dual SLR design

The link configuration overuses available GTY transceivers in a single SLR. Every SLR crossing comes with increased latency and resource utilization. We are expecting multiple SLR crossings for input link data. We conducted a test to get an answer to determine which negative effects might be associated with the SLR crossing. We observed higher latency and clock utilization.

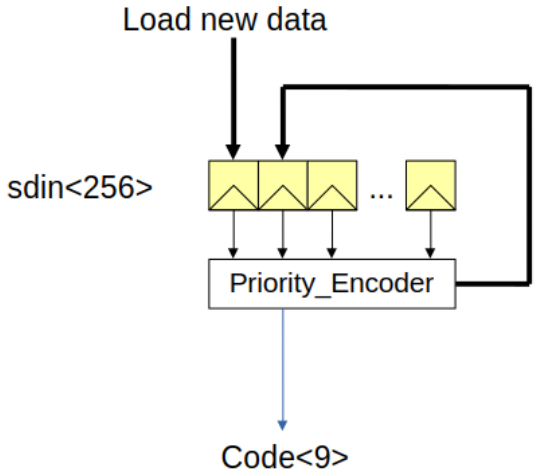
Results from one of earlier builds. Red – OMTF processor, yellow – NN



	Single SLR @240 MHz	Dual SLR @240 MHz
LUT	345983 [20.02%]	336173 [19.45%]
FF	143556 [4.15%]	145413 [4.21%]
CLB	52589 [24.35%]	60567 [28.04%]
BRAM	252 [9.38%]	252 [9.38%]
DSP	441 [3.59%]	441 [3.59%]
Excess latency for SLR crossing	0	2
Clock period after implementation [ns]	4.139 [0.027 ns]	4.112 [0.117 ns]

Single-clock read-write dependencies

Priority Encoder is one of functions inside OMTFProcessor, which provides a position to the first unique reference hit every clock cycle. Those dependencies for algorithms working at higher frequences are constraining timing performance.

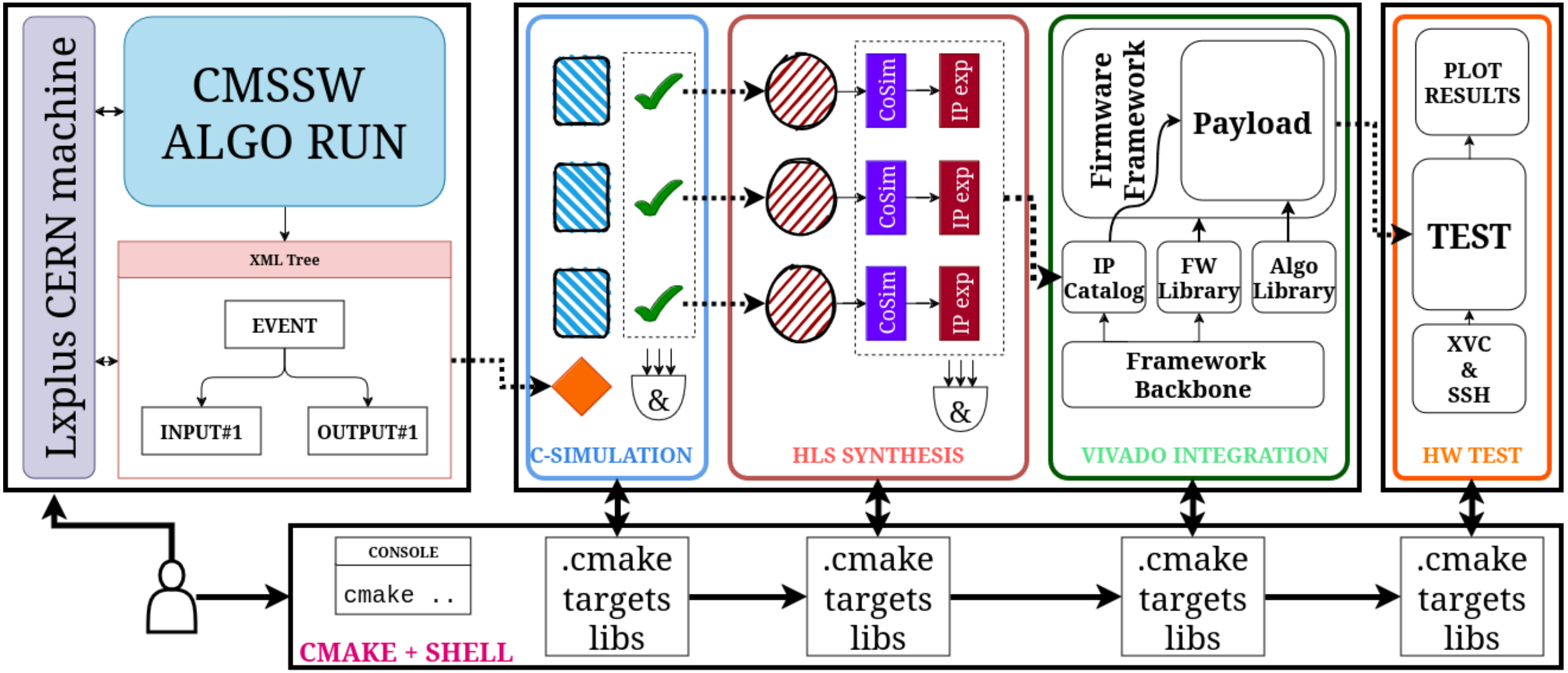


The HLS synthesis result before and after refactorization

Modules & Loops	Issue Type	Violation Type	Distance	Slack	Latency(cycles)	Latency(ns)	Iteration Latency	Interval	Trip Count	Pipelined	BRAM	DSP	FF	LUT
priorityEncoder	Timing Violation			-0.32	0	0.0	-	1	-	yes	0	0	257	1505

Modules & Loops	Issue Type	Violation Type	Distance	Slack	Latency(cycles)	Latency(ns)	Iteration Latency	Interval	Trip Count	Pipelined	BRAM	DSP	FF	LUT
priorityEncoder_v3	Timing Violation			-0.05	0	0.0	-	1	-	yes	0	0	513	34917

Design and verification workflow



Design and verification flow for OMTF algorithm; Created by Pelayo Leguina, Universidad de Oviedo

Verification results

OMTF Processor and Neural Network simulation results, compared with CMSSW outputs prove, that algorithm handles feeding data properly

OMTF Processor C simultaion

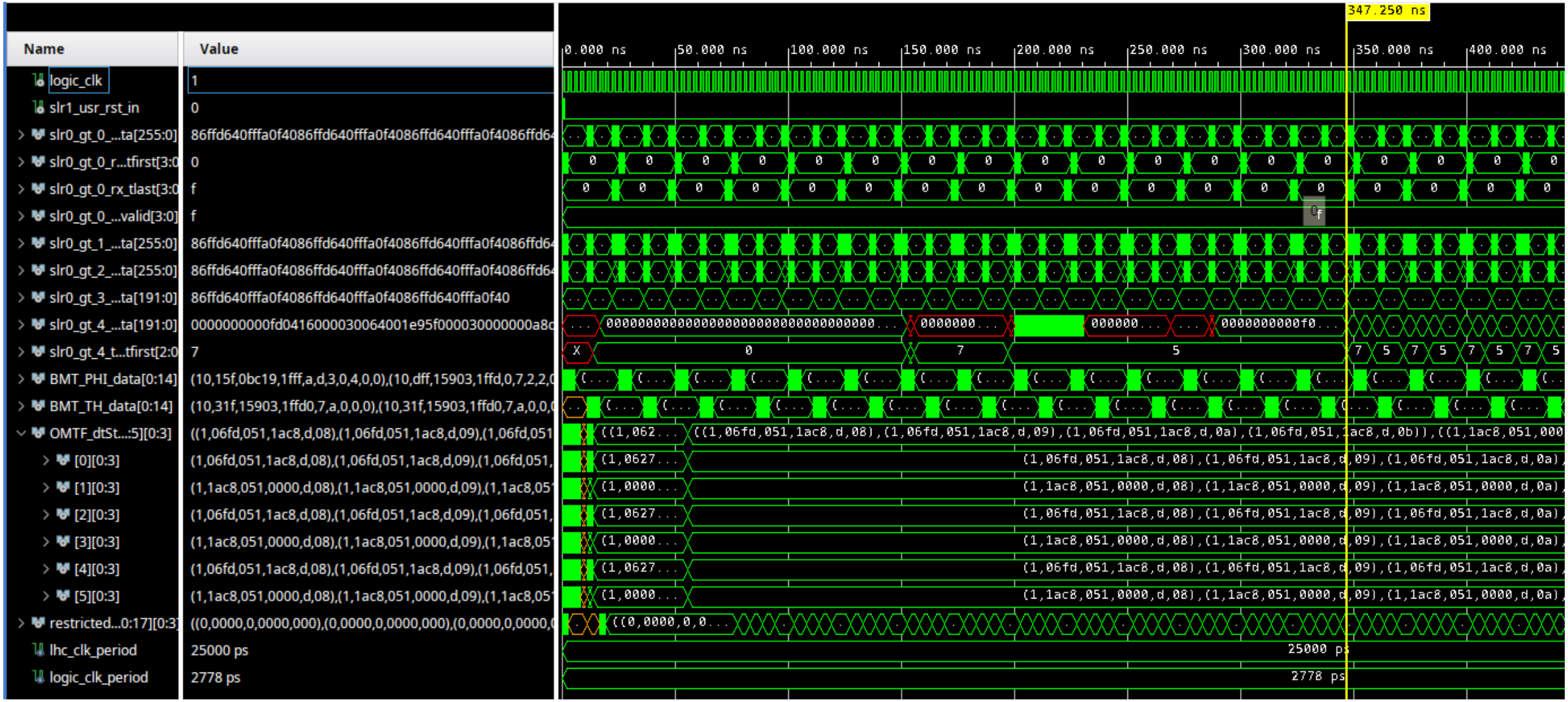
```
Processor Output - bx: 1994, ref_hit_nr: 60, reflayer: 2
Best Restricted Stubs:
  Layer: 0 - Active: 1, Phi: 935, Eta: 92, Quality: 6, DistPhi Phi: 28
  Layer: 1 - Active: 1, Phi: -215, Eta: 92, Quality: 6, DistPhi Phi: -215
  Layer: 2 - Active: 1, Phi: 907, Eta: 79, Quality: 6, DistPhi Phi: 0
  Layer: 3 - Active: 1, Phi: -182, Eta: 79, Quality: 6, DistPhi Phi: -182
  Layer: 4 - Active: 1, Phi: 883, Eta: 69, Quality: 6, DistPhi Phi: -24
  Layer: 5 - Active: 1, Phi: -236, Eta: 69, Quality: 6, DistPhi Phi: -236
  Layer: 10 - Active: 1, Phi: 944, Eta: 86, Quality: 2, DistPhi Phi: 37
  Layer: 11 - Active: 1, Phi: 926, Eta: 81, Quality: 1, DistPhi Phi: 19
  Layer: 13 - Active: 1, Phi: 903, Eta: 78, Quality: 2, DistPhi Phi: -4
GP Out Constrained - valid_out: 1, best_pat_64: 9, pdfSum: 680, fired_cnt: 7
GP Out Unconstrained - valid_out: 1, best_pat_64: 63, pdfSumUnconstr: 791, fired_cnt: 7
<bestStubs reflayer="2" bestPat="5">
  <bestStub layer="0" input="4" eta="92" phi="935" quality="6" phiDist="28" valid="1"/>
  <bestStub layer="1" input="4" eta="92" phi="-215" quality="6" phiDist="-215" valid="0"/>
  <bestStub layer="2" input="6" eta="79" phi="907" quality="6" phiDist="0" valid="1"/>
  <bestStub layer="3" input="6" eta="79" phi="-182" quality="6" phiDist="-182" valid="1"/>
  <bestStub layer="4" input="4" eta="69" phi="883" quality="6" phiDist="-24" valid="1"/>
  <bestStub layer="5" input="4" eta="69" phi="-236" quality="6" phiDist="-236" valid="0"/>
  <bestStub layer="10" input="8" eta="86" phi="944" quality="2" phiDist="37" valid="1"/>
  <bestStub layer="11" input="8" eta="81" phi="926" quality="1" phiDist="19" valid="1"/>
  <bestStub layer="13" input="12" eta="78" phi="903" quality="2" phiDist="-4" valid="1"/>
  <gpResultConstr patNum="9" pdfSum="680">
<gpResultUnconstr patNum="63" pdfSum="791">
```

Neural Network C simultaion

```
calculated sign: -0.9375    expected sign: -0.9375 result: passed
calculated pT: 88.3594  expected pT: 88.3594    result: passed
calculated sign: -0.75  expected sign: -0.75    result: passed
calculated pT: 87.9688  expected pT: 87.9688    result: passed
calculated sign: -1 expected sign: -1   result: passed
calculated pT: 4.625    expected pT: 4.625  result: passed
calculated sign: -1 expected sign: -1   result: passed
calculated pT: 75.8438  expected pT: 75.8438    result: passed
calculated sign: -1 expected sign: -1   result: passed
calculated pT: 60.1172  expected pT: 60.1172    result: passed
calculated sign: -1 expected sign: -1   result: passed
calculated pT: 65.1563  expected pT: 65.1563    result: passed
calculated sign: -1 expected sign: -1   result: passed
calculated pT: 53.7188  expected pT: 53.7188    result: passed
calculated sign: -1 expected sign: -1   result: passed
calculated pT: 36.7891  expected pT: 36.7891    result: passed
calculated sign: -1 expected sign: -1   result: passed
calculated pT: 52.0938  expected pT: 52.0938    result: passed
calculated sign: -0.4375    expected sign: -0.4375 result: passed
calculated pT: 44.5313  expected pT: 44.5313    result: passed
calculated sign: -1 expected sign: -1   result: passed
calculated pT: 8.47656  expected pT: 8.47656    result: passed
calculated sign: -1 expected sign: -1   result: passed
calculated pT: 12.5 expected pT: 12.5   result: passed
calculated sign: -1 expected sign: -1   result: passed
calculated pT: 10.4297  expected pT: 10.4297    result: passed
calculated sign: -1 expected sign: -1   result: passed
calculated pT: 10.9453  expected pT: 10.9453    result: passed
calculated sign: -1 expected sign: -1   result: passed
calculated pT: 9.47656  expected pT: 9.47656    result: passed
calculated sign: -1 expected sign: -1   result: passed
calculated pT: 9.32813  expected pT: 9.32813    result: passed
INFO: [COSIM-1000] *** C/RTL co-simulation finished: PASS ***
```

Verification results

Simulation of integrated algorithm results in proper data propagation



Conclusions

- As for now, the OMTF Processor is ported from Phase-1 and prepared for Phase-2. Neural Network and BMT-L1 input processing blocks are implemented.
- OMTF Processor has been extended to include muon displacement calculation for long-lived particle triggering.
- Neural Network is introduced to improve muon's p_T and charge calculation. Might require further optimization to reduce resource usage, as new calculations eg. beam-spot-unconstrained p_T will be implemented.
- Current design fits into one SLR. We expect to use multiple SLRs due to high input link and BRAM utilization, as well as functional blocks yet to be implemented.
- HLS verification of IP blocks is successful. Verification of integrated algorithm with small input data sampes shows, that data are propagated properly.
- The optimal algorithm's logic placement will be determined in further steps.

Acknowledgements:

The Ministry of Science and Higher Education funds the Poland's participation in Compact Muon Solenoid Experiment

Ministry of Science and Higher Education
Republic of Poland

The OMTF algorithm is developed by a group of research centers:

NATIONAL CENTRE FOR NUCLEAR RESEARCH ŚWIERK

Universidad de Oviedo
Universidá d'Uviéu
University of Oviedo

UNIVERSITY OF WARSAW

POLITECHNIKA WARSZAWSKA
WARSAW UNIVERSITY OF TECHNOLOGY

The work is financed within a project 2021/43/B/ST2/01552 by the National Science Center

NATIONAL SCIENCE CENTRE
POLAND

Bibliography

[1] - CMS collaboration, The Phase-2 Upgrade of the CMS Level-1 Trigger, Tech. Rep. CMS-TDR-021, CERN, Geneva (2020)

[2] - Bluj, Michał et al. “From the Physical Model to the Electronic System - OMTF Trigger for CMS.” *Acta Physica Polonica. B, Proceedings Supplement* 9.2 (2016): 181–188.

[3] - Bunkowski, K. “The Algorithm of the CMS Level-1 Overlap Muon Track Finder Trigger.” *Nuclear instruments & methods in physics research. Section A, Accelerators, spectrometers, detectors and associated equipment* 936 (2019): 368–369.

[4] - Głażewska, Marianna, and Marcin Konecki. “Level-1 Muon Triggers for the CMS Experiment at the HL-LHC.” *Proceedings of Science* 414 (2022): Proceedings of Science, 2022, Vol.414, Article 1219.

[5] - Zabolotny, W. M., and A. Byszuk. “Algorithm and Implementation of Muon Trigger and Data Transmission System for Barrel-Endcap Overlap Region of the CMS Detector.” *Journal of instrumentation* 11.3 (2016): C03004.

[6] - Petersen, Philipp, and Felix Voigtlaender. “Equivalence of Approximation by Convolutional Neural Networks and Fully-Connected Networks.” *arXiv.org* (2021): arXiv.org, 2021-01.

Thank you for your attention