The CMS Barrel Muon Trigger Layer-1 Processor

5.10.2023

3rd CERN System-on-Chip Workshop

CERN

Ioannis Bestintzanos, BMTL-1 team

Phase 2 Barrel Muon Trigger

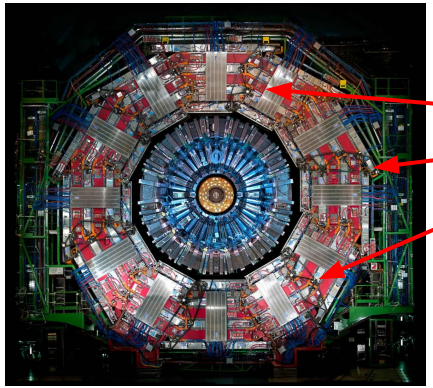
Barrel Muon Trigger (BMT) reconstructs muons of the CMS barrel

DETECTOR → **BMTL-1** → **GMT**

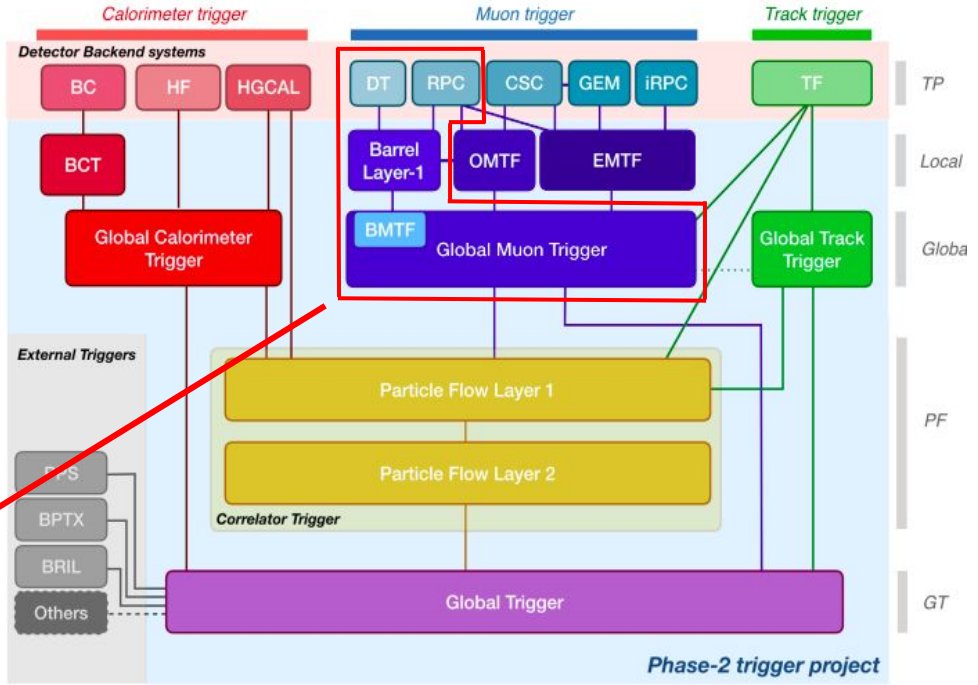
Drift Tubes (DTs) and Resistive Plate Chambers (RPCs) transmit muon hit data to BMTL-1

Builds DTs track segments and clusters RPC hits Merges both subsystem information into "super-primitives"

Matches track segments to reconstruct standalone muon objects

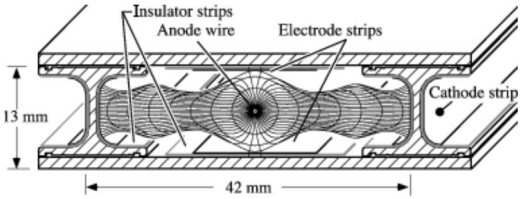


Barrel Muon Trigger



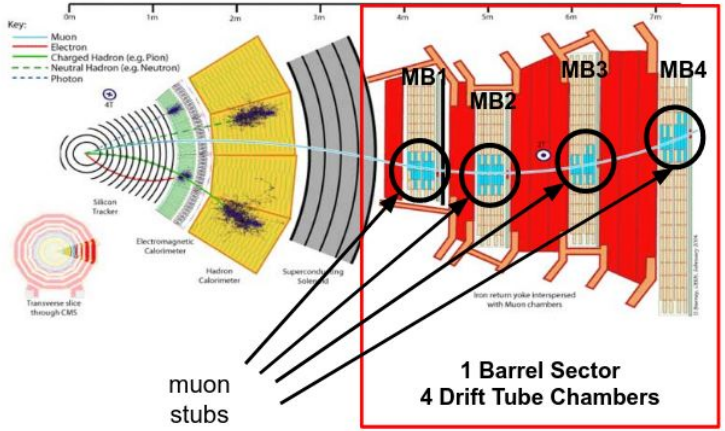
Barrel Muon Trigger

- On detector Board for Drift Tubes (OBDT) transmits detector data to BMTL-1
  - Time digitization of DT signals
- BMTL-1 processes hit information to produce Muon stubs (track segments)
  - Analytical Method (AM) algorithm processes TDC hits and generates Trigger Primitives
    - Bunch crossing, Stub Position, Bending Angle
- Stubs of the 4 chambers are received by GMT
  - Kalman Muon Track Finder (KMTF) algorithm matches tracks and reconstructs muon candidates
    - Assigns position and momentum



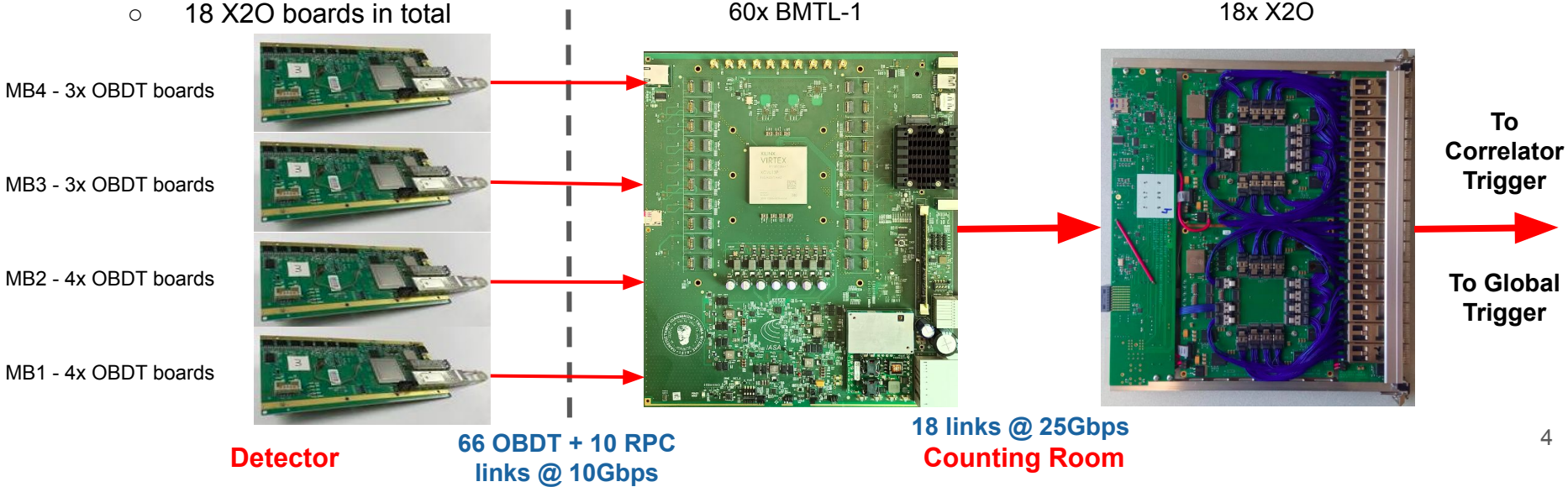
Drift Tube Cell

Detector		BMTL1 AM algorithm	GMT KMTF algorithm
Drift Tube Barrel Sector	Hits on Chamber 4	Stubs of Chamber 4	Reconstructed Muons of Sector
	Hits on Chamber 3	Stubs of Chamber 3	
	Hits on Chamber 2	Stubs of Chamber 2	
	Hits on Chamber 1	Stubs of Chamber 1	

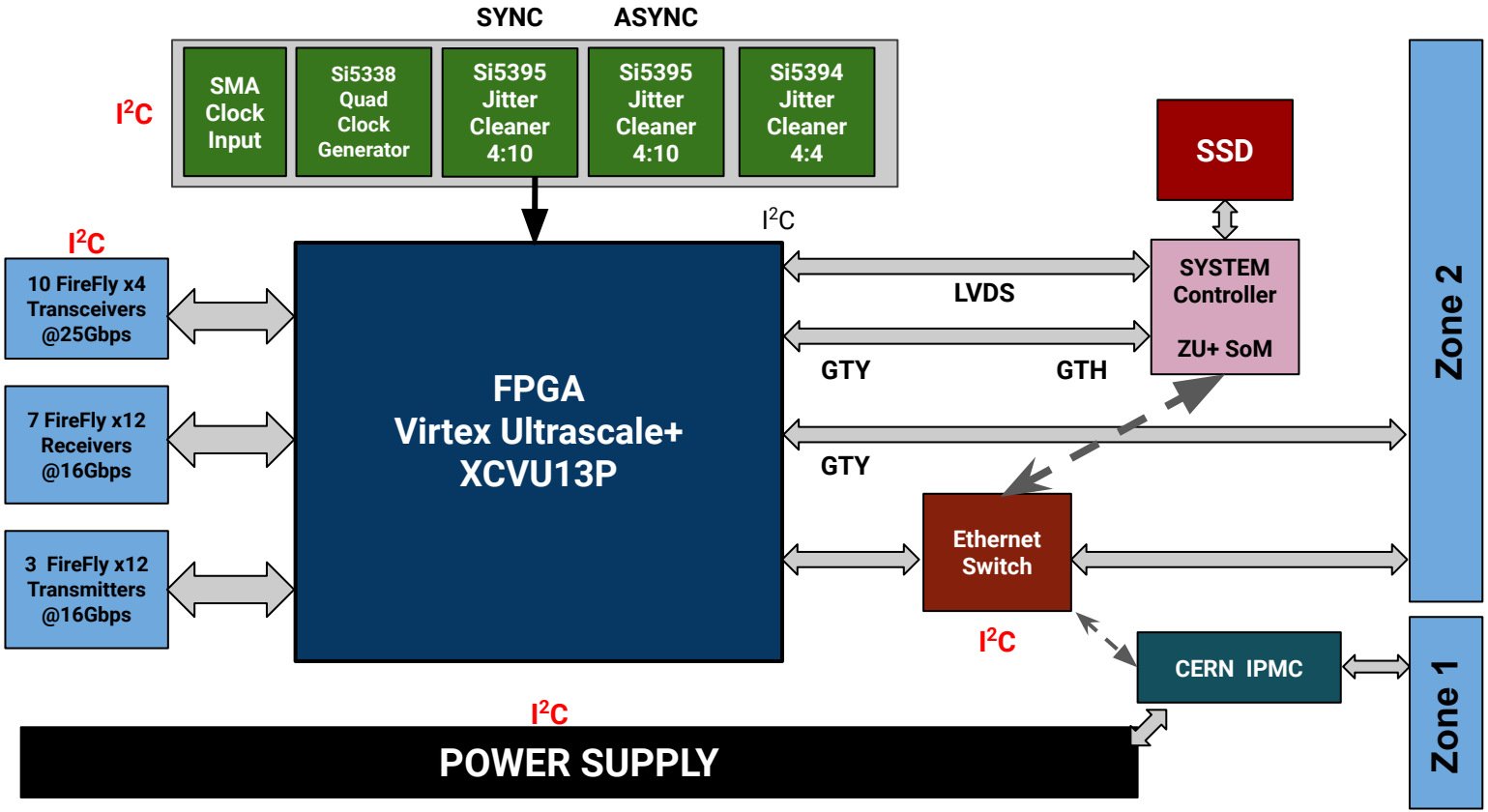


Barrel Muon Trigger

- CMS Barrel Muon system consists of 60 DT sectors
  - Each sector consists of 4 DT chambers
- OBDT boards transmit TDC hits to BMTL-1
  - 14 OBDT boards per sector
- Every BMTL-1 board processes data from 2 Sectors
  - 30 BMTL-1 boards needed for the whole barrel
- GMT receives primitives from BMTL-1
  - Using TMT18
  - 18 X2O boards in total



BMTL-1 ATCA board architecture



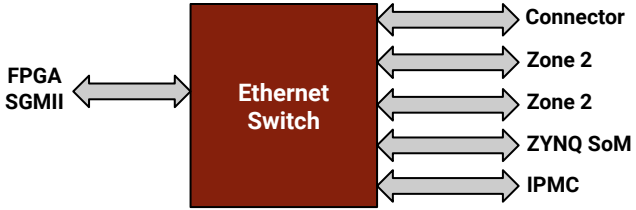
FPGA / Connectivity

- **Xilinx XCVU13P FPGA**
  - *Ultrascale+ architecture*
  - *16 nm lithography*
  - *4 SLRs (Super Logic Regions)*
  - *1,728,000 LUTs*
  - *3,456,000 Flip Flops*
  - *12,288 DSP slices*
  - *128 GTY transceivers @ 28G*
  - *-1 speed grade*

XCVU13P FPGA mounted on the BMTL-1 ATCA board

Optical connectivity

- **40 RX @ 25 Gbps** ⇨ *10 x BiDir FireFly Modules 25G*
- **40 TX @ 25 Gbps**
- **80 RX @ 16 Gbps** ⇨ *7 x Rx FireFly Modules 16G*
- **36 TX @ 16 Gbps** *3 x Tx FireFly Modules 16G*

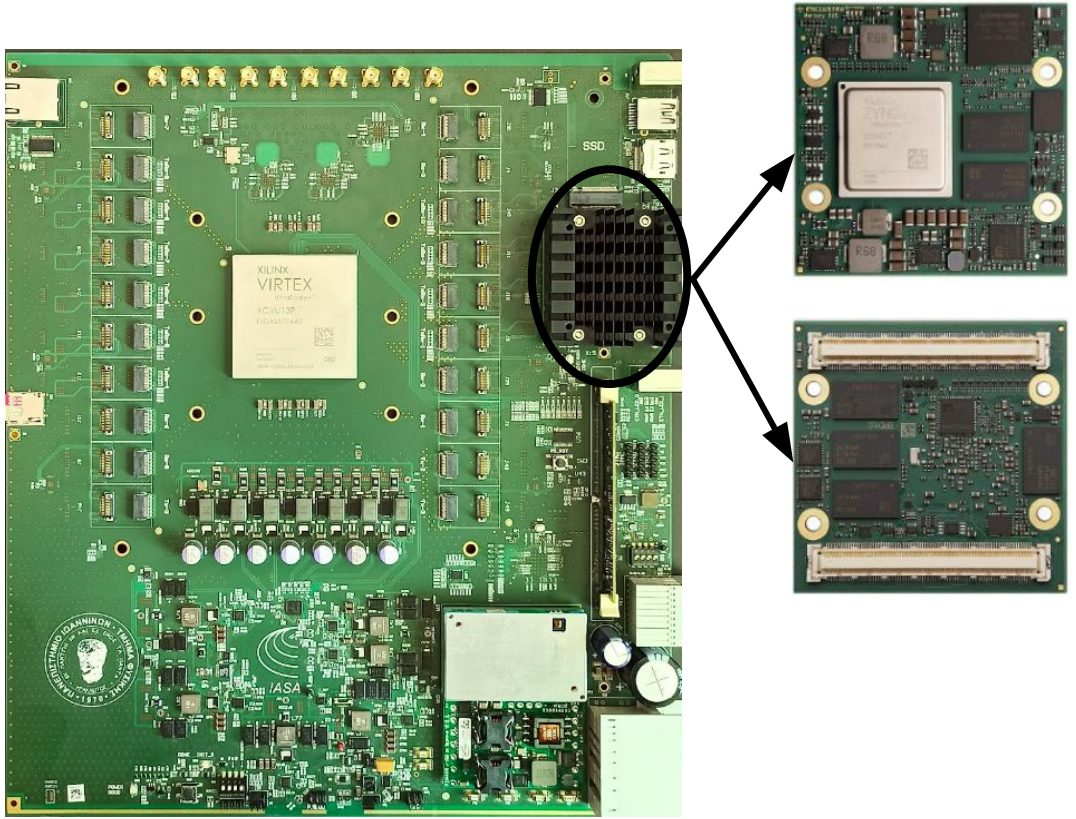


Onboard Ethernet switch circuitry provides access to many subsystems on board

- **120** RX + **76** TX ~ **3.9** ***Tbps***

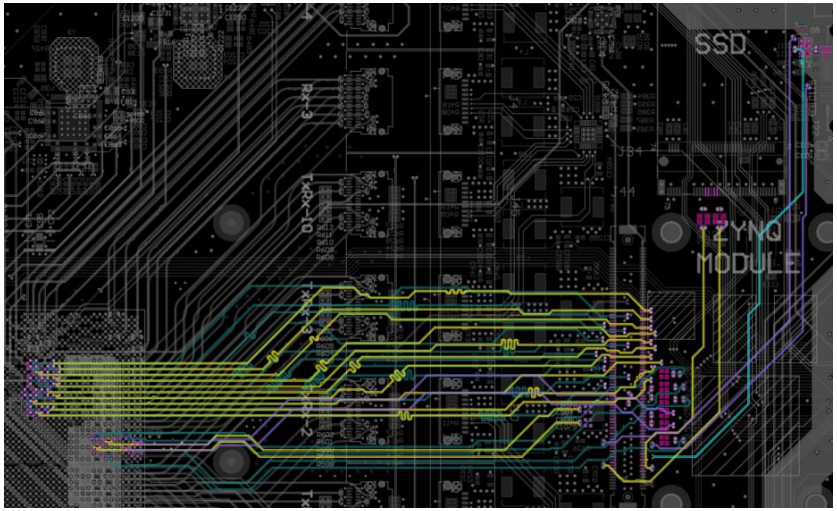
System Controller

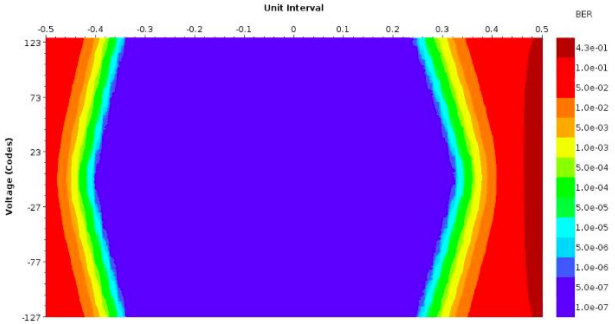
- Enclustra Mercury XU5
  - *XCZU5EV-1FSVC784E SoC*
  - *117k LUTs*
  - *234k Flip Flops*
  - *1248 DSP slices*
- Zynq Ultrascale+ architecture
- Commercially available
- Quad core Arm Cortex A-53
  - *Capable of running Linux*
- Independent RAM on PS & PL
  - *2 GB (PS)*
  - *1 GB (PL)*
- On module storage
  - eMMC flash
  - QSPI
- Gigabit Ethernet (2x)
  - Processor system
  - Programmable logic

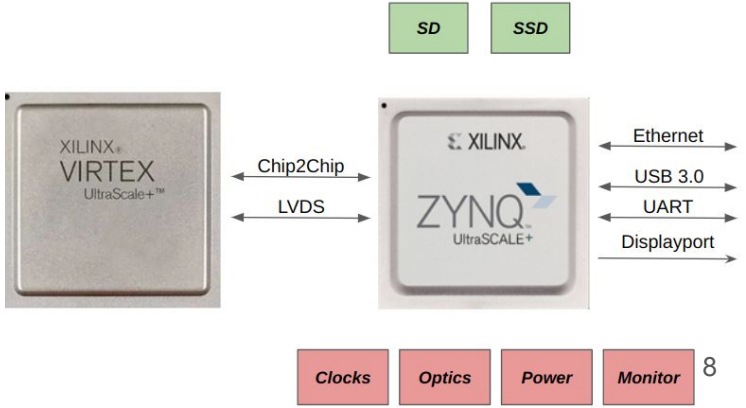


System Controller Connectivity

- Direct connections with the FPGA
  - 4 GTH MGTs
    - 10Gbps
    - 1 MGT dedicated to Chip2Chip
  - 20 LVDS pairs
    - Length matched
    - Suitable for high speed data transfer
- On board storage
  - Solid State Drive
  - SD card
- I2C
- Ethernet
- UART (front panel)
- USB 3.0
- Displayport



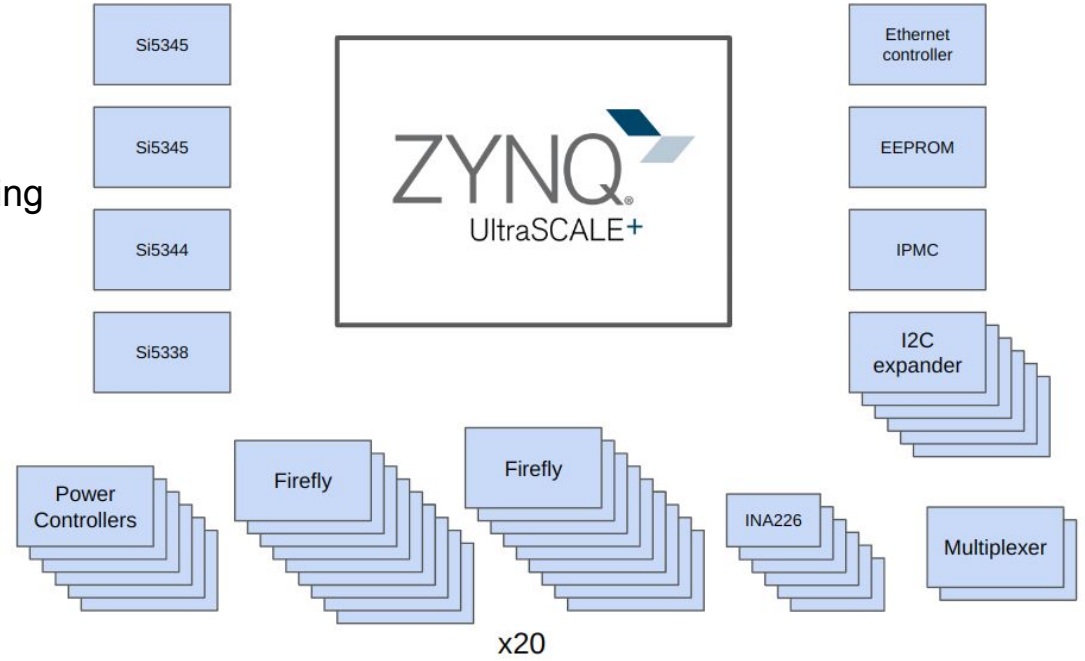




Why use a SoC

- Many peripherals on board
  - I2C, PMBus
    - Clocks
    - Power controllers
    - Temperature monitoring
    - Optics configuration/monitoring
    - Ethernet switch
- Remote board access
  - SSH through network
- Need to program the FPGA
  - JTAG, Serial
  - Debugging
- High speed interconnect with the FPGA
  - AXI Memory Mapped bus
- Real time payload monitoring

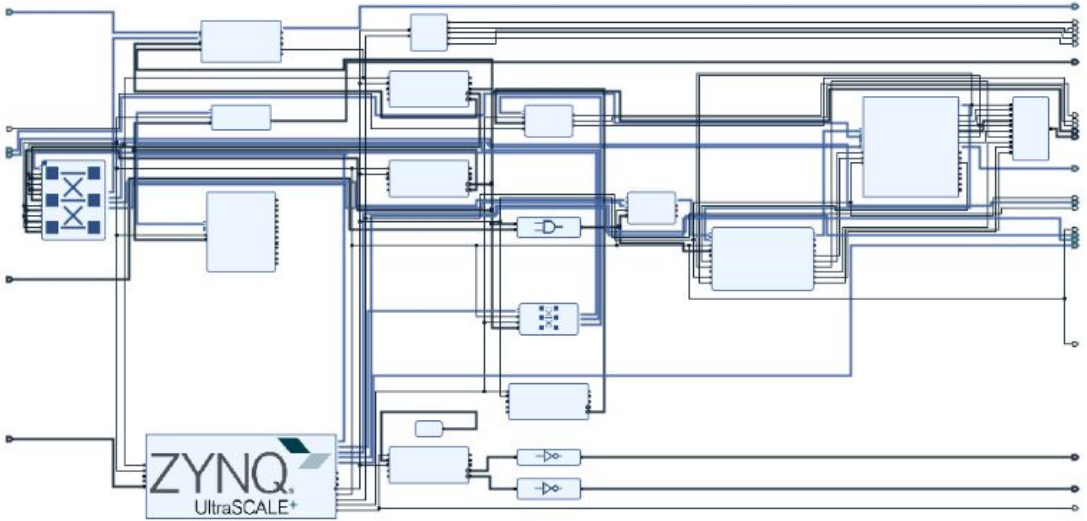
I2C based interfaces

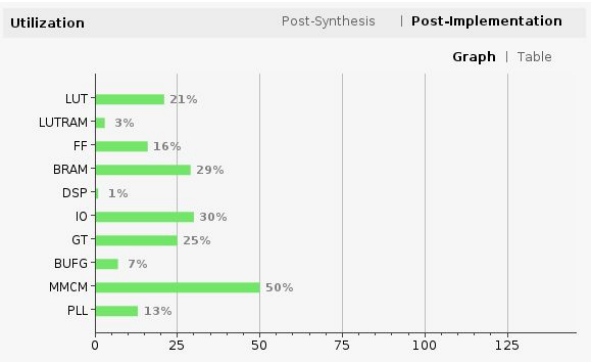


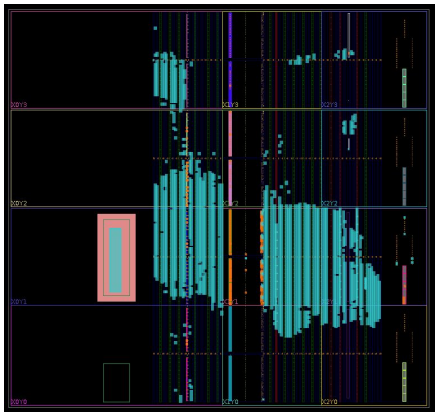
Firmware

Firmware components

- AXI chip2chip
  - ZYNQ-FPGA
- Aurora 64b66b
- AXI firewall
- Debug bridge (XVC)
  - FPGA programming
  - Debug
- DDR4 MIG
  - PL memory
- AXI interconnect

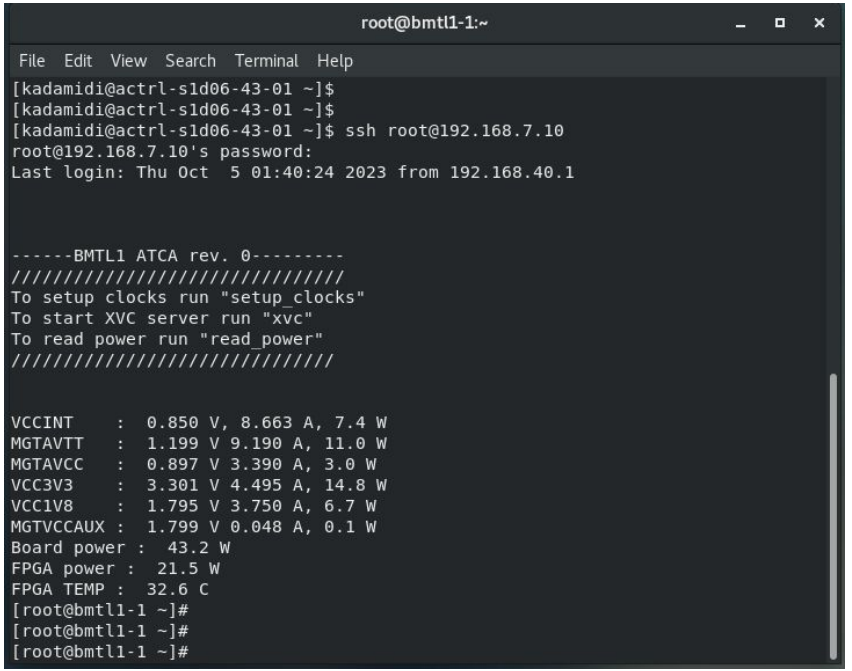






Software

- CentOS operating system
  - Runs on arm cores
- Configuration and monitoring of the on-board peripherals using Python scripts
  - Clock synthesizers
  - Firefly modules
  - Power modules
- Configuration and monitoring of FPGA using IPbus Software
  - EMPbutler SW to control the firmware framework
  - “DTUP” SW to control the algorithm
- Xilinx Virtual Cable server
  - JTAG access to VU13P
  - Programming, ILAs, etc.



Next steps

- **BMTL1 design is under revision**
  - Need arises mainly from parts becoming obsolete
  - Changes in clocks, power supplies, etc.
  - Halogen free material

- **Move to Xilinx Kria SoM**
  - Common across CMS trigger
    - Reduce development effort
  - Lower cost

- **Move configuration commands to C++**
  - Deploy HERD server & plugin
    - High level control of many boards

Summary

- **BMTL1 ATCA board was designed, produced & tested**
- **ZYNQ SoM is used as the system controller**
- **Firmware & software developed to cover requirements**
- **Current setup has been proven successful in pp collisions slice test**

Thank you