FPGA-Based Architectures & DAQ Pathways for Distributed ML Systems

- M.A. Ibrahim , J.R. Berlioz

- EPICS Collaboration Meeting in April 2023
- April 26- 28, 2023

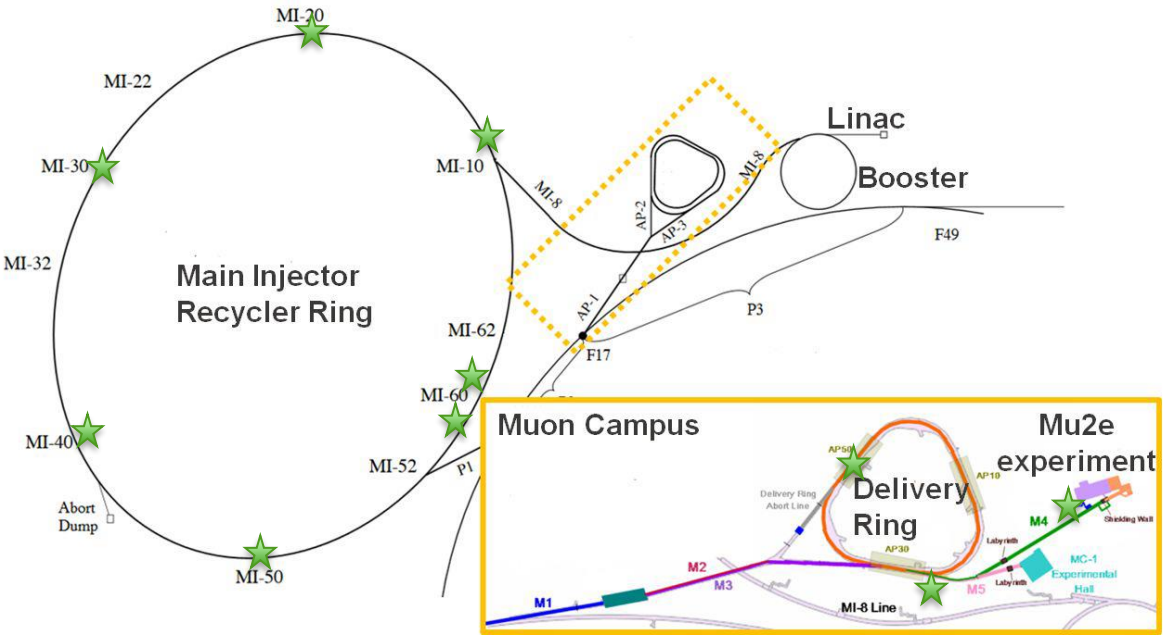
In partnership with:

Overview of AL/ML Activitites/ Projects

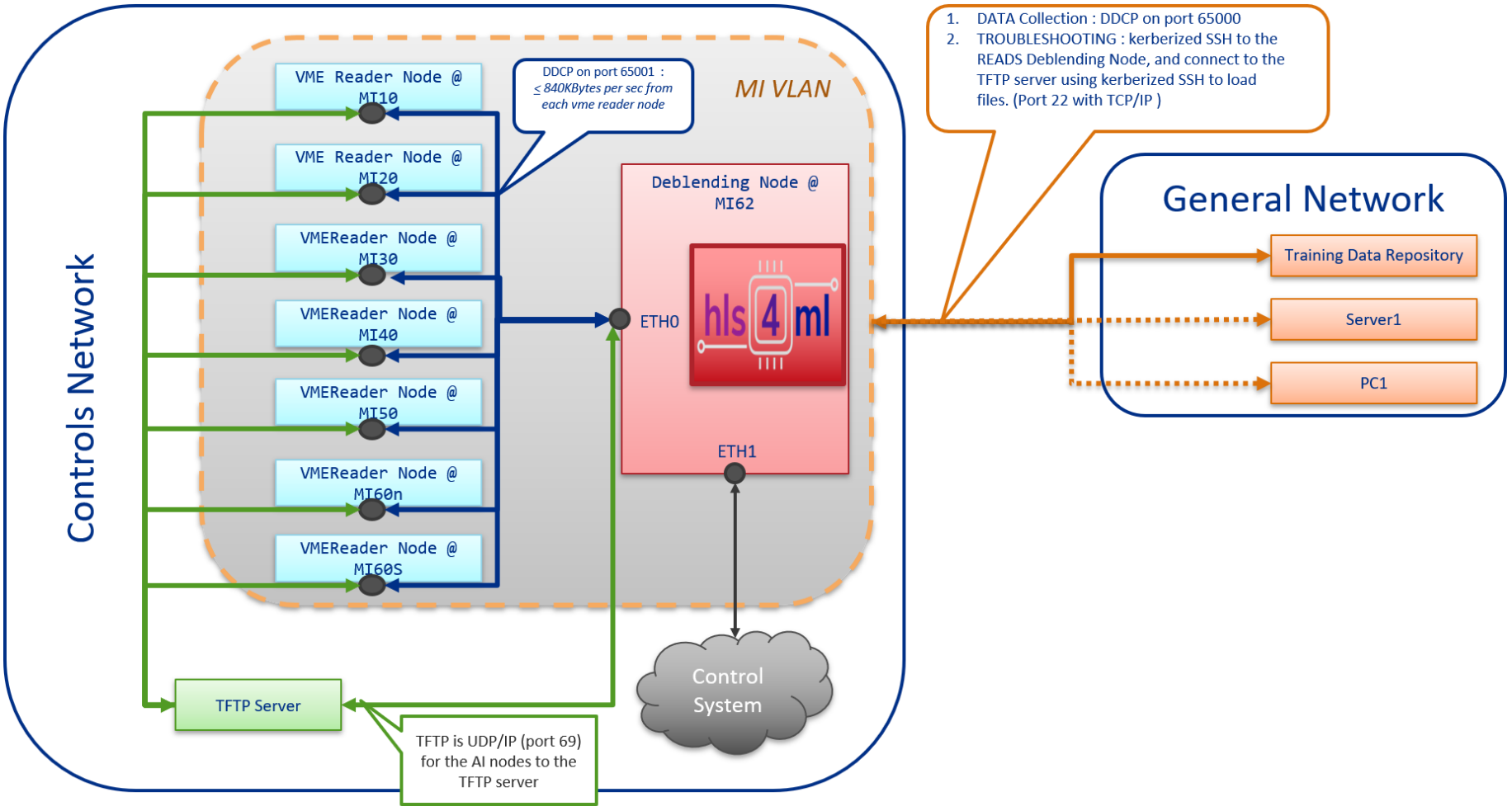
https://indico.cern.ch/event/1133593/

- Machine learning for Linac RF Optimization Longitudinal optimization
- Booster Gradient Magnet Power Supply Control
- “Big Data” Booster Control
- Orbit Alignment at PIP2IT Using Bayesian Optimization
- AI/ ML for NuMI Target System Monitoring
- Real-time quench detection
- FAST/IOTA RF gun stabilization and optimization
- Loss minimization vs MI or RR situation
- Stabilization of 8 GeV slow extraction from Muon-C ring
- 6D Cooling optics design with ML elements

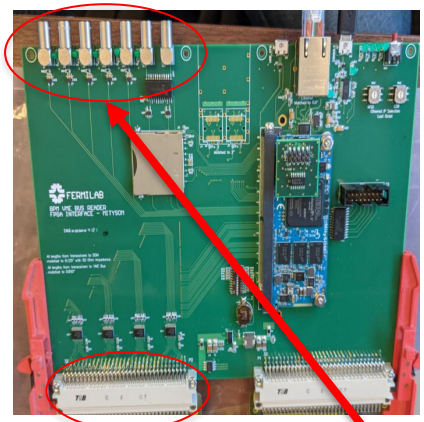
Beam Loss Monitors (BLMs)

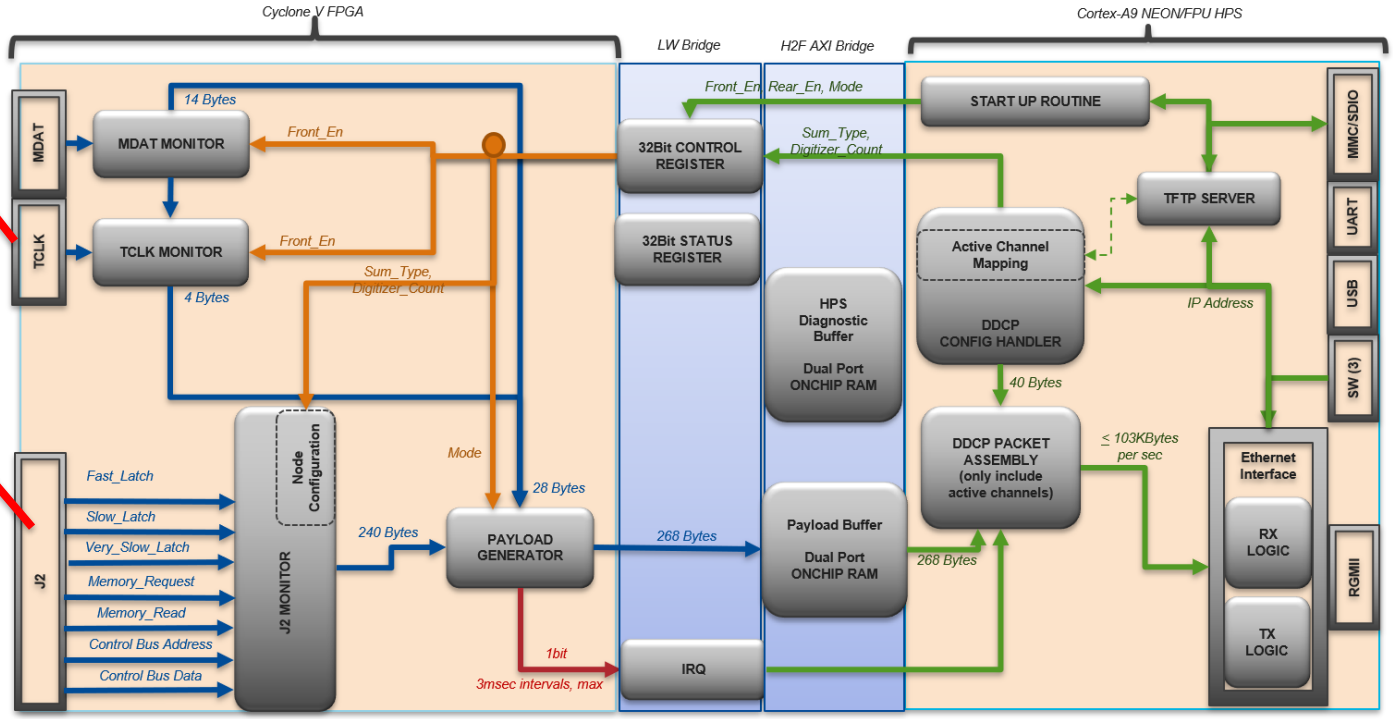


Real-Time Edge AI for Distributed System (READS) Network

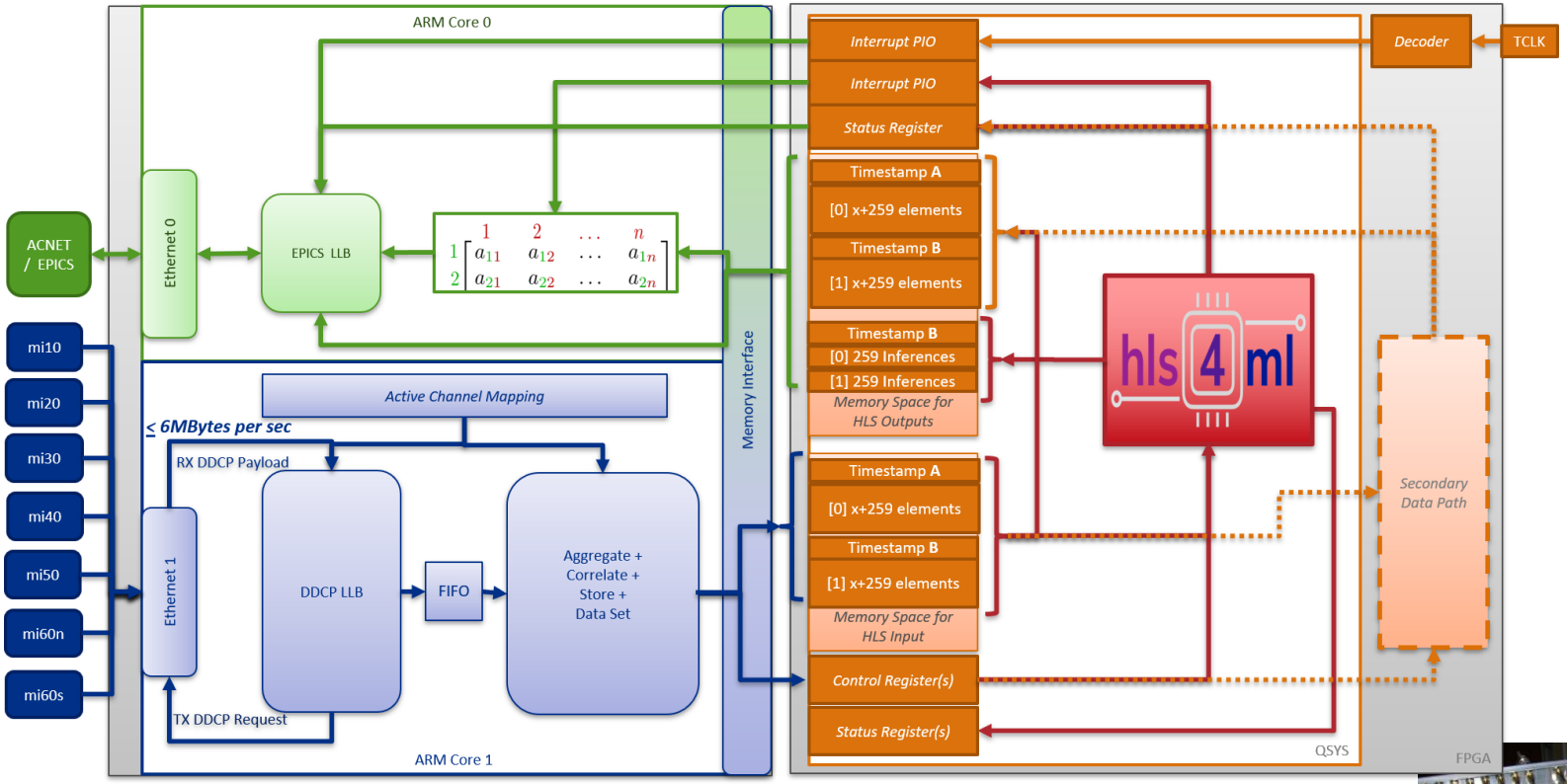


READS Distributed VME Reader Nodes



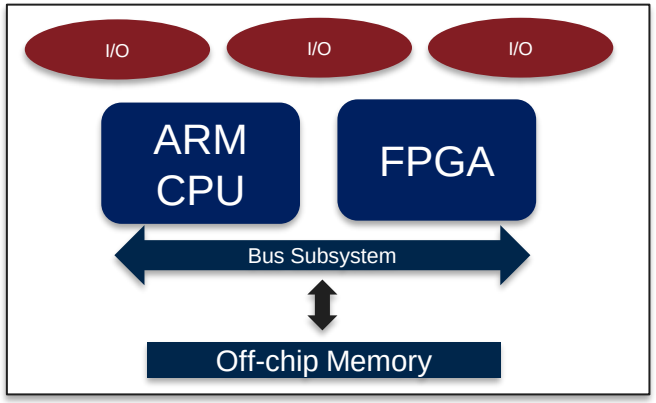


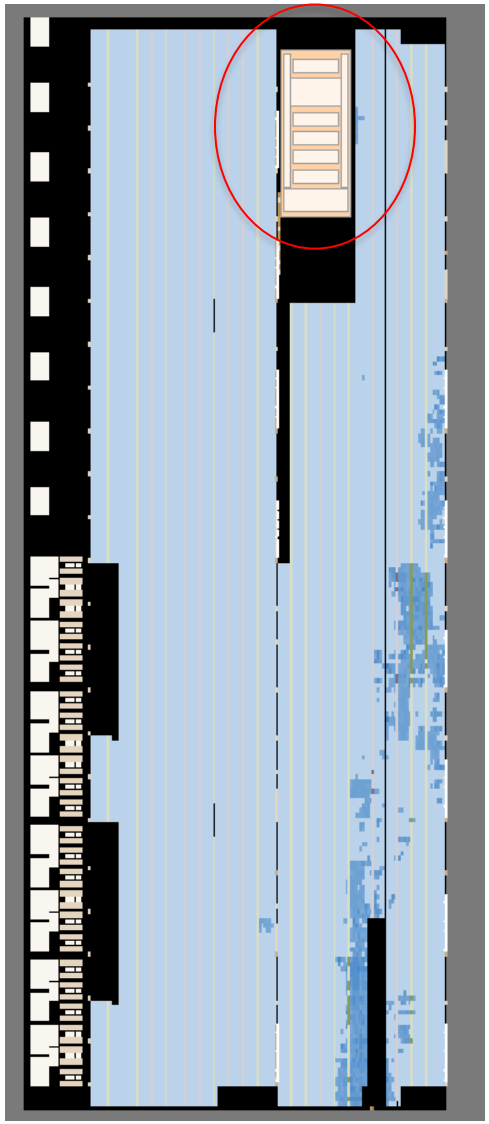
READS Central Deblending Node

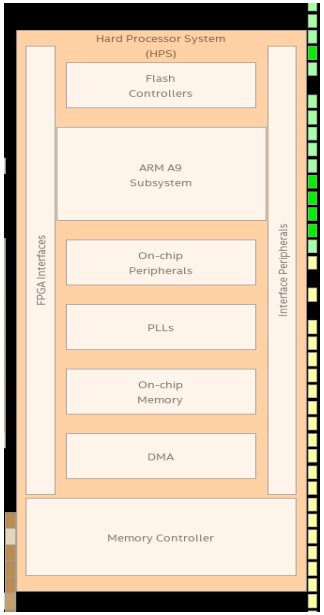


SoC (System on Chip)

It combines a
- Dual-core ARM Cortex-A9 processor
- Field-Programmable Gate Array (FPGA) fabric on a single chip.







SoM (System on Module)

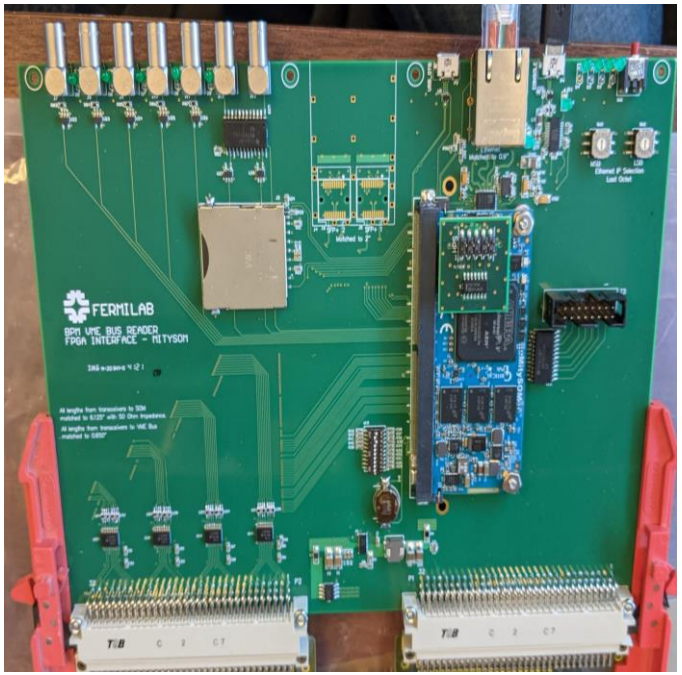
Data Transmission Considerations

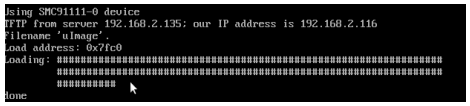
Through ARM:

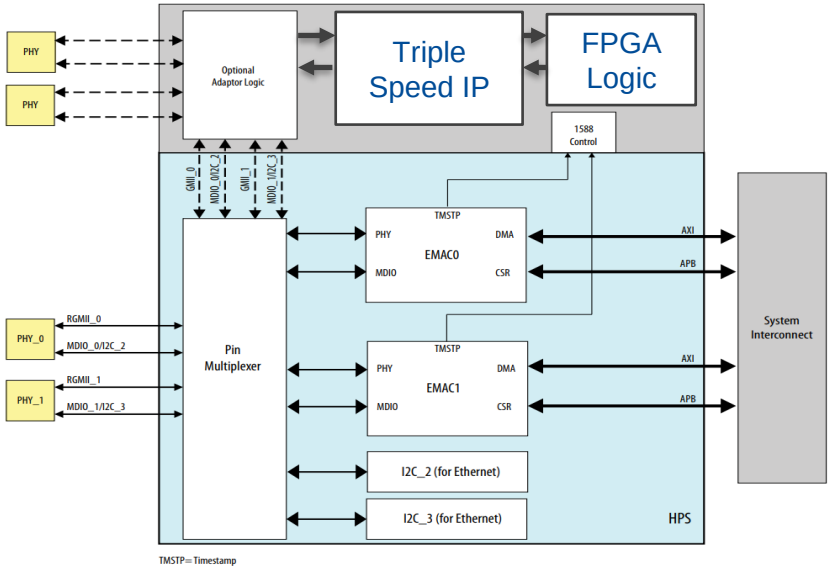
- Rapid prototyping
- More flexibility
- C, C++, Rust, etc.
- TCP/IP Stack readily available
- Longer Latency.
- Load images over the network and SSH.

Through FPGA:

- Needs place and route
- Less flexibility
- VHDL/Verilog
- No IP Stack
- Shorter Latency (Hardware Response).
- Loading image through JTAG is possible.







Ethernet Pins in FPGA or ARM?

FPGA/HPS Data Bridges

hps	Hard Processor System Intel Arria 10 FPGA IP
f2h_cold_reset_req	Reset Input
f2h_warm_reset_req	Reset Input
emif	Conduit
hps_io	Conduit
h2f_reset	Reset Output
h2f_axi_clock	Clock Input
h2f_axi_reset	Reset Input
h2f_axi_master	AXI Master
h2f_lw_axi_clock	Clock Input
h2f_lw_axi_reset	Reset Input
h2f_lw_axi_master	AXI Master
f2h_axi_clock	Clock Input
f2h_axi_reset	Reset Input
f2h_axi_slave	AXI Slave
f2h_irq0	Interrupt Receiver
f2h_irq1	Interrupt Receiver

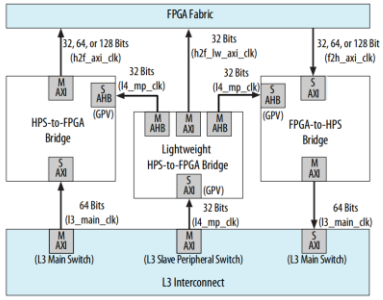


Table 2-2: Common Address Space Regions

Region Name	Base Address	Size
FPGA slaves	0xC0000000	960 MB
Peripheral	0xFC000000	64 MB
Lightweight FPGA slaves	0xFF200000	2 MB

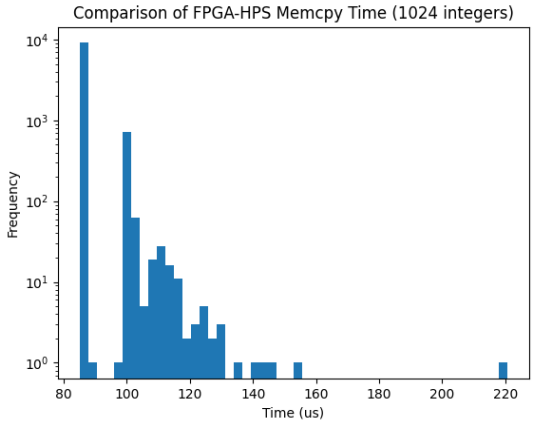
Source: Cyclone V Technical Manual

Exercise: Read 4 KB (1024 32-bit integers) of data through the h2f AXI bridge and calculate the time it takes to read the data.

HPS program can access HPS onchip memory using memcpy.
10,000 readouts were obtained.

1. **MIN**: 85.02 us
2. **MEDIAN**: 85.061 us (~45.8MB/s)
3. **MAX**: 220.772 us

Compare to period of FPGA with 100Mhz Clock (10 ns)
Writing one integer in each clock cycle would be 10ns*1024 = 10.24us.



FPGA Notifications (through PIOs)

Table 2-2: Common Address Space Regions

Region Name	Base Address	Size
FPGA slaves	0xC0000000	960 MB
Peripheral	0xFC000000	64 MB
Lightweight FPGA slaves	0xFF200000	2 MB

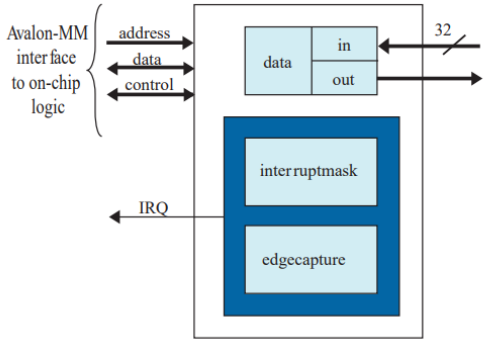
Source: Cyclone V Technical Manual

Register Map for the PIO Core

Offset	Register Name		R/W	(n-1)	...	2	1	0
0	data	read access	R	Data value currently on PIO inputs				
		write access	W	New value to drive on PIO outputs				
1	direction (1)		R/W	Individual direction control for each I/O port. A value of 0 sets the direction to input; 1 sets the direction to output.				
2	interrupt mask (1)		R/W	IRQ enable/disable for each input port. Setting a bit to 1 enables interrupts for the corresponding port.				
3	edge capture (1)		R/W	Edge detection for each input port.				
4	outset		W	Specifies which bit of the output port to set. **Not a physical register**				
5	outclear		W	Specifies which output bit to clear. **Not a physical register.**				

1.Note :This register may not exist, depending on the hardware configuration. If a register is not present, reading the register returns an undefined value, and writing the register has no effect.

Source: [Intel] Embedded Peripherals IP User Guide



Source: [Intel] Embedded Peripherals IP User Guide

altera-gpio.h (Altera gpio driver)
Connect through device tree.
Interface using sysfs

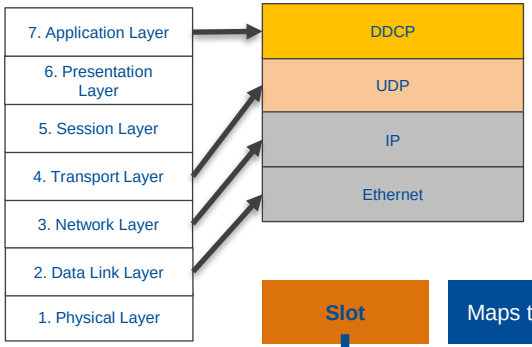
fd_gpio = open("/sys/class/gpio/gpio1984/value", ..)

Polling API

READS Communications

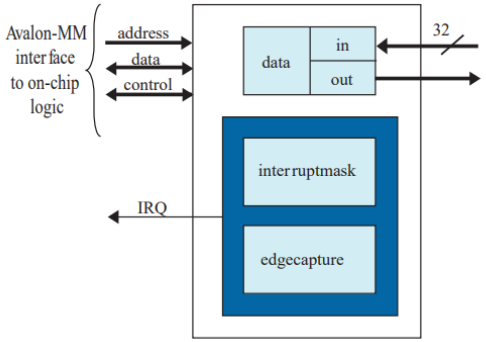
Across Network

OSI Model



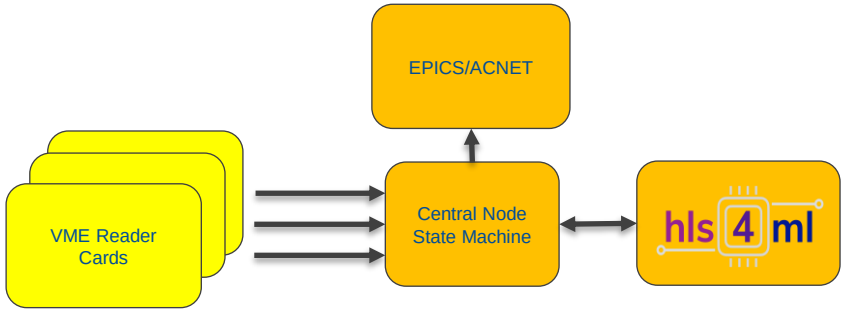
READS uses **DDCP/UDP** packets to:

- Stream data between the captured data to the Central Node.
- Control hardware registers and registers in the software state machine in the VME Reader Cards.



Slot	Maps to a device IP in network.
Feature	Maps to a feature in the device (a register, an array, etc.)
Operation	Operations that can be done on a feature.

To EPICS/ACNET



The State Machine in the central node manages the incoming DDCP stream and presents it to the HLS model.

The output of the HLS4ML model is presented with the original inputs timestamped to an EPICS API that loads it into ACNET.

READS Publications

- Accelerator Real-time Edge AI for Distributed Systems (READS) Proposal (March 2020)
  https://arxiv.org/abs/2103.03928
- Real-Time Edge AI for Distributed Systems (READS): Progress on Beam Loss De-Blending for the Fermilab Main Injector and Recycler (August 2021)
  https://jacow.org/ipac2021/papers/mopab288.pdf
- Optimizing Mu2e Spill Regulation System Algorithms (August 2021)
  https://jacow.org/ipac2021/papers/THPAB243.pdf
- Synchronous High-Frequency Distributed Readout for Edge Processing at the Fermilab Main Injector and Recycler (August 2022)
  https://napac2022.vrws.de/papers/mopa15.pdf
- Semantic Regression for Disentangling Beam Losses in the Fermilab Main Injector and Recycler (August 2022)
  https://napac2022.vrws.de/papers/mopa28.pdf
- Machine Learning for Slow Spill Regulation in the Fermilab Delivery Ring for Mu2e (August 2022)
  https://napac2022.vrws.de/papers/mopa75.pdf