HPS Review, SLAC January 18, 2018

SVT DAQ Update

Ryan Herbst, Ben Reese

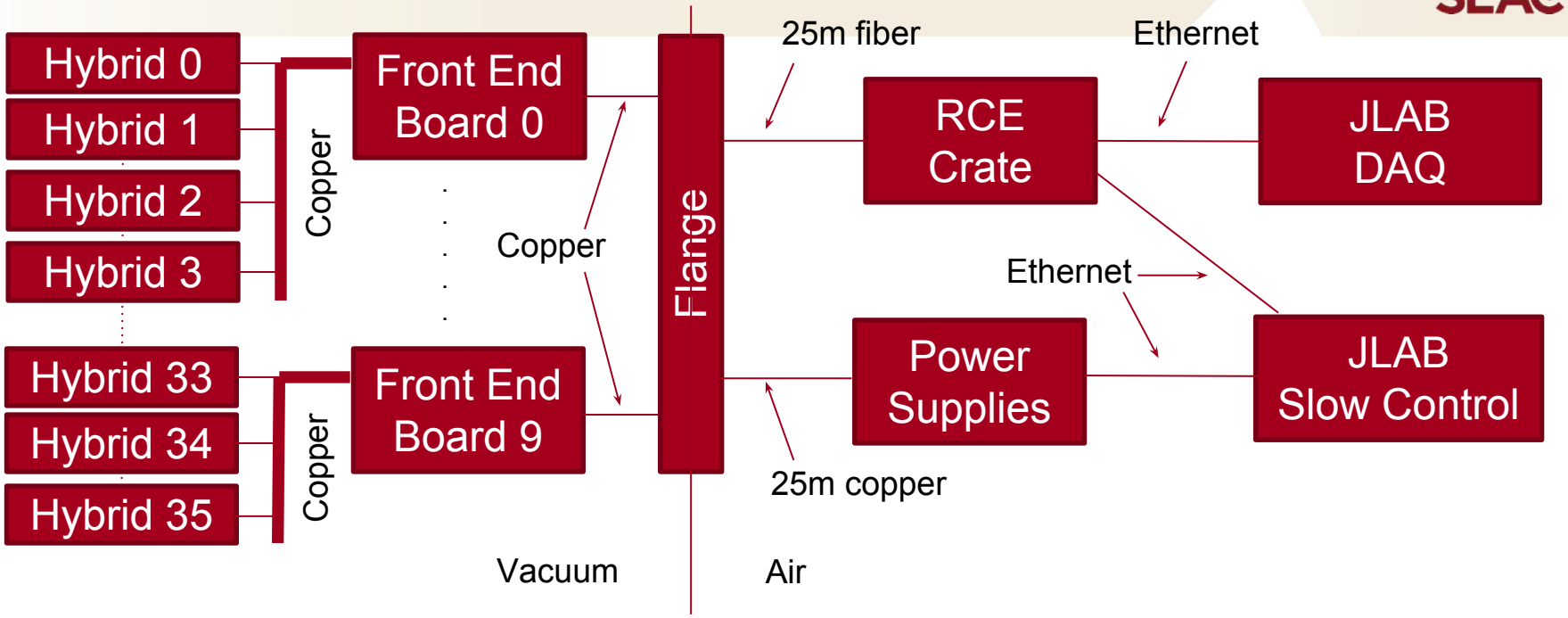
U.S. DEPARTMENT OF ENERGY
Office of Science

SLAC NATIONAL ACCELERATOR LABORATORY

Overview

- SVT DAQ Overview - Existing System
- Planned Updates
- Status of Work

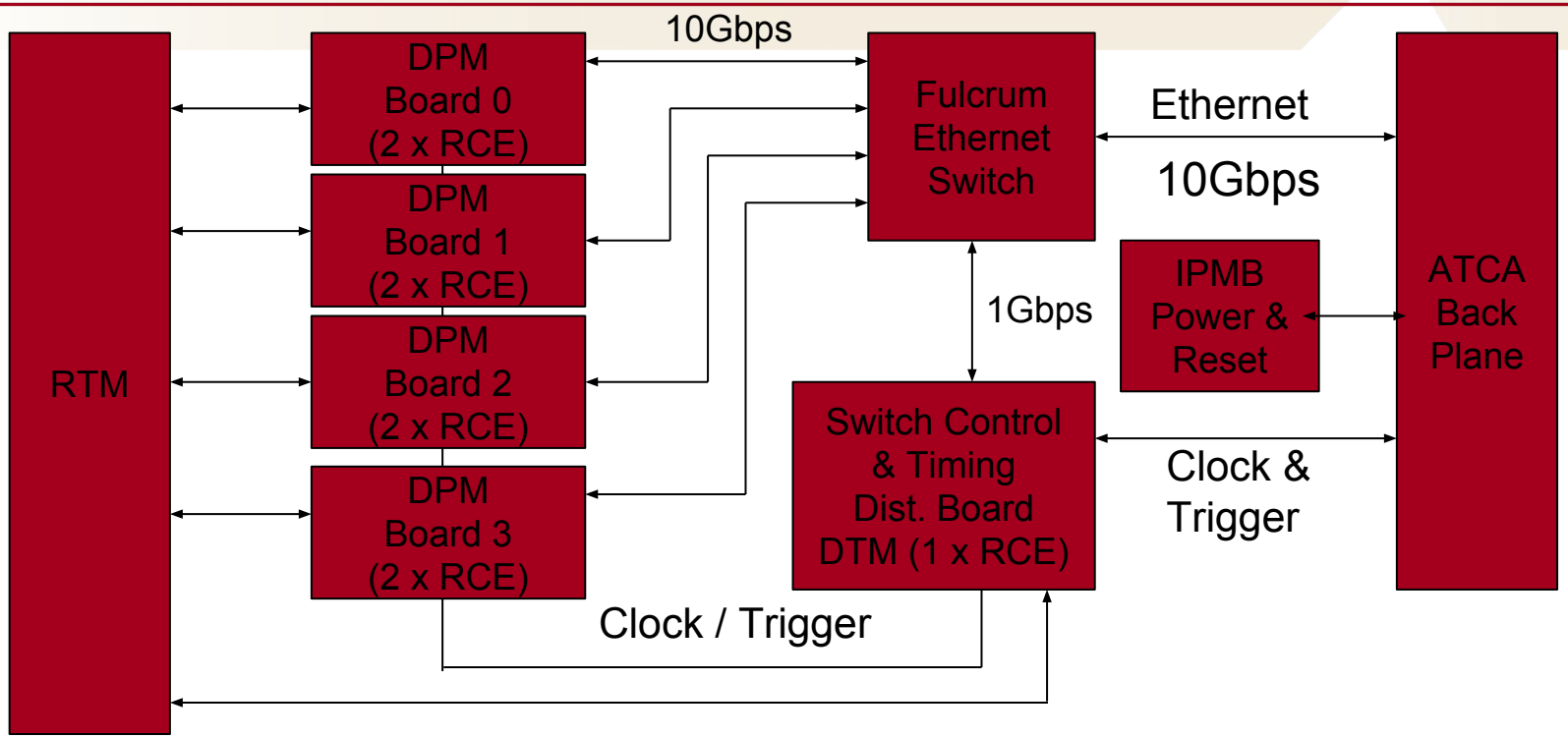
Existing SVT Overview



- 36 hybrids
  - 12 in layers 0 – 3 (2 per module)
  - 24 in layers 4 – 6 (4 per module)
- 10 front end boards
  - 4 servicing layers 1 – 3 with 3 hybrids per board
  - 6 servicing layers 4 – 6 with 4 hybrids per board
- RCE crate: data reduction, event building and JLab DAQ interface

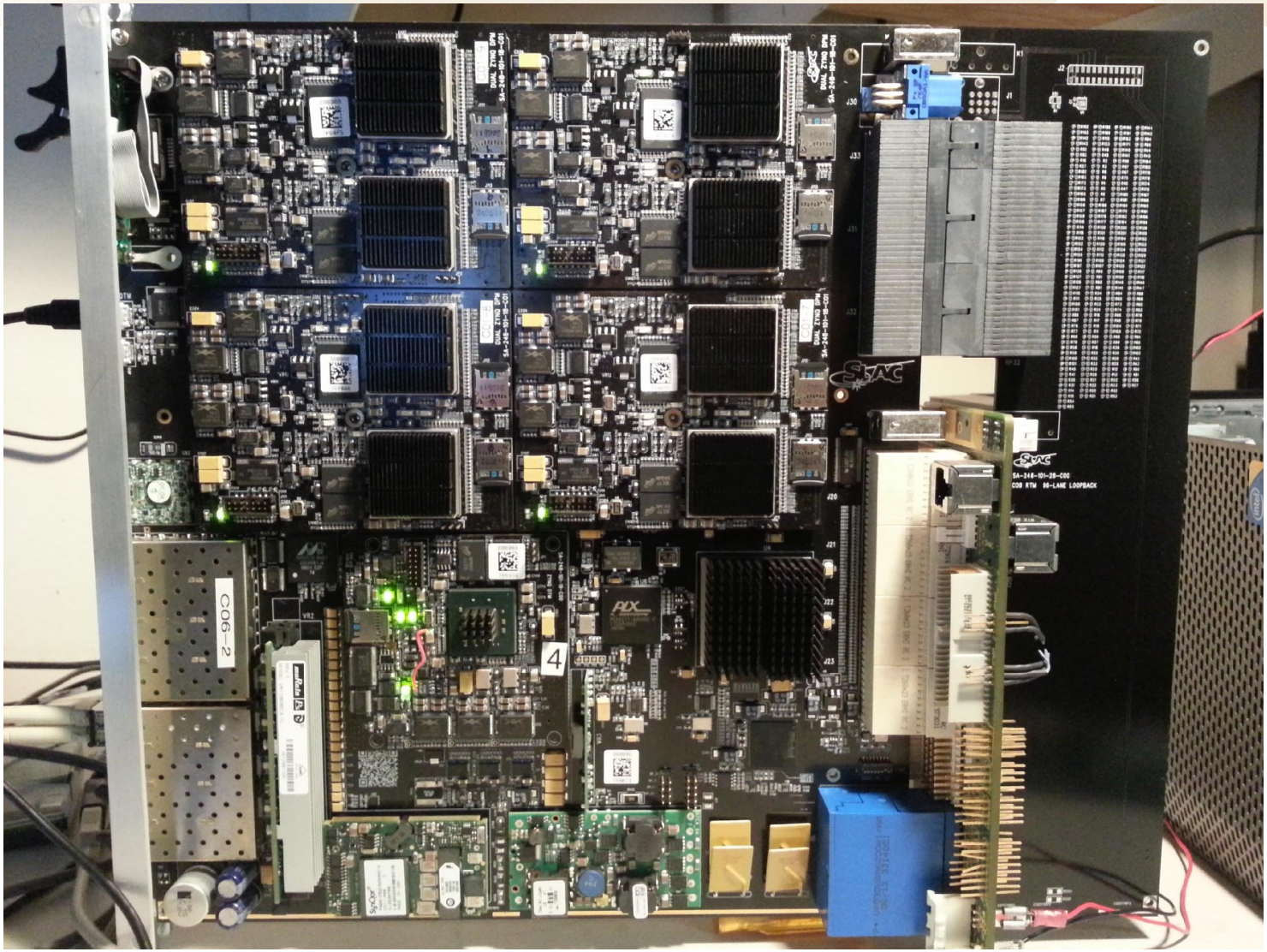
Raw ADC data rate (Gbps)	
Per hybrid	3.33
Per L1-3 Front end board	10
Per L4-6 Front end board	13

SLAC Gen3 COB (Cluster on Board)



- Supports 4 data processing FPGA mezzanine cards (DPM)
  - 2 RCE nodes per DPM
  - 12 bi-directional high speed links to/from RTM (GTP)
- Data transport module (DTM)
  - 1 RCE node
  - Interface to backplane clock & trigger lines & external trigger/clock source
  - 1 bi-directional high speed link to/from RTM (GTP)
  - 6 general purpose low speed pairs (12 single ended) to/from RTM
    - connected to general purpose pins on FPGA

RCE GEN3 COB

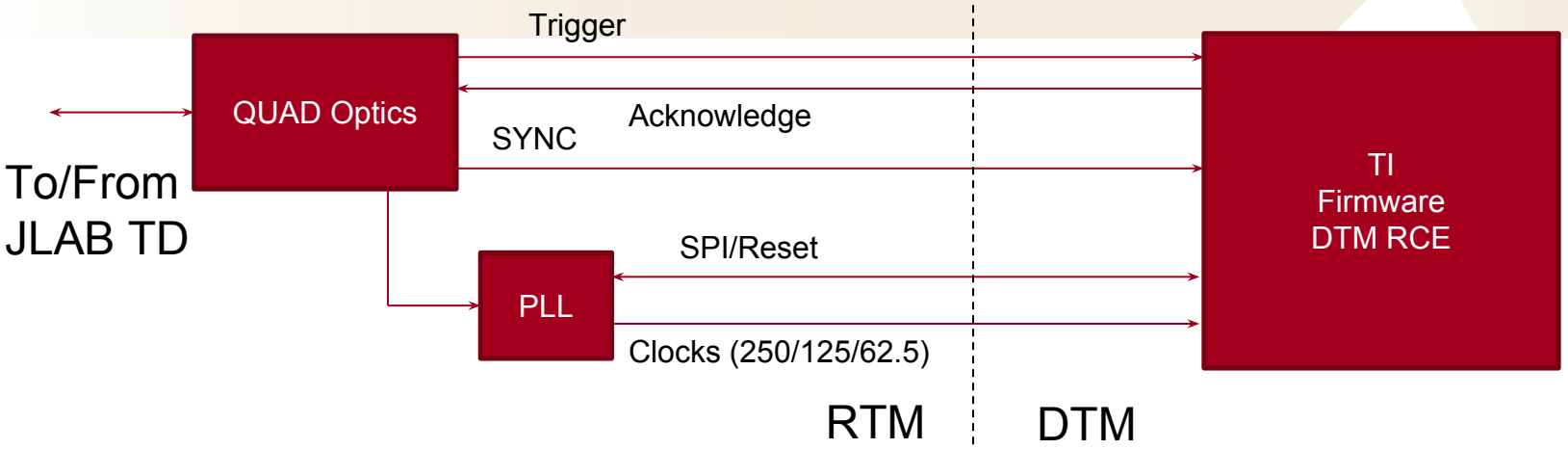


Existing SVT RCE Allocation

SLAC

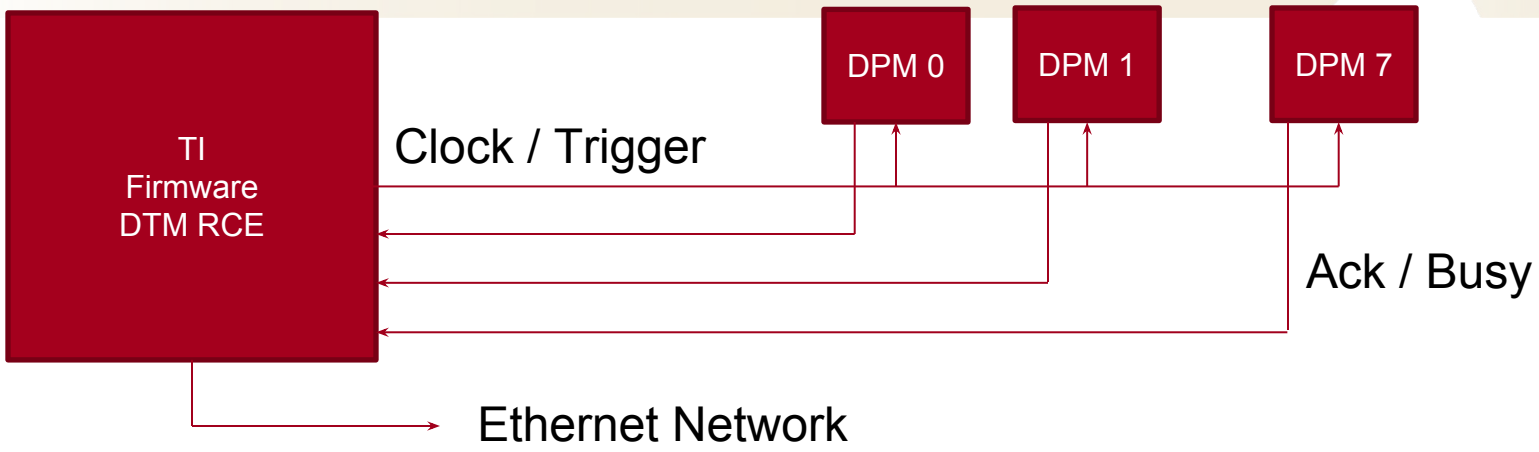
- Two COBs utilized in the SVT readout system
  - 16 RCEs On DPMs (2 per DPM, 4 DPMs per COB)
  - 2 RCEs on DTMs (1 per DTM, 1 DTM per COB)
- 7 RCEs on each COB process data from ½ SVT
  - 18 Hybrids total per COB
  - RCE 0 = 2 hybrids (layer 1)
  - RCE 1 = 2 hybrids (layer 2)
  - RCE 2 = 2 hybrids (layer 3)
  - RCE 3 = 3 hybrids (3 from layer 4)
  - RCE 4 = 3 hybrids (1 from layer 4 / 2 from layer 5)
  - RCE 5 = 3 hybrids (2 from layer 5 / 1 from layer 6)
  - RCE 6 = 3 hybrids (3 from layer 6)
- RCE 7 on COB 0 manages all 10 FE Boards
  - Configuration and status messages
  - Clock and trigger distribution to FE boards & hybrids
- RCE 7 on COB 1 has does not have an SVT specific purpose

SVT Trigger Interface



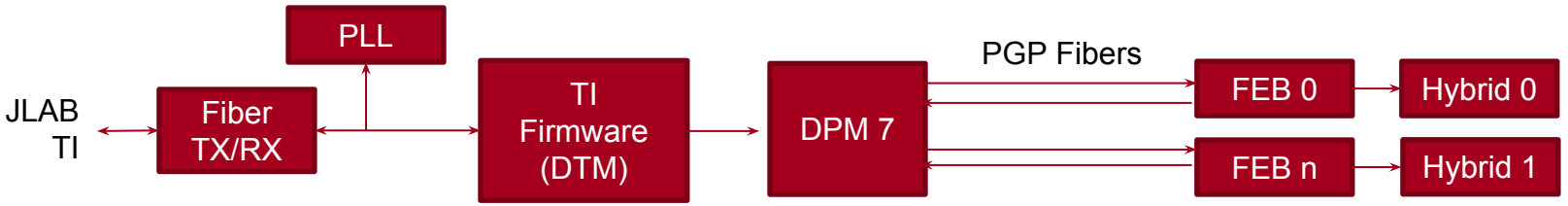
- Replicates portion of JLAB TI Board
- Quad optics and PLL exist on RTM
- TI firmware implemented in RCE FPGA
- Fully allocated available signals between RTM and DTM
  - 1 high speed pair for trigger & SYNC
  - 1 low speed pair for SYNC
  - 2 low speed pairs for PLL SPI and Reset signals
  - 3 low speed pairs for PLL generated clocks (250/125/62.5 Mhz)

SVT Trigger Distribution



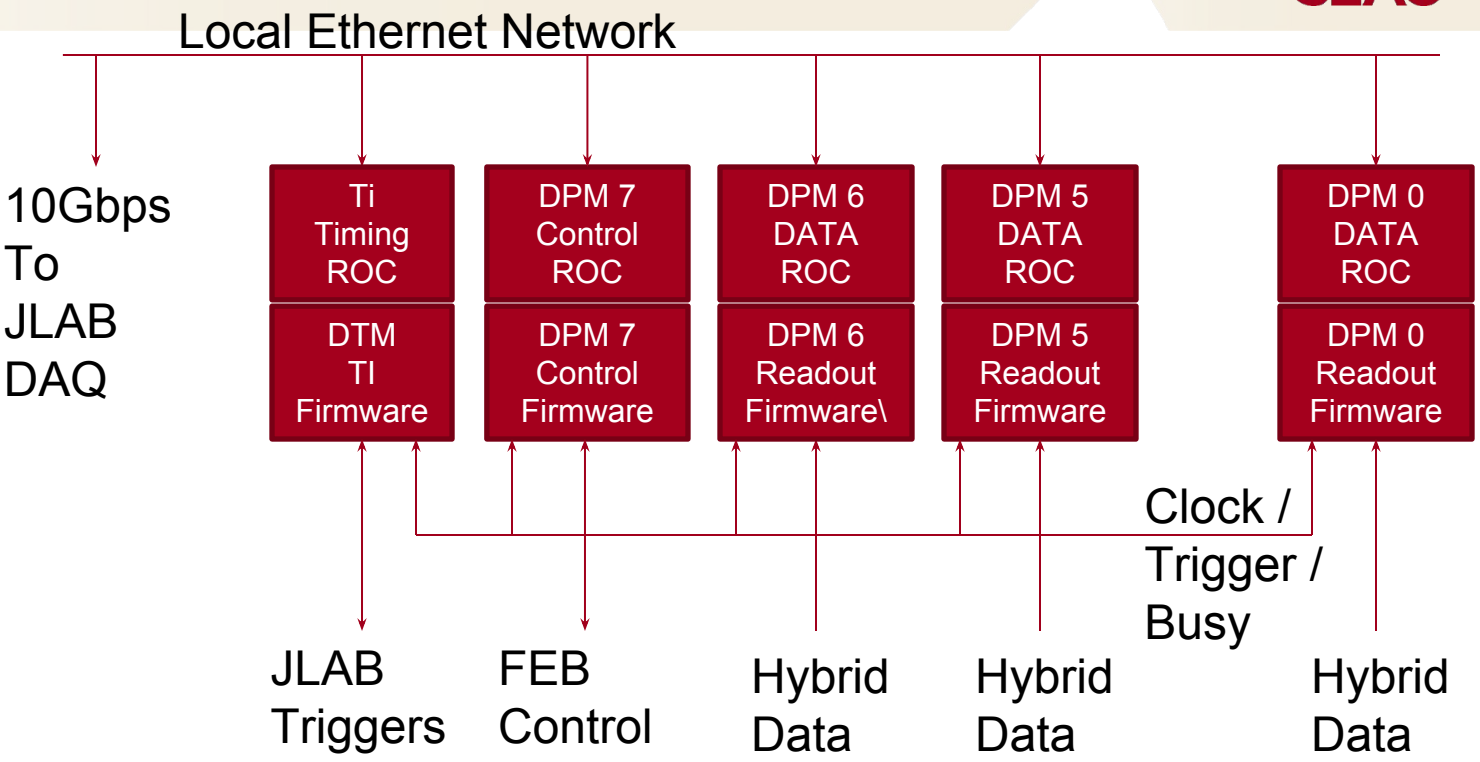
- DTM FPGA has ability to distribute clock and trigger to DPMs
  - Clock and trigger wired as fan out to DPMs
  - Individual feedback signals from each DPM
- 1 pair for clock fan out
- 1 pair for trigger fan out
  - 125Mhz serial protocol transfers 8-bit codes (easily expanded to longer words)
  - Used to distribute event codes to DPMs
    - System clock sync, APV25 sync & JLAB triggers
- 1 pair for trigger data distribution
  - Event and block data
- 1 pair per DPM for feedback
  - Readout and trigger acknowledge
  - Busy

Front End Timing Distribution



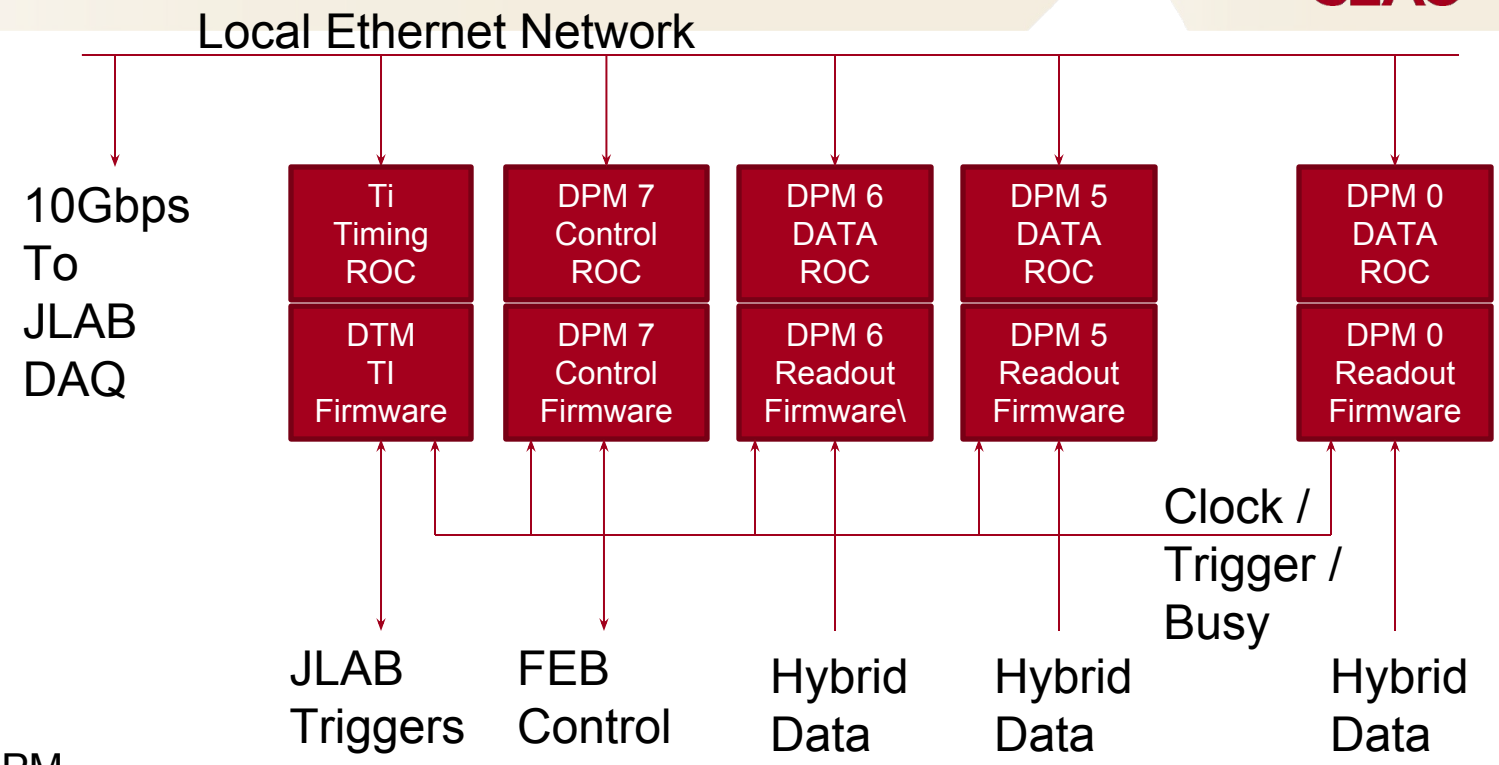
- Control DPM forwards timing information to front end boards over PGP
  - Clock encoded into serial data stream which the front end board recovers
  - Fixed latency path for encoded PLL reset and trigger signals

ROC Instances On SVT



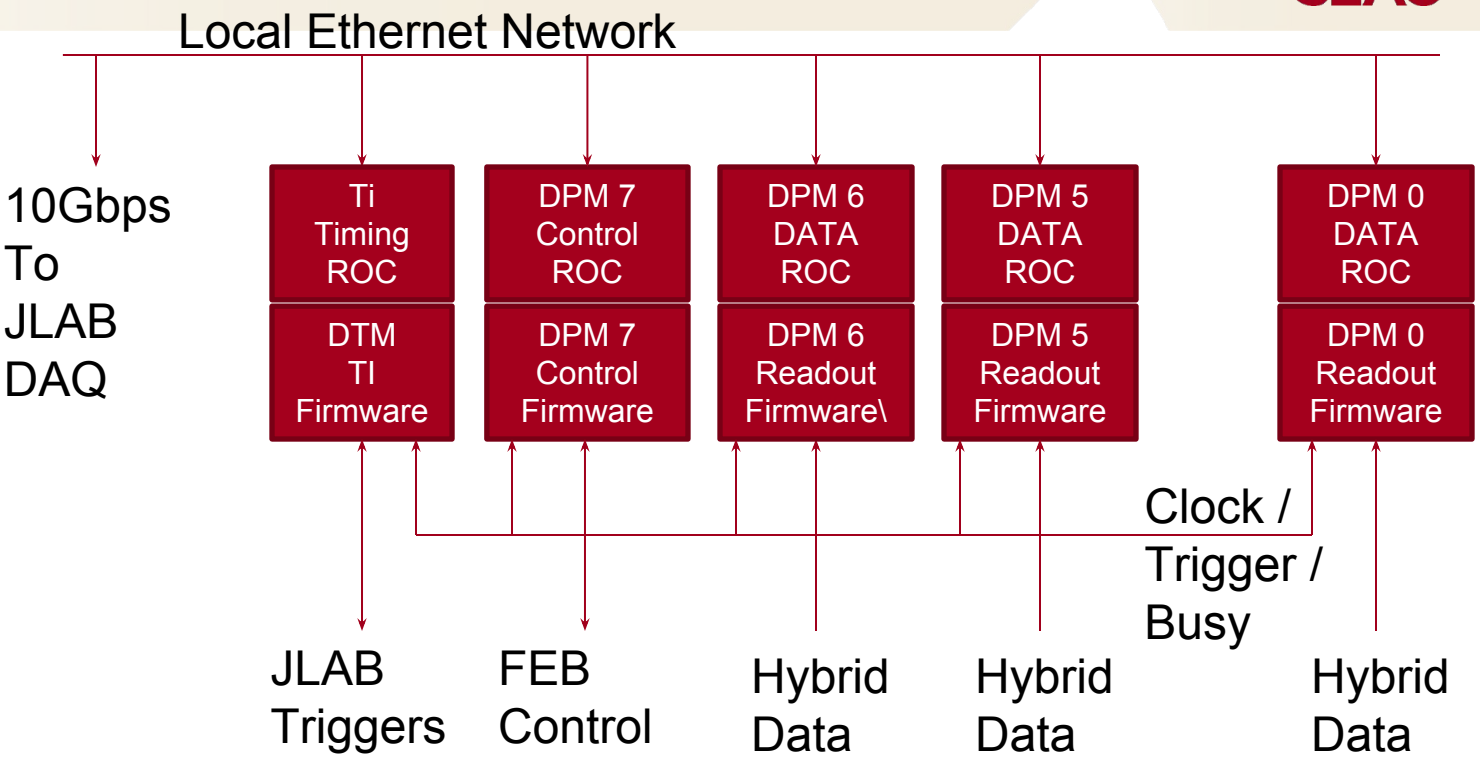
- Data DPM
  - Data processing ROC application
  - Builds event record for 2 or 3 hybrids
    - APV25 ADC Data
    - Hybrid environmental data
  - Operates as slave when interfacing to TI firmware
  - Clock, trigger & event data received over COB signals
  - Busy and acknowledge passed over COB signals

ROC Instances On SVT



- Control DPM
  - FEB control and configuration
  - Formats FEB environmental event data
  - Clock, trigger & event data received over COB signals
  - Busy and acknowledge passed over COB signals

ROC Instances On SVT



- Timing DTM
  - Dataless ROC instances for state transition control
  - Clock, trigger & event sent received over COB signals
  - Busy and acknowledge received over COB signals

Planned SVT DAQ Updates

SLAC

- Add new layer 0 Hybrids
- Update firmware to latest build system and common libraries
- Add bootloader image to FEB, enable firmware downloads
- Update control software to python based Rogue platform
- Move from multiple ROCs (one per RCE) to two ROCs on linux server (one ROC per COB)
- Upgrade FEB Test Stand

Updated SVT RCE Allocation

SLAC

- Two COBs utilized in the SVT readout system
  - 16 RCEs On DPMs (2 per DPM, 4 DPMs per COB)
  - 2 RCEs on DTMs (1 per DTM, 1 DTM per COB)
- 7 RCEs on each COB process data from ½ SVT
  - 19 Hybrids total per COB
  - RCE 0 = 3 hybrids (layer 0 + Layer 1)
  - RCE 1 = 2 hybrids (layer 2)
  - RCE 2 = 2 hybrids (layer 3)
  - RCE 3 = 3 hybrids (3 from layer 4)
  - RCE 4 = 3 hybrids (1 from layer 4 / 2 from layer 5)
  - RCE 5 = 3 hybrids (2 from layer 5 / 1 from layer 6)
  - RCE 6 = 3 hybrids (3 from layer 6)
- RCE 7 on COB 0 manages all 10 FE Boards
  - Configuration and status messages
  - Clock and trigger distribution to FE boards & hybrids
- RCE 7 on COB 1 has does not have an SVT specific purpose

Firmware - Update libraries

The TID-AIR-ES firmware build environment and common libraries have changed a lot since HPS was deployed

- SVN -> GIT
- Consolidated common libraries -> SURF
  - Also lots of fixes and updates in these libraries
- New firmware build system -> Ruckus

Upgrading the HPS firmware to use all of this is necessary to maintain our ability to build it and patch it.

Status - Done. All firmware images have been successfully built using the latest and greatest code.

FEB Firmware - Bootloader

SLAC

- Add a bootloader image to the FEB PROM
- We've had the ability to reflash over PGP, but it was dangerous
  - If something goes wrong, need physical access to JTAG connector at the flange
- Bootloader solves this
  - Two images stored in PROM
  - A stripped down FEB image that is proven to work loads first after powerup.
  - After 10 seconds, it loads the "real" FEB image
    - Can disable this via register access
  - Only the "real" FEB image can be rewritten over PGP
- Status - done. Installed during September JLAB trip

Software - Migrate to Rogue

SLAC

- TID-AIR-ES has a new software framework for communicating with FPGAs - Rogue
- Replacement for old C++ GenDaq/XmlDaq framework
- Rogue features
  - Split C++/Python architecture
  - Most development is done in python
  - Can drop into C++ when performance is critical
  - Built in methods for inter-process coordination
- Status
  - Device configuration code has been rewritten in Rogue.
  - Can use Rogue to configure a FEB and Hybrid on the Test Stand
  - Still need work to deploy on RCE Crate

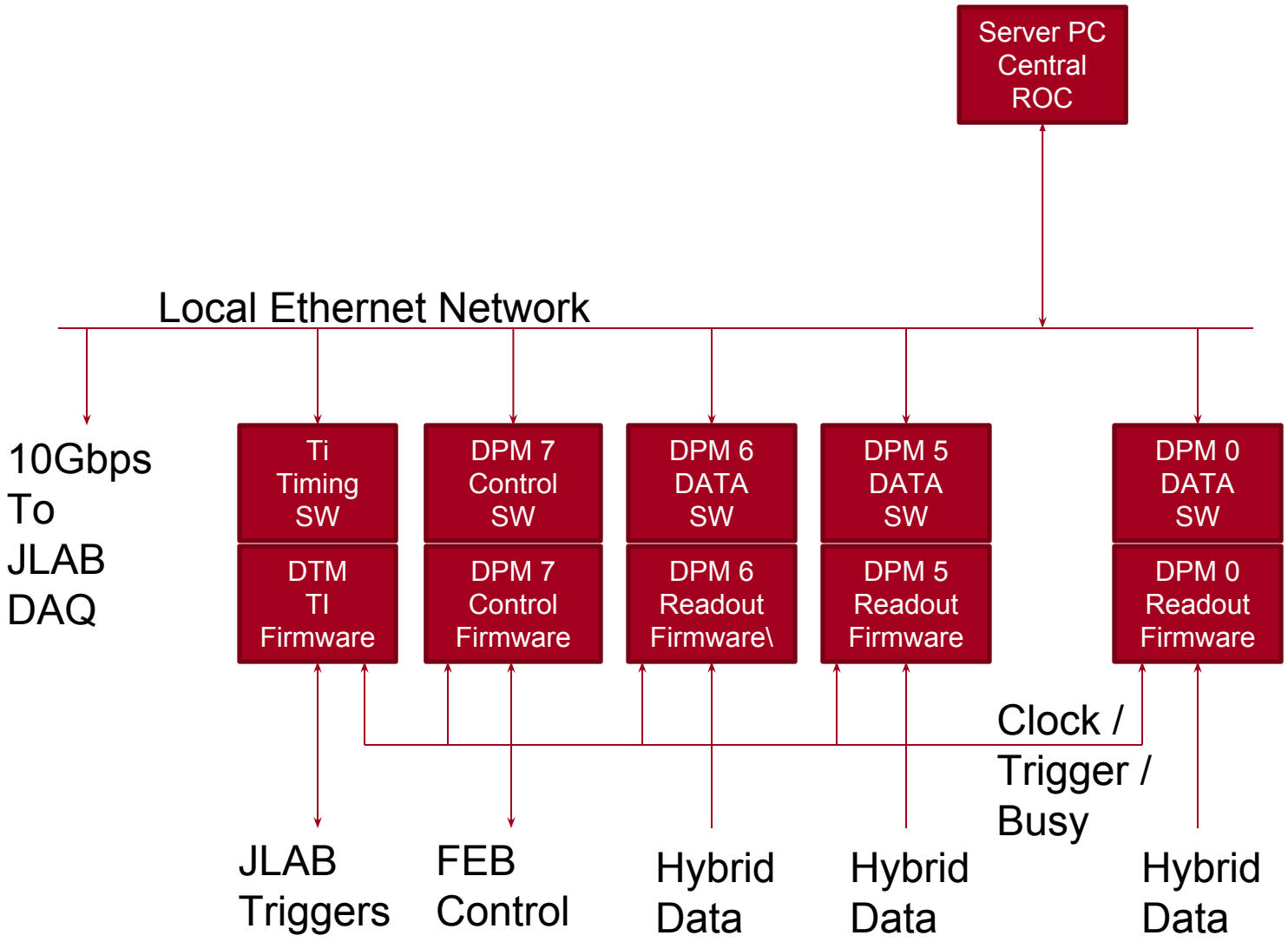
Move from One ROC/RCE to One ROC/COB

SLAC

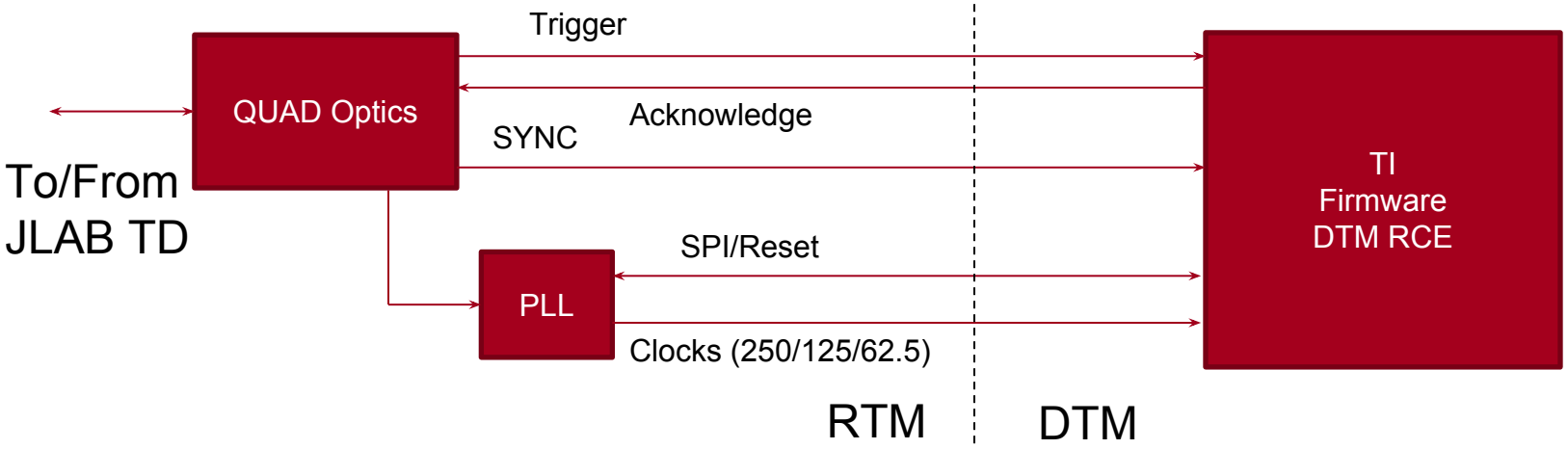
- Existing system has some stability issues at startup and during running
- All ROC instances must constantly stay in SYNC, occasionally during runs one or more ROC instances will freeze up
- Reducing the number of ROC instances will increase stability

- New structure will have a single ROC instances per COB, hosted on a Linux server.
  - ROC on the linux server will executate state transitions through remote python calls to pyrogue running on each RCE
    - Considering direct communication between single ROC and RCE firmware
  - Run data (7 RCEs) and environmental data (1 RCE) will be sent from the RCEs to an event builder on the linux server
    - RSSI firmware to software reliable transfer
    - Event builder will format frames and pass to local ROC data interface

Move from One ROC/RCE to One ROC/COB



Update Trigger Firmware



- The TI firmware has changed, so we need to integrate the new TI logic from JLab into the DTM firmware
  - Successfully accomplished for previous run
- TI firmware is implemented in ISE schematic
  - Export to VHDL and make minor edits to enable compile in Vivado
    - Update block rams, etc

Update FEB Test Stand

SLAC

Old FEB Test Stand

- FEB connected to a PC via standard PGP card
- FEB loaded with special firmware
  - RCE data processing blocks in FEB
  - 3.125 Gbps PGP speed instead of 2.5 Gbps on CTRL link
  - Data PGP links not used, everything over CTRL link.
- Issues
  - The FEB + RCE logic grew to the point that it no longer fit on the FEB FPGA
  - Would prefer for the FEB Test Stand to run the same firmware as SVT FEBs.

Update FEB Test Stand

KCU1500 PCIe card from Xilinx

- General purpose PCIe card with huge FPGA and 8 high speed links
- TID-AIR-ES no longer maintains a custom PGP card. We use this instead.
- Can fit all the RCE ControlDpm and DataDpm firmware.
- Can mimic the RCE enough to attach a FEB with standard firmware directly to a PC
- Upgrading the FEB Test Stand to do this

RCE Software Update

SLAC

Once the FEB Test Stand is working, we will need to deploy Rogue based HPS software on the RCE Crate*

- Rogue will compile and run on the RCE nodes
- Need to configure bridge to CODA - ROC through shared memory
- More work is needed to manage all the Rogue instances on each of the RCE nodes.

- *Also investigating having Rogue instance on server communicate directly to RCE FPGA over Ethernet

RCE at SLAC

SLAC

Currently in the process of spinning up an RCE Crate at SLAC

- Mimic the JLab SVT DAQ setup
- 1 COB+RTM, 1-2 FEBs and 1-8 Hybrids
- RTM and COB shipped from JLab
- Cameron and Omar helping to set up
- Rogue installed and running
- Existing DAQ software running

FEB Hardware Status

- 15 FEBs were made
- 1 was a lemon and never worked
- 10 were deployed in the SVT
- 4 Remaining
- We have 2 backup FEBs at SLAC
  - Can’t get DATA links working on FEB#08
    - Replacing oscillator might fix it
  - FEB#10 seems ok
  - FEB#09 and FEB#03 need to be tested

Thank You!