HPS Collaboration Review, JLAB May 29, 2019

SVT DAQ Update

Cameron Bravo, Ryan Herbst, Ben Reese, JJ Russell

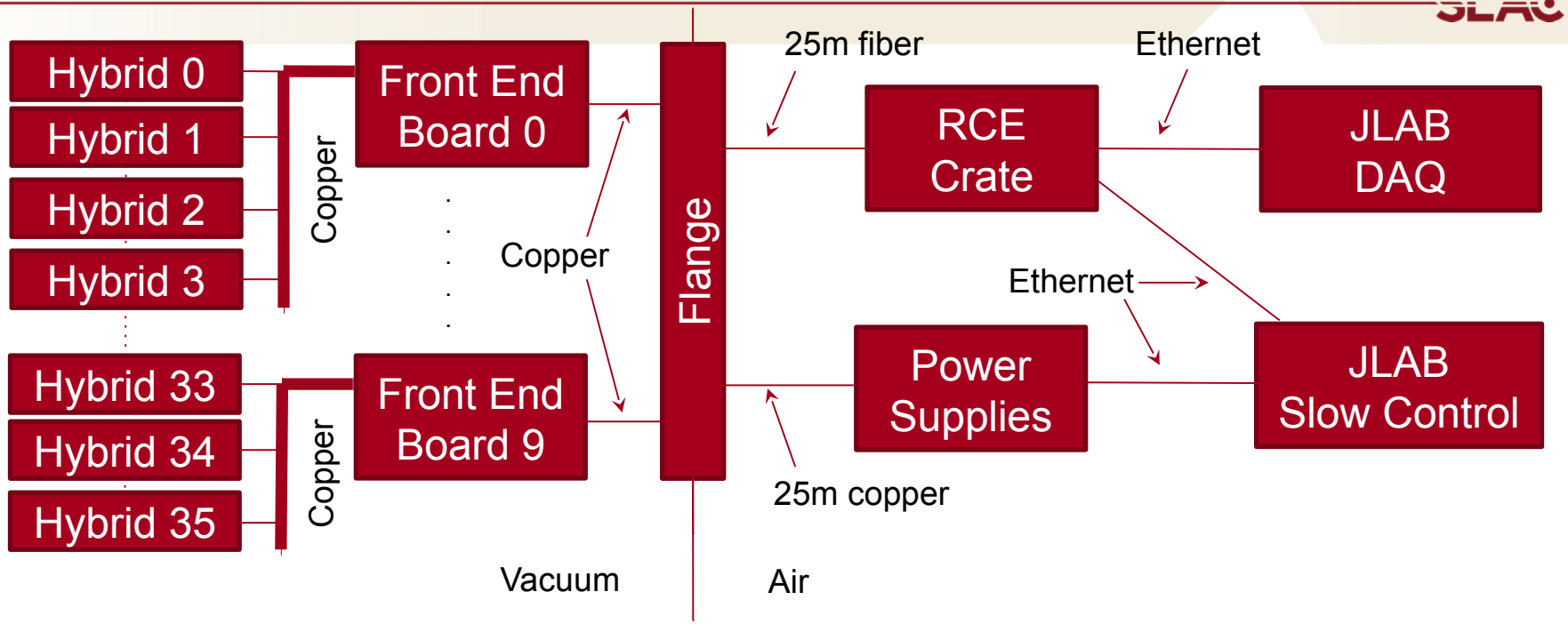
U.S. DEPARTMENT OF ENERGY
Office of Science

SLAC NATIONAL ACCELERATOR LABORATORY

Overview

- SVT DAQ Overview
- Planned Updates
- Status of Work

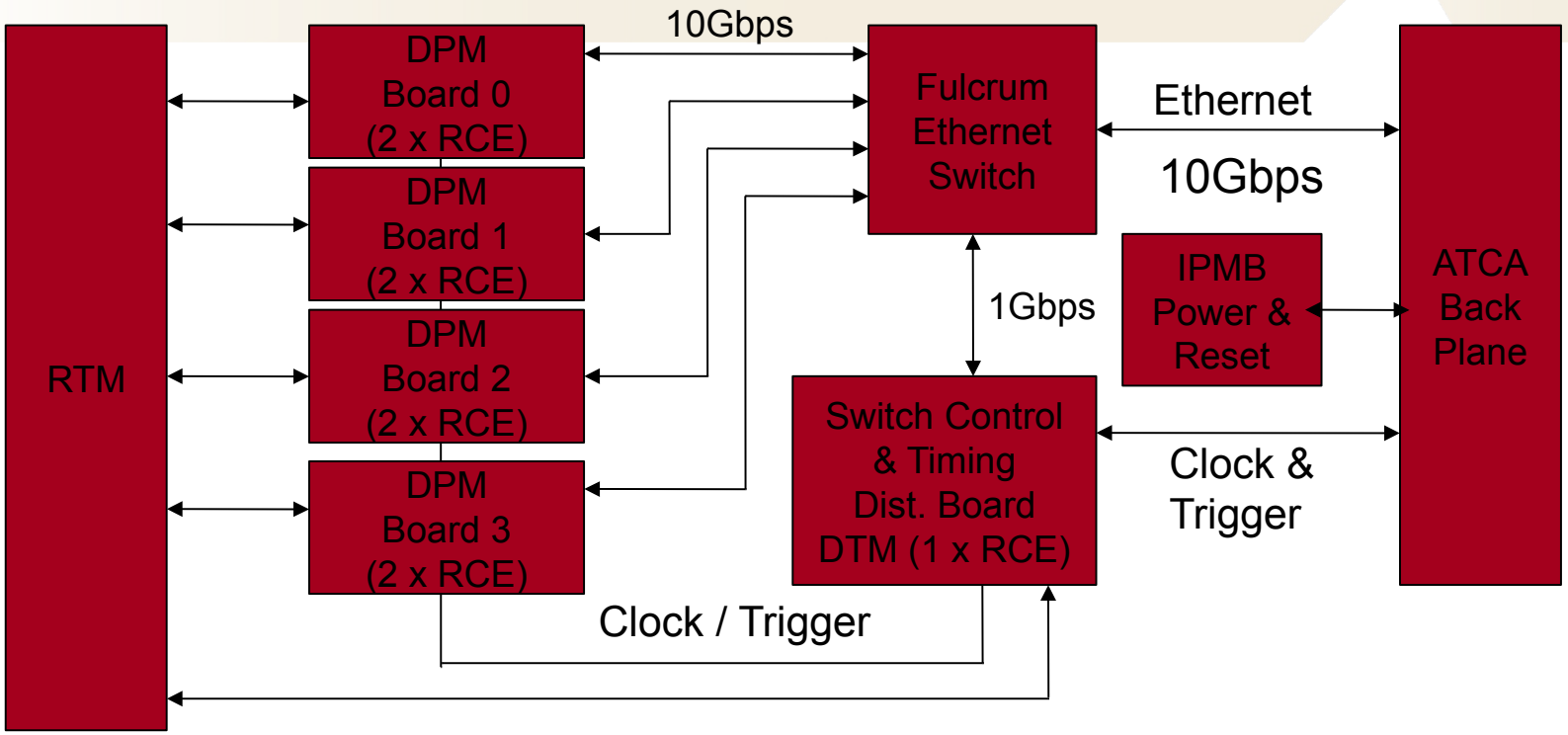
SVT DAQ Overview



- 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 0 – 3 with 4 hybrids per board
  - 6 servicing layers 4 – 6 with 4 hybrids per board
- RCE crate: ATCA, data reduction, event building and JLab DAQ interface

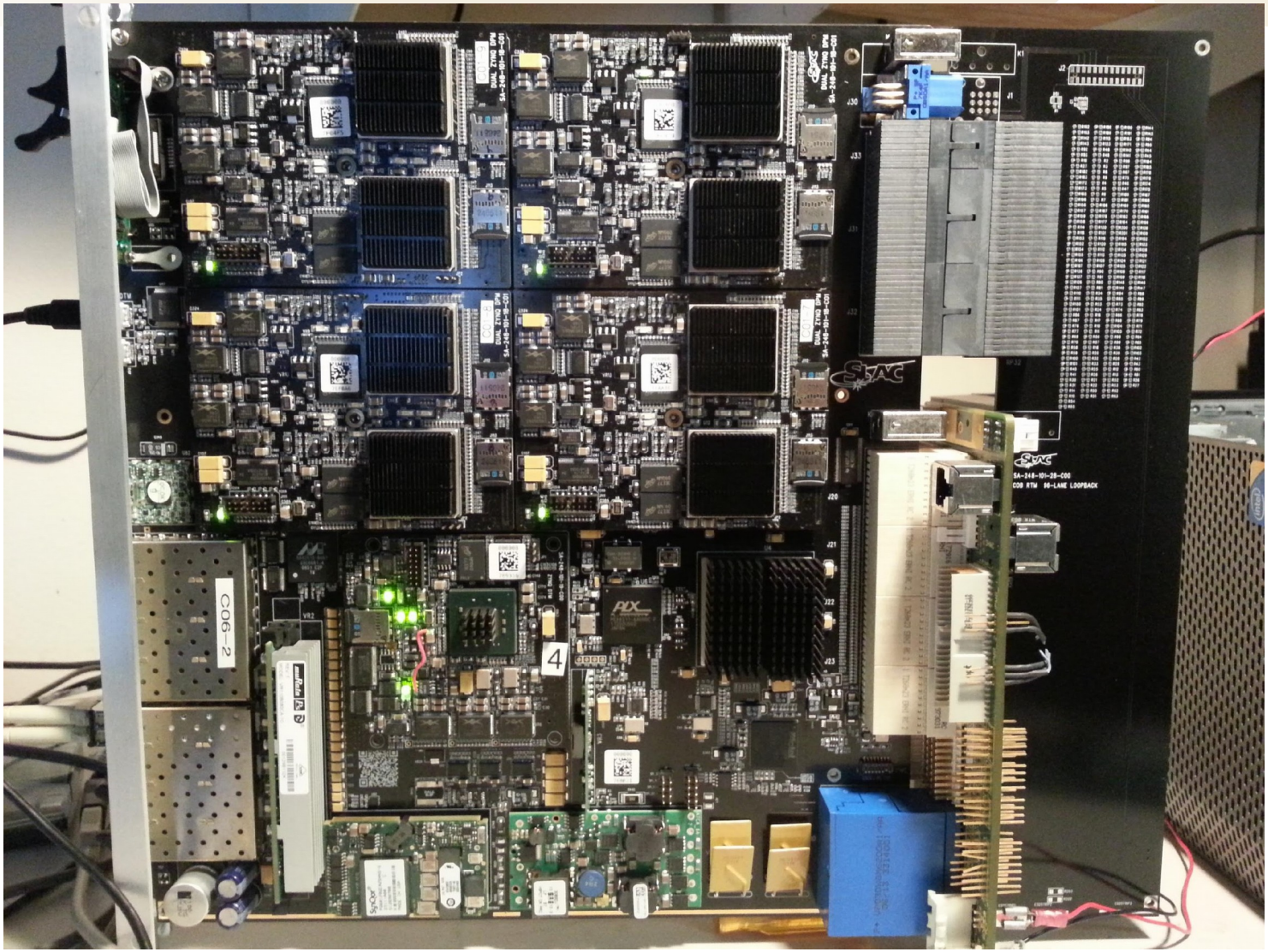
Raw ADC data rate (Gbps)	
Per hybrid	3.33
Per L1-3 Front end board	10
Per L4-6 Front end board	13

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

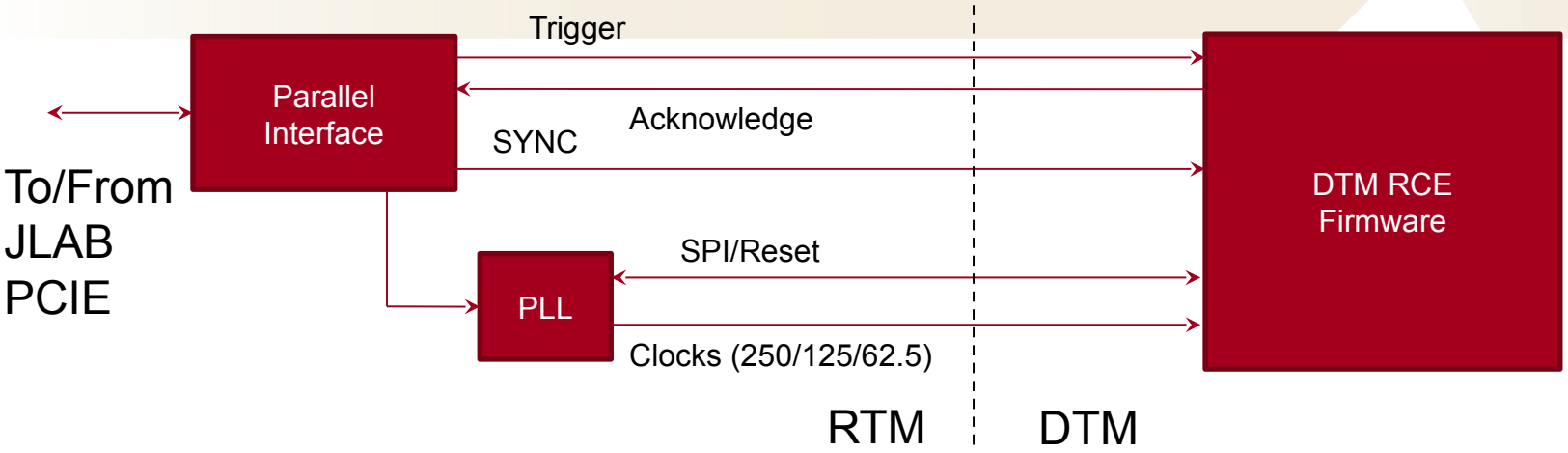
RCE GEN3 COB



SVT RCE Allocation

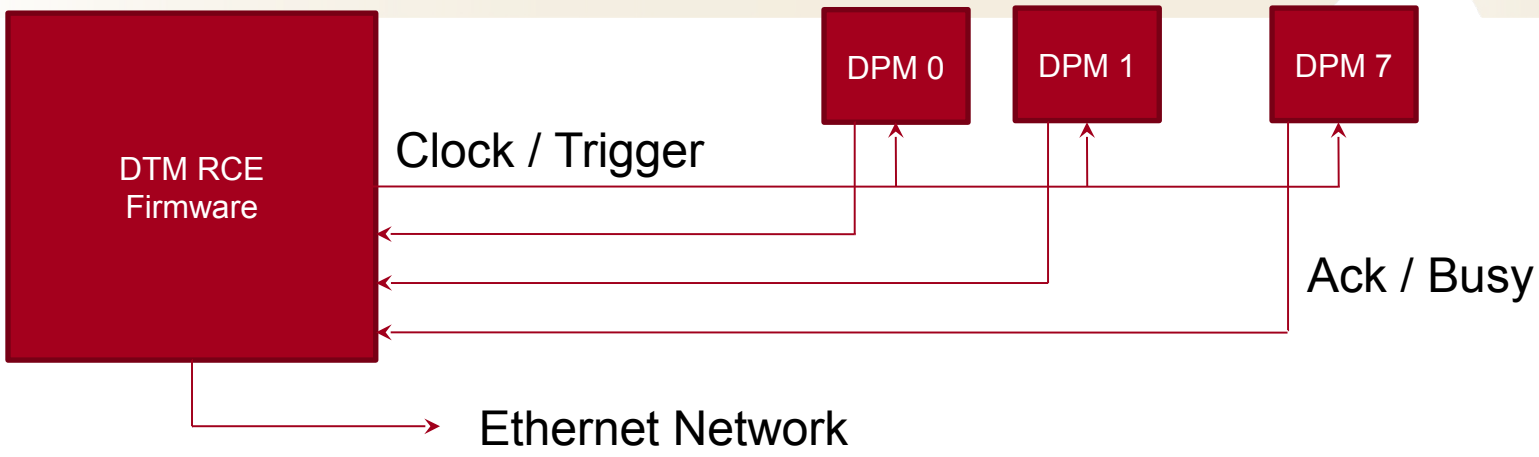
- Two COBs utilized in the SVT readout system
  - 16 RCEs On DPMs (2 per DPM, 4 DPMs per COB)
  - 2 RCEs on DTMs (1 per DTM, 1 DTM per COB)
- 7 RCEs on each COB process data from ½ SVT
- 8th RCE on COB 0 manages all 10 FE Boards
  - Configuration and status messages
  - Clock and trigger distribution to FE boards & hybrids
- 8th RCE on COB 1 is not used

SVT Trigger Interface



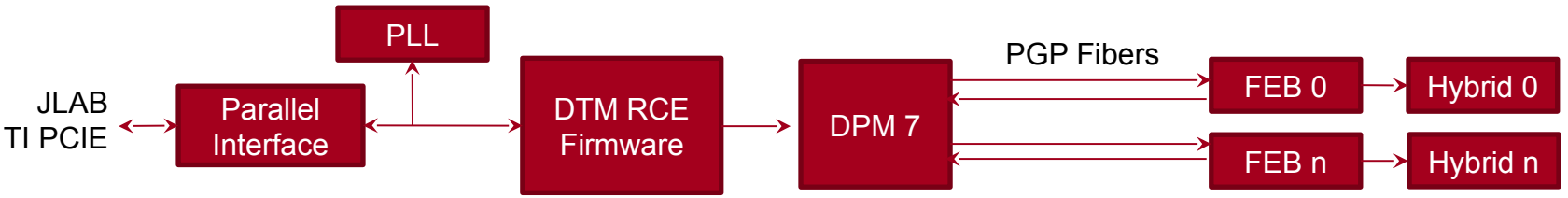
- Parallel interface and PLL exist on new RTM
- Fully allocated available signals between RTM and DTM
  - 1 high speed pair for trigger & SYNC
  - 1 low speed pair for SYNC
  - 2 low speed pairs for PLL SPI and Reset signals
  - 3 low speed pairs for PLL generated clocks (250/125/62.5 Mhz)

SVT Trigger Distribution



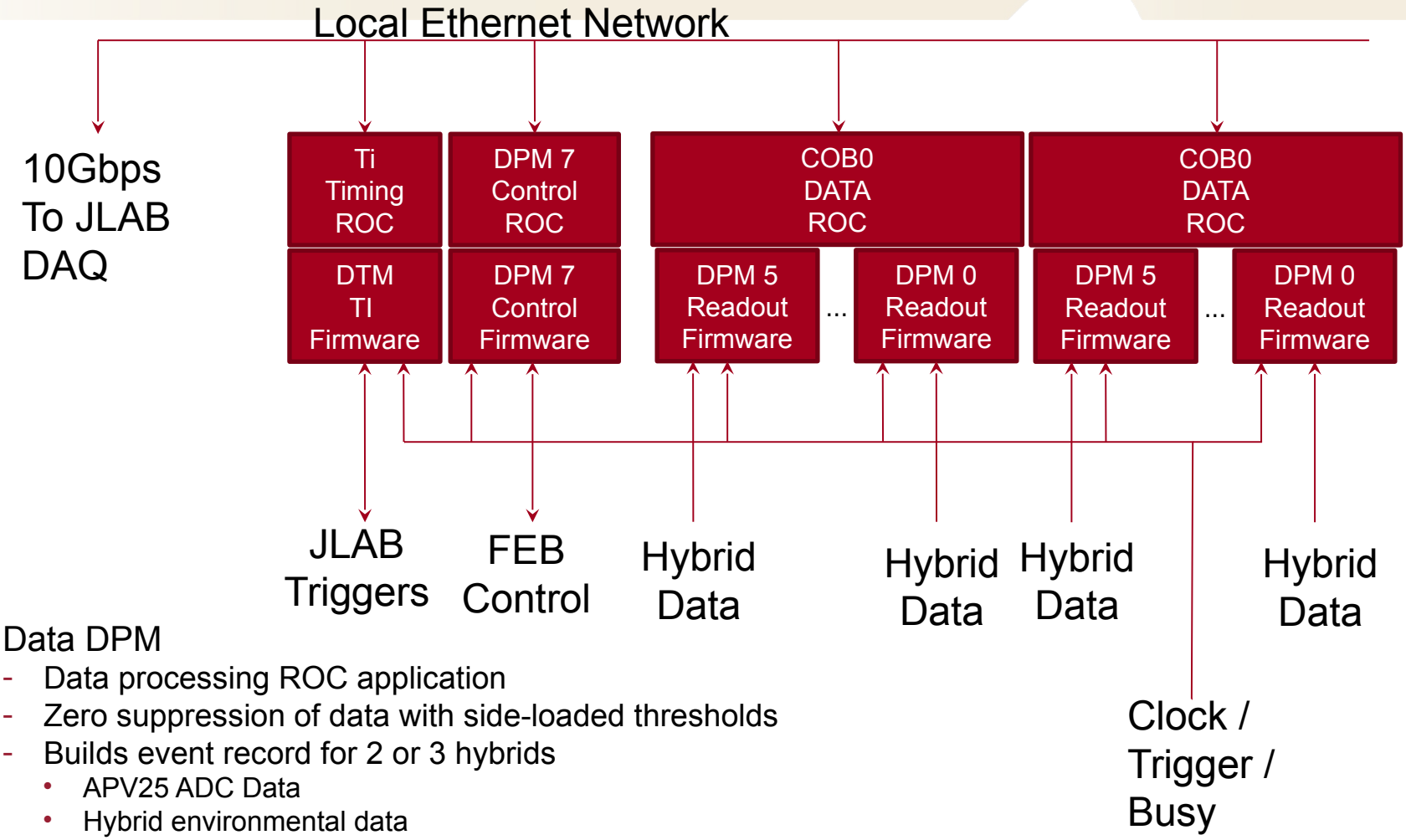
- DTM FPGA has ability to distribute clock and trigger to DPMs
  - Clock and trigger wired as fan out to DPMs
  - Individual feedback signals from each DPM
- 1 pair for clock fan out
- 1 pair for trigger fan out
  - 125Mhz serial protocol transfers 8-bit codes (easily expanded to longer words)
  - Used 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



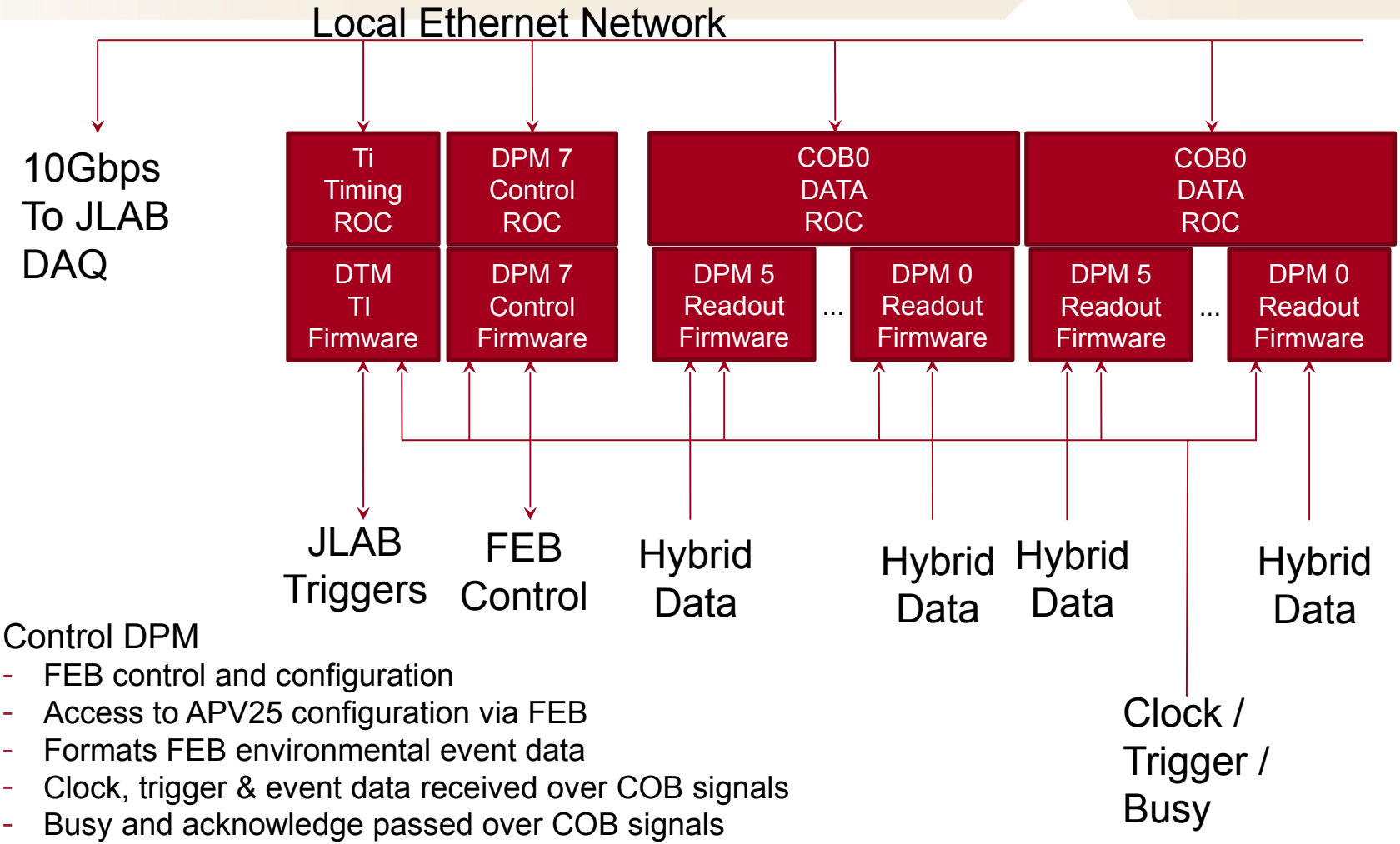
- Control DPM forwards timing information to front end boards over PGP
  - Clock encoded into serial data stream which the front end board recovers
  - Fixed latency path for encoded PLL reset and trigger signals

ROC Instances On SVT



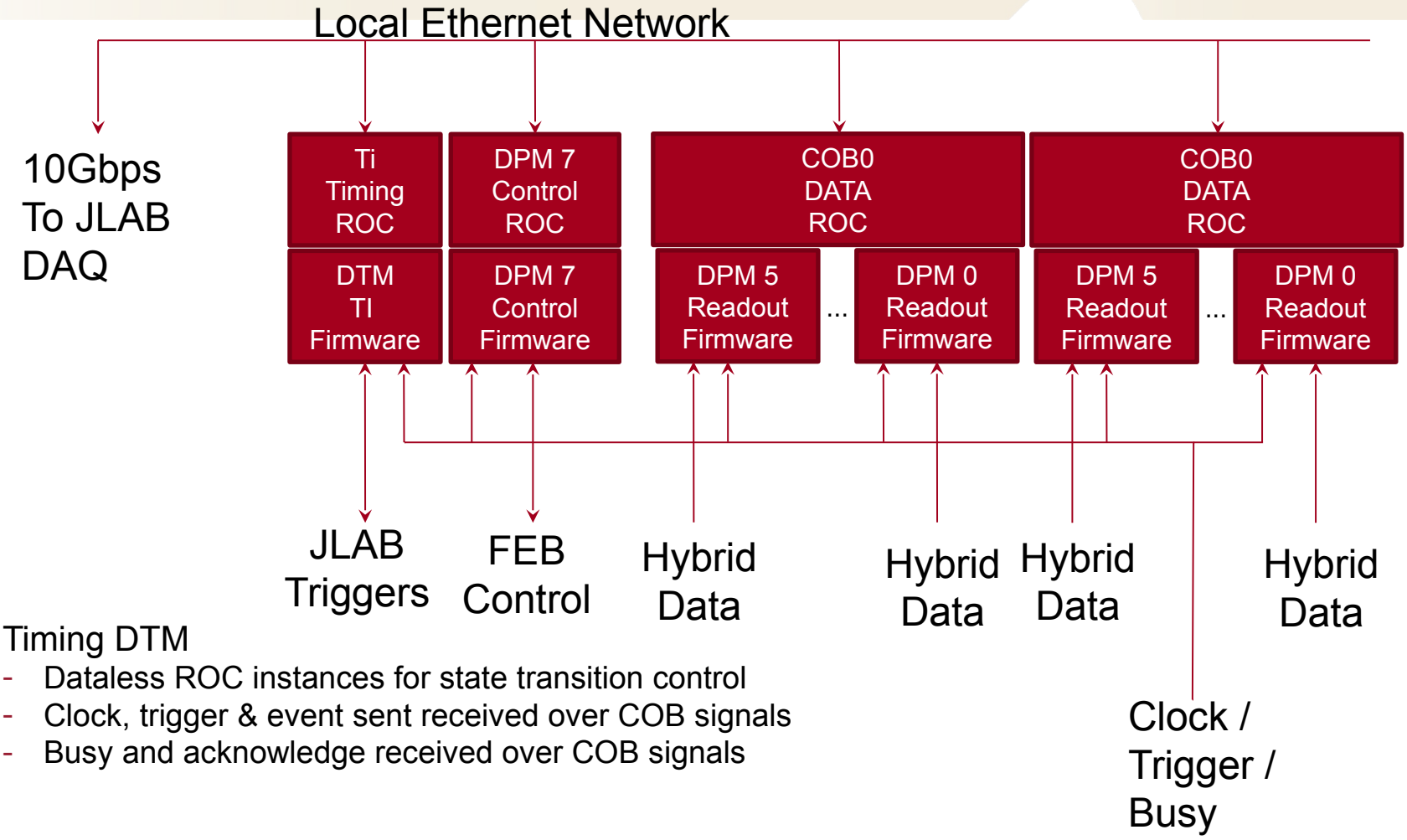
- Data DPM
  - Data processing ROC application
  - Zero suppression of data with side-loaded thresholds
  - Builds event record for 2 or 3 hybrids
    - APV25 ADC Data
    - Hybrid environmental data
  - Clock, trigger & event data received over COB signals
  - Busy and acknowledge passed over COB signals

ROC Instances On SVT



- Control DPM
  - FEB control and configuration
  - Access to APV25 configuration via FEB
  - 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

SVT DAQ Updates

- Add new layer 0 hybrids (only four APVs)
- Update firmware to latest build system and common libraries
- Add bootloader image to FEB, enabling remote firmware downloads
- Updated control software to python based Rogue platform
- Moving from multiple ROCs (one per RCE) to two ROCs on linux server (one ROC per COB)

- Built RCE system at SLAC for development

Firmware - Update libraries

SLAC

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 was necessary to maintain ability to develop and maintain system.

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

FEB Firmware - Bootloader

SLAC

- Added 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 simple, stripped down FEB image loads first
  - After 10 seconds, it loads the full FEB image
    - Can disable this via register access on simple firmware
  - Only the full 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
  - Take software triggered local runs for quick diagnostics
- Status – Mostly done. Still ironing out configuration procedure and integrating into EPICS.

Move from One ROC/RCE to One ROC/COB

SLAC

- Old system had 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

- Now we will have a single ROC instances per COB, hosted on a Linux server.
  - ROC on the linux server will execute state transitions through remote python calls to pyrogue running on each RCE
  - 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
- Currently, this is a very active development!
  - Can step through all state transitions with CODA
  - Data gets stuck somewhere in the pipeline

RCE Software Update

Running rogue based HPS software on the RCE Crate

- Rogue compiles and runs on the RCE nodes
- Need to configure bridge to CODA - ROC through shared memory
- Daemon rogued manages running instances on each of the RCE nodes.
- Status – Done. Stable and have pushed some minor updates

RCE at JLAB

SLAC

Started development with RCE Crate at SLAC

- Moving system to JLab, still one COB+RTM at SLAC
- COB+RTM shipped back to JLab during SVT install
- Ryan, Ben and JJ still doing some dev at SLAC
- Rogue installed and running
- Actively developing integration into CODA
  - Stepping through state transitions
  - Not crashing when run starts
  - Data getting stuck in pipeline, actively being debugged now

Summary

- Have seen many hiccups while upgrading system
- Seeing issues with a small fraction of the installed system, detailed more in Tim's talk
- Still ironing out configuration procedure
- Will start testing CODA integration at JLab soon!
  - I will be here all of next week to help with this

- Thanks for your attention!