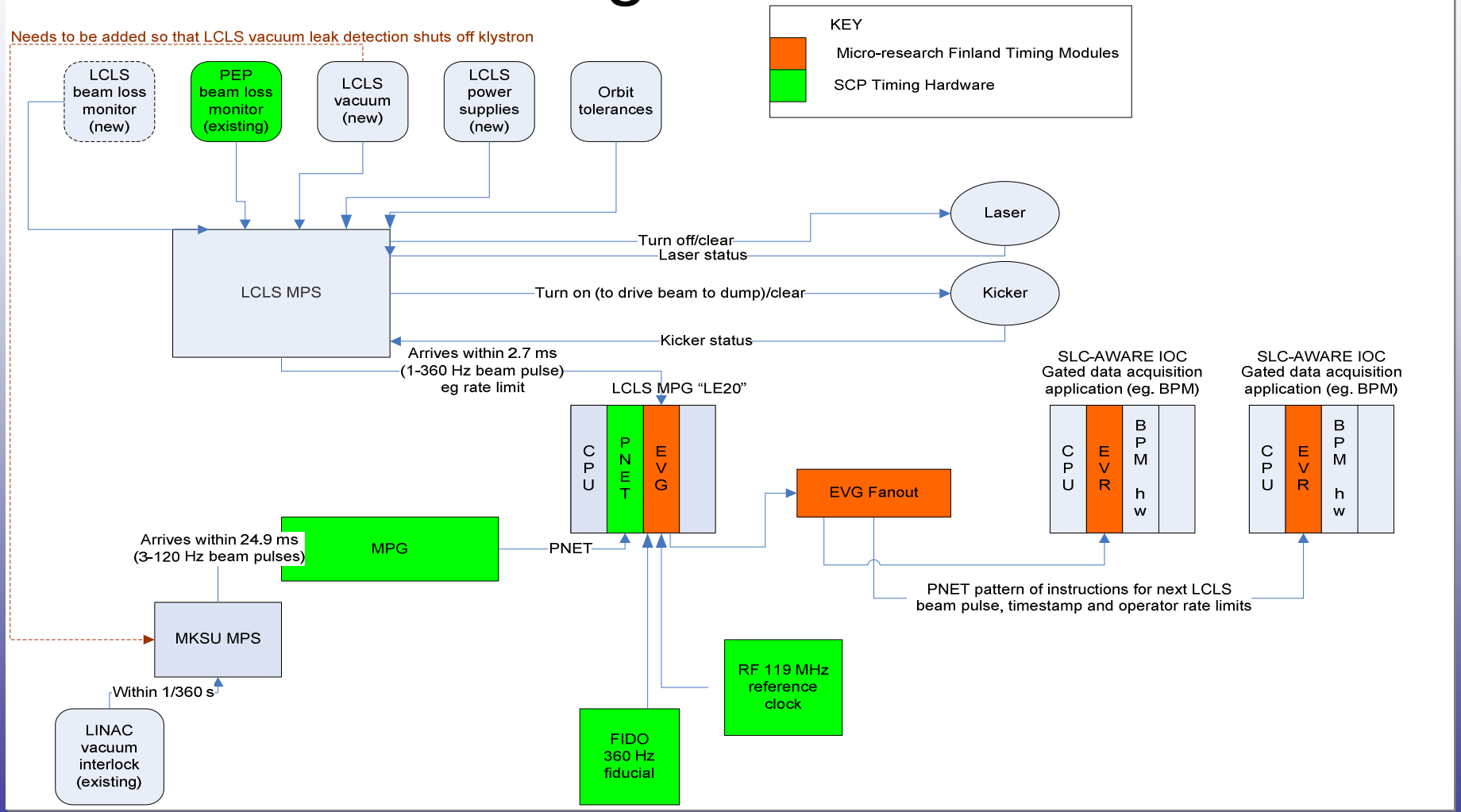
Timing + LLRF on RTEMS

- Overview
  - rtems 4.7 and EPICS R3.14.8.2
  - Timing
    - Target="RTEMS-beatnik" PowerPC mvme5500/6100 hybrid
      - Hardware
        - VME PNET Receiver
        - EVG200 with up to 2K data buffer transfer
        - VME-EVR200 with up to 2K data buffer transfer
        - PMC-EVR200 with up to 2K data buffer transfer
  - LLRF
    - Target="RTEMS-uC5282" m68k uCdimm 5282 Coldfire
      - Hardware
        - PAD Phase and Amplitude Detector – SLAC design
        - PAC Phase and Amplitude Controller – SLAC design

Dayle Kotturi
dayle@slac.stanford.edu

Timing on RTEMS



Dayle Kotturi
dayle@slac.stanford.edu

VME PNET Receiver

- Driver support
  - Init:
    - rc = devRegisterAddress("pnet", atVMEA24,
    - vmePnetAddr, PNET_DATA_NUM_BYTES,
    - (void*)&pLocalBuf);
    - rc = devConnectInterruptVME(PNET_IRQ_VECTOR, pnetISR, 0);
    - rc=devEnableInterruptLevelVME(PNET_IRQ_LEVEL)
  - ISR:
    - for (ii=0; ii<4; ii++) {/* hdr is in first 4 longs, 0-3, and is ignored.
    - set initial ii val to 0 if hdr needed
    - data is in last 4 longs, 4-7, and is of interest */
    - pnet_messages[next_message].data[ii] = in_be32(&(pLocalBuf->data[ii]));
    - }
    - /* NOW update what current_message is (so that it will be what's accessed */
    - current_message = next_message;

Dayle Kotturi
dayle@slac.stanford.edu

VME PNET Receiver data display

Auto Save

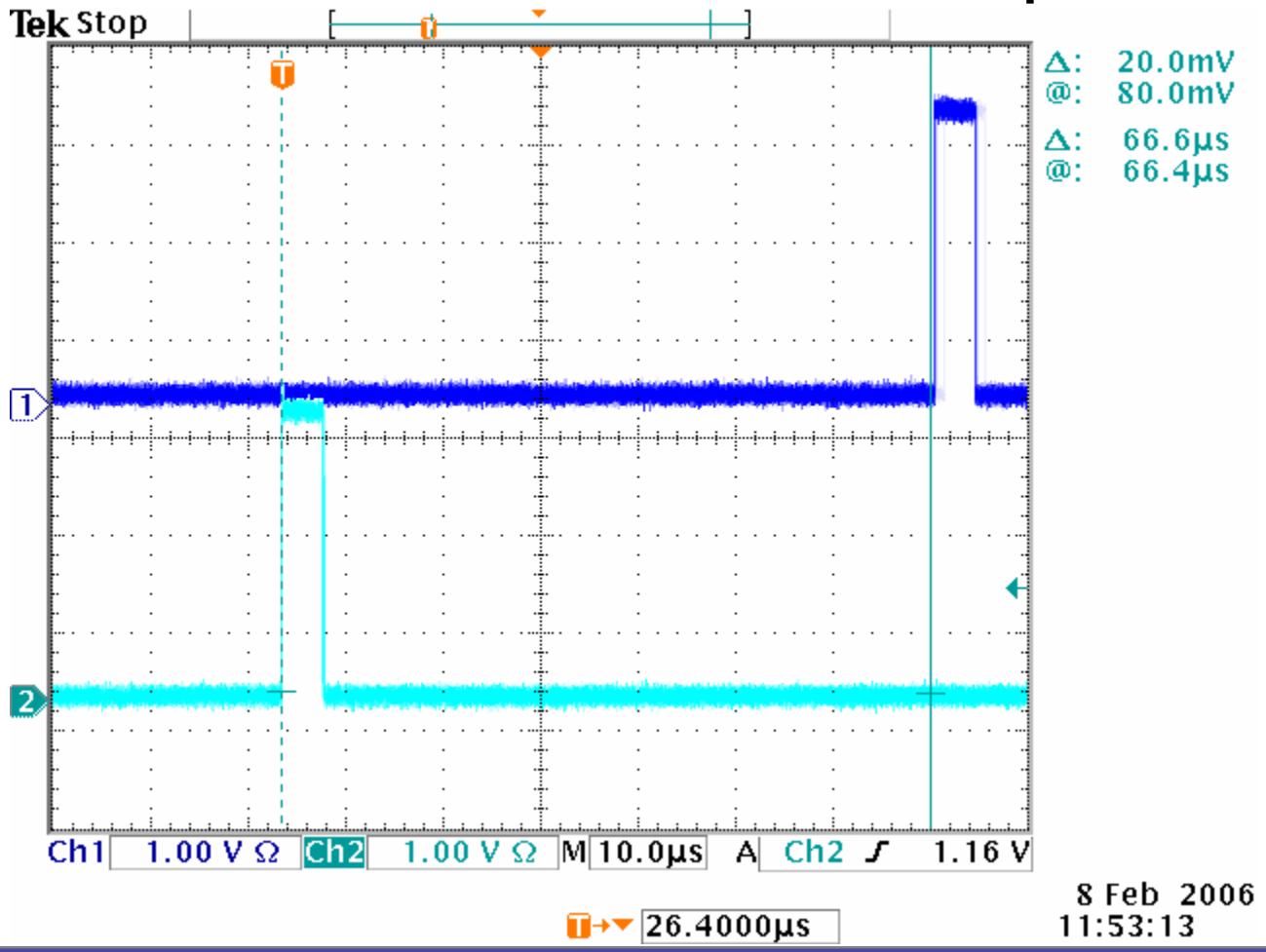
PNET:LE20:1 PNET Buffer

EXIT

Arrival time	Now	1/360 s ago	2/360 s ago	1/120 s ago
Execution time	1/120 s from now	2/360 s from now	1/360 from now	Now
Data Bits 0-31	-2147481856	-2147483392	-2147483648	-2147483648
Data Bits 32-63	1048644	2	100663361	32
Data Bits 64-95	1310720	0	95232	0
Data Bits 96-127	1610612736	1073741824	805322736	-1073741824
YY	7	1	10	0
PP	0	0	0	0
PULSID	58652	58651	58650	58649
Time of day	04/19/06 13:49:55	04/19/06 13:49:55	04/19/06 13:49:55	04/19/06 13:49:55

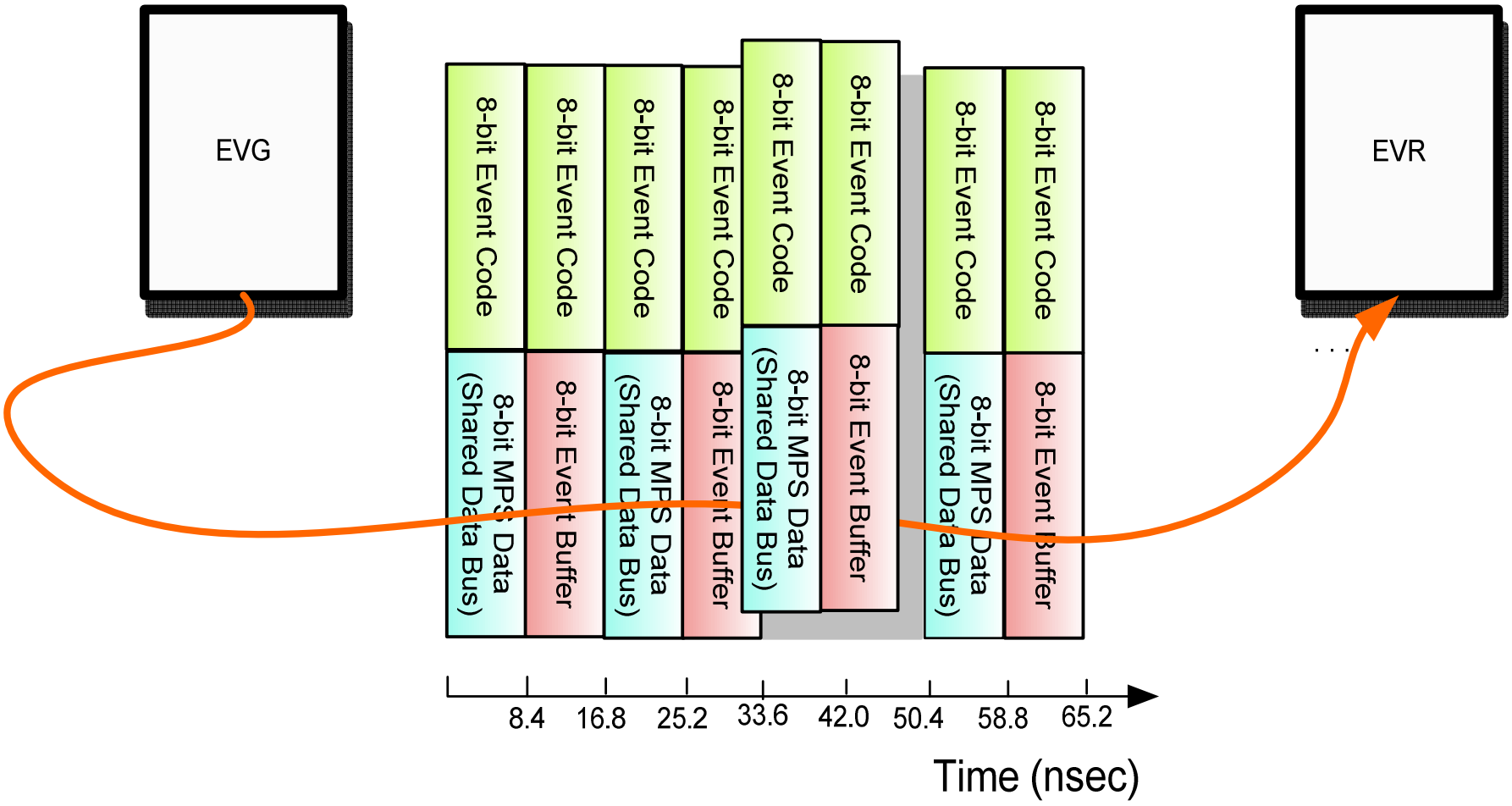
Dayle Kotturi
dayle@slac.stanford.edu

EVG-EVR transfer time for 16B data buffer = 66.6 µsec



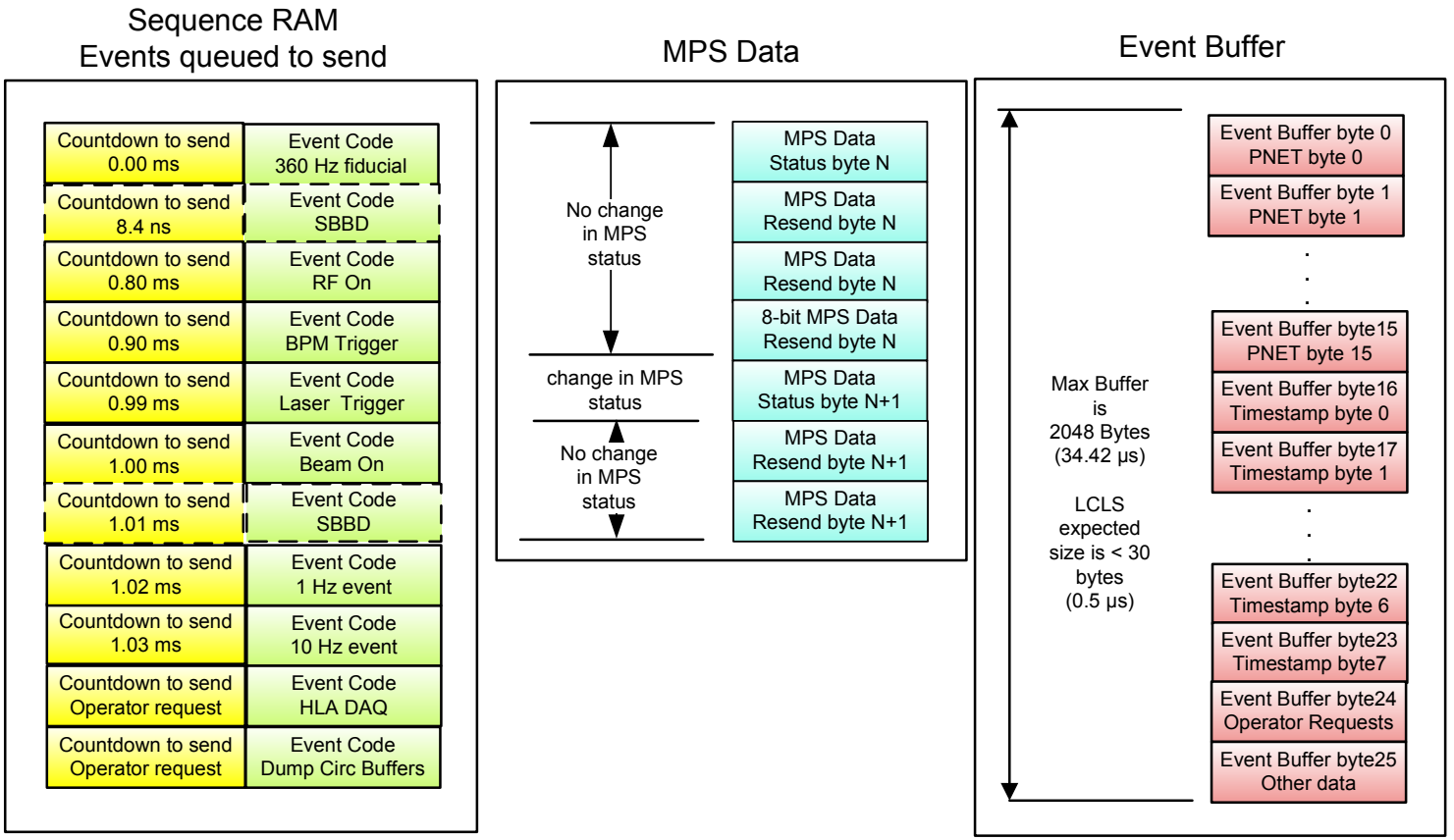
Dayle Kotturi
dayle@slac.stanford.edu

EVG-EVR data transfer sequencing



Dayle Kotturi
dayle@slac.stanford.edu

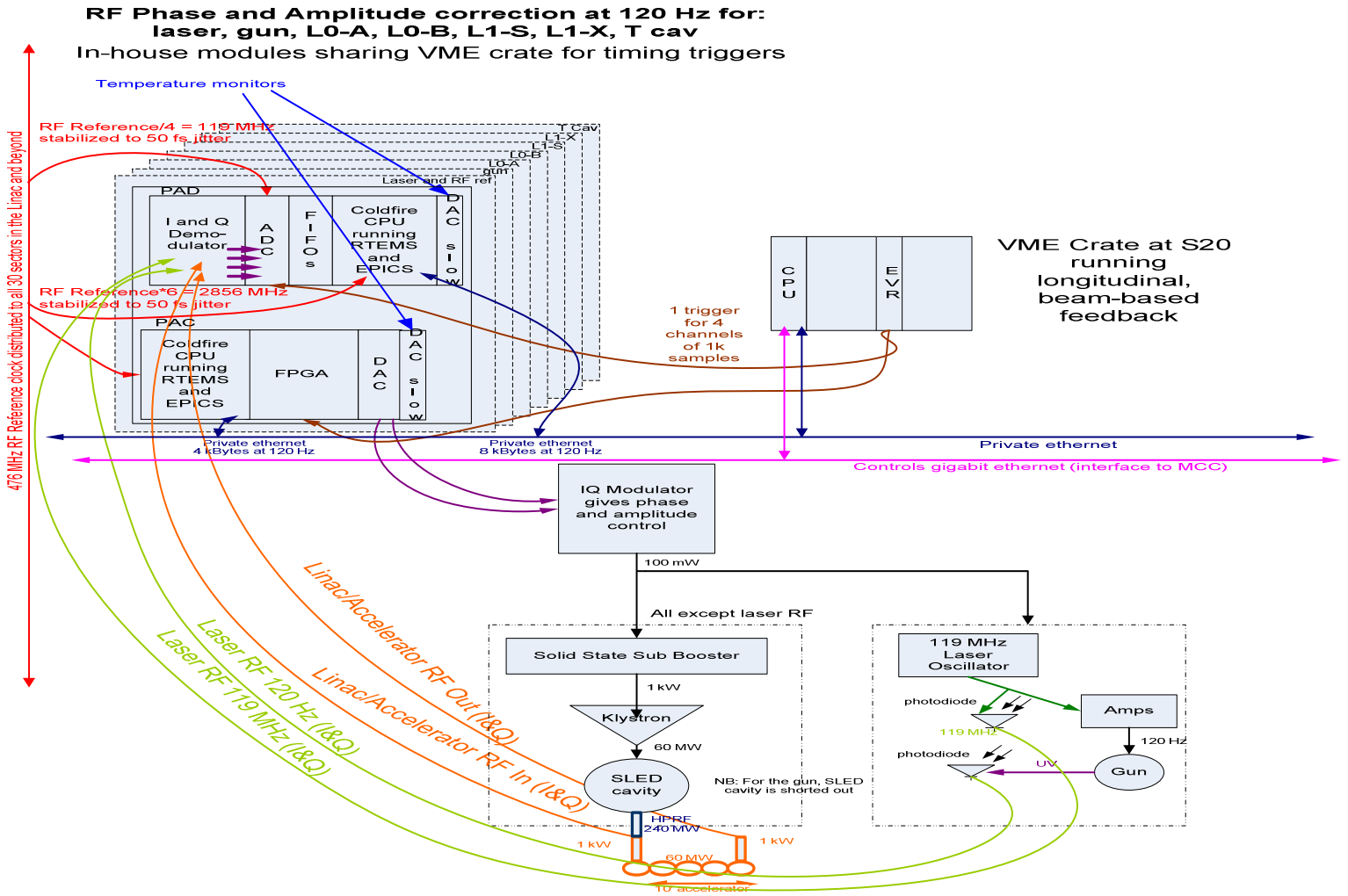
EVG-EVR data types



Two SBBD event codes shown. First is case to send beam to undulator; second is timed to prevent beam from reaching undulator. It's one OR the other, per seq RAM.

Dayle Kotturi
dayle@slac.stanford.edu

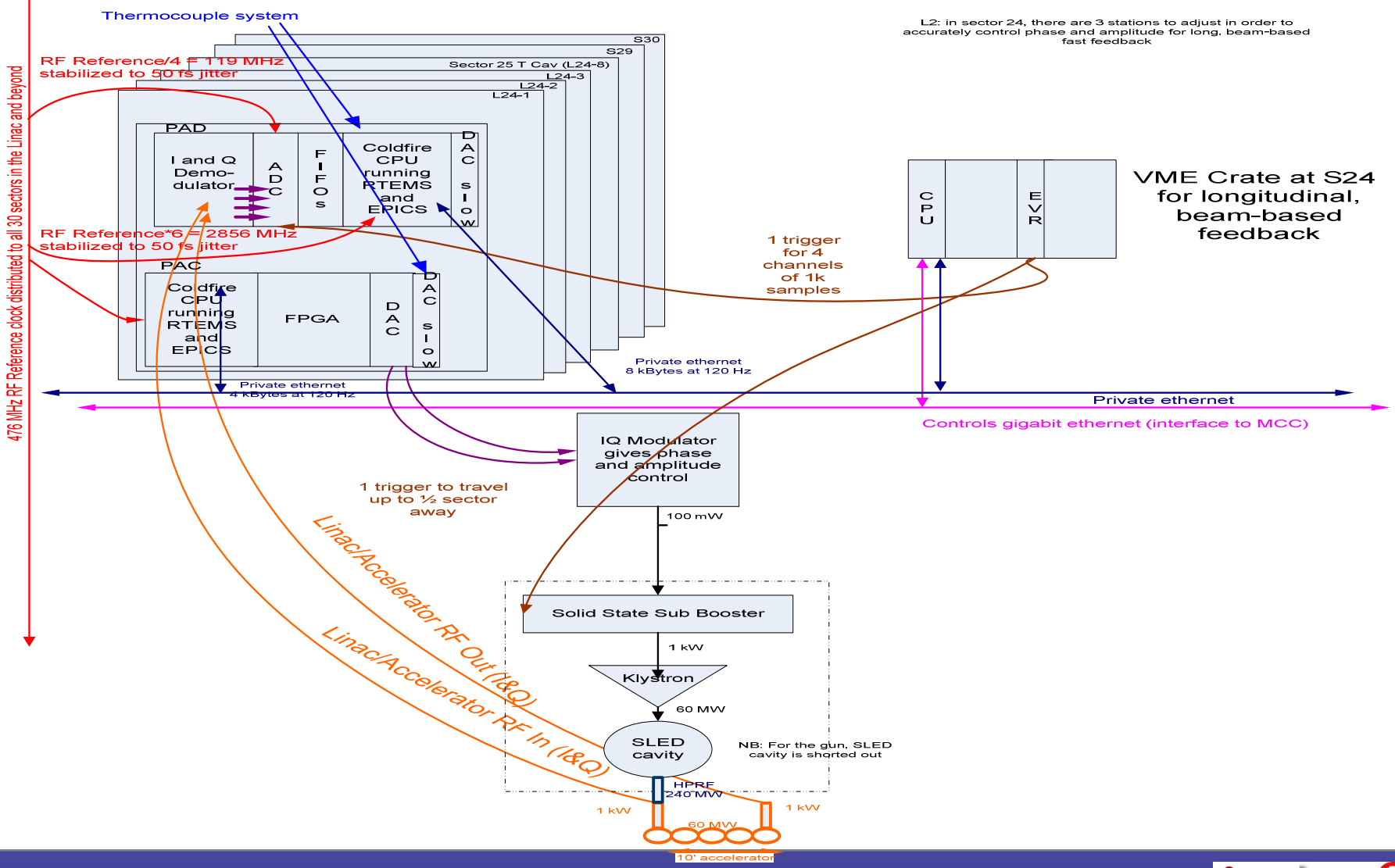
LLRF on RTEMS



Dayle Kotturi
dayle@slac.stanford.edu

RF Phase and Amplitude correction at 120 Hz for: L2, S25 Tcav and L3

In-house modules sharing VME crate for timing triggers



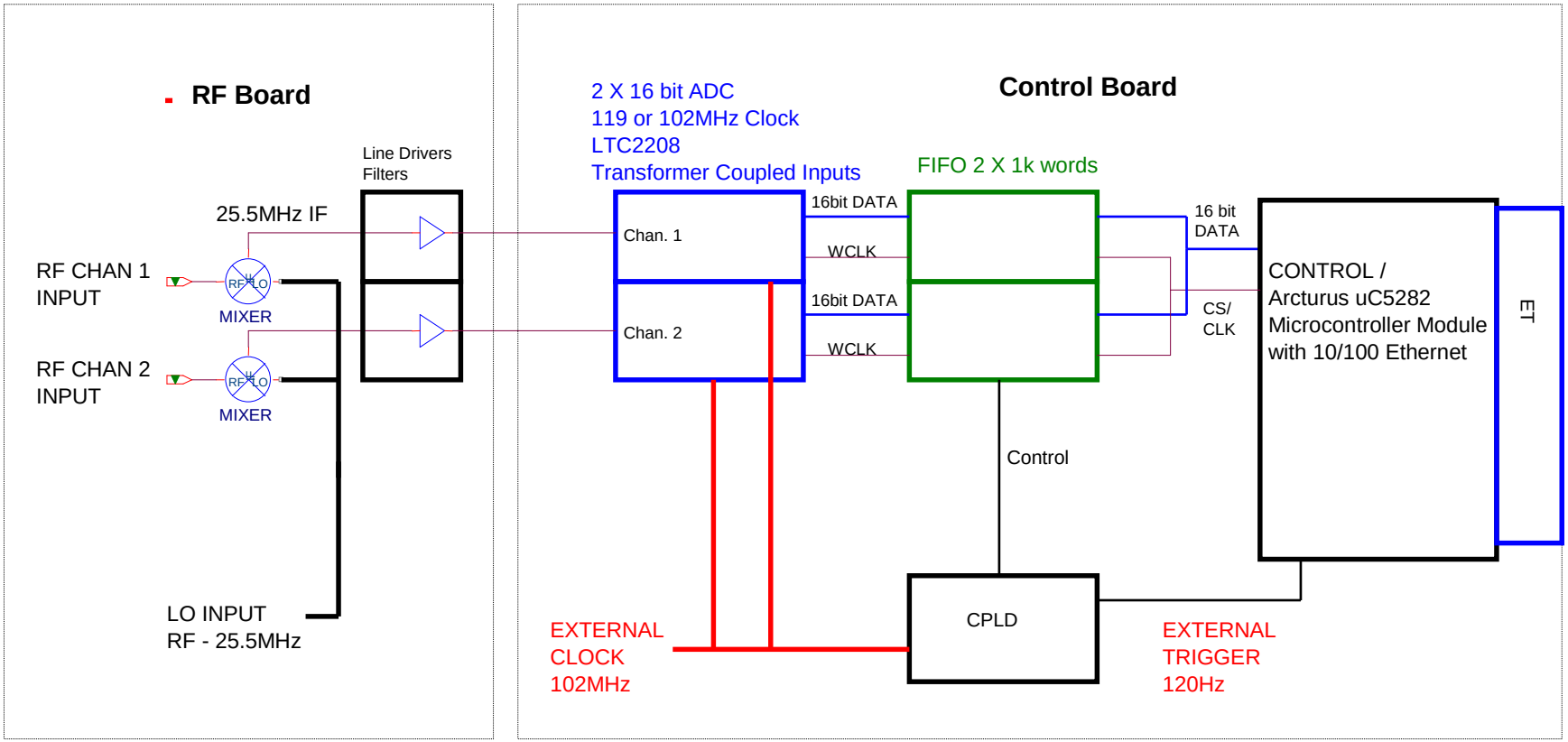
Dayle Kotturi
dayle@slac.stanford.edu

LLRF Component Types

- Three types of components: PAD, VME and PAC.
  - 1. PAD – the phase and amplitude detector uses an embedded IOC (uCdimm 5282 Coldfire)
    - Triggered at 120 Hz, PAD reads 4 channels of accelerator 119 MHz RF(I&Q) from ADCs via TI FIFOs into EPICS waveform record.
    - FIFOs are 65536 words long, but operationally we use fewer (of order 1k), the size depending on the fill time of the cavity
    - ADCs are LTC2208 (16 bit, 130 MHz)
    - Hardware design and CPLD programming by Ron Akre (SLAC)
    - Last summer, no commercial VME ADC board could match these specifications, so we opted for in-house solution.
    - Additional advantage: digitizers can be placed next to the low noise RF components (eliminates transmission of low noise analog signals outside the chassis)

Dayle Kotturi
dayle@slac.stanford.edu

PAD Block Diagram (2 channels)



Dayle Kotturi
dayle@slac.stanford.edu

LLRF Component Types

- 2. VME Feedback Crate uses a mvme6100
  - Provides timing trigger to the PAD from EVR200
  - Receives averaged I&Q (EPICS ai records) from PAD
  - Applies phase and amplitude adjustments from global or local feedback
  - Sends new I&Q (EPICS ao records) to PAC
  - Provides timing trigger to PAC where corrected waveform is sent out (and NEXT PAD values get read…)

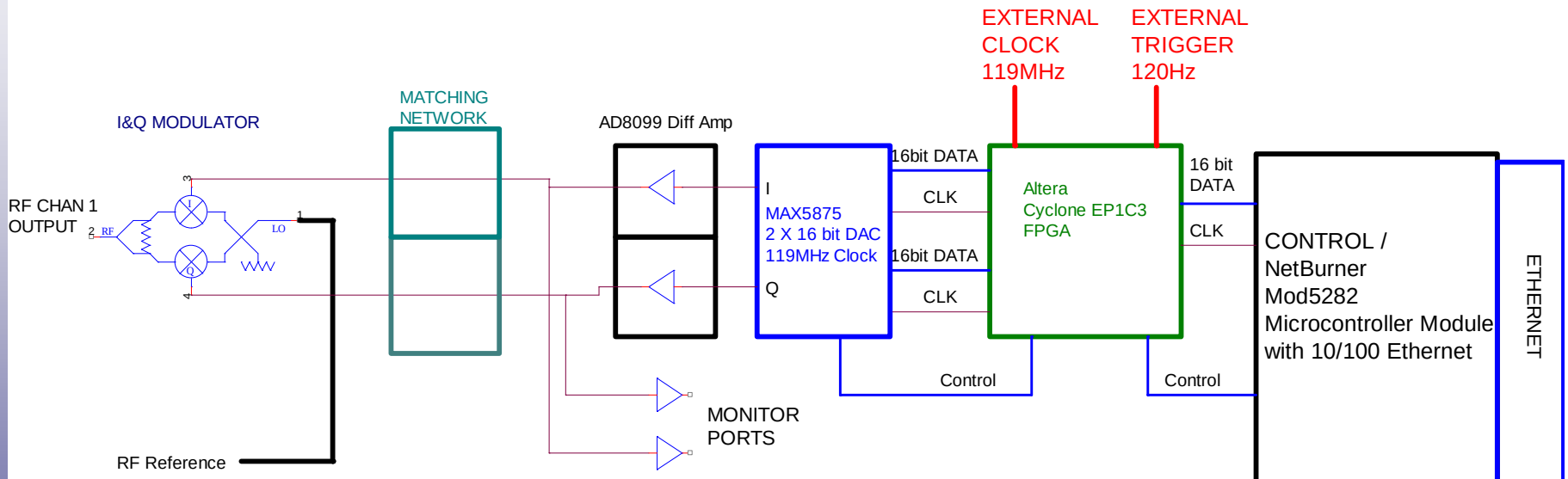
Dayle Kotturi
dayle@slac.stanford.edu

LLRF Component Types

- 3. PAC – the phase and amplitude controller uses an embedded IOC (uCdimm 5282 Coldfire)
  - receives the adjusted I&Q values (EPICS ai records) and computes the waveform to be sent out on next 120 Hz trigger
  - drives an IQ modulator
  - used for control of the LLRF to the solid state sub-booster
  - hardware design and FPGA programming by Jeff Olsen (SLAC)

Dayle Kotturi
dayle@slac.stanford.edu

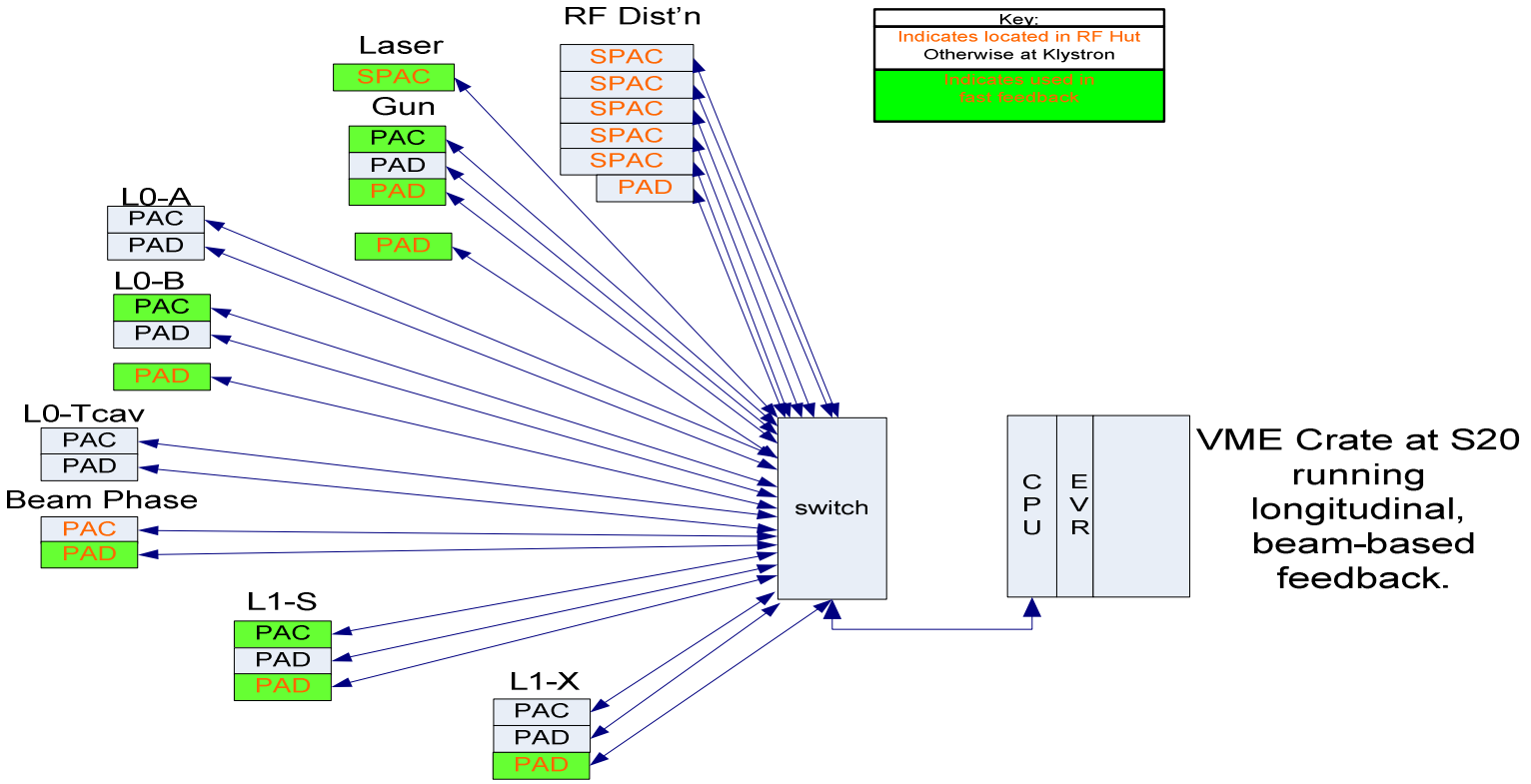
PAC Block Diagram



Dayle Kotturi
dayle@slac.stanford.edu

LLRF Component Instances

RF phase and amplitude correction for LCLS LINAC S20

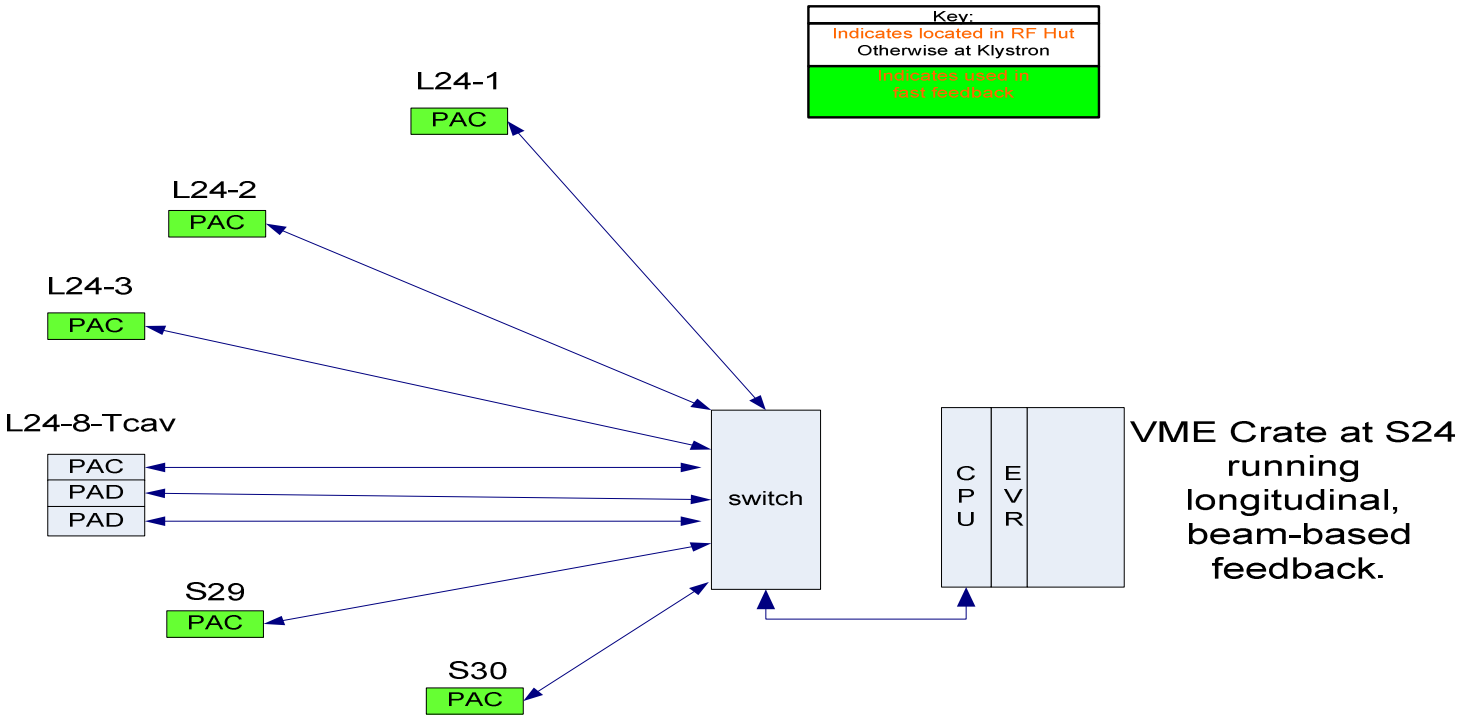


	S20
Fast PACs:	7
Slow PACs (SPACs):	6
PADs:	13
VME crates:	1

Dayle Kotturi
dayle@slac.stanford.edu

LLRF Component Instances

RF phase and amplitude correction for LCLS LINAC S24



	S24
Fast PACs:	6
Slow PACs (SPACs):	0
PADs:	2
VME crates:	1

Dayle Kotturi
dayle@slac.stanford.edu

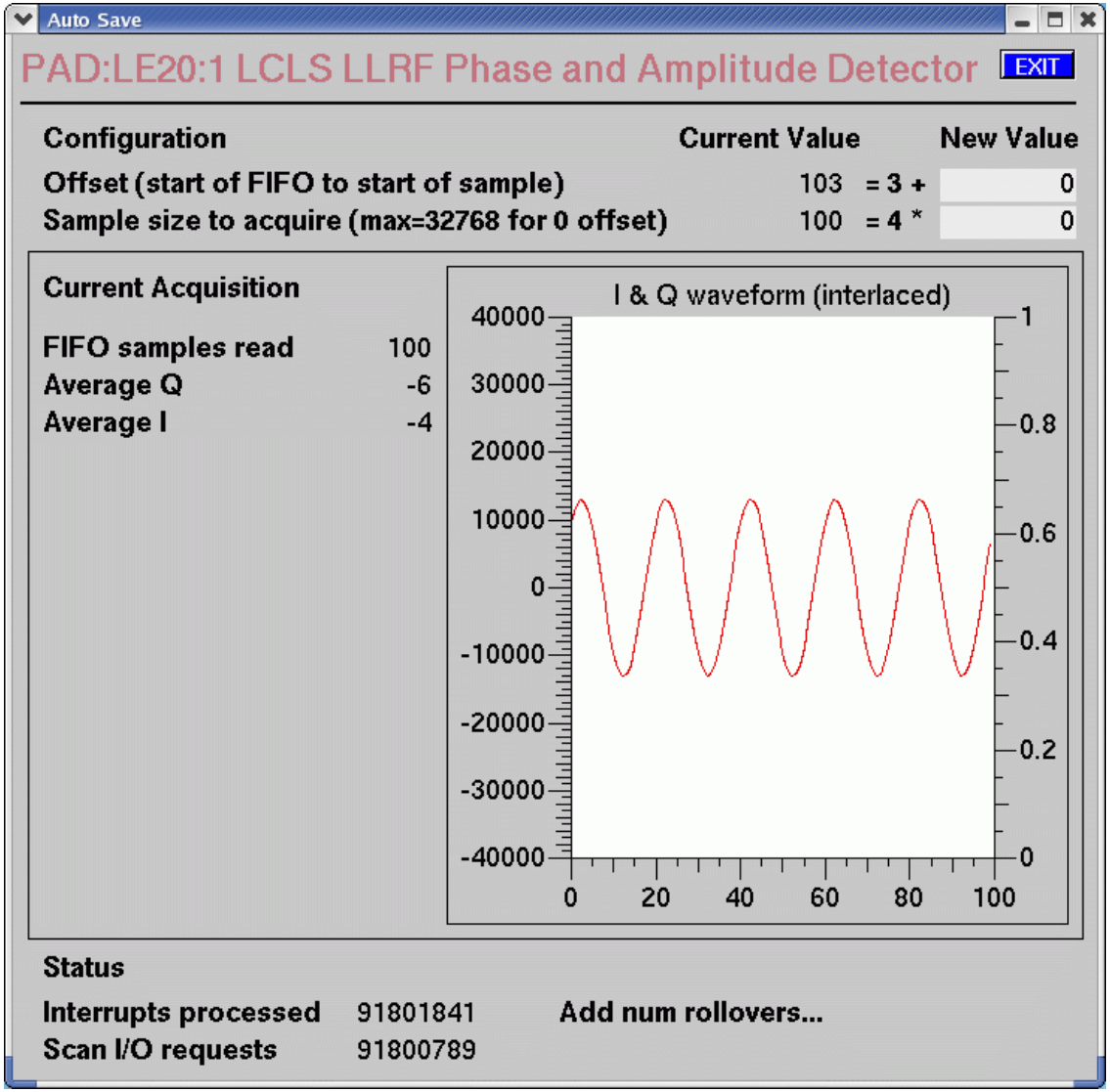
PAD driver details

- Driver support
  - init: sets up dacq task
  - ISR:
    - /* Announce that data is available for read */
    - epicsEventSignal(waitForData);
    - clear the interrupt
  - Dacq task:
    - waitStatus = epicsEventWait(waitForData);
- Device support
- Database records

Dayle Kotturi
dayle@slac.stanford.edu

PAD EDM GUI

Use this panel to change the size of and offset to the sample to be averaged.



Dayle Kotturi
dayle@slac.stanford.edu

PAC details

- Driver support: update FPGA calculation
- Device support
  - init: waveform record bptr is freed and set to mem-mapped FPGA space
  - write: waveform is recalculated and result stored in FPGA
- Database records
  - in ops, new adjustment FLNKs to waveform
  - in cal, new gain or offset FLNKs to waveform

Dayle Kotturi
dayle@slac.stanford.edu

PAC edm control

- There are 2 EDM screens
  - for startup and calibration
    - at startup the amplitude of the calibration waveform can be modified, as well as number of points in wf
    - in calibration, the gain and offset of I&Q can be modified
  - for operation
    - I&Q can be adjusted (scalar applied to WF[i] *gain before offset is added)

Dayle Kotturi
dayle@slac.stanford.edu

RTEMS lessons learned

Dayle Kotturi
dayle@slac.stanford.edu