L1CAL Control Software and ADF-2 Testing

Philippe Laurens
11-Feb-2005
Run IIb L1 Calorimeter Trigger Control Path

L1Cal Expert Programs

L1Cal TCC

Bit3

COOR

Monitoring Clients

Ethernet

L1 Calorimeter Trigger Control Path

V.I. Slave  
ADF  
...  
ADF  
ADF Crate  
#1 of 4  
VME-6U

V.I. Slave  
ADF  
...  
ADF  
ADF Crate  
#4 of 4  
VME-6U

Bit3

V.I. Master (to ADF)
SCL Fanout (to ADF)
SCL/VME (to TAB)
V.I. Master (to VRB)

L1Cal Control Crate  
VME-9U

V.I. Slave  
VRB  
...  
VRB  
SBC  
L1Cal Readout Crate  
VME-9U

TAB  
...  
TAB  
TAB/GAB
Crate Custom-9U

Optical Fiber Splitters to L2Cal Crate

L2

L3

P. Laurens  
Rev: 08-Oct-2003
MSU ADF Test Setup

Waveform Generator (FNAL) -> Signal Multiplexer (FNAL) -> Graphical User Interface (Python) -> L1Cal TCC

RS232

32 Channels of One ADF-2 Card

Bit3 -> L1Cal llb Control Program (C++) -> XML over ITC

Channel Link Tester (Saclay) -> One ADF-2 Card Channel Link Output

RS232
L1Cal llb Control Software

• **Partitioned**
  – Control Code vs. GUI

• **Operating System Independent**
  – Windows & Linux
  – Python & C++
  – Use D0 products ITC and Thread_Util
L1Cal Ilb Control Software

• L1CAL Ilb Control Program

  – Use C++ for robustness and execution speed
  – Control and monitor both ADF and TAB/GAB
  – Only software actually performing VME IOs
  – Interface presenting one L1CAL system to COOR
  – Serve Monitoring Information to Host Monit Clients
  – Ported in large part from current L1FW+L1CAL TCC
  – Allow local command files
  – Keep running Logfile of all actions
L1Cal IIb Control Software

- Py_VME GUI
  - All new for IIb
  - Use Python with Tkinter for coding flexibility
  - Zero, One, or more GUI connected at given time
  - Not needed for COOR control
  - Not needed for monitoring operation
  - On Same or Remote computer
  - Greatly extendable with external python command files (of arbitrary complex)
L1Cal Ilb Control Software

- **Extensions with Python Command Files**
  - Call functions to perform any action the GUI supports: R/W, Config FPGA, program DAC,...
  - Retrieve and use the reply from the Control Program (e.g. value read)
  - Recursive call to lower level command files while passing parameters (e.g. card address)
  - Control Waveform Generator and Multiplexer
  - Request interactive user input (e.g. card S/N)
  - Write test logfiles, any other files e.g. plot files
  - Execute OS commands, e.g. view plots
ADF-2 Production Testing

- Visual Inspection & Ohm meter
- Final Assembly (front panel, screws, etc)
- First Power Up: Configure Board Control PAL over JTAG
- A few VME IO to BC PAL registers
- Check PLL for Frequency Locking to SCLD
- Configure Data Path FPGAs over VME to load test firmware [.5 MB]
- Random Register Test: VME IO reliability with on multiple boards [no error in 1E10]
- Program Pedestal DAC: Measure DAC to ADC ratio [~7]
- “Find_DAC”: DAC/ADC sanity check and tune ADC to mid-scale
- Channel Noise measurements (no input signal) [< 1/4 LSB]
- Measure Frequency Response of all channels (in-band, below and above)
- Crosstalk tests 32 input x 32 output channels [< 1/4 LSB worst case at 2 MHz]
- Channel Link tests (~5mn @MSU, extensively w/TAB @Final) [no error in 1E15]

- Misc Problems found so far: a few shorted or open traces on ADC data to FPGA
Frequency Response of multiple ADF-2 Cards
(Waveform Generator set at 1800mV)