

ECal/Trigger Setup Procedures and Monitoring

Sho Uemura

SLAC

Overview

- Primary use of ECal is trigger
- Setup/checkout: needs to be done at start of running to ensure correct operation
- Monitoring: needs to be done continuously during running to verify correct operation

ECal requirements

- Low noise
 - ▶ Setup: calibrate with window mode run
 - ▶ Monitoring: occupancy
- Reasonable response (gain)
 - ▶ Setup: calibrate gains, preferably without tracker
 - ▶ Monitoring: E/p, E vs. x, occupancy
- Identify bad crystals/channels
 - ▶ Monitoring: occupancy, in-time hits

FADC monitoring

- FADC channels map correctly to crystal IDs
 - ▶ Check for clusters across edges (done)
- Hits are timed in: with trigger, within and between crates
 - ▶ Trigger time vs. cluster time (done)
 - ▶ Relative time of cluster pairs
- Pulse pileup is not causing us to lose hits
 - ▶ How?

Trigger monitoring

- Clusterer works according to spec: use clusters in event stream
 - ▶ Check for clusters across adjacent crystals
 - ▶ Run clustering on readout hits
 - ▶ Check cluster time coincidence
- Trigger decision is correct
 - ▶ Need trigger bits for each of the physics cuts; run singles trigger (preferably in parallel with pair trigger) and check

ECal/FADC set-up

- Any chance of rough gain cal before run?
- Readout window (NSA, NSB): decide in simulation
- Crystal energy discriminator: decide in simulation (is there any reason not to set it low?)
- Readout threshold: decide in simulation (pileup vs. sensitivity trade-off), monitor for noisy crystals
- Calibrate pedestals and pulse shape

Calibrating ECal

- Two populations of single electrons can calibrate the regions where each dominates
 - ▶ Near beam spot: electrons at beam energy
 - ▶ Large angles on electron side (flip field to get both sides): use deflection angle to estimate momentum (E vs. X)
- Pairs? Tridents?
- Track-based calibration is probably better but we shouldn't rely on it

Tasks

- Simulation:
 - ▶ Trigger studies: optimize various DAQ parameters
 - ▶ Test gain calibration methods
- Monitoring:
 - ▶ Lots mentioned above
 - ▶ May want a “shifter-friendly” system that gives a “good/not good” status for each item
 - ▶ Ease of use, stability, speed (remote AIDA?)
- Many good entry points for hps-java initiates