
L2 Overview II, and Summary

James T. Linnemann
Michigan State University
Level 2 Review
Feb 6, 1999

L2 Maximum Event Sizes (FIFO size choice)

- Length = 16B(min) ... 4KB (max) X 16 events
 - includes 12B header and 4B trailer
 - source pads to multiples of 16B with zeros after trailer
 - VRB: 32KB or 64KB, but currently no raw data to L2!
 - 5 KHz max (Cypress) is $16\text{B}/\mu\text{s} \times 200\ \mu\text{s} = 3.2\text{KB}$
 - clearly issue of max, not mean!

<u>Actual Max Event</u>		<u>FIFO "event"</u>	<u>total</u>
FIC/CFT/PS	272B	.5KB	8KB
Cal/MBT	304B	4KB	64KB
Mu/SLIC	.3 to 3KB	.5KB	8KB
Global/MBT	2.3KB	4KB	64KB
	=255 tracks*8B		
	(255*16B = 4KB = STT?)		

SCL INITIALIZE

why we avoid it

- Needed if event fragments don't match
 - must clear all buffers EVERYWHERE and restart
 - violent: touches EVERY front end crate
- Avoidance:
 - redundancy header to trailer (protect 1-bit errors)
 - try to preserve event format (to find trailer)
 - try to preserve event boundary (else must re-init)
 - detect missed event boundary (end or begin)
 - send pads before End Event to reframe if needed

Monitoring (online, data flow)

- Every 5 seconds, via TCC/administrators
- Some by **L1 Scalers**, some by VME
 - **L1 Scalers** available even if alphas crash
- buffer occupancy for data flow diagnosis
 - lots of buffers, need to be able to look at them
 - in all cards owning buffers: FIC, SLIC, MBT, Alpha
 - but DMA: most events into **alpha's buffers**
- **time in state** (like L3 in Run I) in all alphas
 - idle, processing, waiting, interrupt...
- Global's pass fraction by bit #; events vs node

L1 Scalers for Alpha States

- ECL Gates: sampled every beam crossing
- Worker States (5)
 - wait/event, process, wait/admin, interrupt, collecting_status
- Admin States (6)
 - wait, reply/worker, manage_L3, interrupt, L2_Acc/Rej, collecting_status
 - multi-workers: wait more complex:
 - wait/event, wait/worker, wait/L2_Acc/Rej + Processing
- Shows time fraction in state
 - thus <time> in state

Alpha Buffer Monitoring (L1 Scalers)

- Alpha should be where events stack up
- Binned histo: # events in each buffer state
- buffer states monitored in Administrator
 - allocated, processing, wait/L2_Acc/Rej, wait/L3
 - to be allocated to worker n
 - free
- Each L2 crate: 22 scalers/admin + 1/worker

VME (“Slow”) Monitoring (Via Bit3, TCC)

- Exact event accounting from ALL cards
 - evaluated after marked event (CollectStatus)
- All non-Alpha Cards in L2 Crate
 - MBT, SLIC, FIC
 - State and Buffer Occupancy Sampled on board
 - lower statistics, perhaps 500 Hz
- Alphas can monitor *distribution* of event times
 - Circular buffer of start/stop times (Histogram on host)
 - 1 CPU cycle (2ns) resolution possible
 - same mechanism for any state duration
 - or counts sampled by event (e.g. # tracks)

Monitoring: Event Samples

- Histograms of objects found (Examine)
 - few (.2 Hz) Unbiased Sample (No L2 cuts)
 - mostly fails
 - few more events before L3 cuts (after L2)
 - passed L2
 - mostly after passed L3
- Verification: run simulator on these samples
 - check data arrives intact
 - bit by bit compare with online L2 results
 - detects hardware, history bugs (nearly only way)

Test Stand at FNAL

- 4 crates:
 - Global simulator (Admin + Worker)
 - 2 preprocessor simulators (A+2W, A+W+Slic)
 - 1 data source (2alphas, MBT's; own MBus)
- Incomplete system--
 - no L1, L2
 - not enough parts for full code of any/all crates
 - except maybe full playback for Global
 - could reconfigure if need be--painful!
- Copy of some real-time inputs? (grounding!?)

Test Stand: What can it do?

- (Pre-)Commissioning/debugging
 - alpha-alpha and alpha-MBT issues
- Timing, verification of download
 - run in real environment; count clock cycles
 - how good is offline simulator?
- Playback
 - drop data into memory
 - testing pre-release after running in simulator
- Debugging
 - event dump and restart (else debug = deadtime)
 - hard to write event dump/reload!?

Zvtx?

- Zvtx in 6cm bins from L0?
 - actual resolution varies with luminosity
 - IF felt to be worthwhile at L2 resolution
 - better to know Z better than Z=0
 - or avoid making mistakes and possible L dependence
 - studies in progress
- L2STT also considering mechanisms
 - candidate vertex Z's, then algorithm to report one
 - better intrinsic accuracy than L0
 - different luminosity dependence than L0

How does L2STT fit in?

- Well defined protocol allows it feed into a preprocessor via MBT
 - like SLIC's do in muon crate
- Send data to L2CTT crate
 - pt-ordered list and impact-ordered list to Global
 - just pt-ordered from L1CFT, lower resolution
 - extra input already reserved
 - add 1 MBT and alpha to L2CTT crate
 - allows simultaneous input of L1CFT and L2STT
 - can build two kinds of lists in parallel
 - modest cost; \$8K (or use spares / cannibalize test stand)
 - can run in parallel until shaken down

L2 STT

impact on L2 performance

- Heavier loading on new CTT inputs
 - 16 B per track? Duplicates?
- Heavier loading of pt ordered list?
 - More Bytes/track? But can reject tracks, too!
- New output: impact parameter ordered list
 - already included in bandwidth estimates
- More work for Global?
 - Yes, but controlled by scripts
 - Must limit # tracks used in matches!
 - STT can only give higher quality

Budget

- Detailed Cost Estimate Exists
- Latest update is down about 80K\$ (SLIC)
 - nearly back to original estimate
- L2STT loaner crates, L2CTT upgrades
 - included in above estimates
- engineering costs not over till it's over
- how many extra alphas as insurance?
 - may be hard to do a 2nd run

How Many Alphas?

- Roughly X 2 design safety factor (10KHz)
 - more alphas are only lifeboat too slow
 - but gain is not linear, maybe square root
 - cannibalize test stand (IF it becomes unimportant!)
- Other uses for more alphas:
 - Shadow nodes (online test at high statistics)
 - where? Real crate or test stand?
 - potential use in STT?
- Production Alpha order in next few months
 - have requested 2 X for some obsolescent parts

Schedule

- Detailed schedule exists
- SLIC now on critical path
- Rule: L2STT can't compromise schedule
- Prototypes due March-May 1999
- Production May-Dec 1999

Issues

- L2STT design decisions (and money...)
- Prototype to production to installation
- Transition to software
 - simulation needed (decisions, studies, development)
 - Low level software (“drivers”)
 - finish download path, L3 output path
- Manpower crunch coming
 - Monitoring, verification, releases just starting
 - infrastructure, definitions needed soon, then people...
 - who writes global algorithms? (MSU, probably?)
 - studies of trigger scripts, global algorithms

Conclusions

- Solid, modular design for L2 trigger
 - connectivity understood; prototypes in progress
 - clear method for L2STT to integrate
- Time budgets understood from simulation
- Hardware supports appropriate algorithms
- Have the manpower for the hardware
- Have excellent core group for the software
- Request TDR approval:
 - go-ahead for production