L2 Components

James T. Linnemann
Michigan State University
DØ Workshop
July 9, 1998

Agenda

- Schedule this summer JL 5
- Prototype/testing issues JL 15
- MBT progress Bard/Baden 20
- MBT/SLIC Inputs Fortner 20
- SLIC status Fortner 20
- FIC status Le Du 15
- VME Space/Initialization JL/Laurens 15
- Monitoring JL 20
- Alpha ordering/ Test Stands JL 10

Prototypes / Testing

- Connectors on VIPA crates
- Bit3 Extenders
- Any other purchases needed?
- VMETRO VME Bus analyzer?
- Any other test equipment required?

VME and MBus Initialization

- VME initialization:
  - based on VIPA geographical addressing
    - card finds own slot number
    - uses to define VME base address
  - applies to MBT, SLIC, worker Alpha's
  - what about Bit3? (jumpers)
- Mbus initialization:
  - Administrator .exe ASSUMES it knows which cards are in which crates
    - modify source code to move cards
  - it thus addresses relevant control registers
  - it downloads to MBT Mbus base address
  - Alphas get correct .exe based on Internet card
  - Alpha Workers usually have distinct .exe's
    - .exe only work if at correct slot
    - defines own VME and Mbus space

VME Address Space

- Long Note from Philippe
- Issues
  - WHAT IS VISIBLE FROM TCC
    - VME space is 32b/crate = 4GB
    - TCC views through all (16) crates thru single 32MB window on Bit3 617 card (28b)
    - splits for A32, A24, A16 spaces separately
    - interrupts also must be mapped
- Principles
  - equal share of direct mapped VME space
    - for each crate
    - for each card
  - try to confine problems to 617 in TCC crate
  - try to make test code = final code
    - 412 (VME VME) not 617 (VME PCI)
    - equivalent ONLY for direct mapped
- How do we proceed

Bit3 cards

- 617 VME to PCI
  - ok when at most 3 crate test stand
  - limitation is PCI slots in PC
  - 32MB window of 4GB in TCC Comm crate
    - but individual Bit3 driver calls can hit anywhere in TCC crate, just slower
- 412 VME to VME
  - need if 4 or more crates
  - + 6u VME crate with 617
  - EACH have 32MB windows
    - up to 8K pieces of 4KB or more each
  - Paged (indirect) addresses
    - 64kB..128..256..512..1MB window

Communication

- TCC Communication Crate
  - single 617 VME to PCI
    - use 16 windows to map to 412's
  - a 412 VME to VME for each L2 crate

TCC Communication Crate

- 16 windows (1 per L2 crate)
  - enough for 7L2 (no STT?) + L1, L2 FW
  - addr 28-31 define window
  - each window 4GB/16 =1/4 GB = 256 MB
  - upper 8MB = reserve for MPM of 412
  - set up with jumpers, NOT SOFTWARE
- DIRECT MAPPING: only A32
  - A16, A24 by paged mode
    - write base address to 412
  - Interrupts also by mapping in 412
- So, Use A32 when possible?

Proposal for L2 crate

- 256MB window direct mapped (a28)
  - propose divide by 16 8MB blocks
    - 4b card + a24
    - = 15 cards + 8MB MPM
    - thus, 8MB (4?) for important stuff
      - base addresses above a25
- then paged A32
  - 4b card + a28, 256MB each
  - is this enough for 64MB alphas?
    - Size of Universe window(s) on alphas
- Need further conventions for
  - A24 (16 MB total)
  - A16 (64 KB total)
  - for interrupts
- is 16 “cards” the right number
  - 32 seems a waste but...

VME Census

- Card Name
- Master or Slave only?
- Uses of A32
- Uses of A24
- Uses of A16
- Interrupts generated
- Interrupts recognized/handled
- Base address by jumper, geographic, or software?

VBD

- 8kB of A16 to control
  - base address by jumper
  - Input (once per reset, not just per run!)
    - lists of pointers to word counts
    - list of start addresses
- reads word counts by A16/D16
  - = 1 universe window/alpha
- data read incrementing from start address by A24/D32
  - = 2nd universe window/alpha

Fast Monitoring and ECL Scalers

- ECL Scalers in Framework:
  - Always there, even if card crashed
  - Reliable for diagnosis, “stale” only if hung
- ECL in Alphas
  - Alpha Administrator (6 X 32 bits/crate)
    - State
    - Buffer Count
  - Alpha Worker (10 or so X 8 bits/worker)
    - State (simpler)
  - Note: LOTS of other info available by VME but only if Admin or TCC forcibly extract
- ECL in FIC (about 6 in system)
  - keep interface simple (no VME)
  - Buffer count X 4 channels
  - Proposal: 8 bits/FIC
    - jumper select: channel 0:3 or worst of 4
    - 0, 1, 2, 3-6,7-10,11-14,15, 16

Fast Monitoring via VME

- Replicate function of ECL scalers on VME board:
- Define scaler gates, sample these at some reasonable frequency
  - 132 ns not necessary
  - read out at .2 Hz or so
    - 1000 samples =>200 Hz; try 1 KHz or so
    - pick convenient down-scaled from clock
- Appear in monitoring by Administrator copying to TCC DPM
- Applies to:
  - MBT
  - SLIC

VME Scalers: MBT?

- 8 scalers per card
  - same definition as FIC
- OR 17 scalers per card
  - 0, 1, 2, …. 17
- Select Channel: 0…7
- Define buffer count as number of events with SOME information
- OR worst of 8 (including L1 SCL)
  - always arrives first, pushes up apparent occupancy
- OR worst of 7 (excluding L1 SCL)
- Question: How to enforce time-consistent copy-before-read of scalers

VME Scalers: SLIC

- Same definition as for MBT (0:15) for buffer occupancy
- Add State information for
  - Master DSP
  - DSP 0:3
  - How many bits?
- Master DSP copies counters to readout area on receipt of
  Collect Status
- Similar questions on getting consistent counts.