Alternative Ideas for the CALICE Back-End System

Matthew Warren and Gordon Crone University College London 5 February 2002

Concept

Based on PCI bus having higher bandwidth than VME.

- Generic PCI uses a 33Mhz/32 bit bus practical 95MByte/s max.
- VME64 25Mhz/64 bit practical 55MByte/s max.

This is an alternative to the normal crate based design: A PC based system with the BECs on the PCI bus inside the PC.

We made the following assumptions:

- Data rates and buffer sizes as-is from current spec.
- 4 PCI slots per PC utilised.
- Each PCI BEC has 4 x 1GigaByte optical receivers:

4 cards x 4 receivers = 16 channels/PC = we need 6 PC's

- The transmit component is located on a separate card, on a separate bus segment (could even be ISA!).
- PCI sustained data-rate = 85MByte/s
- Think only about full read-out, not small memory.



PCI BEC Card Design and Operation

Each receiver has 4MBytes of buffer: 16MBytes/card. Each card has 1 FPGA for control and PCI interfacing.

After a bunch-train:

- the FE takes \sim 20ms to process and \sim 30ms to send to all BECs.
- Each PC reads-out 4 cards per train: 64MB at 85MBytes/s
- But, hard-disc writes are slower: 40MBytes/s Overall train BE processing: ~1.65s

NOTE: If we ran that fast we could generate 136 GB/hour/PC!



PCI BEC Pros

Faster read-out (2s cycle vs. 10s for VME)Processing available to work new data locally (if not busy)Forces a partitioned DAQ framework - useful integration with HCAL

Cons

Event fragments need to be moved to a central point:

- Requires additional bandwidth and resources (Gbit Ethernet/PC)
- Could become a serious bottleneck on long runs etc.
- Extra cost of high speed network infrastructure

The control of the system is more complex and may be too unwieldy for a test-rig.

Physical size of the SRAM too large for a PCI card (32x2cm² chips).

Could cost MORE than the VME solution:

- Cost/link remains the same, but 6 extra high spec $PC = \pounds 12-15k$
 - + More boards (24 vs. 6) + Separate TX boards (6).

Back-End Mk II

The overall system is much slower that the individual FE to BE links. A sequential system may work. Applies to VME too (but slower).

Ideas for a new BEC design:

Remove the buffering from the BEC altogether:

- The FEC can be considered the buffer
- Use the TX to start transmission from the FECs in sequence
- Multiplex the receivers on each BEC so they can talk directly to the PCI system (via short FIFO)

Data arrives at 96MByte/s and leaves at 85MByte/s we need to:

- Compress the data between RX and PCI (not much info/IP for this)
- Use a large FIFO, or 4M SRAM.
- Move to 64 bit PCI (not 66Mhz -- too new and not CompactPCI)

Disc writes are still the main bottleneck, so:

- Use the PC processor to compress the data before it writes to disc
- Try RAID, Ultra160 SCSI etc (expensive, real throughput unknown)

BE II Costing

BEC: - larger FPGA: £200 - RX: Gbit: $4 \ge \text{\pounds}100 = \text{\pounds}400$ - Memory/FIFO: £100 - PCB: £500 Total: $\pounds 1300 \ge (24 + 4 \text{ spare}) = \pounds 36400$ **BEC-TX:** - 1 x FPGA: £200 - TX: 1 x Gbit: £100 - PCB: £500 Total: $\pounds 800 \ge (6+2) = \pounds 6400$ PC: - High spec (dual processor) PC: $7 \ge \pm 3000 = \pm 21000$ - Gigabit NIC: $(7 + 1) \ge \text{\pounds}300 = \text{\pounds}2400$ - 8 port Gigabit switch: £3000 Total: £26100 **Total for BE: £68900**. [VME = Crate+PC+BECs = £82000]

CompactPCI

Another diversion: Using CompactPCI instead of VME:

- Mostly available in 3U and 6U.
- 8 slots (7 + processor), more requires bridging.
- 33MHz bus, 32/64 bit = 85/170MByte/s
- robust VME like backplane connectors (can hot-swap)
- Ability to debug hardware in a normal PC using adapter

If we managed to squeeze 16 receivers (+ 1 TX) onto a 6U board:

Reading out sequentially we need to move 225MBytes (from spec.)
PCI bus can do it (with larger BEC buffer) in 1.35s,
Harddisc (with 2:1 compression on writes): 2.85s

<u>Ethernet</u>

Diversion 3:

Generate IP type packets at front-end and simply plug this into a high-bandwidth switch (expensive)

This could work well if only a few links operational at one time (which may be the case: the current spec hints at 0.3% utilisation due to the slow BE).