MANCHESTER 1824

The University of Manchester

CALICE Calorimeter for ILC

Future DAQ Directions

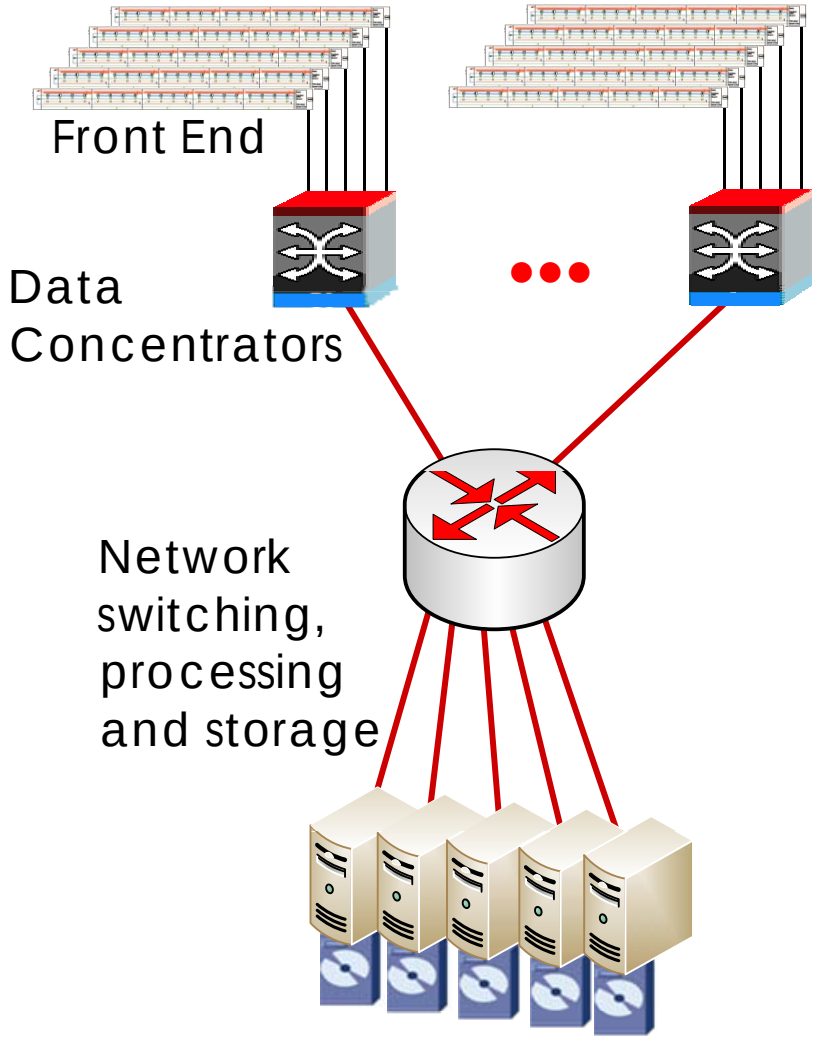
David Bailey for the CALICE DAQ Group

Objectives

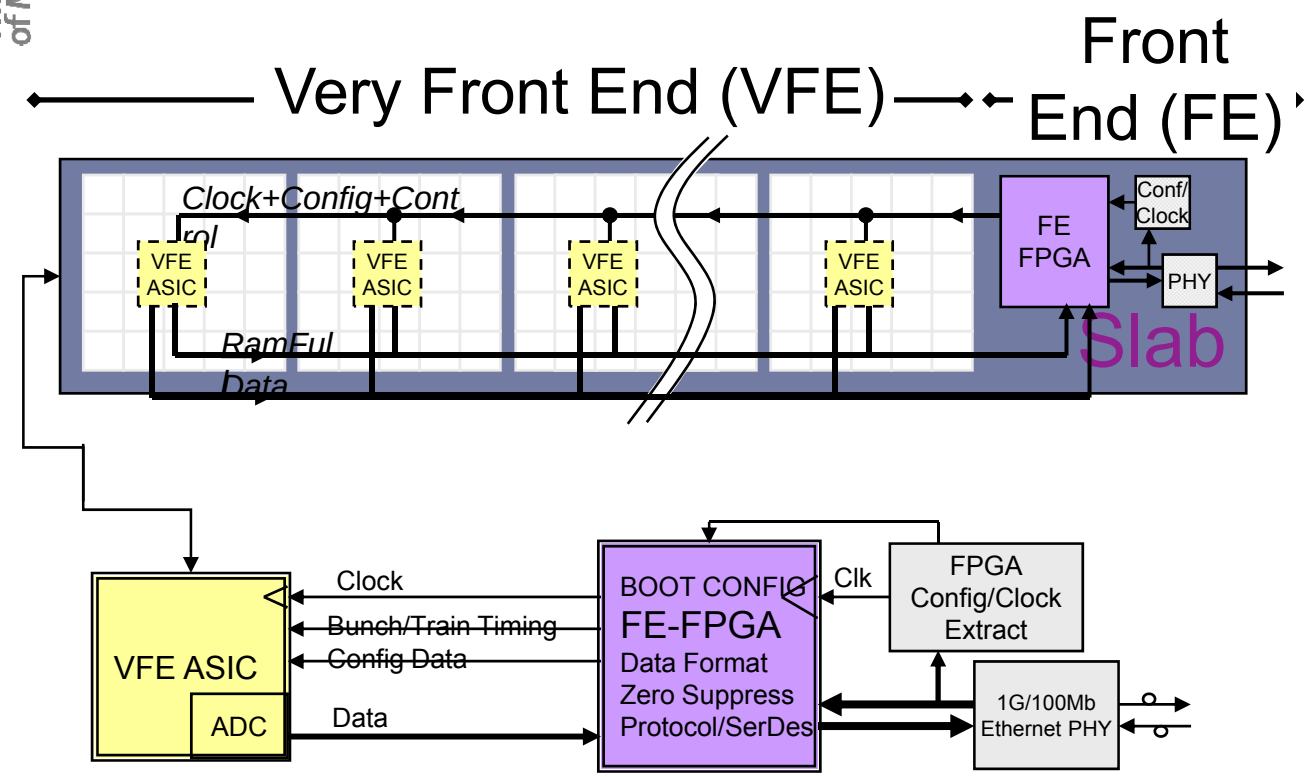
- Not to be prescriptive at present...
  - Utilise as much "off the shelf" technology as possible
    - Minimise cost, leverage industrial knowledge
    - Standard networking chipsets and protocols, FPGAs etc.
    - Not tied in to a particular design based on specific hardware
  - Scalable
    - From single, low-bandwidth systems to high-rate environments
  - As "generic as possible"
    - Clearly there will need to be sub-detector specific hardware at the (very) front-end but try to be homogenous downstream
- ... but attempting to act as a catalyst to use commodity hardware instead of following the usual bespoke route.

Overview

- Classic Design
  - Front-ends read out into on-detector data concentrators
  - Data concentrators drive long links off detector
  - Off detector assembly of complete bunch train data and event storage
- Points to note
  - Triggerless operation
  - Use inter-bunch-train gaps to send data off detector
  - Bunch train data processed/assembled near online asynchronously from readout



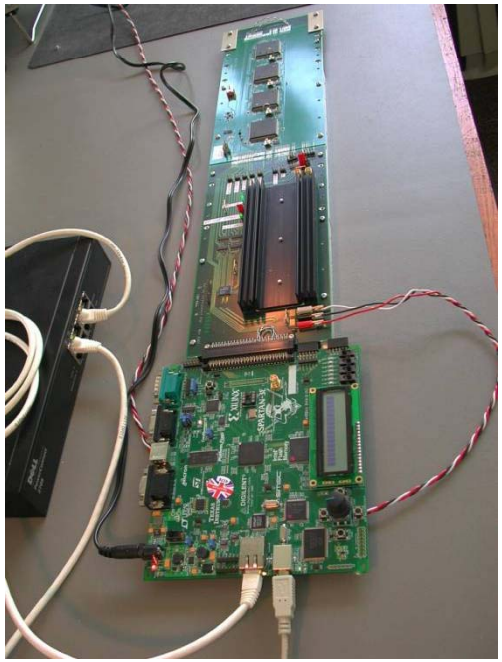
Current Architecture



Schematic layout of VFE for ECAL. Front End FPGA on DIF board provides control and data paths for detector ASICS

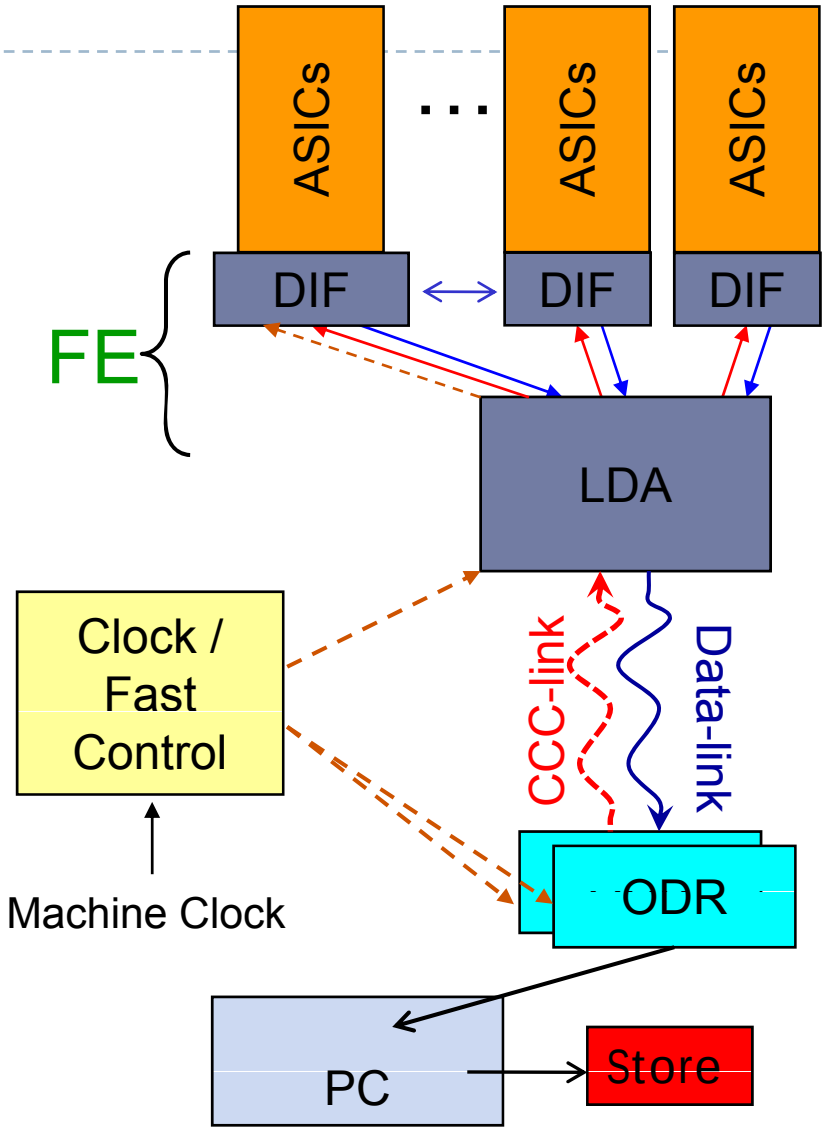
Have to understand data transport on long (1.7m) PCBs

Simulated ECAL slab with prototype DIF



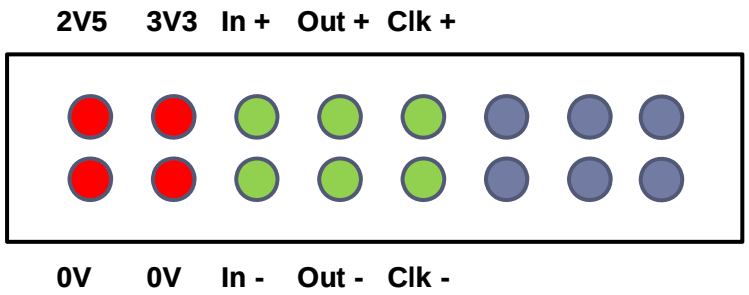
Current Architecture

- DIF
  - Sub-detector specific component (Detector InterFace)
  - Supplied with low jitter clock from first-stage concentrator (LDA)
    - 50MHz with ½ns jitter
    - All detector-specific clocks are derived from input master clock on the DIF
  - Bi-directional serial links to LDA
    - Would like these to be "generic" – driven by highest bandwidth requirement
    - Require fixed latency links if clock and control encoded across them
  - Clock feed through and redundant data links to neighbouring DIF for readout and clock redundancy
  - Standard firmware to talk to DAQ



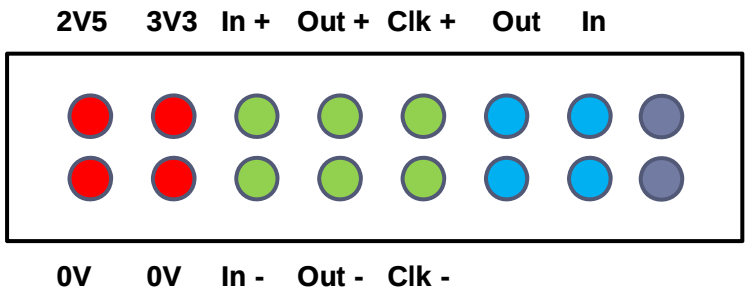
Current Architecture

- DIF→LDA Interface
  - Keep it simple
  - Standard 10/16 pin IDC format connector
    - Power from DIF for potential link serdes (3V3 and 2V5 at 250mA)
    - Input from/to LDA
      - Serial In
      - Serial Out
      - Clock In (may be recovered from link)
      - All LVDS 2V5
      - 8B10B encoding (or Manchester)



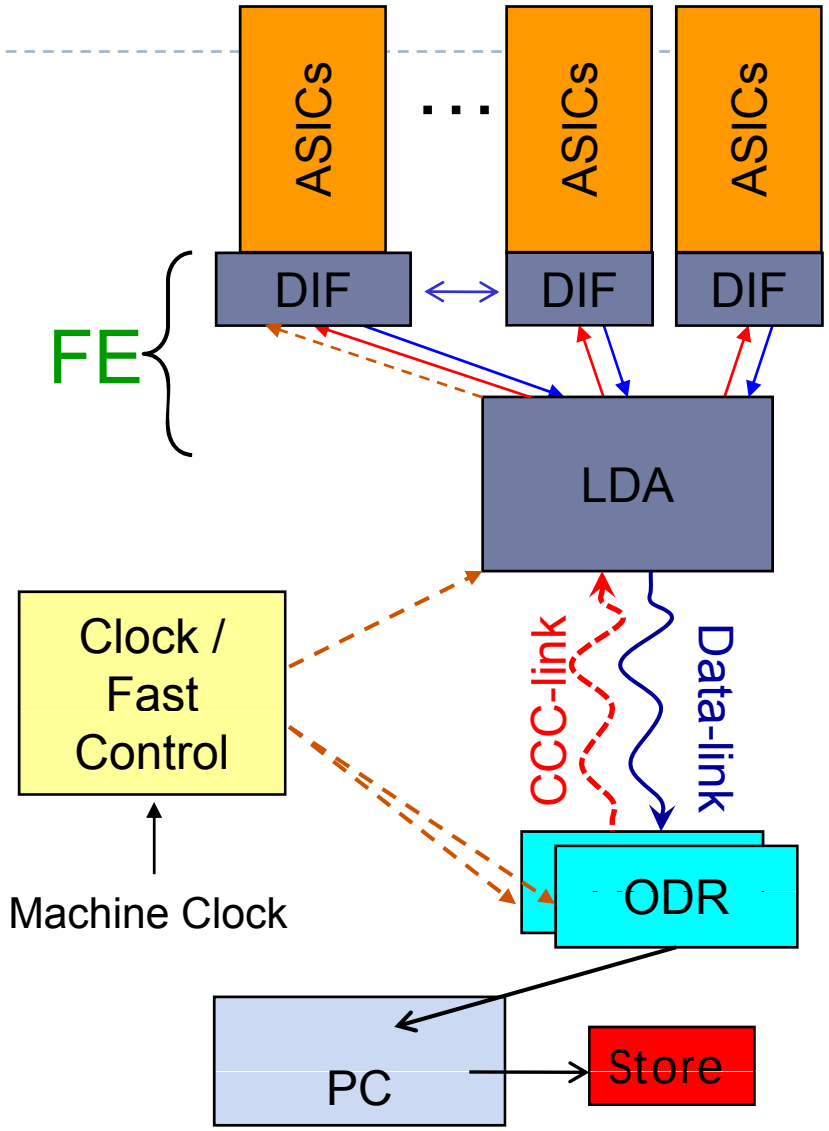
Current Architecture

- DIF→DIF
  - Same format as LDA interface
  - Used for redundant communications and clock between DIFs in case primary link fails
  - Or allows DIF to act as "Interface" to downstream DAQ for many channels from VFE
  - Require 2 extra single ended lines to specify link and clock direction
    - Master/Slave signal to define clock master
    - CMOS 2V5 suggested
    - Maybe use a single 3-state line later



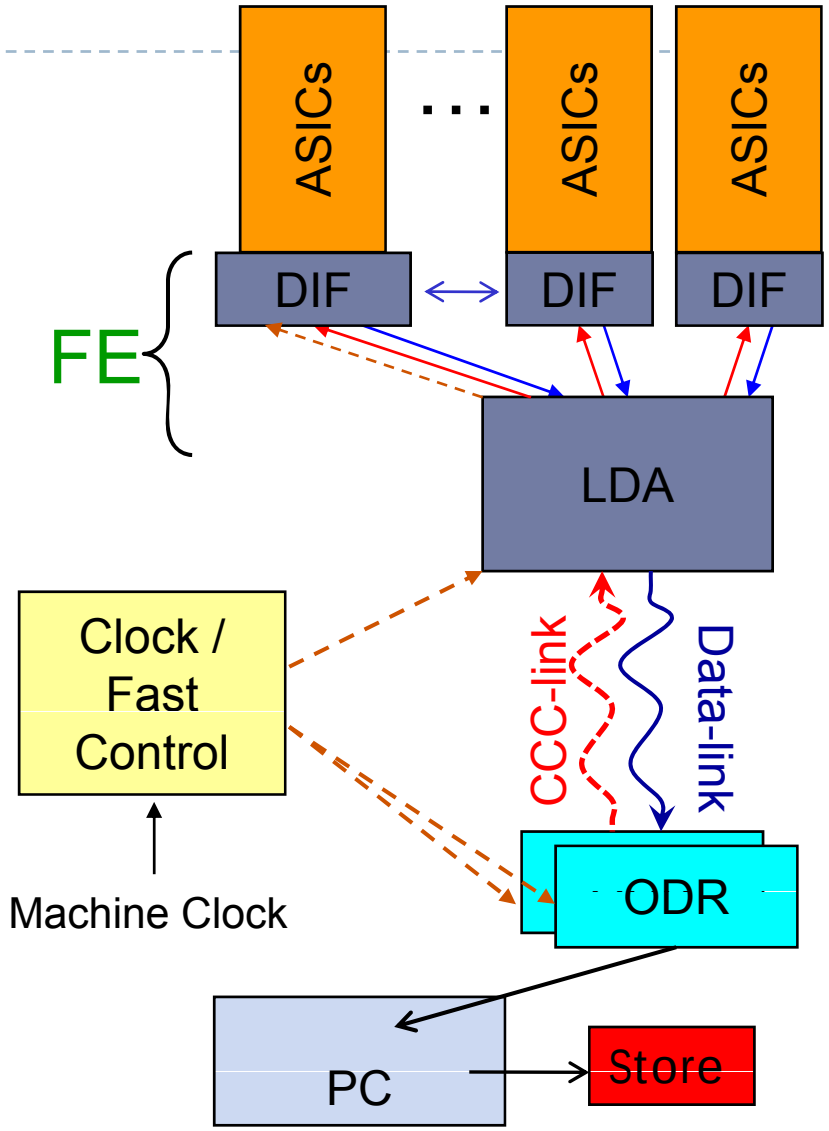
Current Architecture

- LDA
  - On (or very near)-detector data concentrator
    - Clock/control fan-out
    - Data receive and buffering from DIFs
    - Framing/error correction for transmission off detector
    - Possible direct connection to machine timing
    - Fixed latency links to DIF
    - Downstream links *need not be* fixed latency
      - Obviously makes sense to try to have upstream and downstream LDA-Off Detector from the same technology
      - May be able to use commercial Ethernet chipsets



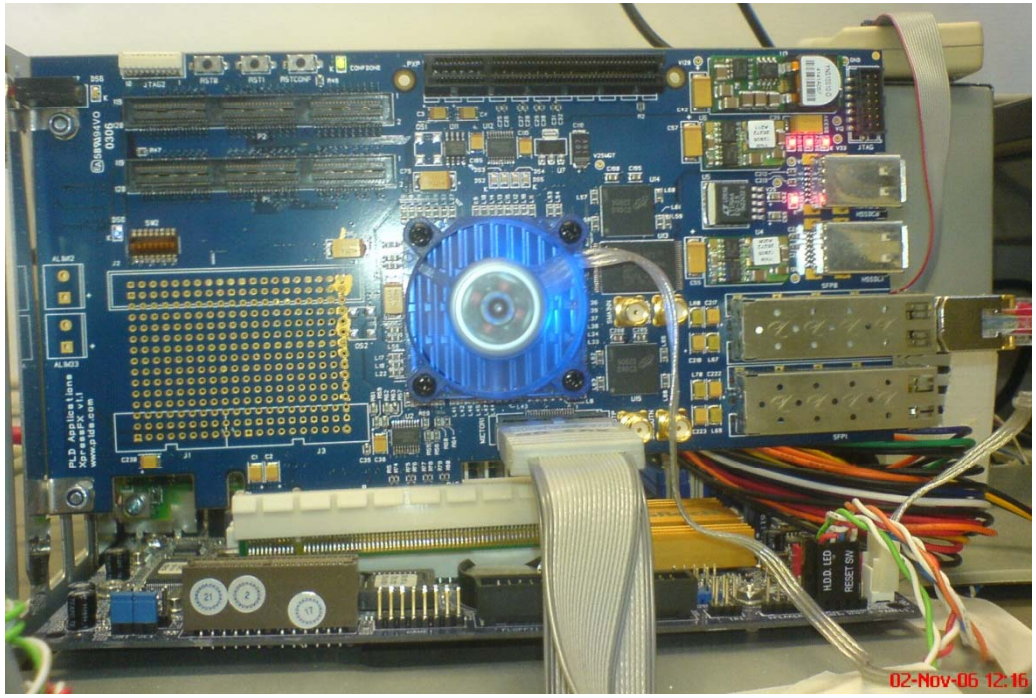
Current Architecture

- ODR
  - Off-Detector Receiver
    - Gets data into a usable form for processing
  - Three logical tasks
    - Receive
    - Process
    - Store
  - Current implementation using Virtex 4 FPGA development boards connected over PCI-express in a PC
  - Input is Ethernet for testing



Prototype ODR

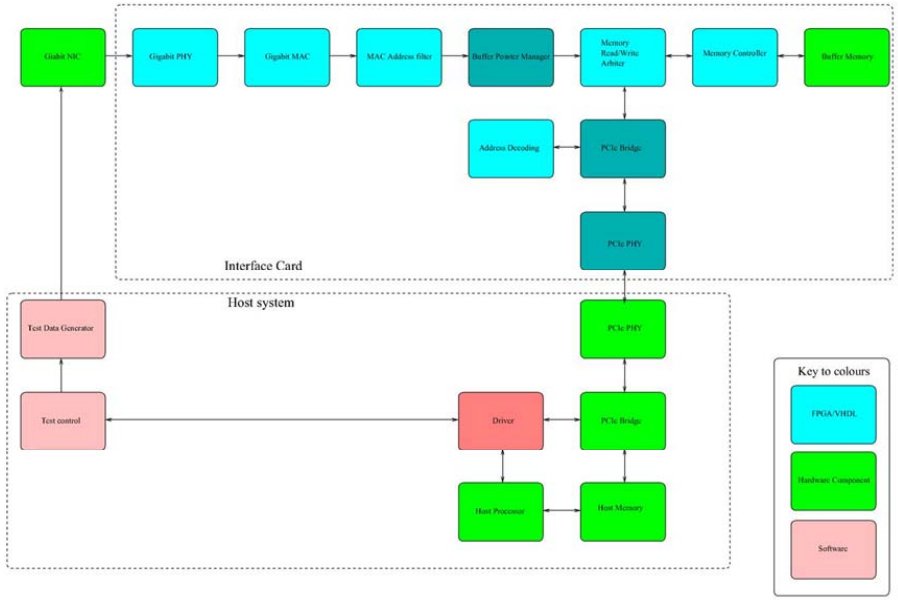
- Development board from PLDApplications
  - 8 lane PCI-e card
  - Xilinx Virtex4FX60 FPGA
  - DDR2 memory
  - 2 SFP cages – 1GigE
  - 2 HSSDC connectors
  - Can drive/receive 1Gbit Ethernet direct to FPGA on the board
  - 10Gbit possible with addition of small daughter card
- Will also be able to test some LDA functionality using on-board LVDS outputs

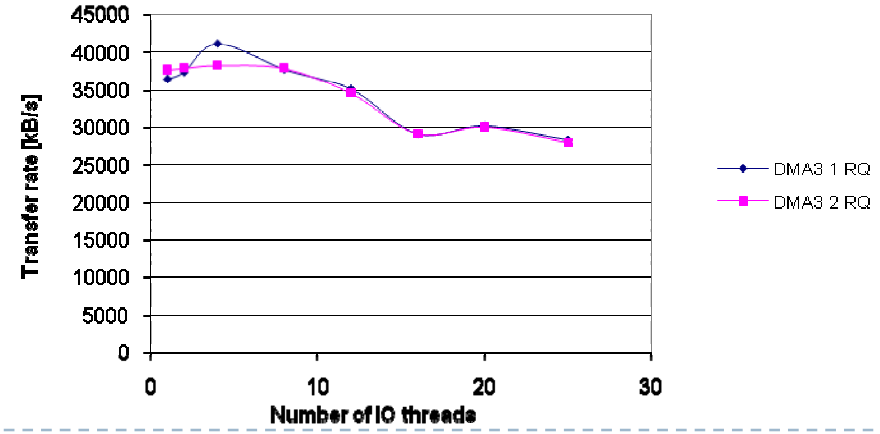


Demonstrated 1Gbit Ethernet operation from this board - See proceedings of IEEE NPSS Real Time Conference 2007 for details

Prototype ODR

- Prototype is working
  - Firmware done
  - Host PC driver working
  - Rate tests underway
    - Achieving reasonable throughput to disk on this first iteration
- Bottom line: The ODR prototype works!





Next steps

- Make an LDA
  - Will certainly use something that exists already
    - Spartan 3 development board or something similar
- Investigate LDA→DIF links
  - How much bandwidth do we really need?
  - Should we try something "off the shelf" like GLINK?
- Have to define ODR→LDA link
  - Does it need to be synchronous for clock & control?
    - i.e. Where does C&C enter the system? LDA or further downstream?
  - Can we use commercial Ethernet chipsets?
- Have to have an answer in about a year...