



HCAL TPG Generation Plans CMS Upgrade Workshop

October 28, 2009
Jeremiah Mans



- System overview and role of the SuperHTR

- Hardware demonstrators : status and current projects
 - MiniCTR1
 - MiniCTR2

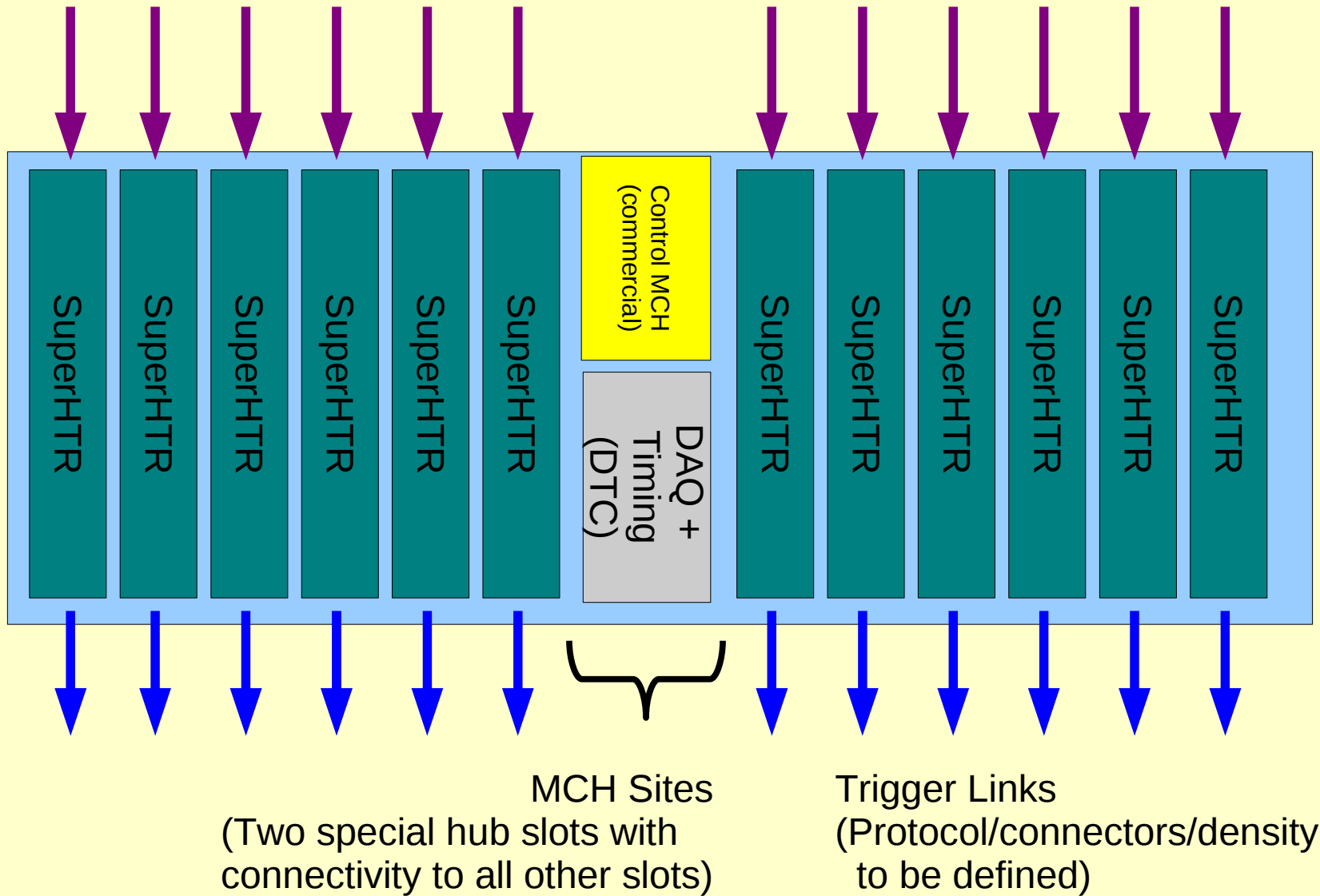
- Plans for TB2010

- Path towards production electronics

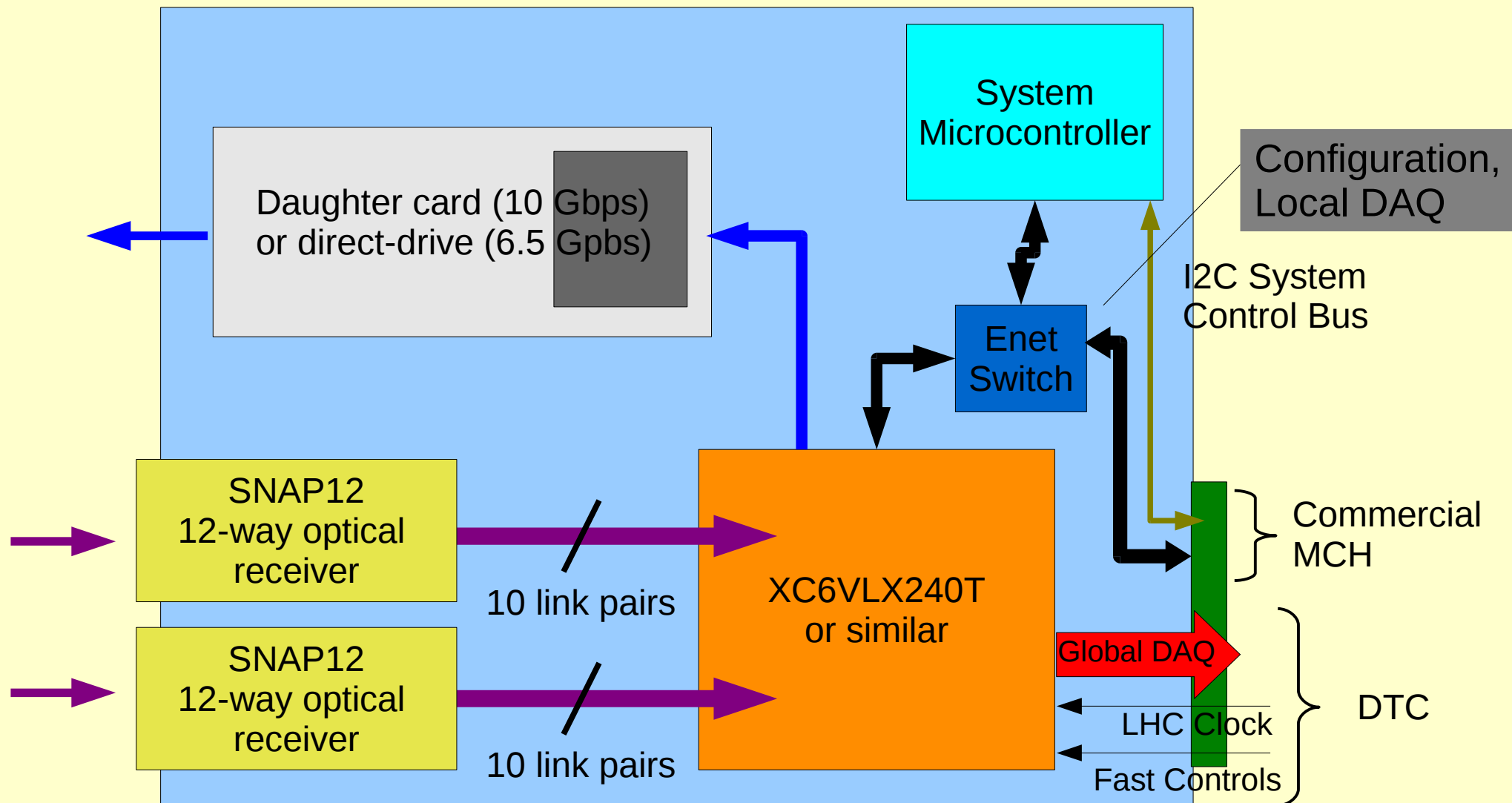
Backend Crate in upgraded HCAL



18-20 Links/
SuperHTR
from FE



Dataflow in/out of SuperHTR



HCAL => Trigger Link



- Several options for the HCAL Trigger Link, which are not *necessarily* exclusive
- Direct-drive optical links from the FPGAs
 - Advantages: match with “optical SLB” upgrade path, fully contained implementation in FW of FPGAs
 - Disadvantage: link b/w limited to ~6.5 Gbps on a practical level (10 Gbps-capable FPGAs are announced, but too expensive for use in HCAL, which can best take advantage of lower speed links from the detector)
- Daughter card concept
 - Advantage: flexibility to match optical SLB now (probably!), upgrade to 10+ Gbps later
 - Disadvantages: engineering complexity



■ uTCA Infrastructure

- Fairly complex microcontroller firmware necessary to interface with uTCA crate controller (MCH)
- Clock distribution and fast controls distribution [see BU talk, HCAL Session]
- Reliable slow controls/local DAQ support [see UVA talk, HCAL Session]

■ Front-end link

- Stability, performance of high rate (>3 Gbps) optical links
- Latency control – link operations mode (LHC-synchronous, asynchronous with IDLE removal)



■ Global DAQ

- Link between SuperHTR and DTC
 - Backpressure/quasi-OOB signalling
- Selective readout support (?)
 - Behavior/requirement is dependent on Global DAQ constraints and requires simulation work

■ Trigger

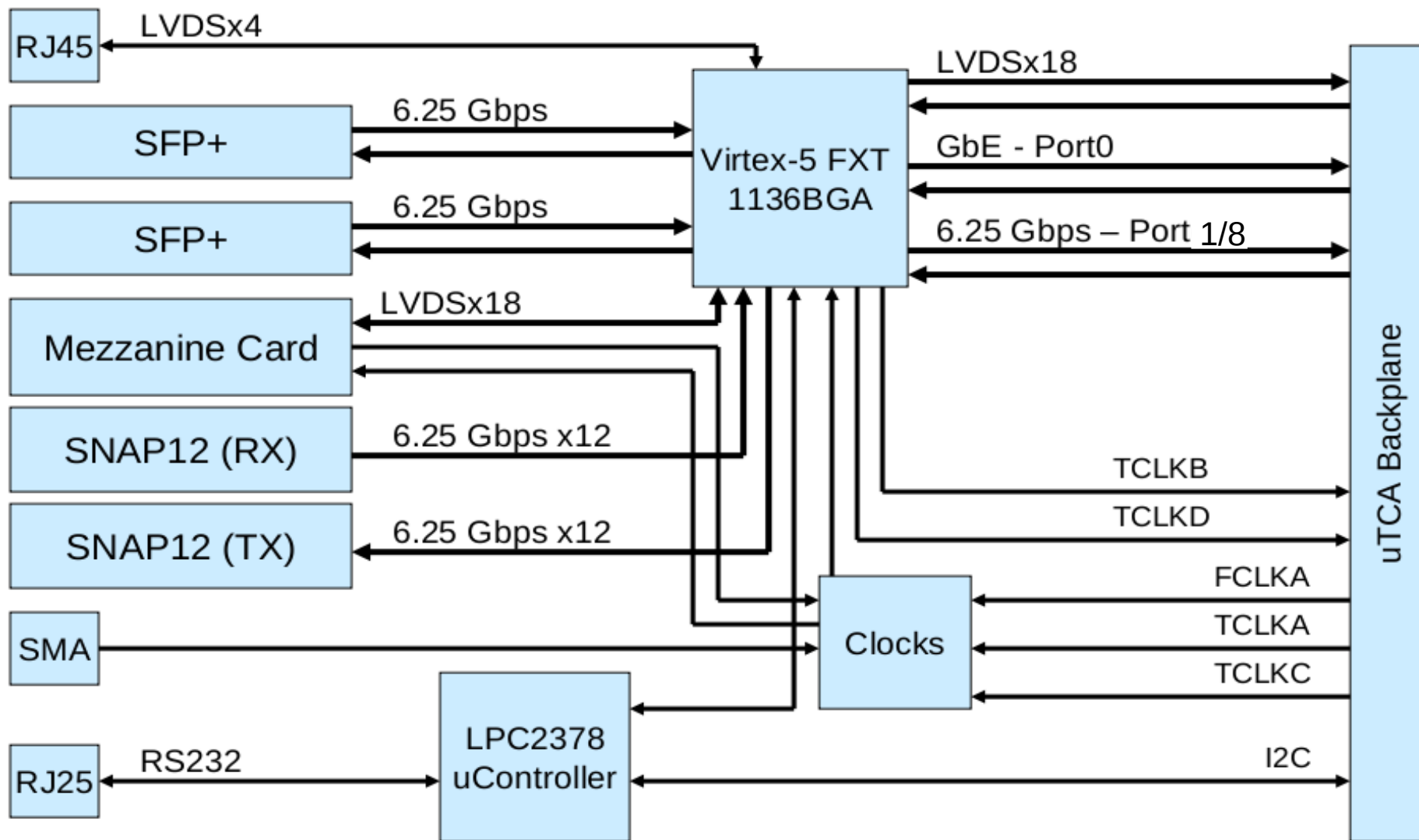
- Data link selection and operation: latency, bandwidth
- Data format contents: issue for simulation, but expect to need a base energy per tower and several additional bits for longitudinal shape information

Hardware Demonstrator I : miniCTR



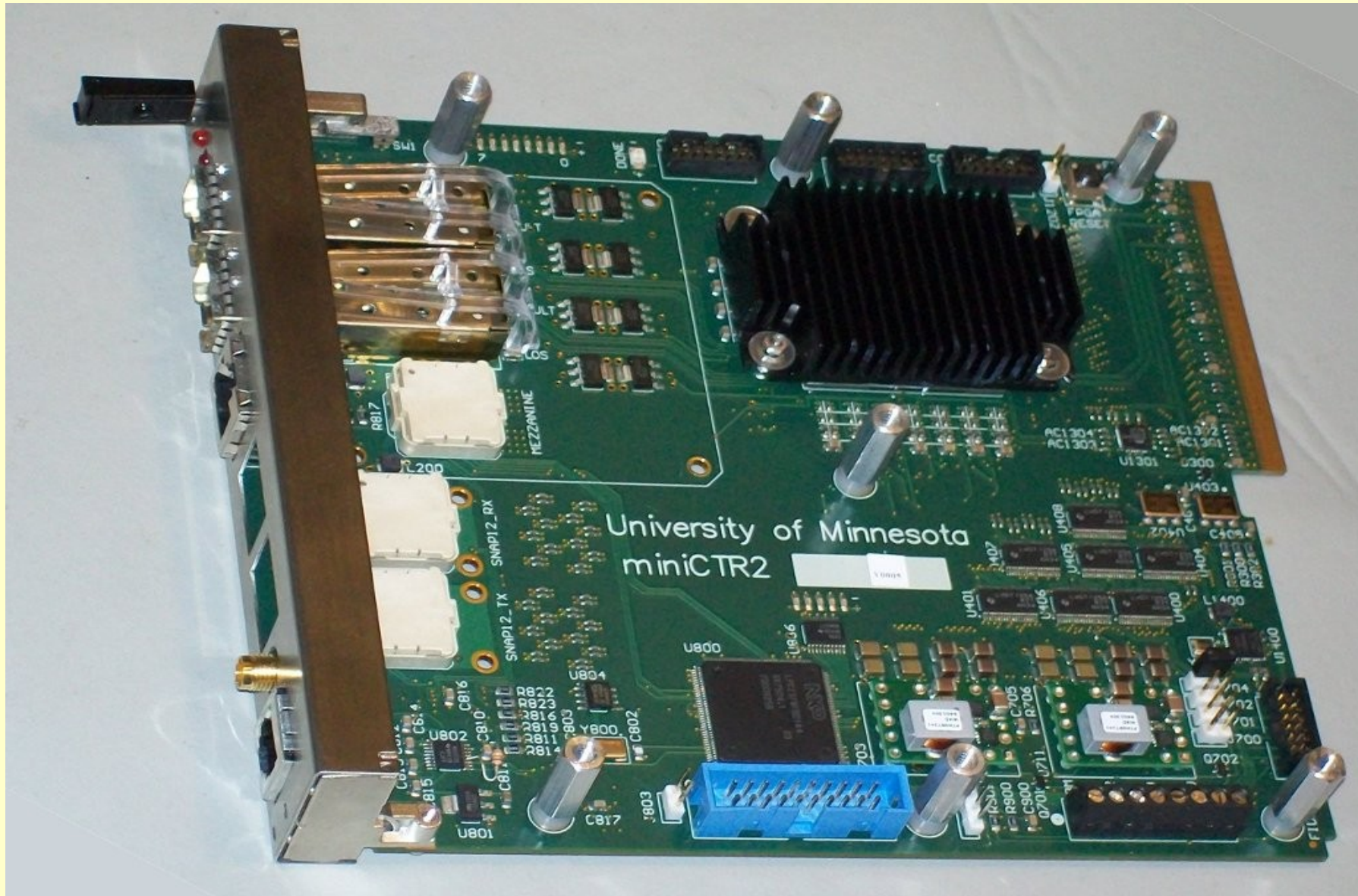
- Completed in Summer 2008
 - Demonstrate workability of uTCA form factor
-
- Utilize new/advanced Virtex 5 FPGAs
 - Develop experience with “3 Gbps-class” built-in deserializers
 - Various tests completed and underway including link stability and latency measurements (with crystals), HCAL link reception (ongoing effort)

Closer to the Final Product: MiniCTR2



Block Diagram

Closer to the Final Product: MiniCTR2



Detailed HW Status



- MiniCTR1 Cards: 4 fully functional cards (2 at CERN, 2 at Minnesota)
- MiniCTR2
 - Bare PCBs : 20
 - Assembled : 5
- Crates
 - Obtained a number of 2-4 slot uTCA crates for a low price
 - Will need a proper 2-MCH crate for TB2010 (probably to be purchased via BU – development need primarily for DTC)
- Controllers (MCH)
 - Purchased 3 NAT MCHs (~\$1.3k/ea)

MiniCTR2s in Development Crate





■ Underway

- Basic functionality tests
 - FPGA configuration, I/O
 - Clock control (48 clock mux bits)
- Microcontroller firmware for interfacing to the MCH (i2c physical layer, with IMPI logical layer on top)

■ Planned

- 2.5 and 5.0 Gbps optical links based on local oscillators (8b/10b)
- Implementation/testing of CERN GBT protocol reception
- Local DAQ capability via Ethernet (8b/10b)
- TB2010: self-trigger capability via LVDS output
- Reception of clocks, fast controls from DTC

From miniCTR2 to SuperHTR



- Research and validation must prove successful for optical links, clocking, DAQ, etc
- Double the incoming link count
 - Replace the outgoing SNAP12 with an incoming
 - Requirement for twice the incoming links: two Virtex 5 FPGAs or a high-scale Virtex 6 FPGA
- Settle trigger I/O questions (direct-drive, daughter card, etc)
- Other projects have *fairly* similar requirements: generate a common design?

Joint work between HCAL and Trigger



- “Early-and-often” joint work between HCAL and trigger will be necessary for a smooth integration of the system
 - Remember: goal is to be able to upgrade HCAL piecemeal (HB and then HE in two subsequent shutdowns) if necessary
- Proposal: Plan for joint testbeam work, possibly in 2010 and for certain in 2011
 - Self-triggering of a detector always highlights all unfortunate behaviors