Development in electronics and DAQ integration in CMS for Run 3 and beyond

Jeroen Hegeman

For the CMS electronics coordination and DAQ groups

April 13, 2016

Ideas for development in electronics and DAQ integration towards Run 4

Jeroen Hegeman

For the CMS electronics coordination and DAQ groups

April 13, 2016

Introduction

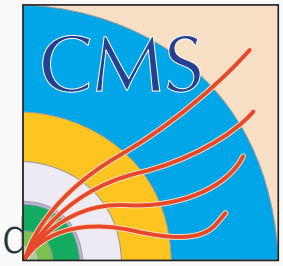
Introduction

- Will focus on the ‘detector side’ of DAQ
- Run 3 is far away
- Run 4 even farther
- This presentation holds no concrete plans, nor recommendations
- But it is good to cultivate ideas/opinions spawning from the current upgrades

Introduction

LS1

Paradigm changes in various places



- Decoupled trigger control from physics algorithms (GT/TCDS)
- DAQ1 -> DAQ2: decoupled HLT from online framework, file-based transfer to the HLT, introduced ‘online cloud’
- First steps from VME to microTCA

YETS16/17, LS2, ...

- New pixel detector -> increase data volume.
- Integrate muon end-cap GEMs.

LS3

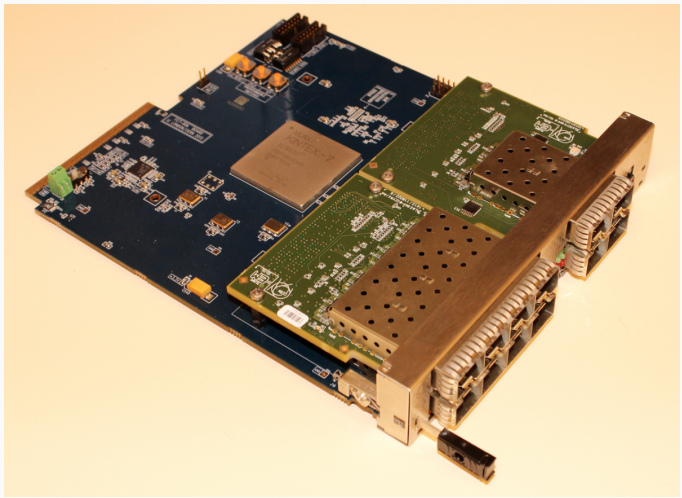
Main Phase 2 upgrades

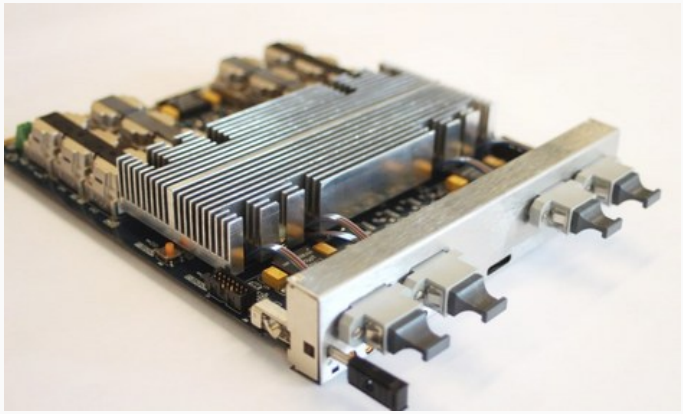
- Detector, DAQ, and electronics overhaul
- Probably: ATCA, GBT. Beyond that?

Electronics standards/form factors

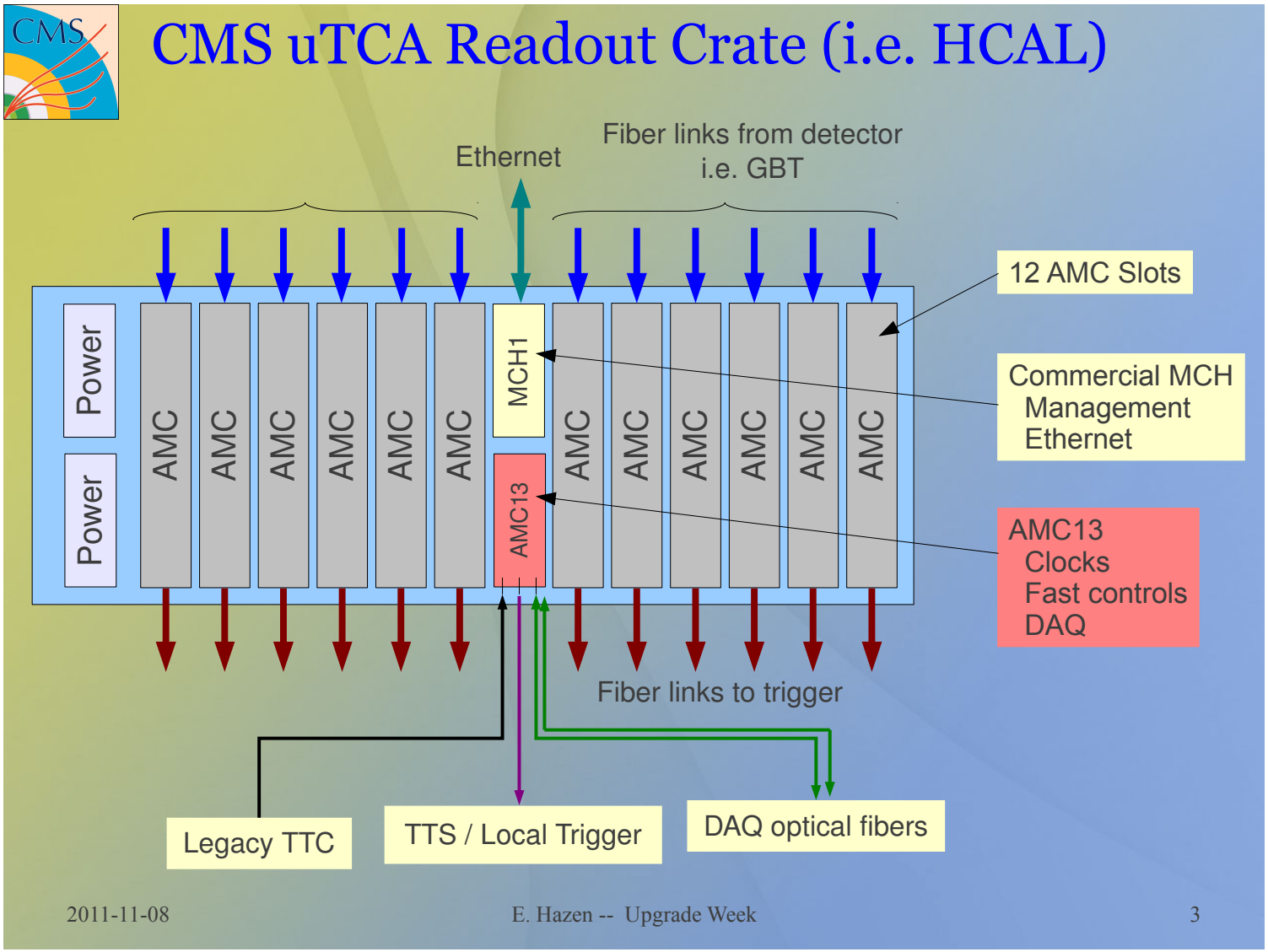
Electronics standards/form factors

- Original CMS systems: mixture of VME, compact-PCI and custom systems
- Phase 1 CMS upgrades based on microTCA
- Future upgrades have tentatively settled on ATCA





CMS microTCA use



Credit: Eric Hazen (BU)

CMS microTCA use

Backplane

- Standard redundant telecom backplane with ports 2 and 3 routed to respective MCHs
- DAQ on AMC port 1
- LHC clock distributed on FCLKA
- Timing and fast control (TCDS) distributed on port 3
- *Note:* No PCI-express fabric clock

Secondary MCH slot used for AMC13

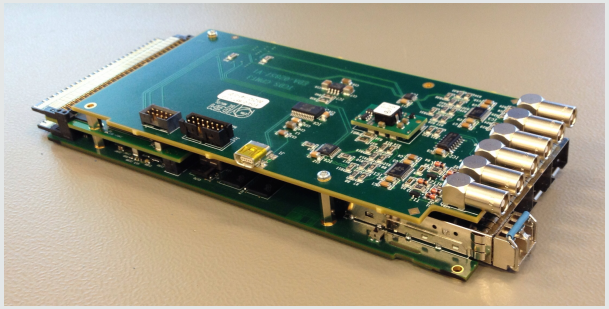
- TCDS interface to microTCA crate
- DAQ connectivity (up to three times 10 Gbps)

MicroTCA in CMS Phase 1

	Board	Main card developer	FPGA	Link connectivity
Everybody (TCDS/DAQ interface)	AMC13	Boston University	Kintex-7 + Virtex-6	3 RX/TX @ 10 Gb
HCAL	uHTR	University of Minnesota	Virtex-6	24 RX @ 6.4Gb 12 TX @ 6.4Gb 2 TX/RX @ 4.8Gb
TCDS, Pixel	FC7	Imperial College London, CERN EP-ESE	Kintex-7	8 TX/RX @ 10Gb (first FMC site) 12 TX/RX @ 10Gb (second FMC site)
Trigger, GEMs	MP7	Imperial College London, CERN EP-ESE	Virtex-7	72 RX @ 13Gb 72 TX @ 13Gb
Calo trigger	CTP7	University of Wisconsin	Virtex-7 + Zynq	67 RX @ 10Gb 48 TX @ 10Gb
Muon track-finders	MTP7	University of Florida	Virtex-7 + Kintex-7	80+4 RX @ 10Gb 28 TX @ 10Gb
DT sector collector	Twin Mux	INFN Padova	Virtex-7	64 RX @ 480Mb 12 RX @ 10Gb 12 TX @ 10Gb

- Clear reduction in electronics diversity, especially in the trigger upgrade.
- All microTCA infrastructure: COTS.

MicroTCA



+ Redundant powering
+ Redundant control hub (MCH)
  - Not used in the CMS scheme
+ Network-connected (ethernet), no point-to-point single-points-of-failure
- Commercial success and future of microTCA is not clear

All-in-all

+ The AMC/mezzanine approach is quite flexible
+ In CMS, microTCA is part of the learning curve towards ATCA

ATCA

ATCA

+ Easier to satisfy power requirements of large FPGAs and many links
+ Networked and self-managed, like microTCA
• Needs work to properly integrate into existing rack/power/cooling infrastructure
- Standard is intrinsically bad for timing/trigger systems
- Modularity-wise looses from good old CAMAC/VME

Lots left to learn

CMS is slowly settling on ATCA as the chassis choice for Phase 2. We'll have to learn en-route, as we did with microTCA.

Tentative CMS plan

- Common shelf specification
- Common IPMC. E.g., design supported by CERN PH-ESE, including
- Follow the CMS microTCA approach: CMS integration switch blade (AMC13++)
  - TTC++
  - TTS++
  - DAQ interface

  hardware module.
- For the moment aiming for 100 Gbps available on the backplane

Looming open questions

- DAQ via backplane or via leaf-card mezzanine?
- Memory-mapped PCIe, or ethernet-based networked access?

Everything part of the online cluster?

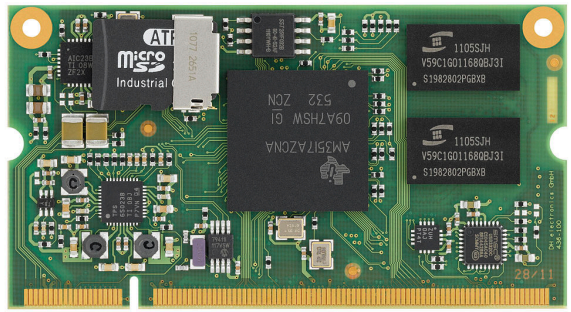
Everything part of the online cluster?

Point of study

Equip all new xTCA boards with embedded Linux endpoint?

System-on-a-module mezzanine

- Mezzanine-based implementation to allow upgrades
  - Could be sourced commercially
  - Atom vs. ARM?
  - Form-factor, pin-out?
- Define minimal CPU <-> board/FPGA communication set
  - PCIe (how many lanes?)
  - Ethernet (IPbus)
  - FPGA config protocol (SPI?)
  - USB (debug terminal)
  - JTAG



Source: Wikimedia Commons [1]

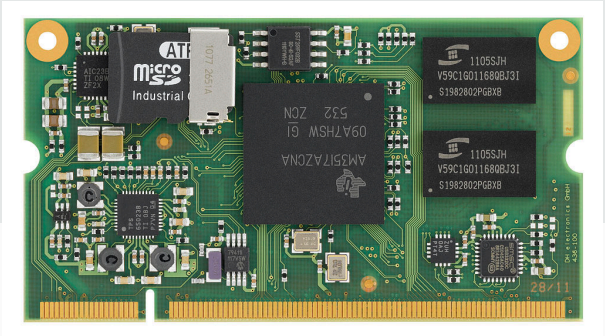
Everything part of the online cluster?

Point of study

Equip all new xTCA boards with embedded Linux endpoint?

OS support

- Would require a dedicated ‘CMS embedded OS’ distribution
- Could make electronics boards less of an exception for the sysadmins than they are now
  - DHCP instead of RARP
  - True TCP/IP instead of IPbus?
  - Network booting?



Source: Wikimedia Commons [1]

Everything part of the online cluster?

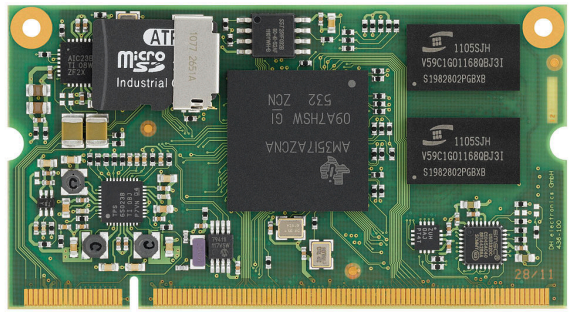
Point of study

Equip all new xTCA boards with embedded Linux endpoint?

Interesting, but no silver bullet

- Have to keep an eye on long-term support/availability
- *Have to make sure we don't retrace our steps on the single-board computer path from before.*

Not intended to compete
with the DAQ path, of course!



Source: Wikimedia Commons [1]

Between hardware and software

Firmware: more and more like software

A bit of a culture change

- Firmware projects becoming larger, with more specialized expert corners (e.g., high-speed links, clocking).
- Multi-developer projects require ‘software-like’ collaboration, code sharing, change tracking, etc.

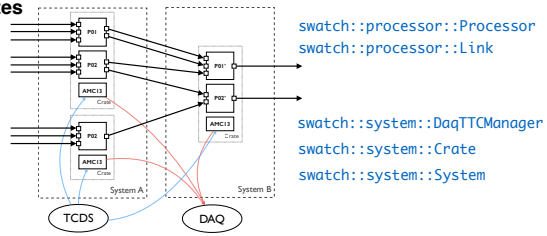
Firmware: more and more like software

Example: L1 trigger upgrade

- Each processor firmware built from configurable list of tags and dependencies (board, algos, etc.).
- Software framework transparently maps common firmware functionality (!) across boards to common API.
- Code integrates with legacy trigger software, but does not depend on it.
- For testing, real software can run against simulated hardware.

SWATCH: Systems & processors (2)

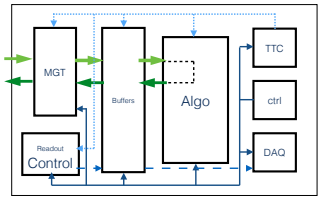
- Upgrade subsystems:
  - **Processor** — uTCA card processing data, transmitted over **optical links**
  - **DaqTTCManager** — Connection to TCDS & DAQ networks (i.e. **AMC13**)
  - Stored in **crates**



- **System** — One or more processors & AMC13s
- Each trigger subsystem = 1 SWATCH system
  - calol1, calol2, ugt, bmtf, omtf, emtf, ugmt

SWATCH: Systems & processors (3)

- Each processor contains following components, representing common basic layout of all processors:
  - TTC block
    - `swatch::processor::TTCInterface`
  - Readout block
    - `swatch::processor::ReadoutInterface`
  - Input optical port
    - `swatch::processor::InputPort`
  - Output optical port
    - `swatch::processor::OutputPort`
  - Algorithm block
    - `swatch::processor::AlgoInterface`



- Also:
  - `swatch::processor::LinkInterface` - aggregates input/output ports

Credit: SWATCH developers

Summary

Summary

- LS2 is a bit of an intermezzo where CMS Trigger/DAQ upgrades are concerned
- But LS3 will see several important changes
- Anticipating important changes in system design, but also in development practices
- Not yet time for hard decisions, but it *is* time to solidify some of the lessons from the Phase 1 upgrade for the future

Backup slides

References

[1] https://commons.wikimedia.org/wiki/File:DHCOM_Computer_On_Module_-_AM35x.jpg