Back-end firmware structure

Tracker Phase II DAQ workshop - 22/03/2016

C. Bonnin, N. Chanon, L. Charles, L. Gross,
J. Hosselet, D. Tromson

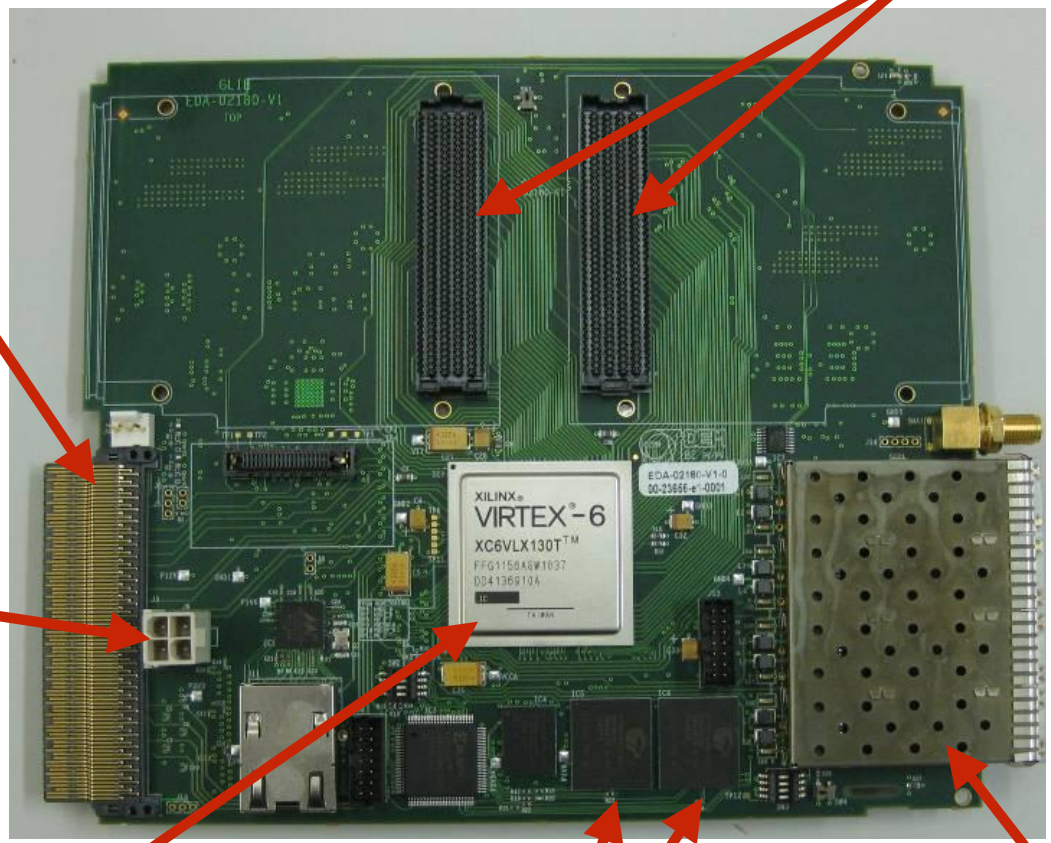
Back-end card: GLIB

GLIB (Gligabit Link Interface Board)

- AMC standard (Advanced mezzanine card)

2 **FMC** slots supporting **HPC** mezzanines

µTCA connector



Port **Gbit Ethernet**:
1 Gbps with IPBUS

FPGA **Virtex 6**
(~130k logic cells)

2*72Mb **SRAM** (static)

4 I/O SFP+ native cages, up to 6.5 Gbps (used to connect 2 GLIBs)

GLIB firmware

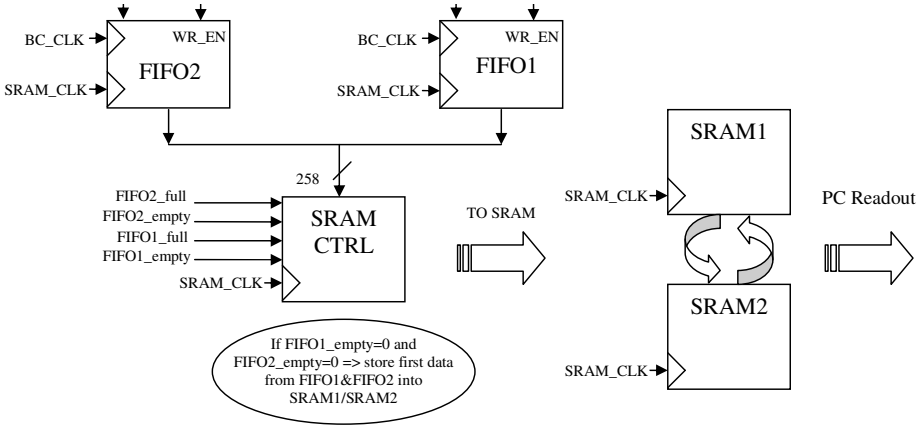
GLIB base system blocks

- **Core DAQ block:** data readout
- **I2C management block:** I2C transactions to configure CBCs
- **IPBus communication:** communication with software via Gbit ethernet
- **FMCs requirements:** trigger and clock + connection to CBCs
- **GBT emulation block:** emulation of LP-GBT

Data readout block

Data readout block

- **Trigger start_daq** action on board
- **Mode storage in continue** (simultaneous read write) using **two GLIB RAM banks**:
  - Data read out from Front-End are stacked into GLIB RAM bank 1
  - When GLIB RAM bank 1 is full
    - Data read out are stacked into GLIB RAM bank 2
    - In parallel, software reads out content of GLIB RAM bank 1
  - When GLIB RAM bank 2 is full
    - Data read out are stacked into GLIB RAM bank 1
    - In parallel, software reads out content of GLIB RAM bank 2



- All readout cycles are based on a synchronization/handshake mechanism between software and firmware
- In case software readout is too slow, trigger is back pressured: no data loss

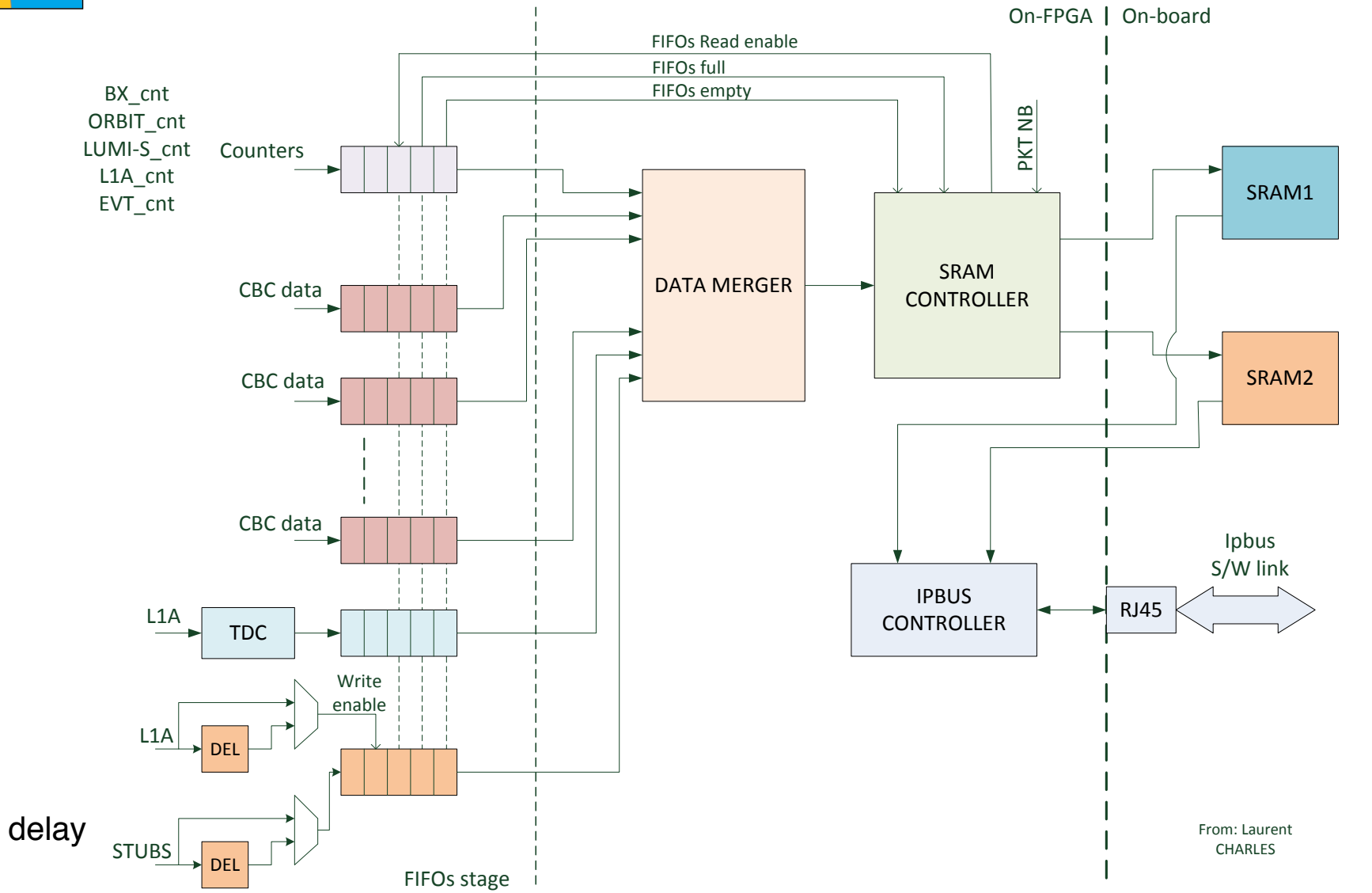
Motivation for two SRAM

A readout architecture adapted to beam test

- In beam test, IPBUS is a simple protocole for communication with software
- **IPBUS introduces long latencies**, due to handshaking, for read/write operations
- With only one SRAM bank, data write operations and software read operations could not be operated at the same time
- With **two SRAM banks**, there is always one of the RAM available for read operations from software : **no dead time**

- In real CMS conditions one would use a **FEROL** card, link to CMS DAQ
- FEROL communication at 6 Gbps (IPBUS <1 Gbps): FEROL is immediately available
- Possibility to write data in FIFO, directly read by FEROL
- FEROL provides buffers : the two SRAM architecture is not needed

GLIB functional diagram (data readout)



I2C Block

I2C Read/Write management block

- I2C = Inter-Integrated Circuit: protocole used to configure the CBC FE

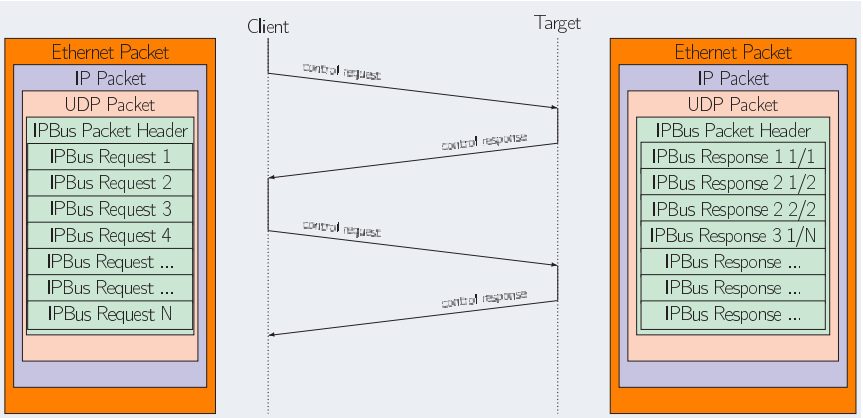
- In GLIB firmware, **I2C transactions are written in SRAM**
  - Save space on FIFO
  - Consequence: read/write of I2C values cannot be done simultaneously with the data acquisition

- **Write mode**
  - Write in GLIB RAM bank all the identifiers (I2C address + value to write), describing the write operations to be performed
  - Trigger write action on board
- **Read mode**
  - Write in GLIB RAM bank all the identifiers (I2C address), describing the read operations to be performed
  - Trigger read action on board
  - Read back I2C values from GLIB RAM bank

IPBUS communication

IPBUS through Gbit ethernet

- **IPBus protocol :** a simple IP-based protocol for controlling host hardware from a client computer
  - Transactional: each request has specific response message to allow confirmation



- **IPBus firmware :** interprets and implements IPbus transactions on FPGA
  - read/write and modify operations
  - transport protocol via UDP (simpler than TCP), correction for packet loss

- **Handshaking** : computer waits for GLIB to check its readiness state
  - Data-readout:
    - start DAQ with handshake on status_flags.CMD_START_VALID
    - each acquisition loop starts and ends with a handshake on flags.SRAM1_full or flags.SRAM2_full
  - I2C: write or read transactions contains handshake on cbc_i2c_cmd_ack

FMC requirements

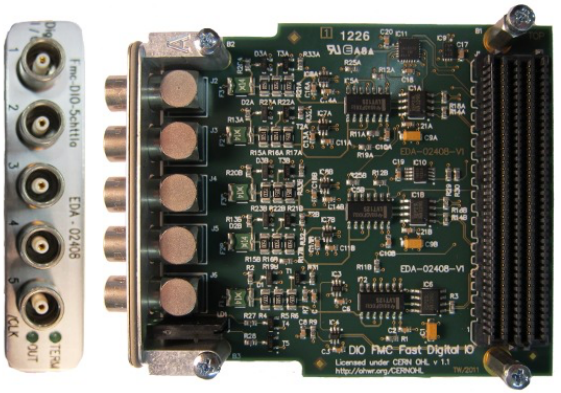
FMC (FPGA Mezzanine Cards) used up to now for beam tests :

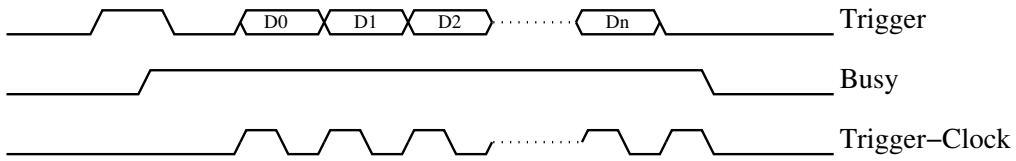
“Bristol” LVDS FMC

- Interface for CBC low voltage differential signal to GLIB
- Data read out to GLIB FIFO
- I2C transactions to configure CBCs
- No handshaking

DIO-5 (Digital I/O 5 channels) FMC

- Connection to **TLU** (Trigger Logic Unit)
- Distribute **trigger** signal from Telescope scintillators
- Data trigger handshake mode with GLIB:

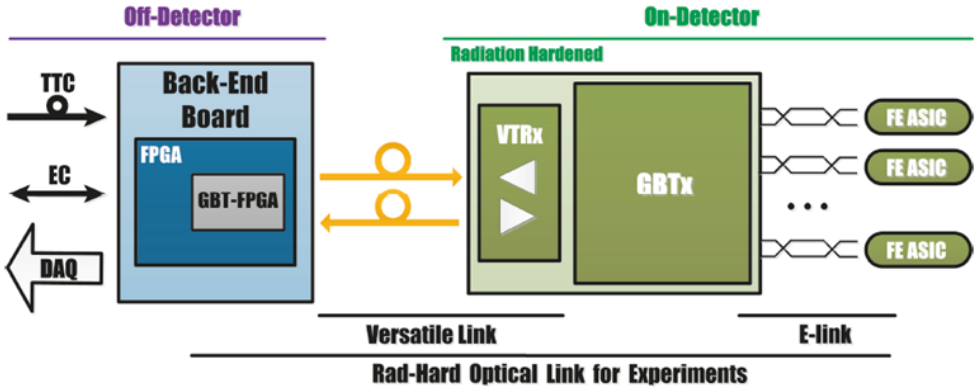




GBT emulation block

GBT (GigaBit Transceiver) emulation

- **GBT chip**: radiation hard ASIC with bi-directional optical link at 4.8 Gbps
- Trigger and timing, data acquisition and slow control
- In phase II tracker, LP-GBT will be used for communication between CIC and the back-end



- **In beam tests:**
  - During 2013 BT (two 2S mini-modules) and 2015 Nov BT (full 2S module), too many slots used for a single GLIB : **two GLIBs** were used
  - Back-end GLIB drives front-end GLIB, distributes clock
  - **GLIB to GLIB communication** via SFP+ optical link implementing **GBT emulation**
  - Back-end GLIB is transparent, trigger from TLU via DIO-5, communication with software via IPBUS
  - Front-end GLIB connection to CBC via FMC mezzanines

Where to find the code ?

All GLIB firmware versions for CBC2: https://svnweb.cern.ch/cern/wsvn/cmsptdaqup/

Mono-GLIB firmware versions:

- One 2xCBC-2 hybrid + DIO-5

https://svnweb.cern.ch/cern/wsvn/cmsptdaqup/CMS_UPGRADES_IPHC/FIRMWARE/PHASE_2/DAQ/TK_DAQ_MONO_GLIB3_1x2CBC2_FMCDIO5_V1

Version was updated for fixing stubs reading during June 2015 BT - *to be published*

- Two 2xCBC-2 hybrids, internal trigger only

https://svnweb.cern.ch/cern/wsvn/cmsptdaqup/Firmware/tags/tracker1.2.glibv3.dualcbc2

- One 8xCBC-2 hybrid + DIO-5

https://svnweb.cern.ch/cern/wsvn/cmsptdaqup/CMS_UPGRADES_IPHC/FIRMWARE/PHASE_2/DAQ/TK_DAQ_MONO_GLIB3_1x8CBC2_FMCDIO5_V1.2

Dual GLIB-based (GBT-FPGA used for FE<->BE communication)

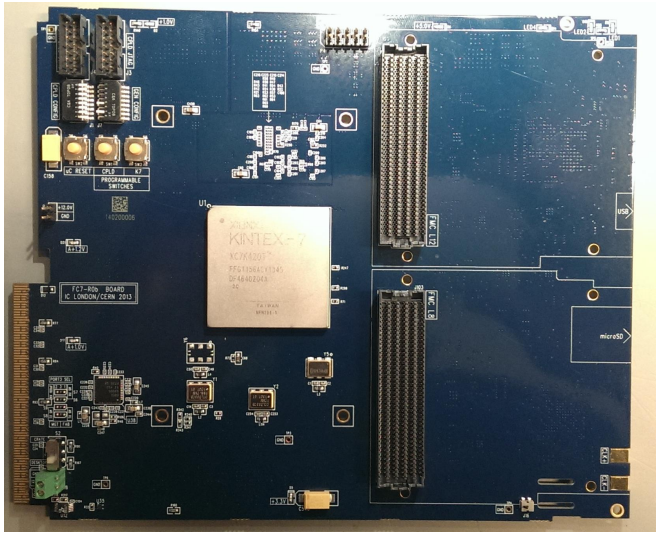
- Two 2xCBC-2 hybrids on FE board + TB-FMCL-PH FMC (LVDS connexion to TLU)

https://svnweb.cern.ch/cern/wsvn/cmsptdaqup/Firmware/tags/tracker1.2.glibv3.dualcbc2.desy2 013

- Two 8xCBC-2 hybrids on FE board + DIO-5 (TTL connexion to TLU)

Used at Nov 2015 BT, *Unpublished yet* - needs refurbishment

Migration GLIB to FC7



FC7 (FMC carrier - Xilinx Series 7)

- FPGA **Kintex 7** (~420k logic cells)
- **No native SFP+** cages
- **No port Gbit** Ethernet
- µTCA connector
- 4 Gb **DDR-3 RAM** (dynamic)
- 2 FMC slots supporting **LPC** mezzanines, located on the same side (front side of xTCA crates)
- µSD card to select f/w versions, I2C values…

Migration of base functionalities

- **Data readout block:** no significant change expected
- **I2C operations in parallel with data taking**:
  - With GLIB f/w, I2C and data readout were mutually exclusive
  - Can be directly written to FIFO since more FIFO will be available in FC7
- **No Gbit ethernet port, no SFP+ cage:** All test systems on crate, action plan needed for data readout and trigger/clock

Improvements foreseen:

- **I2C read-back** once write done (would have been useful at Nov BT with 2 GLIBs) + **write again** if wrong value
- **Broadcasting** + **1 MHz** I2C to be implemented.

Conclusions

GLIB firmware

- **DAQ** Core block and **I2C** management block
  - Data readout adapted to beam tests, with continuous read and write on two SRAMs
  - I2C transactions via SRAM to save space on FIFO
- **IPbus, GBT** and **FMC** blocks: mainly interfacing

- **Code** is available at https://svnweb.cern.ch/cern/wsvn/cmsptdaqup/

- Next back-end card : **FC7**
  - Migration of base functionalities ongoing
  - More functionalities foreseen

Back-up slides

FC7: Test system on crate

FC7 does not have Gbit Ethernet port

- How to send the data to software ?
- If one wants **IPBUS** communication (e.g. for beam test), it has to go through the crate backplane: data readout through crate MCH1 slot
- Can also use **FEROL** card (link to CMS central DAQ)
  - No FEROL FMC (as far as we know), a FMC with SFP+ would work
  - slot MCH2 with AMC13 (has optical output 10 Gbps)

Trigger and clock:

- Until now in beam test, trigger from TLU through **FMC DIO-5**
  - Feasible for one FC7 card on crate, not possible for many FC7
  - Options to distribute clock on half a crate from one FC7+DIO-5: to be investigated
- **CMS Trigger Timing and Control system:**
  - CERN designed a TTC FMC for GLIB (high pin count mezzanine). Could be tested with FC7 (low pin count)
  - TTC support from back-plane is preferred over TTC FMC. AMC13 ?

Every change of architecture needs firmware development

Longer term plans

FC7 for CBC3/MPA + CIC

- **Reading out CIC data ?**
  - CIC prototype available by the end of 2016
  - 2 CIC chips per module, input each 8 CBC3 / MPA chips
  - CIC output data sent to **LP-GBT**
    - LP-GBT (65nm) not yet finalized, connect with CIC through GBT (130nm) instead ?
    - LP-GBT emulator already in place
  - How to handle **CIC I2C** registers ? (e.g. CIC configuration, number of lost stubs)
  - How will change **data acquisition** ?
    - Adapt to CBC3 / MPA formatted by CIC
    - CIC output data generator is available for tests
  - How to handle **L1** data to Tk-finding, and L1 decision for data acquisition (final DTC) ?