LHCb Electronics Upgrade Meeting

Data Format

Thursday 12th December, 2013

Frédéric Machefert

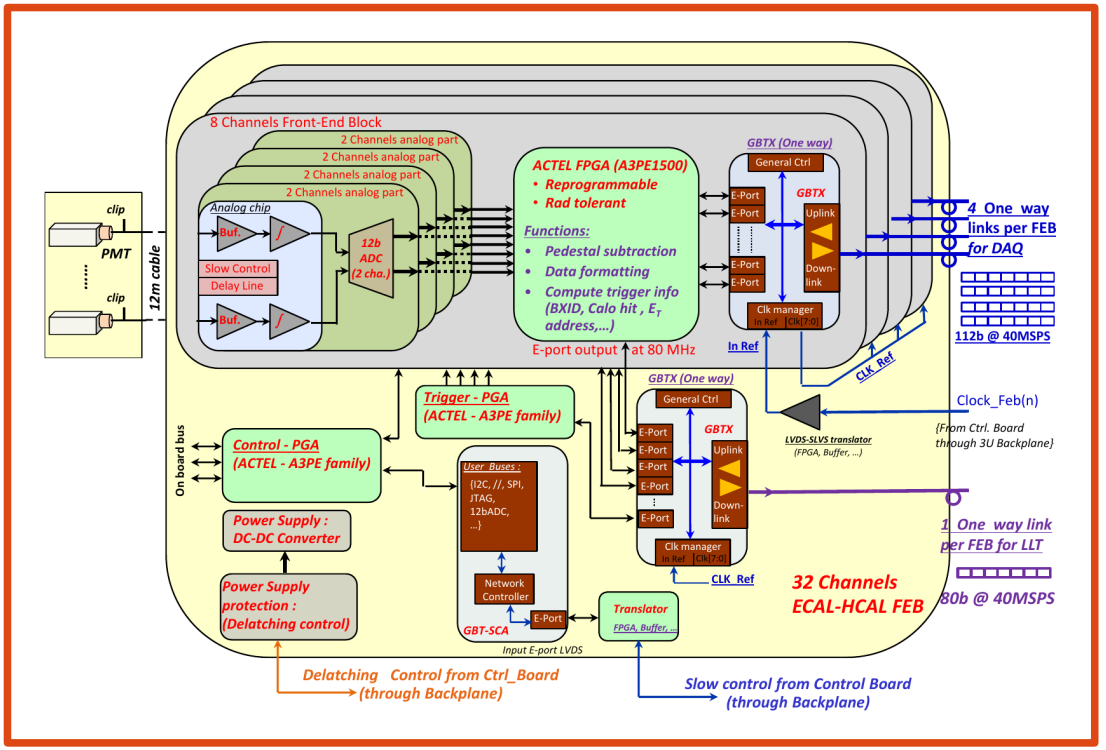
Laboratoire de l'Accélérateur Linéaire

On behalf of the calorimeter upgrade group

Overview of the Front-end

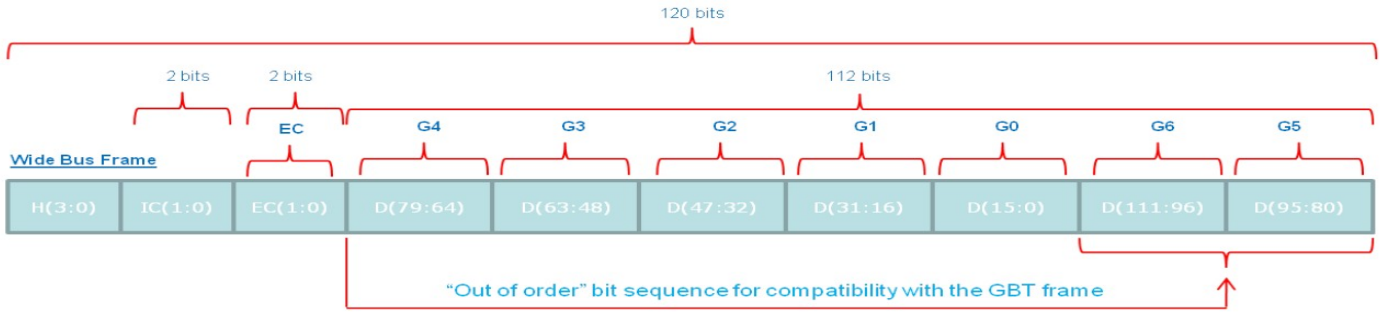
- The calorimeter data are produced by ~250 FEB
- Each FEB receives the signal from 32 PMT and converts the signal into 32 words 12 bit wide (12 bit ADC)
- The almost « unavoidable » scale for the FEB is
  - 4 FE blocks of 8 channels / FEB → 4 opt. Links / FEB
  - 8 x 12 = 96 bits to be sent per block
- 4 Fibres for the DAQ
- An extra fibre is needed for the LLT-Calo
  - FEB/TRIG40 connection
- To send 96 bits, the « standard » (80 bits) data-format of the GBT cannot be used
  - Need to base our design on the « wide-bus » frame implementation
  - Not needed for the LLT-Calo fibre

Digital design



- 5 fibres are necessary to send
  - the Data (4)
  - the LLT-Calo (1)
- No complex compression of the data
  - Simplicity
  - Reduction of the risk of data corruption
- LLT is basically a copy of what is in the L0-Calo except for the opt. link

Data format - DAQ



- We need to base our design on the 112 user bit « wide-bus » frame
- Out of the 112 bits, 96 used for the channels data
- 16 remaining bits for synchronisation/identification
  - Crate identifier (5 bits, could we save 1 bit separating ECAL/HCAL ?)
  - FEB identifier (4 bits)
  - 7 (8?) bit for the BxID
- BxID coded on 12 bits → we miss a few bits
  - 2 solutions
    - Send only 7 (8 ?) bits → probably sufficient for synchronisation
    - Split 12 bit word and send MSB on several links (4+2+2+2+2, 1 extra bit free per link) → more complex

Bx-id	Crate-id	FEB-id	ADC-7	ADC-6	ADC-5	ADC-4	ADC-3	ADC-2	ADC-1	ADC-0
[111,105]	[104,100]	[99,96]	[95,84]	[83,72]	[71,60]	[59,48]	[47,36]	[35,24]	[23,12]	[11,0]

- Need to find an extra bit for the « no data bit » → could be the saved bit from ECAL/HCAL separation
- Data_length not useful → this is a constant for us
- In this design there is no margin
  - no possible correction, nor any parity bit for checking
- The « synchronisation » data could consist in sending a (wrong) specific pattern on the FEB/CRATE id

Data format – LLT-Calo

- The data format of the LLT-Calo (FEB-TRIG40 connection) is not so demanding as for the DAQ. The data to send are (preliminary)
  - 5 bits of local address
  - 8 bits of ET of the highest ET 2x2 cluster
  - 13 bits of total ET on the board
  - 5 bits of crate identifier (4?)
  - 4 bits of FEB identifier
  - 6 bits of multiplicity
  - 1 bit of status
- A total of 42 bits → the standard format is sufficient
- No complex compression needed
- Data correction can still be used on this link
  - Probably more useful on the LLT path than on the DAQ path

Some ideas

- We « lose » the CRATE/FEB id bits
  - Do we need them all the time ?
  - Could we define a « test » mode when we emit FEB/CRATE id → check optical link connexion
  - Could we send the FEB/CRATE id on the data frame only when SYNCH command is active (having a synchronisation bit to identify the frame)
  - Recover 9 bits (« physics » mode) for
    - Full Bxid coding
    - Some protection bits
  - However, this would request 2 working mode for the backend

- Could we have some protection bits in the LLT-Calo link to correalte ADC/LLT information
  - Is it necessary ?
  - Is it feasible at all in the online ?
    - The links do not get to the same hardware...