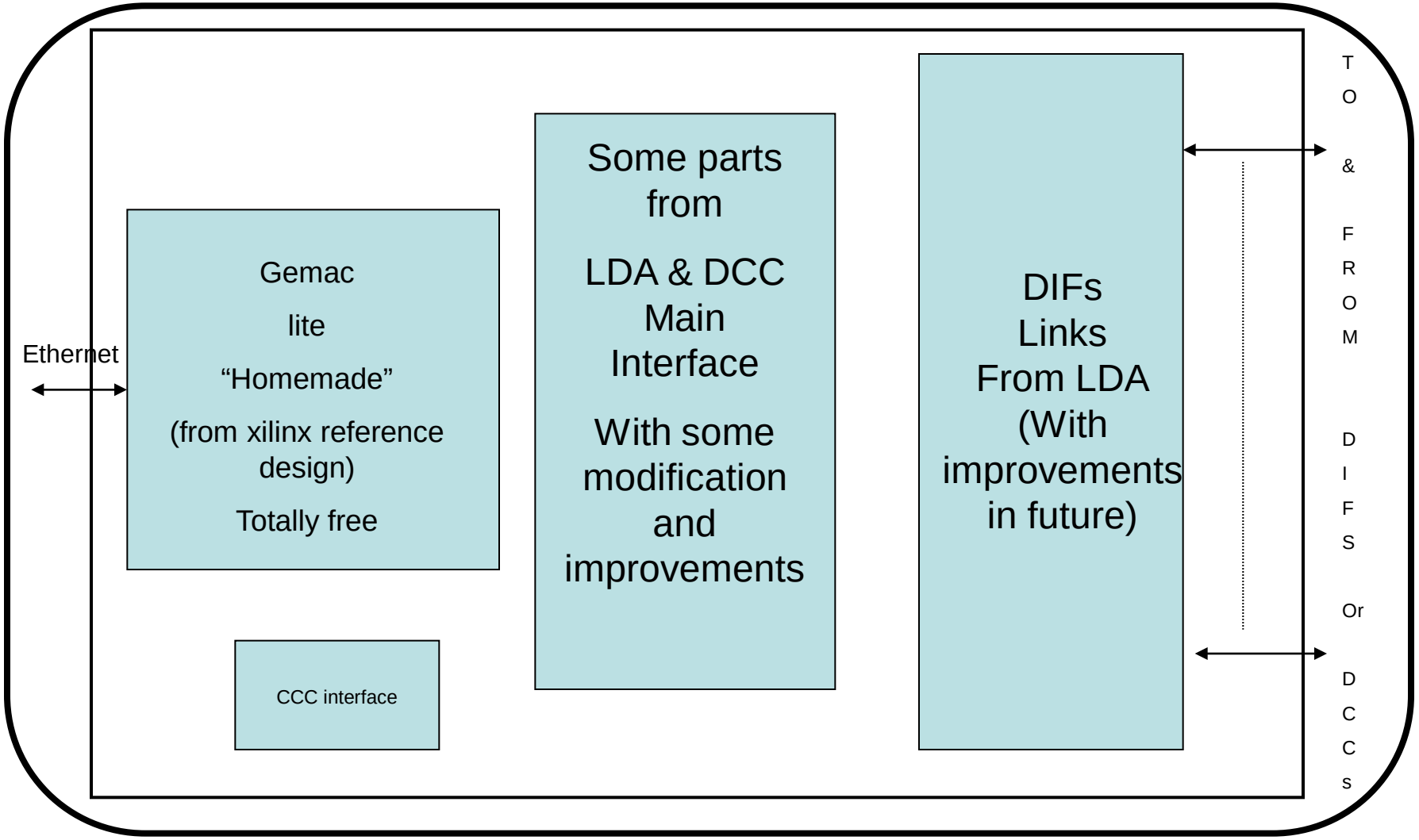
GDCC

Gigabit Data Concentrator Card

Franck Gastaldi
Remi Cornat

GDCC overview



GDCC board specifications

- Main component is a Xilinx Spartan 6 SLX75 FG676-2
- Ethernet connection by fiber or copper cable
- USB2 access for test and (if it work, jtag conversion for FPGA programming)
- Optional VME access foreseen (system management)
  - 8 bits data, 16 bits address, 1 IRQ, No arbitration
  - GDCC could be master (to be tested) or slave by default

  Remark : *not a priority for the firmware development but the hardware will be in place*.
- Some VME signals will be replaced by J-tag access. We will use a specific component (SCANSTA112 : Jtag Bridge)
- 1Gb DDR2 SDRAM will be implemented on the board.
- 1 HDMI connection for the CCC board
- 6 HDMI connection for the DIF boards
- 1 footprint will be foreseen for a new connection with the CCC
- Clock and Trigger will be distributed on FPGA and DIFs via 2 fan-out component with low jitter (MAX9153 : 100ps p-p jitter max, 60ps skew between channels)

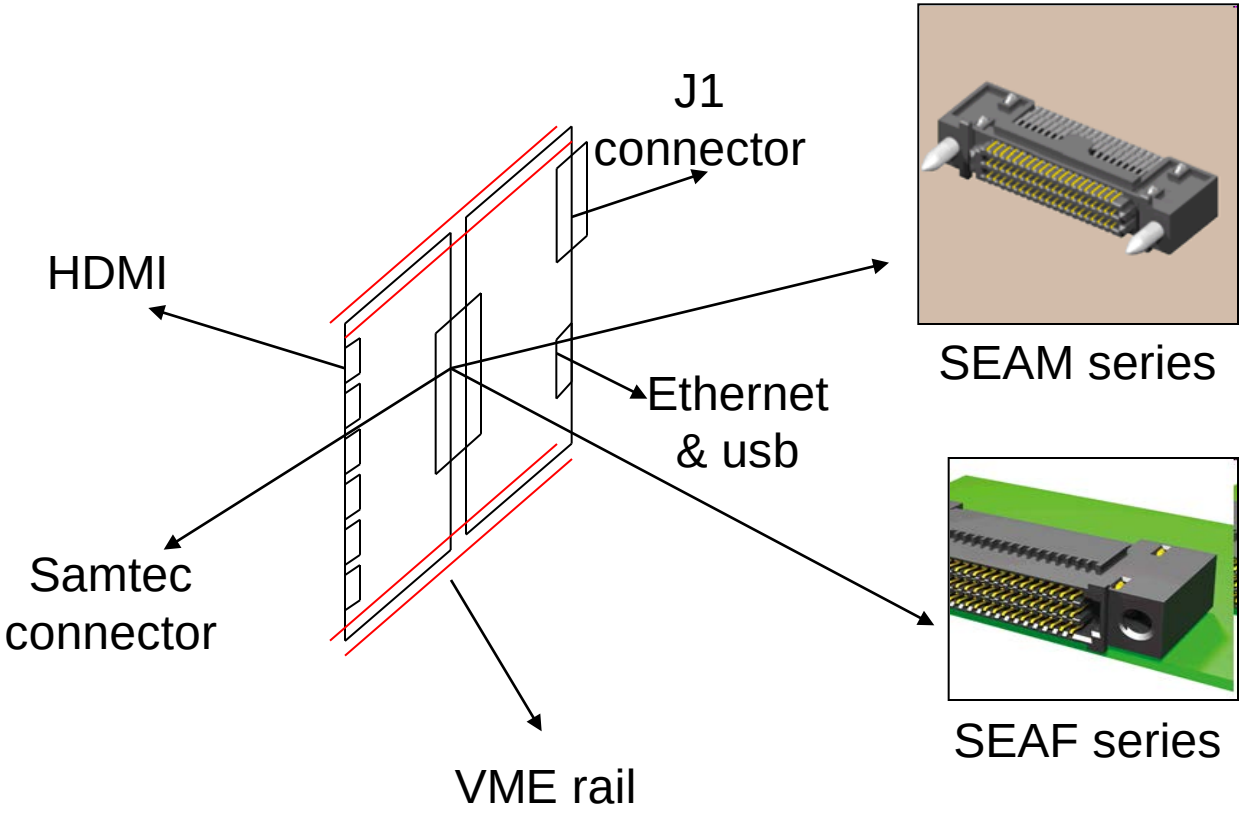
GDCC Board mechanic

- Format : VME 6U (chassis with only J1 connector), chosen for low cost and availability
- Format shared in 2 part (1/3 – 2/3)
  - 1/3 is the mezzanine with the HDMI connections
  - 2/3 is the GDCC “heart” with the main functionalities
- Reliability of mezzanine by a specific Samtec connector (SEAM and SEAF series: 160 pins high speed differential)

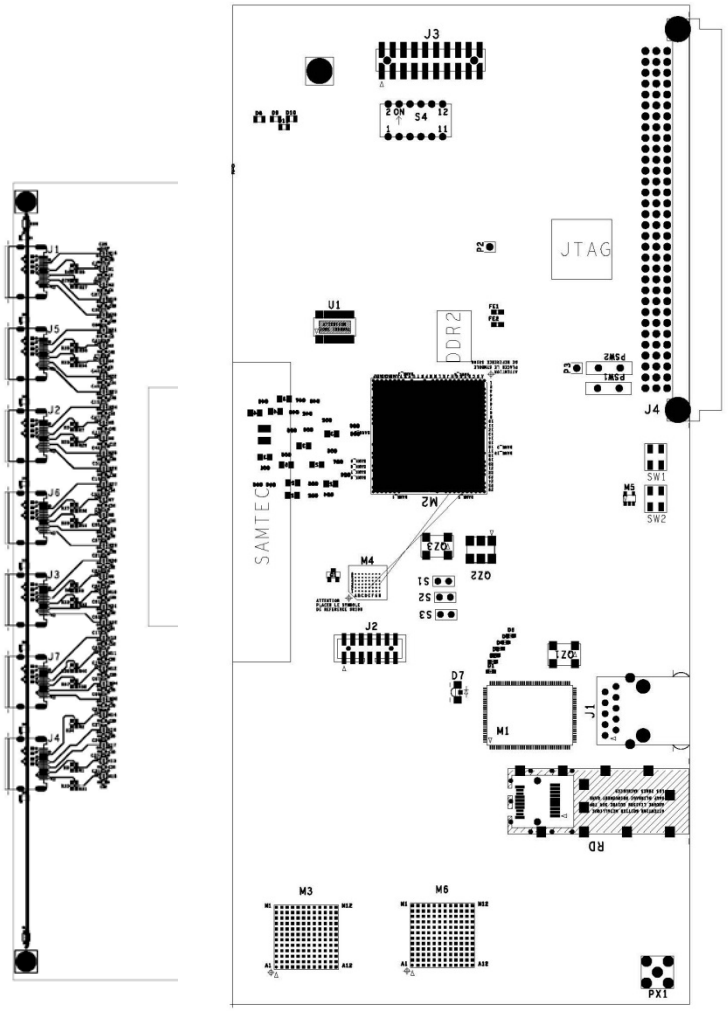
VITA57 standard

GDCC Board integration

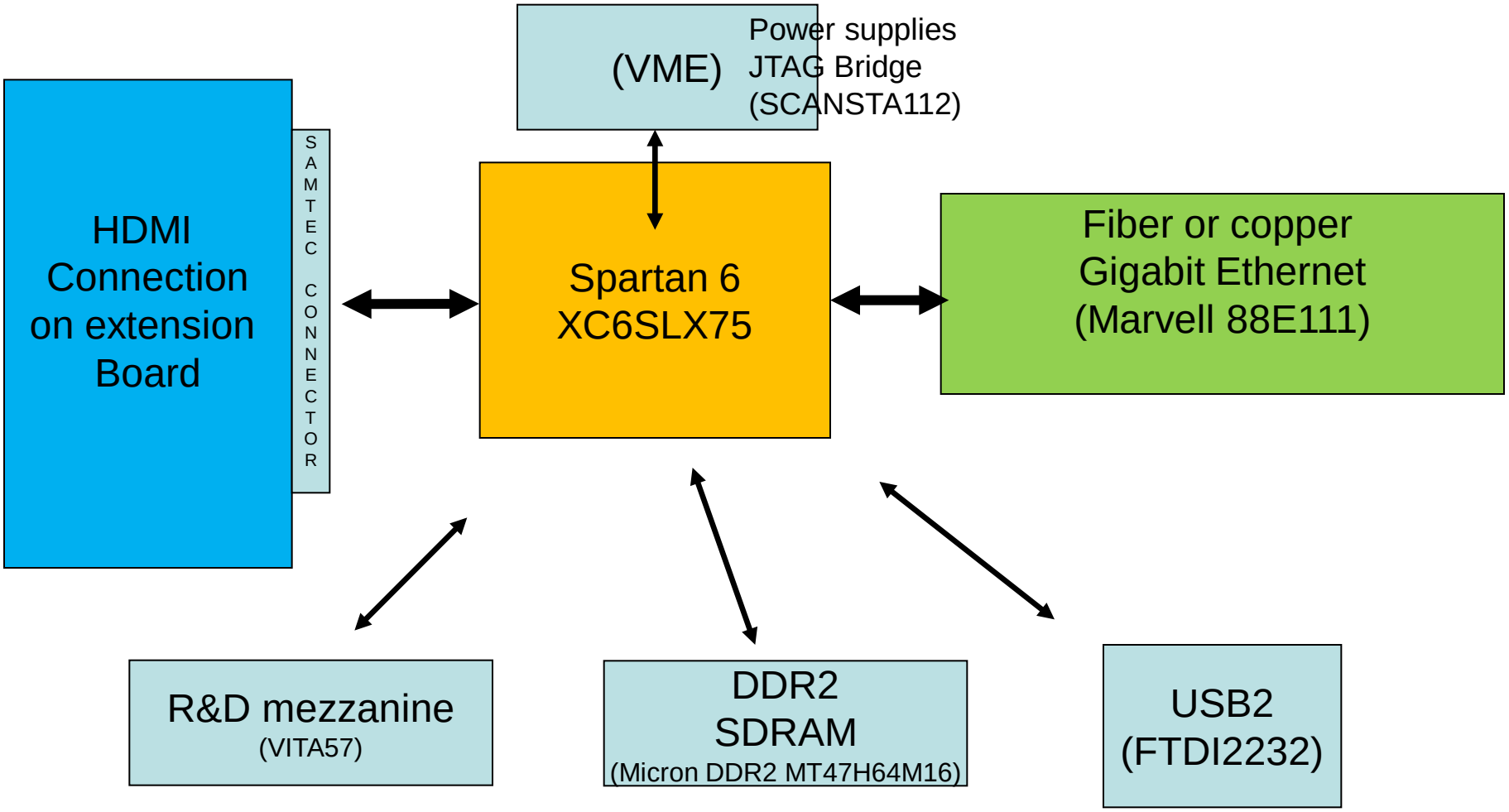
Board to board solution



Preliminary layout



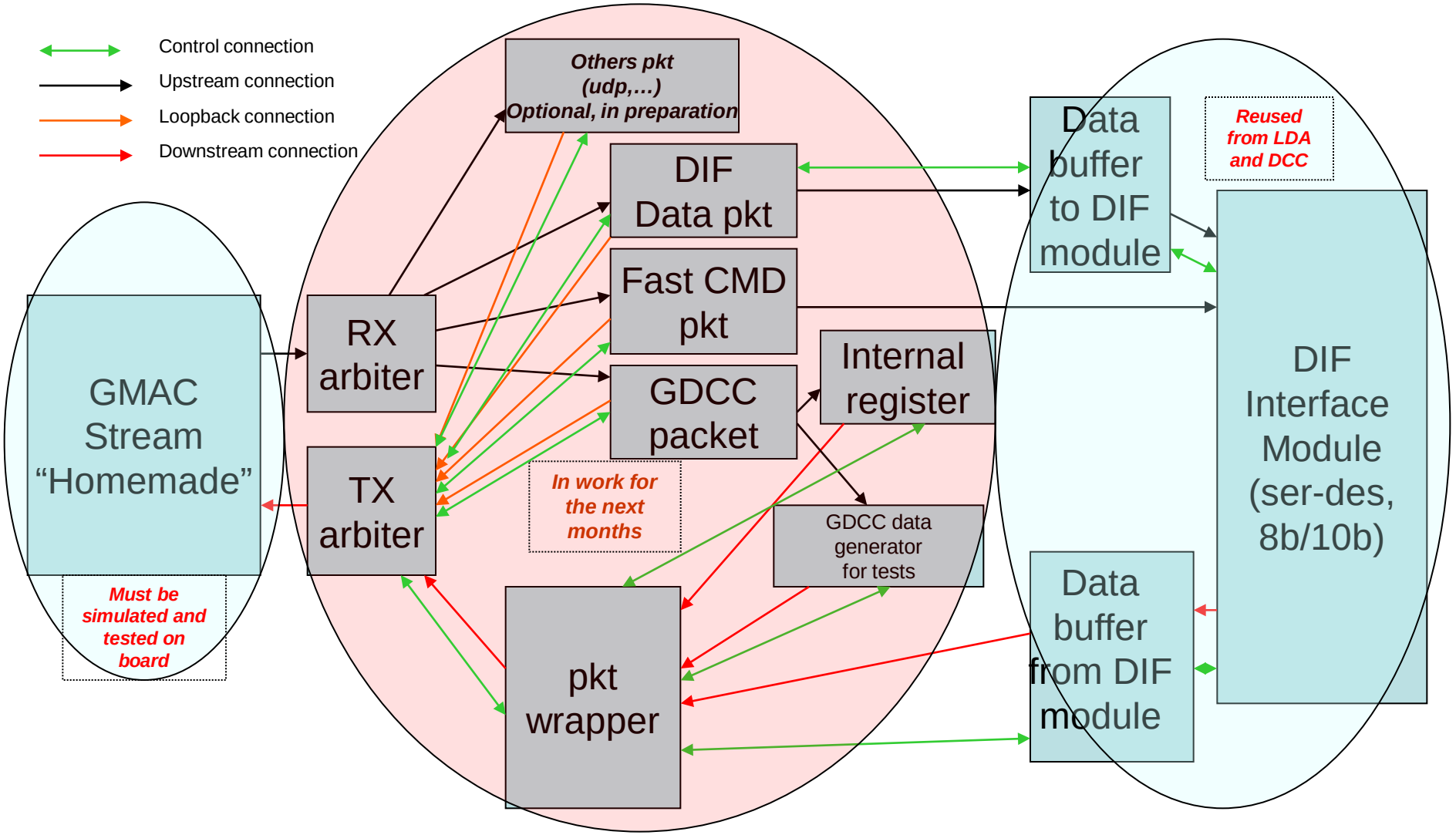
GDCC simple architecture



Firmware

- Mostly reuse VHDL module from LDA and DCC
- Improve the functionalities that weren't perfect during the DHCAL test beam
- Add new functionalities *that have been forget or not thought*…
- Simulations and tests on Xilinx evaluation board (SP601)
- Due to the resources in spartan6 on SP601, we will design a firmware for only 2 or 3 DIF Channels

Firmware architecture (In work)



Cost estimation (<5 boards)

Prototype

Estimation for manufacturing lead-time 15 days min.

- Per board (GDCC main)
  - Board : 300 €
  - Cabling : 300 €
  - Tools : 1200 € / 5 = 240€
  - Component : ~200 €
  - Total : 1040 €
- Per board (mezzanine)
  - Board : 200 €
  - Cabling : 200€
  - Tools : 1200 € / 5 = 240 €
  - Component : ~70 €
  - Total : 710 €
- **TOTAL GDCC** : 1750 € /board (+/- 20%)

The price could be lowered making only one board without mezzanine (real 6U board) or try to find a low cost company (quality ?).

Conclusion

- In couple of week
  - Finish the schematic: waiting an update of the CADENCE library with 3 last components
- The routing started two weeks ago. Expect 2 or 3 months for executing this task. One person works on it and shares his time between several projects
- Until July: put in place the firmware on evaluation board and to prepare the final firmware for the GDCC (1 student will help)
- Consider to send a first board for manufacturing during the summer

Spare

Ethernet packet format structure (1/2)

- Based on LDA document (CERN CALICE twiki):
  - Fast command packet format

Dst MAC	Src MAC	Ethernet Type	Command_Word	DIF Link	Comma	Data	parity	PAD	CRC32
6 Bytes	6 Bytes	2 bytes	2 Bytes	2 Bytes	1 Byte	1 Byte	2 Bytes	Pad to min ethernet size	4 bytes

  - Ethernet type: set to 0x0809 for Fast Command
  - Command word : set to a constant 0xFA57
  - DIF Link : Mask which defines which port the command is for. 0xFFFF is a broadcast
  - Comma : comma character to use
  - Data : send as fast command data
  - Parity : simple check based on even parity scheme

Ethernet packet format structure (2/2)

– GDCC packet format (previous LDA packet format)

Dst MAC	Src MAC	Ethernet Type	GDCC_type	GDCC_modifier	GDCC_pktID	GDCC_dataLength	GDCC_Data	PAD	CRC32
6 Bytes	6 Bytes	2 bytes	2 Bytes	2 Bytes	2 Byte	2 Byte	Variable	Pad to min Ethernet size	4 bytes

- **Ethernet type**:
  - 0x810 : GDCC data pkt
  - 0x811 : DIF data pkt
- **GDCC type**: split in 2 bytes (***not all define currently***)
  - Upper:
    - **0x00** : GDCC registers
    - **0x01** : DIF transport
    - **0xFF** : GDCC pkt generator
  - Lower:
    - **0x01** : Write from PC to GDCC
    - **0x02** : Read from PC to GDCC
    - **0x04** : Read reply from GDCC to PC
    - **0x08** : pkt_DIF from PC to DIF
    - **0x09** : pkt-DIF from DIF to PC
- **GDCC modifier** : indicates which DIF link receive the packet. 0xFFFF indicates a broadcast
- **GDCC pktID** : used o track replies
- **GDCC data length** : How many objects will be in the GDCC data
- **CRC32** : not really used for the moment, but maybe visible.