

# Level-2 calorimeter trigger Upgrade: Technical Part

\* For comments, suggestions, comments please contact : [sartori@fnal.gov](mailto:sartori@fnal.gov)

## Abstract

This is an internal document used by people involved in the technical aspects of the upgrade just to take trace of the improvements, problems or open questions and, in order to have a self-consistent document, few sections contain explanations already given in other papers, notes or documentation. This is an ongoing document and is NOT an official document. Any contribution, suggestions, comments is welcome.

# 1 Overview of L2CAL upgrade hardware configuration

At the hardware level, the basic idea of the L2CAL upgrade is to use Pulsar boards to receive the raw (full 10-bit resolution) trigger tower energy information from the DIRAC boards, merge and convert the data into SLINK format, then deliver the SLINK package to the L2 decision PC. This is very similar to what has been done to all the other L2 trigger data paths for the L2 decision Pulsar upgrade. In that sense, this can be viewed as a natural expansion of the L2 decision upgrade. In fact, since the clustering algorithm would now be done in software inside the L2 decision CPU, the proposed L2CAL system will be much simpler and much more uniform at the hardware (and firmware) level.

For the existing L2CAL system, since the actual clustering (also isolation) is done in hardware (designed in the mid 90's), the system is quite complicated. The entire system consists of 86 9U VME boards in 6 VME crates with a custom P3 backplane, including 72 DCAS, 6 LOCOS, 1 CLIQUE, 6 IsoPick and 1 Iso-Clique boards. The proposed L2CAL upgrade system will consist of 18 (new but identical to the present ones) Pulsar receiver boards, and 6 existing Pulsar SLINK merger boards. Since the Pulsar receiver only needs to receive the raw data and convert it into SLINK format, the firmware for the Pulsar receiver board will be simple. In order to receive the trigger tower energy LVDS signals from the DIRAC boards, a new Pulsar mezzanine card will need to be designed. One mezzanine card is able to receive 4 cables from DIRAC (corresponds to one DCAS input data), thus one Pulsar board can receive 16 cables or 4 DCAS input data.

Figure 1 shows the calorimeter related trigger subsystems for both Level 1 and Level 2, with the red part being the new L2CAL path. To minimize the impact on the running system, i.e. to be able to run in pure parasitic mode during commissioning, we will make a copy of the LVDS input signals (just as we did for all other L2 trigger paths for the L2 decision upgrade). In this case, we will use the LVDS “multi-drop” property, and make long cables in such a way (see Fig. 2) that each DIRAC output signal LVDS cable first has a “drop” at a Pulsar mezzanine card (without termination during commissioning), then goes to the existing DCAS input (which has 100 ohm termination). In other words, the signal splitting is

simply being done with long cables having one additional connector (see Fig. 2).

In the current system, one DCAS board receives four input cables. In the new system, one Pulsar mezzanine card will receive the same amount of input data as one DCAS. Figure ?? shows the mezzanine card design. Note that the length of the mezzanine card will be doubled to allow easy access to all four LVDS cables. With four mezzanine cards per Pulsar board, 18 Pulsars (in two crates, see Fig 3) will be needed to receive all input data. The rest of the system consists of existing Pulsar SLINK mergers (see Fig. 4). Note that the Pulsar based L2CAL will be very flexible, just like the rest of the Pulsar based L2 system. In fact, if really needed, it is even possible to implement the L1 MET trigger using the full 10-bit calorimeter energy information with Pulsar hardware.

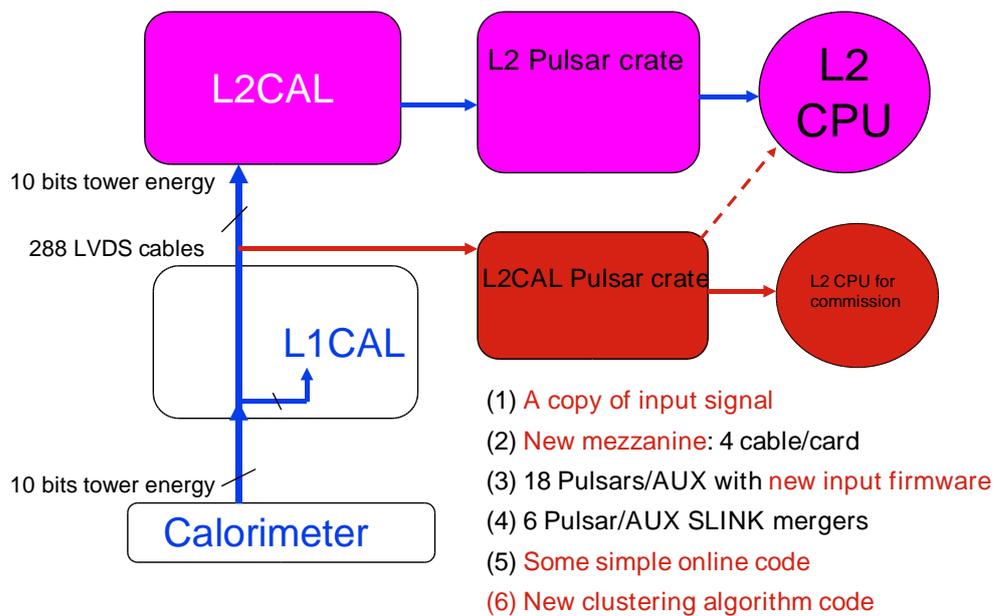


Figure 1: The red path represents the new hardware to be added to the calorimetric trigger system.

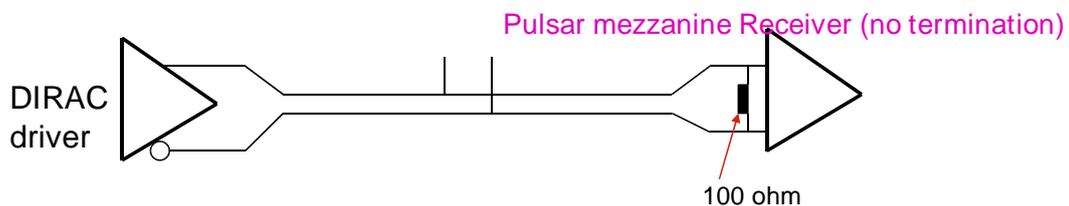


Figure 2: The bypass on the data path to operate the new and old systems together.

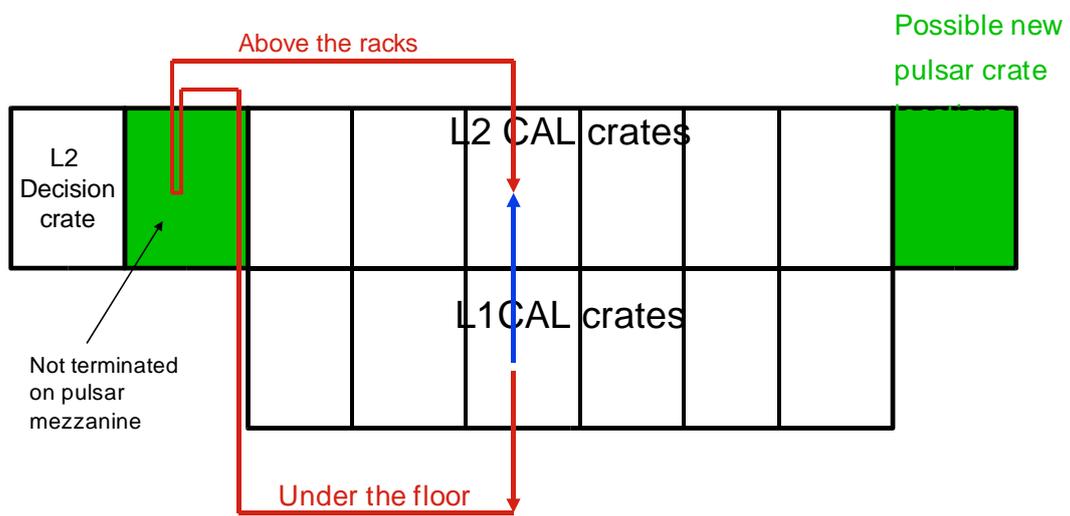


Figure 3: The new Pulsar crates (green) and the basic idea of new cabling (red).

(1 Pulsar: 4 mezzanine x 4 cable = 16) x 18 = 288 input cables total

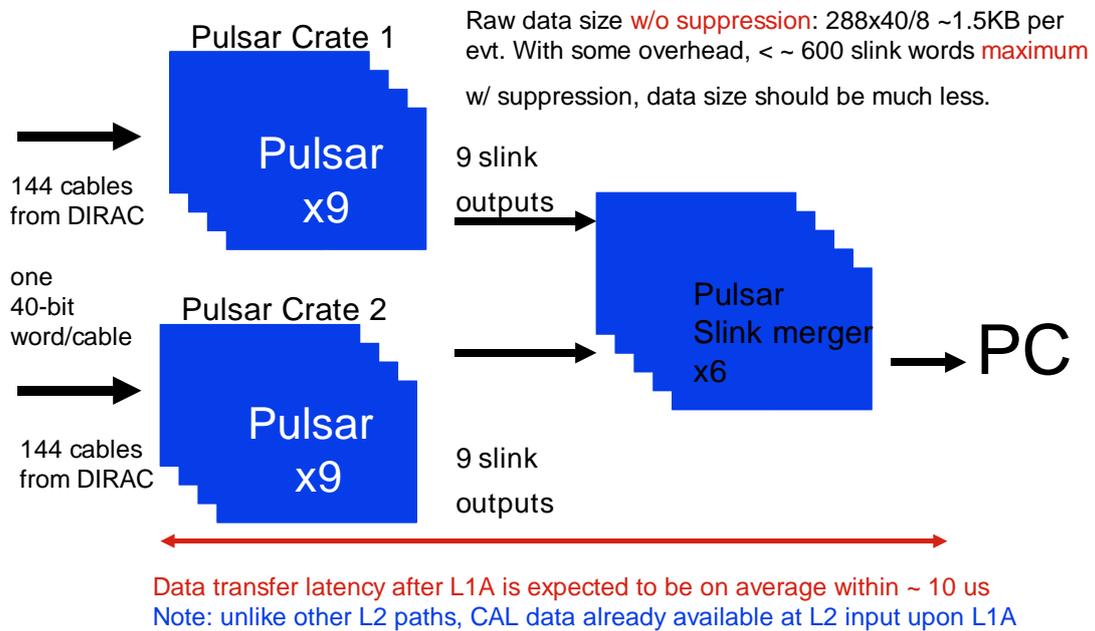


Figure 4: The 24 Pulsars to merge data from the DIRAC cards directly into the Level 2 decision PC. Data transfer from L1 accept to the L2 decision PC is expected to take on average less than  $\sim 10\mu\text{s}$ .

## 2 Pulsar Design Overview

The Pulsar Cluster must take care of receiving LVDS signals (288 cables), processes data and merge them into a single SLINK cable connected to the L2CPU.

The SLINK merger pulsars (see fig 3) have been already developed and tested during the last L2 upgrade. So in the following we describe the design of the 18 pulsars receiving the LVDS signals from DIRAC. The data flow through the pulsar is shown in figure 5.

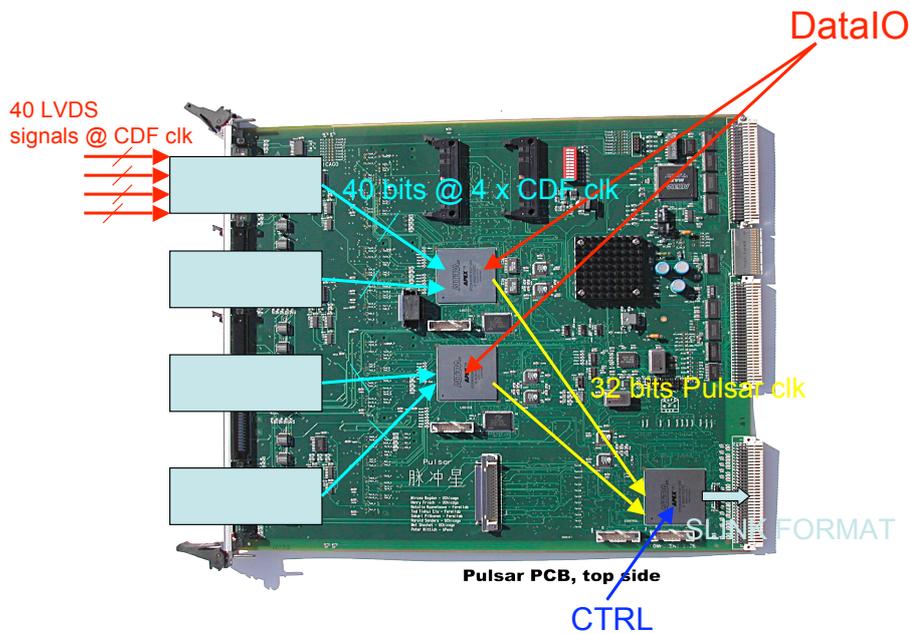


Figure 5: The main blocks of the new Pulsar: 4 Mezzanines and 3 programmable devices (2 IOs and 1 CTRL) Pulsar receives 16 LVDS cables @ CDF clk and the output is a SLINK-format data

The basic tasks of these new pulsars are

- 1 Receive LVDS signals and convert into TTL. Each Board receives 16 LVDS cables (4 on each mezzanine), corresponding to 32 towers.
- 2 Select only the events confirmed by the L1A
- 3 Select only the towers with Non-zero Energy (Zero suppression)
- 4 Flag the tower energy information with the index of the sending tower.
- 5 Merge them into Slink format

The first item will be performed inside the mezzanine card. Items 2,3,4 will be performed inside the IO1 and IO2 FPGAs of the Pulsar and the last one (merging the data) will be distributed in the 3 FPGA along the data flow, starting on the mezzanine and ending on the CTRL FPGA. The firmware of the CTRL already exists, the only parts we have to take care are the Mezzanine design and the IO1-IO2 FPGA firmware.

For each board, 4 **mezzanine cards** receive LVDS signals at the frequency of CDF clock (132 ns), 10 ns within the rising edge. Each mezzanine card converts LVDS into TTL signals using 4 sets of ten receiver chips (we call them Rxs in the following), one set for each cable. The TTL signals go from the Rx chips directly to the FPGA where they are simply latched (160 latches shown in figure 6 as input registers) at the CDF clock frequency, and sent in serial mode (40 bits/word) to the Pulsar motherboard at the frequency of 4 x CDF (See Mezzanine specification for more details). One 40 bits output register interfaces with the Pulsar. Two additional bits take trace of the connector number.

The two **IO FPGAs** receive the four mezzanine outputs at the frequency of 4XCDF clock (33 ns)(see fig 7), preprocess data and merge them into a Slink output. The IO FPGA contains two main logic blocks:

- One main block processes and synchronizes data. In figure 7 it is indicated as Data Controller. For each IO FPGA we have two data controllers, each one receives data from one mezzanine. In fig ?? the main functions of the Data Controller are highlighted: input data are stored in a first block until a L1A confirms them. If they are confirmed a second block completes each data word (tower energy) with an identifying address (Tag Block). The last block performs the zero suppression tower. The two IOs also synchronize the data with the Pulsar clock (80 MHz).
- The output of the two Data Controllers are then merged (fig. 7) and sent to CTRL FPGA.

CTRL takes care of merging the data coming from the two IOs and sends them to SLINK transmitter.

## MEZZANINE FPGA (not detailed)

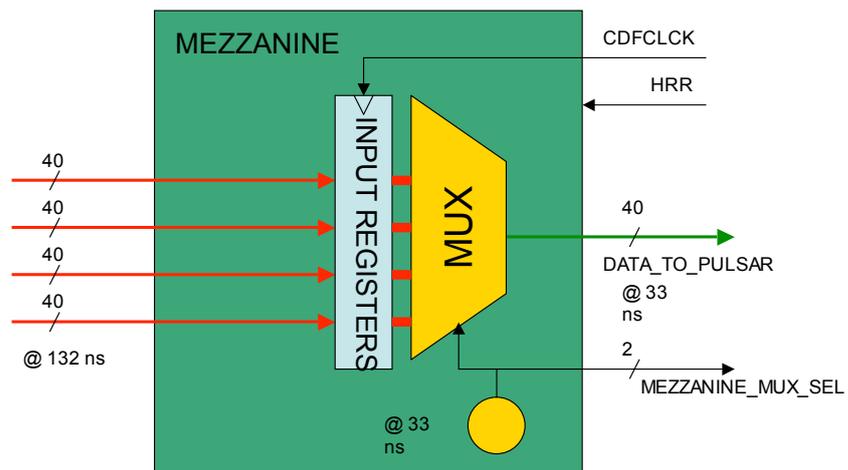


Figure 6: High Level FPGA Mezzanine schema. Data are latched at CDF clk frequency and sent to the Pulsar motherboard at the frequency of 4 x CDF clock.

## DATAIO (not detailed)

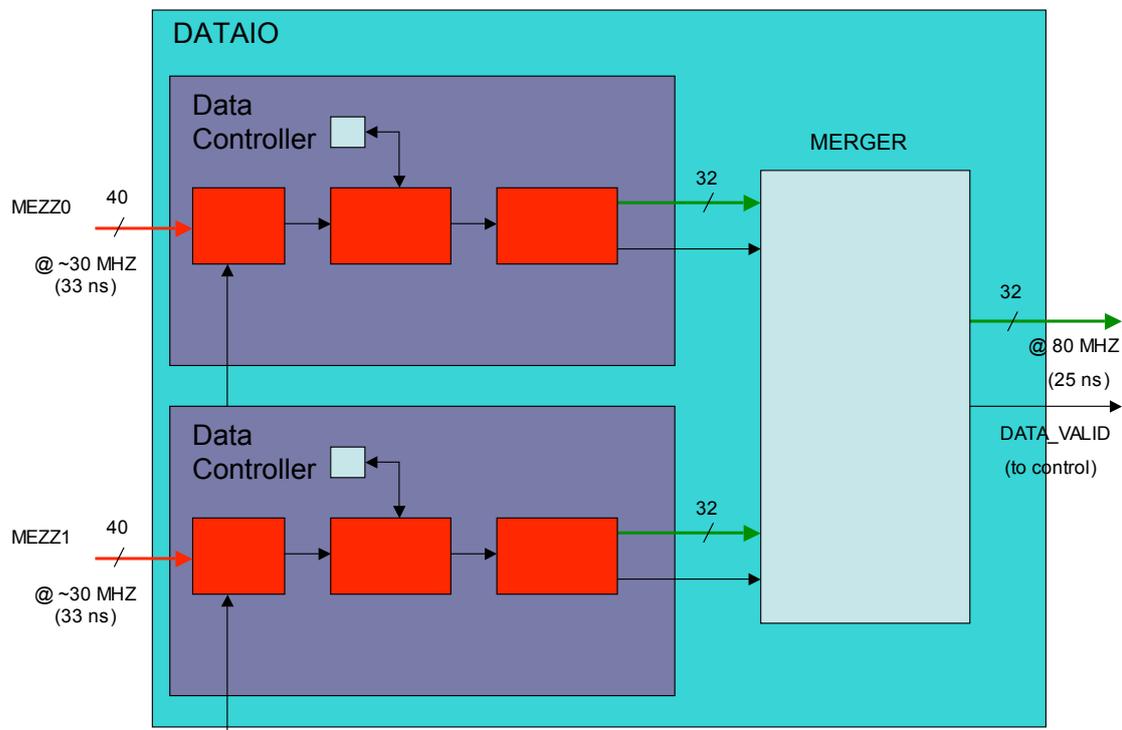


Figure 7: DataIO FPGA scheme. The FPGA receives data from two mezzanine cards, controls and processes data in a Data Controller block and merge them into a Slink output

## Data Controller (not detailed)

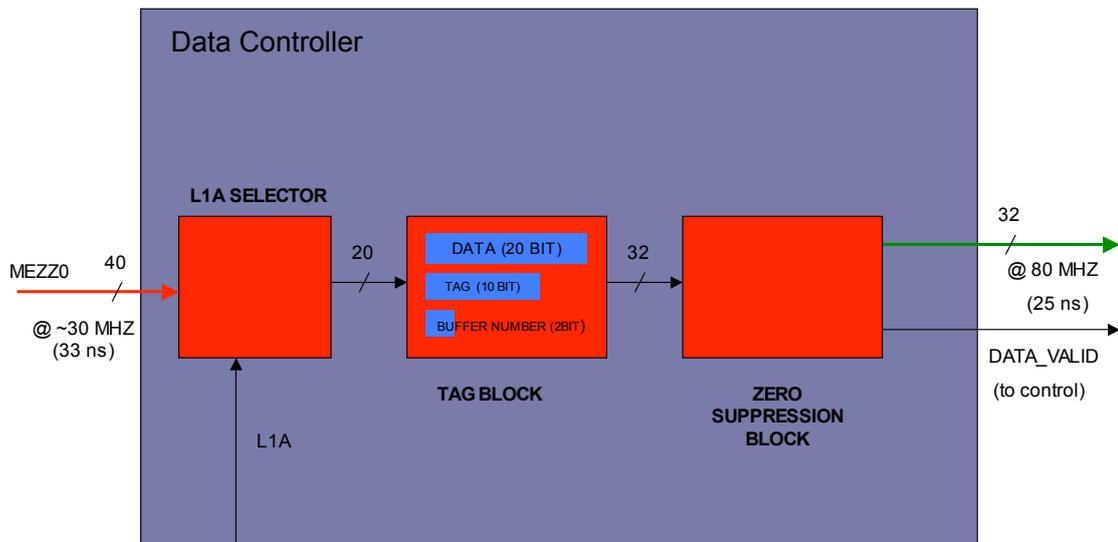


Figure 8: Basic logic components of the DataIO FPGA are put in evidence.

### 3 Mezzanine Specification

The mezzanine card has to perform the following tasks:

- Receives 160 LVDS data @ CDF clock frequency (132 ns) and converts them to TTL signals. The input signals are divided into 4 LVDS cables. Each LVDS cable allows to transfer 2 tower energy information per word. At each CDF clock cycle we receive the information for 8 towers.
- Sends tower information to Pulsar adding 2 bits indicating the data arriving cable.

The data lines available on a single Mezzanine-Pulsar connector is about 64, for a total of 128 pins of which only 79 are available for signals. So the mezzanine can send only 2 tower information at once to the Pulsar (40 bits). Because the mezzanine receives 8 tower information @ CDF clock frequency, we decide to send 2 tower information to Pulsar @ 4xCDF clock frequency.

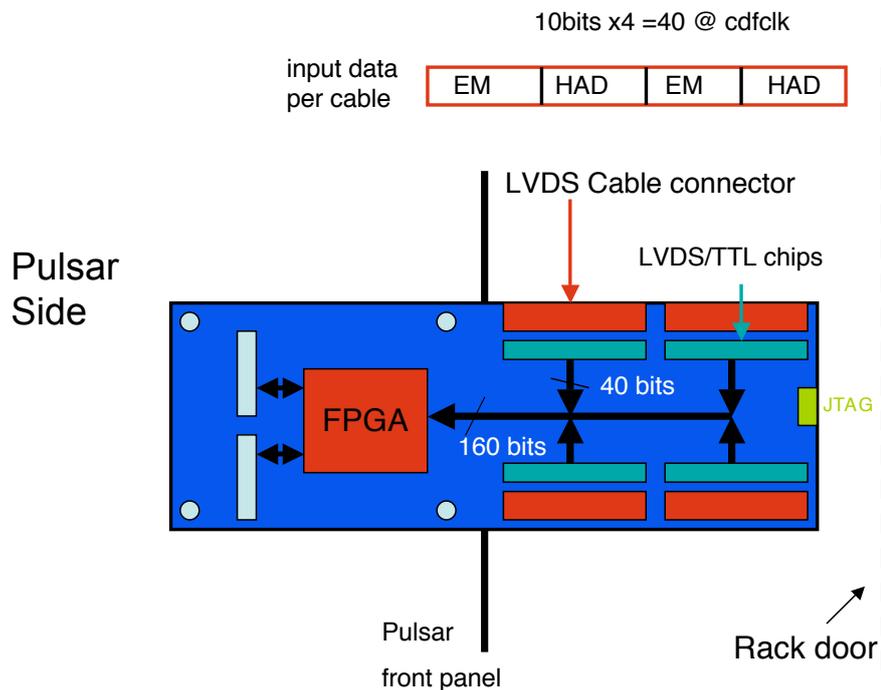


Figure 9: A top view of the mezzanine card.

For the upgrade we need a set of LVDS receiver mezzanine cards ( $18 \times 4 + \text{spares}$ , we call them RX mezzanines), but for testing purpose we also need a set of LVDS transmitter mezzanine cards ( $4 + \text{spares}$ , we call them TX mezzanines). Probably, for

space constraints, the best strategy will be the production of two separate boards, very closely related, one executing the RX function and the other the TX function. The main items concerning with the two boards are the following:

- For the RX board a set of LVDS/TTL converter chips is required (RX chip), while, for the TX board, we need a set of TTL/LVDS converter chips (TX chip). The RX and TX chips have the same sizes and very close pinout, but unfortunately they are not exactly pin-compatible. The difference between the TX and RX mezzanine will be really small.
- For both TX and RX board all the data/control lines are mono directional. In particular the signals changing direction from one board to another are the 160 lines to/from LVDS connectors and the 40 data from/to Pulsar. In addition we probably have few control data lines from/to Pulsar.
- The firmware inside the FPGA should be changed according with the board function. We can also think about the possibility to select the firmware according with a VME register.
- Terminations will be designed on the RX mezzanine near the RX chips, but they cannot be activated until we use the LVDS multidrop function. We are planning to place the footprint on the board with a leg already connected and the other to be connected later, so that the resistors can be soldered when the mezzanine is assembled, even if activated only at the end of the commissioning.
- The lines routed between the LVDS connector and the RX chips cannot be longer than 1.5 cm to be sure the LVDS multidrop function will work correctly.

## Basic requirements and components

In the picture you can see a first attempt of the top-level diagram of the mezzanine.

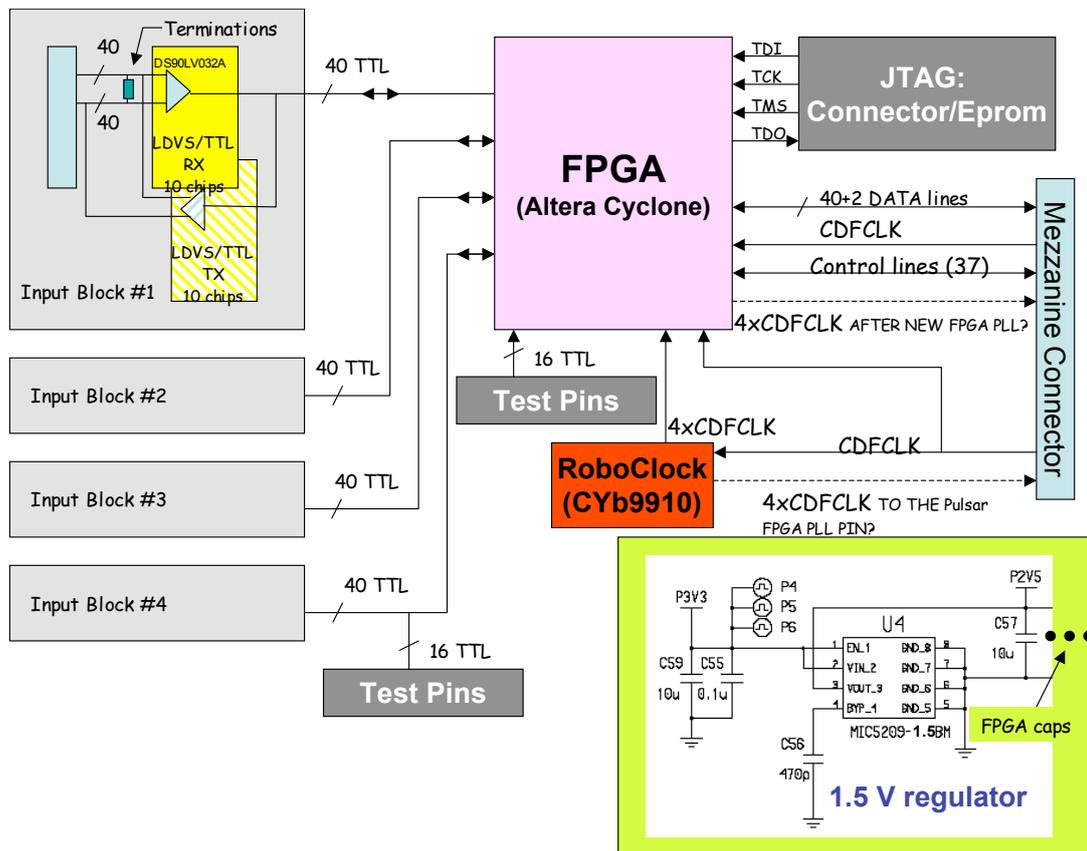


Figure 10: Top-level diagram of the Mezzanine. Both transmitter and receiver features are shown, in particular bidirectional lines are lines changing direction with the RX/TX functions.

4 logical input blocks receive the input data from the LVDS cables at CDFCLOCK frequency. The picture takes into account both the RX and TX mezzanine. In the RX mezzanine the 4 input blocks convert the LVDS signals into TTL signals and send them to the FPGA. The FPGA just merges the 160 bit input data into one 40 bit output data. The output will be sent to the Pulsar at 4x CDFCLOCK frequency. The FPGA also adds 2 bits indicating the data arriving cable. A set of additional control lines is foreseen (of the 37 available lines we need about 10 for the moment). All lines that change direction according with RX/TX function are indicated as bidirectional in the picture. More details are reported in the following section:

- 4 LVDS cable connectors receive the inputs.  
They are 4 **80-pin Honda connectors**. Each connector receives 40 LVDS signals @ CDF\_clk frequency (132 ns).

- 4 input blocks for LVDS to TTL translation.  
Each one includes 10**DS90LV032A** chips In the picture we report both the RX and TX chips. The power supply for them is 3.3 V.
- Power Supply : 3.3 V
- 1 **Cyclone FPGA** 3.3 V for the I/O.
- In order to generate the lower voltage for the core of the FPGA (1.5 V).  
We can use 1.5V regulators (**MIC5209-1.5BM**)
- 1 Eprom to program FPGA
- JTAG Interface
- 2 set of Test pins: for Input Data and for signals to/from mezzanine connector.
- 1 Roboclock to multiply the CDF clock frequency(**CYb9910**). The new clock should be also sent to the Cyclone FPGA, where two internal PLLs can be used to adjust both the CDFclk and  $4xCDFclk$ .
- 2 Mezzanine-Pulsar Connectors. We have available 64X2 pins and about 79 can be used as data/controls signals.  
42 bits have to be used as data (40)control(2) signals from Mezzanine FPGA to IO PULSAR FPGA. Few other of them (less than 10) could be probably used as additional control signals between Mezzanine FPGA and Pulsar. So we can have about 30 signals available and free. We plan to connect all of them to the pins of the Mezzanine FPGA for future and possible uses.

## Interface

In the following you can find a preliminary list of the RX mezzanine Inputs/Outputs. See also the picture Please note that input (output) signals that will be output (input) in transmitter mode are indicated.

## INPUT / OUTPUT DIAGRAM OF MEZZANINE

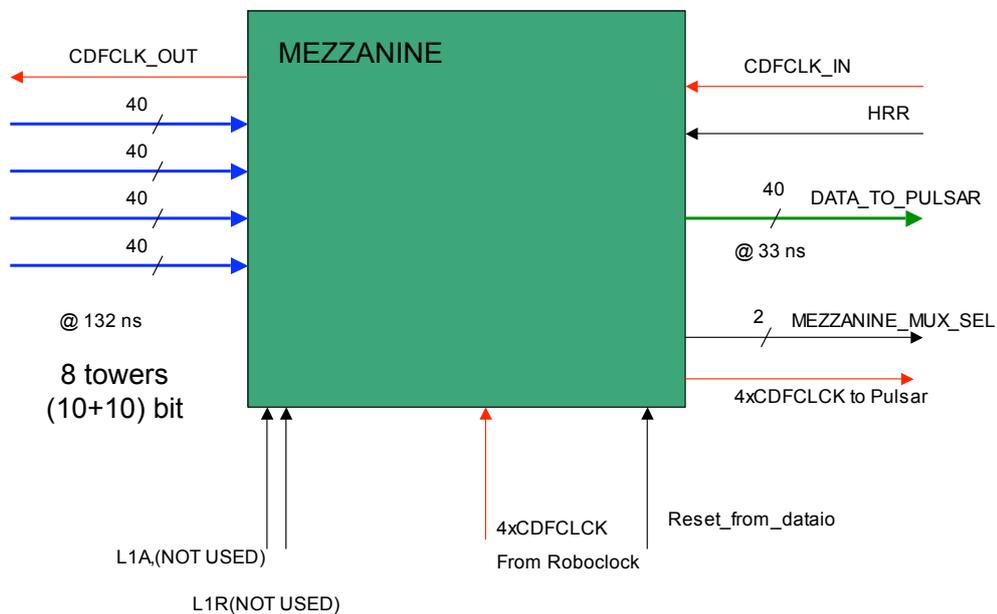


Figure 11: RX Mezzanine Inputs/Outputs.

### Input:

- 40X4 LVDS signals from the cables. (Output in transmitter)
- 4XCFD clk from Roboclock
- CDFCLK\_IN CDF clock, for the input registers. It is better to connect it so that, if necessary, can be adjusted using one of the two PLLs inside the FPGA
- RESET from Pulsar
- L1A\* from Pulsar (NOT USED, just for future possible uses)
- L1R\* from Pulsar (NOT USED, just for future possible uses)
- Signal to issue the HRR operation.

- Reset signal from Pulsar

**Output:**

- 40 TTLs signals to Pulsar (Input in Transmitter)
- 2 Controls Signal: cable connector index
- 4XCDK clk signal. This signal could be sent to Pulsar directly by the Roboclock, or, if necessary, after has been adjusted by the PLL inside the FPGA.
- CDFCLK\_OUT Output of the PLL. CDFCLK\_IN.

## FPGA Requirements

The firmware will be very simple. Altera Cyclone is suggested (EP1CA).

- 250 pins available as data/control/clock pins: 160 TTL (Cables Inputs) plus 40 TTL Output (to Pulsar) plus 39 control/clock signals to be connected to Pulsar. The 160 TTL signals will be Output in transmitter while the 40 TTL Output to Pulsar will be Input from Pulsar.
- Double power supply: 3.3 V for the and 1.5 for the core.
- PQFP Package.
- PLL inside in order to adjust the CDFclk and the 4 x CDF clock, if necessary.

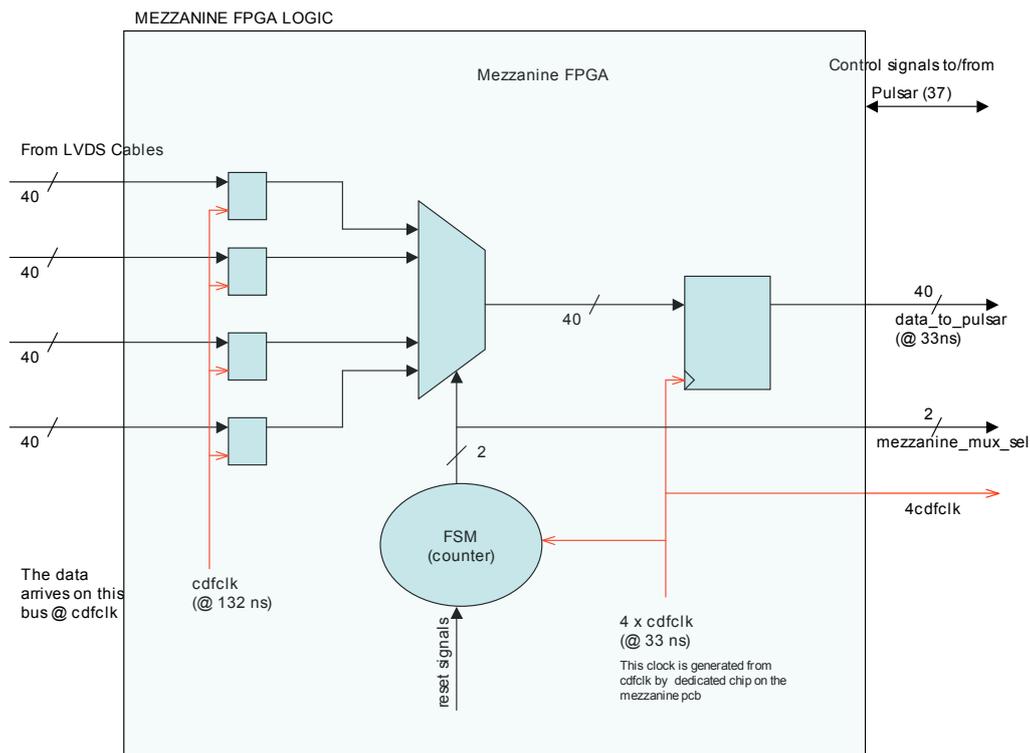


Figure 12: Mezzanine FPGA Logic. It includes 4 input registers and an output register with a MUX in the middle to merge the 4 inputs in a single output. The input registers work at the CFD clock frequency and the output register with a 33 ns clock (4 x CDF clock).

**Test point:**

This is a set of Test Points suggested. Please note we are considering a Receiver Mezzanine. Probably a similar set of test points could be planed also for the Transmitter Mezzanine.

- Vias for each LVDS Cable Inputs (160 signals).
- 1 set of test points for one particular input cable.
- 1 set of test points for FPGA logic just for testing purpouse.It could be placed very closed to the mezzanine connector.
- Vias on output signals (40 FPGA output to Pulsar plus few control signals between Pulsar and FPGA).

## Dummy Mezzanine

In order to think through all the details of the layout we plan to produce a dummy mezzanine. The dummy card design will include:

- 4 connectors
- LVDS/TTL chips
- 1 set of test points
- 1 JTAG connector

We plan to make 4 dummy PCBs and load them with connectors and chips, try them out on a Pulsar board (fully load the 4 mezzanines), connect them with 16 dummy cables and try it out in a Pulsar crate in the system making sure we can close the rack door without any problems.

In order to allow the cable plug into connector without hitting the PCB surface we can exploit the following options:

- 1 Raise connector a little bit when soldering. This option requires a careful test of the connections.
- 2 Have a hole on the PCB, this means that the bottom connector will move up a bit so that there are room for LVDS chips on the lower side. This option requires enough space on the Mezzanine.
- 3 Two cable will be turned by 180 degree.

At the site 13 you can see the most promising design of the mezzanine layout. The design shows the pulsar with 4 mezzanines with real sizes, both for connectors, cables and RX chips.

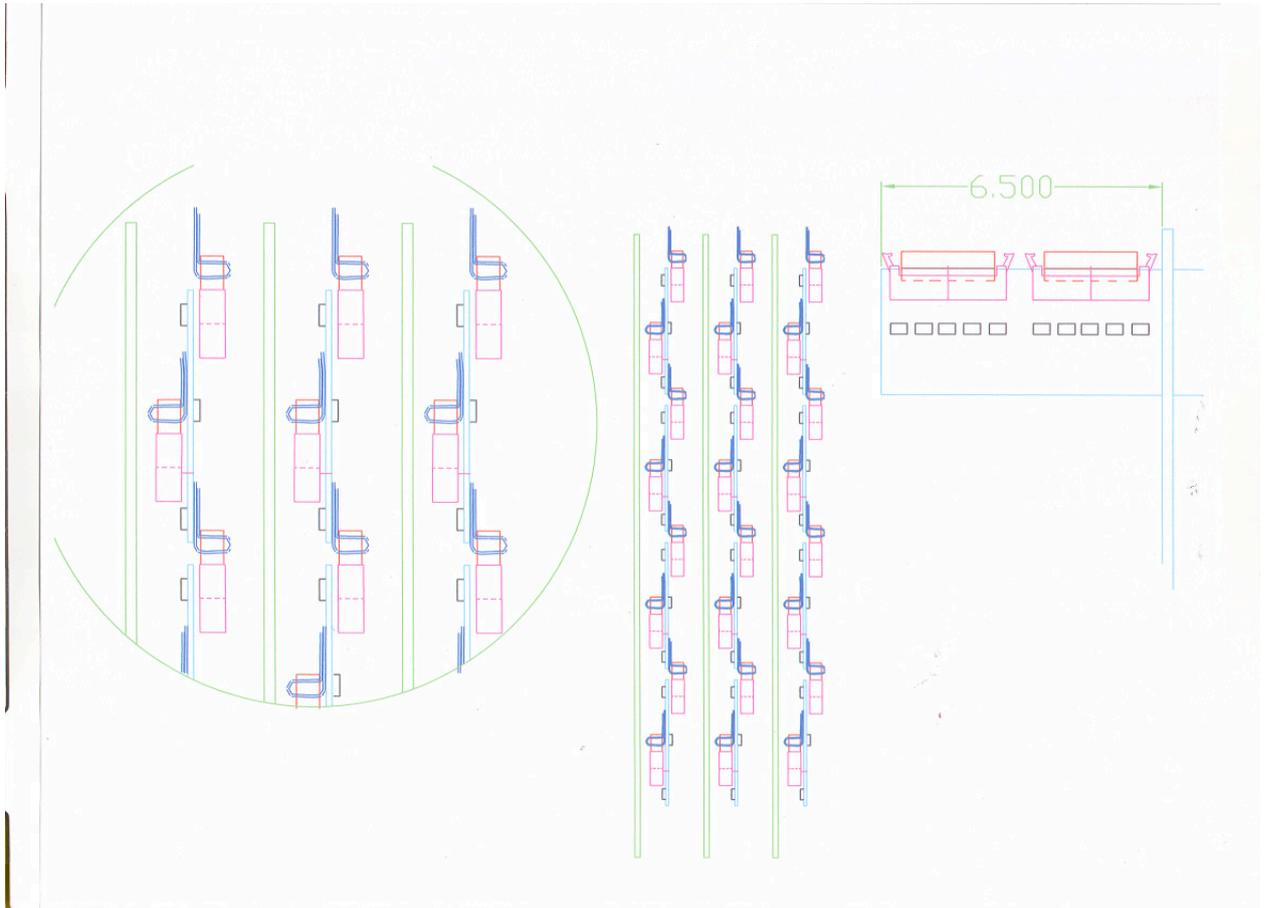


Figure 13: Mezzanine Layout and cabling strategy.

## 4 DATAIO Specification

### INPUT / OUTPUT DIAGRAM OF DATAIO

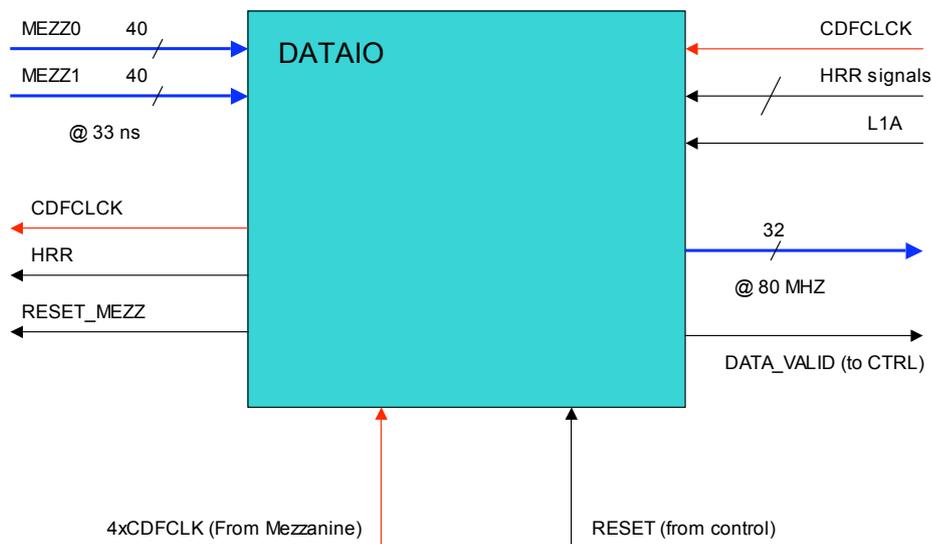


Figure 14: DATAIO FPFA Inputs-Outputs.

Each IO FPGA receives two mezzanine card outputs and stores the information on a DUAL PORT RAM, waiting for the L1A (see fig 15) . Data are stored on the RAM continuously, and a writing address pointer takes trace of the last written data. Because the latency between the L1A and the data is well known and can be fixed in terms of clock cycles, as soon as a L1A/L1R arrives (about  $5.5\mu s$  after the data) we can reconstruct the reading pointer address starting from the actual writing pointer address. The latency can be simply adjusted downloading the right number of cycles in a VME writing register. If a L1A is received the data is sent to a FIFO to be synchronized with the Pulsar clock (80 MHz). For each L1A signal we receive also a L2Buffer information that must identify the event. 20 bits information, corresponding to a single tower, are then flagged combining board index and FPGA index. The previous 2 bits flag coming from the mezzanine complete the stamp for the tower.

A new 32 bits information is now available for each tower at the Pulsar frequency:

- 20 bits energy
- 10 bits for the tower index (576 towers)
- 2 bits for the L2buffer

A zero suppression block enables only those towers with non-zero energy information. A last block merges the two mezzanine data and sends them to the CTRL at the Pulsar frequency.

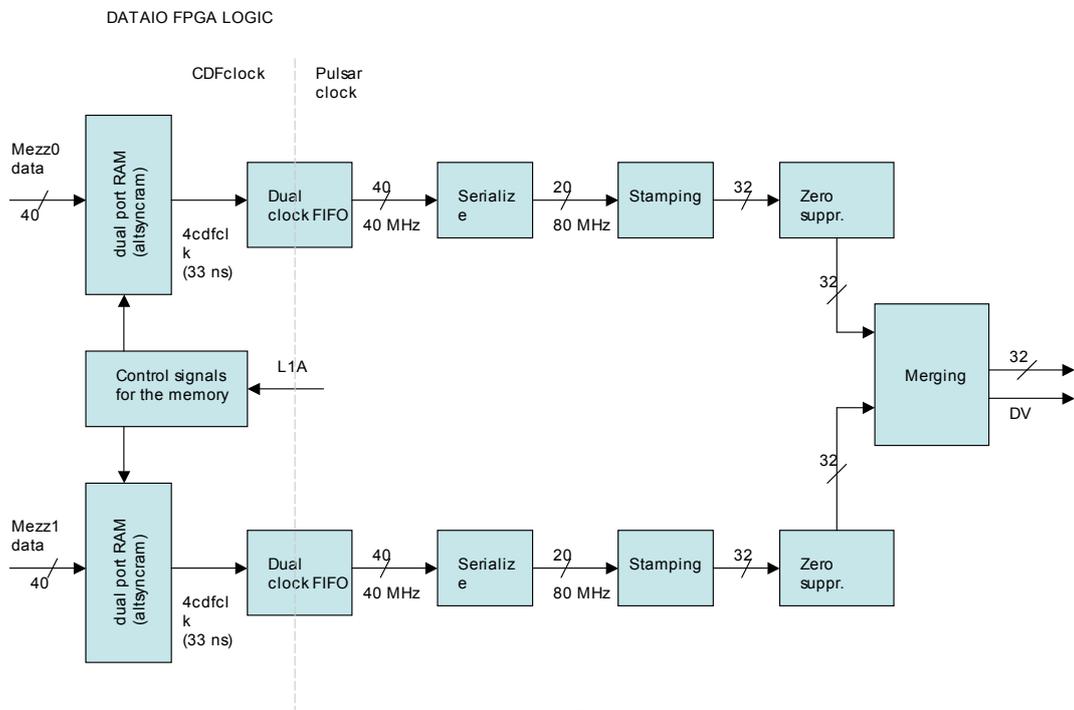


Figure 15: DATAIO FPGA logic.

## DAQ Buffer

We have to save diagnostic information in the same structure created for the Pulsar based Global Level-2 trigger decision crate. The TP2D bank contains this type of information. The DAQ buffer layout is identical for all the Pulsar board. Each Pulsar board contains 6 unique DAQ buffer, which can be redout independently. Each DAQ buffer could either contain input data (data going into Pulsar from upstream, in our case DCAS bank) or output data (data going out of the Pulsar board after internal processing). For each Pulsar board in the redout chain, there are 6 card pointers created in the block, which corresponds to the 6 DAQ buffers. The redout sequence of the pulsar boards in a given slot follows the slot number (start from the lowest slot number).

### Pulsar Inputs-Outputs

For each Pulsar the **Input Signals** are the following:

- 640 – *bit* LVDS signals, organized into 16 cables. Each LVDS cable carries 2 tower information. Each tower correspond to 10 bit Em Energy plus 10 bit had Energy information.

The **Output Signals** of the Pulsar is 1 S-Link cable. The format of the S-Link packet foreseen one Header Word, plus a number of Data Words (the number of data word depends on how many towers have non-zero energy). According with the actual luminosities (see distributions in the proposal Note) a reasonable number of words could be around 300 – 350, followed by a Trailing Word.

- Header word
- Data word: 20 bits Energy plus 10 bits for the tower address plus 2 bits for the Buffer Number.
- Trailing word

The IO FPGA prepares the Data Words, while the CTRL FPGA adds the Header and Trailing words.

The following are **Control Signals** in J2 backplane connector we probably should take care:

- L1A\*: Level 1 Accept
- L1R\*: Level 1 Reject
- L2B0\*: Level 1 Accept Buffer Address 0
- L2B1\*: Level 1 Accept Buffer Address 1 (L2B0, L2B1 indicate the address of the L2 Buffers and they are valid on the L1Accept signal)
- CDF\_GLIVE : Level 1 Accept corresponds to live Beam crossing
- BC\* : Beam crossing. It is defined as being LOW on the leading edge of CLK pulse for which there is beam.
- B0\* : Bunch 0
- Abort : Abort Gap
- Test\*
- RUN\* : Clear halt Condition
- STOP : Set Halt Condition
- HALT\* : Halt filling L1 FIFO/pipelines, is set by STOP\* and cleared by RUN\*
- L2A\* : L2A Accept
- CLK : CDFCLOCK (132 ns)
- CLK\*
- RECOVER\* : Reset FIFOs/Buffer
- ERROR\* : Error on card

- L2BD0\*
- L2BD1\*

The following are the remaining signal in J2 backplane:

- CDF\_L1\_Calib: set to indicate calibration event
- Calib signals
- EVD03

## 5 Testing

### Stand-Alone

We plan a Pulsar Stand-alone test setup. We plan to use two additional dedicated Pulsars: Transmitter and Receiver. We load data directly on a ROM inside the transmitter. Via a VME transaction we write a configuration register (RAM) that sets:

- Number of data to send
- Sequence of the L1A to send ( for example : every data confirmed by a L1A, or L1R)
- Delay In sending data
- Send Data Command

A simple State Machine on the Transmitter sends data to the Pulsar under Test. Data arriving on the Receiver are stored in a RAM and read by a VME transaction.

Same data loaded on the ROM will be simulated on the PC and compared to the data reading data.

### STAND ALONE PULSAR TEST

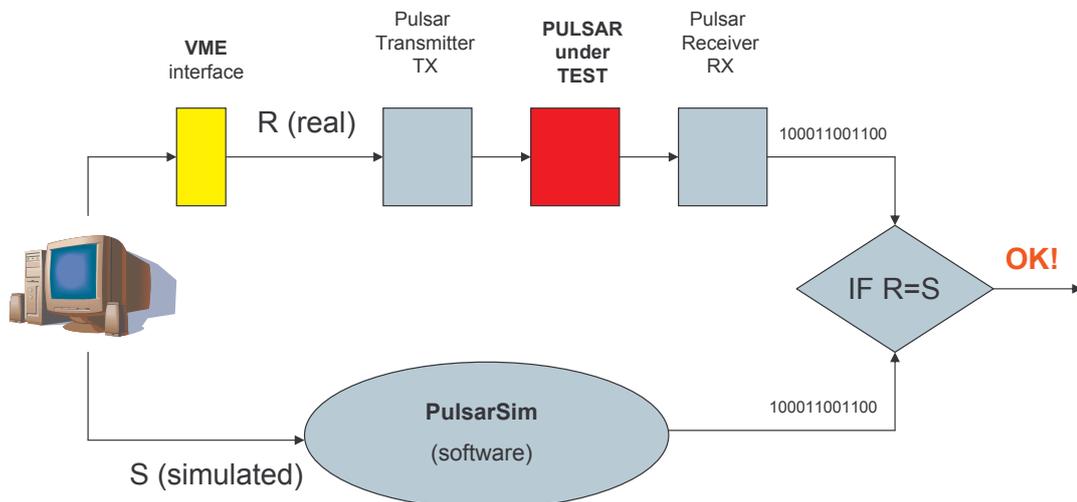


Figure 16: Scheme of the standalone pulsar test.

The transmitter have to perform the following sequence of operations:

- Send different data for each mezzanine (160 bits). one possible implementation could be having 4 ROM, one for each mezzanine transmitters, containing different data. Data could be sent in parallel to the connectors as soon as a Start Command is received.
- Data must be send  $10ns$  before the raising edge of the cdfclock, at the maximum speed of one data each 3 cdf clock cycle (7.5 MhZ). Data must be stable only for only one cfd clock cycle.