ATLAS Level-1 Calorimeter Trigger Update

HUB Firmware Specification

**Dan Edmunds,Yuri Ermoline, Brian Ferguson
Wade Fisher, Philippe Laurens, Spencer Lee
Pawel Plucinski**

Draft for Review

Version: 1.2

18 March 2018

Table of Contents

1. FEX HUB overview

The FEX-Hub module is an integral part of the L1Calo system. Its primary functions are to support FEX system readout, provide switching functionality for module control and DCS IPbus networks and to distribute timing and control signals to the FEX modules. There are to be two Hub modules per shelf. Both Hub modules will receive multi-gigabit FEX data over the ATCA Fabric Interface, which will be fanned out to a ROD mezzanine on the Hub and to the Hub's own FPGA. This high-speed data path will include two data channels from the other Hub module. The Hub module in logical slot 1 will provide switching capability for a network that routes module control signals on the base interface, while the Hub in logical slot 2 will provide switching for a network that routes DCS information. The Hub module in slot 1 will further receive TTC information from the FELIX system, and these signals will be decoded and fanned out to the FEX modules, ROD modules and also to the Hub in slot 2. The fanned-out TTC control data stream will be interleaved with ROD-to-FEX communications including, for example, back-pressure signals. The Hub module has connections to the other slots in the ATCA shelf over three distinct electrical interfaces. ATCA backplane Zone-2 consists of the Fabric Interface and the Base Interface. The Fabric Interface provides 8 differential pairs (channels) from each node slot to each Hub slot (8 to Hub-1 and 8 to Hub-2). There are a total of 8 Fabric Interface channels between Hub-1 and Hub-2 (not 16 total). The Fabric Interface pairs have a nominal bandwidth specification of 10 Gbps / channel. The Base Interface provides 4 differential pairs between each node slot and each Hub slot. There are a total of 4 Base Interface channels between Hub-1 and Hub-2. The Base Interface lines have a nominal bandwidth specification of 500 Mbps / channel, suitable for Gbps Ethernet protocol. Finally, ATCA backplane Zone-1 provides each node and Hub slot with a connection to the Intelligent Platform Management Bus (IPMB) with a total bandwidth of 100 kbps. The Hub module will provide MPO connectors in the ATCA Zone-3 region, which will allow for the routing of fiber-optic cables to/from the MiniPODs on the Hub and ROD modules. The L1Calo FEX-Hub system will consist of eight modules. There will be two eFEX shelves, one jFEX shelf and one L1Topo shelf, each hosting two Hub modules.

1.1 HUB Functionality

1.1.1 Support of the ROD Mezzanine Card

The FEX Hub physically holds the ROD Mezzanine Card and provides electrical connections to it through two 400 pin Meg-Array connectors.

1.1.2 FEX and FEX-Hub Readout Data Distribution

The FEX-Hub receives over the Fabric Interface 6 serial streams of Readout Data from each FEX Module. Each FEX-Hub also receives over the Fabric Interface 2 serial streams of Readout Data from the other FEX-Hub in the crate. These 74 high speed serial streams are fanned out on the FEX-Hub. One copy of each stream is sent to the ROD and one copy is sent to the Hub's own UltraScale FPGA. The Hub FPGA also sends 2 serial streams with its own Readout Data to its own ROD. Each ROD thus receives a total of 76 high speed Readout Data streams: 6 streams from each FEX, 2 streams from the local Hub FPGA and 2 streams from the other Hub's Hub FPGA. The data rate per readout stream will be 10 Gbps or less.

1.1.3 TTC Clock and Data Stream Distribution

The FEX-Hub in Slot 1 uses a 12-channel MiniPOD optical receiver to receive TTC signals from the upstream FELIX system. The FEX-Hub receives two types of TTC signals: a copy of the LHC clock and TTC control data. These signals need to be fanned out to each FEX module, to the local ROD, to the local Hub FPGA and to the FEX-Hub in Slot 2 (including its ROD). The LHC clock is directly forwarded without any processing on the FEX-Hub. The TTC control data will be merged with additional control information coming from the ROD module from each FEX-Hub before being fanned out. The FEX-Hub uses two ports from the Fabric Interface Channel to each Node Slot to fanout these two signals to each FEX. These two TTC and control signals sent to the FEX plus the 6 Readout Data streams received from each FEX use all 8 signals pairs of each Fabric Channel connecting one FEX to the FEX-Hub, albeit with an unconventional port direction usage. The FEX-Hub in Slot 2 does receive the TTC information from FELIX directly, but receives the TTC Clock and the TTC/ROD readout control stream from the FEX-Hub in Slot 1. The FEX-Hub in Slot 2 sends any required ROD readout control data generated by its own ROD to the FEX-Hub in Slot 1 for inclusion in the combined TCC/ROD readout control data stream.

1.1.4 Ethernet Network Switch

The FEX-Hub hosts an un-managed 10/100/1000 Base-T switch to provide the following 19 Gigabit Ethernet connections [see Figure 1]:

- 1 connection on the front panel for the "up-link"
- 12 connections to the "FEX Node" modules in this crate
  via the Base Channel Fabric
- 1 connection to the ROD on this Hub (or IPMC on the other Hub) via the front panel
- 1 connection to the ROD on the other Hub (or IPMC on this Hub) via the front panel
- 1 connection to the other Hub's UltraScale FPGA via the Update Channel Interface
- 1 spare front panel connection

1.1.5 Slow Control (IPbus to HUB and ROD)

HUB: An IPbus interface provides high-level, functional control of the FEX-Hub module. This allows, for example, setting any firmware parameters, controlling modes of operation and reading monitoring data. Figure below shows the Hub's Base Interface Ethernet Switch in the context of the other cards in the ATCA shelf. ROD: An Ethernet link is provided from the main ROD FPGA to the Ethernet switch on the Hub. This will allow a computer using IPbus to:

- Access registers within the ROD FPGA, setting parameters and controlling modes of operation.
- Store FPGA configurations into the SPI-Flash Configuration Memory.
- Initiate the loading of configurations from the SPI-Flash.

This can be used to load a configuration from one of a number of other SPI-Flash sectors. These sectors can be written via IPbus.
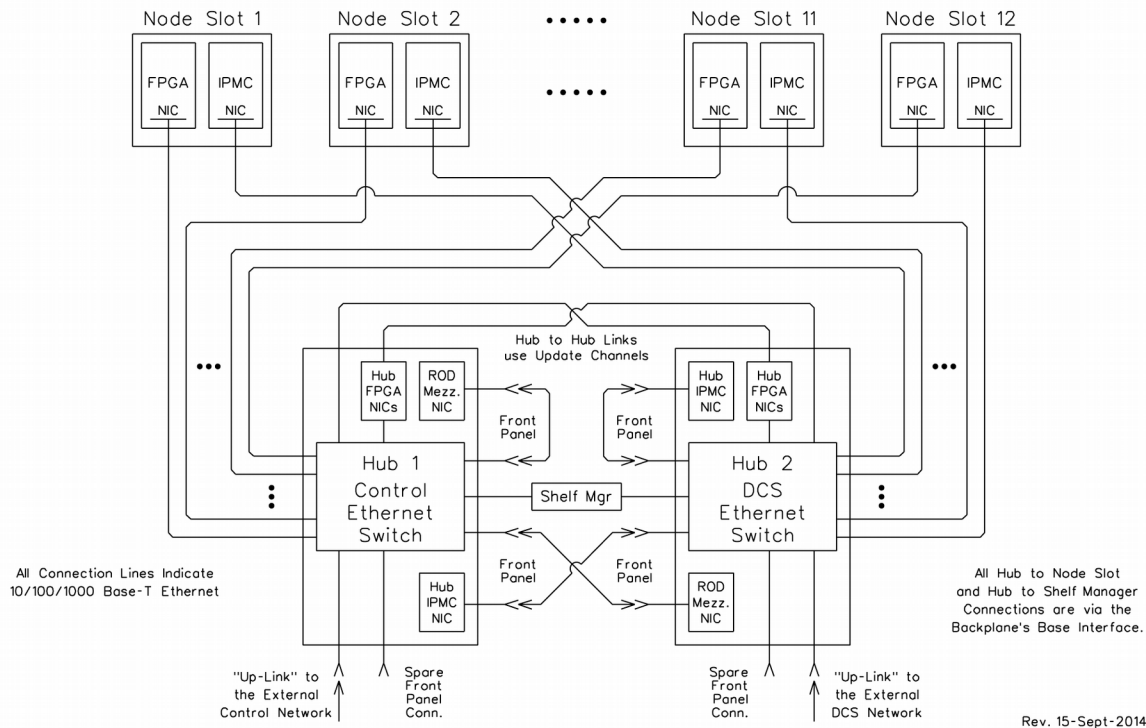
Figure 1: Illustration of FEX-Hub Ethernet network connections.

The Hub Module requires two physical chips for the Ethernet Base Interface connections to its FPGA (Micrel KSZ9031RNX http://ww1.microchip.com/downloads/en/DeviceDoc/00002117B.pdf – it can operate with a 1.8V RGMII port and thus directly connect to the Virtex7 HP I/O pins). Two FPGA MACs are connected to the physical chips via RGMII ports. This chip has both the RGMII signal connection to the FPGA that is used to move the actual Ethernet data and provides access to internal registers and also has a 2 wire serial "Management Data" port.
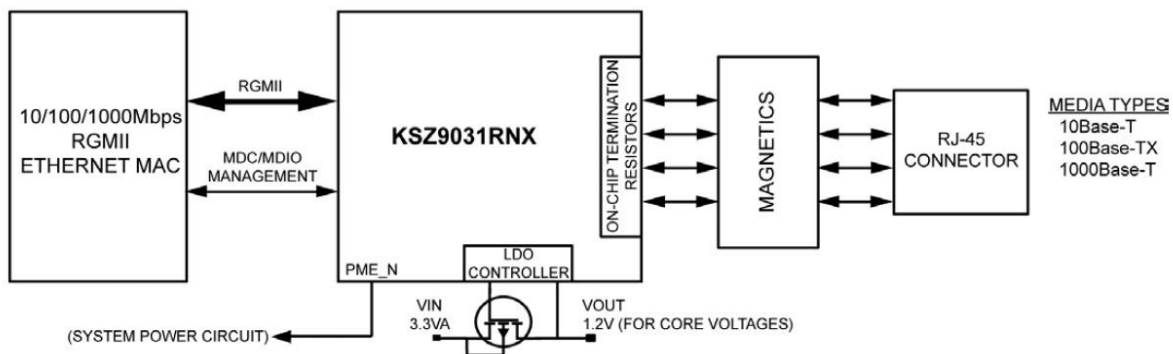
Figure 2: Illustration of Xilinx ETHERNET MAC IP and PHYS chip (KSZ9031RNX).

Figure 3: Schematic view of U21 Phys chip: Power, Clock, Reset.

Figure 4: Scheme of Phys Chip: RGMII, MDC/DMIO, and Base-T Circuits.

Figure 5: Board Reset Distribution, ROD Power Control.

For each of its two Ethernet Phys Chips (PHY) the Hub's FPGA will need to instance a MAC that supports an RGMII connection (along with MDIO/MDC lines) to the PHY. All of these signals to/from the PHY are currently routed through the 1V8 HP Select I/O Bank 68.

After power-up the KSZ9031RNX is configured to RGMII mode if the MODE [3:0] strap-in pins are set to one of the RGMII mode capability options.

There is no reset signal to the KSZ9031RNX from FPGA. An ad hoc manual push button was attached to the PHY chip on the HUB for debugging purposes.

The KSZ9031RNX RGMII port connects to HP I/O pins on the FPGA. The RGMII port consists of 12 signals:

- Transmit Clock to the KSZ9031RNX
- Transmit Control (enable) to the KSZ9031RNX
- Transmit Data 0:3 to the KSZ9031RNX
- Receive Clock from the KSZ9031RNX
- Receive Control (enable) from the KSZ9031RNX
- Receive Data 0:3 from the KSZ9031RNX

The KSZ9031RNX includes a MII Management port. This type of port is also called MDIO Management Data Input/Output. This port allows higher-level devices to monitor and control the KSZ9031RNX. This port allows direct access to the IEEE defined MIIM registers, and the vendor specific registers. This port also allows indirect access to the MMD address space and registers. This port consists of signals: MDC - the clock and MDIO - the data line.

The Hub Module has two KSZ9031RNX PHY chips. There are 14 jumpers associated with each of these PHY chips. These jumpers are resistors that bias a pin in one direction or the other and this value is read when the PHY chip first powers up or is reset.

The KSZ9031RNX has 9 pins (called "Strapping Options") that are read in this way at power up. Because of space limitations and because there is an obvious why that the Hub Module wants some of these Strapping Options set, 4 of them have only one jumper to pull that pin in the direction that is obviously needed for rational operation of the Hub Module.

The PHYADx jumpers set the address of the Management Interface Port on the KSZ9031RNX. The Management Port PHYAD bits 3 and 4 are internally always set to 0,0. Bits 2, 1 and 0 set to Low. Therefore, the PHYADx set to 0.

The Hub Module provides easy control of only the Mode_0 and Mode_1 lines. This provides the following 4 options for the Phys chip (Mode bits listed Mode_3, ..., Mode_0).

    1100 RGMII 1000 Base-T full duplex only
    1101 RGMII 1000 Base-T full or half duplex
    1110 RGMII 10/100/100 all but 1000 half duplex
    1111 RGMII 10/100/1000 full or half duplex

Mode: SET MODE {3..0} = 1100 - RGMII 1000 Base-T full duplex only

The Xilinx Tri-Mode Ethernet MAC core is a parameterizable core, in 1000 Mbps mode, the TEMAC core can also connect with industry standard PHY devices. Optional MDIO interface to managed objects in PHY layers (MII Management).
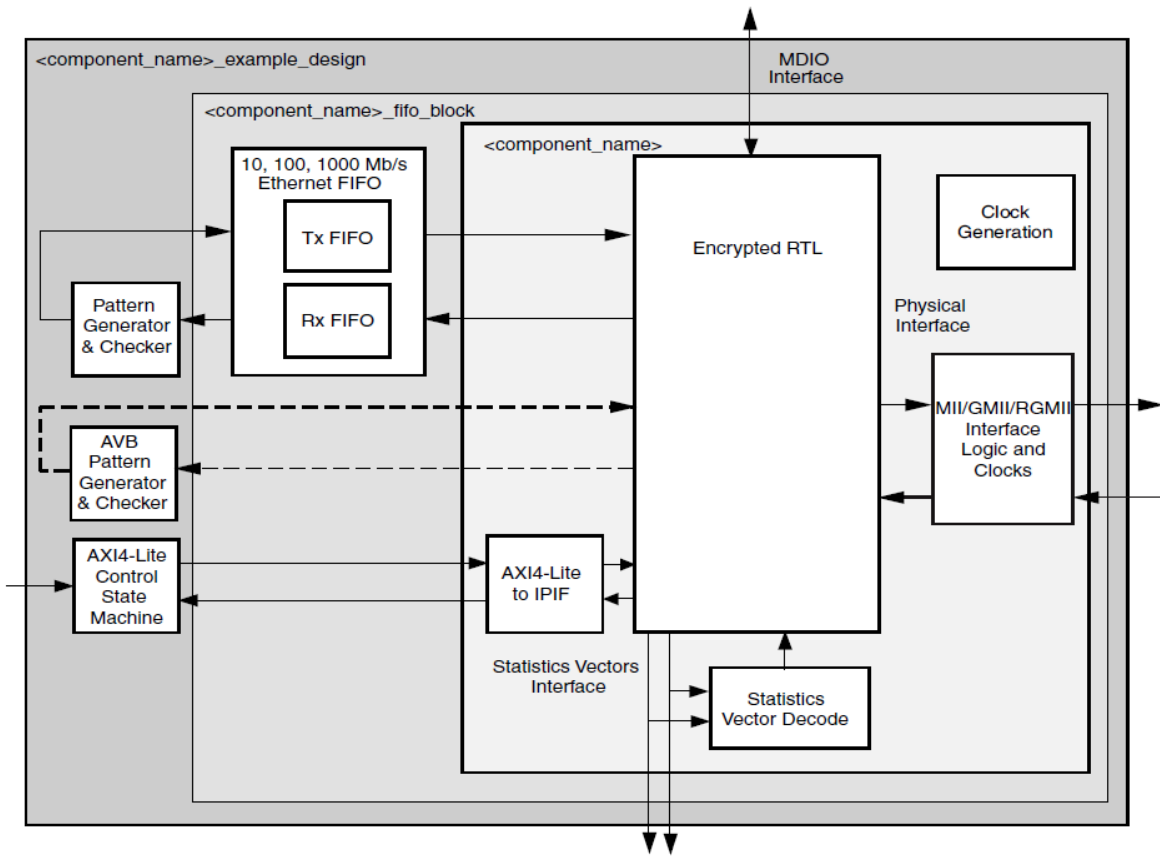
Figure 6: Scheme of Xilinx MAC Example HDL Design (UltraScale).

The following design approach is based on the suggestion by Ed Flaherty (University of Cambridge):
Step 1: Generate Xilinx MAC Example Design (UltraScale RGMII).
Step 2: Modify the Example Design to HUB board hardware.
Step 3: Generate Tx packets with built-in Simple Frame Generator and capture them in Wireshark.
Step 4: Packet generation logic replaced with simple read fifo (Rx looped back to Tx).
Step 5: Packets sent from server (Ostinato) returned via the loopback proving Rx and Tx paths.
Step 6: Use Wireshark to check returned packets

IPBus porting: 3 Steps
Step 2: Stripped down RGMII Example Design
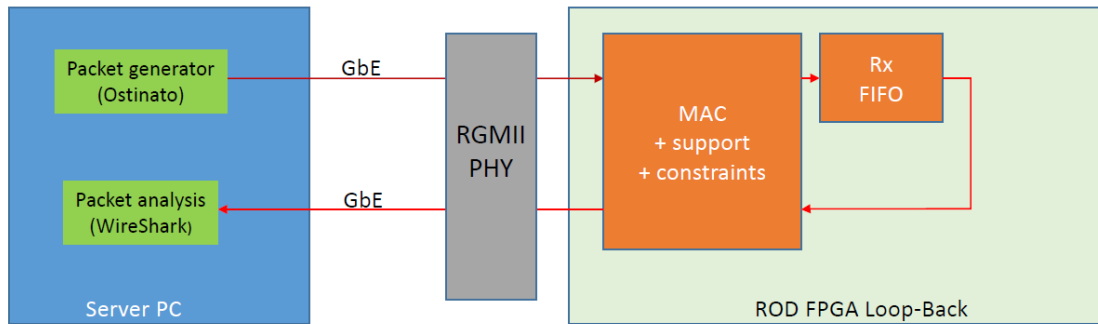
Stripped-down example design
- Packet generation logic replaced with simple read fifo (looped back)
- All of the example design MAC+Support logic and constraints retained
- Verification: Packets sent from server were returned via the loopback proving Rx and Tx paths

Ed Flaherty 7-June-2016

Figure 7: IPBus porting.

As soon as the MAC part works, the IPbus control part and the IPbus slaves can be added to the design, as described in the note: Notes on Firmware Implementation of an IPbus SoC Bus, V1.0   23/5/2012, DMN".
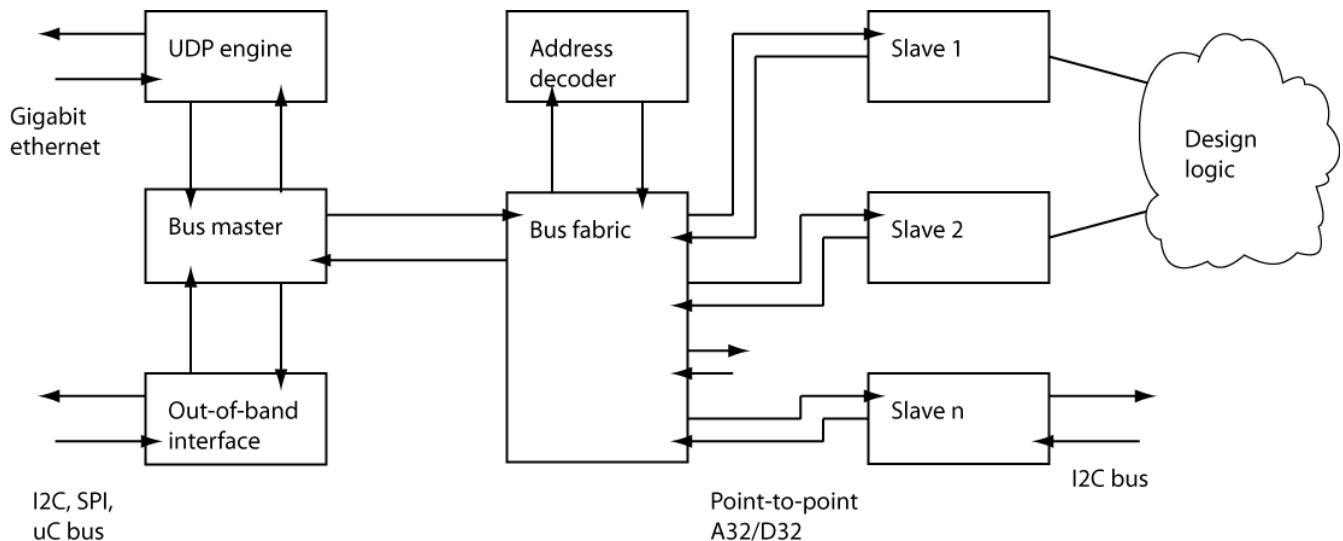
Figure 8: IPBus topology.

The SoC bus is fully synchronous, and operates from a single system clock. There is no constraint onthe relationship of the bus clock to the 125MHz GbE clock, as the firmware contains handshaking logic. For slaves, which do not require wait states, the 32b data path on the SoC bus allows full utilization of the Ethernet interface as long as it runs at >32MHz; this allows straightforward timing closurefor complex multi-slave designs. Designs therefore typically drive the Ipbus clock at ¼ of the GbE clock (i.e. 31.25MHz).

```
                              Hub FPGA Registers
                              ------------------
                                                      Initial Rev.  02-Fev-2017
All IPbus registers are 32-bit, all signals are active HIGH
-----------------------------------------------------------------------------------
Addr        Type   Name         Content        Bit   Comment
-----------------------------------------------------------------------------------
00000000  RO     hub_module   module_type    8     Set in FPGA FW
                               hw_revision    8     idem
                               fw_type        8     idem
                               fw_version     8     idem
-----------------------------------------------------------------------------------
00000001  RO     hub_address  shelf_adrs     8     Shelf Address from the IPMC
                               slot_adrs      8     Backplane Hardware Slot Address
                               adrs_to_rod    8     Overall Hardware Address to the ROD (generated)
                               spare          8     -
-----------------------------------------------------------------------------------
00000002  RO     hub_alerts   no_pll_lock    2     From PLL lock circuits
                               phy_int        2     From Eth Phy chips
                               mpod_int       2     From the MiniPODs
                               hub_smb_alert  1     From 7 DCDC Converters
                               hub_pwr_not_ok 1     From the Power Control circuits
                               no_rod         1     ROD is NOT present
                               rod_smb_alert  1     Power supply problem on the ROD
                               rod_status     3     no ROD power, ROD not config, spare
                               no_sw_loop_det 3     From the Broadcom Switch chips
                               spare          16    -
                                                    * at normal operation all bits = '0'
-----------------------------------------------------------------------------------
00000003  RW     hub_control  other_hub_Clk  1     FPGA internal, selects clock from other Hub
                               fex_clk_dis    1     Disable clock to FEXs
                               mgt_equ        13    Disable equalization in MGT Fanout chips
                               i2c_buf_dis    3     Disable Sensor I2C Bus translator/buffer chips
                               led_drv        3     Front palnel LEDs control
                               rod_pwr_en     1     ROD may turn ON its power supplies
                               sw_loop_det    3     To the Broadcom Switch chips
                               mpod_rst       2     TX and RX MiniPODs resets
                               spare          4     -
                                                    * after power ON all bits set to '0'
-----------------------------------------------------------------------------------

These are 4 "common" HUB registers (ReadOnly and Readwrire):

- RO hub_module:  general HUB information (Module ID, HW/FW versions)
- RO hub_address: external FPGA input pins for shelf and slot addresses, read internally generated address to ROD.
- RO hub_alerts:  all bits ='0' in normal operation, set to '1' by external signals
- RW hub_control: on power ON all bits ='0'; set and cleared via IPbus.
```

Table 1: The initial HUB FPGA register map.

2. Combined_TTC/DATA

The HUB FW is obliged to distribute TTC information throughout the shelf [2]. There are several Combined_TTC links within the shelf, one between each Fex slot (slots 3-14) and each Hub, also one between each Hub and ROD and 2 links between two HUBs.
The additional feature allow to transmit the Reset signal (Aurora Initialization, Figure 9) from the Readout_Ctrl link and distribute it to the appropriate shelf slot. In total, we have the following number of links:

- 12 links between each FEX slot and each HUB
- 1 link between the ROD and each HUB
- 2 links between two HUBs
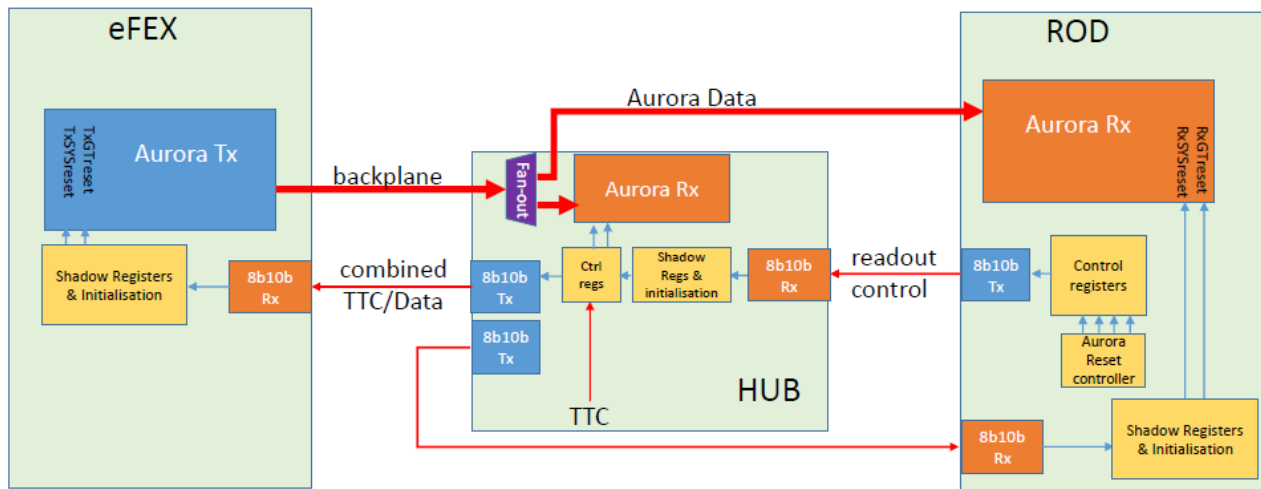
Figure 9. Example of link Reset Routing.

2.1 Combined_TTC/DATA - Physical Implementation

The Combined_TTC/Data link on the HUB FPGA is implemented with the use of several components, including the MGT transceivers (GTH and GTY), control and the diagnostic logic. The Combined_TTC/DATA link is designed to operate at 6.4 Gbps. The physical implementation of the Combined_TTC/DATA links assumes that there are 4 Control Registers on the Hub TX side, and Shadow Registers on the Rx side (Receiver: FEX, ROD and other HUB). The transmitter side generates the 128 bit message from 4 Control registers: Word_0, Word_1, Word_2, and Word_3. The transmitter side logic is in charge to write control information into these Control Registers. The contents of these registers are continuously transmitted to the modules (within the shelf) which receives the data into a duplicate set of 4 registers referred to as shadow registers. Anything written into a Control Register at the transmitter side will appear in the corresponding Shadow register at the receiving side within the following LHC clock. The least significant byte of Word_0 is reserved for the 8b10b Comma character K28.5.

2.2 Combined_TTC/DATA - Control Bit Definition

The bits within the Combined_TTC/DATA control words are primarily defined to provide TTC information as well as initialization functions for all of the Fex data links (Aurora in the case of eFex). The TTC information is sourced from a dedicated TTC interface on the Hub. The Reset information is received from the ROD via the Readout_Ctrl link.

13

Comma Character

Bits 7 to 0 of Word_0 contain the Comma character that maintains alignment of the 4 shadow registers with their corresponding Control registers. The chosen character is K28.5 = 0xBC.

Version

The 4-bit value contains the version number of this overall bit assignment. It will be held at "0000" through the initial debug phases, where many changes may occur.

Reset 3:0

These bits provide a system level reset/enable function. There are four per slot, and the functionality of these bits will be specified by the targets (eFex, jFex, etc)

Level-1 Accept (L1A)

L1 Accept is used to indicate when an event has been accepted by the Central Trigger Processor.

Bunch Counter Reset (BCR)

Entities within the shelf may keep local bunch counters. These are kept in sync using the BCR. The Bunch Counter is a local 12-bit counter that increments each LHC clock. It counts to a value of 3563 and then rolls over to 0. It is reset to a value of '0' when on BCR (Bunch Counter Reset)

Event Counter Reset (ECR)

Entities within the shelf may keep local event counters. These are kept in sync using the ECR. The Event counter is a local 24-bit counter that increments each L1A.   It is reset to '-1' on ECR.

Privileged Readout

This bit indicates that a full data readout has been requested. On each 40 MHz cycle, the delayed L1A, ECR, and BCR are read from the pipeline. If the L1A bit is set, the PRO FIFO is inspected. If the FIFO is not empty (this is the expected condition), the PRO 0 or 1 value is read and asserted on the backplane at the same time as the L1A, ECR, and BCR.

TTC Reserved

These bits are reserved for possible expansion of TTC information in Phase-II.

```
Other_Hub_RO   AL_0   MGT_FO_37   PF  ──→  Rx 3        1   Tx 3  ──→  ST   Combined Data to FEX 08
Other_Hub_RO   AL_1   MGT_FO_38   ST  ──→  Rx 2            Tx 2  ──→  ST   Combined Data to FEX 07
      FEX_9   AL_0   MGT_FO_39   ST  ──→  Rx 1   133      Tx 1  ──→  ST   N.C.
      FEX_9   AL_1   MGT_FO_40   ST  ──→  Rx 0            Tx 0  ──→  ST   N.C.


      FEX_7   AL_0   MGT_FO_25   PF  ──→  Rx 3   *    1   Tx 3  ──→  ST   Combined Data to FEX 06
      FEX_7   AL_1   MGT_FO_26   PF  ──→  Rx 2            Tx 2  ──→  ST   N.C.
      FEX_7   AL_2   MGT_FO_27   PF  ──→  Rx 1   132      Tx 1  ──→  ST   N.C.
      FEX_7   AL_3   MGT_FO_28   PF  ──→  Rx 0            Tx 0  ──→  ST   N.C.


      FEX_8   AL_4   MGT_FO_35   PF  ──→  Rx 3        1   Tx 3  ──→  ST   N.C.
      FEX_8   AL_5   MGT_FO_36   PF  ──→  Rx 2            Tx 2  ──→  ST   N.C.
      FEX_7   AL_4   MGT_FO_29   PF  ──→  Rx 1   131      Tx 1  ──→  ST   N.C.
      FEX_7   AL_5   MGT_FO_30   PF  ──→  Rx 0            Tx 0  ──→  ST   N.C.


      FEX_8   AL_0   MGT_FO_31   PF  ──→  Rx 3   *    1   Tx 3  ──→  ST   N.C.
      FEX_8   AL_1   MGT_FO_32   PF  ──→  Rx 2            Tx 2  ──→  ST   Combined Data to FEX 05
      FEX_5   AL_4   MGT_FO_17   PF  ──→  Rx 1   130      Tx 1  ──→  ST   N.C.
      FEX_5   AL_5   MGT_FO_18   PF  ──→  Rx 0            Tx 0  ──→  ST   Combined Data to FEX 04


      FEX_6   AL_0   MGT_FO_19   PF  ──→  Rx 3        1   Tx 3  ──→  ST   N.C.
      FEX_6   AL_1   MGT_FO_20   PF  ──→  Rx 2            Tx 2  ──→  ST   Combined Data to FEX 03
      FEX_6   AL_2   MGT_FO_21   PF  ──→  Rx 1   129      Tx 1  ──→  ST   N.C.
      FEX_6   AL_3   MGT_FO_22   PF  ──→  Rx 0            Tx 0  ──→  ST   Combined Data to Other Hub


      FEX_6   AL_4   MGT_FO_23   PF  ──→  Rx 3        0   Tx 3  ──→  ST   N.C.
      FEX_6   AL_5   MGT_FO_24   PF  ──→  Rx 2            Tx 2  ──→  ST   N.C.
      FEX_4   AL_2   MGT_FO_09   PF  ──→  Rx 1   128      Tx 1  ──→  ST   N.C.
      FEX_4   AL_3   MGT_FO_10   PF  ──→  Rx 0            Tx 0  ──→  ST   N.C.


      FEX_4   AL_4   MGT_FO_11   PF  ──→  Rx 3   *    0   Tx 3  ──→  ST   N.C.
      FEX_4   AL_5   MGT_FO_12   PF  ──→  Rx 2            Tx 2  ──→  ST   Hub Readout AL_0 to Other Hub
      FEX_5   AL_0   MGT_FO_13   PF  ──→  Rx 1   127      Tx 1  ──→  ST   N.C.
      FEX_5   AL_1   MGT_FO_14   PF  ──→  Rx 0            Tx 0  ──→  PF   Hub Readout AL_1 to Other Hub

                  C.L. ──────→                                  ←────── C.L.

      FEX_5   AL_2   MGT_FO_15   PF  ──→  Rx 3        0   Tx 3  ──→  ST   N.C.
      FEX_5   AL_3   MGT_FO_16   PF  ──→  Rx 2            Tx 2  ──→  ST   N.C.
      FEX_3   AL_0   MGT_FO_01   PF  ──→  Rx 1   126      Tx 1  ──→  ST   N.C.
      FEX_3   AL_1   MGT_FO_02   PF  ──→  Rx 0            Tx 0  ──→  ST   N.C.


      FEX_3   AL_2   MGT_FO_03   PF  ──→  Rx 3   *    0   Tx 3  ──→  ST   N.C.
      FEX_3   AL_3   MGT_FO_04   PF  ──→  Rx 2            Tx 2  ──→  ST   N.C.
      FEX_3   AL_4   MGT_FO_05   PF  ──→  Rx 1   125      Tx 1  ──→  ST   N.C.
      FEX_3   AL_5   MGT_FO_06   PF  ──→  Rx 0            Tx 0  ──→  ST   N.C.


      FEX_4   AL_0   MGT_FO_07   PF  ──→  Rx 3        0   Tx 3  ──→  ST   N.C.
      FEX_4   AL_1   MGT_FO_08   PF  ──→  Rx 2            Tx 2  ──→  ST   N.C.
Combined Data from Other Hub  PF  ──→  Rx 1   124      Tx 1  ──→  ST   N.C.
   Receiver MiniPOD Fiber 8   ST  ──→  Rx 0            Tx 0  ──→  ST   N.C.
```

```
  *  ─→  These Quads receive          ST  ─→  Straight Through
          the LHC locked              PF  ─→  Polarity Flip
          320.6296 MHz                AL  ─→  Aurora Lane Number
          reference clock.            0,1 ─→  UltraScale FPGA
                                              Super Logic Region
  MGT_FO_ ─→  MGT Data Fanout
              Channel Number          N.C. ─→  No Connection

Drw: 22
```

Figure 10. HUB MGT assignments (GTY Transceivers – QUADs 124:133)

```
FEX_10   AL_3   MGT_FO_48   ST  →   Rx 3        1   Tx 3   ⇒  PF   Combined Data to FEX 09
FEX_10   AL_2   MGT_FO_47   PF  →   Rx 2            Tx 2   ⇒  PF   Combined Data to FEX 10
FEX_10   AL_1   MGT_FO_46   PF  →   Rx 1      233   Tx 1   ⇒  ST   N.C.
FEX_10   AL_0   MGT_FO_45   PF  →   Rx 0            Tx 0   ⇒  ST   N.C.

FEX_9    AL_5   MGT_FO_44   PF  →   Rx 3   *    1   Tx 3   ⇒  PF   Combined Data to FEX 11
FEX_9    AL_4   MGT_FO_43   PF  →   Rx 2            Tx 2   ⇒  ST   N.C.
FEX_9    AL_3   MGT_FO_42   PF  →   Rx 1      232   Tx 1   ⇒  ST   N.C.
FEX_9    AL_2   MGT_FO_41   PF  →   Rx 0            Tx 0   ⇒  ST   N.C.

FEX_8    AL_3   MGT_FO_34   ST  →   Rx 3        1   Tx 3   ⇒  ST   N.C.
FEX_8    AL_2   MGT_FO_33   ST  →   Rx 2            Tx 2   ⇒  ST   N.C.
FEX_11   AL_5   MGT_FO_56   PF  →   Rx 1      231   Tx 1   ⇒  ST   N.C.
FEX_11   AL_4   MGT_FO_55   PF  →   Rx 0            Tx 0   ⇒  ST   N.C.

FEX_11   AL_3   MGT_FO_54   PF  →   Rx 3   *    1   Tx 3   ⇒  ST   N.C.
FEX_11   AL_2   MGT_FO_53   PF  →   Rx 2            Tx 2   ⇒  PF   Combined Data to FEX 12
FEX_11   AL_1   MGT_FO_52   PF  →   Rx 1      230   Tx 1   ⇒  ST   N.C.
FEX_11   AL_0   MGT_FO_51   PF  →   Rx 0            Tx 0   ⇒  PF   Combined Data to FEX 13

FEX_10   AL_5   MGT_FO_50   PF  →   Rx 3        1   Tx 3   ⇒  ST   N.C.
FEX_10   AL_4   MGT_FO_49   PF  →   Rx 2            Tx 2   ⇒  PF   Combined Data to FEX 14
FEX_13   AL_1   MGT_FO_64   PF  →   Rx 1      229   Tx 1   ⇒  ST   N.C.
FEX_13   AL_0   MGT_FO_63   PF  →   Rx 0            Tx 0   ⇒  ST   Hub Readout AL_0 to This ROD

FEX_12   AL_5   MGT_FO_62   PF  →   Rx 3        0   Tx 3   ⇒  ST   N.C.
FEX_12   AL_4   MGT_FO_61   PF  →   Rx 2            Tx 2   ⇒  ST   Hub Readout AL_1 to This ROD
FEX_12   AL_3   MGT_FO_60   PF  →   Rx 1      228   Tx 1   ⇒  ST   N.C.
FEX_12   AL_2   MGT_FO_59   PF  →   Rx 0            Tx 0   ⇒  PF   Combined Data to This ROD

FEX_12   AL_1   MGT_FO_58   PF  →   Rx 3   *    0   Tx 3   ⇒  ST   N.C.
FEX_12   AL_0   MGT_FO_57   PF  →   Rx 2            Tx 2   ⇒  PF   Transmitter MiniPOD Fiber 0
FEX_14   AL_5   MGT_FO_74   PF  →   Rx 1      227   Tx 1   ⇒  ST   N.C.
FEX_14   AL_4   MGT_FO_73   PF  →   Rx 0            Tx 0   ⇒  PF   Transmitter MiniPOD Fiber 1

         C.L. ——→                                            ←—— C.L.

FEX_14   AL_3   MGT_FO_72   ST  →   Rx 3        0   Tx 3   ⇒  ST   N.C.
FEX_14   AL_2   MGT_FO_71   PF  →   Rx 2            Tx 2   ⇒  PF   Transmitter MiniPOD Fiber 2
FEX_14   AL_1   MGT_FO_70   PF  →   Rx 1      226   Tx 1   ⇒  ST   N.C.
FEX_14   AL_0   MGT_FO_69   PF  →   Rx 0            Tx 0   ⇒  PF   Transmitter MiniPOD Fiber 4

FEX_13   AL_5   MGT_FO_68   PF  →   Rx 3   *    0   Tx 3   ⇒  ST   N.C.
FEX_13   AL_4   MGT_FO_67   PF  →   Rx 2            Tx 2   ⇒  PF   Transmitter MiniPOD Fiber 6
FEX_13   AL_3   MGT_FO_66   PF  →   Rx 1      225   Tx 1   ⇒  ST   N.C.
FEX_13   AL_2   MGT_FO_65   PF  →   Rx 0            Tx 0   ⇒  PF   Transmitter MiniPOD Fiber 8

Readout Control Data from This ROD  PF  →  Rx 3    0   Tx 3   ⇒  ST   N.C.
        Receiver MiniPOD Fiber 2    ST  →  Rx 2        Tx 2   ⇒  PF   Transmitter MiniPOD Fiber 10
        Receiver MiniPOD Fiber 4    ST  →  Rx 1   224  Tx 1   ⇒  ST   N.C.
        Receiver MiniPOD Fiber 6    ST  →  Rx 0        Tx 0   ⇒  PF   Transmitter MiniPOD Fiber 11
```

* → These Quads receive   ST → Straight Through
       the LHC locked   PF → Polarity Flip
       320.6296 MHz   AL → Aurora Lane Number
       reference clock.   0,1 → UltraScale FPGA
                                          Super Logic Region
MGT_FO_ → MGT Data Fanout   N.C. → No Connection
            Channel Number   C.L. → Center Line of BGA

Drw: 23
Rev. 31-Jan-2018

Figure 11. HUB MGT assignments (GTH Transceivers – QUADs 224:233)

L1ID

The L1ID should match the value in a local Event Counter. However, it is broadcast explicitly over this link to enhance channel reliability. If a CRC error is detected in a message, then all four words must be discarded. In this case, the L1A may be completely missed, thus putting the local entity out of sync with the rest of the system. The L1ID however, may be correctly read in the next message that doesn't contain a crc error.

ECRID

These bits are an extension of the L1ID. In the past, this has been generated on each board independently. Now the L1ID and ECRID bits are combined into a single 32-bit counter and broadcast from the Hub to all other boards.

Control Channel

These bits are reserved for a possible future implementation of a message channel

Link Reset

As the Receiver, the ROD must control all of the Data Link (Aurora) initialisation. The Link Reset signal is therefore derived from the Readout_Ctrl link. The Hub should connect to the appropriate Readout_Ctrl Link Reset for the slot to which the Combined_TTC is connected. Each endpoint also has a local reset timer to control Aurora link establishment. Below is an example of routing from Hub to a Slot-3 Fex

ROD_BUSY

When active, this signal indicates that the ROD cannot currently accept further data from Fex sources. This signal is taken from the Readout_Ctrl and is fanned-out to the shelf FEX's over each slot's Combined_TTC link.

ROD 0 Channel Up, ROD 1 Channel UP

These signals indicate to each FEX whether or not the corresponding ROD Rx Aurora channel is "Up".

ROD Reserved

These bits are reserved for future ROD functionality to be defined

Shelf Number

The Hub provides the shelf number to all Combined_TTC targets via these bits.

CRC (9-bit)

CRC is included to provide additional robustness on this link. Erroneous resets could cause loss of data. Choice of polynomial is from https://users.ece.cmu.edu/~koopman/crc/index.html

The chosen polynomial is 0x17d in Koopman format of 0x2fb in explicit+1 format.

2.2 Combined_TTC/DATA - FW development strategy

The Combined_TTC/DATA FW development comprises several stages. In terms of Combined_TTC/DATA link, the first stage assumes to transmit the Combined_TTC/Data stream to the ROD module, also to the FEX slot 3 (see Figure 12 and 13). In order to debug the design, the standard Xilinx diagnostic components are being used to monitor the data flow (as for example like the ILA and VIO). The physical layer is configured with the use of GT wizard.

For the purpose of the initial test, the HUB transmitter side generates the 128 bits from 4 Control registers but only some static patterns are written into these registers. In the next step, the Readout_Ctrl data received on the HUB from ROD module are retransmitted as the Combined_TTC/Data stream to the ROD and FTM module in slot 3. Once the communication between the HUB and other modules within shelf is established, a test pattern generator will be replaced by real TTC component. Next development steps assumes to add (gradually) the remaining receivers in the shelf.
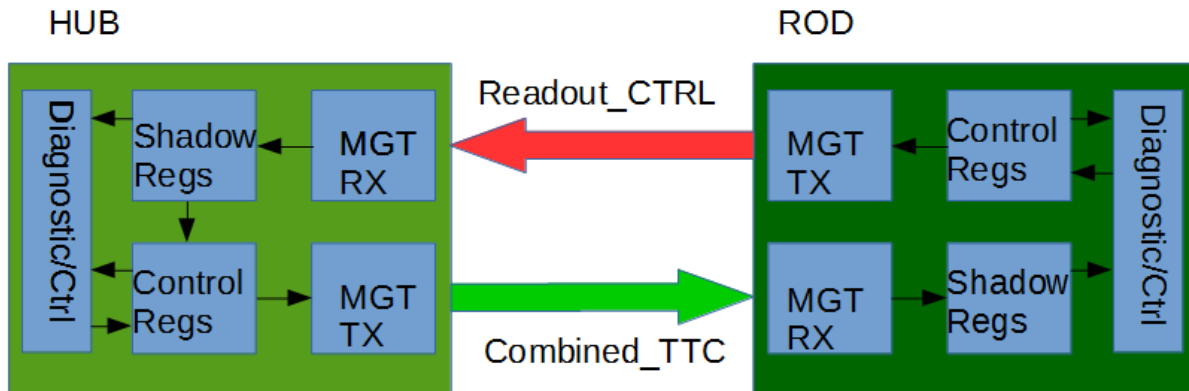
Figure 12. The HUB and ROD setup to test the Readout_Ctrl and Combined_TTC/Data link.

Figure 13. The Combined_TTC Data link test: A correct Combined_TTC stream on the ROD (upper plot) and on the FTM module in slot 3 (lower plot).

word 0 bit	0XBC = K28.5	word 1 bit		word 2 bit		word 3 bit	
0	0	0	L1ID(0)	0	control channel	0	Link_reset 0
1	0	1	L1ID	1	control channel	1	Link_reset 1
2	1	2	L1ID	2	control channel	2	Link_reset 2
3	1	3	L1ID	3	control channel	3	Link_reset 3
4	1	4	L1ID	4	control channel	4	ROD Busy
5	1	5	L1ID	5	control channel	5	Link Enable
6	0	6	L1ID	6	control channel	6	rod 0 channel up
7	1	7	L1ID	7	control channel	7	rod 1 channel up
8	version(0)	8	L1ID	8	control channel	8	0 (ROD reserved)
9	version(1)	9	L1ID	9	control channel	9	0 (ROD reserved)
10	version(2)	10	L1ID	10	control channel	10	0 (ROD reserved)
11	version(3)	11	L1ID	11	control channel	11	0 (ROD reserved)
12	reset	12	L1ID	12	control channel	12	0 (ROD reserved)
13	reset	13	L1ID	13	control channel	13	0 (ROD reserved)
14	reset	14	L1ID	14	control channel	14	0 (ROD reserved)
15	reset	15	L1ID	15	control channel	15	0 (ROD reserved)
16	L1A	16	L1ID	16	control channel	16	0 (ROD reserved)
17	BCR	17	L1ID	17	control channel	17	0 (ROD reserved)
18	ECR	18	L1ID	18	control channel	18	0 (ROD reserved)
19	Privileged Readout	19	L1ID	19	control channel	19	0 (ROD reserved)
20	0 (TTC reserved)	20	L1ID	20	control channel	20	shelf #
21	0 (TTC reserved)	21	L1ID	21	control channel	21	shelf #
22	0 (TTC reserved)	22	L1ID	22	control channel	22	shelf #
23	0 (TTC reserved)	23	L1ID(23)	23	control channel	23	CRC (9-bit)
24	0 (TTC reserved)	24	ECRID(0)	24	control channel	24	CRC (9-bit)
25	0 (TTC reserved)	25	ECRID(1)	25	control channel	25	CRC (9-bit)
26	0 (TTC reserved)	26	ECRID(2)	26	control channel	26	CRC (9-bit)
27	0 (TTC reserved)	27	ECRID(3)	27	control channel	27	CRC (9-bit)
28	0 (TTC reserved)	28	ECRID(4)	28	control channel	28	CRC (9-bit)
29	0 (TTC reserved)	29	ECRID(5)	29	control channel	29	CRC (9-bit)
30	0 (TTC reserved)	30	ECRID(6)	30	control channel	30	CRC (9-bit)
31	0 (TTC reserved)	31	ECRID(7)	31	control channel	31	CRC (9-bit)

Table 2. Combined TTC/Data bit definitions.

3.0 Readout Control (Readout_Ctrl)

The HUB module is obliged to receive the Readout Control (Readout_CTRL) information from the ROD via serial link named Readout_CTRL [2]. The HUB module is the only one module within the shelf which gets the Readout Control data from the ROD. That information is used by the Hub, also fanned out to the rest of the system. The main purpose is to provide resets to all of the data links (Aurora) between the Fex's and the ROD plus HUB module.

3.1 Readout Control - Physical Implementation

The Readout_Ctrl link on the HUB module is implemented with the use of MGT Transceiver GTH), control and diagnostic logic. The Readout_Ctrl link is designed to operate at 6.4 Gbps. When 8b/10b overhead is accounted for, the effective rate is reduced to 6.4 x 0.8 = 5.12 Gbps. In order to to control message transmission within a single LHC clock period, the lengh of the message is limited tp 128 bts. The HUB Readout Control FW features one receiver (RX). In order to debug the design, the standard Xilinx diagnostic components are being used to monitor the data flow (as for example like the ILA and VIO). The physical layer is configured with the use of GT wizard.

The physical implementation of the Readout_CTRL links assumes that there are Control Registers on the Tx ROD side, and Shadow Registers on the Rx HUB side. The transmitter side generates the 128 bit message from 4 Control registers: Word_0, Word_1, Word_2, and Word_3. The transmitter side logic is in charge to write control information into these registers for transmission to the modules within the shelf. These registers are continuously transmitted to the HUB which receives the data into a duplicate set of 4 registers referred to as Shadow Registers. Anything written into a Control Register at the transmitter side will appear in the corresponding Shadow register at the receiving side within the following LHC clock. The least significant byte of Word_0 is reserved for the 8b10b Comma character K28.5.

$$(1/(6.4\text{Gbps} *0.8)) * 128 \text{ bits} = 2.5\text{ns} = 1/40\text{MHz}$$
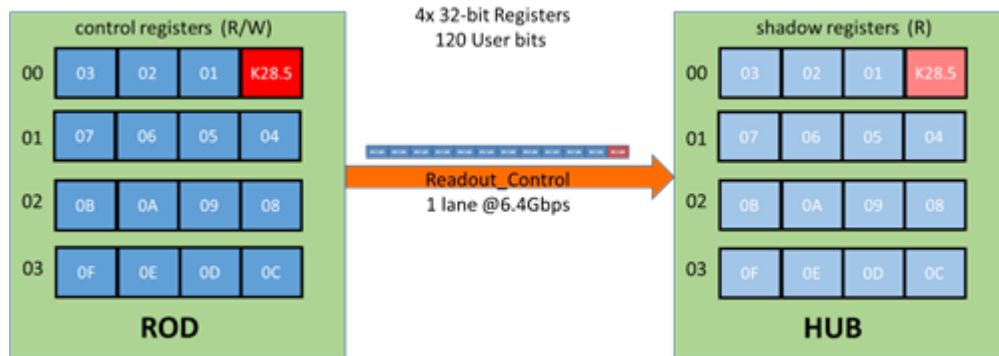
Figure 14. Control and Shadow Register

3.2 Readout Control - Control Bit definition

The bits within the Readout Control words are primarily defined to provide initialization functions for all of the FEX data links.

Comma Character

Bits 7 to 0 of Word_0 contain the Comma character that maintains alignment of the 4 shadow registers with their corresponding Control registers. The chosen character is K28.5 = 0xBC.

Version

The 4-bit value contains the version number of this overall bit assignment.

ROD_BUSY

When active, this signal indicates that the ROD cannot currently accept further data from Fex sources. This signal is fanned-out to the shelf FEX's by the Hub via the Combined_TTC link.

Global_Link_Reset

This single bit is used to reset all of the data (Aurora) links within the shelf. The primary use is in the first initialisation after power-up. The ROD can hold this reset active for an indefinite amount of time. On the trailing edge (deactivation), the eFex's should provide additional timing control for the GTReset and Reset signals on the Aurora interface. This signal is fanned-out to the shelf FEX's by the Hub via the Combined_TTC link.

Channel Up

This reports on whether each slot's Aurora Channel is 'UP' or 'Down'. The information is rebroadcast to each FEX slot in the so that the Aurora transmitter knows if the receiver side of the channel is up. If a channel goes down, the ROD attempt to reset both Rx and Tx ends of the link.

Slot_N_link_reset_M

These signals allow the ROD to reset individual links providing data to it. The active time is indeterminate, and may be very short. On the trailing edge (deactivation), the eFex's should provide additional timing control for the GTReset and Reset signals on the Aurora interface.

Slots 3-8 have 4 resets each. This is to cover the case of jFex where each of 4 processors has its own data link to the ROD. It is expected that eFex boards in these slots will only use the M=0 reset.

Slots 9-14 only have a single reset. These slots will only be populated by eFEX boards which only require one reset.

CRC (9-bit)

CRC is included to provide additional robustness on this link. Erroneous resets could cause loss of data. Choice of polynomial is from https://users.ece.cmu.edu/~koopman/crc/index.html. The chosen polynomial is 0x17d in Koopman format of 0x2fb in explicit+1 format.

3.3 Readout Control - FW development strategy

The HUB Readout Control FW features one receiver (RX). In order to debug the design, the standard Xilinx diagnostic components are being used to monitor the data flow (as for example like the ILA and VIO). The physical layer is configured with the use of GT wizard.

word 0 bit	0XBC = K28.5	word 1 bit		word 2 bit		word 3 bit	
0	0	0	slot 3 link reset 0	0	0	0	0
1	0	1	slot 3 link reset 1	1	0	1	0
2	1	2	slot 3 link reset 2	2	0	2	0
3	1	3	slot 3 link reset 3	3	0	3	0
4	1	4	slot 4 link reset 0	4	0	4	0
5	1	5	slot 4 link reset 1	5	0	5	0
6	0	6	slot 4 link reset 2	6	0	6	0
7	1	7	slot 4 link reset 3	7	0	7	0
8	version 0	8	slot 5 link reset 0	8	0	8	0
9	version 1	9	slot 5 link reset 1	9	0	9	0
10	version 2	10	slot 5 link reset 2	10	0	10	0
11	version 3	11	slot 5 link reset 3	11	0	11	0
12	0	12	slot 6 link reset 0	12	0	12	0
13	0	13	slot 6 link reset 1	13	0	13	0
14	ROD_BUSY (to all slots)	14	slot 6 link reset 2	14	0	14	0
15	Aurora_Init (all links)	15	slot 6 link reset 3	15	0	15	0
16	slot3 channel up	16	slot 7 link reset 0	16	0	16	0
17	slot4 channel up	17	slot 7 link reset 1	17	0	17	0
18	slot5 channel up	18	slot 7 link reset 2	18	0	18	0
19	slot6 channel up	19	slot 7 link reset 3	19	0	19	0
20	slot7 channel up	20	slot 8 link reset 0	20	0	20	0
21	slot8 channel up	21	slot 8 link reset 1	21	0	21	0
22	slot9 channel up	22	slot 8 link reset 2	22	0	22	0
23	slot10 channel up	23	slot 8 link reset 3	23	0	23	CRC (9-bit)
24	slot11 channel up	24	slot 9 link reset	24	0	24	CRC (9-bit)
25	slot12 channel up	25	slot 10 link reset	25	0	25	CRC (9-bit)
26	slot13 channel up	26	slot 11 link reset	26	0	26	CRC (9-bit)
27	slot14 channel up	27	slot 12 link reset	27	0	27	CRC (9-bit)
28	0	28	slot 13 link reset	28	0	28	CRC (9-bit)
29	0	29	slot 14 link reset	29	0	29	CRC (9-bit)
30	0	30	0	30	0	30	CRC (9-bit)
31	0	31	0	31	0	31	CRC (9-bit)

Table 3: Readout_Ctrl bit assignments.

4.0 Specification for the HUB Safe Configuration

Introduction

During the regular operation, the HUB firmware is obliged to control the group of signals which are wired to the FPGA. These signals are handled on the HUB FPGA by the Safe Configuration component. This piece of firmware is in charge to properly receive the signals and control them by the ILA and VIO component. The HUB Safe Configuration needs to present in any type of HUB configuration.

Logic Analyzer (ILA)

The customizable Integrated Logic Analyzer (ILA) IP core is a logic analyzer core that can be used to monitor the internal signals of a design [4]. The ILA core includes many advanced features of modern logic analyzers, including Boolean trigger equations, and edge transition triggers. Because the ILA core is synchronous to the design being monitored, all design clock constraints that are applied to your design are also applied to the components inside the ILA core.

Virtual Input/Output (VIO)

The LogiCORE™ IP Virtual Input/Output (VIO) core is a customizable core that can both monitor and drive internal FPGA signals in real time [5]. The number and width of the input and output ports are customizable in size to interface with the FPGA design. Because the VIO core is synchronous to the design being monitored and/or driven, all design clock constraints that are applied to your design are also applied to the components inside the VIO core. Run time interaction with this core requires the use of the Vivado® logic analyzer feature.

SYSMON

Optionally, the HUB configuration can include the SYSMON (SLR0 and SLR1): Each super logic region: SLR0 and SLR1 has one system monitor to provide for monitoring supply voltages within the SLR. The I2C DRP and JTAG DRP access is limited to the master SLR only (SYSMONE1_X0Y0 for devices with two SLRs) [7]. The system monitors can be placed in the bottom SLR0 (SYSMONE1_X0Y0) and then consecutively in the upper SLR increasing Y locations (SYSMONE1_X0Y1). Monitoring across SLR boundaries is not possible. Temperature, VCCINT, VCCAUX, VCCBRAM measurements are specific to an individual SLR. For the UltraScale FPGAs SYSMONE1, the System Management Wizard provides I2C functionality to the slave SLRs using the DRP port and additional logic. The SYSMON has numerous operating modes that are user-defined by writing to the control registers, which can be accessed through DRP, JTAG or I2C. It is also possible to initialize these register contents when the SYSMON is instantiated in a design using the block attributes.

No	Signal name	Location and I/O Standards	Components	Direction
0	Logic_Clk_320.64_Mhz_to_FPGA_Dir Logic_Clk_320.64_MHz_to_FPGA_Cmp	PACKAGE_PIN K22 PACKAGE_PIN J22 IOSTANDARD LVDS DIFF_TERM_ADV TERM_100 DQS_BIAS TRUE EQUALIZATION EQ_LEVEL0	IBUFGDS	IN
1	Logic_Clk_40.08_Mhz_to_FPGA_Dir Logic_Clk_40.08_Mhz_to_FPGA_Cmp	PACKAGE_PIN J24 PACKAGE_PIN H24 IOSTANDARD LVDS DIFF_TERM_ADV TERM_100 DQS_BIAS TRUE EQUALIZATION EQ_LEVEL0	IBUFGDS	IN
2	Ref_4008_Mhz_from_Other_Hub_Dir Ref_4008_Mhz_from_Other_Hub_Cmp	PACKAGE_PIN H23 PACKAGE_PIN G23 IOSTANDARD LVDS DIFF_TERM_ADV TERM_100 DQS_BIAS TRUE EQUALIZATION EQ_LEVEL0	IBUFGDS	IN
3	Ref_4008_MHz_from_FPGA_to_Rec_Dir Ref_4008_MHz_from_FPGA_to_Rec_Cmp	PACKAGE_PIN AV31 PACKAGE_PIN AW31	Controling only, an VIO component added. A default value set to 0.	OUT

		IOSTANDARD LVDS		
4	PLL_4008_Mhz_Lock_Detect_to_FPGA	PACKAGE_PIN B27 IOSTANDARD LVCMOS18	Monitoring only, an ILA component added.	IN
5	PLL_32064_MHz_Lock_Detect_to_FPGA	PACKAGE_PIN B26 IOSTANDARD LVCMOS18	Monitoring only, an ILA component added.	IN
6	Select_Input_Second_40_Fanout	PACKAGE_PIN A26 IOSTANDARD LVCMOS18	Controling only, an VIO component added. A default value set to 0.	OUT
7	FPGA_SW_A_ATC_LOOP_DET FPGA_SW_B_ATC_LOOP_DET FPGA_SW_C_ATC_LOOP_DET	PACKAGE_PIN AV13 PACKAGE_PIN AT14 PACKAGE_PIN AV15 IOSTANDARD LVCMOS33 SLEW SLOW DRIVE 4	Controling only, an VIO component added. A default value set to 0.	OUT
8	FPGA_SW_A_LOOP_DETECTED FPGA_SW_B_LOOP_DETECTED FPGA_SW_C_LOOP_DETECTED	PACKAGE_PIN AY15 PACKAGE_PIN AU13 PACKAGE_PIN AV16 IOSTANDARD LVCMOS33	Monitoring only, an ILA component added.	IN
9	FPGA_SW_A_MDC FPGA_SW_B_MDC FPGA_SW_C_MDC	PACKAGE_PIN AW13 PACKAGE_PIN AT16 PACKAGE_PIN AY13 IOSTANDARD LVCMOS33 SLEW SLOW DRIVE 4	Controling only, an VIO component added. A default value set to 0.	OUT
10	FPGA_SW_A_MDIO FPGA_SW_B_MDIO FPGA_SW_C_MDIO	PACKAGE_PIN AW15 PACKAGE_PIN AU16 PACKAGE_PIN AY14 IOSTANDARD LVCMOS33 PULLUP TRUE	Monitoring only, an ILA component added.	IN
11	ISO_SLOT_HW_ADRS_0 ISO_SLOT_HW_ADRS_1	PACKAGE_PIN AT12 PACKAGE_PIN AT11	Monitoring only, an ILA component added.	IN

	ISO_SLOT_HW_ADRS_2 ISO_SLOT_HW_ADRS_3 ISO_SLOT_HW_ADRS_4 ISO_SLOT_HW_ADRS_5 ISO_SLOT_HW_ADRS_6 ISO_SLOT_HW_ADRS_7	PACKAGE_PIN AU12 PACKAGE_PIN AU11 PACKAGE_PIN AV11 PACKAGE_PIN AW12 PACKAGE_PIN AW11 PACKAGE_PIN AY12 IOSTANDARD LVCMOS33		
12	SHELF_ADRS_0_TO_FPGA SHELF_ADRS_1_TO_FPGA SHELF_ADRS_2_TO_FPGA SHELF_ADRS_3_TO_FPGA SHELF_ADRS_4_TO_FPGA SHELF_ADRS_5_TO_FPGA SHELF_ADRS_6_TO_FPGA SHELF_ADRS_7_TO_FPGA	PACKAGE_PIN BB15 PACKAGE_PIN BB14 PACKAGE_PIN BA14 PACKAGE_PIN BB13 PACKAGE_PIN BB12 PACKAGE_PIN BB11 PACKAGE_PIN BA12 PACKAGE_PIN BA11 IOSTANDARD LVCMOS33	Monitoring only, an ILA component added.	IN
13	OVERALL_ADRS_0_TO_RES_NET OVERALL_ADRS_1_TO_RES_NET OVERALL_ADRS_2_TO_RES_NET OVERALL_ADRS_3_TO_RES_NET OVERALL_ADRS_4_TO_RES_NET OVERALL_ADRS_5_TO_RES_NET OVERALL_ADRS_6_TO_RES_NET OVERALL_ADRS_7_TO_RES_NET	PACKAGE_PIN BF25 PACKAGE_PIN BE25 PACKAGE_PIN BF26 PACKAGE_PIN BE27 PACKAGE_PIN BF27 PACKAGE_PIN BE28 PACKAGE_PIN BE29 PACKAGE_PIN BE30 IOSTANDARD LVCMOS18 SLEW SLOW DRIVE 4	Controling only, an VIO component added. A default value set to 0.	OUT
14	HUB_I2C_TO_FPGA_SCL_0 HUB_I2C_TO_FPGA_SDA_0 Hub_I2C_TO_FPGA_SCL_1 Hub_I2C_TO_FPGA_SDA_1	PACKAGE_PIN BE16 PACKAGE_PIN BF16 PACKAGE_PIN BA29 PACKAGE_PIN BB29 IOSTANDARD LVCMOS18	No monitoring.	IN
15	I2C_Buf_1501_ENABLE I2C_Buf_1502_ENABLE I2C_Buf_1503_ENABLE	PACKAGE_PIN BA16 PACKAGE_PIN BA15 PACKAGE_PIN BB16 IOSTANDARD LVCMOS33 SLEW SLOW DRIVE 4	Controling only, an VIO component added. A default value set to 1 for I2C_Buf_1501_ENABLE, others set to 0.	OUT
16	Recvr_MiniPOD_INTR_B Trans_MiniPOD_INTR_B	PACKAGE_PIN AR15 PACKAGE_PIN AN16 IOSTANDARD LVCMOS33	Monitoring only, an ILA component added.	IN

17	Recvr_MiniPOD_RESET_B Trans_MiniPOD_RESET_B	PACKAGE_PIN AR14 PACKAGE_PIN AP13 IOSTANDARD LVCMOS33 SLEW SLOW DRIVE 4	Controling only, an VIO component added. A default value set to 1.	OUT
18	Recvr_MiniPOD_SCL Trans_MiniPOD_SCL	PACKAGE_PIN AR12 PACKAGE_PIN AN15 IOSTANDARD LVCMOS33 SLEW SLOW DRIVE 4	Controling only, an VIO component added. A default value set to 0.	OUT
19	Recvr_MiniPOD_SDA Trans_MiniPOD_SDA	PACKAGE_PIN AR13 PACKAGE_PIN AP15 IOSTANDARD LVCMOS33	Monitoring only, an ILA component added.	IN
20	ACCESS_SIGNAL_1_FROM_FPGA ACCESS_SIGNAL_2_FROM_FPGA	PACKAGE_PIN AT30 PACKAGE_PIN AT31 IOSTANDARD LVCMOS18 SLEW SLOW DRIVE 4	Controling only, an VIO component added. A default value set to 0.	OUT
21	HUB_FPGA_LED50_DRV HUB_FPGA_LED51_DRV HUB_FPGA_LED52_DRV	PACKAGE_PIN BF29 PACKAGE_PIN BF30 PACKAGE_PIN BF31 IOSTANDARD LVCMOS18 SLEW SLOW DRIVE 4	Controling only, an VIO component added. A default value set to 0.	OUT
22	Hubs_SMB_Alert_B	PACKAGE_PIN AP16 IOSTANDARD LVCMOS33	Monitoring only, an ILA component added.	IN
23	ALL_HUB_POWER_GOOD_TO_FPGA	PACKAGE_PIN AM16 IOSTANDARD LVCMOS33	Monitoring only, an ILA component added.	IN
24	MGT_FO_EQU_ENB_GRP_1 MGT_FO_EQU_ENB_GRP_2 MGT_FO_EQU_ENB_GRP_3 MGT_FO_EQU_ENB_GRP_4 MGT_FO_EQU_ENB_GRP_5 MGT_FO_EQU_ENB_GRP_6 MGT_FO_EQU_ENB_GRP_7 MGT_FO_EQU_ENB_GRP_8 MGT_FO_EQU_ENB_GRP_9 MGT_FO_EQU_ENB_GRP_10 MGT_FO_EQU_ENB_GRP_11	PACKAGE_PIN C25 PACKAGE_PIN A25 PACKAGE_PIN B25 PACKAGE_PIN A24 PACKAGE_PIN C24 PACKAGE_PIN B22 PACKAGE_PIN D20 PACKAGE_PIN C20 PACKAGE_PIN A23 PACKAGE_PIN A21 PACKAGE_PIN A20	Controling only, an VIO component added. A default value set to 0.	OUT

	MGT_FO_EQU_ENB_GRP_12 MGT_FO_EQU_ENB_GRP_13	PACKAGE_PIN A19 PACKAGE_PIN A18 IOSTANDARD LVCMOS18 SLEW SLOW DRIVE 8		
25	TBD_SPARE_LINK_0_Dir TBD_SPARE_LINK_0_Cmp TBD_SPARE_LINK_1_Dir TBD_SPARE_LINK_1_Cmp TBD_SPARE_LINK_2_Dir TBD_SPARE_LINK_2_Cmp TBD_SPARE_LINK_3_Dir TBD_SPARE_LINK_3_Cmp	PACKAGE_PIN AV26 PACKAGE_PIN AW26 PACKAGE_PIN AT27 PACKAGE_PIN AU27 PACKAGE_PIN AY27 PACKAGE_PIN AY28 PACKAGE_PIN AT29 PACKAGE_PIN AU29 IOSTANDARD LVCMOS18SLEW SLOW DRIVE 4	Controling only, an VIO component added. A default value set to 0.	OUT
26	ROD_PRESENT_B_TO_FPGA	PACKAGE_PIN AW27 IOSTANDARD LVCMOS18	Monitoring only, an ILA component added.	IN
27	ROD_Power_Control_2_FPGA ROD_Power_Control_3_FPGA ROD_Power_Control_4_FPGA	PACKAGE_PIN AW28 PACKAGE_PIN AV30 PACKAGE_PIN AV29 IOSTANDARD LVCMOS18	Monitoring only, an ILA component added.	IN
28	ROD_Power_Enable ROD_Power_Enable_B	PACKAGE_PIN AR17 PACKAGE_PIN AP17 IOSTANDARD LVCMOS33 SLEW SLOW DRIVE 4	Controling only, an VIO component added. A default value: ROD_Power_Enable set 0 ROD_Power_Enable_B set 1	OUT
29	FPGA_RODs_SMBALERT_B	PACKAGE_PIN BB28 IOSTANDARD LVCMOS18	Monitoring only, an ILA component added.	IN
30	Phys_U21_TXD0 Phys_U21_TXD1 Phys_U21_TXD2 Phys_U21_TXD3 Phys_U21_TX_EN Phys_U21_GTX_CLK Phys_U22_TXD0 Phys_U22_TXD1 Phys_U22_TXD2 Phys_U22_TXD3 Phys_U22_TX_EN Phys_U22_GTX_CLK	PACKAGE_PIN BA36 PACKAGE_PIN AY35 PACKAGE_PIN BB36 PACKAGE_PIN BA35 PACKAGE_PIN BB34 PACKAGE_PIN BA34 PACKAGE_PIN AR36 PACKAGE_PIN AT36 PACKAGE_PIN AR35 PACKAGE_PIN AT35 PACKAGE_PIN AU34 PACKAGE_PIN AT34 IOSTANDARD LVCMOS18	Controling only, an VIO component added. A default value set to 0.	OUT
31	Phys_U21_CLK125_LED_MODE	PACKAGE_PIN AU33	Monitoring only, an ILA	IN

	Phys_U21_RXD0_MODE0 Phys_U21_RXD1_MODE1 Phys_U21_RXD2_MODE2 Phys_U21_RXD3_MODE3 Phys_U21_RX_DV_CLK125_EN Phys_U21_RX_CLK_PHYAD2 Phys_U22_CLK125_LED_MODE Phys_U22_RXD0_MODE0 Phys_U22_RXD1_MODE1 Phys_U22_RXD2_MODE2 Phys_U22_RXD3_MODE3 Phys_U22_RX_DV_CLK125_EN Phys_U22_RX_CLK_PHYAD2	PACKAGE_PIN BC31 PACKAGE_PIN BA32 PACKAGE_PIN BB33 PACKAGE_PIN AY33 PACKAGE_PIN BB31 PACKAGE_PIN AV33 PACKAGE_PIN AY34 PACKAGE_PIN AV35 PACKAGE_PIN AV36 PACKAGE_PIN AV34 PACKAGE_PIN AU32 PACKAGE_PIN AW32 PACKAGE_PIN AW33 IOSTANDARD LVCMOS18	component added.	
32	Spare_OSC_TO_FPGA_Dir Spare_OSC_TO_FPGA_Cmp	PACKAGE PIN AU28 PACKAGE PIN AV28 IOSTANDARD LVDS DQS_BIAS TRUE	IBUFGDS	IN

Table 4: "Hub Safe Configuration" signal list, including the location and I/O standard, components which are being used to receive these signals and directionality.

Hub FPGA Signal Types:

Clock and Clock Management Signals:

Logic_Clk_320.64_MHz_to_FPGA_Dir/Cmp
Logic_Clk_40.08_MHz_to_FPGA_Dir/Cmp

These are the main Logic Clock signals for the Hub FPGA. These are AC coupled LHC locked clocks. The FPGA must receive them as differential signals, 100 Ohm terminate them, and DC Bias these FPGA inputs by including the proper combination of DQS_Bias and EQ_Level0 attributes in the setup of these I/O Blocks. Even if these clocks are not used in a minimal safe FPGA design they must still be properly received.

Ref_40.08_MHz_from_Other_Hub_ Dir/Cmp

This is an AC coupled LVDS input to the Hub FPGA. In a shelf with two Hub Modules this clock path is used on the Hub that does not receive an Optical Felix TTC signal. Because this is AC coupled the I/O Block that receives this signal must be setup with the correct combination of the DQS_Bias and EQ_Level0 attributes. This FPGA input should include a 100 Ohm differential terminator. This input needs to be setup and instantiated in all versions of the Hub FPGA firmware.

Ref_40.08_MHz_from_FPGA_to_Rec_ Dir/Cmp

This is an LVDS output from the Hub FPGA that runs to an LVDS receiver and then connects to the reference input on the 40.08 MHz PLL. I believe that we should include this output even in a minimal safe FPGA design even ifvwe only tie the input to the LVDS driver Low.

PLL_40.08_Mhz_Lock_Detect_to_FPGA
PLL_320.64_MHz_Lock_Detect_to_FPGA

These are two Lock Detect signals from the PLL Clock circuits on the Hub Module. When Hi they indicate that the associated PLL has locked onto its reference input. The FPGA needs to always provide 1.8V CMOS receivers for these two signals.

Select_Input_Second_40_Fanout

This signal is an output from the Hub FPGA. It is used to enable or disable the Hub circuit board from sending out 40.08 MHz Reference Clocks on Zone-2 of the Backplane. Recall that in a shelf with 2 Hub Modules, that only the primary Hub that receives the Optical Felix TTC signal will send out its 40.08 MHz Reference Clocks to the FEX modules. The secondary Hub (the Hub that does not receive an Optical Felix TTC signal) must not send out its Zone-2 Reference Clocks. This is a 1.8V CMOS open-drain output from the Hub Module. When Low it enables the Hub Module to send out its Zone-2 Reference Clocks. When the open-drain output transistor in the FPGA I/O Block is Off then its backplane Reference Clock output drives will be shutdown. In a minimal safe design we want this open-drain output to pull this signal Low so that we can see these Reference Clock outputs.

SPARE_OSC_TO_FPGA_Dir/Cmp

This is a spare differential clock input to the Hub FPGA. As long as we are not installing this part (U562) then nothing special needs to be done with these input pins to the FPGA even in a minimal safe FW design.

Ethernet Switch Chip Management Signals:

Each of the 3 Broadcom Ethernet Switch chips has its own set of 4 management signals with the Hub FPGA. See Hub Circuit Diagram 17.

FPGA_SW_X_ATC_LOOP_DET

This is a control signal from the Hub FPGA to the Ethernet Switch chips. I assume that this signal will eventually be controlled from a bit in an IPBus visible register. A minimal Hub FPGA design could just tie this signal Low. A Slow Slew and 4 mA Drive level is fine for this 3.3V CMOS static level output signal.

FPGA_SW_X_LOOP_DETECTED

This is a status signal from the Broadcom Switch chips to the Hub FPGA. I assume that this status bit will eventually be visible in an IPBus register. The FPGA needs to at least always provide a receiver for this 3.3V CMOS input signal.

FPGA_SW_X_MDC

Because the MDC/MDIO interface on the Broadcom Switch chips is in Slave mode, this clock signal runs from the Hub FPGA to the Switch chip. Eventually I assume that these pins on the Hub FPGA will be controlled by firmware that implements the MDC/MDIO part of a MAC. For minimal safe firmware this line could just be driven Low by a Slow Slew 4 mA 3.3V CMOS output driver.

FPGA_SW_X_MDIO

This is a bi-directional data line between the Hub FPGA and the MDC/MDIO interface in the Broadcom Switch chips. Eventually I assume that these pins on the Hub FPGA will be handled by firmware that implements the MDC/MDIO part of a MAC. In a minimal safe Hub FPGA design I would provide a 3.3V CMOS receiver for this signal with a weak pull-up resistor at its input.

Hardware Address Signals:

ISO_SLOT_HW_ADRS_X

These 8 signals bring the backplane Hardware Slot Address to the Hub FPGA. The IPMC module also receives these signals. Even if the Hub FPGA is not going to use this Hardware Slot number information it must still receive these 8 lines with 3.3V CMOS receivers.

SHELF_ADRS_X_TO_FPGA

These are 8 signals that bring the Shelf Address information from the IPMC to the Hub's FPGA. These lines connect to "User I/O" pins on one of the ARM CPUs in the IPMC module. The Hub Firmware needs to at least always provide 3.3V CMOS receives for these signals.

OVERALL_ADRS_X_TO_RES_NET

These are 8 outputs from the Hub FPGA that provide an Overall Hardware Address to the ROD mezzanine. This Overall Hardware Address is made up of the Slot Hardware Address that comes from the Backplane and the Shelf Hardware Address that comes from the IPMC. In a minimal safe FPGA implementation I would just tie all 8 of these Overall Hardware Address lines Low. These can be 1.8V CMOS Slow Slew 4 mA Drive outputs.

I2C Bus Signals:

Hub_I2C_to_FPGA_SCL
Hub_I2C_to_FPGA_SDA

Note that this I2C bus makes TWO connections to the Hub FPGA. It connects to pins:

BE16   IO_L23P_T3U_N8_I2C_SCLK_65
BF16   IO_L23N_T3U_N9_I2C_SDA_65

that provide for a Slave I2C interface to the FPGA System Monitor. It connects to pins:

BA29   IO_L18P_T2U_N10_AD2P_67
BB29   IO_L18N_T2U_N11_AD2N_67

so that one can implement a Master I2C interface so that for example the Hub FPGA can talk to the Hub's DCDC Converter power supplies. In a minimal safe design both of these I2C signals at both pin pairs should connect to 1.8V CMOS Receivers.

I2C_Buf_150X_ENABLE

These are control signals from the Hub FPGA to the Sensor I2C Bus translator/buffer chips. These translator/buffer chips allow the overall Sensor I2C Bus on the ROD-Hub cards to be divided up into sections. Eventually we may need to control dividing up the Sensor I2C bus either from bits in an IPBus visible register of from a control signal from the ROD to the Hub. For a minimal safe Hub FPGA design these 3 I2C buffer control signals can just be tied Hi which will enable these I2C translator/buffers. These control signals are 3.3V Slow Slew 4 mA Drive outputs from the Hub FPGA.

MiniPOD Management Signals:

Each of the MiniPODs has 4 management signals associated with it. The Hub's Receiver and Transmitter MiniPODs each have their own private set of these 4 management signals. The setup of the MiniPOD management signals on the Hub is very similar to that used on the CMX card.

Recvr_MiniPOD_INTR_B
Trans_MiniPOD_INTR_B

Interrupt signals from the MiniPODs to the FPGA. Low indicates interrupt. The FPGA should always provide 3.3V CMOS receivers for these two signals.

Recvr_MiniPOD_RESET_B
Trans_MiniPOD_RESET_B

These are Reset signals from the FPGA to the MiniPODs. Low indicates Reset. A minimal safe design should at least drive these lines Hi with a Slow Slew 4 mA Drive 3.3V CMOS output. In a full

firmware design I assume that these Reset signals will be controlled from bits in an IPBus visible register.

Recvr_MiniPOD_SCL
Trans_MiniPOD_SCL

These are the clock signals for the bi-directional serial data path between the FPGA and the MiniPODs. In a minimal safe design I would drive this signals Low with a Slow Slew 4 mA Drive 3.3V CMOS output.

Recvr_MiniPOD_SDA
Trans_MiniPOD_SDA

These are the bi-directional data lines for the serial data path between the FPGA and the MiniPODs. In a minimal design I would provide a 3.3V CMOS receiver for these signals.

Miscellaneous Select I/O Signals:

ACCESS_SIGNAL_X_FROM_FPGA

These two output signals from the FPGA run to a translator/buffer chip and then to pins in the front panel J2 connector. See Hub Circuit Diagram 53. The purpose of these signals is to allow one to see with an oscilloscope some aspect of the FPGA's operation. Unless needed for something special these 1.8V CMOS output signals can be configured for a Slow Slew rate and modest 4 mA Drive. A minimal safe Hub FPGA design could just tie these 2 signals Low.

HUB_FPGA_LED5X_DRV

These three signals are outputs from the Hub FPGA that control front panel LEDs. These are 1.8V CMOS signals and using a Slow Slew and 4 mA of Drive would be fine.
A minimal safe Hub FPGA design could just tie these 3 output signals Low.

Hubs_SMB_Alert_B

This is an input signal to the Hub's FPGA from the 7 DCDC Converter power supplies on the Hub Module. When Low it indicates that one or more of these Hub power supplies is in trouble (or at least wants attention). Normally this signal should be Hi. This is a 3.3V CMOS signal and the FPGA must always have a receiver for it even if it does not make use of this signal.

ALL_HUB_POWER_GOOD_TO_FPGA

This is an input signal to the Hub's FPGA from the Power Control circuits on the Hub Module. This is a 3.3V Select I/O signal in Bank 84. When Low this signal indicates that there is some kind of power supply problem on the Hub Module. When Hi this signal indicates: that all 7 DCDC Converters report that they are operating correctly, and that the Isolated +12V supply is Enabled, and that the output from the Linear MGT_AVAUX and BULK_2V5 supplies is good. The Hub FPGA must always

instance a 3.3V CMOS receiver for this signal even if it does not make use of this signal. This signal comes through a 2.7k Ohm isolation resistor from a 5V source.

MGT_FO_EQU_ENB_GRP_XX

These 13 control signals are outputs from the Hub FPGA that enable or disable the equalization in the MGT Fanout chips. All fanout chips that service a given data source either have their equalization enabled or disabled. I assume that eventually these 13 equalization enable signals will be separately controllable from bits in an IPBus visible register. In a minimal safe Hub FPGA design all of the Equalization Enable signals can just be tied HI. I would drive them as 1.8V CMOS outputs with Slow Slew and 8 mA of Drive. The loading on these signals will only be about 150 uA but the high Drive level may help if we need to ramp up the FAN_1V8 rail voltage.

TBD_SPARE_LINK_X_Dir/Cmp

These are 8 To Be Determined spare lines between the Hub and ROD FPGAs. Currently none of them have an assigned function. If needed they can be operated as either 8 separate 1.8V CMOS signals or as 4 separate LVDS signals. As with all CMOS signals we can not just leave them floating. As unassigned spare signals the agreement with ROD is that we will run them as 8 separate CMOS signals and that the Hub will drive them and the ROD will receive them. Further the agreement with ROD is that the Hub will keep these signals tri-stated until the ROD asserts it Power Control #2 signal HI to the Hub. See Hub Circuit Diagram 42. Even in a minimal safe firmware design the Hub needs to implement this functionality. These 8 lines should be driven with Slow Slew 4 mA Drive 1.8V CMOS output blocks.

Phys_U21 and Phys_U22 signals:

Each Phys Chip makes 16 connections with the Hub's FPGA. In the real Hub operation these 16 signals will be managed by MAC IP firmware. This RGMII connection between the FPGA and the Phys Chips includes signals that may require fast slew rates, higher drive current, and DCI back (series) termination. These RGMII signals have been laid out carefully on the Hub circuit to minimize cross-talk, reflections, and signal loss. Careful consideration must be given to how the RMGII signals are handled in the FPGA I/O Blocks to provide both a good RGMII interface and to minimize interference with other parts of the Hub Module. For a minimal safe Hub FPGA design we still should handle these CMOS signals in a defined rational way. These signal should be driven with Slow Slew 1.8V CMOS outputs tied Low:

    Phys_U2X_TXD0
    Phys_U2X_TXD1
    Phys_U2X_TXD2
    Phys_U2X_TXD3
    Phys_U2X_TX_EN
    Phys_U2X_GTX_CLK

Phys_U2X_MDC

We should provide 1.8V CMOS receivers for the following signals:

Phys_U2X_RXD0__MODE0
Phys_U2X_RXD1__MODE1
Phys_U2X_RXD2__MODE2
Phys_U2X_RXD3__MODE3
Phys_U2X_RX_CLK__PHYAD2
Phys_U2X_RX_DV__CLK125_EN
Phys_U2X_CLK125__LED_MODE
Phys_U2X_MDIO
Phys_U2X_INT_B

ROD Present and ROD Power Control Signals:

These 7 signals have to do with letting the Hub know whether or not a ROD is installed on it, Controlling (i.e. enabling) the power on the ROD, and letting the Hub know whether or not all ROD power is good and that the ROD is ready for normal trigger system operation.

ROD_PRESENT_B_TO_FPGA

This signal is a pull-down resistor on the ROD. When the ROD is NOT present this signal goes HI on the Hub. The Hub must always provide a 1.8V CMOS receiver for
this signal even if it does not use it.

ROD_Power_Control_2_FPGA

This is an input to the Hub's FPGA. When HI this signal indicates that all power supplies on the ROD are operating correctly. The Hub must always provide a 1.8V CMOS receiver for this signal even if it does not use it.

ROD_Power_Control_3_FPGA

This is an input to the Hub's FPGA. When HI this signal indicates that the ROD is Configured and fully ready for normal L1Calo operation. The Hub must always provide a 1.8V CMOS receiver for this signal even if it does not use it.

ROD_Power_Control_4_FPGA

This is a spare power control signal. In a minimal safe firmware design the Hub should provide a 1.8V CMOS receiver for this signal with a weak pull-up resistor at its input.

ROD_Power_Enable
ROD_Power_Enable_B

These two signal are outputs from the Hub FPGA that through some hardwired logic on the Hub tell the ROD when it may turn ON its power supplies. These are 3.3V Slow Slew 4 mA Drive CMOS outputs. A minimal safe design (that just locks OFF the ROD power) should set ROD_Power_Enable Low and set ROD_Power_Enable_B Hi.

FPGA_RODs_SMBALERT_B

This is an input to the Hub's FPGA that when LOW indicates a power supply problem on the ROD. During normal operation this signal should always be HI. Pull up current to 1V8 is provided by a 1k Ohm resistor on page 25 of the ROD schematics. A normal 1.8V CMOS receiver should be used. The FPGA needs to always provide a receiver for this signal.

5. Readout Data (Aurora 8b10b)

Figure 15 shows the Hub's distribution of readout data in the context of the cards in the ATCA shelf. The readout data comes from the Node slots and from the FPGA on each Hub module. All of this data flows to both the ROD and to the FPGA on each Hub. The arrangement shown in Figure 14 supports 2 independent streams of readout data. That is, the readout stream processed by the ROD and Hub FPGA on Hub-1 can be independent of the readout stream flowing into Hub-2. The FEX-HUB receives 6 serial streams of Readout Data from each FEX Module using the Aurora8b10b protocol. Currently the link rate is set to 6.4Gbps. The Aurora implementation for the HUB FPGA will be conducted with the use of Aurora 8b10b IP core (Wizard). The Xilinx Aurora 8B/10B core supports the AMBA® protocol AXI4-Stream user interface [3]. The core implements the Aurora 8B/10B protocol using the high-speed serial transceivers. The Aurora 8B/10B core (Figure 15) is a scalable, lightweight, link-layer protocol for high-speed serial communication. The protocol is open and can be implemented using Xilinx FPGA technology. The protocol is typically used in applications requiring simple, low-cost, high-rate, data channels and is used to transfer data between devices using one or many transceivers. The Aurora8b10b IP core supports: 7 series GTX/GTH, UltraScale GTH, UltraScale+ GTH, GTP transceivers. Officially the Aurora8b10b IP core does not support the GTY transceivers. The HUB FPGA is equipped with two type of MGT transceivers (GTY and GTH) and therefore, two different implementation method will be used. The GTY requires non-standard implementation. That method assumes that the Aurora IP core (wizard) will generate the Aurora 8b10b protocol files, while the GT wizard will be used to configure the physical layer. In the final step, the merging process is foreseen to provide the Aurora8b10b for GTY transceivers.
The Xilinx statement is that there is not really technical reason behind Aurora 8b10b not supporting GTY transceivers. But since GTY transceivers are meant to be targeted for higher line rate, they recommend to use the Aurora 64b66b for better throughput.
Finally, we figured out the method to implement the Aurora8b10b protocol in GTY transceivers. The FW design is based on the Xilinx example. And the method is presented below.
Aurora8b10b IP core configures two parts: link layer (protocol) and physical layer (GT part). In real, the Aurora8b10b core uses the GT wizard to configure the physical layer. Basically, two IP cores are being used to provide full setup. Thus, in order to implement the Aurora8b10b in GTY, the GT wizard needs to be run "manually" to configure the GTY transceivers.

Hub-Module Readout Data Distribution

FEX Slot 3   FEX Slot 4   • • • • •   FEX Slot 13   FEX Slot 14

FEX Logical Slot Numbers

Hub 1

ROD Mezz.   2   Hub Virtex FPGA

74   74

74 Channel 2 Way Fanout

Hub 2

Hub Virtex FPGA   2   ROD Mezz.

74   74

74 Channel 2 Way Fanout

Each line indicates six Fabric Interface Channels carrying FEX Slot Readout Data to the ROD's and to the Hub-Module's Virtex FPGAs

Hub to Hub Links use the Fabric Interface.

The FEX Slot Readout Data on each Fabric Interface Channel is received by an MGT Transceiver on both the ROD's and the Hub-Module's Virtex FPGAs.

Drw: 2   Rev. 15-Feb-2016

Figure 15. Illustration of FEX-Hub distribution of high-speed readout data signals.

Aurora Channel Partners

User Application

User Interface

Aurora Core

Aurora Lane 1

Aurora Channel

Aurora Lane n

Aurora Core

User Interface

User Application

User Data   8B/10B Encoded Data   User Data

X13009
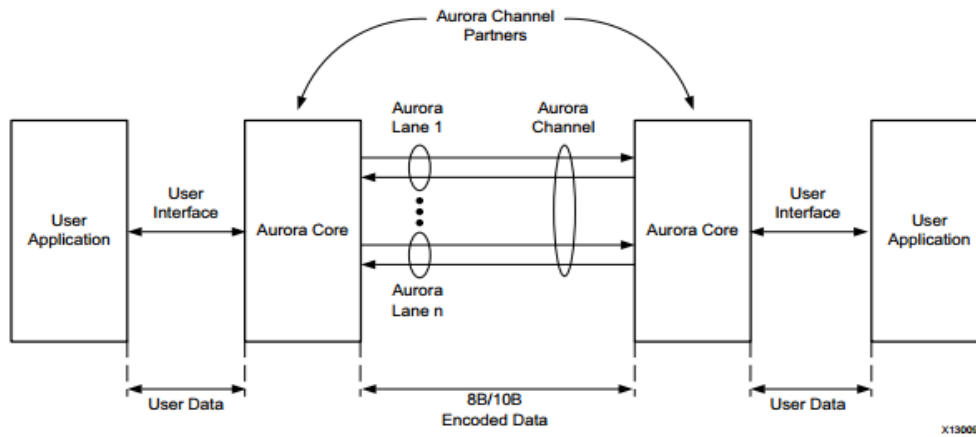
Figure 16. The Aurora 8b10b overview (Xilinx).

Aurora 8b10b implementation in GTY – recipe (1)

Standard Aurora8b10b IP core (GTH) configures the Physical and Link Layer:

Aurora 8b10b implementation in GTY – recipe (2)

Once the Aurora8b10b core is generated, open IP example Design
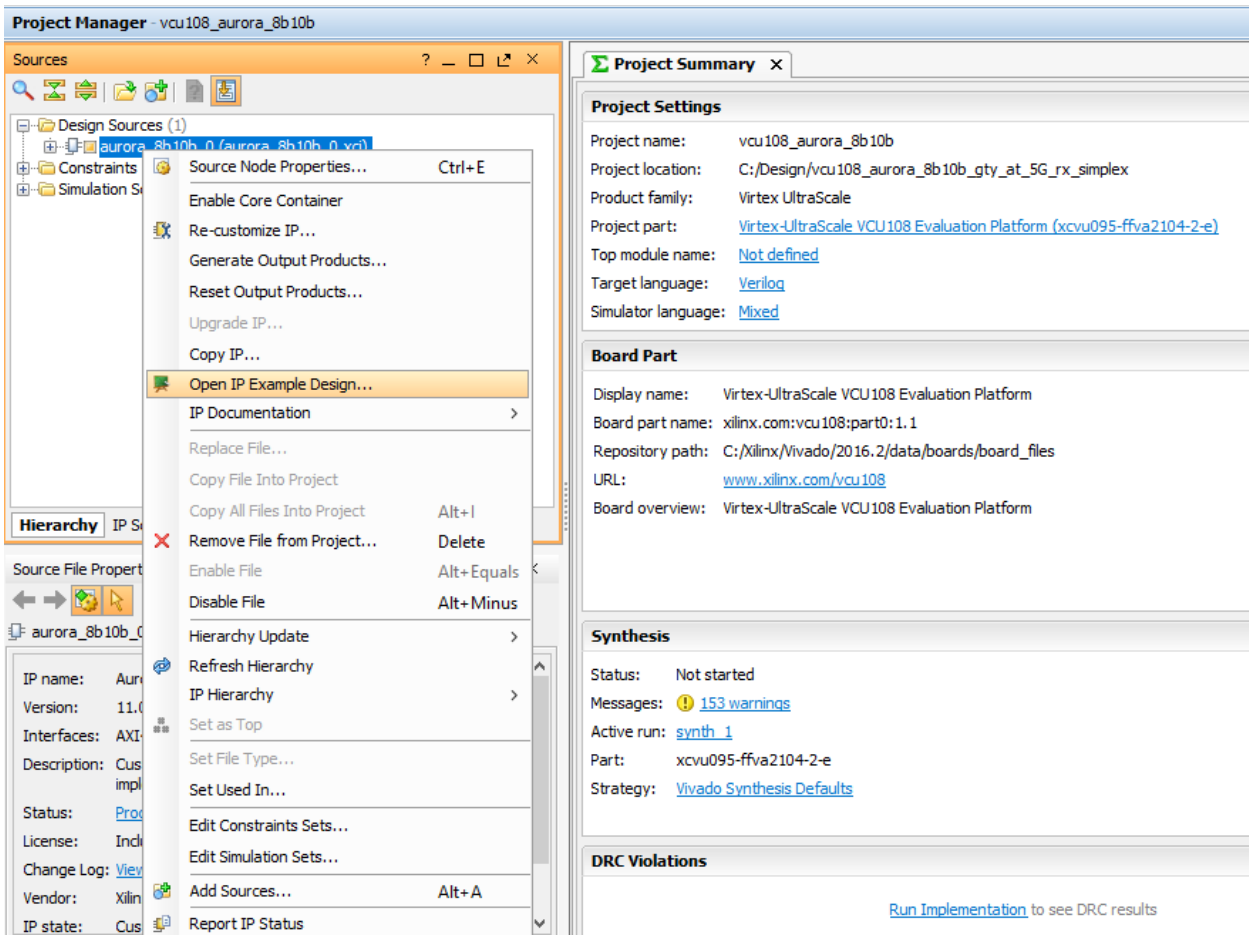
Aurora 8b10b implementation in GTY – recipe (3)

Open GT wizard to configure the GT part of the protocol:

Aurora 8b10b implementation in GTY – recipe (4)

Once the GT files are generated, replaced them with the files generated by Aurora8b10b IP core:
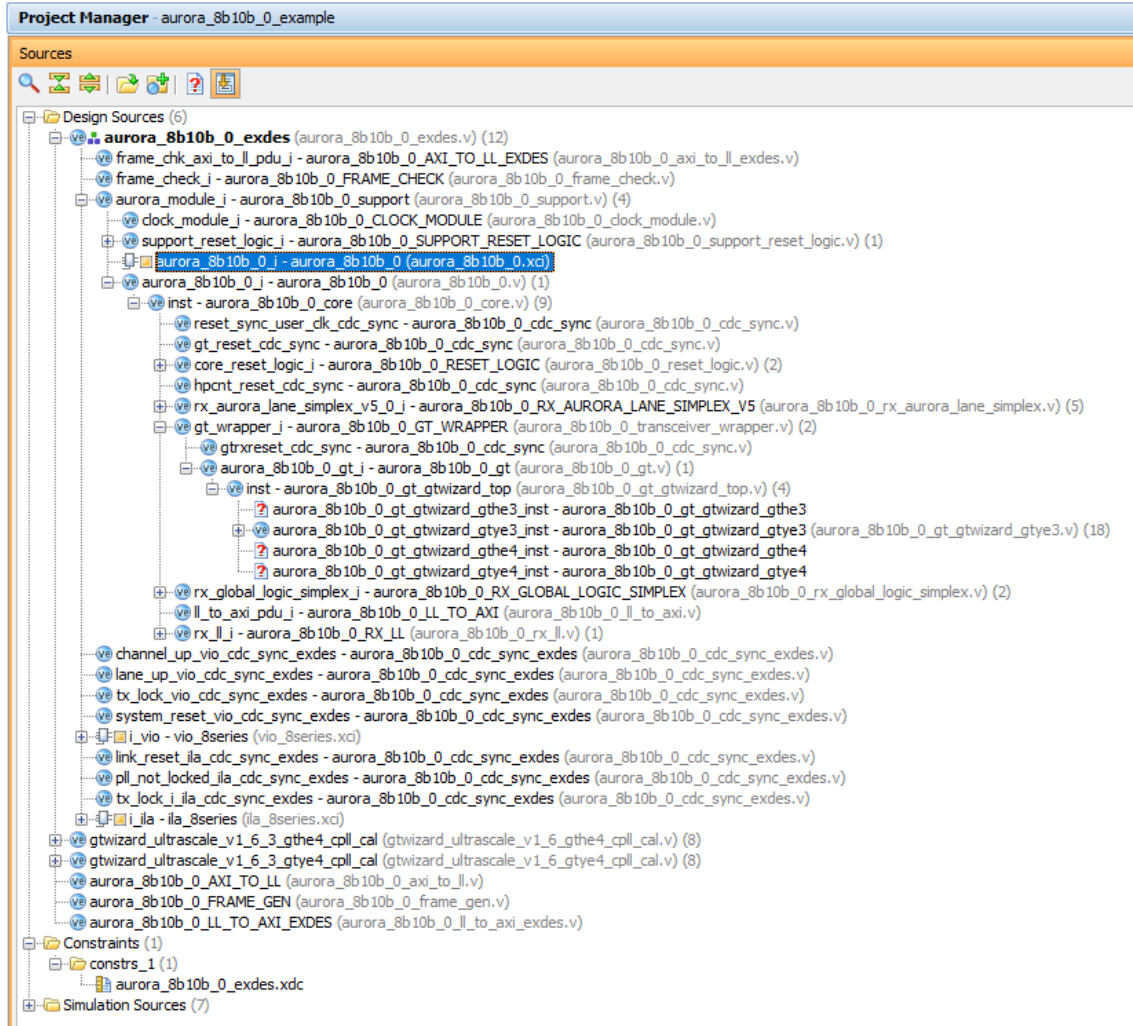
Aurora 8b10b implementation in GTY – recipe (5)

Visual inspection needed (design, xdc file, system clock, mgt referene clock):
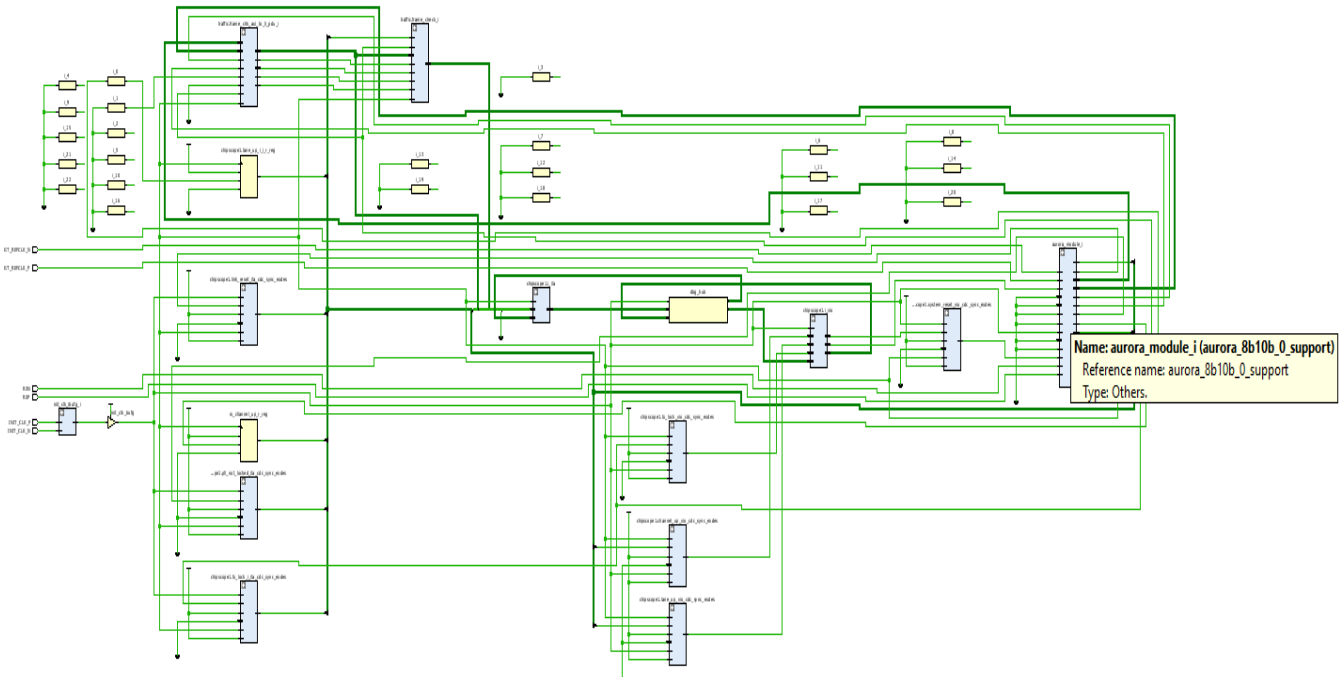
Aurora 8b10b implementation in GTY – test (1)

ZYNQ Board (GTX) and VCU108 Xilinx Dev Board setup

Image 1: This is the setup which is used to test the Aurora8b10b protocol in GTY transceivers. The ZynQ module (from PANDA experiment) was used to transmit the readout data (Aurora8b10b with GTX transceivers), and the Xilinx module dev board as the receiver (Aurora8b10b with GTY transceivers).

Aurora 8b10b implementation in GTY – test (2)

ZYNQ Board (GTX) and VCU108 Xilinx Dev Board: Links are up and stable

In order to control the Aurora8b10b protocol the VIO components is being used.

Aurora 8b10b implementation in GTY – test (3)

ZYNQ Board (GTX) and VCU108 Xilinx Dev Board: TX and RX data check OK.
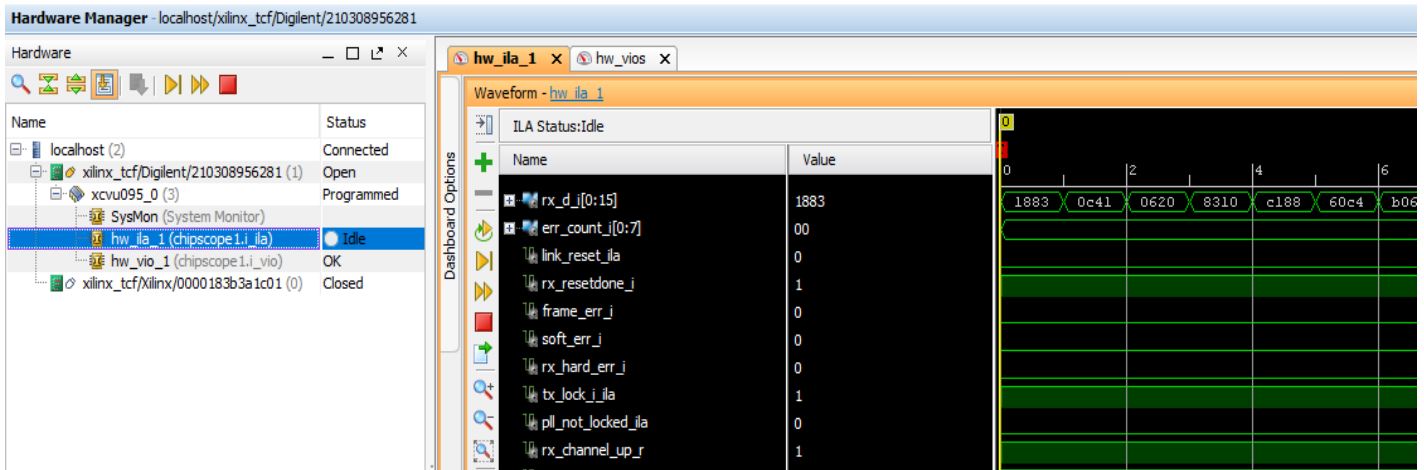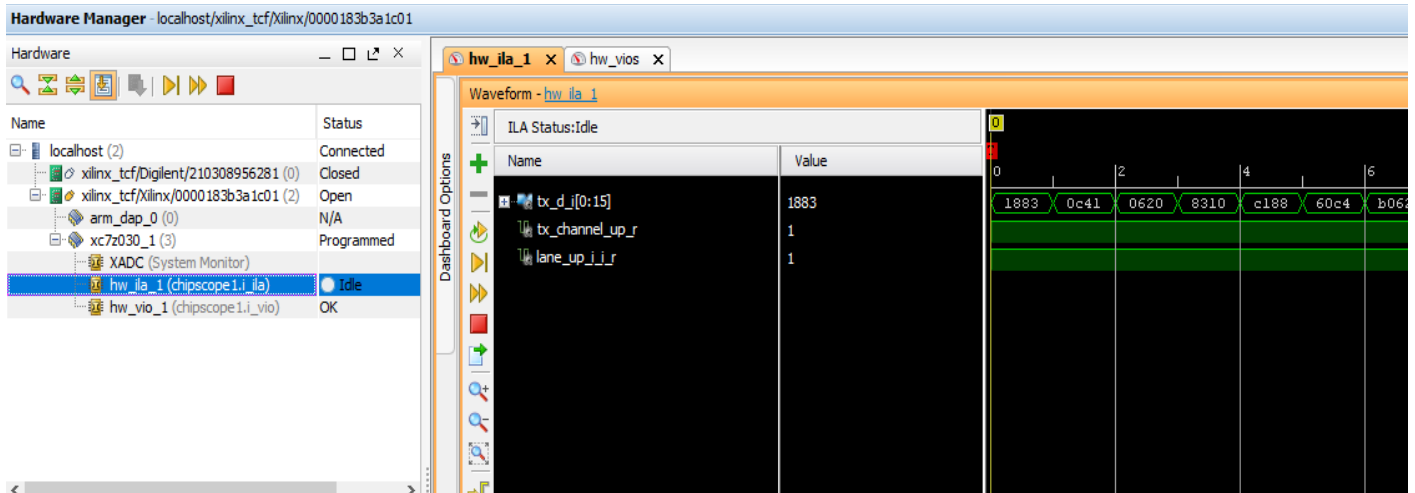
5.1 Aurora Line Mapping

The Aurora Line Mapping, 4 lines options (option A: 1,2,3,4 and option B: 3,4,5,6) and 6-lane mapping, respectively.

	1,2,3,4 option			3,4,5,6 option			6-Lane		
	lane	quad	rx MGT	lane	quad	rx MGT	lane	quad	rx MGT
Slot 3	1	126	1	1	126	1	1	126	1
	2	126	0	2	126	0	2	126	0
	3	125	3	3	125	3	3	125	3
	4	125	2	4	125	2	4	125	2
	5	125	1	5	125	1	5	125	1
	6	125	0	6	125	0	6	125	0
slot 4	1	124	3	1	124	3	1	124	3
	2	124	2	2	124	2	2	124	2
	3	128	1	3	128	1	3	128	1
	4	128	0	4	128	0	4	128	0
	5	127	3	5	127	3	5	127	3
	6	127	2	6	127	2	6	127	2
slot 5	1	127	1	1	127	1	1	127	1
	2	127	0	2	127	0	2	127	0
	3	126	3	3	126	3	3	126	3
	4	126	2	4	126	2	4	126	2
	5	130	1	5	130	1	5	130	1
	6	130	0	6	130	0	6	130	0
slot 6	1	129	3	1	129	3	1	129	3
	2	129	2	2	129	2	2	129	2
	3	129	1	3	129	1	3	129	1
	4	129	0	4	129	0	4	129	0
	5	128	3	5	128	3	5	128	3
	6	128	2	6	128	2	6	128	2
slot 7	1	132	3	1	132	3	1	132	3
	2	132	2	2	132	2	2	132	2
	3	132	1	3	132	1	3	132	1
	4	132	0	4	132	0	4	132	0
	5	131	1	5	131	1	5	131	1
	6	131	0	6	131	0	6	131	0
slot 8	1	130	3	1	130	3	1	130	3
	2	130	2	2	130	2	2	130	2
	3	130	1	3	130	1	3	130	1
	4	130	0	4	130	0	4	130	0
	5	131	3	5	131	3	5	131	3
	6	131	2	6	131	2	6	131	2
slot 9	1	133	1	1	133	1	1	133	1
	2	133	0	2	133	0	2	133	0
	3	232	0	3	232	0	3	232	0
	4	232	1	4	232	1	4	232	1
	5	232	2	5	232	2	5	232	2
	6	232	3	6	232	3	6	232	3
slot 10	1	233	0	1	233	0	1	233	0
	2	233	1	2	233	1	2	233	1
	3	233	2	3	233	2	3	233	2
	4	233	3	4	233	3	4	233	3
	5	229	2	5	229	2	5	229	2
	6	229	3	6	229	3	6	229	3
slot 11	1	230	0	1	230	0	1	230	0
	2	230	1	2	230	1	2	230	1
	3	230	2	3	230	2	3	230	2
	4	230	3	4	230	3	4	230	3
	5	231	0	5	231	0	5	231	0
	6	231	1	6	231	1	6	231	1
slot 12	1	227	2	1	227	2	1	227	2
	2	227	3	2	227	3	2	227	3
	3	228	0	3	228	0	3	228	0
	4	228	1	4	228	1	4	228	1
	5	228	2	5	228	2	5	228	2
	6	228	3	6	228	3	6	228	3
slot 13	1	229	0	1	229	0	1	229	0
	2	229	1	2	229	1	2	229	1
	3	225	0	3	225	0	3	225	0
	4	225	1	4	225	1	4	225	1
	5	225	2	5	225	2	5	225	2
	6	225	3	6	225	3	6	225	3
slot 14	1	226	0	1	226	0	1	226	0
	2	226	1	2	226	1	2	226	1
	3	226	2	3	226	2	3	226	2

5.2 Proposed Aurora, Readout_Control and Combined_TTC/Data initialization sequence

This is the proposed Aurora, Readout_Control and Combined_TTC/Data initialization scheme. This is preliminary model (Ed Flaherty), it will be discussed, and it might be changed.

1) Hub configuration
2) Hub Asserts PWR_CON1 to power-up the ROD
3) ROD asserts PWR_CON2 to indicate Power Good
4) After configuration, the ROD asserts PWR_CON3 indicating ready to run
4) ROD Power-Up timer is started
4) Hub Combined TTC/Data GT reset pulse is asserted
5) Hub Combined TTC/Data GT reset pulse is de-asserted
6) Aurora reset propagates from Readout_CTRL to COMBINED_TTC
7) ROD Power-Up timer is de-asserted.
7) All Aurora channel reset timers in Hub, ROD and FEX's start simultaneously
8) All Aurora channel reset timers in Hub, ROD and FEX's de-assert simultaneously
9) All Aurora receivers should assert Channel-UP simultaneously

Hub ROD PWR-UP handshake signals
PWR_CON1: power-on from Hub
PWR_CON2: power-good from ROD
PWR_CON3: rod ready to run
PWR_CON4: reserved – ether direction

Figure 17. Sequence of events for the Aurora and Combined_TTC/Data initialization.

Figure 18. Illustration of Hub/ROD/FEX Aurora initialization sequence (Ed Flahery's Proposal).
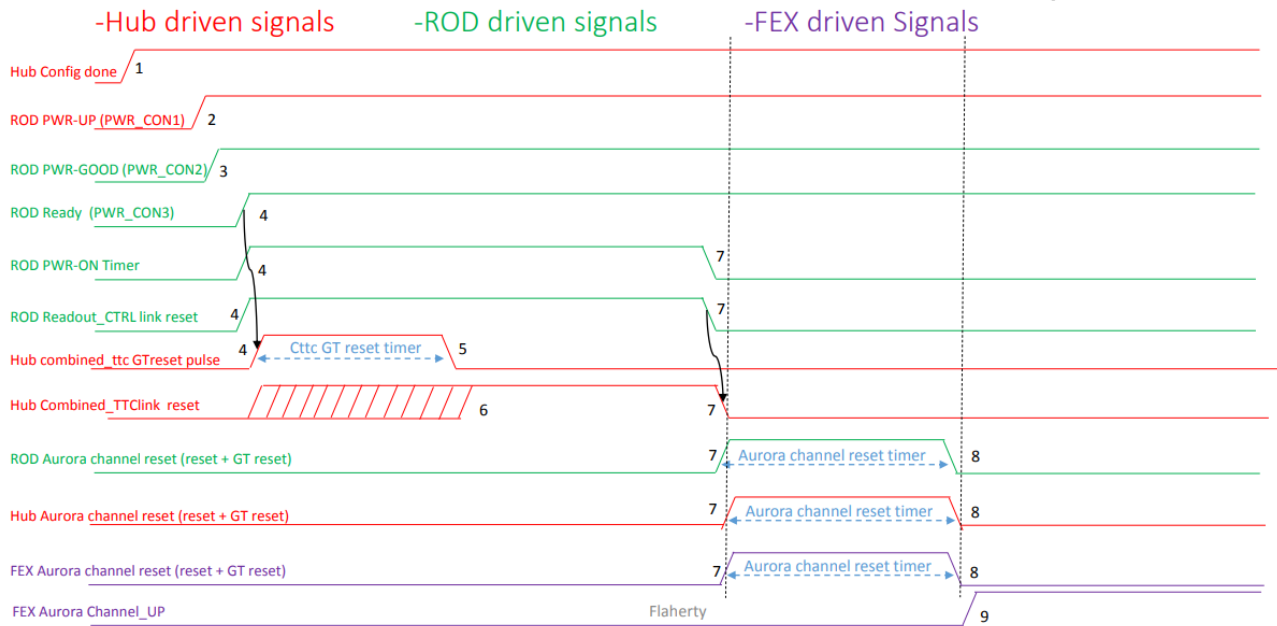
- **ROD Power-UP timer**
  - Pulse Width not very critical – just "long"
  - Currently 32-bit countdown starting from X"3FFFFFFF" @125MHz clock = 8.6 seconds

- **Aurora Channel Reset Timer: used at all Rx & Tx interfaces**
  - 4 outputs: Tx_reset, Tx_GTReset, Rx_reset, Rx_GTReset
  - Timing critical for all 4 outputs
    - All aurora interfaces should start at the same time (based upon combined_TTC link reset)
  - currently running from 125MHz clock
    - May be better to change to 40MHz clock in future

- **Hub GTReset Pulse**
  - Pulse Width not very critical, but must be significantly shorter than the ROD Power-UP pulse
    - 1 sec ?

Adding new signal ("backplane reset") from ROD to Hub – proposal.

A new signal would be treated as a "backplane reset". The undefined "PWR_CON4" can be used for this, but there are other spares that could also be used as well. The proposal assumes that it may be useful for the ROD to have the ability to issue a reset to the Hub for the Readout_Control and Combined_TTC/Data links. So for example; on the initial power-up, the ROD could then completely control the start-up of the Aurora interfaces including the paths carrying the link reset signals. This would move the timer presented in Figure 18 from the Hub to the ROD. While running, if there is a problem in this path, the ROD will not currently be able to restart any of the Aurora links. If it attempts to restart a link, and the link does not come up, it's then stuck. By having this signal, it could attempt to restart the whole backplane (either automatically, or by IPBus command). Also, as a receiver of the Combined_TTC and transmitter of Readout_ctrl, the ROD may be in a position to detect a problem in this link when the Hub can not. Obviously, the Hub can also reset this link if it determines that there is a problem.

6.0 IBERT test.

In order to test the specific MGT channels the IBERT FW was provided for the FTM, ROD and HUB modules. The table describes the list of tests and results.

No	Test description	Result
1	4 MiniPOD Receiver MGT channels	No errors
2	6 lanes of MGT data from all 12 of the FEX slot	No errors
3	Combined_TTC/Data to the 12 FEX slots	No errors
4	Combined_TTC/Data to the ROD on This HUB	No errors
5	Combined_TTC/Data Data to the Other HUB	No errors
6	Combined_TTC/Data Data that was sent out by the Other HUB	No errors
7	Readout Control Data that was sent out by the ROD on This HUB	No errors
8	HUB sends out two lanes of Readout data to the Other HUB	No errors
9	HUB receives two lanes of Readout data from the Other HUB	No errors
10	FPGA on This HUB sends one lane of its readout data to the ROD on This HUB	No errors

6.1 Combined_TTC/ Data to the FEX 3; test@6.4Gbps:

6.2 Combined_TTC/Data to the FEX 4; test@6.4Gbps:

6.3 Combined_TTC/Data to the FEX 5; test@6.4Gbps:

6.4 Combined_TTC/Data to the FEX 6; test@6.4Gbps:

6.5 Combined_TTC/Data to the FEX 7; test@6.4Gbps:

6.6 Combined_TTC/Data to the FEX 8; test@6.4Gbps:

6.7 Combined_TTC/Data to the FEX 9; test@6.4Gbps:

6.8 Combined_TTC/Data to the FEX 10; test@6.4Gbps:

6.9 Combined_TTC/Data to the FEX 11; test@6.4Gbps:

7.0 Combined_TTC/Data to the FEX 12; test@6.4Gbps:

7.1 Combined_TTC/Data to the FEX 13; test@6.4Gbps:

7.2 Combined_TTC/Data to the FEX 14; test@6.4Gbps:

7.3 IBERT Slot no 3 + extra MGT channel tests, tests@6.4Gbps covered the area:

- 4 MiniPOD Receiver MGT channels
- 6 lanes of MGT data from the FEX slot 3
- Combined_TTC/Data to the ROD on This HUB
- Combined_TTC/Data Data to the Other HUB
- Combined_TTC/Data Data that was sent out by the Other HUB
- Readout Control Data that was sent out by the ROD on This HUB
- HUB sends out two lanes of Readout data to the Other HUB
- HUB receives two lanes of Readout data from the Other HUB
- FPGA on This HUB sends one lane of its readout data to the ROD on This HUB

NOTE: The errors were injected to check if the link is alive

7.4 IBERT (eye diagram); FEX slot 3, test@10.26Gbps:

7.5 IBERT (eye diagram); MiniPods, test@9.6Gbps:
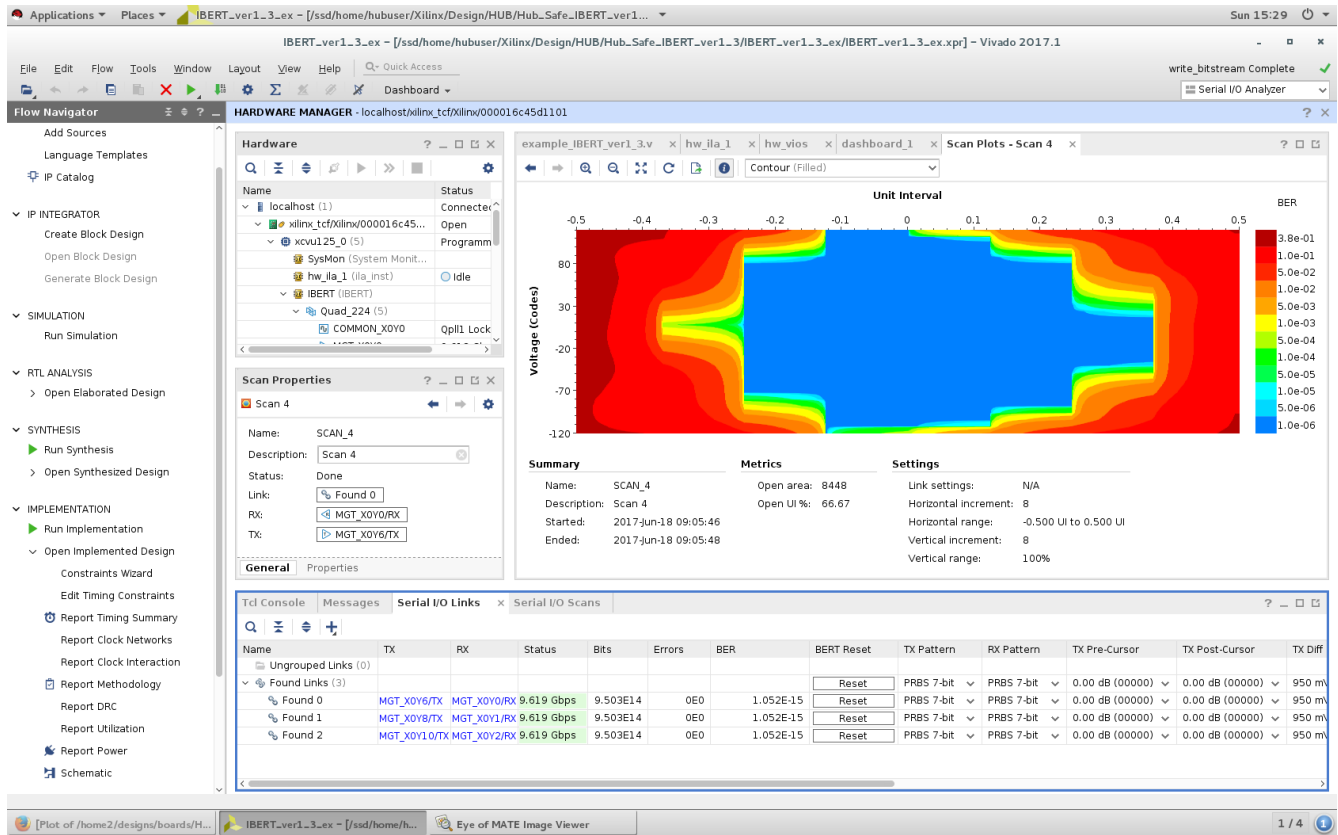
7.6 IBERT Slot no 4, this test@6.4Gbps covered the area:

- 6 lanes of MGT data from the FEX slot 4

NOTE: The errors were injected to check if the link is alive

7.7 IBERT (eye diagram); FEX slot 4, test@6.4 Gbps:

7.8 IBERT Slot no 5, this test@6.4Gbps covered the area:

- 6 lanes of MGT data from the FEX slot 5

NOTE: The errors were injected to check if the link is alive

7.9 IBERT Slot no 6, this test@6.4Gbps covered the area:

- 6 lanes of MGT data from the FEX slot 6

8.0 IBERT Slot no 7, this test@6.4Gbps covered the area:

- 6 lanes of MGT data from the FEX slot 7

8.1 IBERT Slot no 8, this test@6.4Gbps covered the area:

- 6 lanes of MGT data from the FEX slot 8

NOTE: The errors were injected to check if the link is alive

8.2 IBERT Slot no 9, this test@6.4Gbps covered the area:

- 6 lanes of MGT data from the FEX slot 9

NOTE: The errors were injected to check if the link is alive

8.3 IBERT Slot no 10, this test@6.4Gbps covered the area:

- 6 lanes of MGT data from the FEX slot 10

NOTE: The errors were injected to check if the link is alive

8.4 IBERT Slot no 11, this test@6.4Gbps covered the area:

- 6 lanes of MGT data from the FEX slot 11

NOTE: The errors were injected to check if the link is alive

8.5 IBERT Slot no 12, this test@6.4Gbps covered the area:

- 6 lanes of MGT data from the FEX slot 12

NOTE: The errors were injected to check if the link is alive

8.6 IBERT Slot no 13, this test@6.4Gbps covered the area:

- 6 lanes of MGT data from the FEX slot 13

NOTE: The errors were injected to check if the link is alive

8.7 IBERT Slot no 14, this test@6.4Gbps covered the area:

- 6 lanes of MGT data from the FEX slot 14

NOTE: The errors were injected to check if the link is alive

9.0 HUB FW development plans (this list is in progress)

Config ver	HUB FW features	Status
1	Readout Control (Readout_Ctrl) data from the ROD; Combined_TTC/Data to This ROD; Combined_TTC/Data to FEX 3.	Done
2	Readout Control (Readout_Ctrl) data from the ROD; Combined_TTC/Data to This ROD; Combined_TTC/Data to FEX 3, Readout data from FEX 3 (Aurora8b10b)	In progress
3	Readout Control (Readout_Ctrl) data from the ROD; Combined_TTC/Data to This ROD, Combined to FEX 3, Readout data from FEX 3 (Aurora8b10b);	waiting
4	Readout Control (Readout_Ctrl) data from the ROD; Combined_TTC/Data to This ROD; Combined_TTC/Data to FEX 3, Readout data from FEX 3 (Aurora 8b10b); Readout data from FEX 13 (Aurora8b10b)	waiting
5	Readout Control (Readout_Ctrl) data from the ROD; Combined_TTC/Data to This ROD; Combined_TTC/data to FEX 3, Readout data from FEX 3 (Aurora 8b10b); Readout data from FEX 13 (Aurora8b10b); Combined_TTC/Data to Other HUB	waiting
6	Readout Control (Readout_Ctrl) data from the ROD; Combined_TTC/Data to This ROD; Combined_TTC/data to FEX 3, Readout data from FEX 3 (Aurora 8b10b); Readout data from FEX 13 (Aurora8b10b); Combined_TTC/Data to Other HUB; Readout AL_0 data to This ROD; Readout AL_1 data to This ROD	waiting
7	Readout Control (Readout_Ctrl) data from the ROD; Combined_TTC/Data to This ROD; Combined_TTC/data to FEX 3, Readout data from FEX 3 (Aurora 8b10b); Readout data from FEX 13 (Aurora8b10b); Combined_TTC/Data to Other HUB; Readout AL_0 data to This ROD; Readout AL_1 data to This ROD; Readout AL_0 data to Other HUB; Readout AL_1 data to Other HUB;	waiting
8	Readout Control (Readout_Ctrl) data from the ROD; Combined_TTC/Data to This ROD; Combined_TTC/data all FEXs, Readout data from all FEXs (Aurora 8b10b); Combined_TTC/Data to Other HUB; Readout AL_0 data to This ROD; Readout AL_1 data to This ROD; Readout AL_0 data to Other HUB; Readout AL_1 data to Other HUB;	waiting
9	Readout Control (Readout_Ctrl) data from the ROD; Combined_TTC/Data to This ROD; Combined_TTC/data all FEXs, Readout data from all FEXs (Aurora 8b10b); Combined_TTC/Data to Other HUB; Readout AL_0 data to This ROD; Readout AL_1 data to This ROD; Readout AL_0 data to Other HUB; Readout AL_1 data to Other HUB; IPBus component	waiting

References:

1. Technical Specification, ALAS Level-1 Calorimeter Trigger Upgrade, FEX System Switch Module (FEX Hub) Prototype, Dan Edmunds,Yuri Ermoline, Brian Ferguson, Wade Fisher, Philippe Laurens, Spencer Lee, Pawel Plucinski; 2018

2. Specification for Readout Control & Combined TTC/Data Serial links in L1Calo, Ian Brawn, Ed Flaherty, Weiming Qian, David Sankey, Saeed Taghavi; 2018.

3. Aurora 8B/10B v11.0, LogiCORE IP Product Guide, Vivado Design Suite, PG046 October 5, 2016.

4. Virtual Input/Output v3.0, LogiCORE IP Product Guide, Vivado Design Suite, PG159 October 4, 2017.

5. Integrated Logic Analyzer v6.1 , LogiCORE IP Product Guide, Vivado Design Suite, PG172 April 6, 2016.

6. Vivado Design Suite User Guide, Getting Started, UG910 (v2017.1) April 5, 2017

7. UltraScale Architecture System Monitor, User Guide, UG580 (v1.8) June 15, 2017.