FERMILAB-TM-2553-E-PPD

Data Formatter Design Specification

Jamieson Olsen[1], Tiehui Ted Liu[1], Yasuyuki Okumura[1,2]

[1]Fermi National Accelerator Laboratory, Batavia, Illinois 60510, USA
[2]University of Chicago, Chicago, Illinois 60637, USA

March 20, 2013

Abstract

The Fast TracKer (FTK) processor is an upgrade which adds a hardware-based tracking system to the ATLAS level-2 trigger system for high luminosity operation of the Large Hadron Collider (LHC), aimed for instantaneous luminosity of 3×10^{34} cm^{-2}s^{-1} with 75 overlapping events per bunch crossing of 40 MHz. The FTK system will reconstruct tracks using data from the inner Pixel and SCT silicon detector at input rate up to 100 kHz of level-1 trigger rate. In order to sustain the high input rate it is necessary to organize FTK as a set of independent engines, each working on a different region of the silicon tracker. The FTK system requires a layer of hardware to remap the ATLAS inner detector geometry to match the FTK η-ϕ towers. This hardware layer is the Data Formatter system. The Data Formatter system also performs clustering and data sharing for overlapping of neighboring η-ϕ towers to avoid inefficiency due to the finite size of the beam luminous regions in z coordinate and variety curvature of tracks. Based on the current design requirement and the need for future expansion capability, a full mesh Advanced Telecom Computing Architecture (ATCA) backplane interconnect is found to be a natural fit for the Data Formatter design. Our baseline design also works well as a general purpose FPGA-based processor board. The Data Formatter may prove useful in scalable systems where highly flexible, non-blocking, high bandwidth board to board communication is required. This specification note is aimed to present an overview of the current Data Formatter system design and show the data-driven bandwidth requirement study to support the current design.

Contents

List of Figures

List of Tables

1 Introduction

The Large Hadron Collider (LHC) at CERN will extend the frontiers of particle physics with its unprecedented high energy and luminosity. Inside the LHC, bunches with more than 10^{11} protons will collide every 25 ns to provide 14 TeV proton-proton collisions at a design luminosity above 1×10^{34} cm^{-2}s^{-1}. The ATLAS (A Toroidal LHC ApparatuS) detector is a general purpose detector located at one of the collision points of the LHC to search new and very rare physics phenomena.

Since it is essential to increase the luminosity in order to conduct this type of study at ATLAS, improvement of the trigger system is necessary to enable reasonable reduction of the background online. The Fast TracKer (FTK) [3] is a trigger upgrade program that permits quick track reconstruction to provide the full track lists to the Level-2 algorithm with instantaneous luminosity of 3×10^{34} cm^{-2}s^{-1} at a level-1 trigger rate up to 100 kHz. The FTK system is a massively parallel hardware-based processing engine for fast tracking based on inner Pixel and SCT silicon hits. In order to sustain the high input rate, it is necessary to organize FTK as a set of independent engines, each working on a different region of the silicon detector. Each FTK η-ϕ tower region has its own "core processor" for a total 64 engines working independently. This segmentation generates some inefficiency at region boundaries due to the finite size of the beam's luminous region and the variety pf curvature for the low-momentum charged particle that must be removed by allowing a overlap region at the boundary.

The FTK system includes a Data Formatter system to remap the ATLAS inner detector geometry to match the FTK η-ϕ towers. The Data Formatter system also performs hit-clustering and data sharing in overlap regions. Based on the current design requirements and the need for future expansion capabilities, a full mesh Advanced Telecom Computing Architecture (ATCA) backplane interconnect is found to be a natural solution for the Data Formatter design. We present the current design specification of the Data Formatter system, as well as the data-driven bandwidth requirement estimation study to support the design. This specification note consists of the following sections:

- ATLAS Inner Detector system (Section 2)
- Overview of Fast Tracker and concept of the Data Formatter system (Section 3)
- System-level design of the Data Formatter System (Section 4)
- Functions and design of the hardware components (Section 5, 6, 7)
- Data-driven bandwidth requirement estimation study (Section 8).

The early note of the Data Formatter hardware specification [1] is available to offer an historical look back at the early Data Formatter design process.

2 The Inner Tracking Detector

2.1 Layout of Pixel and SCT detectors

Figure 1 shows ATLAS Pixel and SCT silicon detectors. The Pixel detector is composed of three barrel layers (radius 50 to 123 mm) and six end-cap disks (at $z = 495$ mm to 650 mm). All of the 1,744 pixel modules are identical and consist of 46,080 readout channels (pixel sensor size is 400 μm $\times$ 50 μm with thickness of 250 μm) per module for a total of 80.4 million readout channels.

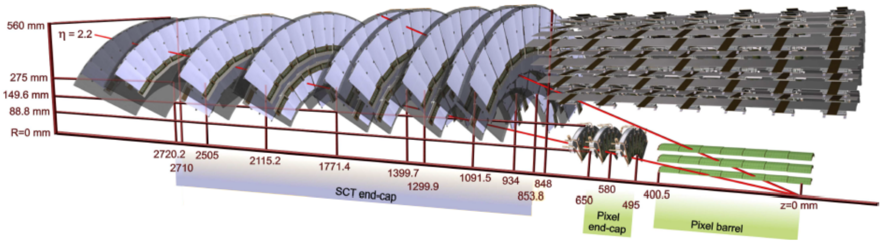


Figure 1: The ATLAS inner Pixel and SCT detector modules.

The SCT detector is composed of four barrel layers (radius 299 to 514 mm) and six end-cap disks (at $z = 853$ to 2720 mm) to measure the charged particle positions with two stereo silicon strip layers. The SCT barrel module consists of 768 silicon strips (63 mm × 80 mum with thickness of 285 μm). Four SCT barrel layers are constructed from 2,112 sensors mounted to "stave" support structures. SCT end-cap modules are trapezoidal shaped and come in three varieties: inner modules measure 45 mm, 55 mm, 61 mm (inner width, outer width, length); middle modules measure 55 mm, 75 mm, 119 mm; and outer modules measure 56 mm, 72 mm, 123 mm. A total of 1,976 modules are used to construct the SCT end-cap disks.

2.2 Readout Chains of Pixel and SCT detector

Inner detector front end electronics are implemented in radiation-hardened application-specific integrated circuit (ASIC) devices which are mounted on the modules. These front end ASICs interface the silicon sensors and incorporate analog circuitry to amplify the signals and compare the signal level against a programmable threshold. Pixel ASICs store the "hit" pixel coordinate as well as time stamps (BCID, L1ID) and amplitude (time over threshold) in a buffer, which is read out following L1 trigger accept signal. The SCT ASICs store the "hit" strip address as well as time stamps and 3-bit timing information for previous, current, and next bunch crossing hit information. The data format and the contained information are summarized in Appendix I.

Chains of front end ASICs are connected over fiber optic links to the Readout Driver (ROD) electronics, which are located off-detector in USA15 in the ATLAS cavern. RODs receive serialized data from the detector after a L1 trigger accept and are responsible for de-serializing the data, error checking, local event building and data monitoring tasks. Each ROD services between 6 and 26 fiber input from pixel modules or up to 96 fiber inputs from SCT modules. Table 1 summarizes the number of modules and readout links for pixel and SCT detectors. The current system makes use of 222 fiber optic SLINK [16] cables which send the data to the Read-Out System (ROS) located downstream in the ATLAS DAQ system. Appendix A summarizes the details of the partitioning of the Pixel and SCT readout (module-ROD links). The number of readout links is planned to be increased to maintain the bandwidth for the higher hit occupancy after scheduled shutdown from 2013.

2.3 Expansion with Insertable B-Layer (IBL)

Plans are currently underway to install an "insertable B-layer" (IBL) pixel detector during the scheduled shutdown from 2013. The IBL consists of additional pixel modules arranged in a barrel layer near the beam pipe at a radius of approximately 34 mm. A total of 224 modules

Sub-detector	Partition	Modules	ROLs
Pixel	Barrel 0	286	44
	Barrel 1	494	38
	Barrel 2	676	26
	End-Cap A	144	12
	End-Cap C	144	12
SCT	Barrels A	1056	22
	Barrels C	1056	22
	End-Cap A	988	23
	End-Cap C	988	23

Table 1: Data Formatter input readout links

will be mounted to 14 stave structures [1].

3 The Fast Tracker and Data Formatter System

3.1 Overview of the Fast TracKer system and FTK tower structure

The Fast TracKer (FTK) system finds and fits tracks using the inner detector silicon layers for every event that passes the level-1 trigger decision at rates up to 100 kHz. The FTK receives the SLINK outputs from Pixel and SCT RODs, which are duplicated at the output stage of ROD modules.

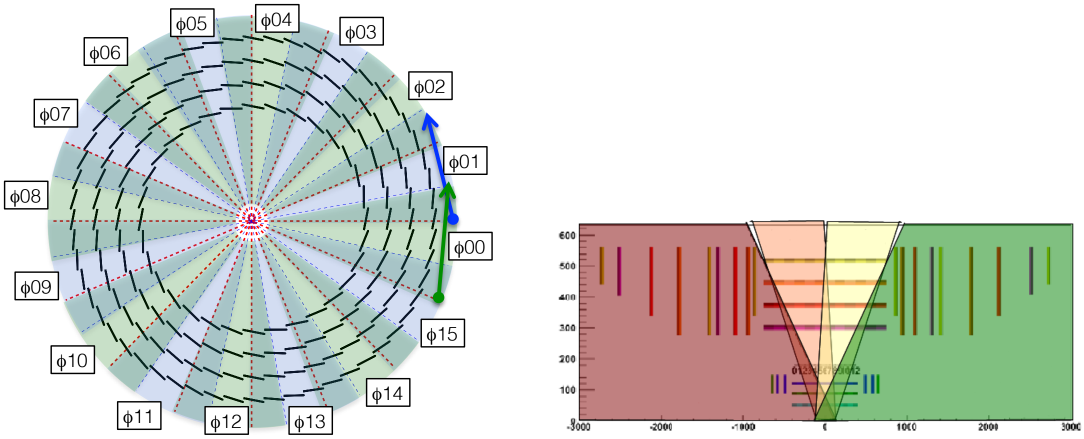
In order to sustain a 100 kHz level-1 trigger rate, FTK is organized as a set of independent engines arranged in η-ϕ towers. The first step is to divide the detector into 16 sectors in ϕ coordinate as shown in Figure 2(a). The sectors have overlapping regions to avoid intrinsic inefficiency due to variation curvature of charged tracks at the boundary regions. The second step is to divide the η range into four intervals: "C-Side Endcap", "C-Side Barrel", "A-Side Barrel" and "A-Side Endcap" as shown in Figure 2(b). The overlap in η takes into account the size of finite size of the beam's luminous region in z.

Each FTK η-ϕ tower core processor consists of a track finder stage followed by two sequential stages of track fitters. The track finder uses pattern recognition associative memories to quickly find track candidates in coarse resolution roads. Roads which match the selection criteria are then analyzed in the track fitting stages using full resolution silicon hits. The extracted track parameters are reported to the level-2 trigger algorithm. The track finder and the first stage of track fitter make use of subset of silicon layers, while all layers are used in the final fitting to maximize fake track rejection. Using two track fitting stages results in a good balance between resource usage and tracking performance. Appendix A summarizes the details of the partitioning of the FTK tower structure.

3.2 The Data Formatter System

The FTK system runs the tracking algorithms using Pixel and SCT silicon hits which are sent from the RODs on SLINK fibers. Before the Pixel and SCT hits are used by the FTK core crates they must be remapped and repackaged into symmetric η-ϕ towers. Data duplication is required in the overlap regions at the tower boundaries. The FTK system proposal includes a hardware-based Data Formatter system at the input of the FTK system to remap input data to match the FTK η-ϕ tower structure. Considering the high level-1 trigger rate up

[1] 14 RODs will be used to read out the module data, and 112 ROLs will be implemented where 56 links are used for ROS, and the other 56 for FTK.



(a)

(b)

Figure 2: (a) The 16 FTK ϕ sectors. Note overlapping is implemented at the boundary to avoid inefficiency due to finite curvature of low momentum tracks. As example the coverage of the $\phi00$ sector and $\phi01$ sector are shown with green and blue arrows respectively in the figure. (b) The Four FTK η regions. Note the significant overlap in the high occupancy central barrel regions.

to 100 kHz and high hit occupancy of the inner detectors, the Data Formatter system is required to support massive bandwidth in the system. Recognizing that the module-ROD mapping will likely change over time, it is critical to select a backplane technology that is robust and flexible enough to handle upstream hardware configuration changes and allow for future expansion.

The Data Formatter system is required to satisfy following design requirements at the maximum level-1 trigger rate of 100 kHz and with the maximum luminosity of $\mathcal{L} = 3 \times 10^{34}\text{cm}^{-2}\text{s}^{-1}$:

- Receive data from the RODs
- Remap the ATLAS inner detector geometry to match the FTK η-ϕ towers
- Send data to the 64 downstream FTK processors
- Maintain flexibility to handle upstream hardware configuration changes

In addition to the above requirements the Data Formatter system also performs clustering of Pixel and SCT hits. The conceptual diagram of the Data Formatter system is shown in Figure 3.

We have extensively analyzed and simulated data flow in the Data Formatter system. Based on the performance criteria and design requirements we conclude that a full mesh Advanced Telecom Computing Architecture (ATCA) backplane is natural fit for the Data Formatter design.

4 The Data Formatter System

The full mesh ATCA backplane is central to the Data Formatter system design. In the full mesh ATCA backplane all boards in the shelf are directly connected using multiple high speed serial links. Since all backplane links are point-to-point there is no need for bus arbitration, maximizing data transfer efficiency and minimizing system latency. Field programmable gate

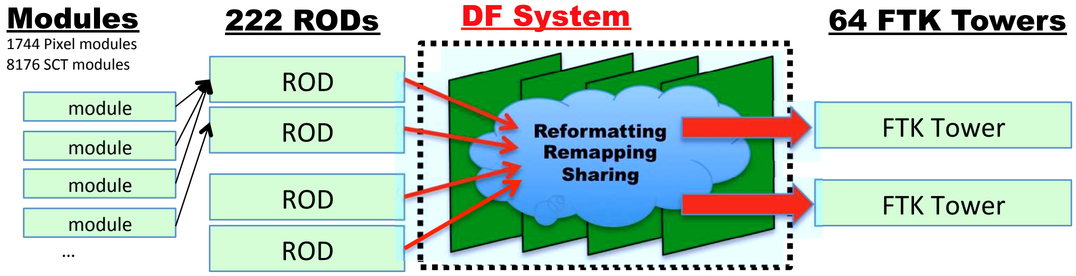


Figure 3: Data from 222 RODs to the Data Formatter and are sent to downstream 64 FTK towers after clustering, remapping, reformatting, and sharing.

array (FPGA) devices are used for the Data Formatter "processors". One FPGA is assigned for each FTK η-ϕ tower, and 64 FPGAs are used in total. Each of the 64 Data Formatter FPGAs receive data from upstream RODs, share data over the ATCA backplane and inter-shelf links, and finally send data downstream to the FTK core crates. In following sections we introduce the conceptual design of the Data Formatter boards and how data flow influences the overall system level design.

4.1 Conceptual Board-Level Design

Figure 4 shows the conceptual design of the Data Formatter board. Fiber links from the Pixel and SCT RODs are received on mezzanine cards. The mezzanine cards contain FPGAs which are used to run cluster finding algorithms. Clusters from the mezzanine card are fed into two large FPGAs on the Data Formatter board (yellow lines). These two FPGAs share data over an on-board local bus (orange line), the full mesh backplane (pink line), and bi-directional optical fiber links driven by the rear transition module (RTM) (green line). Fiber optic transceivers on the RTM are used to send data downstream to the FTK core processor crates as well (blue line).

4.2 Data Sharing Paths

The Data Formatter system is equipped with three types of internal communication paths to connect the 64 FPGAs in the system. The first path is a local bus which connects the FPGAs on the same Data Formatter board (Figure 5(a)). The second path is the full mesh fabric interface in the ATCA backplane. All FPGAs in the shelf are directly connected over high speed serial links in the full mesh backplane (see Figure 5(b)). Data Formatter boards are organized in order to minimize data sharing across the backplane. There are, however, cases where FPGAs must share data across shelf boundaries. Inter-shelf communication occurs using fiber optic transceivers on the RTM (Figure 5(c)). Using the three types of communication links each FPGA has a path to all other FPGAs in the system.

4.3 System-Level Design of Data Formatter System

The Data Formatter system will consist of four 14-slot ATCA shelves, 32 main boards, 64 FPGAs, 32 RTMs, and up to 128 mezzanine cards. Data Formatter FPGAs use integrated multi-gigabit serial transceivers to interface to the ATCA backplane and fiber transceivers on the RTM. The FPGA serial transceivers are rated for speeds up to 10 Gb/s. Figure 6 illustrates the substantial connectivity achievable with the Data Formatter system. Each line

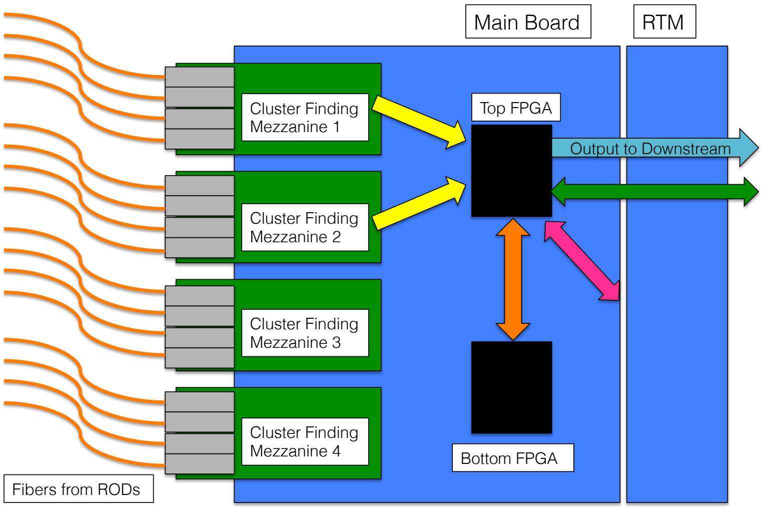
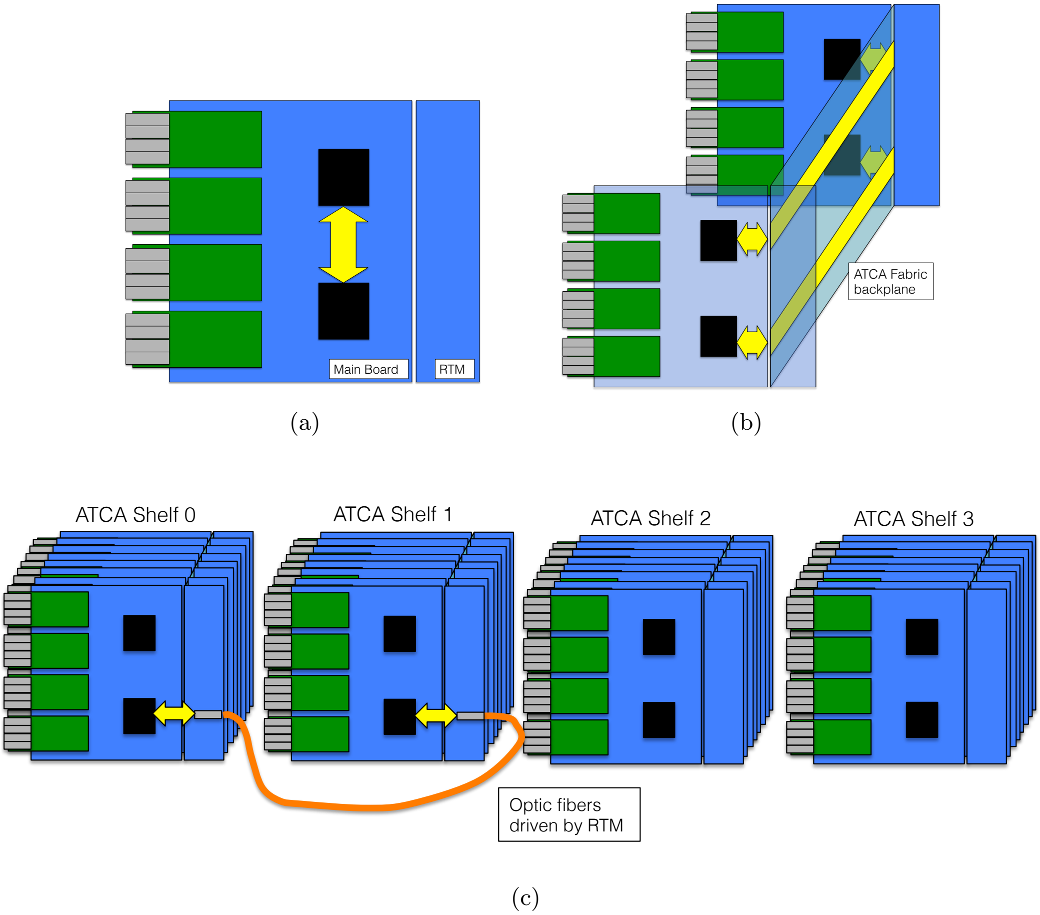

Main Board
RTM
Cluster Finding Mezzanine 1
Cluster Finding Mezzanine 2
Cluster Finding Mezzanine 3
Cluster Finding Mezzanine 4
Top FPGA
Output to Downstream
Bottom FPGA
Fibers from RODs

Figure 4: Data Formatter conceptual board design. Each Data Formatter board supports up to four cluster finder mezzanine cards and two FPGAs. The RTM board contains fiber optic transceivers for sending data downstream to the FTK core processor crates.

in the diagram represents a serial link up to 10 Gb/s. Inter-shelf links shown in this figure accurately reflect the results of our bandwidth study (refer to Section 8.1.2).

4.4 Input and Output Flexibility

The Data Formatter system will support up to 128 mezzanine cards. Each mezzanine card will support up to four SFP+ optical transceivers for a total of 512 input links. This is significantly more than the current 222 optical fibers from Pixel and SCT RODs which leaves room for future expansion. If more than 512 input links are required then additional Data Formatter boards and mezzanine boards may be installed. RTM boards support up to eight QSFP+ and six SFP+ optical transceivers; half of the transceivers are directly connected to the Top FPGA and the other half are directly connected to the Bottom FPGA. Optical or copper serial transceiver modules may be installed in any SFP+ or QSFP+ location on the RTM.



(a)

(b)

(c)

Figure 5: Three types of communication in the Data Formatter. (a) A local bus connects the two FPGAs on the board. (b) All FPGAs in the shelf are directly connected using the ATCA backplane fabric interface. (c) Inter-Shelf communication uses RTM transceivers.

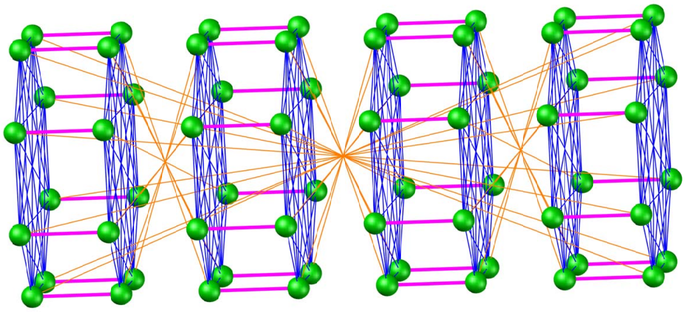


Figure 6: A 3D representation of FPGA interconnects in the Data Formatter system. 64 FPGAs (green) are connected through the ATCA backplane Fabric Interface (blue), local buses (purple) and inter-shelf links (orange). Each FPGA uses one inter-shelf link. This diagram assumes 4 shelves with 8 Data Formatter boards per shelf. For this picture, the initial fiber connection introduced in Section 8.1.2 is shown. An animated version is available [13].

5 Data Formatter Board

5.1 Design Overview

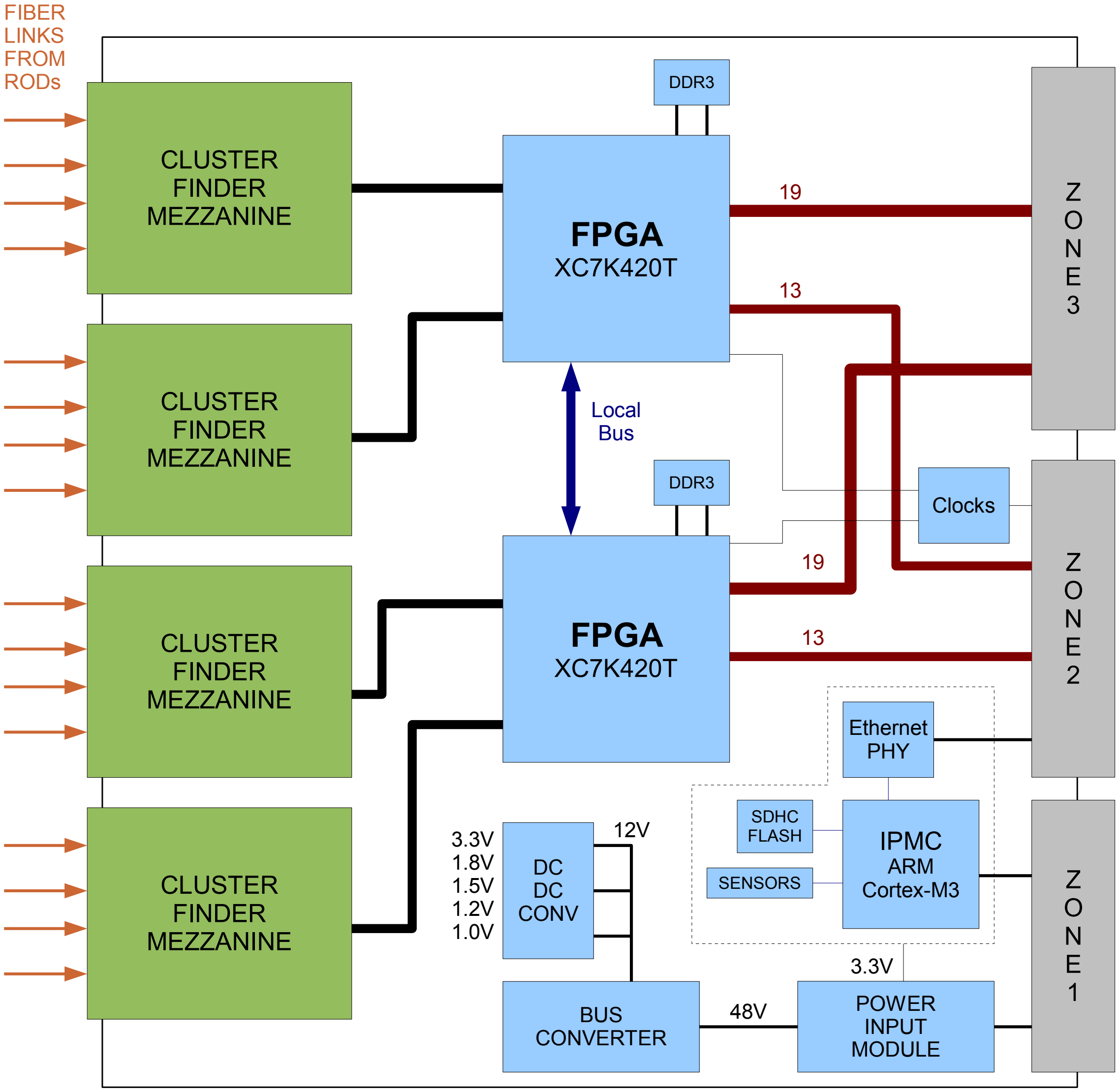


Figure 7: The Data Formatter board block diagram.

The section introduces functions and design details of the Data Formatter board. The Data Formatter block diagram is shown in Figure 7. As described in Section 4, two FPGAs (Section 5.2) are mounted on each Data Formatter board so as to cover two FTK $\eta - \phi$ towers. FPGAs are the Data Formatter's processing engine: they receive cluster data from the mezzanine cards, share data with other FPGAs (i.e. other FTK $\eta - \phi$ towers), and send the data downstream to the FTK core processor crates. Additional circuitry on the Data Formatter board is used to power and configure the FPGAs, implement slow controls and monitoring, provide clocks and synchronization, and support the ATCA system management protocol.

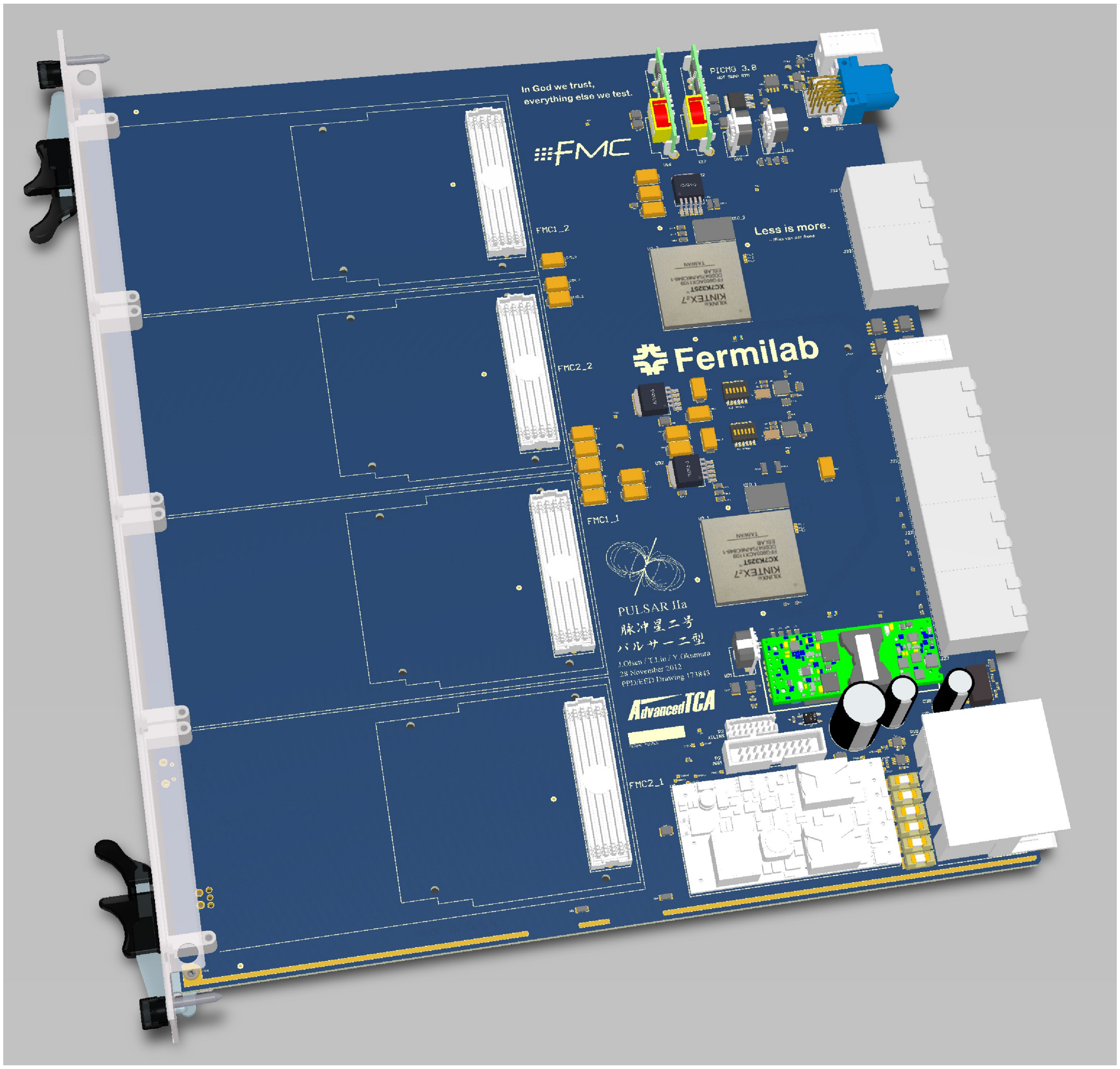


Figure 8: The Data Formatter Board 3D model in Altium layout software.

5.2 Core Processing Engines (FPGAs)

Two large FPGAs form the processing core of the Data Formatter board. These two FPGAs are symmetric and have identical pinouts. The Xilinx Kintex XC7K420T FPGA features 32 high-speed serial (GTX) transceivers which support data rates up to 12.5Gb/s [2], 420k logic cells, 400 IO pins, and 4Mbytes of internal dual port RAM and FIFOs. Firmware studies for the FPGAs are mentioned in Appendix E.

5.2.1 Mezzanine Card Interface

Fiber optic links from the Pixel and SCT RODs are received on the Mezzanine Cards. Each FPGA connects to two Mezzanine Cards as shown in Figure 7. The Mezzanine Card interface uses the FMC (VITA 57) standard and is based on low voltage differential pair (LVDS) signals. Refer to Section 6 for details on this interface.

[2] The speed depends on choice of FPGAs varying from 6.6Gb/s to 12.5Gb/s

5.2.2 Fabric Interface

The ATCA backplane full mesh Fabric Interface supports communication between two FPGAs (i.e. FTK $\eta - \phi$ towers) assigned to the different boards in the same shelf. These connections are shown with dark red lines linking the FPGAs to Zone-2 connector in Figure 7. FPGAs use their internal high speed serial transceivers (GTX) to communicate over the Fabric Interface. In a full mesh ATCA backplane all slots are directly connected with dedicated channels. Each channel consists of four full-duplex ports. Each backplane port is rated for up to 10Gbps.

Data Formatter boards use two of the four available ports on the ATCA full mesh backplane. The Top FPGA connects to port 0 of fabric channels 1 through 13 (see Appendix B and [5]). The bottom FPGA connects to port 1 of fabric channels 1 through 13. This means that Data Formatter boards may be installed in slots 3-14. All top FPGAs are directly connected and all bottom FPGAs are directly connected through the backplane Fabric Interface.

An efficient, lightweight yet robust communication protocol will be used on the serial links connecting FPGAs. We are currently evaluating the Aurora protocol developed by Xilinx.

5.2.3 Inter-FPGA link on board (Local Bus)

The local bus on the board supports communication between two FPGAs (i.e. FTK $\eta - \phi$ towers) assigned to the same boards. A pair of local buses connects the two FPGAs on each Data Formatter board. The local bus is shown with blue lines linking the two FPGAs in Figure 7. The bus consists of LVDS differential pairs, and the maximum data transfer rate of each line varies from 1.2Gbps to 1.6Gbps, depending on the FPGA choice. The number of pairs of the differential LVDS is still flexible and to be determined so that the bandwidth satisfies the given requirement for high luminosity runs.

5.2.4 RTM Interface

The RTMs support data transmission downstream to the FTK processors and data sharing across shelf boundaries. The connection to RTM is shown with red lines linking the FPGAs to Zone-3 connector in Figure 7. Each FPGA has 19 GTX transceivers reserved for RTM connections. 17 GTX channels are reserved for data transmission downstream to FTK, and 2 GTX channels are reserved for inter-shelf communication. RTM optical and copper pluggable transceivers support bidirectional data rates up to 10Gb/s for a maximum bandwidth of 380Gb/s per RTM. Refer to Section 7 for the RTM design details.

5.2.5 DDR3 Memory

Each FPGA has 835 36k-bit internal BlockRAMs which may be configured as FIFOs ROMs, single-port RAM, or dual-port RAM. In the event more memory is required for event buffering an external DDR3 memory device is included in the design. This DDR3 memory chip is 256MBytes and has a 16 bit data bus operating at 800Mb/s for a maximum data rate varying from 6.6Gb/s to 12.5Gb/s, depending on the FPGA speed grade. The effective data rate will vary due to DDR3 signaling overhead but is expected to be nearly 70% of maximum for block sequential transfers and 40% for random access. The Xilinx Memory Interface Generator (MIG) tool creates a firmware interface for the DDR3 memory chip.

5.3 Microcontroller

A small microcontroller is used for various board management functions such as slow controls, board monitoring, FPGA configuration, and ATCA shelf management. The microcontroller

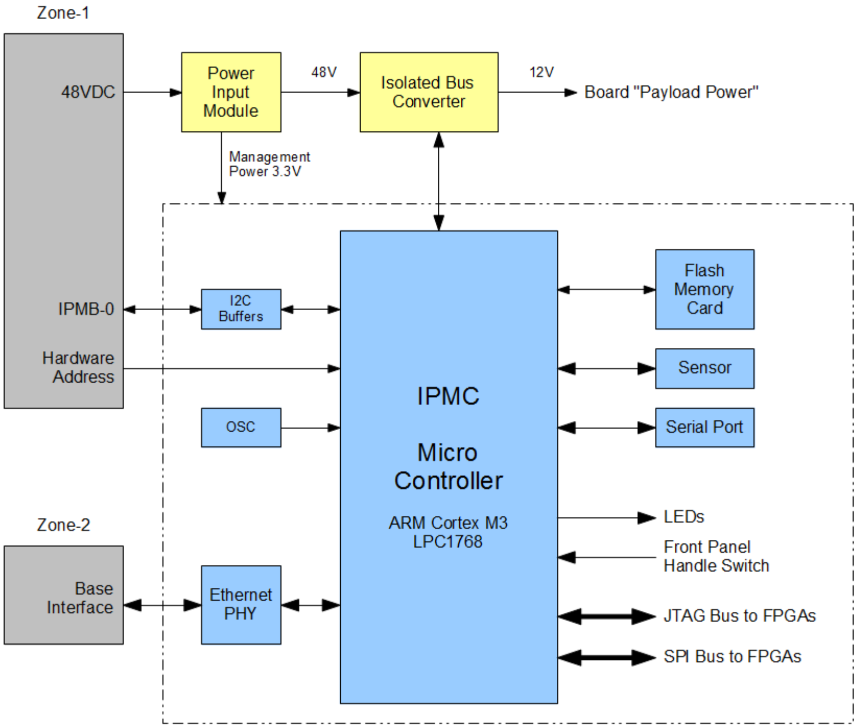


Figure 9: The Data Formatter microcontroller and associated circuitry.

communicates with the ATCA shelf manager boards through the Zone-1 connector. The Zone-2 connector Base Interface enables the microcontroller to interface with Ethernet networks via the hub board located in slot 1. Microcontroller connections are shown in Figure 9.

An ARM Cortex-M3 microcontroller has been selected for use on the Data Formatter board. This microcontroller runs at up to 100MHz and features 512kB Flash program memory, 64kB RAM, an Ethernet MAC interface, and various on-board peripherals such as I2C, RS232 UARTs, and SPI.

The microcontroller implements the ATCA Intelligent Platform Management Interface (IPMI) and the slow controls interface (See Appendix B.5 for more details).

5.3.1 IPMI Controller

ATCA hardware achieves high availability through a robust hardware management scheme. Redundant shelf manager boards communicate with boards and components over redundant I2C buses. The IPMI controller (IPMC, see Appendix B) is used to report board telemetry data to the shelf manager and coordinates the hot swap power sequencing.

The microcontroller has three I2C buses. The first two I2C buses are used for the IPMI buses (IPMB). The third I2C bus connects to a temperature sensor, the main power bus converter, and the RTM.

A commercial IPMC software reference design is available, however this code is closed-source and has expensive licensing terms. A group of Engineers from various HEP laboratories have formed an ad-hoc committee and we are working towards a simple, modular and device-independent open-source solution.

The IPMI protocol includes support for firmware updates. However, the “remote update” functionality should not be used for downloading FPGA firmware because the I2C data rates are very slow, on the order of 100kbps.

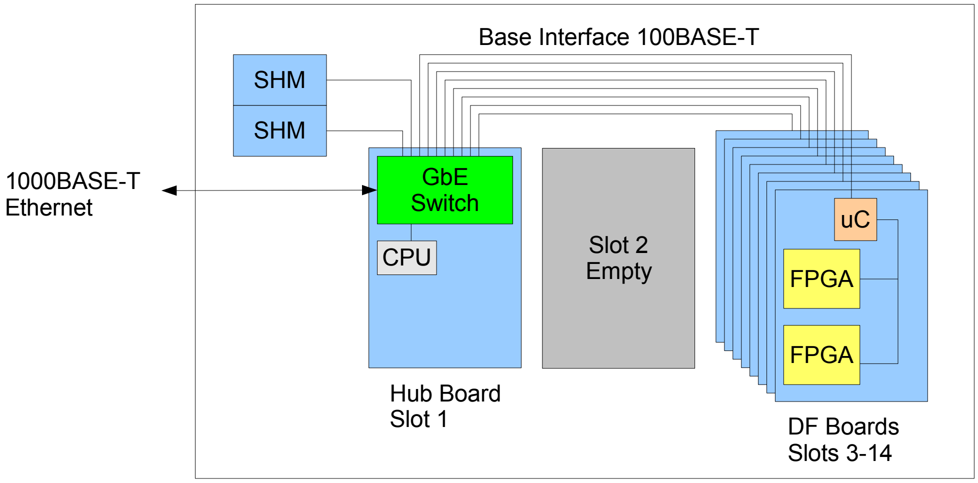


Figure 10: A typical shelf configuration. The Slot 1 Base Interface hub connects the Shelf Manager boards and the Data Formatter board microcontrollers and provides one or more external Gigabit Ethernet ports. Note that the high speed Fabric Interface is not shown here. An optional CPU on the hub may be used as a "crate controller" if required.

5.3.2 Ethernet Interface

The microcontroller features a hardware-based 100BASE-T Ethernet MAC which enables the use of TCP-IP protocols such as WWW, Telnet and FTP. This interface is intended for downloading FPGA firmware and slow controls. Initial testing with our Cortex-M3 development board has demonstrated FTP transfer rates of up to 2MB/s.

A 100BASE-T Ethernet PHY chip (National DP83848) connects the microcontroller to the Base Interface channel 1 on the Zone-2 connector. Note that Base Interface channel 2 is not used. This means that an ATCA hub board (or "switch blade") may be installed only in slot 1 as shown in Figure 10.

5.3.3 Flash Memory

A removable micro-SDHC flash memory card is used for local file storage. FPGA firmware image files will be downloaded over the Base Interface Ethernet interface and stored on the flash card. Alternatively the flash card may be removed from the Data Formatter and inserted into a laptop the files transferred directly to the card. The flash card uses the FAT32 file system and supports long file names.

5.3.4 FPGA Interface

The microcontroller drives the JTAG bus which connects to the two main FPGAs. The JTAG bus is used for programming the FPGAs. After the FPGAs are programmed the microcontroller uses an SPI bus to access internal registers and buffers. The maximum frequency of the SPI bus clock and JTAG clock is 50MHz. It should be noted that this interface is intended for SLOW controls and monitoring. It is not intended for high speed data transfers.

5.3.5 Board Sensors and RTM Management

The microcontroller's third I2C bus connects to a sensor chip (LTC2990) which contains monitors ambient air temperature and the +3.3V management power supply. This I2C bus also connects to the 12VDC bus converter, which monitors input and output voltages, currents, and temperature. Finally the I2C bus connects to the RTM management connector J30.

5.3.6 USB Serial Port

The microcontroller serial port is used for debugging and configuring the network parameters. A small USB to RS232 interface chip (Silabs CP2102) is used and a micro-USB connector is provided on the Data Formatter front panel. The serial port is configured as 115.2k baud, 8 data bits, 1 stop bit, no parity.

5.3.7 Microcontroller Software

The microcontroller will need to switch efficiently between many different tasks such as: IPMI protocol handling, TCP-IP networking, slow monitor management, flash card file systems and general bit-banging on the I/O pins. The ARM community has a wide selection of small-footprint, efficient, real-time operating systems (RTOS) that are ideal for our application. We have purchased the KEIL MDK-Professional tools and are developing our software around KEIL RTX RTOS kernel.

5.4 Clock Distribution

Data Formatter boards are data driven and essentially do not require synchronizing to a master clock. However, we have built in provisions to synchronize Data Formatter boards across the ATCA backplane.

5.4.1 Local Oscillators

The Kintex FPGAs require extremely clean clocks to drive the high speed serial transceivers (GTX). Two clocks are generated locally using high quality precision oscillators and programmable low-jitter PLL-based clock buffers.

The main clock is 200MHz and is intended for general purpose clocking in the FPGAs. This clock also drives the DDR3 controller in the FPGA. An additional clock generator drives two reference clocks to the FPGA GTX refclk input pins.

5.4.2 Backplane Clocks

Shelf level synchronization may be achieved using the ATCA backplane Synchronization Interface, which consists of 6 differential clock signals bused to all slots. The first 4 clocks (CLK1A, CLK1B, CLK2A, and CLK2B) are reserved for standard telecom frequencies. Backplane clocks CLK3A and CLK3B are user-defined and are connected to the Data Formatter FPGAs through multi-point LVDS transceivers. (A Data Formatter FPGA may act has the clock master and drive these clock lines.)

If shelf-wide synchronization with the trigger DAQ is required there are a few options. One option is to develop a mezzanine card with a CERN TTC receiver on it. This special mezzanine card could be plugged into an available Data Formatter board and the firmware could pass the DAQ timing signals through the FPGA to the backplane bus. Another option would be to develop a simple TTC receiver ATCA board and install this special board in slot 2.

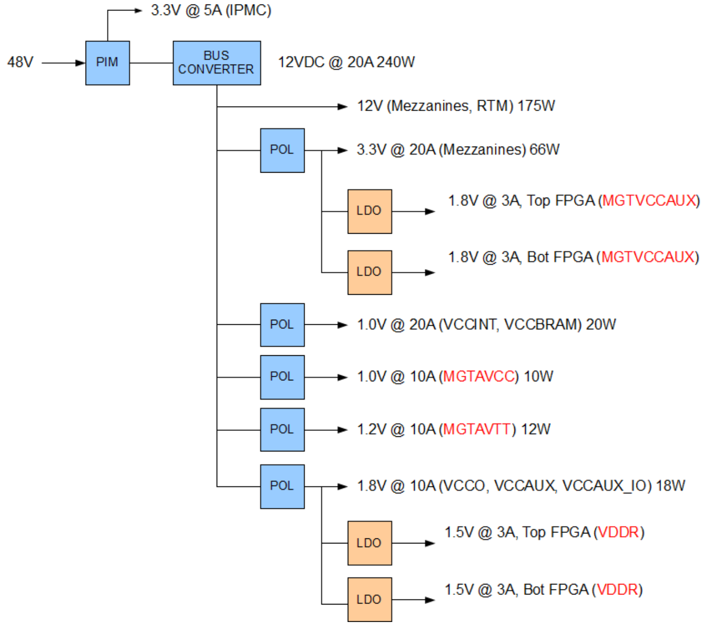

3.3V @ 5A (IPMC)
48V
PIM
BUS CONVERTER
12VDC @ 20A 240W
12V (Mezzanines, RTM) 175W
POL
3.3V @ 20A (Mezzanines) 66W
LDO
1.8V @ 3A, Top FPGA (MGTVCCAUX)
LDO
1.8V @ 3A, Bot FPGA (MGTVCCAUX)
POL
1.0V @ 20A (VCCINT, VCCBRAM) 20W
POL
1.0V @ 10A (MGTAVCC) 10W
POL
1.2V @ 10A (MGTAVTT) 12W
POL
1.8V @ 10A (VCCO, VCCAUX, VCCAUX_IO) 18W
LDO
1.5V @ 3A, Top FPGA (VDDR)
LDO
1.5V @ 3A, Bot FPGA (VDDR)

Figure 11: The Data Formatter power distribution. POL refers to a “Point-of-Load” switching regulator. LDO refers to a “Low Drop Out” linear regulator.

5.5 Power Distribution

5.5.1 Power Distribution Hierarchy

The ATCA backplane supplies dual redundant -48VDC power feeds. These buses are fused, filtered, and converted to the extremely quiet and well-regulated 1.0VDC required by the Kintex FPGAs. The power distribution system is shown in Figure 11.

Immediately after the Zone-1 power connector the power feeds are fused and connected to an ATCA-specific Power Input Module (PIM, GE PIM300FZ). The PIM module filters the power feeds and combines them using diodes. The PIM also generates +3.3V for powering the IPMC microcontroller and associated circuitry.

An isolated bus converter (GE EBDW020A0B) steps the 48VDC down to 12VDC rated for up to 20A. This 12VDC supply powers the RTM and Mezzanine cards. Smaller Point-of-Load switching regulators (GE NQR010A and NSR020A0) generate lower voltages. A few power rails are generated with linear regulators (Linear Tech LT1764A). FPGA GTX transceivers require very quiet power supplies and may not be shared with the general purpose digital power rails.

5.5.2 Power Control Sequence

Immediately upon board insertion the PIM provides power to the IPMC microcontroller. The IPMC and shelf manager negotiate power requirements and once that process completes the shelf manager allows the IPMC to enable the bus converter, which powers up the board and RTM. (Xilinx Kintex FPGAs do not require any specific power supply sequencing as long as each power rail ramps up is monotonic.)

To remove a board from the ATCA shelf the user must first open the lower handle just slightly so that a microswitch is opened. The IPMC sends a message to the shelf manager and slowly blinks the blue “HS” LED. Once the shelf manager has responded the IPMC disables the bus converter and the board may be moved from the shelf.

5.6 Prototype Board

A prototype Data Formatter board is currently being fabricated. The prototype board design is very close to the production board but differs in the FPGA choice. In consequence it differs in GTX transceiver connectivity to the RTM and Fabric Interface, too. The block diagram is shown in Figure 12.

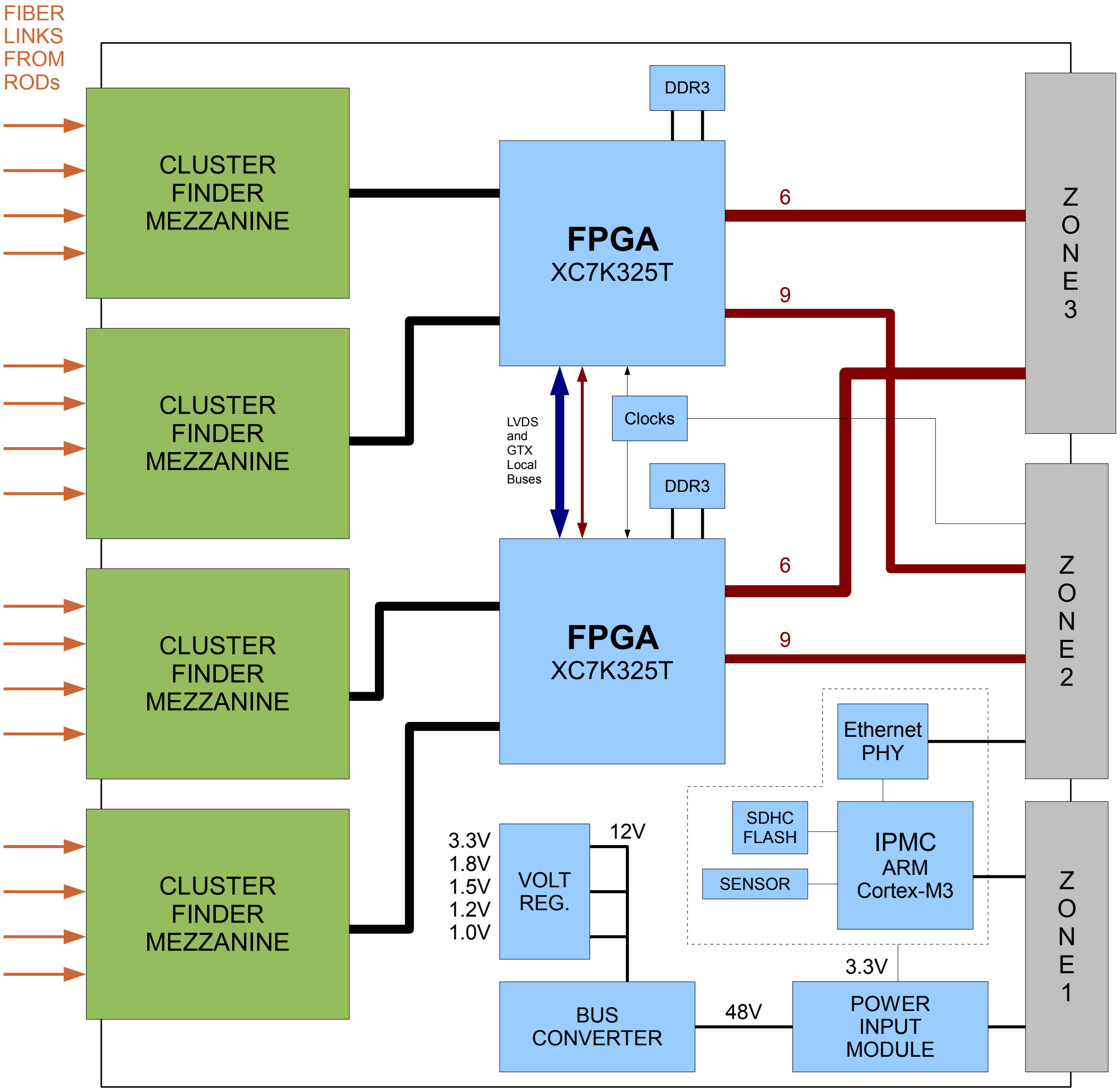


Figure 12: The prototype Data Formatter board uses two smaller FPGAs with reduced RTM and Fabric Interface connectivity.

The prototype Data Formatter uses the Xilinx Kintex XC7K325T device in the FF900 package while the production board will use the Kintex K420T in a FF1156 package. All Kintex FPGAs are currently in full production. The Kintex K325T FPGAs have 16 GTX transceivers. Six GTX transceivers are used for the RTM, 9 GTX transceivers are used for the Fabric Interface, and the last GTX transceiver is used for the FPGA local bus. The limited Fabric Interface connections means that up to eight prototype boards may only be installed in slots 3-10. Each FPGA connects to one QSFP+ transceiver (T1/B1) and two SFP+ transceivers (T5/B5 and T6/B6) on the RTM.

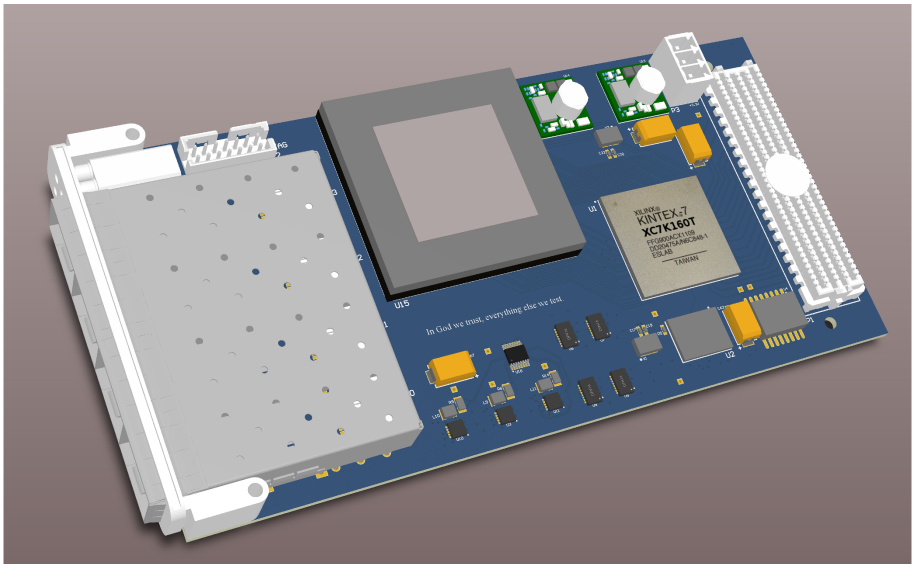


Figure 13: A test Mezzanine Card. This design features four SFP+ pluggable serial transceivers, a small Kintex FPGA, configuration flash memory, DDR3 memory, power supplies, local oscillators, a test socket and FMC connector.

6 Mezzanine Cards

6.1 Introduction

This section focuses on the interface between Mezzanine Cards and Data Formatter boards. The connector mechanical design, bandwidth, power, pin assignments, configuration, and clocking will be discussed. Specific information pertaining to the FTK cluster finding Mezzanine Card is covered in Section 6.8.

The Data Formatter board supports up to four Mezzanine Cards as shown in Figure 7. Data Formatter Mezzanine Cards use the FPGA Mezzanine Card (FMC) connector which is defined by the VITA 57.1 specification [9]. Figure 13 shows a test Mezzanine Card design.

The FMC connector is used on many FPGA development boards from Xilinx, Digilent, and others [10]. By using an FMC connector it is possible to use commercially available FPGA development boards to test prototype Mezzanine Cards. Likewise, the Data Formatter boards are compatible with a growing selection of FMC I/O modules [11].

6.2 FMC Connector

The FMC Connector has 400 pins and provides connectivity for up to:

- 160 single-ended or 80 differential user defined signals
- 10 GTX transceivers
- 2 GTX clocks
- 2 clocks
- 4 differential clocks
- 159 ground and 15 power connections

The connections between the FMC connector and the Data Formatter board FPGAs implements a subset of this connectivity:

- 58 differential user defined pairs:
  - 34 LA pairs (LA00-LA33)
  - 24 HA pairs (HA00-HA23)
- 2 differential clocks
- 159 ground and 10 power connections

The FMC connector is available in High and Low Pin Count versions. The High Pin Count (HPC) has all 400 pins loaded, while the LPC version fills only rows C, D, G, and H for a total of 160 pins. The Data Formatter supports a subset of the HPC signals. The Data Formatter FMC connector pinout is listed in Appendix C.1. The Data Formatter system does not use the connectivity for GTX.

6.3 Signal Levels and Bandwidth

All mezzanine card connector signals are routed directly to the Data Formatter FPGAs. The Data Formatter FPGA I/O banks have a VCCO voltage of +2.5V. Signal levels must not exceed +2.5V on any signal pin or else the Data Formatter FPGAs may be damaged.

Single ended signals should use the `LVCMOS25` I/O standard. Differential signals should use the `LVDS` or `LVDS25_25` I/O standards. If required a differential pair may be treated as two independent single-ended signals, however these traces are routed as differential striplines and crosstalk may be an issue.

Modern FPGAs support data rates exceeding 1Gbps per LVDS pair. If all LA and HA banks are used the total bandwidth on the HPC connector is over 70Gbps using Xilinx Kintex FPGAs in the slowest (-1) speed grade.

6.4 Mezzanine Card Dimensions

The Mezzanine Card follows the standard CMC/PMC dimensions of 74mm x 149mm. The mechanical drawing is shown in Appendix C.3.

The Mezzanine Card will use a standard CMC/PMC style bezel which is 74mm wide. The inter-board spacing (or connector "stack height") is 10mm. An EMC gasket may be used. With a standard CMC/PMC style bezel there is just enough room to fit four SFP optical transceivers and some LEDs.

The component side of the Mezzanine Card faces the Data Formatter board when installed. The Data Formatter boards have no components in this area therefore the maximum component height is 10mm. It is possible to place parts on the “solder side” of the mezzanine card, as long as the total height does not exceed the ATCA maximum component height of 22.5mm.

6.5 Power

The Data Formatter board connects the VADJ pins to 3.3V. According to the VITA specification the FMC connector is rated for up to 4A on the VADJ pins, up to 1A on the 12VDC pins, and up to 3A on the 3P3V pins. Therefore the maximum power available to a mezzanine card is 35W. Mezzanine Cards do not support hot swap. The standby power pin 3P3AUX is not connected on the Data Formatter board.

6.6 Management and JTAG Interfaces

When the Mezzanine Card is installed it should ground the PRSNT_M2C_L pin low. Slow controls may use the I2C bus signals SCK and SDA. The SCK and SDA signals should be pulled up to +2.5V through 4.7k resistors on the mezzanine card.

Mezzanine Cards may use the JTAG interface for programming FPGAs. The Data Formatter is the JTAG bus master and may be used for programming devices on the Mezzanine Cards. However, as this JTAG interface is very slow it is highly recommended that Mezzanine Cards store firmware images locally in flash memories or PROMs. Mezzanine Card designers are strongly encouraged to provide a separate JTAG programming header for local programming and debugging. JTAG signals should use the `LVCMOS_25` I/O standard.

6.7 Clocks

The FMC connector has two dedicated differential output clocks, CLK0_M2C and CLK1_M2C. These signals are routed to global multi-region clock-capable (MRCC) pins on the Data Formatter FPGAs.

The Data Formatter board can drive a differential clock to the mezzanine board on any signal pair in the LA or HA banks. It is recommended to route the LA00, LA01, HA00, HA01 signals to clock-capable pins on the mezzanine card FPGAs.

It is highly recommended that Mezzanine Card designs include a high quality low-jitter local oscillator to provide a clean reference clock source if multi-gigabit serial links are used.

6.8 Cluster Finder Mezzanine Card

The FTK_IM Mezzanine Card receives up to four SLINK fibers and unpacks the ROD data and performs SCT and Pixel detector clustering. The prototype mezzanine card contains four SFP optical transceivers, two FPGAs, and DDR memories and has been developed by INFN [4].

The prototype FTK_IM mezzanine card has four 24-bit parallel data buses (A,B,C, and D) and associated control signals (HOLD, DV). These signals are single-ended and single-data rate synchronous to a 40MHz output clock. Additional pins are used for status, control and other user defined signals.

Using the FMC connector and high speed LVDS signaling it is possible to expand the four output buses to 32 bits wide and still have many signals left for other purposes. Modern FPGAs include serializer-deserializer logic built into the I/O pin. For example, the Xilinx

ISERDES2 and OSERDES2 primitives are capable of serializing and deserializing up to 8 data bits per LVDS signal pair at rates exceeding 1Gbps [14].

6.9 Prototype mezzanine card

A test Mezzanine Card has been designed to test the Data Formatter prototype system (Section 5.6). Figure 13 shows the picture of the test mezzanine card to be used for prototype board tests.

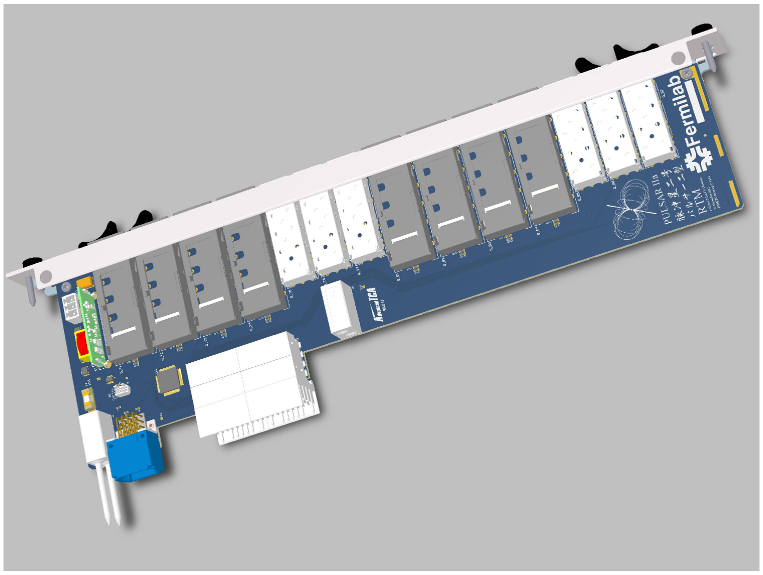


Figure 14: Data Formatter Rear Transition Module (RTM). Up to 8 QSFP and 6 SFP transceivers are supported.

7 Rear Transition Module

We make use of the Rear Transition Module (RTM) to operate the bi-directional high speed serial link drivers of SFP+ and QSFP+ modules. In the Data Formatter system it is used to transmit data downstream to the FTK core crates. Serial transceivers on the RTM are also used for communication between ATCA shelves. Eight QSFP+ and six SFP+ transceivers are used on each RTM. The RTM design is shown in Figure 14. This board complies with the new PIGMG 3.8 Zone-3A specification [6] which defines the connectors, board mechanics, power and management.

7.1 Transceiver Modules

Pluggable transceivers on the RTM are used for inter-shelf data sharing and for downstream transmission to the FTK Aux Cards (AUX) and Second Stage Boards (SSB). Each Data Formatter FPGA drives 19 GTX 10Gb/s transceivers to the RTM. Up to four Quad Small Form Factor Pluggable (QSFP+) transceivers and up to 3 Small Form Factor (SFP+) per FPGA are supported.

QSFP transceivers have four independent full-duplex channels in a 18mm x 52mm x 8.5mm package. Channel bandwidth is nominally 3.125Gb/s (10Gb/s in QSFP+). Single-mode fiber, multi-mode fiber, and lower-cost copper cable QSFP transceivers are available. SFP transceivers have a single full-duplex channel in a 13mm x 56mm x 8.5mm package. Channel bandwidth is nominally 1.25 to 4.25 Gb/s (10 Gb/s in SFP+). Single-mode and multi-mode fiber versions are available. All transceivers have a I2C interface which allows the host to read manufacturer ROM data and operating parameters such as temperature, voltage, and optical signal strength.

7.2 Mechanical Dimensions

The RTM board measures 322.25mm x 92mm as per the PICMG 3.8 specification. A rear panel and ejector handle will also be used. No backplane is used in the Zone-3 region; the RTM connectors mate directly with connectors on the Data Formatter board.

7.3 RTM Power

Both +12V and +3.3V power is supplied through the management connector (P30 on the Data Formatter and RP30 on the RTM). The +3.3V power is limited to just a few watts and is intended only to power the RTM management circuitry. Both the +3.3V management power and the +12V main power are controlled by the Data Formatter IPMC microcontroller in accordance with the PICMG 3.8 specification. The RTM will support hot swap and will use a microswitch to monitor the position of the lower handle.

A non-isolated DC-DC converter is used to step down the +12V main power to +3.3V on the RTM. The +3.3V rail will be used for powering the transceivers. When all transceivers are installed and operating we estimate total RTM power consumption to be on the order of 20W.

7.4 Management Interface

The Data Formatter microcontroller's third I2C bus connects to the P30/RP30 connector. We intend to implement a local IPMI bus (IPMB-L) protocol on this bus and use it to monitor temperature, power and link status for all transceivers located on the RTM. A small ARM Cortex-M3 microcontroller (LPC1317) is used to handle the IPMI protocol. According to the PICMG 3.8 specification the RTM shall appear as an MMC controller to the front board IPMC microcontroller.

7.5 Channel Assignments

The PICMG 3.8 specification defines three ADF data connectors for the Zone-3A RTM interface. The Data Formatter RTM uses two of these connectors (J32 and J33). Each ADF connector has 40 differential pairs, which are divided into 20 channels numbered 0 through 19. The top FPGA connects to J32 and the bottom FPGA connects to J33. Refer to Appendix D.1 for details.

7.6 Prototype Board

The first batch of ten RTM boards have been produced. We are currently writing software for the RTM microcontroller.

8 Bandwidth Requirements

The study to evaluate bandwidth requirements in the Data Formatter System will be presented in this section. A software model of the Data Formatter has been created so that data flow in the system may be simulated using actual event data records taken in 2012. The initial hardware configuration used in this study is explained in Section 8.1, and details of data analysis and the results are summarized in Section 8.2.

8.1 Hardware Configuration

In order to estimate the system bandwidth requirements, we first define the initial hardware configuration. This configuration describes the relationship between Pixel and SCT RODs and Data Formatter boards. The Data Formatter board and shelf assignments, as well as inter-shelf communication links are also taken into consideration. Since the data traffic inside of the Data Formatter System is expected to be largely dependent system configuration, we start with a configuration which we expect will minimize data volume on the backplane and inter-shelf links.

8.1.1 ROD - FPGA - FTK Tower Assignment

FPGA-FTK Tower Initial Assignment First we define the assignment of 64 FTK η-ϕ towers to the 64 FPGAs in four ATCA crates of the Data Formatter System. Table 2 shows the initial mapping. On the Table, "ϕ XX Y Z" indicates the FTK towers linked from the FPGAs. Refer Figure 2(a), 2(b) for FTK η-ϕ tower partitioning, and the numbering scheme of ϕ sectors.

Shelf0	Board0	Board1	Board2	Board3	Board4	Board5	Board6	Board7
Top FPGA	ϕ00 C E	ϕ00 C B	ϕ00 A B	ϕ00 A E	ϕ02 C E	ϕ02 C B	ϕ02 A B	ϕ02 A E
Bottom FPGA	ϕ01 C E	ϕ01 C B	ϕ01 A B	ϕ01 A E	ϕ03 C E	ϕ03 C B	ϕ03 A B	ϕ03 A E

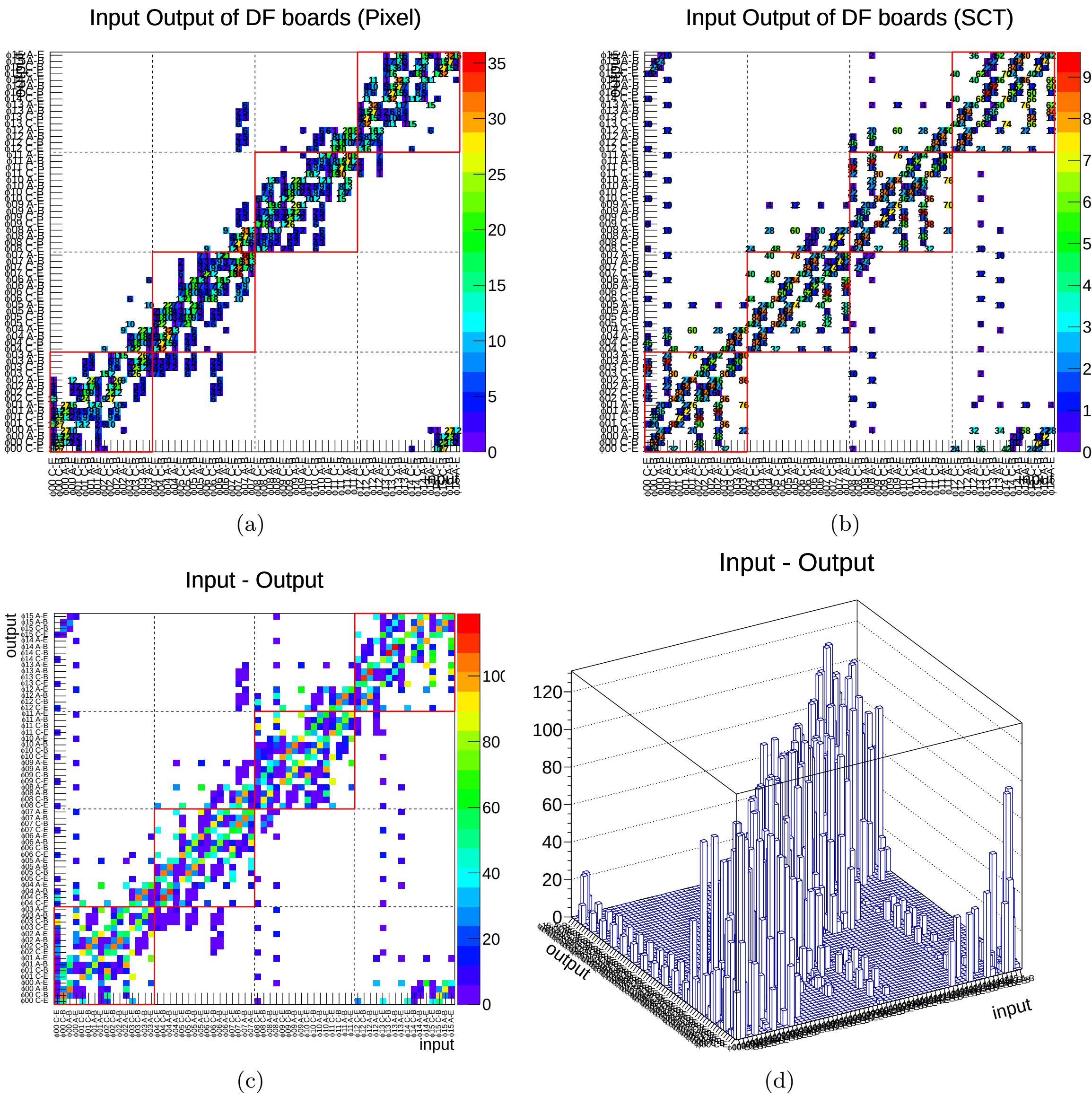
Shelf1	Board0	Board1	Board2	Board3	Board4	Board5	Board6	Board7
Top FPGA	ϕ04 C E	ϕ04 C B	ϕ04 A B	ϕ04 A E	ϕ06 C E	ϕ06 C B	ϕ06 A B	ϕ06 A E
Bottom FPGA	ϕ05 C E	ϕ05 C B	ϕ05 A B	ϕ05 A E	ϕ07 C E	ϕ07 C B	ϕ07 A B	ϕ07 A E

Shelf2	Board0	Board1	Board2	Board3	Board4	Board5	Board6	Board7
Top FPGA	ϕ08 C E	ϕ08 C B	ϕ08 A B	ϕ08 A E	ϕ10 C E	ϕ10 C B	ϕ10 A B	ϕ10 A E
Bottom FPGA	ϕ09 C E	ϕ09 C B	ϕ09 A B	ϕ09 A E	ϕ11 C E	ϕ11 C B	ϕ11 A B	ϕ11 A E

Shelf3	Board0	Board1	Board2	Board3	Board4	Board5	Board6	Board7
Top FPGA	ϕ12 C E	ϕ12 C B	ϕ12 A B	ϕ12 A E	ϕ14 C E	ϕ14 C B	ϕ14 A B	ϕ14 A E
Bottom FPGA	ϕ13 C E	ϕ13 C B	ϕ13 A B	ϕ13 A E	ϕ15 C E	ϕ15 C B	ϕ15 A B	ϕ15 A E

Table 2: FPGA assignment in the Data Formatter system. 64 FPGAs are mounted on 32 Data Formatter boards in 4 ATCA shelves. Each FPGA is linked to one FTK towers. The notation of "ϕXX A/C E/B" indicates the location of the corresponding FTK towers(see Table 18).

ROD - FPGA Mapping Next, we determine the optimal relationship between RODs (SLINK fibers) and Data Formatter FPGAs (η-ϕ towers). Refer to Appendix A for the details of mapping between SLINK fibers and silicon detector modules. The 2D pixel cluster finder algorithms are logic and resource intensive. In order to balance Mezzanine Card resources we limit the number of Pixel SLINK fibers to three per Data Formatter FPGA. Most FPGAs receive two Pixel SLINK fibers, and four FPGAs exceptionally is assigned to three Pixel fibers. The initial cabling between 222 RODs and Data Formatter FPGAs are determined for the bandwidth study. The full mapping lists are shown in Appendix H.



(a)

(b)

(c)

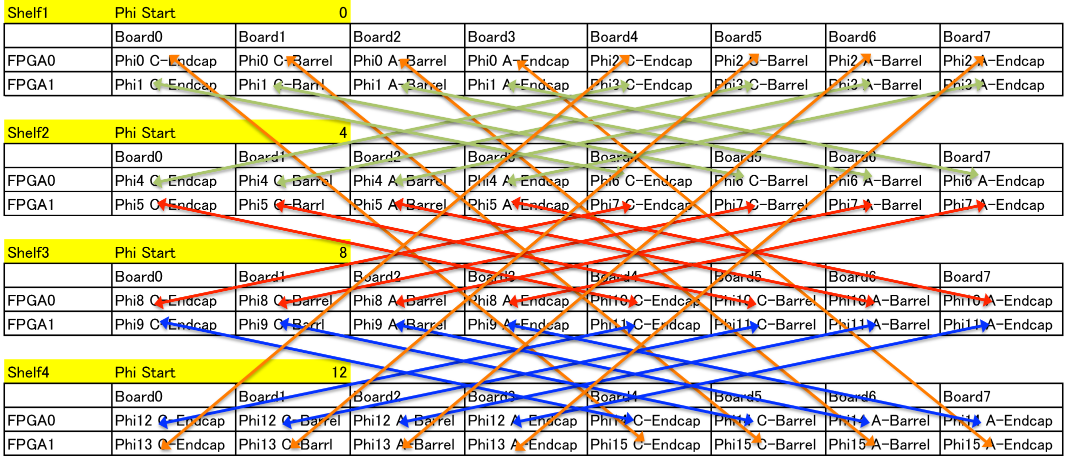
(d)

Figure 15: (a) and (b) show number of modules for source and destination FPGAs for Pixel and SCT, respectively. (c) and (d) show the sum of both Pixel and SCT.

Input and Output Correlation Figure 15 shows the data-traffic matrix inside of the Data Formatter System. The x-axis indicates the source FPGAs and the y-axis shows the destination FPGAs. Off-diagonal components, where the source FPGAs and the designation FPGAs are not the same, shows the data traffic in the Data Formatter System. Red squares in the figures represent ATCA shelf boundaries. Points outside these boxes represent data which must be shared between shelves via additional links which are discussed in the next section.

8.1.2 Inter-Shelf Communication

Our optimized hardware configuration minimizes – but does not eliminate – data sharing between the four ATCA shelves. Fiber or copper links driven by the RTMs are be used for this purpose, which provides flexible external board-to-board communication beyond the ATCA shelves. Figure 16 shows how the inter-shelf links are connected.



Shelf1	Phi Start 0							
	Board0	Board1	Board2	Board3	Board4	Board5	Board6	Board7
FPGA0	Phi0 C-Endcap	Phi0 C-Barrel	Phi0 A-Barrel	Phi0 A-Endcap	Phi2 C-Endcap	Phi2 C-Barrel	Phi2 A-Barrel	Phi2 A-Endcap
FPGA1	Phi1 C-Endcap	Phi1 C-Barrl	Phi1 A-Barrel	Phi1 A-Endcap	Phi3 C-Endcap	Phi3 C-Barrel	Phi3 A-Barrel	Phi3 A-Endcap

Shelf2	Phi Start 4							
	Board0	Board1	Board2	Board3	Board4	Board5	Board6	Board7
FPGA0	Phi4 C-Endcap	Phi4 C-Barrel	Phi4 A-Barrel	Phi4 A-Endcap	Phi6 C-Endcap	Phi6 C-Barrel	Phi6 A-Barrel	Phi6 A-Endcap
FPGA1	Phi5 C-Endcap	Phi5 C-Barrl	Phi5 A-Barrel	Phi5 A-Endcap	Phi7 C-Endcap	Phi7 C-Barrel	Phi7 A-Barrel	Phi7 A-Endcap

Shelf3	Phi Start 8							
	Board0	Board1	Board2	Board3	Board4	Board5	Board6	Board7
FPGA0	Phi8 C-Endcap	Phi8 C-Barrel	Phi8 A-Barrel	Phi8 A-Endcap	Phi10 C-Endcap	Phi10 C-Barrel	Phi10 A-Barrel	Phi10 A-Endcap
FPGA1	Phi9 C-Endcap	Phi9 C-Barrl	Phi9 A-Barrel	Phi9 A-Endcap	Phi11 C-Endcap	Phi11 C-Barrel	Phi11 A-Barrel	Phi11 A-Endcap

Shelf4	Phi Start 12							
	Board0	Board1	Board2	Board3	Board4	Board5	Board6	Board7
FPGA0	Phi12 C-Endcap	Phi12 C-Barrel	Phi12 A-Barrel	Phi12 A-Endcap	Phi14 C-Endcap	Phi14 C-Barrel	Phi14 A-Barrel	Phi14 A-Endcap
FPGA1	Phi13 C-Endcap	Phi13 C-Barrl	Phi13 A-Barrel	Phi13 A-Endcap	Phi15 C-Endcap	Phi15 C-Barrel	Phi15 A-Barrel	Phi15 A-Endcap

Figure 16: Direct fiber connection between two FPGAs for inter-shelf communication. Lines indicate bi-directional communication between two FPGAs.

8.2 Data Analysis

In the following sections we present the data-driven bandwidth requirement estimation for data paths in the Data Formatter System. In all cases we use actual collision data taken in 2012. This study is based on the Data Formatter System hardware configuration introduced in the previous sections.

First we explain the details of the input data volume from the silicon detectors (Section 8.2.3), and the total data volume in the output to the downstream FTK processors for all 64 FTK η-ϕ towers (Section 8.2.4). Section 8.2.6 shows the expected data volumes in the three types of serial links in the Data Formatter System. In order to estimate the bandwidth requirement to maintain the data sharing at the highest expected luminosity environment of $\mathcal{L} = 3 \times 10^{34}$ cm^{-2}s^{-1}, the results are extrapolated according to the difference of number of pileup collisions, collision energy, and bunch spacing. Table 9, 10 summarizes the expected data volume and required bandwidth for the three types of Data Formatter serial links.

8.2.1 Datasets

We used about 10k collision events recorded in Muon Trigger stream with $\mathcal{L} = 6.4{\times}10^{33}$ cm^{-2}s^{-1}. The corresponding average of number of interaction per bunch crossing is $\langle\mu\rangle = 30.0$. In order to extract the collision events, we select the events recorded at the collision bunch crossing according to the event BCID. Table 3 summarizes the data sample conditions.

8.2.2 Number of Hits (N_{Pixel} and N_{Strip})

The Pixel and SCT hit counts take the following factors into account:

Run Number	214523
Lumi-block	153
Number of events	10,644
Data Stream	Muon Stream
Collision Energy	$\sqrt{s} = 8$ TeV
Average number of interactions per bunch crossing	$\langle\mu\rangle = 30.0$
Instantaneous luminosity	$\mathcal{L} = 6.4 \times 10^{33}\mathrm{cm}^{-2}\mathrm{s}^{-1}$
Number of collision bunches at ATLAS	1,368
Bunch Spacing	50 ns
SCT Readout mode	"X1X" timing cuts

Table 3: Data Sample summary. See the text for the SCT readout mode.

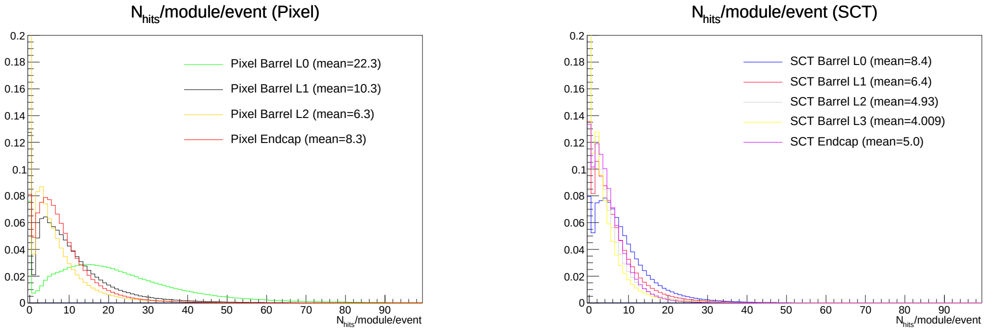
- No clustering is performed for both Pixel and SCT data. Normally clustering will be handled by FPGAs on the Mezzanine Cards. By ignoring the data reduction associated with clustering our bandwidth estimates are conservative.
- The "01X" [3] timing requirement, also known as the SCT readout edge mode, is emulated in the data analysis while the SCT readout timing requirement was "X1X" in the run.

Note all the bandwidth requirements discussed in this section are based on this counting method for the Pixel and SCT hits. Appendix O provides more information about the number of hits expected when clustering is applied.

8.2.3 Input Data Volume

First, the input data volume is checked using the data sample of actual collision data with $\sqrt{s} = 8$TeV, $\langle\mu\rangle = 30.0$, and 50 ns bunch spacing.

Figure 17 shows distributions of the numbers of hits per silicon module for different detector regions for Pixel and SCT, separately. Table 4 summarizes the average numbers of hits per module in each detector region.



(a)

(b)

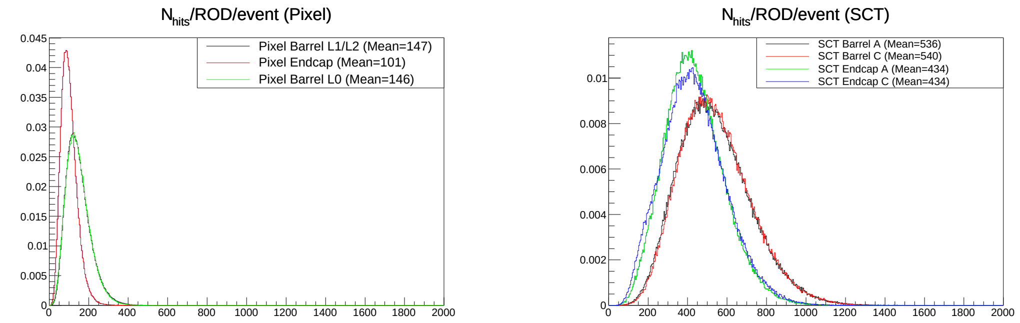
Figure 17: N_{hits} per module per event for (a) Pixel and (b) SCT.

[3] These three characters represent three bunch crossings around the bunch crossing pointed to by the L1 trigger. These bunch crossings are called "previous", "central" and "next" in that order. "01X" stands for the requirement that the channel is active in the central bunch crossing with the middle "1" and inactive in the previous bunch crossing with the first "0". The last "X" represents that we do not care about the next bunch crossing. Only the hits satisfying this timing requirement will be recorded.

	$\langle N_{\mathrm{hits}}/\mathrm{module}/\mathrm{event}\rangle$
Pixel (Barrel L0)	23
Pixel (Barrel L1)	11
Pixel (Barrel L2)	6.8
Pixel (Endcap)	8.8
SCT (Barrel L0)	16/module (8.2/layer of SCT doublet)
SCT (Barrel L1)	12/module (6.2/layer of SCT doublet)
SCT (Barrel L2)	10/module (4.8/layer of SCT doublet)
SCT (Barrel L3)	7.6/module (3.8/layer of SCT doublet)
SCT (Endcap)	10/module (5.0/layer of SCT doublet)

Table 4: Number of hits per module per event.

Figure 18 shows distributions of the numbers of hits per silicon module for different detector regions for Pixel and SCT, separately. Table 5 summarizes the average numbers of hits per module in each detector region.



(a) (b)

Figure 18: Observed number of hits per module per ROD. Note that exceptionally the “01X” timing cut for SCT is not applied in the table.

Region	N_{ROD}	$\langle N_{\mathrm{hits}}/\mathrm{ROD}/\mathrm{event}\rangle$
Pixel Barrel (L0)	44	146
Pixel Barrel (L1 or L2)	64	147
Pixel Endcap	24	101
SCT Barrel A	22	536
SCT Barrel C	22	540
SCT Endcap A	23	434
SCT Endcap C	23	434

Table 5: First column shows number of RODs used for seven detector regions, and second column shows number of average number of hits per a ROD per event ($\langle N_{\mathrm{hits}}/\mathrm{ROD}/\mathrm{event}\rangle$).

8.2.4 Output Data Volume

In this section we show the number of hits sent downstream to FTK with $\sqrt{s} = 8$TeV, $\langle\mu\rangle = 30.0$, and 50 ns bunch spacing. Figure 21 shows the number of output hits from each detector region to the individual 64 FTK towers. (Refer to Table 18 for the 64 tower ID definitions.) The individual numbers for the Figure are summarized in Appendix K.

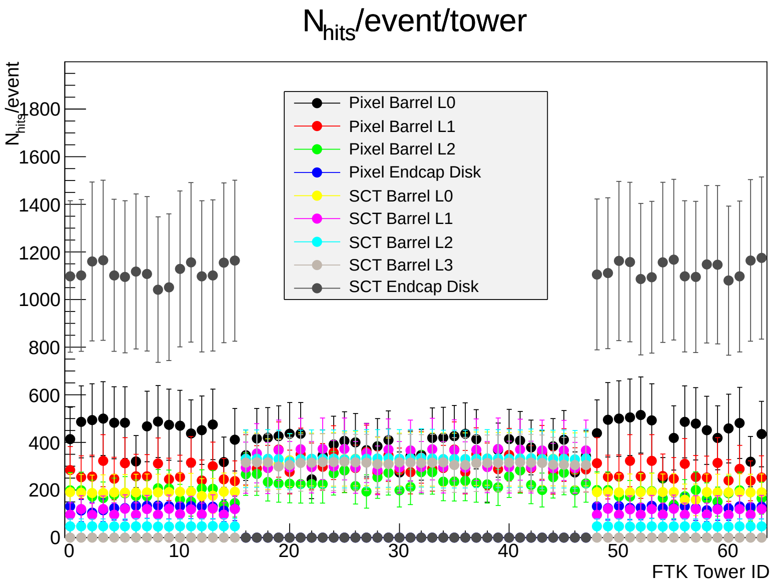
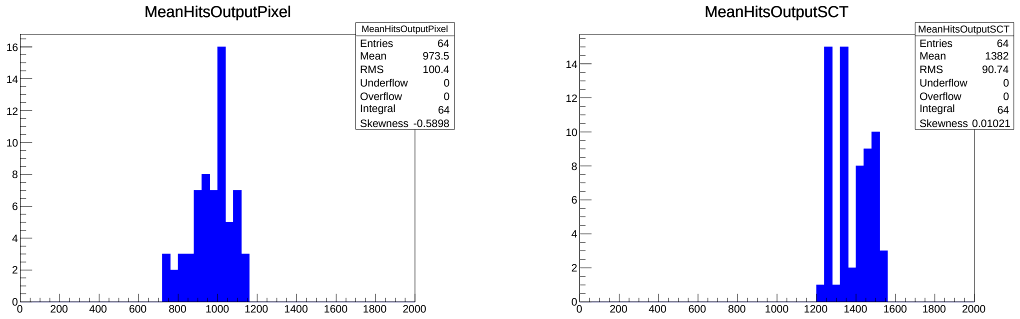


Figure 19: Summary of N_{hits} to individual 64 FTK towers for each detector region.

The number of output N_{Pixel} and N_{SCT} for all the 64 FTK towers are shown in Figure 21.



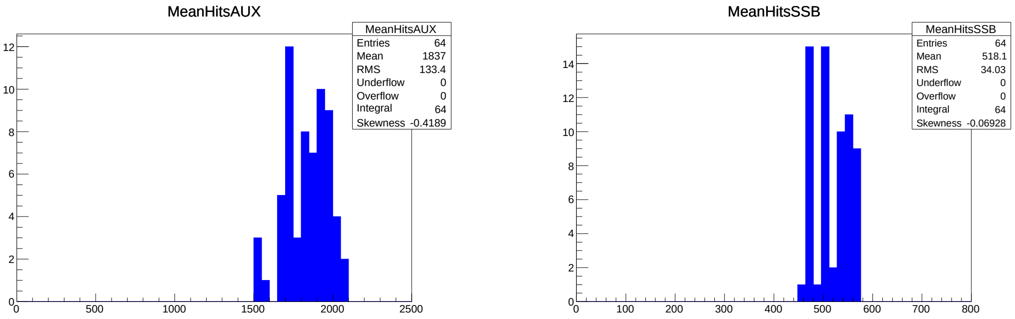
(a)

(b)

Figure 20: Summary of N_{hits} to individual 64 FTK towers for each detector region for (a) N_{Pixel}, and (b) N_{SCT}.

Up until this point we have considered the Data Formatter output as a single stream. In practice each Data Formatter FPGA drives multiple data links downstream to FTK, hence the number of transceivers on the RTM. The AUX Cards implement the first track fitting stage while the aptly-named Second Stage Boards (SSB) implement the second stage fitting (see Section 3.1). Figure 21 shows the N_{hits} to the AUX Cards and SSB separately.

Note that that the AUX Card will receive all Pixel hits and SCT hits from five out of eight layers. The Second Stage Boards will receive the remaining three SCT layers. Therefore N_{hitsAUX} is counted as "$N_{\mathrm{Pixelhits}} + 5/8 \times N_{\mathrm{SCThits}}$" event-by event, and N_{hitsSCT} is counted as "$3/8 \times N_{\mathrm{hits}}$" as approximation. Table 7 summarize the expectation of the typical data traffic among the 64 FTK towers and the maximum case for output N_{Pixel}, output N_{SCT}, output N_{AUX}, and output N_{SSB}, respectively.



(a)

(b)

Figure 21: Summary of N_{hits} to individual 64 AUXs and SSBs for each detector region.

	average of $\langle N_{\mathrm{hits}} \rangle$	maximum of $\langle N_{\mathrm{hits}} \rangle$
Total (output stream)	2.4e+03	2.6e+03
Pixel Total (output stream)	9.7e+02	1.1e+03
SCT Total (output stream)	1.4e+03	1.5e+03
AUX	1.8e+03	2.1e+03
SSB	5.2e+02	5.7e+02

Table 6: Summary of N_{hits} sent to the downstream from the DF system. Typical data traffic for 64 FTK η-ϕ towers and the maximum case are summarized for output N_{Pixel}, output N_{SCT}, output N_{AUX}, and output N_{SSB}, respectively.

8.2.5 Data Flow Routing Rules

We simulate the number of hits transferred inside of the Data Formatter System to estimate data traffic. We have developed software to simulate the data traffic with the following routing rules:

1. Does the hit need to go to another shelf? If so, the hit will be sent over the inter-shelf link. If the hit arrived at the destination FPGA then, go to 4, otherwise go to next 2.

2. Does the hit need to go to other FPGA on the board? If so, the hit will be sent over the local bus. If the hit arrived at the destination FPGA then, go to 4, otherwise go to next 3.

3. Does the hit need to go to another FPGA in the same shelf? If so, the hits will be sent over the backplane Fabric Interface to the destination board. The hit should arrive at the destination FPGA at this step, and go to next 4.

4. Send the data downstream to FTK.

The data flow model described here enables us to count the number of hits passing through individual serial links implemented in the Data Formatter System. The corresponding flowchart for the simulation is shown in Appendix L.

8.2.6 Expected Number of Hits

Our software simulation model enables us to run data records through the Data Formatter system on an event-by-event basis as we evaluate the number of hits transferred on each serial link. The typical N_{hits} for the individual links are summarized in Figure 22 for (a) local bus communication, (b) ATCA Fabric Interface communication, and (c) inter-shelf communication, respectively. Most links in the system are utilized. The number of active links are:

- 64 out of 64 inter-FPGA local bus
- 409 out of 448 ATCA backplane Fabric Interface links
- 45 out of 64 inter-shelf links

Our data flow routing algorithms select the "most direct" route between FPGAs. In some cases this methodology results in links which are either unused or under-utilized, as is shown in the broad distribution in Figure 22. This suggests that by tuning the routing algorithms we should be able to divert traffic away from high volume links and take advantage of the available bandwidth on low-volume links.

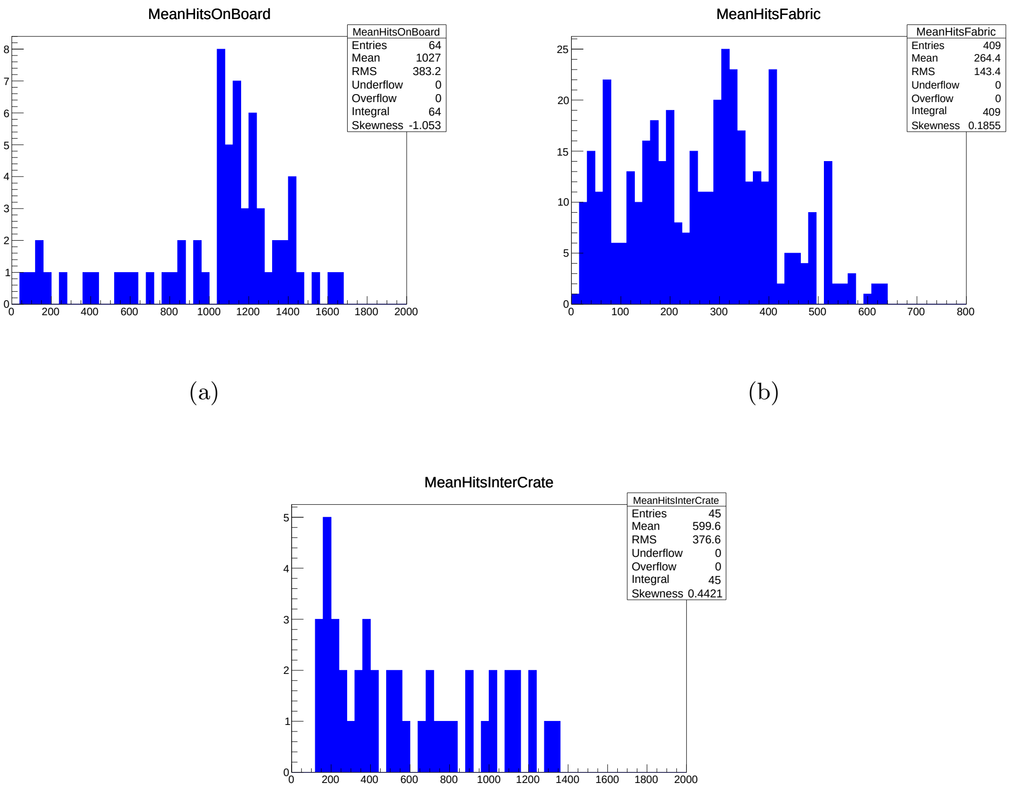
	average of $\langle N \rangle$	maximum of $\langle N \rangle$
Fabric	2.6e+02	6.3e+02
Local Bus	1.0e+03	1.7e+03
Inter-Crate	6.0e+02	1.3e+03

Table 7: Summary of N_{hits} transferred in the DF system. Typical data traffic and the maximum case are summarized for each Data Formatter link type. The typical value is average of the $\langle N \rangle$ for all the active links. This data set was collected under the following conditions: $\sqrt{s} = 8\text{TeV}$, $\langle \mu \rangle = 30.0$, with 50 ns bunch spacing.

8.2.7 Bandwidth Requirements

To estimate the data bandwidth requirements for system links we must first convert the number of hits N_{hits} into 32-bit words N_{words}. The Data Formatter FPGAs do not see the incoming SLINK data records, but rather they see the output of the cluster finder Mezzanine Cards. This output format is shown in Figure 23.

Using the Mezzanine Card format, we can approximate the number of 32-bit words using the formula $N_{\text{words}} \sim N_{\text{Pixel Hits}} + 0.5 \times N_{\text{SCT hits}} + N_{\text{Pixel module}} + N_{\text{SCT module}}$. (Our word count also takes into account the data format header words, which is why $N_{\text{Pixel module}} + N_{\text{SCT module}}$ are included in the formula.) The N_{hits} analysis is discussed in Section 8.2.4 and Section 8.2.6 will be translated into N_{words} as summarized in Figure 24 and Figure 25. The typical N_{words} and the maximum case are summarized in Table 8.



(a)

(b)

(c)

Figure 22: Summary of sum of N_{Pixel} and N_{SCT} transferred in the DF system. (a) shows data volume transferred via the so-called inter-FPGA link connecting two FPGAs on a board. (b) shows data volume transferred via the so-called fabric link supported by the ATCA backplane connecting two FPGAs on the different board in the same ATCA shelf. (c) shows data volume transferred via the so-called inter-Shelf link connecting two FPGAs via optical fibers driven by the RTM. The connectivity via this link is totally flexible. See the text for more discussion.

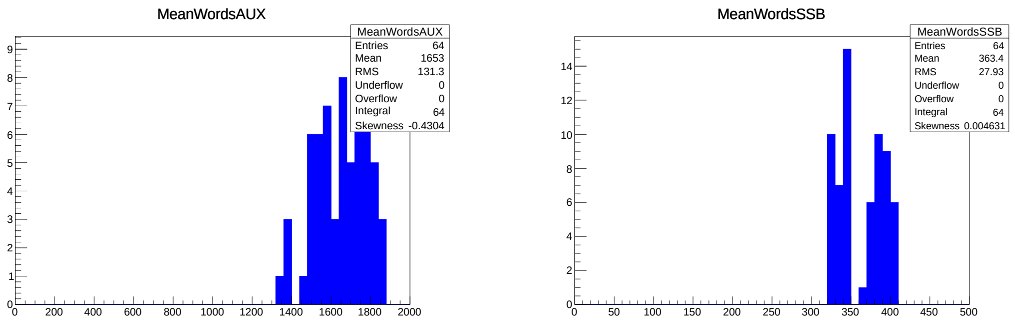
Bit	31	30	29	28	27	26	25	24	23	22	···	14	13	12	11	10	09	···	01	00
Pixel Module	1	Reserved												0	0	module number				
Pixel Hit	0	Reserved		dE/dx	column width			column coordinate					row width		row coordinate					
SCT Module	1	Reserved												1	module number					
SCT Hit	0	Reserved		Valid Hit	Hit 2 width			Hit 2 coordinate					Hit 1 width			Hit 1 coordinate				

Figure 23: Input data format

8.2.8 Extrapolation to Target LHC Operating Conditions

In order to extrapolate from the reference run ($\sqrt{s} = 8$ TeV, $\langle\mu\rangle = 30.0$, and 50 ns bunch spacing) into target condition with $\sqrt{s} = 14$ TeV, $\langle\mu\rangle = 70.0$, and 25 ns bunch spacing, the data volume dependence on the $\langle\mu\rangle$ is studied.

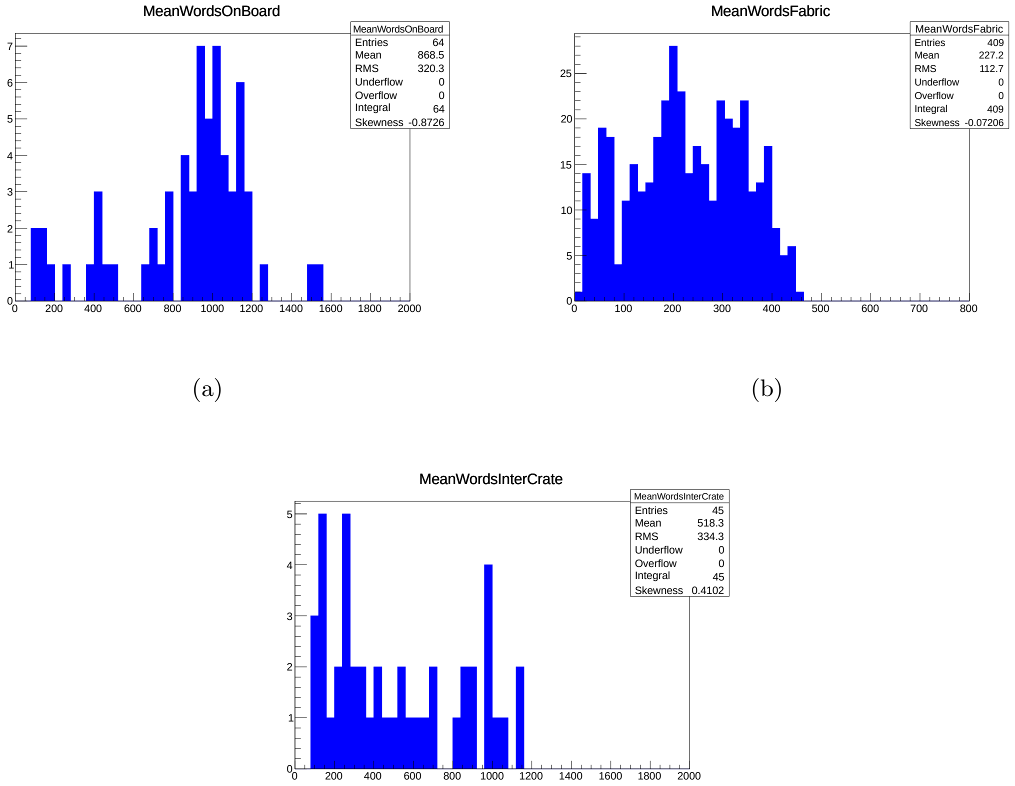
$\langle\mu\rangle$ Difference To confirm the linearity with the real collision data up to $\langle\mu\rangle = 70$, we analyzed the special runs with $\langle\mu\rangle$ taken in July 2012 as well. For this study, we used collision events triggered by a random trigger in the minimum bias stream to get rid of the trigger



(a)

(b)

Figure 24: Summary of N_{words} to individual 64 AUXs and SSBs for each detector region, corresponding to Figure!24. This data set was collected under the following conditions: $\sqrt{s} =$ 8 TeV, $\langle\mu\rangle = 30.0$, and 50 ns bunch spacing.



(a)

(b)

(c)

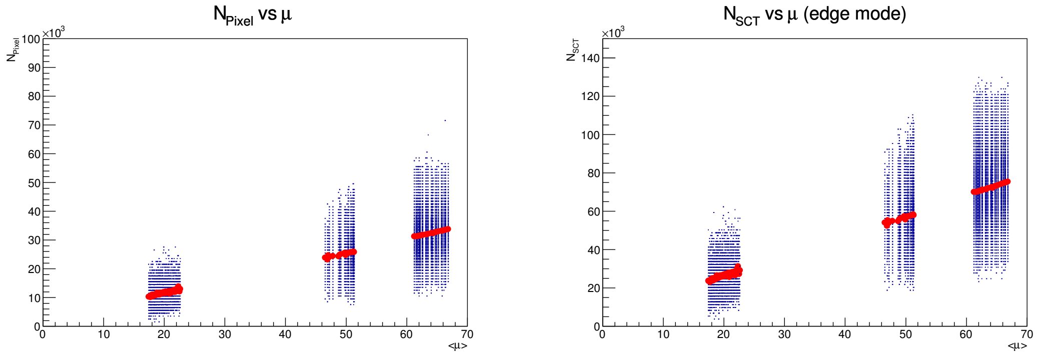
Figure 25: Summary of N_{words} transferred in the DF system, corresponding to Figure 22. This data set was collected under the following conditions:t $\sqrt{s} = 8$ TeV, $\langle\mu\rangle = 30.0$, and 50 ns bunch spacing.

configuration difference between run periods. The linearity as a function of $\langle\mu\rangle$ is shown up to $\langle\mu\rangle = 70$ for both Pixel and SCT detectors. Based on this observation, we scale up the

	average of $\langle N \rangle$	maximum of $\langle N \rangle$
AUX	1.7e+03	1.9e+03
SSB	3.6e+02	4.1e+02
Fabric	2.3e+02	4.6e+02
Local Bus	8.7e+02	1.5e+03
Inter-Crate	5.2e+02	1.1e+03

Table 8: Summary of N_{words} for each type of links with the definition of encoding shown in Figure 23. This data was collected under the following conditions: $\sqrt{s} = 8$ TeV, $\langle \mu \rangle = 30.0$, with a 50 ns bunch spacing.

number of hits to extrapolate the data volume from $\langle \mu \rangle = 30$ up to $\langle \mu \rangle = 70$. More discussion is shown in Appendix M.



(a)

(b)

Figure 26: Data volume dependence on $\langle \mu \rangle$ with $\sqrt{s} = 8$ TeV. (a) Number of Pixel raw hits, and (b) Number of SCT raw hits passing "01X" timing cut (emulating "edge mode" operation).

Energy and Bunch-Spacing Difference To understand the difference between two conditions:

- $\sqrt{s} = 8$ TeV and 50 ns bunch spacing
- $\sqrt{s} = 14$ TeV and 25 ns bunch spacing

the two MC samples are compared as shown in the Figure 27 for Pixel detector behavior [4]. By comparison between $\sqrt{s} = 14$ TeV samples and extrapolated point of $\sqrt{s} = 8$ TeV samples, we extracted roughly 45% increase for the $\sqrt{s} = 14$ TeV and 25 ns bunch spacing. Then this 45% increase of the N_{hits} applied for for both Pixel and SCT number of hits extrapolation for the energy and bunch crossing difference.

Further discussion for the MC-Data comparison study are shown in Appendix N.

8.2.9 Bandwidth Requirement Summary

Taking the difference of 1) luminosity, 2) beam energy, and 3) bunch-crossing, a scale factor of about 3.4 is applied to the hit count [5], and the N_{words} is scaled up accordingly to estimate operating with $\sqrt{s} = 14$ TeV, $\langle \mu \rangle = 70.0$, and 25 ns bunch spacing.

[4] MC comparison for SCT was not done so far due to the difference of the readout configuration between the two samples. The "X1X" cut is pre-applied to 50 ns bunch spacing runs, and "01X" cut is pre-applied to 25 ns bunch spacing runs.

[5] For pileup multiplicity difference 70./30. In total $70./30 \times 1.45 \sim 3.4$

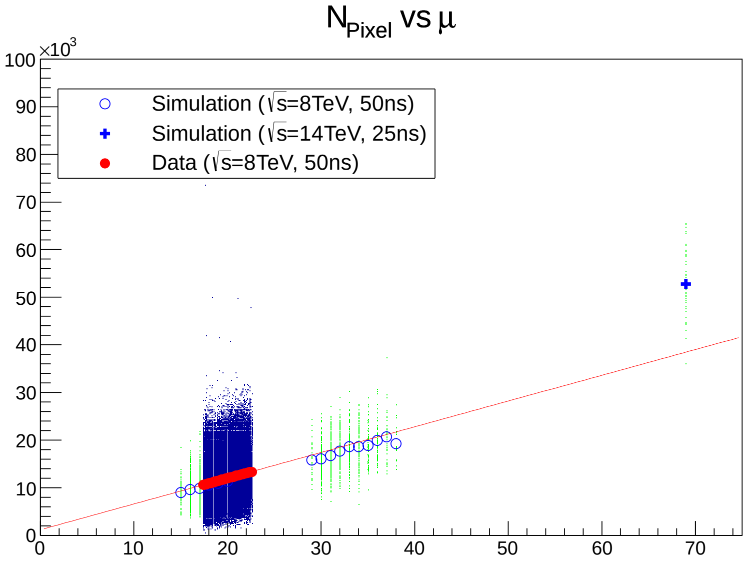


Figure 27: Number of Pixel hits as a function of $\langle\mu\rangle$. Two different MC samples are plotted for 14 TeV and 8 TeV. Just for reference, the data in 2012 are also superimposed.

Then the number of words are translated into number of bits. Each word is 32-bits as shown in Figure 23. Since these data words will be sent over high speed serial links we must also take into account the overhead associated with 8b/10b encoding. Thus each 32-bit data word will be transmitted as 40 bits.

Table 9 and Table 10 summarize the expected N_{hits} and N_{words} under target LHC operating conditions. Both the typical value and the maximum case are shown in this table. The maximum N_{words} determines the bandwidth requirement for each link, which is shown in the third column of Table 10. The rightmost column shows the system bandwidth capacity.

	average of $\langle N\rangle$	maximum of $\langle N\rangle$
Total (output stream)	8.0e+03	8.8e+03
Pixel Total (output stream)	3.3e+03	3.8e+03
SCT Total (output stream)	4.7e+03	5.2e+03
AUX	6.2e+03	7.1e+03
SSB	1.8e+03	2.0e+03
Fabric	9.0e+02	2.2e+03
Local Bus	3.5e+03	5.6e+03
Inter-Crate	2.0e+03	4.4e+03

Table 9: Summary of the expected N_{hits} with $\sqrt{s} = 14$ TeV, $\langle\mu\rangle = 70.0$, and 25 ns bunch spacing. These numbers have been extrapolated as discussed in Section 8.2.8.

	average of $\langle N \rangle$	maximum of $\langle N \rangle$	BW Required	Link Capacity
Total (output stream)	5.9e+03	6.6e+03	-	-
Pixel Total (output stream)	3.4e+03	3.9e+03	-	-
SCT Total (output stream)	2.6e+03	2.8e+03	-	-
AUX	5.0e+03	5.6e+03	22.6	48
SSB	9.6e+02	1.1e+03	4.31	6.0
Fabric	6.7e+02	1.4e+03	5.73	10
Local Bus	2.5e+03	4.7e+03	18.7	24
Inter-Crate	1.5e+03	3.5e+03	13.8	20

Table 10: Summary of the expected N_{words} at $\sqrt{s} = 14$ TeV, $\langle \mu \rangle = 70.0$, and 25 ns bunch spacing. These values have been extrapolated as discussed in Section 8.2.8. The third column shows the maximum worst-case link bandwidth requirements (in Gb/s). The forth column shows the preliminary system bandwidth capacity (in Gb/s) with the current design.

8.3 Conclusion

In order to study the bandwidth requirements of the Data Formatter system we have first defined what we consider is the optimal hardware configuration of RODs, SLINKs, Data Formatter boards, and ATCA shelves based on our study. Real event data records have been run through a simulation model of the entire Data Formatter system and the number of hits flowing through each link have been analyzed. These results have been scaled and extrapolated out to the target LHC operating conditions of $\sqrt{s} = 14$ TeV, $\langle \mu \rangle = 70.0$, and 25 ns bunch spacing. Under these target operating conditions the Data Formatter design meets bandwidth requirements for all internal data links. The IBL data volume case will be added in future after fixing all the hardware configuration of IBL detectors.

Given the Data Formatter flexible design, there exists significant reserve capacity in the system. The full mesh ATCA backplane makes possible many equivalent alternate paths between FPGAs. Data may be diverted from high-volume links to under-utilized links, effectively reducing the bandwidth requirements while at the same time improving system latency. These studies, as well as their impact on the FPGA data routing engine firmware, are ongoing.

Appendix A System Analysis

The ATLAS detector and readout electronics were not originally designed to support a track trigger. Specifically, the physical locations of the inner detector modules do not line up with symmetric ϕ region boundaries. Furthermore, the mapping between modules and RODs introduces another level of asymmetry into the readout scheme. Early on we recognized that these factors would prove to be the primary challenge faced in designing the Data Formatter. In this section we present a summary of our early analysis.

A.1 Pixel and SCT readout partitioning

The Data Formatter system receives silicon hits via SLINK fibers from the Pixel and SCT RODs. Table 11 and Table 12 summarize the arrangement of the ROD modules for Pixel and SCT. The RODs are connected to the upstream silicon detector modules over optic fiber links. Table 13 14 summarize the number of detector modules for Pixel and SCT. In total 222 RODs modules are used to read out hit data from 1,744 Pixel modules, 4,224 SCT barrel modules, and 3,952 SCT end-cap modules. Our early simulation results [1] indicated that the amount of data traffic between Data Formatter boards is highly dependent on the module-ROD mapping, and Figure 28 29 30 31 show the module-ROD mapping for the current existing detectors in x-y view. Numbers in the figures indicate "ROD IDs" assigned to the modules. (The ROD ID is explained in the Appendix F.) It is important to note that module-ROD mapping is optimized to balance data traffic over the optic fiber links and to use the bandwidth resources most effectively; the mapping was never intended to organize hits into the uniform symmetric η-ϕ towers that a tracking trigger system requires.

This study shows that some RODs cover a significant ϕ regions or even non-contiguous detector regions. The following are examples of such exceptions. Figure 28(c) shows each ROD covers 1/4 of the ring in the second disk of Endcap Pixel (as an example of large ϕ coverage). Figure 28(a) shows a ROD that covers two separate regions in the second layer of Pixel Barrel (as an example of the separate coverage). The RODs are the "worst case" configurations and result in maximum data sharing in the Data Formatter system.

	Barrel - A	Barrel - C	Endcap - A	Endcap - C
Layer/Disk 0	22	22	8	8
Layer/Disk 1	19	19	4	4
Layer/Disk 2	13	13	-	-
total	54	54	12	12

Table 11: Number of RODs for Pixel readout. RODs for Endcap Disk0 cover Endcap Disk2 as well. 132 RODs are used in total.

Barrel - A	Barrel - C	Endcap - A	Endcap - C
22	22	23	23

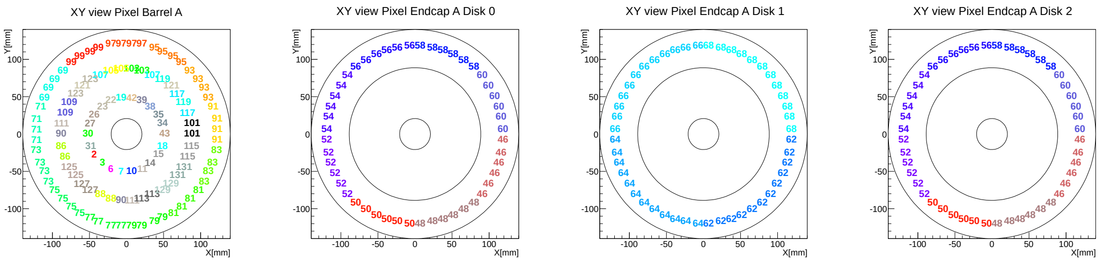
Table 12: Number of RODs for SCT readout. 90 RODs are used in total. The SCT RODs boundary does not correspond to the layer/disk (See Figure 30 and Figure 31), and therefore only the sum for all the layers/disks are shown in the Table, differently from case of Pixel.

	Barrel - A	Barrel - C	Endcap - A	Endcap - C
Layer/Disk 0	143	143	48	48
Layer/Disk 1	247	247	48	48
Layer/Disk 2	338	338	48	48
total	728	728	144	144

Table 13: Number of modules for Pixel readout. The numbers shown in the table are identical to those of input fibers to 132 Pixel RODs. 1744 modules are used in total.

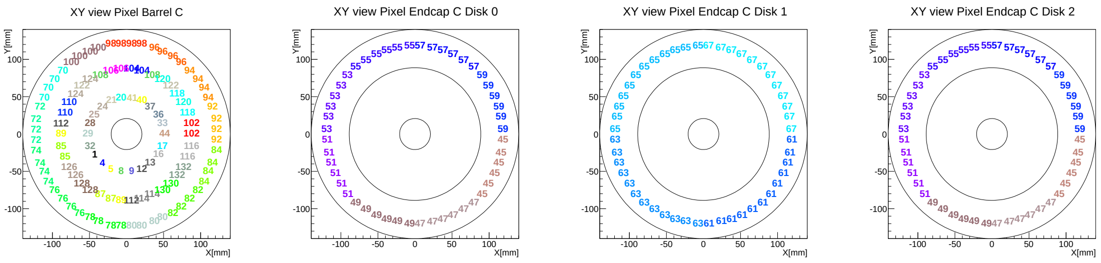
	Barrel - A	Barrel - C	Endcap - A	Endcap - C
Layer/Disk 0	384	384	184	184
Layer/Disk 1	480	480	264	264
Layer/Disk 2	576	576	264	264
Layer/Disk 3	672	672	264	264
Disk 4	-	-	264	264
Disk 5	-	-	264	264
Disk 6	-	-	184	184
Disk 7	-	-	184	184
Disk 8	-	-	104	104
total	2112	2112	1976	1976

Table 14: Number of modules for SCT readout. Note that SCT doublet layers are counted individually, and the numbers shown in the table are identical to those of input fibers to 90 SCT RODs. 4088 modules (8176 layers) are used in total.



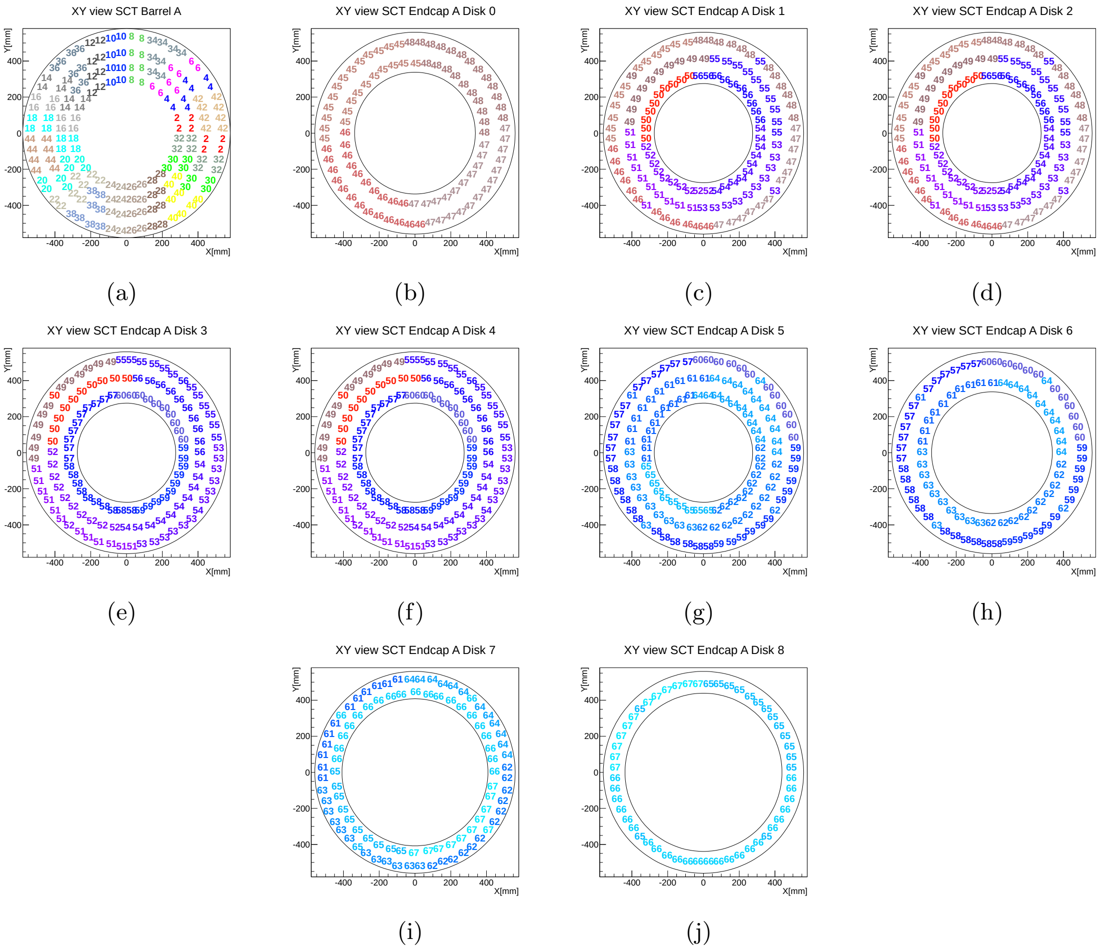
(a) (b) (c) (d)

Figure 28: (a) Pixel A Barrel (3 layers), (b) (c) (d) Pixel A Endcap 3 Disks.



(a) (b) (c) (d)

Figure 29: The RODs coverage for Pixel modules in C-sides are shown in x-y plane. (a) Pixel C Barrel (3 layers), (b) (c) (d) Pixel C Endcap 3 Disks.


XY view SCT Barrel A

XY view SCT Endcap A Disk 0

XY view SCT Endcap A Disk 1

XY view SCT Endcap A Disk 2

XY view SCT Endcap A Disk 3

XY view SCT Endcap A Disk 4

XY view SCT Endcap A Disk 5

XY view SCT Endcap A Disk 6

XY view SCT Endcap A Disk 7

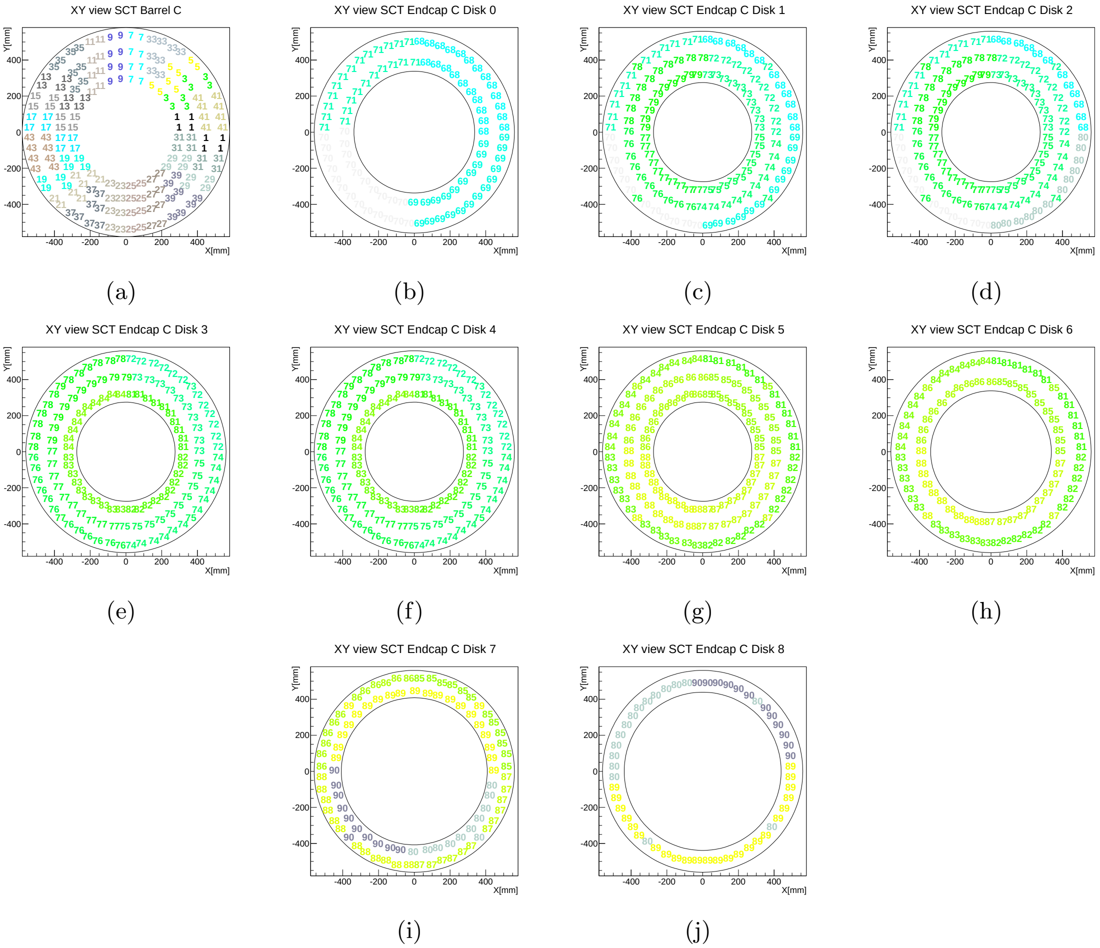
XY view SCT Endcap A Disk 8

(a) (b) (c) (d)

(e) (f) (g) (h)

(i) (j)

Figure 30: (a) SCT A Barrel (4 lures), (b) - (J) SCT A Endcap 9 Disks.



(a) (b) (c) (d)

(e) (f) (g) (h)

(i) (j)

Figure 31: (a) SCT C Barrel (4 layers), (b) - (J) SCT C Endcap 9 Disks.

A.2 FTK η-ϕ Tower Partitioning

The FTK system divides the detector volume into 64 symmetric sub-regions according to the (η, ϕ) of the silicon module (Figures 2(a) and 2(b)).

In ϕ the detector is divided into 16 sectors of $360^{\circ}/16 = 22.5^{\circ}$ with 10° overlap. Table A.2 describes the FTK ϕ sector boundaries (refer to Figure 2(a) for the sector numbering convention). The overlap is designed for variation of curvature for $p_T > 1$ GeV, and multiple scattering.

FTK ϕ Sector	0	1	2	3	4	5	6	7
Start ϕ	-22.5°	0°	22.5°	45°	67.5°	90°	112.5°	135°
End ϕ	10°	32.5°	55°	77.5°	100°	122.5°	145°	167.5°

FTK ϕ Sector	8	9	10	11	12	13	14	15
Start ϕ	157.5°	180°	202.5°	225°	247.5°	270°	292.5°	315°
End ϕ	190°	212.5°	235°	257.5°	280°	302.5°	325°	347.5°

In η the detector is divided into four intervals with overlap due to the finite size of the beam's luminous region in z (Figure 2(b) 32). The boundary between barrel and end-cap towers is based on the edge of the most outer layer of the SCT barrels, where $\cos\theta \sim 35^{\circ}(-\sim 35^{\circ})$ and $\eta \sim 1.17(-1.17)$ with respect to the detector origin. Many modules located at Pixel barrel layers are assigned to the end-cap FTK towers. Since the boundary overlap is designed to cover collisions at -120 mm $< z < 120$ mm, then the boundary is defined with respect to $z_0 = -120$ mm and $z_0 = 120$ mm. Modules satisfying $\eta < -1.3$ with respect to $z_0 = 120$ mm (black line) are assigned to "C-Side Endcap", $\eta > -1.03$ with respect to $z_0 = -120$ mm and $\eta < -0.23$ with respect to $z_0 = 120$ mm (red lines) are assigned to "C-Side Barrel" towers, $\eta > 0.23$ with respect to $z_0 = -120$ mm and $\eta < 1.03$ with respect to $z_0 = 120$ mm (green lines) are assigned to "A-Side Barrel" towers, and $\eta > 1.3$ with respect to $z_0 = -120$ mm (blue line) are assigned to "A-Side Endcap" towers. Significant numbers of modules are shared by neighboring towers in the inner layers.

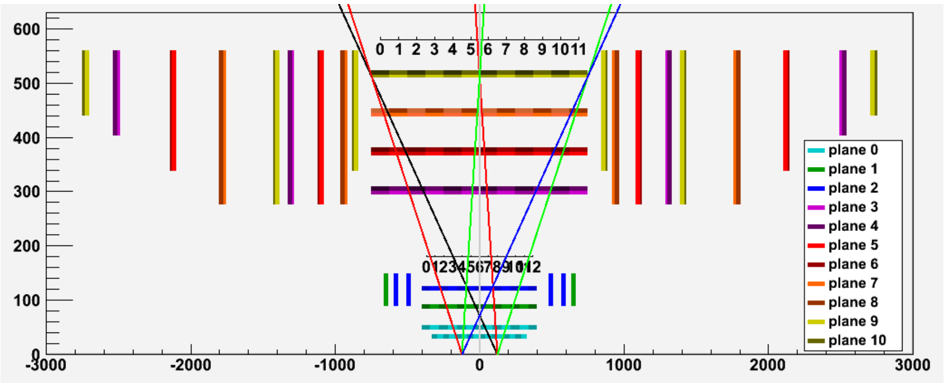


Figure 32: FTK Tower η boundaries.

Appendix B AdvancedTCA Hardware Overview

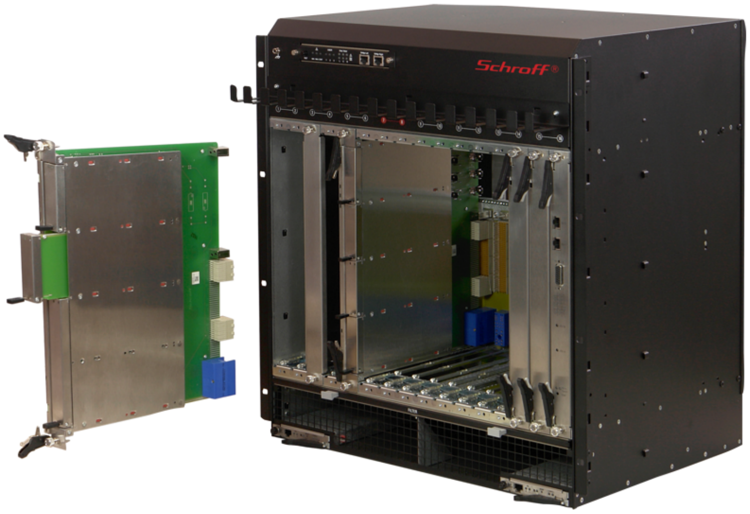


Figure 33: An ATCA board and 14-slot shelf.

Virtually every component in the ATCA shelf is a “field replaceable unit” (FRU) which means it may be replaced without powering down the shelf. Boards, fans, power entry modules, and shelf manager boards are *hot-swappable* and redundant. From the ground up ATCA has been designed for high availability operation.

B.1 Shelf

Boards are inserted into the ATCA shelf slots. Large shelf units contain 14 slots in a vertical configuration; smaller shelf units generally orient the blades horizontally. A typical 14 slot ATCA shelf is shown in Figure 33. For our application a 14 slot shelf will be used.

Each board is 8U (322.25mm) by 280mm deep. The width of each slot is considerably wider than VME at 30mm, which allows for taller components such as connectors, mezzanine cards, power converters, and large capacitors.

B.2 Backplane

PICMG 3.0 specifies three backplane connector zones. Zone-1 is near the bottom of the board and this connector is used for redundant 48VDC power and Intelligent Platform Management Controller (IPMC) management signals.

High speed data communication between boards occurs on the Zone-2 connectors. A few clocks and other synchronization signals are bused to all slots in the shelf, however a vast majority of the Zone-2 connections are point to point high speed serial links. ATCA is often described as “protocol agnostic” which means that the PICMG 3.0 specification simply describes the physical and electrical characteristics of these connections. The high speed serial data protocol is user defined. Zone-2 is comprised of two type of connections: the *Base Interface* and *Fabric Interface*.

The Base Interface is wired as dual star topology. There are two redundant hub slots in logical slots 1 and 2. Each hub slot has a direct connection to every other slot in the shelf. The Base Interface protocol is TCP/IP over Ethernet (100BASE-T or 1000BASE-T) and is intended for out of band management operations such as board control and monitoring.

Figure 34: A Shelf Manager board.

High speed data transfers take place on the Zone-2 Fabric Interface. The Fabric Interface is available in full-mesh, dual-star, dual-dual-star and replicated-mesh topologies. The Data Formatter will use the full-mesh configuration, which features eight 100Ω differential signal pairs (4 RX pairs and 4 TX pairs) between each slot. Each differential signal pair is rated for speeds up to 10Gb/s (high-performance “40G” backplanes) or up to 3Gbps (“standard” backplanes made with FR4 material).

The final backplane region is the Zone-3 area at top of the backplane. Connectors in the Zone-3 area are intended for passing data from the front board to the rear transition module (RTM). There is no backplane in this zone; rather the front board connectors mate directly with the connectors on the RTM. In the original PICMG 3.0 standard this zone was user-defined. The new PICMG 3.8 standard [6] defines the Zone-3 area as having three ADF style connectors for high speed serial data and a small blue ridiculously expensive connector for power and management.

B.3 Backplane Clocks

Data transfers across the Fabric Interface use high speed serial links, which embed the clock in the data stream (standard 8B/10B encoding). Therefore, data transfers across the Fabric Interface are inherently self-clocking and do not require a separate master clock.

PICMG 3.0 does however define a set of clocks for general synchronization and timing. The Synchronization Clock Interface consists of six differential clock lines bused to all boards. ATCA boards may listen or drive any of these clock lines, as negotiated by the electronic keying feature of the IPMI management interface. Two of the four clocks (CLK3a and CLK3b) are user-defined and may be used for experiment synchronization and various DAQ control signals.

B.4 Update Interface

The Update Channel interface consists of 10 differential pairs in a point to point connection between two boards. This interface is optional and varies by backplane manufacturer. This interface is not used by the Data Formatter.

B.5 Intelligent Platform Management Interface

ATCA hardware incorporates an Intelligent Platform Management Interface (IPMI), which is required on all shelf components. Through this interface the Shelf Manager card can query sensors and control shelf components. For example, if the Shelf Manager detects an over-temperature condition on a board then the fan speed may be increased or the board could be powered down.

High availability operation is archived through redundancy built into the IPMI specification. Each shelf has dual redundant Shelf Manager cards, one is shown in Figure 34. If the master Shelf Manager fails then control automatically transferred to the slave unit. Like other ATCA components the Shelf Manager boards support hot swap operation. The heart of the Shelf Manager card is a single board computer running Linux. It is possible to log into the Shelf Manager through telnet, SSH, or a serial terminal; however the user will typically interact with Shelf Manager through the web interface. An Ethernet port is located on the front panel of the Shelf Manager, and there are also jumpers to connect the Shelf Manager to the backplane Base Fabric network as well.

Shelf Manager cards communicate over the dual redundant Intelligent Platform Management Bus (IPMB), which uses the I^2C protocol for the base layer. Typically the following sensors are monitored: temperature, fan speed, voltage and current, and board handle switch status. The board or FRU must also report back a description, serial number, manufacturer name, part number, and various hardware, firmware, and software version numbers. The IPMI protocols are fairly complex and a microcontroller (Intelligent Platform Management Controller, or IPMC) must be used.

Hot swap operation is implemented by monitoring the status of a microswitch in the board handle and controlling the DC-DC converters on the board. Removing a component from an ATCA shelf requires following a simple procedure which involves opening the ejector handle slightly, then watching the blue HS LED until it indicates that the board has completed the shutdown procedure, then the board may be removed safely from the system.

The IPMI specification supports firmware downloads to the IPMC microcontrollers. Downloading large FPGAs may be possible over the IPMI bus but this operation would be extremely slow given the I^2C transmission speeds.

IPMC reference designs are available commercially available. The reference designs fully implement the latest IPMI specification and have been debugged and technical support is provided. However the commercial reference designs are strictly licensed, closed source, and discourage collaboration by requiring non-disclosure agreements. As an alternative, several HEP laboratories have produced open-source designs [8] for IPMC controllers.

B.6 Network Connectivity

All ATCA backplanes support a Dual Star Base Interface network which is based on Gigabit Ethernet. The Base Interface is generally used for high speed board management tasks such as board control and status and downloading firmware. Logical slots 1 and 2 form the hubs of two separate star Ethernet networks. Logical slots 3 through 14 each have two Base Interface ports: channel 1 connects to the slot 1 network and channel 2 connect to the slot 2 network. The Base Interface network uses 1000BASE-T Ethernet, which consists of 4 bidirectional signal pairs (similar to a “CAT5” cable). A small transformer and a PHY chip are required to interface to 1000BASE-T network.

The backplane Fabric Interface may also be used for high speed Ethernet communication between hubs and boards. The PICMG 3.1 specification details how 1000BASE-BX is used for Gigabit Ethernet (or 10G Ethernet) communication between node and hub boards. 1000BASE-BX consists of two differential pairs running at 1.25Gbps and is 8B/10B encoded. 1000BASE-BX may be directly interfaced with an serial transceiver in an FPGA, no dedicated Ethernet PHY chip is required. The PICMG 3.1 specification also describes 10G Ethernet connections, which involve all four backplane ports in the channel “bonded” (synchronized) and running at 3.125Gbps.

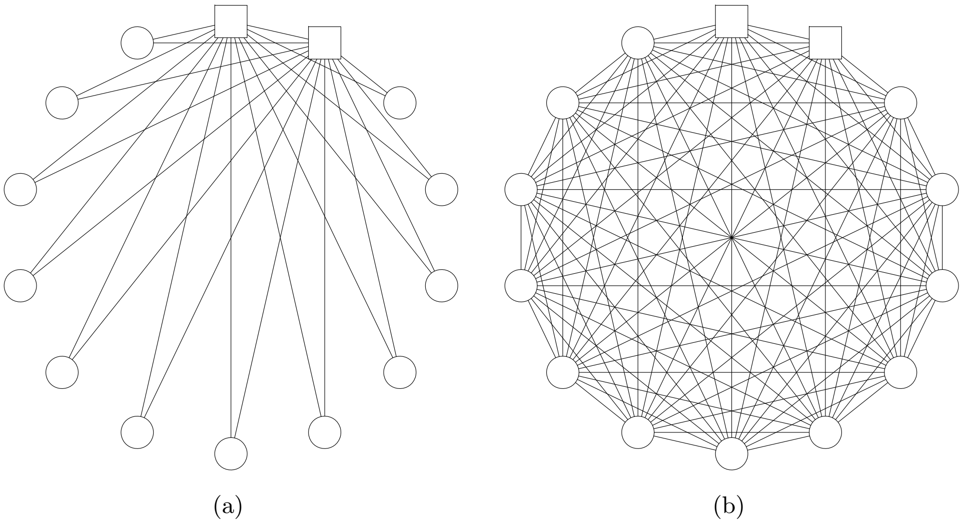


Figure 35: ATCA Backplane Connections. The Base Interface (a) consists of two independent 1000BASE-T Ethernet star networks. The Full Mesh Fabric Interface (b) connects all slots with bidirectional channels rated for up to 40Gbps. The Fabric Interface is “protocol agnostic”.

B.7 Hub and System Controller Boards

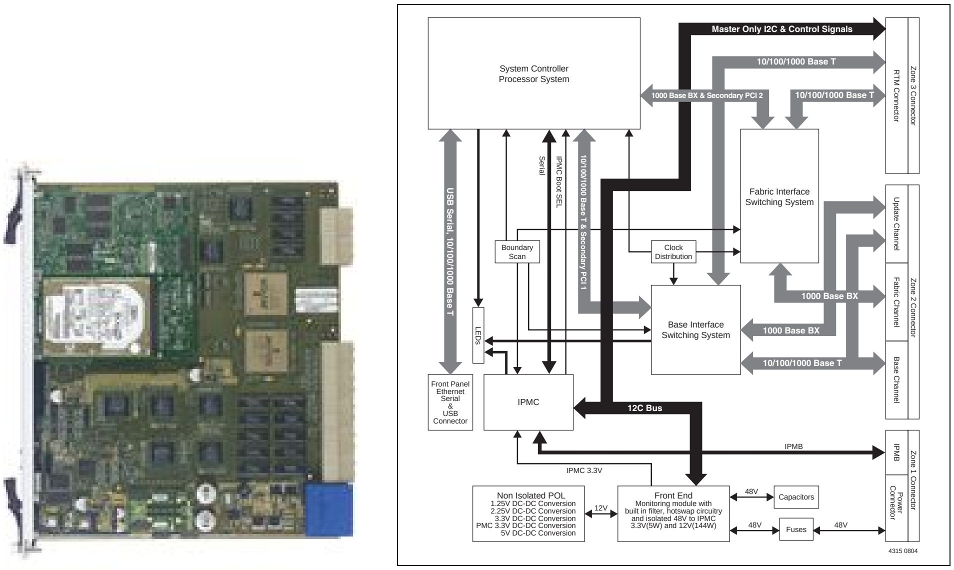
ATCA System Controller boards combine a CPU, memory, hard drive and several Gigabit Ethernet network switches (Figure 36). Hub boards (or “switch blades”) generally omit the CPU, memory and hard drive to reduce cost. All board types will at a minimum support a Gigabit Ethernet Base Interface switch. More advanced boards may provide higher speeds (10GbE), high performance multi-core CPUs, and support for a switch on the Fabric Interface.

B.8 Power Supply

ATCA evolved out of the telecommunications industry, which has historically used a -48VDC power distribution. The shelf incorporates redundant power entry modules, each of which has a connection for the -48V supply and return lines. ATCA hardware supports up to 200W per slot.

The Power supplies are also redundant. A common configuration is a 1U rackmount chassis with three power supplies which operate in an “N+1” redundant mode. Output diodes and special circuitry is employed to implement dynamic load sharing and hot swap capability. Therefore, a failed supply can be shutdown or replaced without interrupting shelf operation.

Our experience with 48VDC “N+1” redundant power supplies has been extremely positive. For instance, when a power supply fails it is shut down and the other supplies automatically take up the load without interruption. Then, during a normally scheduled controlled access the faulty supply is simply replaced. Local voltage regulation on the board (with isolated DC-DC converters) is reliable and eliminates the need for remote sensing which is common on low-voltage high-current power supplies. Compared to a large low-voltage high-current power supply a board mounted DC-DC converter can simply react faster to the highly dynamic load often associated with high performance FPGAs, resulting in improved voltage regulation.



(a) (b)

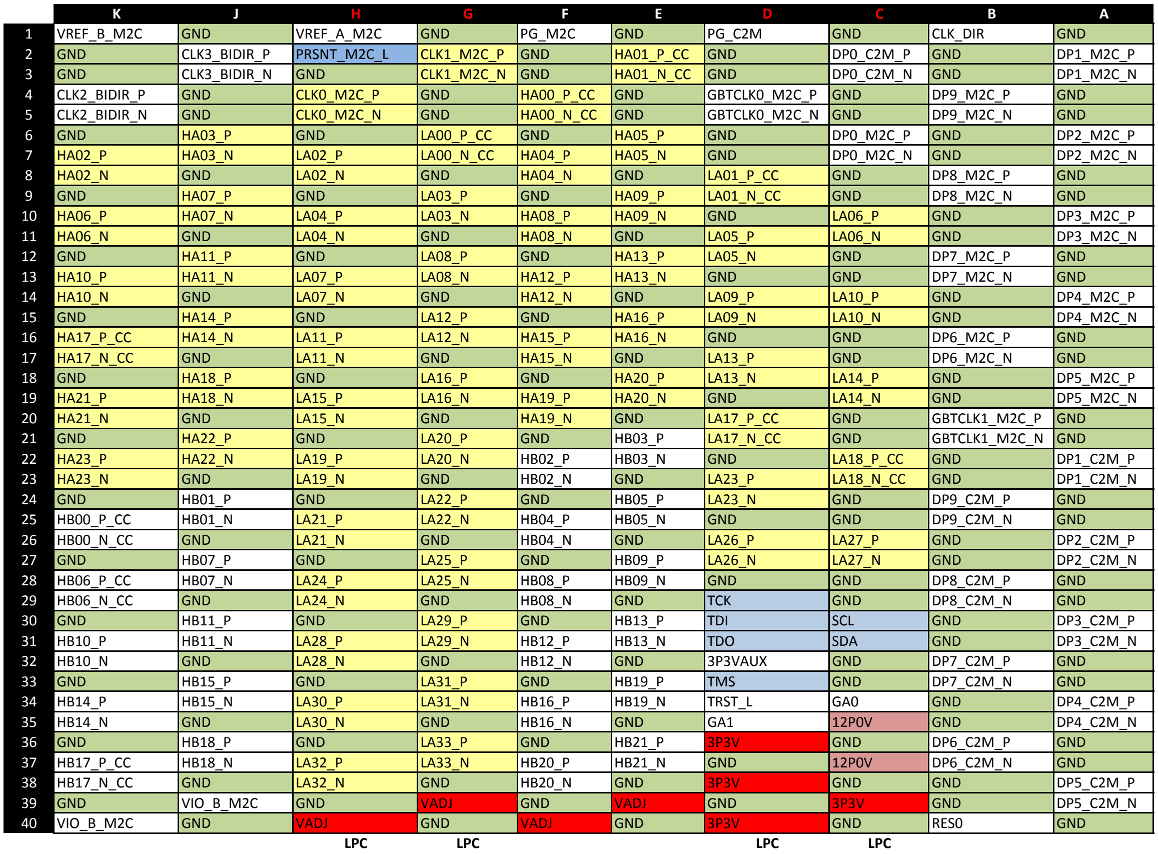
Figure 36: The Motorola ATCA F101 System Controller and "Switch Blade". This board combines a single-core CPU with a Gigabit Ethernet Base Interface switch (1000BASE-T). The CPU also connects to a Gigabit Ethernet Fabric Interface switch (1000BASE-BX, PICMG 3.1).

Appendix C Mezzanine Card Details

C.1 FMC Pinout

Notes:

1. LPC connectors populate rows C, D, G and H (160 pins).
2. HPC connectors populate all rows (400 pins).
3. The Data Formatter board supports all colored pins. All other pins are not connected.
4. The Data Formatter board sets VADJ to +3.3VDC.
5. PRSNT_M2C_L should be grounded on the mezzanine card.
6. SCL and SDA have 4.7k pullup resistors to +3.3V on the Data Formatter board.
7. Refer to the Samtec website [15] for HPC and LPC connector datasheets and drawings.



	K	J	H	G	F	E	D	C	B	A
1	VREF_B_M2C	GND	VREF_A_M2C	GND	PG_M2C	GND	PG_C2M	GND	CLK_DIR	GND
2	GND	CLK3_BIDIR_P	PRSNT_M2C_L	CLK1_M2C_P	GND	HA01_P_CC	GND	DP0_C2M_P	GND	DP1_M2C_P
3	GND	CLK3_BIDIR_N	GND	CLK1_M2C_N	GND	HA01_N_CC	GND	DP0_C2M_N	GND	DP1_M2C_N
4	CLK2_BIDIR_P	GND	CLK0_M2C_P	GND	HA00_P_CC	GND	GBTCLK0_M2C_P	GND	DP9_M2C_P	GND
5	CLK2_BIDIR_N	GND	CLK0_M2C_N	GND	HA00_N_CC	GND	GBTCLK0_M2C_N	GND	DP9_M2C_N	GND
6	GND	HA03_P	GND	LA00_P_CC	GND	HA05_P	GND	DP0_M2C_P	GND	DP2_M2C_P
7	HA02_P	HA03_N	LA02_P	LA00_N_CC	HA04_P	HA05_N	GND	DP0_M2C_N	GND	DP2_M2C_N
8	HA02_N	GND	LA02_N	GND	HA04_N	GND	LA01_P_CC	GND	DP8_M2C_P	GND
9	GND	HA07_P	GND	LA03_P	GND	HA09_P	LA01_N_CC	GND	DP8_M2C_N	GND
10	HA06_P	HA07_N	LA04_P	LA03_N	HA08_P	HA09_N	GND	LA06_P	GND	DP3_M2C_P
11	HA06_N	GND	LA04_N	GND	HA08_N	GND	LA05_P	LA06_N	GND	DP3_M2C_N
12	GND	HA11_P	GND	LA08_P	GND	HA13_P	LA05_N	GND	DP7_M2C_P	GND
13	HA10_P	HA11_N	LA07_P	LA08_N	HA12_P	HA13_N	GND	GND	DP7_M2C_N	GND
14	HA10_N	GND	LA07_N	GND	HA12_N	GND	LA09_P	LA10_P	GND	DP4_M2C_P
15	GND	HA14_P	GND	LA12_P	GND	HA16_P	LA09_N	LA10_N	GND	DP4_M2C_N
16	HA17_P_CC	HA14_N	LA11_P	LA12_N	HA15_P	HA16_N	GND	GND	DP6_M2C_P	GND
17	HA17_N_CC	GND	LA11_N	GND	HA15_N	GND	LA13_P	GND	DP6_M2C_N	GND
18	GND	HA18_P	GND	LA16_P	GND	HA20_P	LA13_N	LA14_P	GND	DP5_M2C_P
19	HA21_P	HA18_N	LA15_P	LA16_N	HA19_P	HA20_N	GND	LA14_N	GND	DP5_M2C_N
20	HA21_N	GND	LA15_N	GND	HA19_N	GND	LA17_P_CC	GND	GBTCLK1_M2C_P	GND
21	GND	HA22_P	GND	LA20_P	GND	HB03_P	LA17_N_CC	GND	GBTCLK1_M2C_N	GND
22	HA23_P	HA22_N	LA19_P	LA20_N	HB02_P	HB03_N	GND	LA18_P_CC	GND	DP1_C2M_P
23	HA23_N	GND	LA19_N	GND	HB02_N	GND	LA23_P	LA18_N_CC	GND	DP1_C2M_N
24	GND	HB01_P	GND	LA22_P	GND	HB05_P	LA23_N	GND	DP9_C2M_P	GND
25	HB00_P_CC	HB01_N	LA21_P	LA22_N	HB04_P	HB05_N	GND	GND	DP9_C2M_N	GND
26	HB00_N_CC	GND	LA21_N	GND	HB04_N	GND	LA26_P	LA27_P	GND	DP2_C2M_P
27	GND	HB07_P	GND	LA25_P	GND	HB09_P	LA26_N	LA27_N	GND	DP2_C2M_N
28	HB06_P_CC	HB07_N	LA24_P	LA25_N	HB08_P	HB09_N	GND	GND	DP8_C2M_P	GND
29	HB06_N_CC	GND	LA24_N	GND	HB08_N	GND	TCK	GND	DP8_C2M_N	GND
30	GND	HB11_P	GND	LA29_P	GND	HB13_P	TDI	SCL	GND	DP3_C2M_P
31	HB10_P	HB11_N	LA28_P	LA29_N	HB12_P	HB13_N	TDO	SDA	GND	DP3_C2M_N
32	HB10_N	GND	LA28_N	GND	HB12_N	GND	3P3VAUX	GND	DP7_C2M_P	GND
33	GND	HB15_P	GND	LA31_P	GND	HB19_P	TMS	GND	DP7_C2M_N	GND
34	HB14_P	HB15_N	LA30_P	LA31_N	HB16_P	HB19_N	TRST_L	GA0	GND	DP4_C2M_P
35	HB14_N	GND	LA30_N	GND	HB16_N	GND	GA1	12P0V	GND	DP4_C2M_N
36	GND	HB18_P	GND	LA33_P	GND	HB21_P	3P3V	GND	DP6_C2M_P	GND
37	HB17_P_CC	HB18_N	LA32_P	LA33_N	HB20_P	HB21_N	GND	12P0V	DP6_C2M_N	GND
38	HB17_N_CC	GND	LA32_N	GND	HB20_N	GND	3P3V	GND	GND	DP5_C2M_P
39	GND	VIO_B_M2C	GND	VADJ	GND	VADJ	GND	3P3V	GND	DP5_C2M_N
40	VIO_B_M2C	GND	VADJ	GND	VADJ	GND	3P3V	GND	RES0	GND
			LPC	LPC			LPC	LPC		

Table 15: The FMC Connector Pinout. The Data Formatter board supports a subset of High Pin Count (HPC) connections.

C.2 FMC Connector

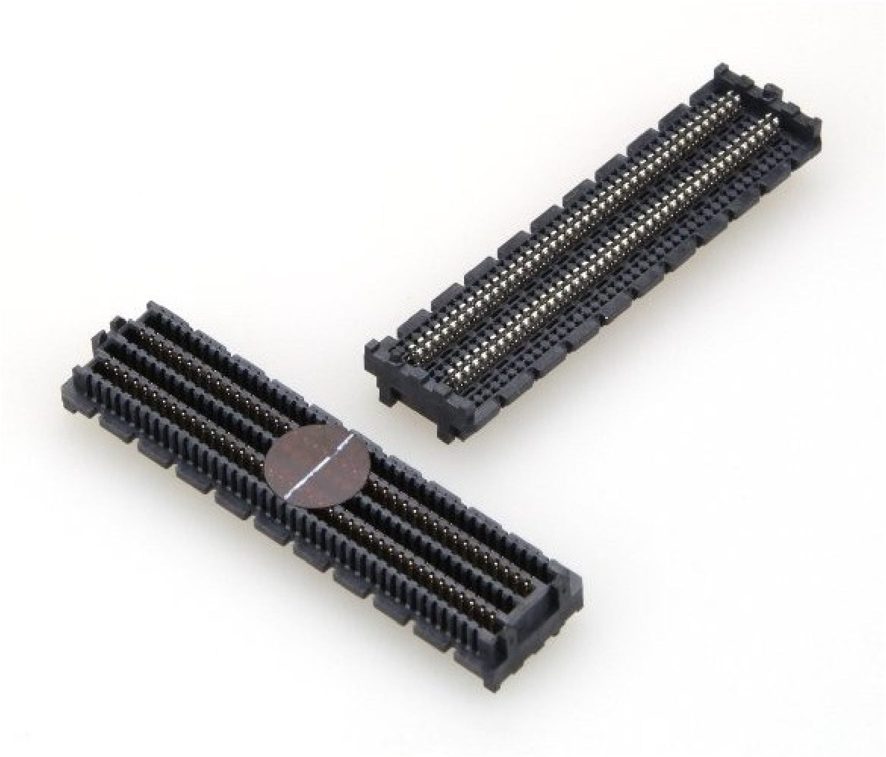


Figure 37: The male FMC connector, which is soldered on the mezzanine card. The male connector is available in 8.5mm and 10mm stack height versions. The LPC version is shown here.

C.3 Dimensions

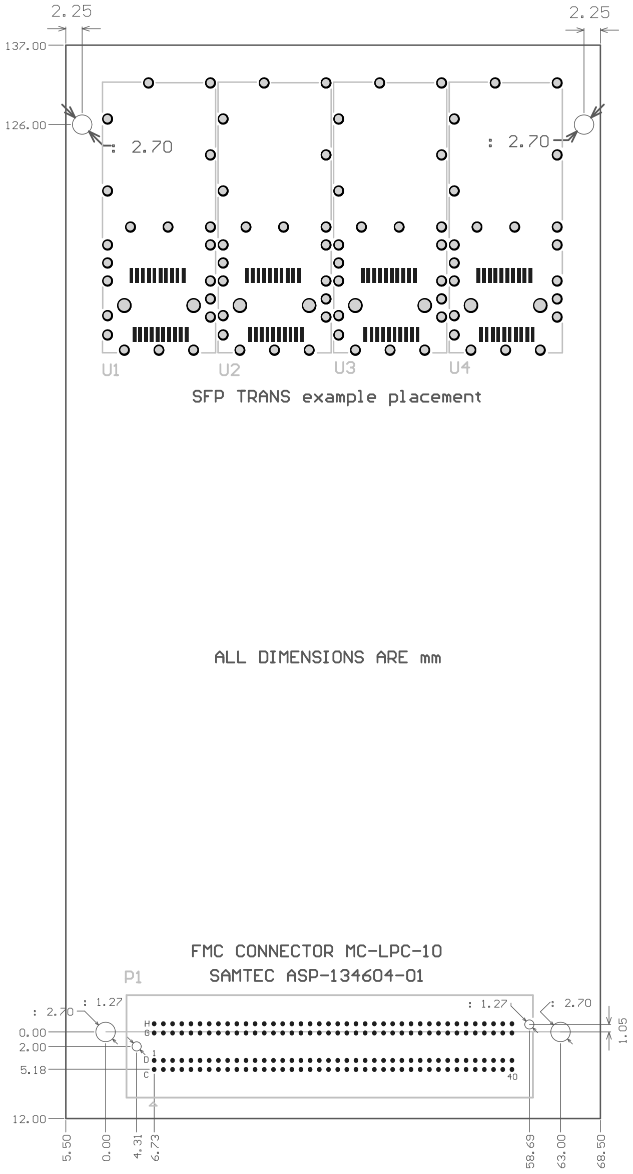


Figure 38: The mezzanine card as viewed looking at the “component side” of the board, as in Figure 13. A low pin count (LPC) connector is shown here.

C.4 Bezel Detail

Mezzanine cards use the standard CMC/PMC bezel, which measures 10mm high by 74mm wide. When installed on a Data Formatter board the bezel will be flush with the front panel. Although space is tight, up to four SFP optical transceivers may be mounted on a mezzanine card.

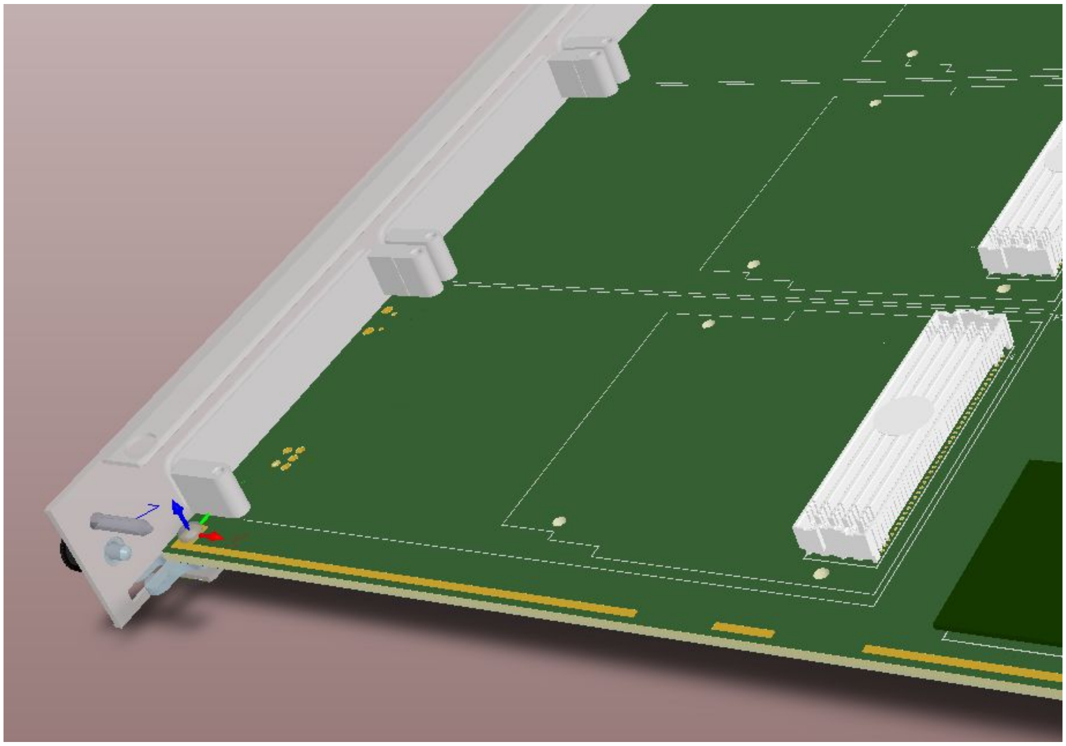


Figure 39: The Data Formatter board detail showing the mezzanine card FMC connector, bezel and front panel.

Appendix D RTM Details

D.1 Transceiver Numbering

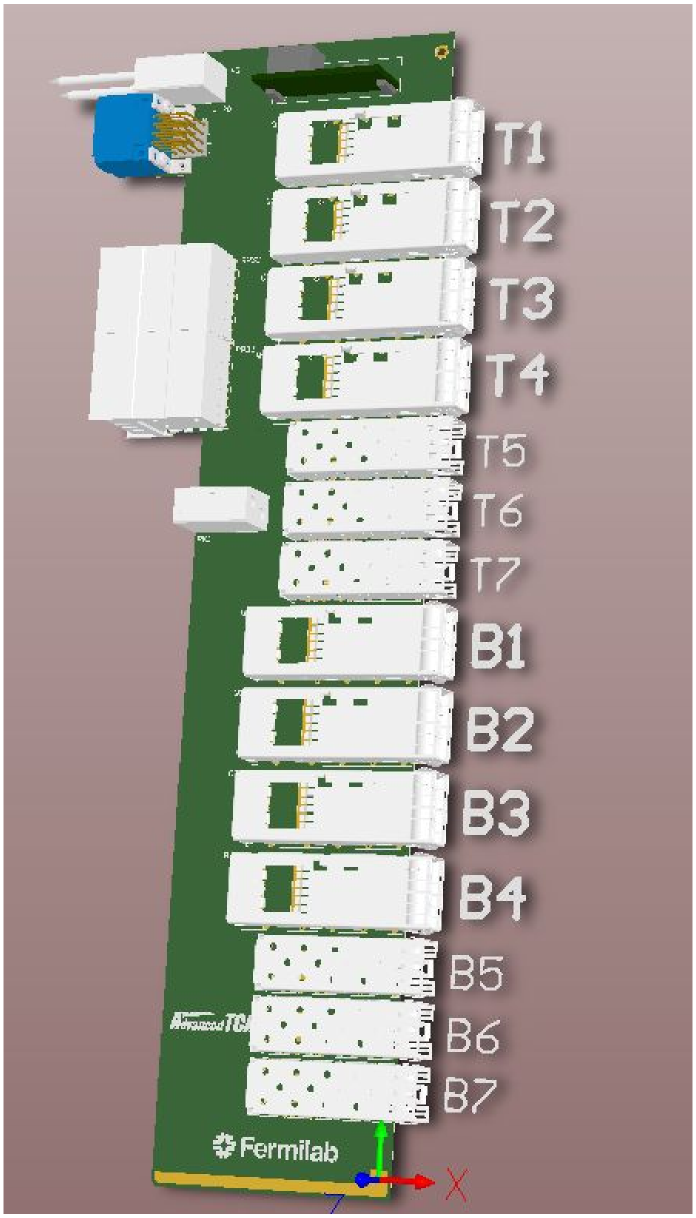


Figure 40: RTM transceiver identifiers

D.2 Transceiver Connector Port Assignments

Connector	Port Number	RTM Transceiver	Transceiver Channel
RP32	19	QSFP T1	1
RP32	18	QSFP T1	2
RP32	17	QSFP T1	3
RP32	16	QSFP T1	4
RP32	15	QSFP T2	1
RP32	14	QSFP T2	2
RP32	13	QSFP T2	3
RP32	12	QSFP T2	4
RP32	11	QSFP T3	1
RP32	10	QSFP T3	2
RP32	9	QSFP T3	3
RP32	8	QSFP T3	4
RP32	7	QSFP T4	1
RP32	6	QSFP T4	2
RP32	5	QSFP T4	3
RP32	4	QSFP T4	4
RP32	3	SFP T5	n/a
RP32	2	SFP T6	n/a
RP32	1	SFP T7	n/a
RP32	0	(reserved)	n/a

Table 16: Port Assignments for Zone-3A connectors J32/RP32. These transceivers connect to the top FPGA.

Connector	Port Number	RTM Transceiver	Transceiver Channel
RP33	19	QSFP B1	1
RP33	18	QSFP B1	2
RP33	17	QSFP B1	3
RP33	16	QSFP B1	4
RP33	15	QSFP B2	1
RP33	14	QSFP B2	2
RP33	13	QSFP B2	3
RP33	12	QSFP B2	4
RP33	11	QSFP B3	1
RP33	10	QSFP B3	2
RP33	9	QSFP B3	3
RP33	8	QSFP B3	4
RP33	7	QSFP B4	1
RP33	6	QSFP B4	2
RP33	5	QSFP B4	3
RP33	4	QSFP B4	4
RP33	3	SFP B5	n/a
RP33	2	SFP B6	n/a
RP33	1	SFP B7	n/a
RP33	0	(reserved)	n/a

Table 17: Port Assignments for Zone-3A connectors J33/RP33. These transceivers connect to the bottom FPGA.

D.3 Rear Panel Detail

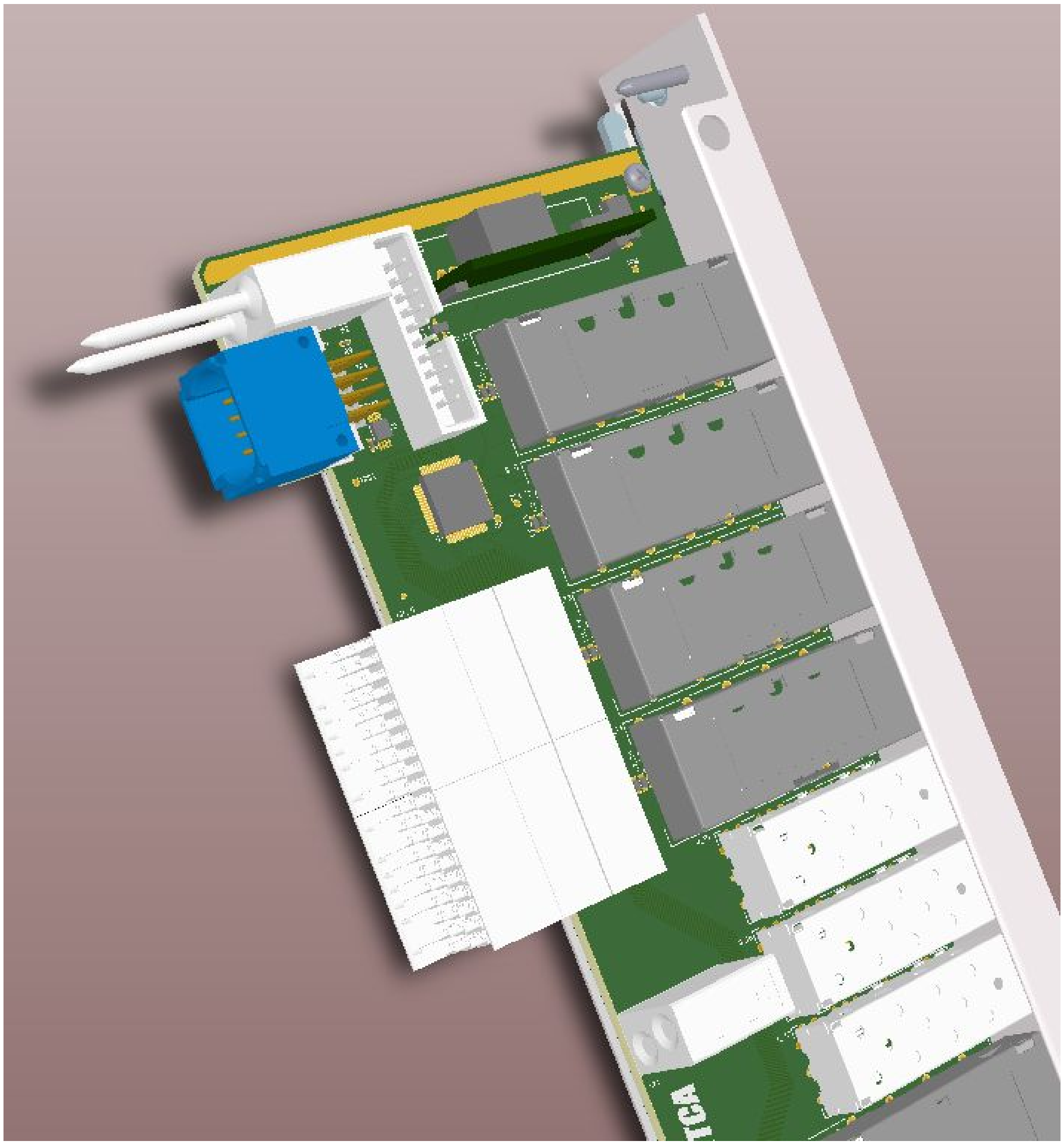


Figure 41: RTM detail showing the PICMG 3.8 management connector (blue), the high speed ADF connectors RP32 and RP33. A small ARM microcontroller and debug/programming header is shown. RTM transceivers are powered by the 3.3V 20A switching regulator shown at the top of the card.

Appendix E Initial Firmware Study

Firmware development is ongoing and we have the Xilinx Kintex development board [10] on hand. In the following sections we discuss general firmware development, FPGA performance tests, and present some initial thoughts about a packet switch algorithm that permits high speed switching the in Data Formatter FPGAs.

E.1 Serial Transceivers

Xilinx Kintex FPGAs use internal multi-gigabit serial transceivers, commonly referred to as MGTs or GTX modules. In the highest speed grade devices the GTX transceivers support data rates up to 12.5Gbps.

The SLINK firmware was originally developed for FPGAs with external SERDES chips. Subsequent incarnations used internal SERDES components but have targeted ALTERA FPGAs primitives. We have successfully adapted the SLINK firmware to use Kintex GTX transceivers.

Serial links internal to the Data Formatter system will likely use the Aurora protocol. Aurora is a light-weight low-overhead protocol which supports basic flow control and data integrity checking. We are currently testing the Aurora-GTX firmware core designs.

FPGA I/O pin resources have evolved rapidly over the past few years. In the Kintex family each I/O pin contains a serializer (OSERDES2) and de-serializer (ISERDES2). These components allow for serial communications up to 1.6Gbps per LVDS pin pair without using any GTX resources. We intend to use these I/O serializer functions for high speed communication between the Data Formatter FPGAs and the Mezzanine Cards.

E.2 Packet Switch Preliminaty Consideration

There is no central switch or router in the Data Formatter system. Data packets are routed between FPGAs in a distributed routing scheme. Although the connections between FPGAs are numerous (see Figure 6) not all FPGAs are directly connected with point to point links. Therefore it may be necessary for the FPGA routing logic to support “route through” or packet re-transmission. For example, a packet entering the FPGA on Fabric Interface channel 3 may be re-transmitted on the local-bus, or it may go to another shelf over the inter-shelf link, or even another Fabric Interface channel.

As a data packet arrives in the FPGA (from any input) its module-ID is evaluated against a routing table and the destination FPGAs output (or outputs) is calculated. The packet is then sent through a possible switch network implemented in the FPGA logic, as shown in Figure 42. A few switch input ports are reserved for the mezzanine cards and a few switch output ports are reserved for the formatting logic, which concatenates and repackages the data before transmission to the AUX cards and SSB boards.

A simple “crossbar” network switch architecture is inherently blocking and would cause latency to necessarily increase. An alternative switch architecture based on a banyan network is shown in Figure 43. The banyan network consists of switch nodes which include local buffering in FIFOs. We are currently simulating the performance of such a switch and evaluating the FPGA logic resources required.

Our data volume analysis (see Section 8) shows that not all links are equally utilized. Under-utilized links may be used more effectively by tweaking the routing tables in each FPGA. In some cases, an alternate or less direct route involving lightly-used links may be more advantageous in terms of latency and overall throughput. A switch architecture that supports “route through” allows for link-utilization optimization. We are currently evaluating the feasibility of such optimization.

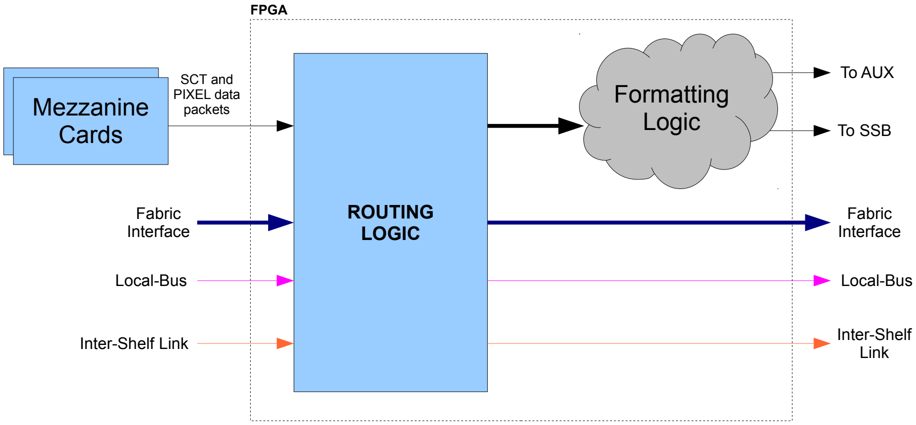


Figure 42: Overview of the FPGA routing firmware. Data packets containing SCT and Pixel hits and clusters are output from a pair of mezzanine cards. Data packets also arrive on the Fabric Interface channels, the local-bus link, and inter-shelf link. It is possible for the firmware to support "route through" which enables an incoming packet to be re-transmitted over any output.

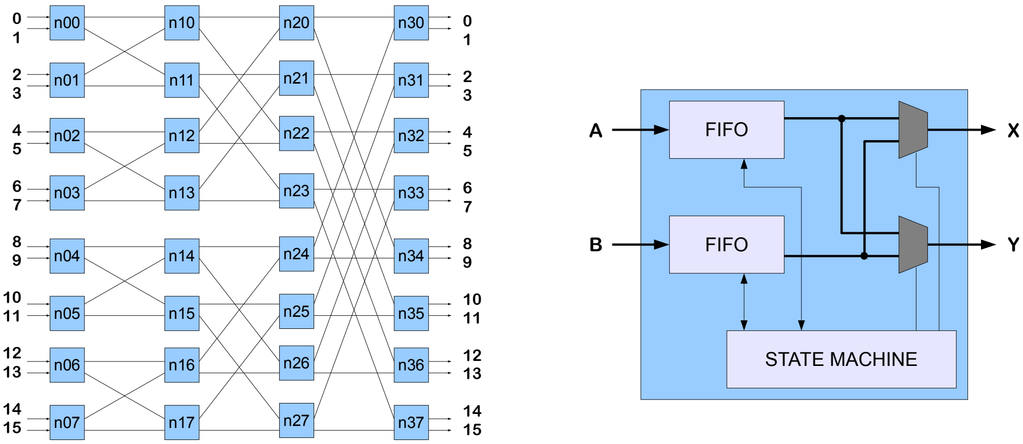


Figure 43: A possible 16x16 port network switch implemented in the FPGA logic. The switch nodes (right) contains FIFOs for internal data buffering.

Appendix F ROD IDs

ROD ID numbers used in Figure 28, Figure 29, Figure 30 and Figure 31 are summarized for 222 RODs in the current existing ATLAS DAQ system [17, 18].

F.1 132 Pixel RODs

ID	ROD Address	Side	Barrel/Endcap
1	0x00130005	C	Barrel
2	0x00130006	A	Barrel
3	0x00130007	A	Barrel
4	0x00130008	C	Barrel
5	0x00130009	C	Barrel
6	0x00130010	A	Barrel
7	0x00130011	A	Barrel
8	0x00130012	C	Barrel
9	0x00130014	C	Barrel
10	0x00130015	A	Barrel
11	0x00130016	A	Barrel
12	0x00130017	C	Barrel
13	0x00130018	C	Barrel
14	0x00130019	A	Barrel
15	0x00130020	A	Barrel
16	0x00130021	C	Barrel
17	0x00130105	C	Barrel
18	0x00130106	A	Barrel
19	0x00130107	A	Barrel
20	0x00130108	C	Barrel
21	0x00130109	C	Barrel
22	0x00130110	A	Barrel
23	0x00130111	A	Barrel
24	0x00130112	C	Barrel
25	0x00130114	C	Barrel
26	0x00130115	A	Barrel
27	0x00130116	A	Barrel
28	0x00130117	C	Barrel
29	0x00130118	C	Barrel
30	0x00130119	A	Barrel
31	0x00130120	A	Barrel
32	0x00130121	C	Barrel
33	0x00130307	C	Barrel
34	0x00130308	A	Barrel
35	0x00130309	A	Barrel
36	0x00130310	C	Barrel
37	0x00130311	C	Barrel
38	0x00130312	A	Barrel
39	0x00130314	A	Barrel
40	0x00130315	C	Barrel
41	0x00130316	C	Barrel
42	0x00130317	A	Barrel
43	0x00130318	A	Barrel
44	0x00130319	C	Barrel
45	0x00120205	C	Endcap
46	0x00120206	A	Endcap
47	0x00120207	C	Endcap
48	0x00120208	A	Endcap
49	0x00120209	C	Endcap
50	0x00120210	A	Endcap
51	0x00120211	C	Endcap
52	0x00120212	A	Endcap
53	0x00120214	C	Endcap
54	0x00120215	A	Endcap
55	0x00120216	C	Endcap
56	0x00120217	A	Endcap
57	0x00120218	C	Endcap
58	0x00120219	A	Endcap
59	0x00120220	C	Endcap
60	0x00120221	A	Endcap
61	0x00121609	C	Endcap
62	0x00121610	A	Endcap
63	0x00121611	C	Endcap
64	0x00121612	A	Endcap
65	0x00121614	C	Endcap
66	0x00121615	A	Endcap
67	0x00121616	C	Endcap
68	0x00121617	A	Endcap

ID	ROD Address	Side	Barrel/Endcap
69	0x00112405	A	Barrel
70	0x00112406	C	Barrel
71	0x00112407	A	Barrel
72	0x00112408	C	Barrel
73	0x00112409	A	Barrel
74	0x00112410	C	Barrel
75	0x00112411	A	Barrel
76	0x00112412	C	Barrel
77	0x00112414	A	Barrel
78	0x00112415	C	Barrel
79	0x00112416	A	Barrel
80	0x00112417	C	Barrel
81	0x00112418	A	Barrel
82	0x00112419	C	Barrel
83	0x00112420	A	Barrel
84	0x00112421	C	Barrel
85	0x00112505	C	Barrel
86	0x00112506	A	Barrel
87	0x00112507	C	Barrel
88	0x00112508	A	Barrel
89	0x00112509	C	Barrel
90	0x00112510	A	Barrel
91	0x00112511	A	Barrel
92	0x00112512	C	Barrel
93	0x00112514	A	Barrel
94	0x00112515	C	Barrel
95	0x00112516	A	Barrel
96	0x00112517	C	Barrel
97	0x00112518	A	Barrel
98	0x00112519	C	Barrel
99	0x00112520	A	Barrel
100	0x00112521	C	Barrel
101	0x00111705	A	Barrel
102	0x00111706	C	Barrel
103	0x00111707	A	Barrel
104	0x00111708	C	Barrel
105	0x00111709	A	Barrel
106	0x00111710	C	Barrel
107	0x00111711	A	Barrel
108	0x00111712	C	Barrel
109	0x00111714	A	Barrel
110	0x00111715	C	Barrel
111	0x00111716	A	Barrel
112	0x00111717	C	Barrel
113	0x00111718	A	Barrel
114	0x00111719	C	Barrel
115	0x00111720	A	Barrel
116	0x00111721	C	Barrel
117	0x00111805	A	Barrel
118	0x00111806	C	Barrel
119	0x00111807	A	Barrel
120	0x00111808	C	Barrel
121	0x00111809	A	Barrel
122	0x00111810	C	Barrel
123	0x00111811	A	Barrel
124	0x00111812	C	Barrel
125	0x00111814	A	Barrel
126	0x00111815	C	Barrel
127	0x00111816	A	Barrel
128	0x00111817	C	Barrel
129	0x00111818	A	Barrel
130	0x00111819	C	Barrel
131	0x00111820	A	Barrel
132	0x00111821	C	Barrel

F.2 90 SCT RODs

ID	ROD Address	Side	Barrel/Endcap
1	0x00220005	C	Barrel
2	0x00210005	A	Barrel
3	0x00220007	C	Barrel
4	0x00210007	A	Barrel
5	0x00220008	C	Barrel
6	0x00210008	A	Barrel
7	0x0022000a	C	Barrel
8	0x0021000a	A	Barrel
9	0x00220100	C	Barrel
10	0x00210100	A	Barrel
11	0x00220101	C	Barrel
12	0x00210101	A	Barrel
13	0x00220103	C	Barrel
14	0x00210103	A	Barrel
15	0x00220104	C	Barrel
16	0x00210104	A	Barrel
17	0x00220105	C	Barrel
18	0x00210105	A	Barrel
19	0x00220107	C	Barrel
20	0x00210107	A	Barrel
21	0x00220108	C	Barrel
22	0x00210108	A	Barrel
23	0x0022010a	C	Barrel
24	0x0021010a	A	Barrel
25	0x00220000	C	Barrel
26	0x00210000	A	Barrel
27	0x00220001	C	Barrel
28	0x00210001	A	Barrel
29	0x00220003	C	Barrel
30	0x00210003	A	Barrel
31	0x00220004	C	Barrel
32	0x00210004	A	Barrel
33	0x00220009	C	Barrel
34	0x00210009	A	Barrel
35	0x00220102	C	Barrel
36	0x00210102	A	Barrel
37	0x00220109	C	Barrel
38	0x00210109	A	Barrel
39	0x00220002	C	Barrel
40	0x00210002	A	Barrel
41	0x00220006	C	Barrel
42	0x00210006	A	Barrel
43	0x00220106	C	Barrel
44	0x00210106	A	Barrel
45	0x00230000	A	Endcap
46	0x00230100	A	Endcap
47	0x0023010b	A	Endcap
48	0x0023000a	A	Endcap
49	0x00230001	A	Endcap
50	0x00230002	A	Endcap
51	0x00230101	A	Endcap
52	0x00230102	A	Endcap
53	0x0023010a	A	Endcap
54	0x00230109	A	Endcap
55	0x00230009	A	Endcap
56	0x00230008	A	Endcap
57	0x00230003	A	Endcap
58	0x00230103	A	Endcap
59	0x00230108	A	Endcap
60	0x00230007	A	Endcap
61	0x00230004	A	Endcap
62	0x00230107	A	Endcap
63	0x00230104	A	Endcap
64	0x00230006	A	Endcap
65	0x00230105	A	Endcap
66	0x00230005	A	Endcap
67	0x00230106	A	Endcap
68	0x00240100	C	Endcap
69	0x00240000	C	Endcap
70	0x0024000b	C	Endcap
71	0x0024010a	C	Endcap
72	0x00240101	C	Endcap
73	0x00240102	C	Endcap
74	0x00240001	C	Endcap
75	0x00240002	C	Endcap
76	0x0024000a	C	Endcap
77	0x00240009	C	Endcap
78	0x00240109	C	Endcap
79	0x00240108	C	Endcap
80	0x00240005	C	Endcap
81	0x00240103	C	Endcap
82	0x00240003	C	Endcap
83	0x00240008	C	Endcap
84	0x00240107	C	Endcap
85	0x00240104	C	Endcap
86	0x00240106	C	Endcap
87	0x00240004	C	Endcap
88	0x00240007	C	Endcap
89	0x00240105	C	Endcap
90	0x00240006	C	Endcap

Appendix G FTK Tower ID

The FTK simulation software identifies the 64 towers by number (0 to 63) as shown in Table 18. FTK tower partitions are shown in Figure 2(a) and Figure 2(b). Towers 0 to 15 cover the C-Side Endcap, towers 16 to 31 cover the C-Side Barrel, towers 32 to 47 cover the A-Side Barrel, and lastly towers 48 to 63 cover the A-Side Endcap. Note that the FTK tower 0 does not correspond to $\phi00$ but to $\phi15$.

	C-Side Endcap	C-Side Barrel	A-Side Barrel	A-Side Endcap
$\phi00$	1	17	33	49
$\phi01$	2	18	34	50
$\phi02$	3	19	35	51
$\phi03$	4	20	36	52
$\phi04$	5	21	37	53
$\phi05$	6	22	38	54
$\phi06$	7	23	39	55
$\phi07$	8	24	40	56
$\phi08$	9	25	41	57
$\phi09$	10	26	42	58
$\phi10$	11	27	43	59
$\phi11$	12	28	44	60
$\phi12$	13	29	45	61
$\phi13$	14	30	46	62
$\phi14$	15	31	47	63
$\phi15$	0	16	32	48

Table 18: FTK Tower ID and Tower Location.

Appendix H SLINK - FPGA Assignments

The mapping between the SLINK fibers and the Data Formatter FPGAs are summarized in Table 19 and Table 20. This arrangement has been selected to minimize data traffic in the Data Formatter system.

Tower ID	Pixel	SCT
0	0x00120205 (45) 0x00130105 (17)	0x00220001 (27) 0x00240004 (87)
1	0x00120220 (59) 0x00130307 (33)	0x00220009 (33) 0x00240105 (89)
2	0x00130310 (36) 0x00130319 (44)	0x00240101 (72) 0x00240103 (81)
3	0x00120218 (57) 0x00121616 (67)	0x00220006 (41) 0x00240104 (85)
4	0x00120216 (55) 0x00130315 (40)	0x00240100 (68) 0x00240102 (73)
5	0x00130108 (20) 0x00130316 (41)	0x00240109 (78)
6	0x00130109 (21) 0x00130112 (24)	0x00240106 (86) 0x00240107 (84)
7	0x00120214 (53) 0x00121614 (65) 0x00130114 (25)	0x0024010a (71)
8	0x00111715 (110) 0x00130117 (28)	0x00220102 (35) 0x00240108 (79)
9	0x00120211 (51) 0x00130118 (29)	0x00220109 (37) 0x00240006 (90)
10	0x00121611 (63) 0x00130005 (1)	0x00240008 (83) 0x0024000a (76)
11	0x00120209 (49) 0x00130008 (4)	0x00220106 (43) 0x00240009 (77)
12	0x00112507 (87) 0x00130009 (5)	0x00240007 (88) 0x0024000b (70)
13	0x00130012 (8) 0x00130014 (9)	0x00240003 (82)
14	0x00130017 (12) 0x00130018 (13)	0x00240001 (74) 0x00240005 (80)
15	0x00120207 (47) 0x00121609 (61)	0x00240000 (69) 0x00240002 (75)
16	0x00111721 (116) 0x00112421 (84)	0x00220004 (31)
17	0x00111706 (102) 0x00112512 (92)	0x00220005 (1)
18	0x00111806 (118) 0x00111808 (120)	0x00220007 (3)
19	0x00112515 (94) 0x00130311 (37)	0x00220008 (5)
20	0x00111708 (104) 0x00112517 (96)	0x0022000a (7)
21	0x00111710 (106) 0x00112519 (98)	0x00220100 (9)
22	0x00111712 (108) 0x00112521 (100)	0x00220101 (11)
23	0x00111810 (122) 0x00111812 (124)	0x00220103 (13)
24	0x00111717 (112) 0x00112406 (70) 0x00112408 (72)	0x00220104 (15)
25	0x00112505 (85) 0x00130121 (32)	0x00220105 (17)
26	0x00111815 (126) 0x00112410 (74)	0x00220107 (19)
27	0x00111817 (128) 0x00112509 (89)	0x00220108 (21)
28	0x00112412 (76) 0x00112415 (78)	0x0022010a (23)
29	0x00111719 (114) 0x00112417 (80)	0x00220000 (25)
30	0x00111819 (130) 0x00112419 (82)	0x00220002 (39)
31	0x00111821 (132) 0x00130021 (16)	0x00220003 (29)

Table 19: ROD assignment for C-Side FTK tower (Tower0-31). The numbers noted in the brackets are ROD IDs, corresponding to numbers on Figure 29, and Figure 31.

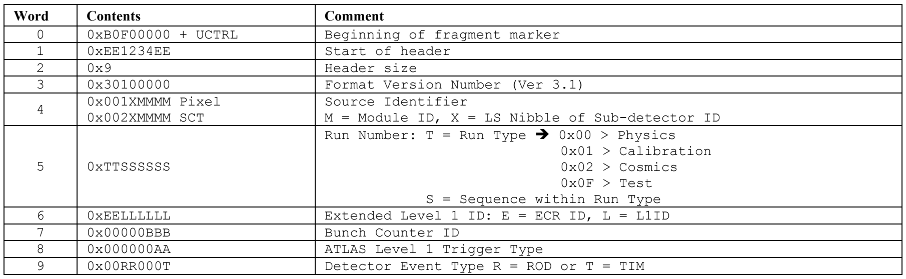
Tower ID	Pixel	SCT
32	0x00111720 (115) 0x00112420 (83)	0x00210004 (32)
33	0x00111705 (101) 0x00112511 (91)	0x00210005 (2)
34	0x00111805 (117) 0x00111807 (119)	0x00210007 (4)
35	0x00112514 (93) 0x00130312 (38)	0x00210008 (6)
36	0x00111707 (103) 0x00112516 (95)	0x0021000a (8)
37	0x00111709 (105) 0x00112518 (97)	0x00210100 (10)
38	0x00111711 (107) 0x00112520 (99)	0x00210101 (12)
39	0x00111809 (121) 0x00111811 (123)	0x00210103 (14)
40	0x00111716 (111) 0x00112405 (69) 0x00112407 (71)	0x00210104 (16)
41	0x00112506 (86) 0x00130120 (31)	0x00210105 (18)
42	0x00111814 (125) 0x00112409 (73)	0x00210107 (20)
43	0x00111816 (127) 0x00112510 (90)	0x00210108 (22)
44	0x00112411 (75) 0x00112414 (77)	0x0021010a (24)
45	0x00111718 (113) 0x00112416 (79)	0x00210000 (26)
46	0x00111818 (129) 0x00112418 (81)	0x00210002 (40)
47	0x00111820 (131) 0x00130020 (15)	0x00210003 (30)
48	0x00120206 (46) 0x00130106 (18)	0x00210001 (28) 0x00230107 (62)
49	0x00120221 (60) 0x00130308 (34)	0x00210009 (34) 0x00230005 (66)
50	0x00130309 (35) 0x00130318 (43)	0x00230007 (60) 0x00230009 (55)
51	0x00120219 (58) 0x00121617 (68)	0x00210006 (42) 0x00230006 (64)
52	0x00120217 (56) 0x00130314 (39)	0x00230008 (56) 0x0023000a (48)
53	0x00130107 (19) 0x00130317 (42)	0x00230001 (49)
54	0x00130110 (22) 0x00130111 (23)	0x00230003 (57) 0x00230004 (61)
55	0x00120215 (54) 0x00121615 (66) 0x00130115 (26)	0x00230000 (45)
56	0x00111714 (109) 0x00130116 (27)	0x00210102 (36) 0x00230002 (50)
57	0x00120212 (52) 0x00130119 (30)	0x00210109 (38) 0x00230105 (65)
58	0x00121612 (64) 0x00130006 (2)	0x00230101 (51) 0x00230103 (58)
59	0x00120210 (50) 0x00130007 (3)	0x00210106 (44) 0x00230102 (52)
60	0x00112508 (88) 0x00130010 (6)	0x00230100 (46) 0x00230104 (63)
61	0x00130011 (7) 0x00130015 (10)	0x00230108 (59)
62	0x00130016 (11) 0x00130019 (14)	0x00230106 (67) 0x0023010a (53)
63	0x00120208 (48) 0x00121610 (62)	0x00230109 (54) 0x0023010b (47)

Table 20: ROD assignment for A-Side FTK tower (Tower32-63). The numbers noted in the brackets are ROD IDs, corresponding to numbers on Figure 28 and Figure 30.

Appendix I Pixel and SCT Data Format

I.1 S-Link Header and Trailer

The data format of Pixel and SCT ROD output links are summarized in the ROD User Manual [12]. ROD outputs use the SLINK protocol.

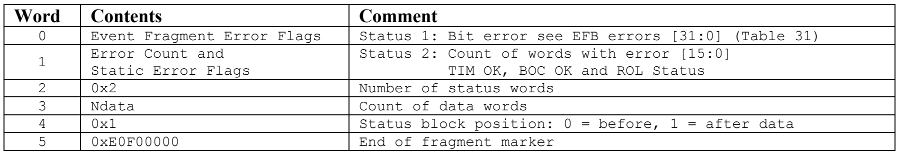


Word	Contents	Comment
0	0xB0F00000 + UCTRL	Beginning of fragment marker
1	0xEE1234EE	Start of header
2	0x9	Header size
3	0x30100000	Format Version Number (Ver 3.1)
4	0x001XMMMM Pixel 0x002XMMMM SCT	Source Identifier M = Module ID, X = LS Nibble of Sub-detector ID
5	0xTTSSSSSS	Run Number: T = Run Type ➔ 0x00 > Physics 0x01 > Calibration 0x02 > Cosmics 0x0F > Test S = Sequence within Run Type
6	0xEELLLLLL	Extended Level 1 ID: E = ECR ID, L = L1ID
7	0x00000BBB	Bunch Counter ID
8	0x000000AA	ATLAS Level 1 Trigger Type
9	0x00RR000T	Detector Event Type R = ROD or T = TIM

Figure 44: ROD Event Header. Refer Table 21 for further details of Source ID words.

0x0013XXXX	Pixel Barrel B-layer
0x0011XXXX	Pixel Barrel L1 and L2
0x0012XXXX	Pixel Endcap disks
0x0021XXXX	SCT A Side Barrel layers
0x0022XXXX	SCT C Side Barrel layers
0x0023XXXX	SCT A Side Endcap disks
0x0024XXXX	SCT C Side Endcap disks

Table 21: Source IDs (See Figure 44).



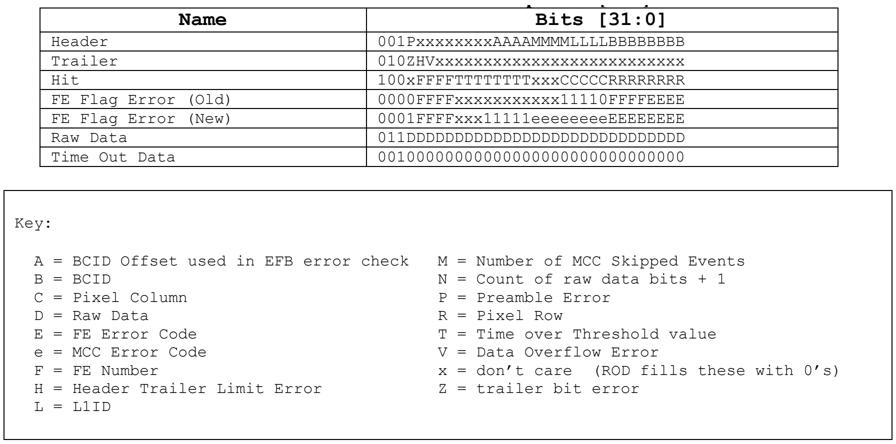
Word	Contents	Comment
0	Event Fragment Error Flags	Status 1: Bit error see EFB errors [31:0] (Table 31)
1	Error Count and Static Error Flags	Status 2: Count of words with error [15:0] TIM OK, BOC OK and ROL Status
2	0x2	Number of status words
3	Ndata	Count of data words
4	0x1	Status block position: 0 = before, 1 = after data
5	0xE0F00000	End of fragment marker

Figure 45: ROD event trailer

I.2 Pixel Raw Data

The Pixel-ROD data words are 32 bits wide, which consist of series of data for individual FE readout modules that connected to the ROD. Since only the header words are reformatted by the router FPGAs on the RODs, refer Figure 47 for the header word definition for the FTK inputs.

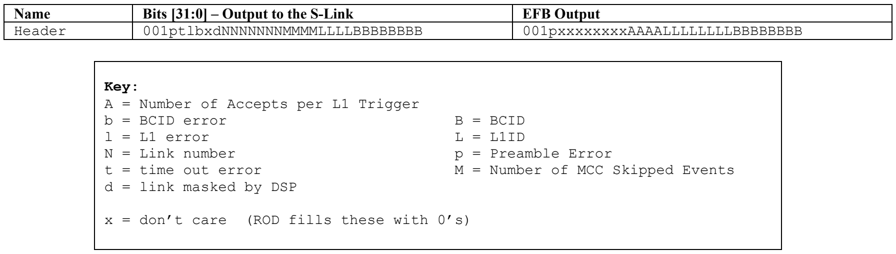
The "Link Number" in the header word definition indicates the module ID of the data packet. It is translated into the "FMT" ID and "LINK" ID. Note that the these definition depends on the configuration of the readout speed (40 MHz or 80MHz). Generally speaking all layer-2 RODs use 40MHz mode, the layer-1, B-layer and disc RODs use the 80MHz readout mode. In addition the B-layer modules send their data on two links (resulting in an effective 160 Mb/s) whereas all other modules have only one link.



Name	Bits [31:0]
Header	001PxxxxxxxxAAAAMMMMLLLLBBBBBBBB
Trailer	010ZHVxxxxxxxxxxxxxxxxxxxxxxxxxx
Hit	100xFFFFTTTTTTTTxxxCCCCCRRRRRRRR
FE Flag Error (Old)	0000FFFFxxxxxxxxxxx11110FFFFEEEE
FE Flag Error (New)	0001FFFFxxx11111eeeeeeeeEEEEEEEE
Raw Data	011DDDDDDDDDDDDDDDDDDDDDDDDDDDDD
Time Out Data	00100000000000000000000000000000

```
Key:

  A = BCID Offset used in EFB error check   M = Number of MCC Skipped Events
  B = BCID                                  N = Count of raw data bits + 1
  C = Pixel Column                          P = Preamble Error
  D = Raw Data                              R = Pixel Row
  E = FE Error Code                         T = Time over Threshold value
  e = MCC Error Code                        V = Data Overflow Error
  F = FE Number                             x = don't care  (ROD fills these with 0's)
  H = Header Trailer Limit Error            Z = trailer bit error
  L = L1ID
```

Figure 46: Pixel RAW data definition. Note that the header word will eventually be reformatted in the ROD, which is shown Figure 47.



Name	Bits [31:0] – Output to the S-Link	EFB Output
Header	001ptlbxdNNNNNNNMMMMLLLLBBBBBBBB	001pxxxxxxxxAAALLLLLLLLLBBBBBBBB

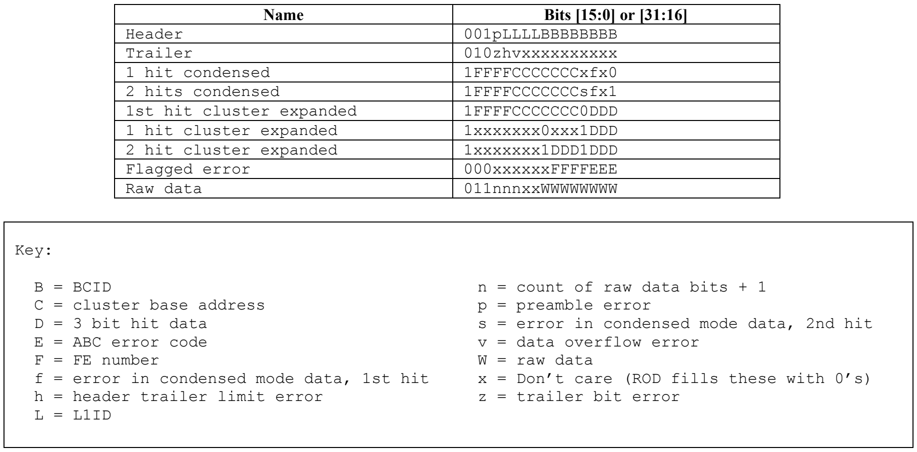
```
Key:
A = Number of Accepts per L1 Trigger
b = BCID error                         B = BCID
l = L1 error                           L = L1ID
N = Link number                        p = Preamble Error
t = time out error                     M = Number of MCC Skipped Events
d = link masked by DSP

x = don't care  (ROD fills these with 0's)
```

Figure 47: Header words in Pixel output.

I.3 SCT Raw Data

All event data is in 16-bit words packed in 32-bit frames. Since only the header words are reformatted by the router FPGAs on the RODs, refer Figure 49 for the header word definition for the FTK inputs.

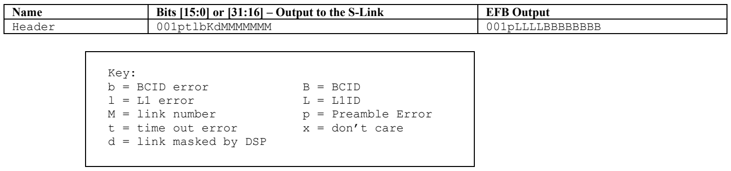
The SCT has two readout modes: condensed and expanded. The expanded mode was used for 2012 operations while the condensed mode will be used for higher luminosity runs going forward. The expanded mode supports multiple edge modes (e.g. the neighboring three bunch crossings). Condensed mode uses only the ("01X") edge mode.



Name	Bits [15:0] or [31:16]
Header	001pLLLLBBBBBBBB
Trailer	010zhvxxxxxxxxxx
1 hit condensed	1FFFFCCCCCCCxfx0
2 hits condensed	1FFFFCCCCCCCsfx1
1st hit cluster expanded	1FFFFCCCCCCC0DDD
1 hit cluster expanded	1xxxxxxx0xxx1DDD
2 hit cluster expanded	1xxxxxxx1DDD1DDD
Flagged error	000xxxxxxFFFFEEE
Raw data	011nnnxxWWWWWWWW

```
Key:

  B = BCID                                    n = count of raw data bits + 1
  C = cluster base address                    p = preamble error
  D = 3 bit hit data                          s = error in condensed mode data, 2nd hit
  E = ABC error code                          v = data overflow error
  F = FE number                               W = raw data
  f = error in condensed mode data, 1st hit   x = Don't care (ROD fills these with 0's)
  h = header trailer limit error              z = trailer bit error
  L = L1ID
```

Figure 48: SCT RAW data definition. Note that the header word will eventually be reformatted in the ROD, which is shown Figure 49.



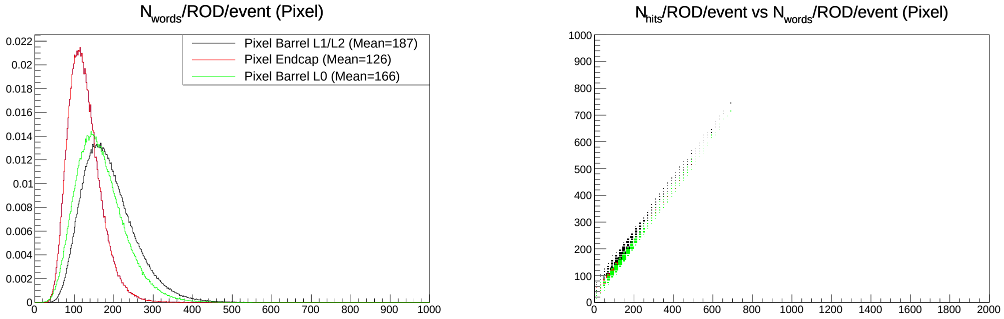
Name	Bits [15:0] or [31:16] – Output to the S-Link	EFB Output
Header	001ptlbKdMMMMMMM	001pLLLLBBBBBBBB

```
Key:
b = BCID error              B = BCID
l = L1 error                L = L1ID
M = link number             p = Preamble Error
t = time out error          x = don't care
d = link masked by DSP
```

Figure 49: Header words in SCT output. The "Link Number" in the Table indicates the modules ID and layers of the SCT doublet.

I.4 Number of SLINK Words

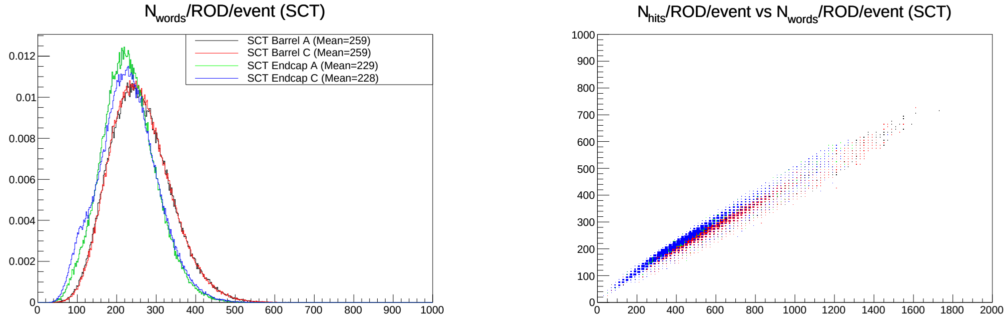
We studied the number of words per ROD as part of our input data volume study using the actual collision data with $\sqrt{s} = 8$ TeV, $\langle \mu \rangle = 30.0$, and 50 ns bunch spacing. Refer Section 8.2.1 for details of the data-sample.



(a)

(b)

Figure 50: Input data volume distribution for Pixel RODs. (a) shows number of 32-bit S-Link words per ROD per event distribution. (b) shows correlation between the N_{hits} and N_{words} (Refer Figure 18).

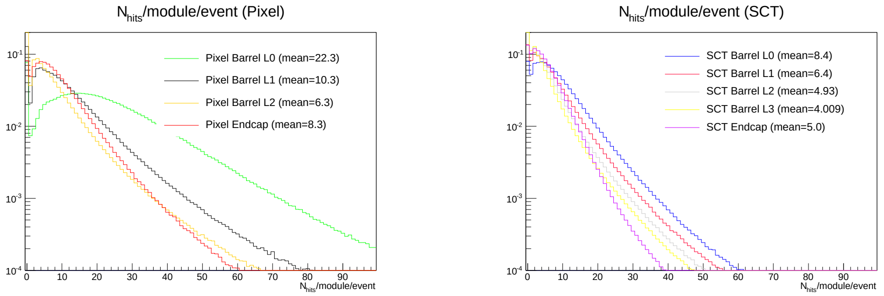


(a)

(b)

Figure 51: Input data volume distribution for SCT RODs. (b) shows number of 32-bit S-Link words per ROD per event distribution. (c) shows correlation between the N_{hits} and N_{words} (Refer Figure 18).

Appendix J Tails in the Number of Hits per Module



(a) (b)

Figure 52: N_{hits} per module per event distribution in log scale. Details of this tail component are shown in Tables 22 and 23. LHC operating conditions were $\sqrt{s} = 8$ TeV, $\langle\mu\rangle = 30.0$, with a 50 ns bunch spacing.

	$N_{\text{hits}} \geq 5$	$N_{\text{hits}} \geq 10$	$N_{\text{hits}} \geq 15$	$N_{\text{hits}} \geq 20$
pixel (Barrel L0)	87.8	76.4	62.5	48.5
pixel (Barrel L1)	67.6	40.6	23.2	13.5
pixel (Barrel L2)	62.9	29.6	14.8	8.1
pixel (Endcap)	37.8	18.7	8.7	4.3

Table 22: Tail components of the N_{hits} per module per event distribution for Pixel shown in percentage term. LHC operating conditions were $\sqrt{s} = 8$ TeV, $\langle\mu\rangle = 30.0$, with a 50 ns bunch spacing.

	$N_{\text{hits}} \geq 5$	$N_{\text{hits}} \geq 10$	$N_{\text{hits}} \geq 15$	$N_{\text{hits}} \geq 20$
SCT (Barrel L0)	63.8	32.2	15.4	7.7
SCT (Barrel L1)	49.5	21.0	9.4	4.6
SCT (Barrel L2)	37.9	14.2	6.1	3.0
SCT (Barrel L3)	29.4	10.3	4.4	2.2
SCT (Endcap)	43.5	13.6	4.2	1.5

Table 23: Tail components of the N_{hits} per module per event distribution for SCT shown in percentage term. LHC operating conditions were $\sqrt{s} = 8$ TeV, $\langle\mu\rangle = 30.0$, with a 50 ns bunch spacing.

Appendix K Downstream Data Flow

In Table 24 and Table 25 we summarize the dataflow from the Data Formatter to the 64 individual FTK towers. This analysis was done with real collision data collected from runs taken at $\sqrt{s} = 8$ TeV, $\langle\mu\rangle = 30.0$, with a 50 ns bunch spacing. Please refer the N_{hits} definition in Section 8.2.2. The numbers are extrapolated into $\sqrt{s} = 14$ TeV, $\langle\mu\rangle = 70.0$, and 25 ns bunch spacing conditions in Table K and Table K.

Tower	P-B-L0	P-B-L1	P-B-L2	P-EC	S-B-L0	S-B-L1	S-B-L2	S-B-L3	S-EC	N_{pxl}	N_{SCT}
0	414	283	199	130	191	96	47	0	1097	1026	1431
1	486	254	199	114	189	119	47	0	1101	1052	1456
2	493	255	172	105	187	95	47	0	1160	1025	1489
3	500	323	166	115	187	119	47	0	1165	1104	1518
4	482	247	180	124	189	96	47	0	1102	1033	1434
5	483	313	186	124	189	121	47	0	1095	1105	1452
6	319	258	175	133	189	96	47	0	1118	885	1450
7	467	257	173	134	190	119	47	0	1108	1031	1465
8	487	311	207	134	189	96	46	0	1042	1139	1374
9	474	245	206	132	190	121	46	0	1052	1057	1409
10	469	255	160	131	193	98	47	0	1129	1015	1466
11	437	315	151	132	192	118	47	0	1156	1036	1513
12	451	240	205	131	175	97	47	0	1097	1028	1416
13	475	299	204	130	175	122	48	0	1101	1108	1446
14	317	245	136	113	192	96	47	0	1155	811	1490
15	411	238	144	121	192	119	47	0	1163	914	1522
16	346	327	266	0	329	293	331	315	0	939	1269
17	416	290	267	0	327	354	331	316	0	974	1328
18	419	292	234	0	309	292	331	316	0	945	1248
19	426	370	227	0	309	367	330	298	0	1023	1304
20	436	276	225	0	325	295	319	305	0	937	1245
21	437	349	225	0	329	371	330	313	0	1011	1343
22	245	299	225	0	328	295	331	315	0	769	1270
23	344	297	225	0	325	373	331	317	0	867	1346
24	391	357	270	0	328	299	332	317	0	1017	1276
25	406	282	282	0	330	372	329	318	0	969	1349
26	400	293	216	0	331	295	329	316	0	909	1272
27	367	360	193	0	331	355	331	315	0	920	1333
28	383	274	253	0	314	282	332	308	0	910	1236
29	410	346	272	0	311	370	333	309	0	1028	1323
30	273	283	199	0	327	295	331	316	0	755	1269
31	342	275	213	0	330	366	332	314	0	830	1341

Table 24: The number of hits in the output stream from DF to AUX or SSB. These numbers are extracted from a run with $\sqrt{s} = 8$ TeV and $\langle\mu\rangle = 30$(Run:214523, LB:153). This Table references Tower IDs 0 to 31 for the C-Side. Number of SCT hits are counted with "edge mode" emulation.

Tower	P-B-L0	P-B-L1	P-B-L2	P-EC	S-B-L0	S-B-L1	S-B-L2	S-B-L3	S-EC	N_{pxl}	N_{SCT}
32	347	336	274	0	329	296	335	314	0	958	1274
33	418	290	274	0	328	371	332	315	0	983	1347
34	420	292	235	0	327	295	330	316	0	947	1268
35	426	369	235	0	327	367	330	305	0	1030	1329
36	436	275	238	0	328	294	330	305	0	948	1256
37	413	350	230	0	331	369	331	312	0	993	1343
38	222	299	224	0	332	296	332	313	0	745	1273
39	367	287	211	0	329	372	332	313	0	864	1346
40	414	348	257	0	329	296	332	315	0	1018	1271
41	407	282	282	0	330	370	332	316	0	971	1347
42	379	291	221	0	331	296	333	315	0	892	1276
43	344	358	199	0	332	366	330	314	0	902	1342
44	383	273	254	0	330	292	330	306	0	911	1257
45	410	347	273	0	327	366	330	307	0	1030	1330
46	273	284	199	0	328	292	329	305	0	756	1254
47	343	286	227	0	330	366	333	305	0	855	1334
48	439	312	200	131	191	96	47	0	1105	1082	1440
49	495	254	200	124	191	121	47	0	1111	1072	1470
50	500	257	173	132	190	96	47	0	1162	1061	1495
51	505	322	172	125	190	119	46	0	1157	1125	1512
52	515	257	194	126	191	95	46	0	1086	1093	1418
53	492	322	194	134	192	120	47	0	1094	1142	1453
54	248	259	166	123	191	95	46	0	1156	796	1489
55	418	247	138	123	191	120	47	0	1167	926	1526
56	486	310	174	131	159	95	47	0	1098	1101	1399
57	478	255	199	124	158	120	47	0	1095	1056	1420
58	451	253	163	114	191	95	48	0	1148	981	1482
59	419	313	151	121	189	119	46	0	1146	1004	1501
60	458	240	186	122	190	95	46	0	1080	1005	1410
61	481	289	198	130	191	119	47	0	1097	1099	1454
62	319	238	122	129	189	96	47	0	1164	808	1496
63	435	252	165	130	189	120	47	0	1175	981	1531

Table 25: The number of hits in the output stream from DF to AUX or SSB. These numbers are extracted from a run with $\sqrt{s} = 8$ TeV and $\langle \mu \rangle = 30$(Run:214523, LB:153). This Table references Tower IDs 32 to 63 for the A-Side. Number of SCT hits are counted with "edge mode" emulation.

Tower	P-B-L0	P-B-L1	P-B-L2	P-EC	S-B-L0	S-B-L1	S-B-L2	S-B-L3	S-EC	N_{pxl}	N_{SCT}
0	1401	958	672	441	646	326	158	0	3712	3472	4841
1	1645	858	673	384	639	404	159	0	3726	3559	4927
2	1668	864	582	355	633	322	157	0	3925	3469	5037
3	1690	1092	563	390	634	403	158	0	3942	3735	5137
4	1630	837	610	418	640	326	158	0	3728	3495	4851
5	1634	1057	630	419	640	411	158	0	3705	3740	4913
6	1081	872	592	450	640	324	160	0	3782	2995	4906
7	1580	869	586	453	643	404	159	0	3749	3489	4955
8	1647	1051	701	454	640	326	157	0	3525	3853	4648
9	1604	829	698	446	644	408	157	0	3558	3576	4767
10	1588	861	541	442	652	331	158	0	3819	3433	4961
11	1478	1067	512	447	648	399	159	0	3912	3504	5118
12	1527	811	695	444	593	327	161	0	3712	3476	4792
13	1608	1012	691	439	593	411	162	0	3726	3750	4892
14	1072	829	460	382	648	325	160	0	3907	2743	5040
15	1391	804	488	410	650	403	159	0	3936	3092	5148
16	1170	1105	902	0	1114	992	1121	1066	0	3177	4292
17	1408	981	905	0	1107	1197	1121	1068	0	3294	4493
18	1418	987	791	0	1044	989	1121	1068	0	3196	4222
19	1440	1252	770	0	1044	1241	1117	1009	0	3461	4411
20	1476	933	763	0	1099	999	1080	1032	0	3171	4211
21	1477	1182	761	0	1113	1254	1116	1060	0	3420	4543
22	829	1012	762	0	1110	999	1121	1066	0	2603	4297
23	1165	1006	762	0	1099	1261	1121	1073	0	2932	4554
24	1322	1208	912	0	1111	1010	1123	1073	0	3442	4318
25	1373	953	953	0	1115	1260	1112	1076	0	3279	4563
26	1354	991	730	0	1121	998	1114	1070	0	3075	4304
27	1243	1217	654	0	1121	1200	1121	1067	0	3113	4509
28	1297	927	856	0	1061	954	1125	1042	0	3080	4182
29	1386	1171	920	0	1052	1252	1127	1045	0	3477	4477
30	924	957	672	0	1107	998	1121	1068	0	2553	4294
31	1157	930	720	0	1116	1238	1122	1063	0	2807	4539

Table 26: The number of hits in output stream from DF to AUX or SSB. The numbers extracted from a run with $\sqrt{s} = 8$ TeV and $\langle\mu\rangle = 30$ (Run:214523, LB:153) **and extrapolated into** 14 TeV **and** $\langle\mu\rangle = 70$. This Table references Tower IDs 0 to 31 for the C-Side. Number of SCT hits are counted with "edge mode" emulation.

Tower	P-B-L0	P-B-L1	P-B-L2	P-EC	S-B-L0	S-B-L1	S-B-L2	S-B-L3	S-EC	N_{pxl}	N_{SCT}
32	1093	1060	864	0	1038	933	1054	989	0	3016	4014
33	1317	914	864	0	1033	1169	1047	993	0	3096	4243
34	1322	921	741	0	1030	931	1040	994	0	2985	3995
35	1341	1163	740	0	1031	1155	1039	960	0	3244	4185
36	1373	865	749	0	1033	925	1039	960	0	2987	3958
37	1302	1101	725	0	1043	1162	1042	982	0	3128	4229
38	701	941	705	0	1045	932	1046	987	0	2347	4010
39	1156	903	663	0	1037	1173	1045	987	0	2722	4241
40	1303	1095	809	0	1035	932	1044	992	0	3207	4004
41	1282	888	888	0	1039	1164	1045	995	0	3058	4242
42	1193	918	697	0	1044	932	1050	993	0	2808	4019
43	1085	1127	628	0	1045	1154	1040	988	0	2840	4227
44	1208	861	799	0	1041	918	1039	963	0	2868	3961
45	1293	1092	859	0	1030	1152	1041	966	0	3244	4188
46	861	894	626	0	1032	921	1035	961	0	2380	3950
47	1080	899	714	0	1040	1154	1048	962	0	2693	4204
48	1383	982	630	412	601	303	149	0	3482	3407	4535
49	1558	801	629	391	601	382	148	0	3499	3378	4630
50	1574	808	544	417	598	303	147	0	3660	3343	4708
51	1591	1016	543	393	599	374	146	0	3646	3543	4764
52	1623	810	612	396	602	301	145	0	3420	3441	4468
53	1550	1015	611	421	605	376	148	0	3447	3597	4576
54	781	815	524	388	603	300	145	0	3642	2508	4690
55	1318	777	434	388	602	379	147	0	3677	2917	4805
56	1530	977	547	413	500	301	148	0	3458	3467	4407
57	1505	804	628	390	498	377	148	0	3449	3327	4471
58	1420	796	514	360	601	301	150	0	3616	3090	4667
59	1320	987	474	383	596	374	146	0	3610	3164	4727
60	1444	755	584	383	598	298	146	0	3401	3166	4442
61	1516	910	625	410	602	375	148	0	3456	3461	4581
62	1005	751	384	405	596	303	148	0	3667	2546	4714
63	1370	793	521	408	596	379	148	0	3700	3092	4822

Table 27: The number of hits in the output stream from DF to AUX or SSB. The numbers extracted from a run with $\sqrt{s} = 8$ TeV and $\langle\mu\rangle = 30$ (Run:214523, LB:153) **and extrapolated into** 14 TeV **and** $\langle\mu\rangle = 70$. This Table references Tower IDs 32 to 63 for the A-Side. Number of SCT hits are counted with "edge mode" emulation.

Appendix L Data Flow Model

Each Data Formatter FPGA must determine where each incoming data packet will go. Figure 53 shows the decision making process, which was introduced in Section 8.2.5.

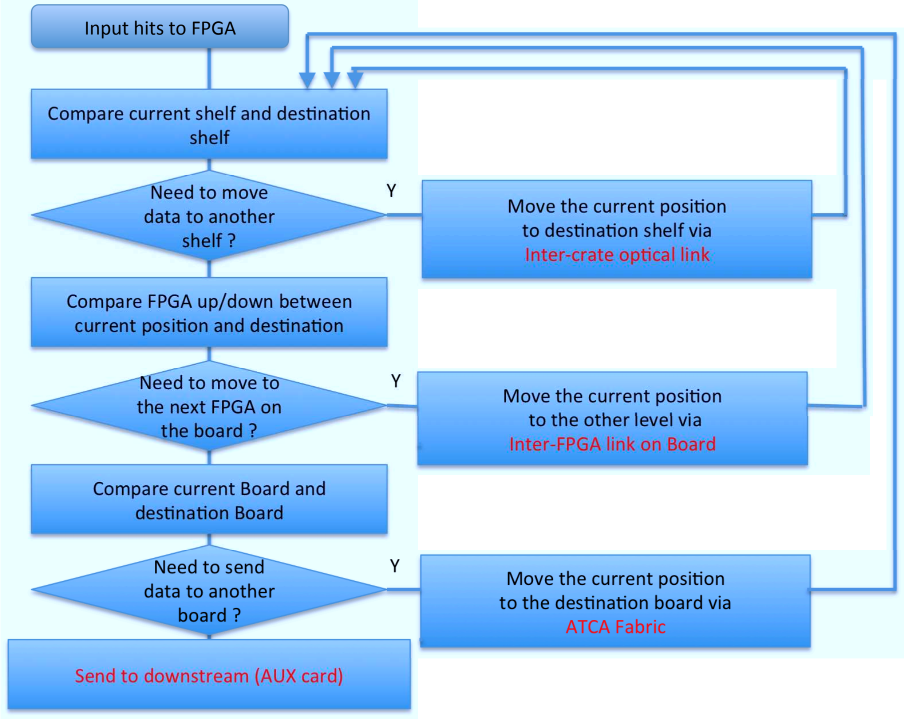

Input hits to FPGA
Compare current shelf and destination shelf
Need to move data to another shelf ?
Y
Move the current position to destination shelf via Inter-crate optical link
Compare FPGA up/down between current position and destination
Need to move to the next FPGA on the board ?
Y
Move the current position to the other level via Inter-FPGA link on Board
Compare current Board and destination Board
Need to send data to another board ?
Y
Move the current position to the destination board via ATCA Fabric
Send to downstream (AUX card)

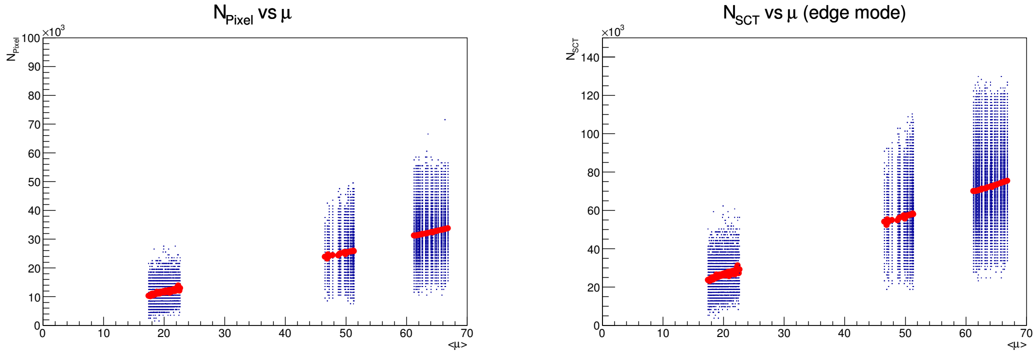
Figure 53: Flowchart of the simulation of data sharing inside the Data Formatter system.

Appendix M Data Volume Luminosity Dependence

In this section the data volume is studied as a function of pileup $\langle\mu\rangle$. We use nominal configuration runs and also special high $\langle\mu\rangle$ runs taken in 2012 with $\sqrt{s} = 8$ TeV are used so that it is possible to measure data volume at high pileup collisions with $\langle\mu\rangle$ up to ~ 70. One of the special runs (run 206717) covers $46 < \langle\mu\rangle < 52$ and the other covers (run 206725) covers $61 < \langle\mu\rangle < 67$. Only one bunch is filled in run 206717, and two bunches without train structure are filled in run 206725. To compare them without bias due to the operating conditions the following criteria are applied:

- Events are required to be recorded by random trigger.
- Events are required to be corresponding to the first bunch crossings of the train structures (see Figure 56, 57).
- SCT hit timing is required to be "010" or "011" in counting number of SCT raw hits. Hits at both of SCT doublet layers are counted.

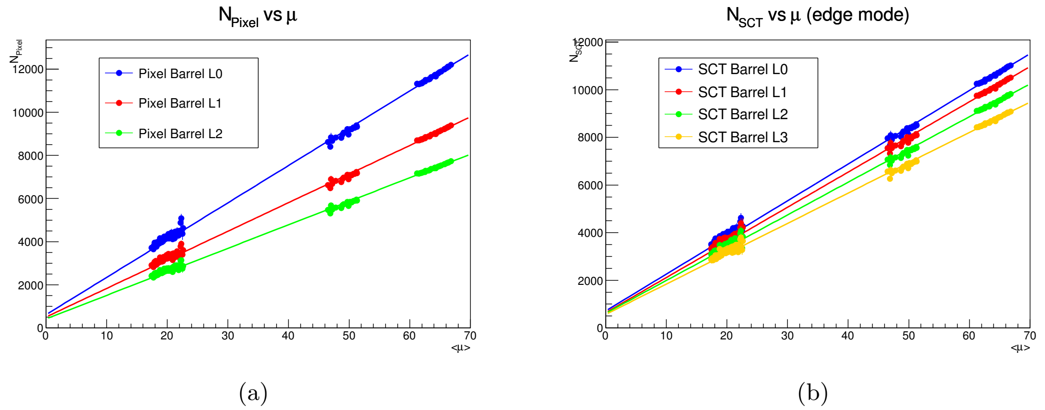
The $\langle\mu\rangle$ and data volume have linearity up to $\langle\mu\rangle \sim 70$. Refer the simulation study to see the linearity [3].



(a)

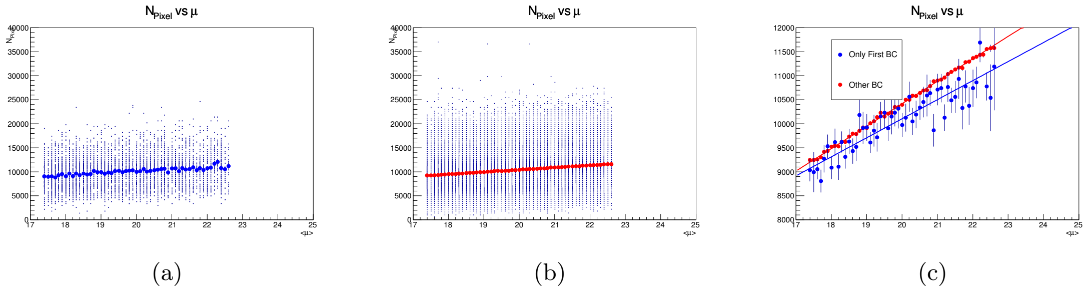
(b)

Figure 54: Data volume dependence on $\langle\mu\rangle$ with $\sqrt{s} = 8$ TeV. (a) Number of Pixel raw hits, and (b) Number of SCT raw hits passing "01X" timing cut (emulating "edge mode" operation).



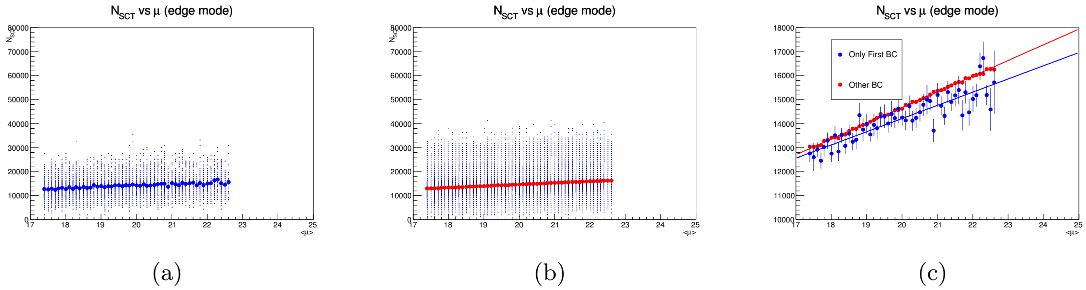
(a) (b)

Figure 55: Data volume dependence on $\langle\mu\rangle$ with $\sqrt{s} = 8$ TeV at individual Barrel layers. (i.e. Endcap is not included.) (a) Number of Pixel raw hits, and (b) Number of SCT raw hits passing "01X" timing cut (emulating "edge mode" operation).



(a) (b) (c)

Figure 56: (a) Number of Pixel hits (barrel) in first collisions of trains and (b) the other collisions. (c) Fitting results with the first polynomial function.



(a) (b) (c)

Figure 57: (a) Number of SCT hits (barrel) in first collisions of trains and (b) the other collisions. (c) Fitting results with the first polynomial function.

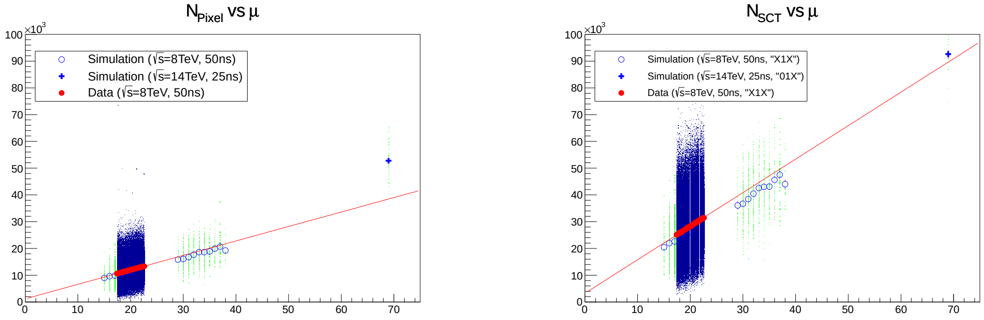
Appendix N Data Volume Monte Carlo Comparison

We used the following three types of data samples to compare the behavior of N_{hits} between MC samples and Data.

- Collision data ($\sqrt{s} = 8$ TeV, 50 ns bunch-spacing, $\langle \mu \rangle \sim 20$), fitted with linear function, shown with closed circles.
- MC samples ($\sqrt{s} = 8$ TeV, 50 ns bunch-spacing), shown with open circles.
- MC samples ($\sqrt{s} = 14$ TeV, 25 ns bunch-spacing), shown with closed crosses.

Note that the SCT readout condition is different in the 50 ns bunch-spacing MC sample ($\sqrt{s} = 8$ TeV) and the 25 ns bunch-spacing ($\sqrt{s} = 14$ TeV). The readout bunch window configuration is "X1X" for the 50 ns sample, and "01X" (edge modes) for the 25 ns sample. In order to compare the real data and the 50 ns bunch-spacing samples, "X1X" cut is emulated in the offline analysis for the real collision data sample [6].

Figure 58(a) and Figure 58(b) show the number of detected hits per event as a function of $\langle \mu \rangle$ for pixel hits (N_{Pixel}) and SCT hits (N_{SCT}), respectively. Figure 59(a) and Figure 59(b) show the number of detected hits at individual barrel layers for the pixel and SCT respectively.

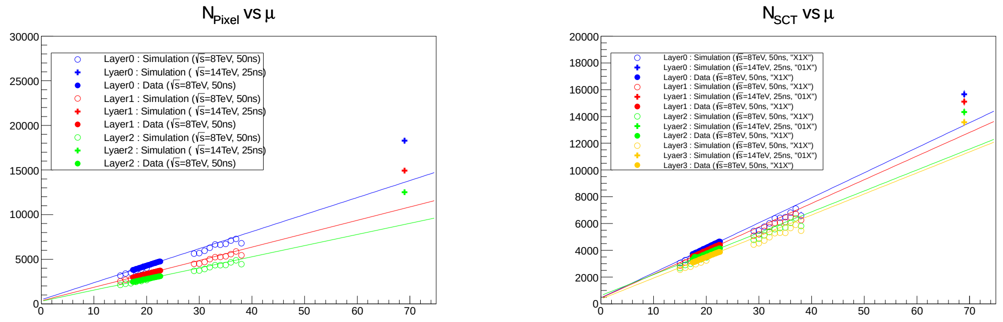


(a)

(b)

Figure 58: Number of (a) Pixel and (c) SCT hits as a function of $\langle \mu \rangle$. MC plots are superimposed to compare it with the data.

[6] In the MC analysis, it was not possible to emulate certain timing cuts due to technical limitations.



(a)

(b)

Figure 59: Number of (a) Pixel and (c) SCT hits as a function of $\langle \mu \rangle$ for individual barrel layers. MC plots are superimposed to compare it with the data.

Appendix O Clustering

O.1 Output Data Volume

Up until this point we have assumed no input data volume reduction due to clustering. In this section we present numbers for data volume and bandwidth requirements assuming that 2D Pixel clustering reduces the hit count by a factor of 3. Likewise we assume that linear clustering algorithms reduce the number of SCT hits by a factor of 2. Table 28 and Table O.1 show output data volume (N_{hits}) with the assumption of the data reduction. Note that the numbers are extrapolated into the high luminosity LHC condition of $\sqrt{s} = 14$ TeV, $\langle\mu\rangle = 70$, and 25 ns bunch spacing.

Tower	P-B-L0	P-B-L1	P-B-L2	P-EC	S-B-L0	S-B-L1	S-B-L2	S-B-L3	S-EC	N_{pxl}	N_{SCT}
0	447	292	198	130	304	151	73	0	1684	1068	2212
1	525	262	198	114	301	188	73	0	1694	1099	2255
2	533	264	172	105	298	150	72	0	1780	1073	2300
3	540	334	166	115	298	187	72	0	1787	1156	2345
4	521	256	180	124	301	151	72	0	1693	1081	2218
5	523	323	186	124	301	191	72	0	1680	1156	2245
6	346	266	175	133	301	151	74	0	1713	921	2239
7	506	266	173	134	303	188	74	0	1698	1079	2263
8	527	321	207	135	302	152	73	0	1597	1190	2125
9	513	253	206	132	304	191	72	0	1614	1104	2181
10	508	263	160	131	307	154	73	0	1731	1061	2265
11	472	325	151	132	305	185	73	0	1772	1081	2336
12	487	247	205	131	280	152	74	0	1683	1071	2189
13	513	308	203	130	280	191	74	0	1689	1155	2235
14	342	252	135	113	305	152	73	0	1770	842	2301
15	444	245	144	121	306	188	73	0	1784	953	2351
16	373	334	261	0	525	461	513	483	0	968	1982
17	449	297	262	0	521	555	513	483	0	1008	2072
18	453	299	229	0	491	459	513	483	0	981	1946
19	460	380	223	0	491	575	511	457	0	1063	2034
20	472	283	222	0	517	463	494	467	0	977	1941
21	472	359	221	0	524	582	511	479	0	1052	2095
22	265	308	221	0	523	464	514	482	0	794	1983
23	373	305	221	0	517	586	513	485	0	899	2101
24	423	366	265	0	524	469	515	485	0	1054	1993
25	439	289	276	0	526	584	508	488	0	1004	2107
26	432	300	212	0	528	464	509	484	0	944	1985
27	396	368	190	0	528	557	513	482	0	954	2080
28	413	280	248	0	500	443	515	472	0	942	1929
29	442	354	266	0	496	582	516	473	0	1063	2066
30	295	289	194	0	522	463	513	482	0	778	1980
31	368	281	208	0	526	574	513	481	0	858	2095

Table 28: The number of hits in the output stream from DF to AUX or SSB. The numbers extracted from a run with $\sqrt{s} = 8$ TeV and $\langle\mu\rangle = 30$ (Run:214523, LB:153) **assuming clustering reductions and extrapolated into** 14 TeV **and** $\langle\mu\rangle = 70$. This table summarized Tower ID 0 to 31 for C-Side. Number of SCT hits are counted with "edge mode" emulation.

Tower	P-B-L0	P-B-L1	P-B-L2	P-EC	S-B-L0	S-B-L1	S-B-L2	S-B-L3	S-EC	N_{pxl}	N_{SCT}
32	348	321	250	0	489	433	483	448	0	919	1853
33	420	277	250	0	487	543	479	449	0	948	1958
34	422	279	215	0	485	432	476	449	0	916	1842
35	428	353	215	0	486	535	476	435	0	996	1932
36	439	262	218	0	487	429	475	434	0	919	1825
37	416	334	211	0	491	539	476	444	0	961	1951
38	224	286	205	0	492	432	479	447	0	715	1850
39	370	274	193	0	488	544	478	447	0	836	1957
40	417	332	235	0	488	432	478	450	0	984	1849
41	409	269	258	0	490	540	479	451	0	936	1959
42	381	278	202	0	492	432	480	449	0	861	1853
43	346	341	182	0	492	535	476	447	0	869	1950
44	385	261	232	0	490	426	475	435	0	877	1826
45	412	330	249	0	485	534	476	436	0	992	1932
46	274	270	181	0	487	427	474	434	0	726	1822
47	344	272	207	0	490	535	480	435	0	823	1941
48	441	299	186	122	283	141	68	0	1582	1048	2075
49	498	244	186	116	283	178	68	0	1591	1044	2120
50	503	247	161	124	282	141	68	0	1664	1034	2154
51	509	310	160	116	282	174	67	0	1657	1096	2180
52	520	248	181	117	284	140	67	0	1553	1066	2043
53	496	310	181	125	285	175	68	0	1563	1112	2091
54	249	249	155	115	284	139	67	0	1655	768	2145
55	421	237	128	115	284	176	67	0	1671	902	2198
56	490	298	161	122	236	140	68	0	1571	1072	2015
57	481	245	185	116	235	175	68	0	1570	1027	2048
58	454	243	152	107	283	140	69	0	1642	955	2134
59	422	301	140	113	281	174	67	0	1639	976	2161
60	461	230	173	113	282	139	67	0	1542	977	2030
61	484	277	184	121	283	175	68	0	1569	1067	2096
62	321	229	113	120	281	141	69	0	1667	782	2158
63	437	242	154	121	281	176	68	0	1680	953	2206

Table 29: The number of hits in the output stream from DF to AUX or SSB. The numbers are extracted from a run with $\sqrt{s} = 8$ TeV and $\langle\mu\rangle = 30$ (Run:214523, LB:153) **assuming clustering reductions and extrapolated into** 14 TeV **and** $\langle\mu\rangle = 70$. This table summarized Tower ID 32 to 64 for the A-Side. Number of SCT hits are counted with "edge mode" emulation.

O.2 Bandwidth Requirement

Table 30 and Table 31 show data volume transferred in the DF system and the corresponding data bandwidth requirements, assumption clustering data reduction. Note that the numbers are extrapolated to high luminosity LHC operating conditions $\sqrt{s} = 14$ TeV, $\langle\mu\rangle = 70$, and 25 ns bunch spacing.

	average of $\langle N\rangle$	maximum of $\langle N\rangle$
Total (output stream)	3.4e+03	3.9e+03
Pixel Total (output stream)	1.1e+03	1.3e+03
SCT Total (output stream)	2.3e+03	2.6e+03
AUX	2.6e+03	2.8e+03
SSB	8.8e+02	1.0e+03
Fabric	3.8e+02	1.1e+03
Local Bus	1.5e+03	2.4e+03
Inter-Crate	8.7e+02	1.9e+03

Table 30: Summary of the expected N_{hits} with $\sqrt{s} = 14$ TeV, $\langle\mu\rangle = 70.0$, and 25 ns bunch spacing. These figures are extrapolated as described in Section 8.2.8. Clustering is assumed.

	average of $\langle N\rangle$	maximum of $\langle N\rangle$	BW requirement	Reserved BW
Total (output stream)	2.6e+03	2.8e+03	-	-
Pixel Total (output stream)	1.2e+03	1.4e+03	-	-
SCT Total (output stream)	1.4e+03	1.5e+03	-	-
AUX	2.0e+03	2.2e+03	8.98	48
SSB	5.2e+02	5.9e+02	2.37	6
Fabric	2.9e+02	6.6e+02	2.62	10
Local Bus	1.1e+03	1.9e+03	7.52	24
Inter-Crate	6.6e+02	1.4e+03	5.58	20

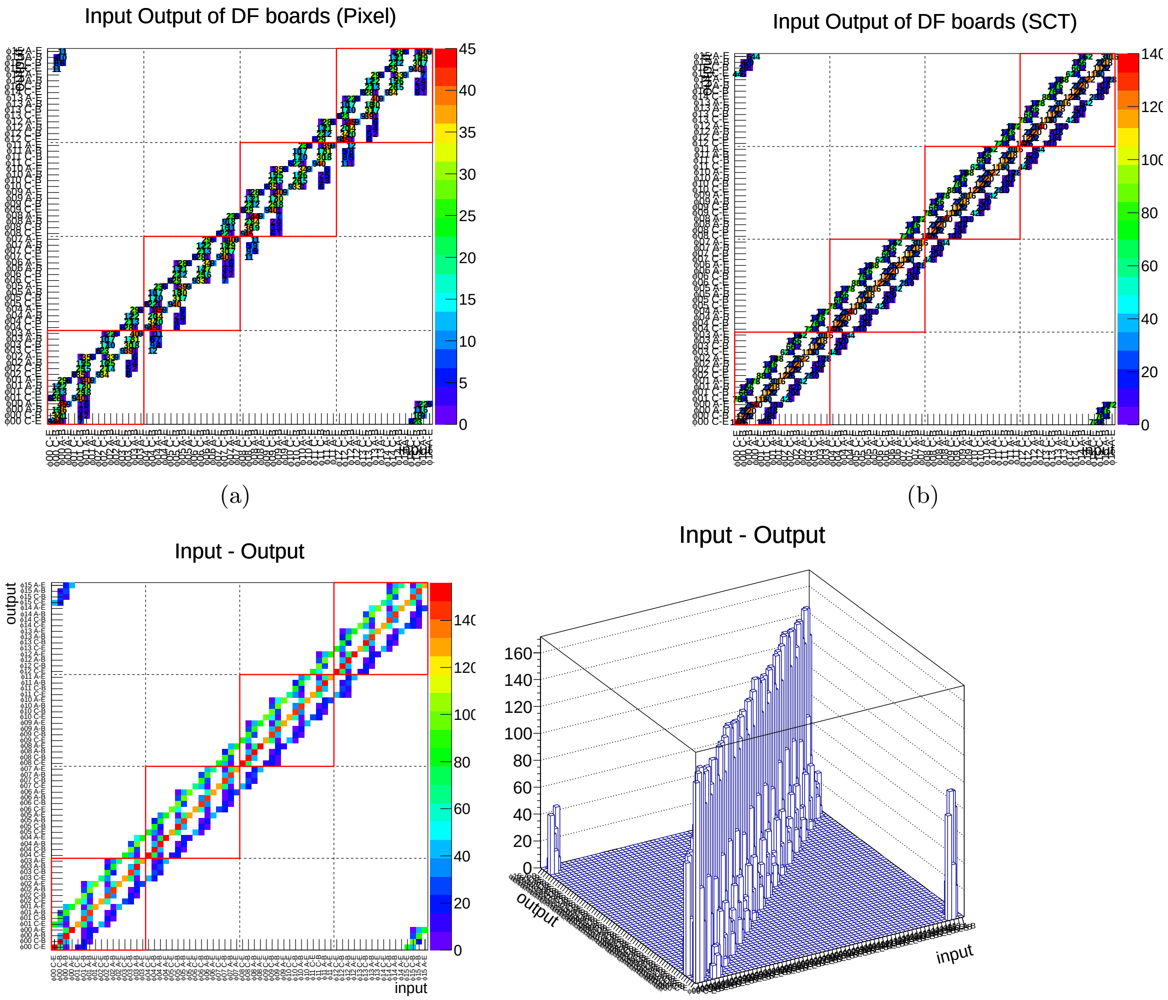
Table 31: Summary of the expected N_{words} with $\sqrt{s} = 14$ TeV, $\langle\mu\rangle = 70.0$, and 25 ns bunch spacing. These figures are extrapolated as described in Section 8.2.8. The third column shows the "worst case" bandwidth requirement in Gb/s. The forth column shows the reserved bandwidth (in Gb/s) for the current Data Formatter design. Clustering is assumed.

Appendix P Unconstrained Data Volume Study

As previously mentioned the inner detector readout system was not originally designed for a track trigger. Modules were connected to RODs to minimize data rates and balance bandwidth. In this section we consider Data Formatter performance assuming an idealized module-ROD and ROD-DF mapping.

P.1 Data Sharing

Refer to Figure 15 to compare these idealized results with the "real world" module-ROD cabling constraints.



(a)

(b)

(c)

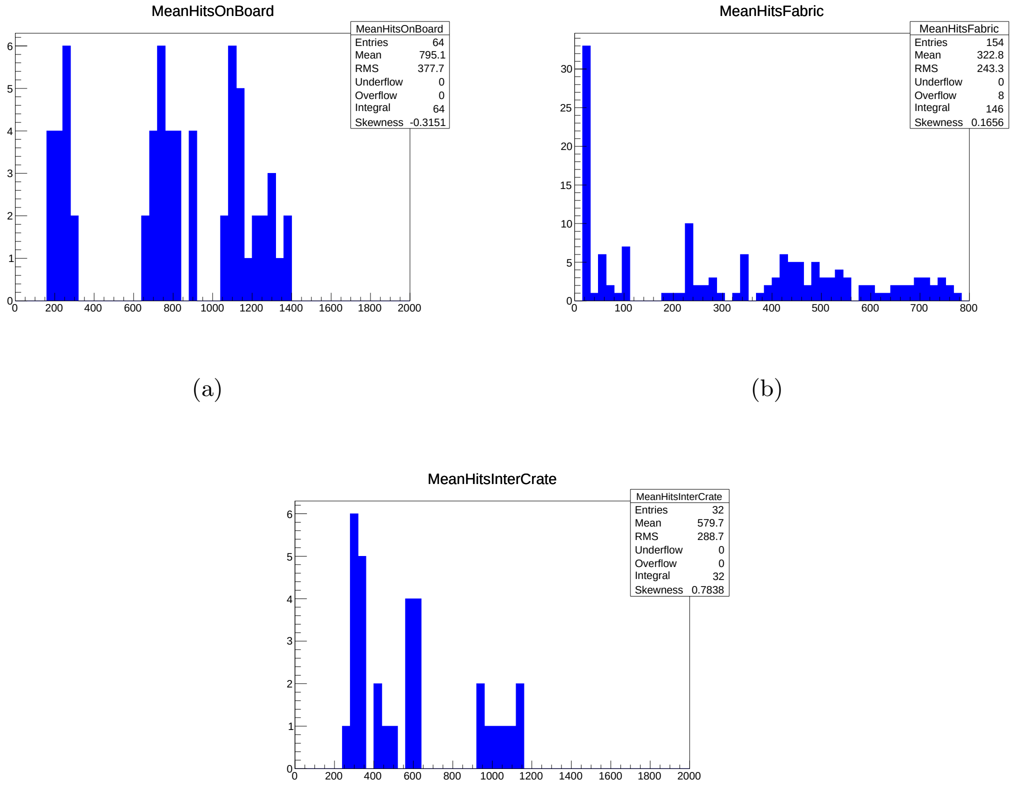
(d)

Figure 60: (a) and (b) show module sharing between two FPGAs for data from Pixel and SCT, respectively. (c) and (d) show the sum of both Pixel and SCT.

The details of the cabling can be found at `http://hep.uchicago.edu/~okumura/works/docs/20120628/idealstudy_input.xlsx`.

P.2 Data Volume Study Results

After removing the "real world" module-ROD cabling constraints we expect the data volume inside the Data Formatter system to change significantly. Figure 62 and Figure 61 show the N_{hits} and N_{words}, respectively. These figures assume LHC operating conditions of $\sqrt{s} = 8$ TeV, $\langle\mu\rangle = 30.0$, with a 50 ns bunch spacing. Table 32 shows the expected number of words and the required bandwidth for $\sqrt{s} = 14$ TeV, $\langle\mu\rangle = 70.0$, and 25 ns bunch spacing.

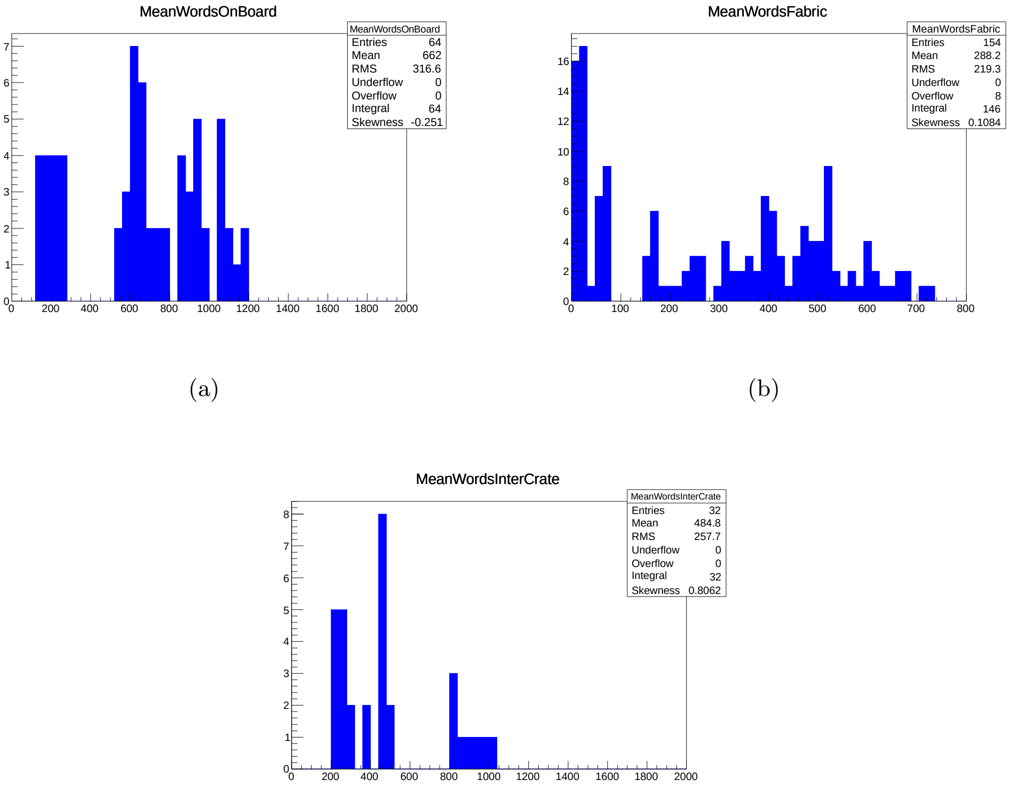


(a)

(b)

(c)

Figure 61: Summary of sum of N_{Pixel} and N_{SCT} transferred in the DF system **without module-ROD cabling constraints.**. These three histograms show the data volume transferred over the (a) local bus, (b) backplane Fabric Interface, and (c) the inter-shelf links. LHC operating conditions are $\sqrt{s} = 8$ TeV, $\langle\mu\rangle = 30.0$, with a 50 ns bunch spacing. Refer Figure 22 for the original study.



(a)

(b)

(c)

Figure 62: Summary of N_{words} transferred in the Data Formatter system, assuming no module-ROD cabling constraints. LHC operating conditions are $\sqrt{s} = 8$ TeV, $\langle \mu \rangle = 30.0$, with a 50 ns bunch spacing. Refer to Figure 25 for the original study.

	average of $\langle N \rangle$	maximum of $\langle N \rangle$	BW requirement	Reserved BW
Total (output stream)	5.9e+03	6.6e+03	-	-
Pixel Total (output stream)	3.4e+03	4.0e+03	-	-
SCT Total (output stream)	2.6e+03	2.9e+03	-	-
AUX	5.0e+03	5.6e+03	22.5	48
SSB	9.6e+02	1.1e+03	4.3	10
Fabric	9.7e+02	2.6e+03	10.5	10
Local Bus	2.0e+03	3.6e+03	14.4	24
Inter-Crate	1.5e+03	3.2e+03	12.7	20

Table 32: Summary of the expected N_{words} and bandwidth requirements with $\sqrt{s} = 14$ TeV, $\langle \mu \rangle = 70.0$, and 25 ns bunch spacing with no module-ROD cabling constraints. The third column shows the "worst case" bandwidth requirement in Gb/s. The forth column shows the system bandwidth in Gb/s.

References

[1] Data Formatter Initial Design Study and Hardware Specification (DRAFT)
J. Olsen, T. Liu, B. Penning, H.L. Li
Fermi National Accelerator Laboratory and The University of Chicago.
Fermilab Technical Publication TM-2546-PPD

[2] J. Olsen et al. “A Data Formatter for the ATLAS Fast Tracker” in IEEE Real Time Systems Symposium, Berkeley, CA, 2012.

[3] FTK: a hardware track finder for the ATLAS trigger Technical Proposal

[4] A Fast General-Purpose Clustering Algorithm
Based on FPGAs for High-Throughput Data Processing
A. Annoiv and M. Beretta
INFN - Laboratori Nazionali di Frascati, via E. Fermi 40, Frascati

[5] PICMG 3.0 AdvancedTCA Base Specification (Shortform)
http://www.picmg.org/v2internal/shortformspecs.htm

[6] PICMG 3.8 AdvancedTCA Rear Transition Module Zone 3A Specification
http://www.picmg.org

[7] CERN SLINK Homepage
http://hsi.web.cern.ch/hsi/s-link

[8] CERN xTCA Resources Wiki
https://twiki.cern.ch/twiki/bin/view/XTCA/WebHome

[9] FPGA Mezzanine Card Specification (VITA57.1)
http://www.vita.com/fmc.html

[10] Xilinx Kintex KC705 Development Board
http://www.xilinx.com/products/boards-and-kits/EK-K7-KC705-G.htm

[11] Xilinx Development Boards http://www.xilinx.com/products/boards_kits/fmc.htm

[12] ATLAS Silicon Readout Driver (ROD) Users Manual
http://www-eng.lbl.gov/~jmjoseph/Atlas-SiROD/Manuals/usersManual-v164.pdf

[13] Data Formatter FPGA Interconnections (animation)
http://home.fnal.gov/~jamieson/temp/DF.mpg

[14] Xilinx Serial I/O for 7-Series FPGAs UG471
http://www.xilinx.com/support/documentation/user_guides/ug471_7Series_SelectIO.pdf

[15] Samtec VITA 57 FMC SEARAY Connectors
http://www.samtec.com/search/vita57fmc.aspx

[16] Owen Boyle, Robert McLaren, Erik van der Bij,
The S-LINK Interface Specification, 1997

[17] ATLAS software Pixel Cabling Service https://svnweb.cern.ch/trac/atlasoff/browser/InnerDetector/InDetDetDescr/PixelCabling/trunk/share/Pixels_Atlas_IdMapping_May08.dat

[18] ATLAS software SCT Cabling Service https://svnweb.cern.ch/trac/atlasoff/browser/InnerDetector/InDetDetDescr/SCT_Cabling/trunk/share/SCT_Sept08Cabling_svc.dat