Specification of the

FTK / L2 Interface Crate (FLIC)

for the

ATLAS Fast-Tracker

John Anderson, Robert Blair, Gary Drake,
Andrew Kreps, Jimmy Proudfoot, Jinlong Zhang

Argonne National Laboratory

Version 1.7.2
April 2th, 2016

Contents

1. Introduction

The Fast-TracKer (FTK) is an electronics system that identifies and fits tracks in the inner detector silicon layers for every event that passes the Level 1 trigger. It uses all 12 silicon layers over the full rapidity range covered by the barrel and the disks. It receives a copy of the Pixel data and the Semiconductor Tracker (SCT) data as it is sent from the Read-Out Drivers (RODs) to the Read-Out System (ROSs) following a Level 1 trigger [1]. The FTK system calculates the helix parameters of all tracks with P_T above a minimum value, typically 1 GeV/c. This information, along with the hits, are sent to the ReadOut System (ROS) computers, which in turn makes the information available to the Level 2 Trigger. The Level-2 processors can request the track information in a Region of Interest or in the entire detector. Normally the inner detector RODs send the data directly to the ROSs. The transmission of data to the FTK system is accomplished by using a new output card from the ROD called a dual-output HOLA, which provides the FTK with an identical copy of the silicon data being sent to the ROSs. A block diagram of the FTK system is shown in Figure 1.

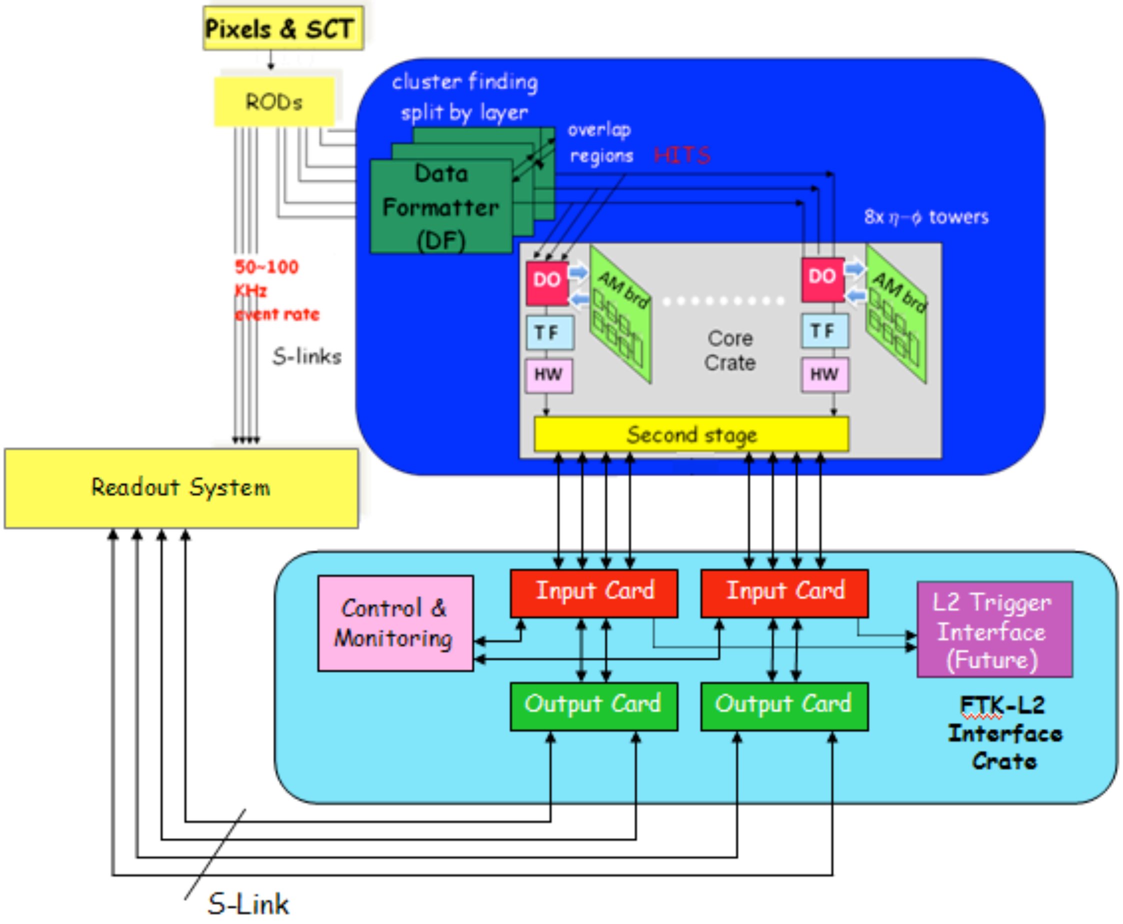

Figure 1. Block Diagram of the FTK System

The FTK system is a scalable processor, designed to function at the LHC Phase I luminosity of $3\cdot10^{34}$ cm^{-2}s^{-1}. It is designed to allow expansion and upgrade to operate at the higher Phase II luminosities.

This document describes specifications and requirements of a subsystem of the FTK called the FTK - L2 Interface Crate (FLIC). As shown in Figure 1, data is processed through the FTK system, with the final processing performed by the Core Crates. After an event is processed and track information constructed by the Core Crates, they in turn send track information to the FLIC as a data record. The FLIC then sends the data to the ROSs, where it is made available to the Level 2 trigger.

Each Core Crate processes data for every event that has been accepted by the Level 1 Trigger (L1). There are eight Core Crates in the system. Each Core Crate covers 45 degrees of the detector in phi as shown in Figure 2. Because the event processing is performed by discrete Core Crates, it is necessary to define an overlap area in each region, where Core Crates share information between crates to facilitate efficient track identification and reconstruction. The overlap regions for each detector region are 10 degrees on one side, as shown in Figure 2. The data from this region is sent to the neighboring Core Crate (clockwise in the figure.)

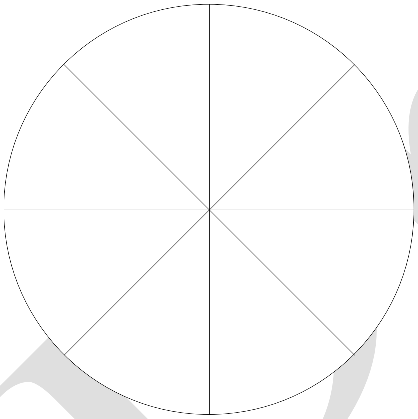

Figure 2a. Definition of the 8 Core Crate regions in phi-space of the detector.

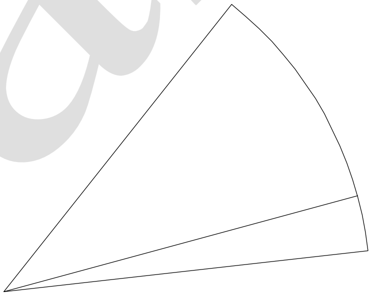

Figure 2b. Definition of an overlap region for a Core Crate region.

An event will generally have multiple tracks, but blank events with zero tracks will occur. The tracks are identified in the Core Crates. For each track, the Core Crates produce a set of helix parameters, which are summary information about the track. The data includes the road ID, the five helix parameters D0, Z0, PHI0, COTTH, and CURV, and a goodness of fit CHISQ. The Core Crates will also send hit ID information, so that the High-Level Trigger (HLT) can refine the calculations of the helix parameters and the track fitting. Note that this makes the data records variable length. After an event is processed and track information constructed by the Core Crates, they in turn send track information to the FLIC as a data record, using data push. The FLIC receives these records and buffers them, pending output to the ROSs. The event data is identified by the Level 1 ID, along with other information, which is part of the data record header sent by the Core Crate. Since each Core Crate may require different amounts of time for processing a given event, the records from the individual core crates may arrive at the FLIC at different times but always less than a maximum time referenced to the time of the event, and always time-ordered. The Core Crates will send a record even if there is no track information from a particular region for a given event, so that the HLT knows that no event fragments have been lost.

Because of considerations of data rates, each Core Crate will have two transceiver links to the FLIC. Generally the two links will function in a combined way to transfer data to the FLIC, e.g. even/odd data based on the event ID. However, during the Technical Design Report (TDR) process, the core crate decided to use each link to transmit data for half core crate thus two links became independent.

As for data processing in the FLIC, for the baseline design, there will be no event concentration, reconstruction, or overlap removal. The FLIC will combine the two data streams from each Core Crate, and send this information directly to the ROSs as a single data stream. The HLT will perform the event reconstruction. In the baseline design, overlap removal will be done in the Core Crate. However, it is desirable to include an option where additional functionality might be incorporated into the FLIC, such as primary vertex finding or beam spot determination. This will be described in Section 3.

The data received by the FLIC will be buffered in a FIFO, pending processing. After processing (if any), the Input Card must reformat the data to conform to formatting rules for a ROD [2]. The header words must contain specific information about the event, which is part of the data that is sent from the Core Crates. Other header words, as well as the trailer words, will be formed within the FLIC. This is described in Section 3. This data is then stored in an output FIFO, pending transfer to the ROSs. In the case where there is no data processing or event concentration (the baseline design), the FLIC will transfer the data as promptly as it is made available. In this case, the different data streams from the Core Crates may operate independently.

The data will be sent from the FLIC to the ROS using S-Link protocol [3]. The system shall implement two S-Link output channels for each Core Crate data stream (a pair of transceivers from each Core Crate as described above).

In addition to serving as an interface to the Level 2 trigger, the FLIC subsystem shall also provide monitoring and diagnostic capability. This implies a means of controlling the system through peripheral means. The system shall have an interface to the TDAQ control system to facilitate monitoring and control. For monitoring, it is desirable to implement "spy buffers" so that the overall performance of the system can be monitored. For diagnostics, it is desirable to have the ability to process both fixed data and user-defined data through the processing chain. These features will be described in Section 3.

Given the overall features of the FLIC subsystem as described above, a conceptual design of the subsystem is described in Section 2.

2. Conceptual Design

The FLIC will be implemented as a separate crate. Only one crate is needed for the system, and it will process the data streams as received from the eight Core Crates. We propose to use an Advanced Telecommunication Computing Architecture (ATCA) crate for this subsystem. This will allow the implementation of full event processing within the crate as a future option. The architecture required for this functionality is a X2 mesh configuration, where every payload card (Input Card) can communicate with two dedicated slots in the crate. One is used for control and monitoring. The second would be used for future functions, such as event processing or triggering.

In this conceptual design, there would be two Input Cards and two Output Cards. Each Input Card would service 4 Core Crates. Each input from the Core Crates will use two SFP links to send data to the front panel of the FLIC Input Card. The data from each Core Crate will be formatted into the standard ATLAS ROD fragments on the Input Card, and then passed to the Output Card, which is a Rear Transition Module (RTM, in ATCA parlance.) The data is passed from Input Card to Output Card using the zone 3 connector of the ATCA crate. The Output Card holds eight SLINK connectors, with two for each output channel, which send the data to the ROSs. The new ROS deployed after the first LHC long shutdown can handle the 100KHz request rate, so two SLINK connectors will send data to ROS without any additional load balancing operation. The drawing in Figure 3 illustrates the basic concept.

For the basic functionality of the system, Zone 3 of the ATCA crate would be used to send event data from the Input Card to the Output Card. To facilitate an optional trigger processor in the future, the data streams will be copied to Zone 2, utilizing one of the two star connections in the backplane mesh. Each of the mesh networks on the backplane provides four pairs of send and four pairs of receive connections to the dedicated processors in the crate. The trigger connection would use one of these sets. The combined transmission rate of the four pairs is 10 Gbps. For the baseline design, only the connectivity is needed to facilitate a future implementation.

Each Input Card will require the capability to communicate with a single-board processor blade in the crate, to facilitate slow control communication and the readout of monitor information (e.g. spy buffers – see Section 3.4). This function will be provided by the using the second of the two star connections on the Zone 2 backplane mesh. This feature is needed for the baseline design. The blade will communicate with the TDAQ Information Service (IS) system over standard Ethernet. The blade is envisaged to be a commercial processor, and must be compliant with the ATCA Zone 2 connectivity for 10 Gbps communication.

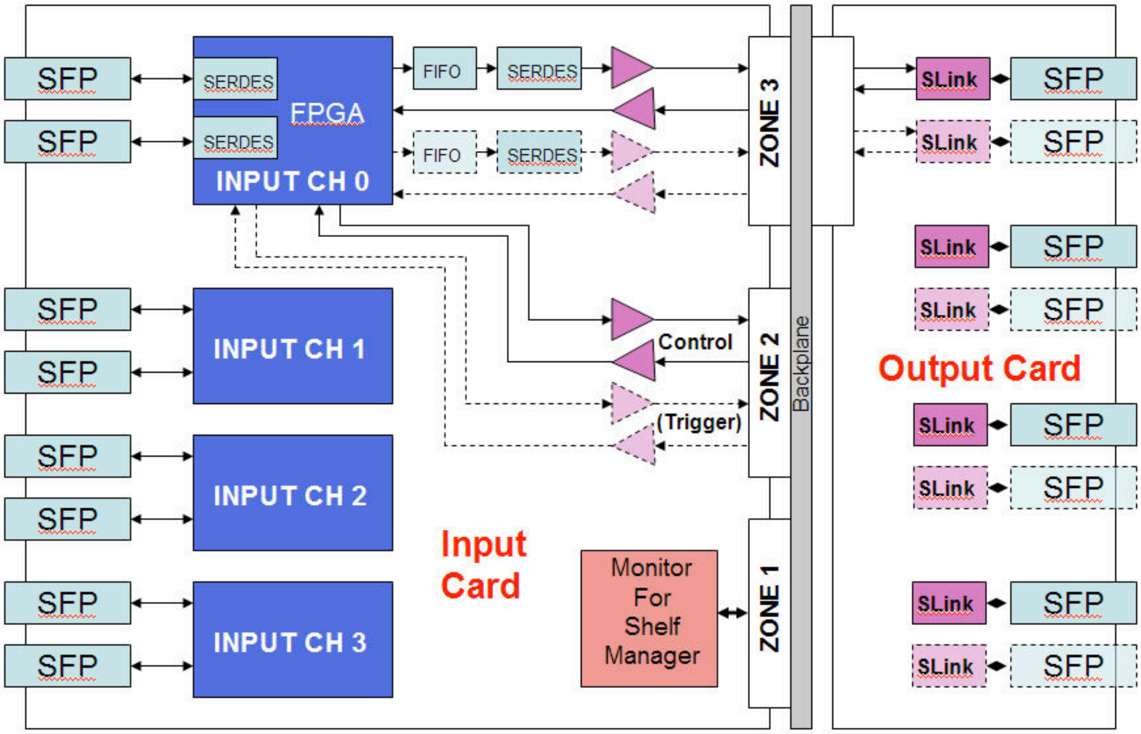

Figure 3. Conceptual design of the FLIC Input & Output Cards in ATCA. Dashed & light colors are optional features.

As shown in Figure 3, the Input Card has a connection to Zone 1. This is a slow control link with the Shelf Manager that is standard in the ATCA crate. This connection will be used to monitor the voltages on the cards, and perhaps other slow control processes.

3. System Specifications

3.1 General Input Data Parameters

3.1.1 Maximum Input Event Rate

The maximum sustained input event rate is 100 KHz. This corresponds to the maximum Level 1 trigger rate for ATLAS.

3.1.2 Average FTK Event Size

An average event will have ~300 tracks, and 12 hits per track. Assume that on the average this data is spread out uniformly in the 8 Core Crates. This is based on assumptions and simulations of the typical event size at design phase-1 luminosity (L ~ 3E34.)

3.1.3 Maximum Input Data Payload Rate into System

Assume ~100 bytes per track, plus an overhead of ~60 bytes per event for header and trailer (See Section 3.2.) From 3.1.1 and 3.1.2, the average data rate into the system is:

100 KHz * [(300 tracks/event * 100 bytes/track) + 60 bytes]
= ~3E9 Bytes/sec

Assuming that the data is evenly distributed among the 8 Core Crates, the rate per Core Crate link is 375 MByte/sec = 3 Gbps.

3.1.4 Total Number of Data Inputs into the System

There will be a maximum of 16 inputs into the system. This corresponds to two data streams per Core Crate.

3.1.5 Input Data Connectivity with Core Crate

The system will use point-to-point connections for the input data from the Core Crates.

3.1.6 Input Data Protocol

The input data from the Core Crates will use a serial bit stream, 8b/10b encoding.

3.1.7 Minimum Link Speed

From 3.1.3 – 3.1.6, the minimum link speed must be:

3E9 bits/sec * 10/8 / (2 links/Core Crate) = 1.875 Gbps.

3.1.8 Input Data Physical Connection

Each input data stream from the Core Crates shall use an SFP transceiver, Finisar FTLF8524P2xNy or equivalent. FTK printed circuit boards shall be designed to meet the mechanical dimensioning provided in the SFP Multi-Sourcing Agreement (MSA).

3.1.8.1 Electrical Signal Connections to Fiber Interface

The physical interconnection to the SFP fiber transceiver modules shall be a direct, unbuffered CML or PECL connection between a compatible FPGA (or SERDES chip) and the SFP device. AC coupling of the serial data shall be used, either provided within the SFP module or on the printed circuit board.

3.1.8.2 SFP General Electrical Requirements

High speed SFP fiber interfaces require a +3.3V power supply with a local LC filter. The SFP cage shall be electrically isolated from the front panel of the PC board but shall be connected to ground through a separate trace distinct from all PC board planes that connects to ground only where the power enters the board (star grounding). The PC board designer is strongly advised to provide multiple grounding points for the SFP cage with stuffing options to either ground the cage to the local ground plane or the distinct return trace.

3.1.8.3 Transmission Medium

The transmission media will be fiber optic. The fiber optic link shall be implemented using two hot-pluggable SFP transceivers each utilizing a duplex LC connector. 50/125um fiber pairs shall be used to minimize loss.

3.1.8.4 Transmission Data Rate

The bi-directional fiber optic link shall be capable of simultaneously transmitting and receiving at 3.25Gb/sec. This is the raw data rate, not the payload data rate.

3.1.9 Input Data Flow Control

The Core Crates operate as "data-push." The receivers for the system shall assert flow control back to the Core Crates using the transmit side of the SFP transceiver. The Input Card shall assert control in the event of buffers nearly full, either in the Level 2 Interface Crate, or by the ROSs. (Format to be specified.)

3.2 Input Data Format

3.2.1 Data Record Defined

The Core Crates develop a set of helix parameters for each track in an event. The resulting information that is sent from the Core Crates to the FLIC is organized in ***records***, each corresponding to one event (one Level 1 Accept). A ***record*** contains multiple ***track frames***, each of which contains information that describes a specific track in that event, including the helix parameters as well as hit information. There may be a variable number of ***frames*** in a ***record***. A ***record*** is terminated by a series of words that comprise the ***record trailer***. The general organization of a ***record*** is shown in Fig. 4. The format is described below.

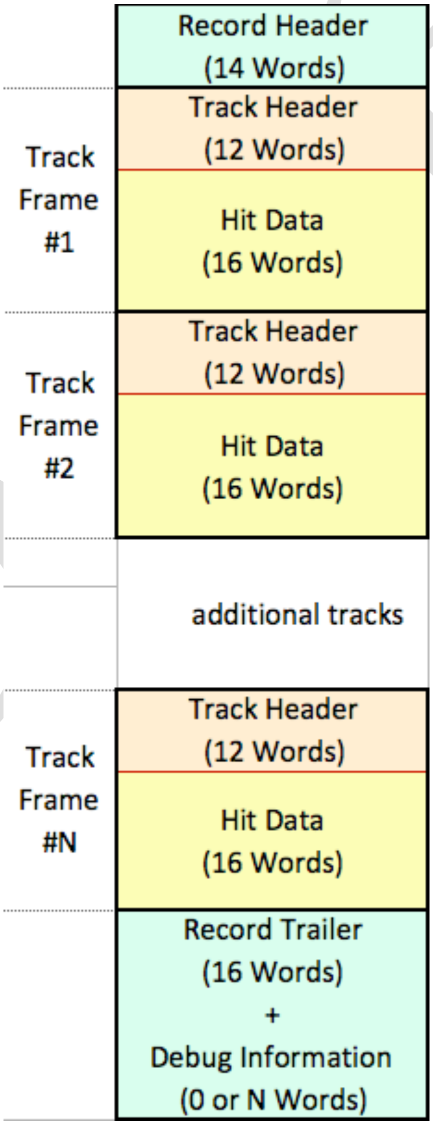

Figure 4. Organization of a data record.

3.2.2 Data Record Rules

Event frame arrival at the FTK-LVL2 Interface from each Core Crate shall be time ordered or sequential with Level 1 event number. This is desirable in order to aid in efficient event construction in the FTK-LVL2 Interface. Independent of an event's track occupancy, each core crate shall transmit to the FTK-LVL2 Interface an explicit record for each event. Specifically, if no tracks are found in a core crate for a given event, the core crate is required transmit a record of identical format to a normal event record to the FTK-LVL2 Interface that corresponds to no tracks having been found for that event. This shall be the primary method of determining when a full event has been transferred to the FTK-LVL2 Interface.

3.2.3 Data Record Format

Each ***record*** begins with a ***record header*** that describes the event. The first 4 words contains a fixed binary code to provide a way to identify the start of a record. Each ***record*** ends with a ***record trailer,*** which contains a series of words with fixed binary codes to identify the end of the record. In the middle are ***track frames*** containing the data received from the tracking detectors, that the FLIC must modify on-the-fly to merge in detector geometry information. The details are described below.

3.2.3.1 The format of the ***record header*** is shown in Table 1. It contains 14 words as shown, with a fixed format. The first 4 words of the ***record header*** are a fixed binary code to allow the FTK-LVL2 Interface to identify the start of a record. The RH02 are set to 0xCAFx where the last four bits are the slink status word that can not be fixed by the upstream. Note that in the data emulation mode 0xCAFx is to be set as 0xCAFE.

3.2.3.2 RH07 plus RH08 gives the 0x0+31bits L1ID. Two models have been implement in the SSBe: 1. Extended L1ID, same as the realistic data. The last 24 bits works as counter. The first 6 bits starts from 1 and increase when the last 24 bits saturate.

3.2.3.3 The format of the ***record trailer*** is shown in Table 2. It contains 16 words as shown, with a fixed format.

- In the first two words 0xE0DAxxxx, the first word RTF01 is a fixed binary code 0xE0DA which identifies the beginning of the record trailer. The second word RTF02, the bits 15-4 are the length of the debug words, while the bits 3-0 are the slink status word that can not be fixed by the upstream.
- RTF03 (0xE0DF) is used as the identifier for the end of the debugging information in the record trailer. For word RTF04, bits 15-4 are the length of the debug words

while the bits 3-0 are the slink status word that can not be fixed by the upstream.

- The RTFD are debug information. If there are no debug information the length of the RTFD will be 0; if there is debug information the length of the RTFD will be fixed N. Note the N has not been fixed in v1.7.2 yet.
- The RTF05 plus RTF06 gives the 0x0+31bits L1ID. It is simple counter counting how many event record has been generated in the SSBe.
- The last four words (0xE0F0 0xA5A5 0x5A5A 0x0E0F) comprise a fixed binary code that identifies the end of the event record.

RH01	B	0	F	0
RH02	C	A	F	x
RH03	F	F	1	2
RH04	3	4	F	F
RH05	0	Run Number [30:16]		
RH06	Run Number [15:0]			
RH07	Extended Level 1 ID [31:16]			
RH08	Extended Level 1 ID [15:0]			
RH09	Reserved			
RH10	Reserved	BCID [11:0]		
RH11	Reserved			
RH12	Reserved		Level 1 Trigger Type [7:0]	
RH13	Reserved		Detector Event Type [7:0]	
RH14	Reserved			TIM [3:0]

Table 1. Format of the Record Header

RTF1	E	0	D	A
RTF2	Length of Debug Block			🟥
RTFD(1)	debug information			
RTFD(N)	debug information			
RTF3	E	0	D	F
RTF4	Length of Debug Block			🟥
RTF5	L1ID [31:16]			
RTF6	L1ID [15:0]			
RTF7	Error Flag [31:16]			
RTF8	Error Flag [15:0]			
RTF9	Reserved			
RTF10	Reserved			
RTF11	Reserved			
RTF12	Reserved			
RTF13	E	0	F	0
RTF14	A	5	A	5
RTF15	5	A	5	A
RTF16	0	E	0	F

Table 2. Format of the Record Trailer

3.2.4 Track Frame Input Format

Each ***track frame*** contains information that describes one track within a given event. A ***frame*** begins with a ***track header*** that contains twelve 16-bit words, which contain FTK local identifiers and the track helix parameters. This is followed by a ***hit data block*** consisting of sixteen 16-bit words that enumerate the hit data ordered by detector layer. The format of a ***track frame*** is shown in Table 3. The details are described below.

3.2.4.1 The format of the ***track header*** is shown in light red in Table 3. It contains 12 words as shown, with a fixed format and fixed length.

- Bits 11:0 of the first word of the ***track header*** are a fixed binary code to allow the FTK-LVL2 Interface to identify the start of a ***track frame***.
- Bits 12 of the first word of the ***track header*** is the Link ID which identifies from which link SSB sends this record to the FLIC (pSSB – 0, all tracks

are from A-Side; fSSB – 1, all tracks are from C-side)

- Bits 15:0 of the second word is the sector number of the track from the Core Crate.
- Bits 6:0 of the third word is the Tower ID of the track from the Core Crate.
- Bits 11:8 of the third word is the number of the track fit from the Core Crate.
- Bits 11:0 of the fourth word is the Layer map for the hit data.
- Bits 7:0 of the fifth word, the sixth word are the ROAD ID.
- Words TH07 – TH12 contain the helix parameters for the track.

3.2.4.2 The format of the ***hit data block*** is shown in light yellow in Table 3. It contains 14 words as shown, with a fixed format and fixed length. Bit 15 in words IBLb, PL0b, PL1b and PL2b indicates whether the cluster exceeds the sized of pixel clustering grid (s = 0 or 1). The data is arranged by tracker layer as follows:

- Words IBLa/IBLb : IBL Layer
- Words PL0a/PL0b: Pixel Layer 0
- Words PL1a/PL1b: Pixel Layer 1
- Words PL2a/PL2b: Pixel Layer 2
- Word SAx0: SCT Axial Layer 0
- Word SSt0: SCT Stereo Layer 0
- Word SAx1: SCT Axial Layer 1
- Word SSt1: SCT Stereo Layer 1
- Word SAx2: SCT Axial Layer 2
- Word SSt2: SCT Stereo Layer 2
- Word SAx3: SCT Axial Layer 3
- Word SSt3: SCT Stereo Layer 3

<table>
<tr><td>TH1</td><td colspan="2">Res [3:0]</td><td>L</td><td colspan="2">B</td><td colspan="2">D</td><td>A</td></tr>
<tr><td>TH2</td><td colspan="8">SECTOR NUMBER</td></tr>
<tr><td>TH3</td><td colspan="3">Reserved</td><td colspan="2">TF#</td><td>Res</td><td colspan="2">Tower Number</td></tr>
<tr><td>TH4</td><td colspan="3">Reserved</td><td colspan="5">Layer Map [11:0]</td></tr>
<tr><td>TH5</td><td colspan="5">Reserved</td><td colspan="3">ROAD ID [23:16]</td></tr>
<tr><td>TH6</td><td colspan="8">ROAD ID [15:0]</td></tr>
<tr><td>TH7</td><td colspan="8">TRACK CHISQ</td></tr>
<tr><td>TH8</td><td colspan="8">TRACK D0</td></tr>
<tr><td>TH9</td><td colspan="8">TRACK Z0</td></tr>
<tr><td>TH10</td><td colspan="8">TRACK COTTH</td></tr>
<tr><td>TH11</td><td colspan="8">TRACK PHI0</td></tr>
<tr><td>TH12</td><td colspan="8">TRACK CURV</td></tr>
<tr><td>IBLa</td><td>0</td><td colspan="2">COL WIDTH</td><td colspan="5">COL COORDINATE [11:0]</td></tr>
<tr><td>IBLb</td><td>s</td><td colspan="2">ROW_WIDTH</td><td colspan="5">ROW COORDINATE [11:0]</td></tr>
<tr><td>PL0a</td><td>0</td><td colspan="2">COL WIDTH</td><td colspan="5">COL COORDINATE [11:0]</td></tr>
<tr><td>PL0b</td><td>s</td><td colspan="2">ROW_WIDTH</td><td colspan="5">ROW COORDINATE [11:0]</td></tr>
<tr><td>PL1a</td><td>0</td><td colspan="2">COL WIDTH</td><td colspan="5">COL COORDINATE [11:0]</td></tr>
<tr><td>PL1b</td><td>s</td><td colspan="2">ROW_WIDTH</td><td colspan="5">ROW COORDINATE [11:0]</td></tr>
<tr><td>PL2a</td><td>0</td><td colspan="2">COL WIDTH</td><td colspan="5">COL COORDINATE [11:0]</td></tr>
<tr><td>PL2b</td><td>s</td><td colspan="2">ROW_WIDTH</td><td colspan="5">ROW COORDINATE [11:0]</td></tr>
<tr><td>SAx0</td><td>0</td><td colspan="2">HIT2 WIDTH</td><td>ReV</td><td colspan="4">HIT2 COORDINATE [10:0]</td></tr>
<tr><td>SSt0</td><td>0</td><td colspan="2">HIT1 WIDTH</td><td>ReV</td><td colspan="4">HIT1 COORDINATE [10:0]</td></tr>
<tr><td>SAx1</td><td>0</td><td colspan="2">HIT2 WIDTH</td><td>ReV</td><td colspan="4">HIT2 COORDINATE [10:0]</td></tr>
<tr><td>SSt1</td><td>0</td><td colspan="2">HIT1 WIDTH</td><td>ReV</td><td colspan="4">HIT1 COORDINATE [10:0]</td></tr>
<tr><td>SAx2</td><td>0</td><td colspan="2">HIT2 WIDTH</td><td>ReV</td><td colspan="4">HIT2 COORDINATE [10:0]</td></tr>
<tr><td>SSt2</td><td>0</td><td colspan="2">HIT1 WIDTH</td><td>ReV</td><td colspan="4">HIT1 COORDINATE [10:0]</td></tr>
<tr><td>SAx3</td><td>0</td><td colspan="2">HIT2 WIDTH</td><td>ReV</td><td colspan="4">HIT2 COORDINATE [10:0]</td></tr>
<tr><td>SSt3</td><td>0</td><td colspan="2">HIT1 WIDTH</td><td>ReV</td><td colspan="4">HIT1 COORDINATE [10:0]</td></tr>
</table>

Table 3. Format of a track frame describing one track.

Data Processing

3.3.1 Input Buffering

Data received shall be stored in a ring buffer, of depth 20 KByte for each link. This corresponds to ~8 events, assuming 20 KByte/event / 8 Core Crates.

3.3.2 Flow Control to Core Crate

The Input Card shall assert Flow Control back to the Core Crate when the input buffer contents reach 80% of capacity

3.3.3 Time Ordering

Event arrival from each Core Crate must be time ordered or sequential. This is desirable in order to aid in efficient event processing in the ROS Interface.

3.3.4 Event Processing

3.3.4.1 Track data as received from the Core Crates, contains ***Link ID*** and ***Sector Number*** fields that must be translated into ***Module Numbers*** by the FLIC. These Module Numbers provide an index for later data analysis of event geometry.

3.3.4.2 The Module Numbers are loaded into the FLIC by control software at system startup, or whenever new geometry maps are generated.

3.3.4.3 As data is received from the Core Crates, each Event Record is buffered into a FIFO. In parallel with the reception of data, the ***Link ID*** and ***Sector Number*** received in the Track Header of each track frame are used to provide the base address to a fast static RAM lookup table. As each word of the track frame is received from the Core Crate, a pipelined lookup of Module Numbers from the static RAM is merged with the incoming data to form a differently formatted data word that contains all the information from the Core Crate plus the associated Module Number.

3.3.4.3.1 The two 16-bit words received from the Core Crate per pixel layer are reformatted into two 32-bit words in order to pack the Module Number in with the row and column coordinate data. Each 16-bit word received from the Core Crate per SCT layer is expanded to a single 32-bit word containing the Core Crate data plus the Module Number.

3.3.4.3.2 As each Module Number is retrieved from the lookup table, the input word(s) from the input FIFO are flushed, data merged by a state machine, and the newly formed 32-bit words pushed into a second FIFO.

3.3.4.3.3 When an entire track record of Core Crate data has been reformatted and merged with Module Number information, an output state machine then transmits the entire record to the ROS.

3.3.4.4 As an option for future functionality, the system shall include connectivity such that the sixteen data streams from the Core Crate are sent to a central processor board in the system, for the purpose of performing additional processing on reconstructed events, including the option of an advanced lower-level trigger. This functionality is left undefined presently. The system shall be designed to provide this connectivity to allow easy expansion of this functionality.

3.3.5 Maximum Processing Time

The processing time of the FLIC is defined by the size of the events sent by the Core Crate. The data processing function of the FLIC necessitates that at least one full event be buffered within the FLIC at all times. The FLIC shall not internally buffer more than two events' worth of data at any time. Assuming 40 tracks in the average event from a single Core Crate, the average event will be 540 words long. Using a processing rate of 156MHz, two event's buffering time will be less than 7 microseconds. When large events pass through the system this delay may increase to 10-15 usec until those events are processed.

3.3.6 Output Buffering

Output data shall be written to a FIFO of depth 100 KByte, pending transfer to the ROSs. Event arrival from each Core Crate must be time ordered or sequential. This is desirable in order to aid in efficient event processing in the ROS Interface.

3.3.7 Output Data Flow Control

The FLIC subsystem operates as "data-push" in writing data to the ROSs. The ROSs may assert flow control back to the FLIC subsystem. The Input Card shall implement flow control, and suspend the transmission of data until buffers are free in the ROSs. This shall be implemented on a link-by-link basis.

3.4 Monitoring, Diagnostics, and Control

3.4.1 Monitoring Functions

3.4.1.1 A monitor of the number of events received from each Core Crate shall be stored in a register, and made available to the monitor process in the crate. The registers shall be 32-bit, one for each of the eight Core Crates. A mechanism for clearing the registers shall be implemented by a command from the monitor process.

3.4.1.2 A monitor of the number of events sent to each S-link shall be stored in a register, and made available to the monitor process in the crate. The registers shall be 32-bit, one for each of the eight outputs. A mechanism for clearing the registers shall be implemented by a command from the monitor process.

3.4.1.3 A monitor with a list of the sizes of events received from each Core Crate shall be stored in registers, and made available to the monitor process in the crate. (Format, depth and implementation TBD.) A mechanism for clearing the lists shall be implemented by a command from the monitor process.

3.4.1.4 A monitor with a list of the sizes of events sent to each of the S-links shall be stored in registers, and made available to the monitor process in the crate. (Format, etc. should be the same as the item above) A mechanism for clearing the lists shall be implemented by a command from the monitor process.

3.4.1.5 A monitor of the XOFF states shall be stored in a register, and made available to the monitor process in the crate. The registers shall be 16-bit, one for each Core Crate input and one for each S-Link output stream.

3.4.1.6 A process shall be provided to store all event data input data in a circular buffer, which can be read either periodically, or stopped and read if an error occurs. (Buffer depth to be determined.)

3.4.1.7 A process shall be provided to capture every Mth Core Crate input and store the input in a FIFO, which can be read by the monitor process in the crate, where “M” is a programmable quantity.

3.4.1.8 A process shall be provided to read out the occupancy of output and input FIFOs by the monitor process in the crate.

3.4.2 Control Functions

3.4.2.1 A mechanism shall be provided to enable and disable each input channel.

3.4.2.2 A mechanism shall be provided to enable and disable each output channel.

3.4.2.3 A mechanism shall be provided to clear all event output FIFOs.

3.4.2.4 A mechanism shall be provided to clear any Core Crate input FIFOs.

3.4.2.5 A mechanism shall be provided to clear all counters and lists used for monitoring.

3.4.2.6 A mechanism shall be provided to set and reset XOFF for each S-Link output and for each of the input links.

3.4.3 Diagnostic Functions

3.4.3.1 A mechanism shall be provided to process fixed data through the crate. The data should be inserted as close to the Core Crate input as possible. The data should be downloaded from the crate processor and a mechanism to allow Core Crate multiple event downloading must be available. The number of events is TBD.

3.4.3.2 A mechanism shall be provided to loop over, process and output the diagnostic data described above at several rates (TBD) and for a fixed number or unlimited number of repetitions. On the output end the extended L1ID of the data must be incremented by 0x1000000 . This ensures that the events all have unique identifiers (eL1IDs).

3.5 General Output Data Parameters

3.5.1 Maximum Output Event Rate

The system must be capable of sustained operation at the maximum Level 1 trigger rate of 100 KHz.

3.5.2 Average FTK Event Size @ Maximum Event Rate

The average output event size for data received from the Core Crates is 20 KByte/event. This corresponds to 64 bytes/track * 300 tracks/event. This is based on assumptions and simulations of the event size at design phase-1 luminosity (L ~ 3E34.). The FLIC adds only a few additional words of header and trailer, which is small compared to the size of the data received.

3.5.3 Maximum Output Data Payload Rate

The maximum output data rate for the system is 2 GByte/sec. This corresponds to the maximum trigger rate and the average event size. For 100 kHz rate to the ROS, each output link can process records with 22.7 tracks in average.(TO be changed for v1.7.0)

3.5.4 Total Number of Data Outputs from the System

There will be a maximum of 16 outputs from the system. This corresponds to two data streams per Core Crate.

3.5.5 Output Data Connectivity with ROSs

The system will use point-to-point connections for the output data to the ROSs.

3.5.6 Output Data Protocol

The output data will use S-Link protocol as a serial bit stream. The system shall use fixed-width, 32-bit transfers. The data payload transfer rate is specified at 400 MB/sec.

3.5.7 Output Data Physical Connection

Each output data stream from the Core Crates shall use an SFP transceiver, Finisar FTLF8524P2xNy or equivalent. FTK printed circuit boards shall be designed to meet the mechanical dimensioning provided in the SFP Multi-Sourcing Agreement (MSA).

3.5.7.1 Electrical Signal Connections to Fiber Interface

The physical interconnection to the SFP fiber transceiver modules shall be a direct, unbuffered CML or PECL connection between a compatible FPGA (or SERDES chip) and the SFP device. AC coupling of the serial data shall be used, either provided within the SFP module or on the printed circuit board.

3.5.7.2 SFP General Electrical Requirements

High speed SFP fiber interfaces require a +3.3V power supply with a local LC filter. The SFP cage shall be electrically isolated from the front panel of the PC board but shall be connected to ground through a separate trace distinct from all PC board planes that connects to ground only where the power enters the board (star grounding). The PC board designer is strongly advised to provide multiple grounding points for the SFP cage with stuffing options to either ground the cage to the local ground plane or the distinct return trace.

3.5.7.3 Transmission Medium

The transmission media will be fiber optic. The fiber optic link shall be implemented using two hot-pluggable SFP transceivers each utilizing a duplex LC connector. 50/125um fiber pairs shall be used to minimize loss.

3.5.7.4 Transmission Data Rate

The bi-directional fiber optic link shall be capable of simultaneously transmitting and receiving at 3.25Gb/sec. This is the raw data rate, not the payload data rate.

3.5.8 Output Data Flow Control

The system operates as “data-push” to the ROSs. The ROS can assert flow control back to the Core Crates using the transmit side of the SFP transceiver. The Output Card may receive flow control in the event of buffers nearly full, either in the Level 2 Interface Crate, or by the ROSs.

3.6 Output Data Format

3.6.1 Record Defined

The ROS expects the data sent to it to have a format as shown in Figure 4. The FLIC will use the left-most configuration of the data payload.

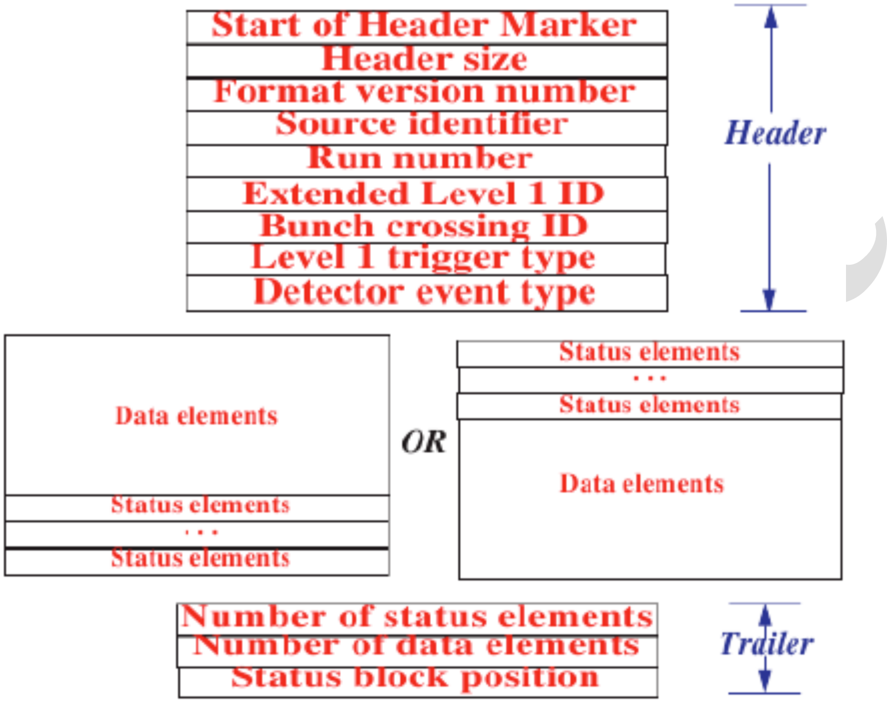

Figure 4. General Format of ROD Data

3.6.2 Output Data Header Defined

The header has the following elements (9 long-words, 36 bytes total), as defined by the S=Link specification

3.6.2.1 Start of Header Marker
Length: 32 bits
Encoding: binary, fixed value for a ROD: 0xEE1234EE
Source: Formed within the FLIC

3.6.2.2 Header Size
Length: 32 bits
Encoding: binary, fixed value 0x09
Source: Formed within the FLIC

3.6.2.3 Format Version Number
Length: 32 bits
Encoding: binary, global variable
Source: Formed within the FLIC from firmware

3.6.2.4 Source Identifier
Length: 32 bits
Encoding: unique Subdetector ID, fixed value 0x7f for FTK (bits 23:16). Module ID for each output S-link to ROS (bits 15:0), 0x0000-0x000F (TBD).
Source: Formed within the FLIC

3.6.2.5 Run Number
Length: 32 bits
Encoding: binary integer, unique number for the experiment
Source: Passed to FLIC from Core Crate

3.6.2.6 Extended Level 1 ID
Length: 32 bits
Encoding: binary, 24-bit L1ID + 8-bit ECRID
Source: Passed to FLIC from Core Crate

3.6.2.7 Bunch Crossing ID
Length: 32 bits
Encoding: binary, 12-bit bunch crossing ID, padded to 32 bits
Source: Passed to FLIC from Core Crate

3.6.2.8 Level 1 Trigger Type
Length: 32 bits
Encoding: binary, 8-bits from L1 Trigger, padded to 32 bits
Source: Passed to FLIC from Core Crate

3.6.2.9 Detector Event Type
Length: 32 bits
Encoding: binary, detector specific
Source: Passed to FLIC from Core Crate

3.6.3 Output Data Payload Formatting Defined

The data formatting follows from Section 3.6.1. The data must be packed into 32-bit words, as shown in Table 4.

3.6.3.1 There are an indeterminate number of track records, consisting of *Track Headers* (shaded orange) and *Track Data* (shaded yellow, two intensities to separate Pixel from SCT). Each track record defines a single track and is of fixed length. The track records are derived from Core Crate data described previously in Section 3.2.4.

3.6.3.2 The words following the track records are debug information and global quantities (shaded white). Total N/2+4 words if there is debug information; 4 words if there is no debug information.

3.6.4 Status Elements

3.6.4.1 Then up to one long words followed are the FLIC status (presently undefined.). Zero word if there is no FLIC error, otherwise one status word will be created.

<table>
<tr><th></th><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>09</th><th>08</th><th>07</th><th>06</th><th>05</th><th>04</th><th>03</th><th>02</th><th>01</th><th>00</th></tr>
<tr><td>SH01</td><td colspan="32">Start of Header Marker - Fixed Value 0xEE1234EE</td></tr>
<tr><td>SH02</td><td colspan="32">Header Size - Fixed Value 0x00000009</td></tr>
<tr><td>SH03</td><td colspan="32">Format Version Number - Formed within FLIC from firmware</td></tr>
<tr><td>SH04</td><td colspan="32">Source Identifier - Fixed Value 0x0000007F</td></tr>
<tr><td>RH01</td><td>0</td><td colspan="31">Run Number [30:0]</td></tr>
<tr><td>RH02</td><td colspan="32">Extended Level 1 ID [31:0]</td></tr>
<tr><td>RH03</td><td colspan="20">Reserved</td><td colspan="12">BCID [11:0]</td></tr>
<tr><td>RH04</td><td colspan="24">Reserved</td><td colspan="8">Level 1 Trigger Type [7:0]</td></tr>
<tr><td>RH05</td><td colspan="8">Reserved</td><td colspan="8">Detector Event Type [7:0]</td><td colspan="12">Reserved</td><td colspan="4">TIM [3:0]</td></tr>
<tr><td>TH1</td><td colspan="3">Res [3:0]</td><td>L</td><td colspan="4">B</td><td colspan="4">D</td><td colspan="4">A</td><td colspan="16">SSB SECTOR NUMBER</td></tr>
<tr><td>TH2</td><td colspan="4">Reserved</td><td colspan="4">TF#</td><td>Res</td><td colspan="7">Tower Number</td><td colspan="4">Reserved</td><td colspan="12">Layer Map [11:0]</td></tr>
<tr><td>TH3</td><td colspan="8">Reserved</td><td colspan="24">ROAD ID [23:0]</td></tr>
<tr><td>TH4</td><td colspan="16">TRACK CHISQ</td><td colspan="16">TRACK D0</td></tr>
<tr><td>TH5</td><td colspan="16">TRACK Z0</td><td colspan="16">TRACK COTTH</td></tr>
<tr><td>TH6</td><td colspan="16">TRACK PHI0</td><td colspan="16">TRACK CURV</td></tr>
<tr><td>IBLa</td><td colspan="16">Reserved</td><td colspan="4">Reserved</td><td colspan="12">Module ID[11:0]</td></tr>
<tr><td>IBLb</td><td>0</td><td colspan="3">COL WIDTH</td><td colspan="12">COL COORDINATE [11:0]</td><td>s</td><td colspan="3">ROW_WIDTH</td><td colspan="12">ROW COORDINATE [11:0]</td></tr>
<tr><td>PL0a</td><td colspan="16">Reserved</td><td colspan="4">Reserved</td><td colspan="12">Module ID[11:0]</td></tr>
<tr><td>PL0b</td><td>0</td><td colspan="3">COL WIDTH</td><td colspan="12">COL COORDINATE [11:0]</td><td>s</td><td colspan="3">ROW_WIDTH</td><td colspan="12">ROW COORDINATE [11:0]</td></tr>
<tr><td>PL1a</td><td colspan="16">Reserved</td><td colspan="4">Reserved</td><td colspan="12">Module ID[11:0]</td></tr>
<tr><td>PL1b</td><td>0</td><td colspan="3">COL WIDTH</td><td colspan="12">COL COORDINATE [11:0]</td><td>s</td><td colspan="3">ROW_WIDTH</td><td colspan="12">ROW COORDINATE [11:0]</td></tr>
<tr><td>PL2a</td><td colspan="16">Reserved</td><td colspan="4">Reserved</td><td colspan="12">Module ID[11:0]</td></tr>
<tr><td>PL2b</td><td>0</td><td colspan="3">COL WIDTH</td><td colspan="12">COL COORDINATE [11:0]</td><td>s</td><td colspan="3">ROW_WIDTH</td><td colspan="12">ROW COORDINATE [11:0]</td></tr>
<tr><td>SAx0</td><td>0</td><td colspan="3">HIT2 WIDTH</td><td>RSV</td><td colspan="11">HIT2 COORDINATE [10:0]</td><td colspan="4">Reserved</td><td colspan="12">Module ID[11:0]</td></tr>
<tr><td>SSt0</td><td>0</td><td colspan="3">HIT2 WIDTH</td><td>RSV</td><td colspan="11">HIT1 COORDINATE [10:0]</td><td colspan="4">Reserved</td><td colspan="12">Module ID[11:0]</td></tr>
<tr><td>SAx1</td><td>0</td><td colspan="3">HIT2 WIDTH</td><td>RSV</td><td colspan="11">HIT2 COORDINATE [10:0]</td><td colspan="4">Reserved</td><td colspan="12">Module ID[11:0]</td></tr>
<tr><td>SSt1</td><td>0</td><td colspan="3">HIT2 WIDTH</td><td>RSV</td><td colspan="11">HIT1 COORDINATE [10:0]</td><td colspan="4">Reserved</td><td colspan="12">Module ID[11:0]</td></tr>
<tr><td>SAx2</td><td>0</td><td colspan="3">HIT2 WIDTH</td><td>RSV</td><td colspan="11">HIT2 COORDINATE [10:0]</td><td colspan="4">Reserved</td><td colspan="12">Module ID[11:0]</td></tr>
<tr><td>SST2</td><td>0</td><td colspan="3">HIT2 WIDTH</td><td>RSV</td><td colspan="11">HIT1 COORDINATE [10:0]</td><td colspan="4">Reserved</td><td colspan="12">Module ID[11:0]</td></tr>
<tr><td>SAx3</td><td>0</td><td colspan="3">HIT2 WIDTH</td><td>RSV</td><td colspan="11">HIT2 COORDINATE [10:0]</td><td colspan="4">Reserved</td><td colspan="12">Module ID[11:0]</td></tr>
<tr><td>SSt3</td><td>0</td><td colspan="3">HIT2 WIDTH</td><td>RSV</td><td colspan="11">HIT1 COORDINATE [10:0]</td><td colspan="4">Reserved</td><td colspan="12">Module ID[11:0]</td></tr>
<tr><td>RF0</td><td colspan="4">E</td><td colspan="4">0</td><td colspan="4">D</td><td colspan="4">A</td><td colspan="16">Length of Debug Block (N)</td></tr>
<tr><td>RFd</td><td colspan="16">debug information</td><td colspan="16">debug information</td></tr>
<tr><td>RF1</td><td colspan="4">E</td><td colspan="4">0</td><td colspan="4">D</td><td colspan="4">F</td><td colspan="16">Length of Debug Block (N)</td></tr>
<tr><td>RF2</td><td colspan="16">L1ID [31:16]</td><td colspan="16">L1ID [15:0]</td></tr>
<tr><td>RF3</td><td colspan="16">Error Flag [31:16]</td><td colspan="16">Error Flag [15:0]</td></tr>
<tr><td>RF4</td><td colspan="16">Reserved</td><td colspan="16">Reserved</td></tr>
<tr><td>RF5</td><td colspan="16">Reserved</td><td colspan="16">Reserved</td></tr>
<tr><td>RF6</td><td colspan="16">FLIC Status (format TBD)</td><td colspan="16">FLIC Status (format TBD)</td></tr>
<tr><td>SLF1</td><td colspan="16">Number of Status Elements - Bytes 3-2, 0x0000</td><td colspan="16">Number of Status Elements - Bytes 1-0, 0x0000</td></tr>
<tr><td>SLF2</td><td colspan="16">Number of Data Elements, Bytes 3-2</td><td colspan="16">Number of Data Elements, Bytes 1-0 (TH1-RF4)</td></tr>
<tr><td>SLF3</td><td colspan="16">Status Block Position - Bytes 3-2, Fixed Value 0x0000</td><td colspan="16">Status Block Position - Bytes 1-0, Fixed Value 0x0001</td></tr>
</table>

Table 4. Format of Output Data Payload

3.6.5 Output Data Trailer Defined

The trailer has the following elements (3 long-words, 12 bytes total):

3.6.5.1 Number of Status Elements
Length: 32 bits
Encoding: binary, the number of FLIC status words, up to 0x10
Source: Formed within the FLIC

3.6.5.2 Number of Data Elements
Length: 32 bits
Encoding: binary, total number of data payload words
Source: Formed within the FLIC

3.6.5.3 Status Block Position
Length: 32 bits
Encoding: binary, fixed value, 0x01
(status block follows data)
Source: Formed within the FLIC

References

[1] M. Shochet, et al., “FTK: A Hardware Track Finder for the ATLAS Trigger Technical Proposal, ” CERN, EDMS: ATL-D-ES-0036 v1.0.

[2] ATLAS TDAQ Group, “The Raw Event Format in the ATLAS Trigger & DAQ,” CERN, EDMS: ATL-D-ES-0019

[3] R. McLaren et al., “The S-LINK Interface Specification,” CERN, EDMS: 110828 v.4.