DEVELOPMENT OF A TIMING AND DATA LINK FOR EIC COMMON HARDWARE PLATFORM*

Paul Bachek†, Thomas Hayes, Kevin Mernick, Geetha Narayan, Freddy Severino
C-AD, Brookhaven National Laboratory, Upton, NY, USA
Joseph Mead, NSLS-II, Brookhaven National Laboratory, Upton, NY, USA

Abstract

Modern timing distribution systems benefit from high configurability and the bidirectional transfer of timing data. The Electron Ion Collider (EIC) Common Hardware Platform (CHP) will integrate the functions of the existing RHIC Real Time Data Link (RTDL), Event Link, and Beam Sync Link, along with the Low-Level RF (LLRF) system Update Link (UL), into a common high speed serial link. One EIC CHP carrier board supports up to eight external 8 Gbps high speed links via SFP+ modules, as well as up to six 8 Gbps high speed links to each of two daughterboards. A daughterboard will be designed for the purpose of timing data link distribution for use with the CHP. This daughterboard will have two high speed digital crosspoint switches and a Xilinx Artix Ultrascale+ FPGA onboard with GTY transceivers. One of these will be dedicated for a high-speed control and data link directly between the onboard FPGA and the carrier FPGA. The remaining GTY transceivers will be routed through the crosspoint switches. The daughterboard will support sixteen external SFP+ ports for timing distribution infrastructure with some ports dedicated for transmit only link fanout. The timing data link will support bidirectional data transfer including sending data or events from a downstream device back upstream. This flexibility will be achieved by routing the SFP+ ports through the crosspoint switches which allows the timing link datapaths to be forwarded directly through the daughterboard to the carrier and into the FPGA on the daughterboard in many different configurations.

INTRODUCTION

To support the EIC project, an upgraded timing distribution network is being developed to take the place of several existing distributed timing links [1][2][3]. The upgraded timing data link (TDL) will be the primary mechanism for connecting devices in the accelerator complex that require real time information about the machine. The TDL provides high speed low latency deterministic timing data distribution to connected devices. It is used to send data and status information for distributed feedback and machine protection systems. Notable improvements over previous implementations are more efficient transfer of data bidirectionally, and more configurable data filters. The TDL defines the network protocol and structure which will be realized by interconnected CHP systems. The CHP systems will support various pluggable special function

daughtercards, one of which is dedicated to TDL distribution. This daughtercard in conjunction with the CHP carrier will constitute the backbone of the TDL network infrastructure.

TIMING DATA LINK

The TDL is a networking protocol which defines layers one through four of the OSI model [4], consisting of the physical, data link, network, and transport layers. These layers represent a standard conceptual segmentation of functionality, the actual implementation doesn't necessarily have such a clear separation of components. The TDL network allows devices connected in a tree structure to reliably communicate prioritized timing critical information bidirectionally with deterministic timing when necessary.

Physical Layer

The TDL physical layer is responsible for the transfer of raw data bytes between directly connected systems. Data bytes of 8 bits are encoded into symbols of 10 bits using the 8b/10b encoding scheme. Symbols are then transferred serially one bit at a time, LSB first, at a rate of 8 Gbps. Accounting for the encoding overhead, a data throughput of 6.4 Gbps is achieved. Data bits are electrically represented as differential pair CML signals. A link has two AC coupled differential pairs, one each for transmitted and received data. The 8b/10b encoding scheme keeps track of running disparity and ensures sufficient transitions for clock recovery and DC balance on the line. Data byte alignment is achieved by detecting valid 8b/10b K.28.1 comma symbols.

The CML electrical signal pairs are converted to optical signals using an SFP+ module optical transceiver. Each optical transceiver supports two optical fiber connections, one for each direction of data transfer. The physical medium for transmission of the optical signals is single-mode or multi-mode optical fiber. The optical fiber will make the long runs necessary to connect systems separated by a large physical distance.

A key feature of the physical layer is clock recovery from the received data. The 8 Gbps data line rate is transmitted synchronous to the accelerator master oscillator running at 100 MHz. A clock and data recovery (CDR) unit inside the PHY provides a divided down clock of the line frequency to the receiver. This recovered clock is then used as a synchronous copy of the 100 MHz master oscillator clock.

Data Link Layer

The TDL data link layer is responsible for transferring data words between directly connected systems. No

hardware addressing is implemented in the TDL so words are transferred between devices in a point-to-point fashion. A TDL data word is made up of 64 bits of data. Words are transferred one byte at a time in big endian format. Word alignment is achieved similarly to byte alignment by detecting K.28.1 comma symbols. A comma symbol appears as the least significant byte of every idle word. Other control (K) symbols may be used for out-of-band signalling to facilitate extended functionality such as link latency measurements.

At least two idle words are guaranteed to be transmitted within a 10 µs time interval to support word alignment within a reasonable time. Once synchronized, data words are transferred at a rate of 100 MHz. Word synchronization has the added benefit of phase aligning the recovered 100 MHz clock with the transmitter clock. This way one word is transferred per cycle of the accelerator master oscillator.

Network Layer

The TDL network layer is responsible for crafting and routing data packets between networked systems. A TDL data packet is made up of 64 bits of data broken up into four 16-bit fields. The fields in a packet are encoded as shown in Table 1. Packets are of fixed size matching the size of a word and therefore one word always contains exactly one packet.

Table 1: TDL Data Packet Encoding

Packet Type	Bits 63:48	Bits 47:32	Bits 31:16	Bits 15:0
Data	Nonzero PID	Data	Data	Data
Event	0x0000	0x0000	0x0000	Event ID
Reserved	0x0000	0x0000	Nonzero	Any
Reserved	0x0000	Nonzero	Any	Any

The upper field of a packet represents the packet identifier (PID). The PID of 0x0000 is reserved to represent special function packets. An event packet is encoded with the upper three fields as all zero and the lower field as the event code. Packets with nonzero PIDs may use the remaining 48 bits to encode data as needed. This allows packets to be filtered by their PID, or other fields such as event ID (EID), to limit unnecessary traffic on network segments where applicable.

Devices which generate timing data are referred to as timing data generators (TDG). The overall network topology follows a tree structure with a global TDG (GTDG) at the root node. Timing data packets are broadcast to all connected devices at the next level down the tree structure referred to as downstream TDGs (DTDG). The DTDGs can then filter packets based on their contents and rebroadcast the relevant traffic to DTDGs or endpoint devices further downstream. Endpoints can be thought of as leaf nodes in the tree structure. The network structure can support any

tree breadth and depth only limited by ports available on a DTDG.

Endpoint devices send packets to an upstream TDG (UTDG) towards the GTDG but they must contend with each other for access to the limited upstream links. Upstream packets are assigned equal priority and are arbitrated in a round robin fashion. This introduces non-determinism in upstream packet transmission. The data sent upstream to a DTDG can then be rebroadcast downstream to other branches of the tree. Packets received from UTDGs have the highest priority on the link to facilitate deterministic transmission of critical timing data downstream, especially from the GTDG. Packets being sent upstream for rebroadcast have the lowest priority when contending for the downstream broadcast link which introduces another non-determinism in upstream data transmission. Locally generated events and data within a TDG have the second and third level of priority, respectively. A block diagram of the packet routing logic is shown in Figure 1. Once a packet makes it onto the downstream link, it is then guaranteed to be delivered with deterministic timing to all devices on the subtree.
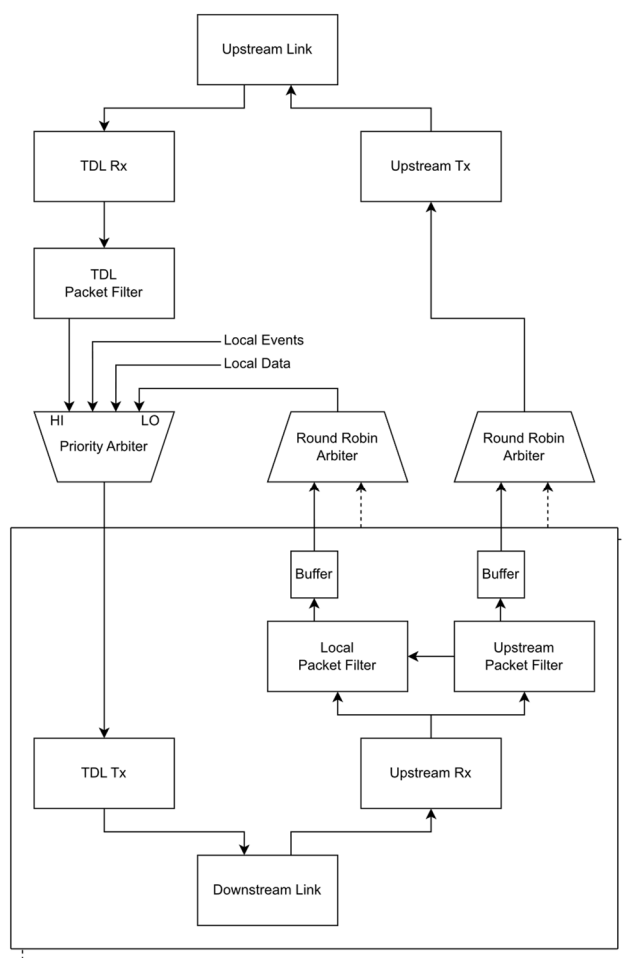

Figure 1: TDL packet routing logic.

The packet filters may contain many entries each of which define a bit mask followed by a pattern match. This allows only pertinent fields within a data packet to be

masked out and filtered on to match a specific pattern. Packets being sent upstream can be rebroadcast locally or be sent further upstream. These two routing options are mutually exclusive with precedence given to packets heading further upstream. This is indicated in the diagram by the arrow from the upstream filter to the local filter representing a logical gate to prevent replication of packets.

Transport Layer

The TDL transport layer is responsible for the reliable transmission of datagrams called "Update Frames" at a fixed interval. The GTDG generates an update packet followed by a timestamp packet once every 10 µs. An Update Frame consists of 1000 TDL packets that are transmitted during this interval, some of which may be idle words. This includes the update packet, timestamp packet, at least two idle words, and any additional data packets which may be transmitted during this interval.

Not all devices on the network will receive every packet within an Update Frame as some may be filtered out. This is a desirable feature of the TDL which helps to conserve bandwidth on network segments which only require a subset of the timing data available on the link. Every device must receive the update and timestamp packets at a minimum in order to achieve successful reception of an Update Frame. Update packets, also called "Update Events" are transmitted as event packets with the special EID of 0x0001. An Update Event can be thought of as the start of frame for an Update Frame. It is followed by a timestamp packet which contains the current real time clock value.

Timestamp packets are encoded as data packets with PID 0xA001 as shown in Table 2 with fields for seconds and ticks. The seconds field is a count of the number of seconds since the defined time zero and works like Unix time but with the beginning of the epoch shifted. This was done due to using only 31 bits for the timestamp. For the current LLRF update link, the offset between Unix time and UL timestamps is 0x3B02DC40 seconds. The rollover of the UL timestamps will occur in 2069, rather than in 2038 if they had not been offset from Unix time. The TDL timestamp epoch will be redefined prior to EIC commissioning to push the timestamp rollover further into the future. The ticks field is a count of 10 µs ticks elapsed since the last second.

Table 2: Timestamp Data Packet Encoding

Packet Type	Bits 63:48	Bits 47:17	Bits 16:0
Timestamp	PID 0xA001	Seconds	Ticks

HARDWARE IMPLEMENTATION

Implementation of the TDL is achieved by use of CHP systems equipped with SFP+ networking daughtercards interconnected by optical fiber. A CHP system can be configured as a GTDG or DTDG, generally referred to as a TDG. The TDL network follows a tree structure with the GTDG as the root node, DTDG as subtree nodes, and other devices as leaf nodes.

CHP systems make use of Xilinx Ultrascale+ devices for their high-speed multi-gigabit transceivers (MGT) and programmable logic. The MGTs implement the physical TDL network electrical interface. Packet filtering and link aggregation are performed in the programmable logic fabric. A CHP system equipped with two SFP+ daughtercards will constitute a typical TDG implementation.

CHP Carrier

The CHP carrier is the main system constituting a TDG. It has two slots which each accommodate a special function daughtercard. In the configuration as a TDG, both daughtercard sites will typically be populated by SFP+ daughtercards. The CHP itself can support up to eight SFP+ connections without any daughtercards fitted. A diagram of the datapaths on the CHP is shown in Figure 2 with arrows indicating the directionality of dataflow. The datapaths are highly configurable to support many different connection configuration options.
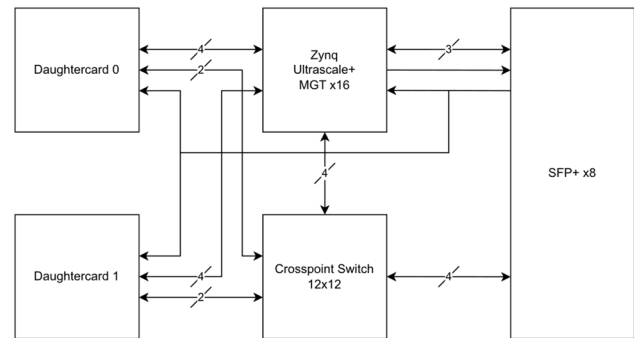

Figure 2: CHP carrier datapaths.

The CHP carrier has a Zynq Ultrascale+ SoC with 16 GTH MGTs and a 12x12 crosspoint switch. Each MGT channel implements one transmit and one receive datapath. The 16 MGT channels are split up with four connections to SFP+ ports, four connections to the crosspoint switch, and four connections to each daughtercard site. The crosspoint switch also has four connections to SFP+ ports and two connections to each daughtercard site. One of the SFP+ ports connected to an MGT channel is dedicated to TDL fanout with the receiver path fanned out as transmit only to each daughtercard site.

The CHP carrier provides flexible clocking options to provide its own 100 MHz reference clock for the MGT transceivers either internally, via an external reference, or by clock recovery from the TDL. Possible clock sources include analog PLLs, a digital PLL, beam synchronous clocks, and an external clock. These clocks are all forwarded to both daughtercard sites as well.

SFP+ Daughtercard

The SFP+ daughtercard is analogous to a network switch specific to the TDL network. It supports 16 SFP+ ports, 13 bidirectional ports, and three transmit-only ports for unidirectional broadcasting of the TDL. An Artix Ultrasale+ FPGA with 12 GTY transceivers is used for link aggregation with the aid of two 12x12 crosspoint switches.

A diagram of the datapaths on the SFP+ daughtercard is shown in Figure 3. The connections to the CHP carrier are labelled with the devices to which they connect on the carrier side.
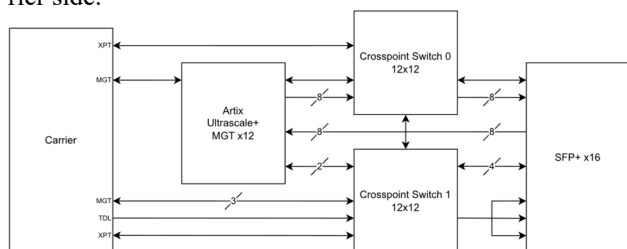

Figure 3: SFP+ daughtercard datapaths.

One MGT channel from the carrier is directly connected to another MGT channel on the daughtercard to be used as a dedicated chip to chip communication link. The other MGT channels and crosspoint switch connections from the carrier are all connected to a crosspoint switch on the daughtercard for high configurability. Eight of the MGT channels are connected to SFP+ ports through a crosspoint switch on the transmitter, but directly connected on the receiver. This allows the downstream data source to be configurable while upstream data will always be sent to the FPGA for filtering and aggregation.

The MGTs include hardware built in for 8b/10b encoding as well as comma symbol detection. This is configured to implement data byte alignment and the datapath width is configured as 64-bit to provide word alignment as well. The MGT also has a CDR unit which provides a recovered clock from the received data stream. This clock is 100 MHz and synchronous to the accelerator master oscillator as required. This clock is then forwarded to the programmable logic fabric clocking network to support synchronous logic operation.

Packet forwarding and filtering is implemented in programmable logic using a pattern checking block. This block has configuration registers to set the desired patterns and enable or disable a fixed number of possible pattern entries. Downstream TDL broadcasting is performed in programmable logic with the highest priority to meet the deterministic timing requirement. Upstream link aggregation is performed in programmable logic as well using round robin arbiters to ensure equal sharing of upstream bandwidth between downstream devices.

Timing Data Generator

A typical TDG is configured as a CHP system equipped with two SFP+ daughtercards. This provides a total of 40 SFP+ ports for use with the TDL network. It is worth noting that other configurations are possible with only one or even no SFP+ daughtercards, the CHP carrier by itself could potentially function as a TDG. This allows for the use of other special function daughtercards in addition to or even in place of SFP+ daughtercards. A DTDG will have one connection to a UTDG eventually connecting to the GTDG at the highest level. The TDG will support the generation of periodic events such as 1 Hz, 10 Hz, 100 Hz, etc. Local events and data can be broadcast by DTDGs to lower level subtrees of the network or globally by the GTDG.

The Update Frame facilitates the synchronous operation of many different types of systems connected to the TDL. All devices are guaranteed to receive the Update Frames at the same time and can then act on the data contained within the frame at this known coordinated time interval. The specific implementation of the TDL network intended for use with EIC is shown in Figure 4. It includes the GTDG at the highest level with the ESR, HSR, and RCS machines segmented at the first level, which then fan out to the lower level systems distributed in different physical areas of the accelerator complex. Certain subsystems, such as the hadron injector complex, electron injection Linac, and Strong Hadron Cooling, will be segmented at the geographic level from one of the machine level systems.
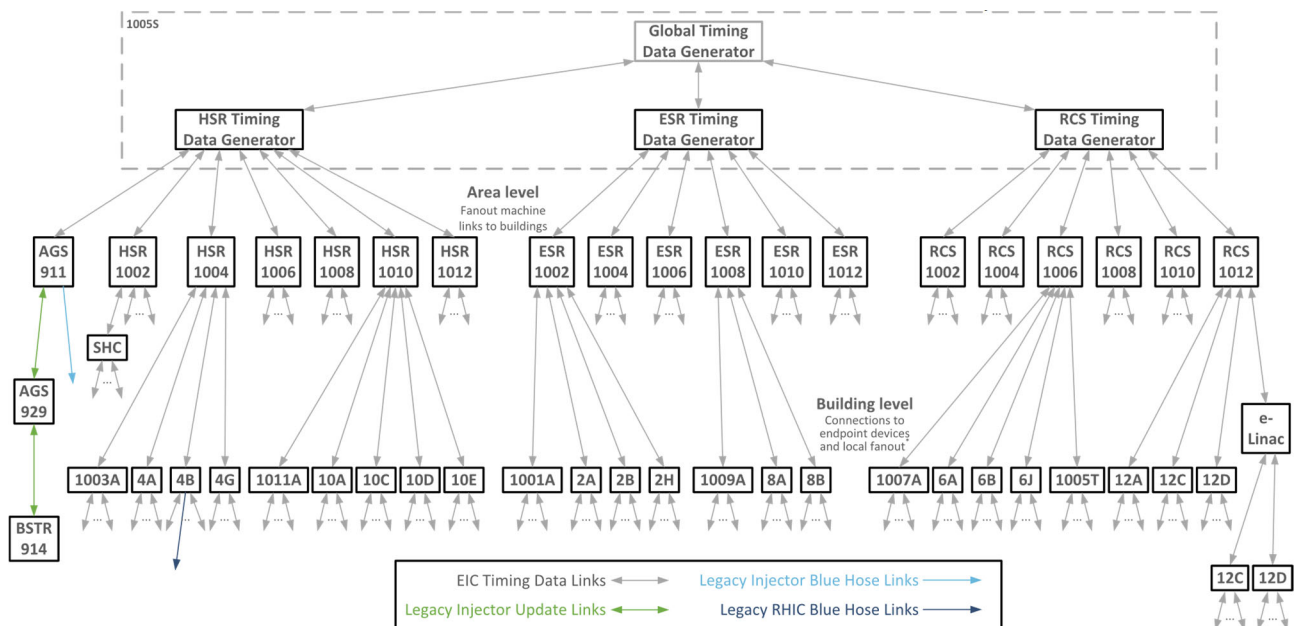

Figure 4: EIC TDL Network Topology.

CONCLUSION

The upgraded TDL for EIC will be implemented as a single networking protocol to consolidate the functions of several existing timing distribution links. The tree structured network will interconnect the various subsystems in the accelerator complex. It will reliably distribute timing information deterministically and with low latency. The TDL provides features such as event and data filtering and bidirectional data transfer between devices within the network. The TDG systems will be implemented primarily using CHP systems and SFP+ daughtercards for link distribution and aggregation. Xilinx Ultrascale+ devices will be used to implement the core functionality of the TDL within the CHP systems. The network topology very naturally lends itself to use within the EIC machine given its strict timing requirements and system hierarchy.

REFERENCES

[1] L.T. Hoff, "Real-Time Scheduling of Software Tasks," in *Proc. ICALEPCS'95*, Chicago, IL, 1995.

[2] T. Kerner, C. R. Conkling Jr, and B. Oerter, "V123 Beam Synchronous Encoder Module", in *Proc. PAC'99*, New York, NY, USA, Mar. 1999, paper MOP20, pp. 699-701.

[3] T. Hayes, F. Severino, and K. S. Smith, "A Deterministic, Gigabit Serial Timing, Synchronization and Data Link for the RHIC LLRF", in *Proc. PAC'11*, New York, NY, USA, Mar.-Apr. 2011, paper MOP282, pp. 642-644.

[4] ISO/IEC standard 7498-1:1994, http://standards.iso.org