CDF/DOC/TRIGGER/CDFR/????
Version 0.8
September 22, 2005

Pulsar Firmware for L2 trigger upgrade

S. Pitkanen, V.Rusu, C. Lin,B. Reisert, T. Liu

Contents

List of figures

List of tables

1 Introduction

This document describes the functionality and the implementation of the Pulsar firmware for the L2 Pulsar trigger system. All of the firmware, described in this document, has been tested with beam (except firmware for XFT). For each firmware, possible additional functionalities and optimizations of the implementation are listed. Some may be implemented at a later stage only if necessary. This document is still a work in progress.

1.1 L2 Pulsar trigger system overview

The L2 Pulsar trigger system was commisioned in two phases. In Phase 1 the system was limited by hardware used to sink data to a PC. Each data stream required it's own PCI card. In Phase 2 the hardware was replaced by newer improved version of the hardware. The new hardware can sink four data streams on one PCI card.

1.1.1 Phase 1

For the RunIIb Level-2 decision crate upgrade, the Pulsar boards are used as the universal interface to receive trigger data fragments from the upstream Level-1 trigger systems: Muon, XTRP, Global Level-1, ShowerMax, SVT, Cluster, and Isolated Cluster. The processed data are then converted to CERN S-LINK[1] format and merged into two streams to be sent to a commodity PC running the Level-2 trigger algorithm. The Pulsar crate receives the Level-2 decision back from the PC and forwards the decision to the Trigger Supervisor.

Figure 1 shows the baseline configuration for the RunIIb Level-2 decision crate. We use a total of six Pulsar boards to source-in the Level-1 trigger data. In addition, two Pulsar S-LINK Merger boards are used to combine the data into a single S-LINK packet. The Pulsar Muon board receives the 16 muon fiber data (from Matchbox and PreMatchbox) as well as the XTRP and L1 trigger bits (from FRED). The Pulsar Cluster board receives the L2 cluster (from LOCOS and CLIQUE), isolated cluster data (from ISOPICK and ISOCLIQUE), and the cluster sum ET bits (from FRED). The Cluster board replaces the functionality of the RunIIa CLIST and ISOLIST boards. The ShowerMax data from SMXR arrive at the Level-2 decision crate via 48 fibers. We use three Pulsar boards to receive the ShowerMax data (16 fibers per board). A forth Pulsar board is used to merge the output S-LINK data from the three Pulsar ShowerMax boards to a single stream. The outputs of the Muon, Cluster and ShowerMax S-LINK Merger boards are merged at the Global S-LINK Merger before being sent to the PC. Due to the long latency of the SVT data, the SVT tracks are sent directly to the Decision PC from the Pulsar SVT board. The L2toTS board is the last Pulsar board in the chain. It is responsible for negotiating the L2 decision between the PC and the Trigger Supervisor.

---

[1] SLINK is a specification for an industrial standard data-link. For more information on SLINK, see CERN SLINK web page [1].

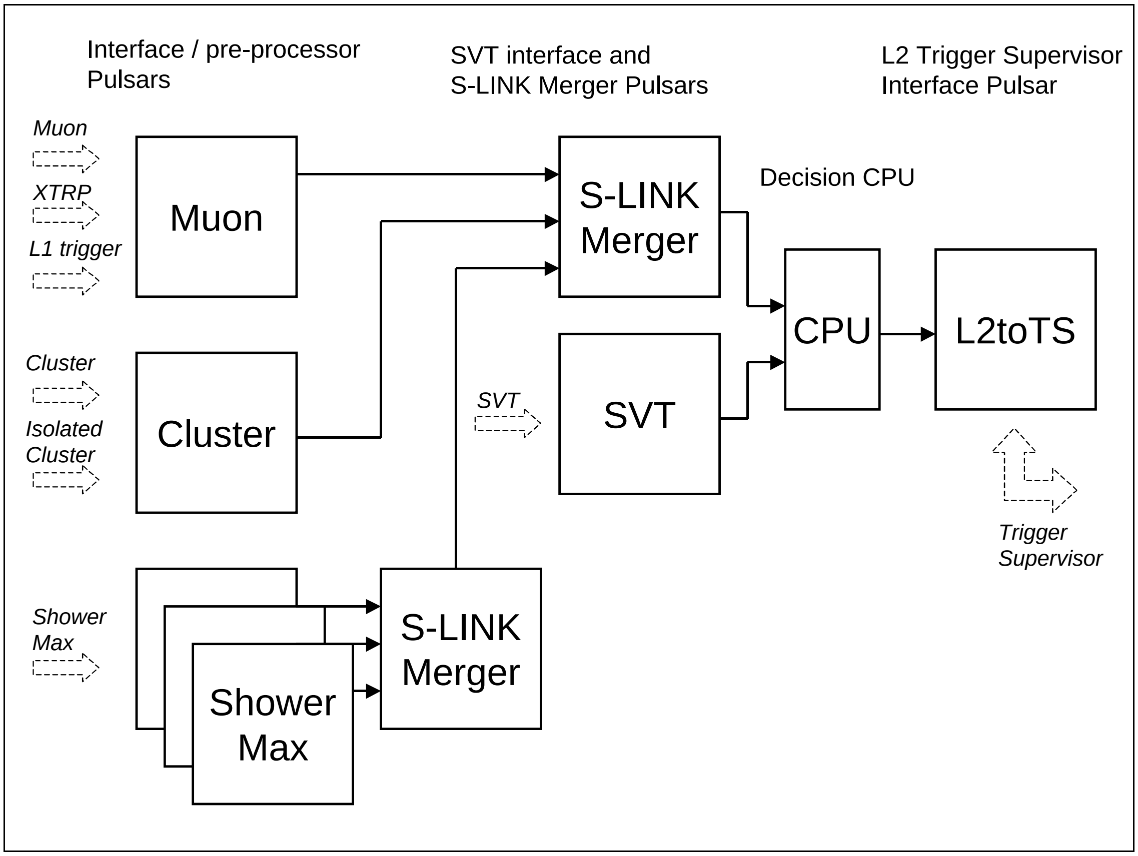

Figure 1: Phase 1 L2 Pulsar trigger system configuration block diagram

1.1.2 Phase 2

In the Phase 2 L2 trigger system the S-LINK cards used to sink data to the Decision CPU were replaced with a never model that could sink four S-LINK channels instead of just one. One of the S-LINK Merger boards could now be removed. Data path for XFT is planned, but not yet implemented.

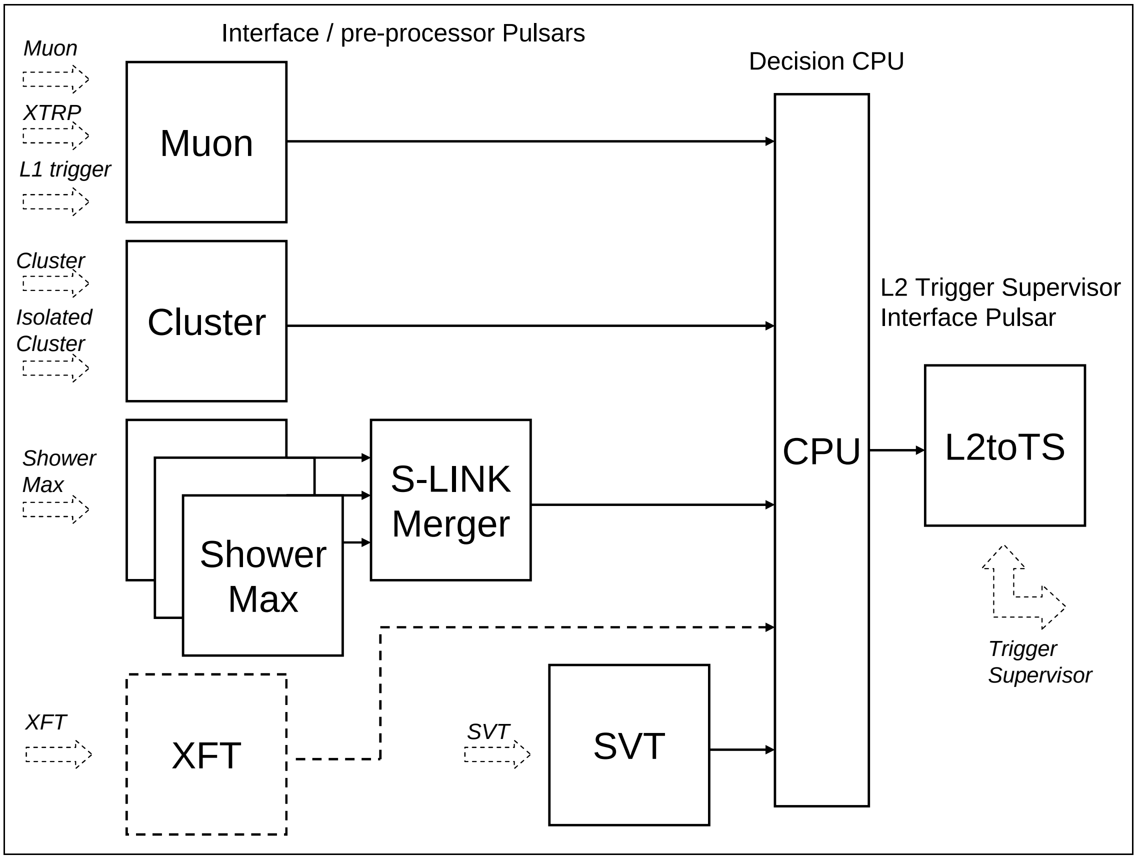

Figure 2: Phase 2 L2 Pulsar trigger system configuration block diagram

1.2 Pulsar board

Pulsar (**Puls**er **a**nd **R**ecorder) is a 9U VME board. Figure 3 shows a block diagram of the Pulsar board. The board is designed to be a multi-purpose interface board for the CDF Trigger system. There are dedicated connections on the Pulsar board for receiving data from the L1 global trigger (FRED/preFRED), XTRP, and SVT. In addition, custom and commodity mezzanine cards can be mounted on the four available Common Mezzanine Card (CMC)[2] slots to receive data from other sources (e.g. Hotlink fiber , Taxi fiber, and CERN S-LINK data). The Pulsar board also has a dedicated connection for negotiating the trigger decision with the Trigger Supervisor. On the output side, the Pulsar board uses the standard S-LINK protocol for transmitting data downstream via one of the two S-LINK channels on the P3 connector to either another Pulsar board or to a commodity PC.

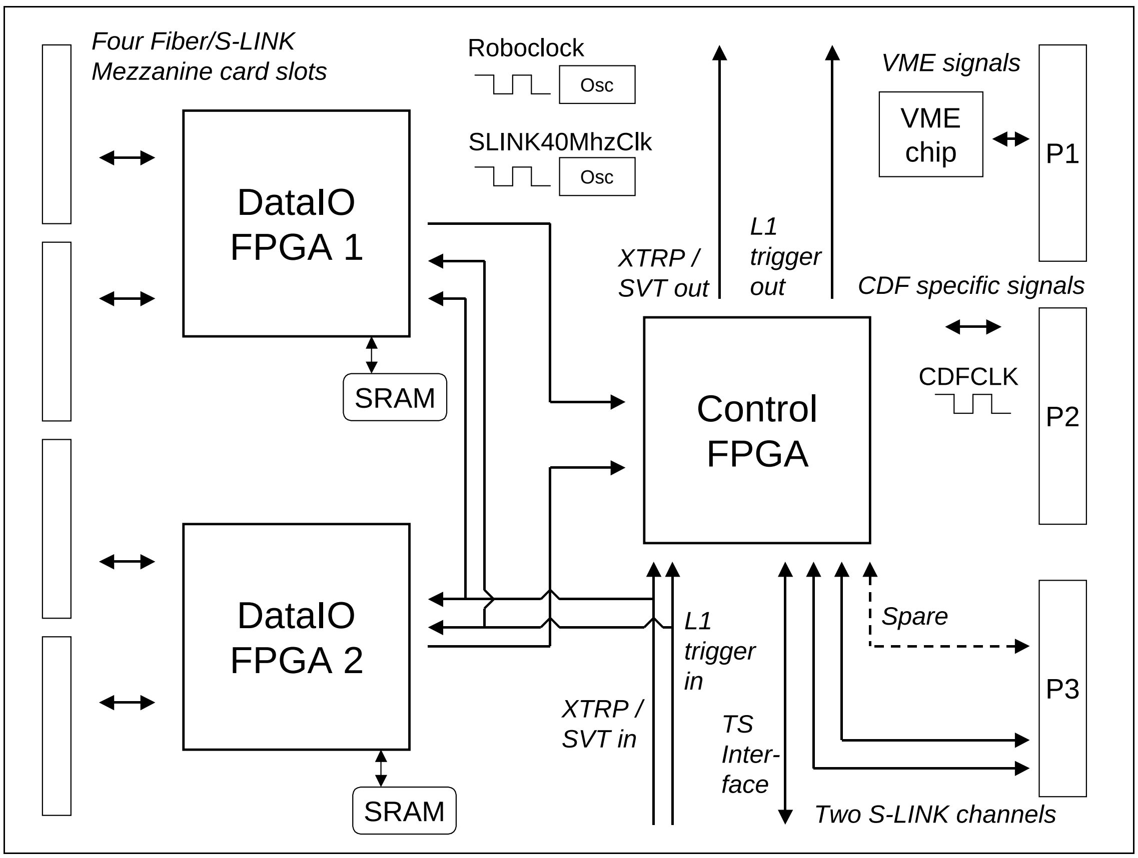

Figure 3: Pulsar board block diagram.

The Pulsar motherboard is dominated by three FPGAs: two DataIO FPGAs and one Control FPGA. Each DataIO FPGA is responsible for processing data for two mezzanine cards. The mezzanine cards can either be receivers or transmitters. For most applications, the Control FPGA is used to merge data from the two DataIO FPGAs. It is also where the merged S-LINK data is sent out of the Pulsar board via the P3 connector. There are

[2] CMC is an IEEE draft standard for a family of mezzanine cards designed to be used interchangeably on VME, VME64, VMC64X and CompactPCI cards. P1386/Draft 2.4a.

two independent S-LINK bi-directional channels on the P3 that can be used to send data downstream or source-in S-LINK data packages. The input signal from the XTRP/SVT and the L1 trigger bits are available to all three FPGAs. However, only the Control FPGA has the capability to transmit L1 trigger bits and XTRP/SVT signals (either real or simulated) out of the Pulsar board. For the Pulsar Trigger Supervisor interface, it is also the Control FPGA that carries out the handshaking with the Trigger Supervisor.

There are three clock signals available to all the FPGAs; two signals are generated from the onboard oscillators (RoboClock and SLINK40MhzClk) and the third signal is the standard CDFClock (132nSec), which comes from the VME backplane. Roboclock is used to run the internal logic inside the FPGAs. An 80 MHz oscillator is currently used to generate the signal. SLINK40MhzClk is used for receiving and transmitting S-LINK data. As the name suggests, a 40 MHz oscillator is used to generate the SLINK40MhzClk signal.

We have chosen Altera 20K400 APEX FPGA (652 pin BGA package) for both the DataIO and Control FPGA. In addition to the internal 26KB memory, each DataIO FPGA is connected to a 128K×36 SRAM (CY7C1350).

For more detailed information of the Pulsar board, see CDF note 6259 [2] and Pulsar project webpage [3].

1.2.1 AUX card

As mentioned in the previous section, the S-LINK data packet can be driven off or sent into the Pulsar board via the P3 connector. To implement this feature, a simple VME64X transition module (AUX card) needs to be mounted on the backside of the VME crate. The transition module has two CMC slots for mounting CERN's Link Source Cards (LSC) for transmitting SLINK data or Link Destination Cards (LDC) for receiving SLINK data. It is possible to mount one of each so the Pulsar board can simultaneously receive and transmit S-LINK data via P3.

Additionally, the custom AUX card has a bi-directional SVT connector and a TS connector (not mounted on the current AUX cards) to supplement the existing interface on the Pulsar motherboard. There are also two Lemo connectors on the AUX card to bring NIM type signals into the Pulsar board. The NIM inputs have been used on the Pulsar SVT board to measure the timing of the SVX system.

1.2.2 Mezzanine cards

One of the strengths of the Pulsar board is its ability to source-in an assortment of input data types. For each type of input data not supported by the Pulsar motherboard, simple custom mezzanine cards can be built and mounted on the CMC slots to bring the data into the DATAIO FPGAs. There are currently three types of mezzanine cards in use on the Pulsar boards: Hotlink, Taxi, and S-LINK mezzanine cards. Hotlink and Taxi mezzanine cards (transmitter and receiver versions) are custom made. The S-LINK cards are commercial cards designed at CERN. Both Hotlink and Taxi mezzanine cards have four channels per card. There is also a version of Hotlink card, where two of the Hotlink optical channels are replaced by an LVDS connector (for CLIQUE LVDS signal).

Hotlink receiver mezzanine cards are used to receive Level-1 Muon Matchbox and PreMatchbox data. The hybrid Hotlink LVDS receiver card is used for Calorimetry Cluster data. Taxi receiver cards are used to receive ShowerMax and Calorimetry Isolated Cluster data. Aside from the AUX card implementation, the commercial S-LINK LDC and LSC cards can also be mounted on the Pulsar motherboard for S-LINK transmission. The transmitter versions of the mezzanine cards are used to transmit simulated L1 trigger data. This feature is extremely useful in debugging the L2 system without having to depend on colliding beam data. We will not discuss this testing feature any further in this document. A detailed discussion on the transmitter firmware is described in document [4].

2 Firmware functionality and implementation

In the L2 Pulsar trigger system, the Pulsar boards are labeled based on the functionality. The different labels and the input sources are:

- Pulsar Muon board – Muon, XTRP, and L1 trigger bits,
- Pulsar Cluster board – Cluster and isolated cluster,
- Pulsar ShowerMax board - SMXR ShowerMax ,
- Pulsar SVT board – SVT,
- Pulsar XFT board –XFT,
- Pulsar SLINK Merger board – SLINK data packets,
- Pulsar L2toTS board – interface between decision PC and the Trigger Supervisor.

In this section, we describe the functionality and the implementation of the different firmware for the different types of Pulsar boards. First the functionality is described, followed by a description of the firmware implementation. Pulsar XFT board is not currently used in the L2 Pulsar trigger system, but it's use is planned in the future and the firmware for this board is already implemented, so it is also described in this document.

The Pulsar firmware is written in VHDL, the Very High Speed Integrated Circuit (VHSIC) Hardware Description Language. First, we will define some common terminologies. The term 'component' is used later on in this document. Components are natural abstract blocks of a firmware design. A whole system is modeled using a hierarchy of components. A component is normally modeled in a separate file for easy management.

Figure 3 shows a generic block diagram of a firmware for a Pulsar board. The different blocks in this diagram represent common components used in all the firmware for the L2 trigger system.

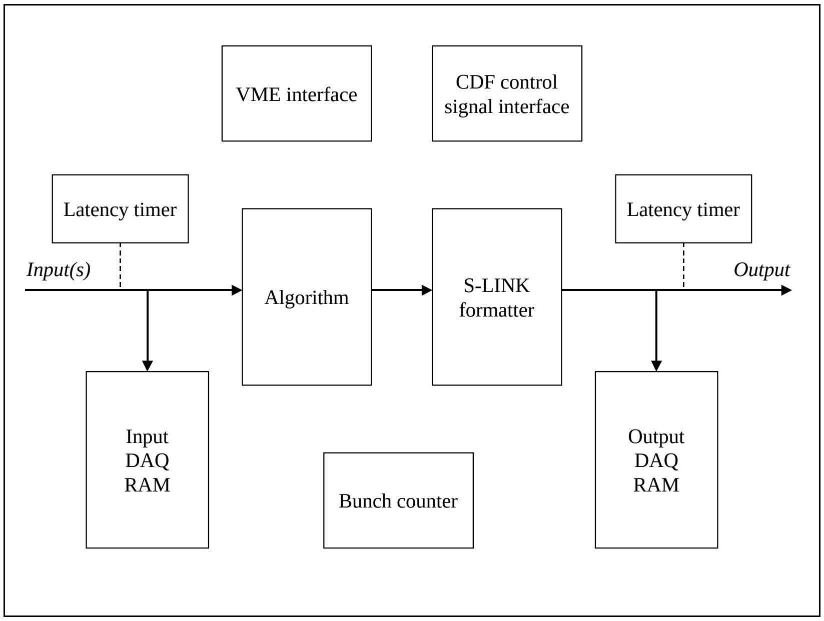

Figure 4: Generic Pulsar firmware block diagram

In the next section the functionality and the implementation of some common components is described. These common components are widely used in the firmware for the different types of Pulsar boards. Section 2.2 describes the functionality and the implementation of the firmware for the interface Pulsar boards. Sections 2.3 and 2.4 describe the functionality and the implementation of the firmware for the S-LINK Merger and L2toTS boards.

2.1 Common components

To reduce the overhead of firmware development, many components of the Pulsar firmware are written so that the same firmware could be used on different Pulsar interface boards. In this section, we will describe the functionality and the implementation of some common components used on the Pulsar boards.

2.1.1 VME interface

Functionality

A VME interface component handles the communication between each FPGA and to the VME chip on board. Through the VME interface the user can read and set control register values, read status register values, send pulse signals to the FPGAs and read DAQ RAMs.

Interface to VME bus

Signals from the VME bus to each FPGA on the Pulsar board go through the P1 and P2 back plane connectors and the VME interface chip. The VME interface chip takes care of the raw VME commands on the back plane and translates them into local commands for the three FPGAs. Each FPGA sees the 32-bit wide data bus, 21-bit (bits 23 to 2) address bus and a few control signals. Control signals instruct the FPGA when to strobe data and address and whether the VME command is a read or a write.

Implementation

The VME interface component implementation is identical for all DataIO and Control FPGA firmware. However, the definition of some control and status registers may be different in different firmware. Furthermore, the DAQ RAM sizes can be different, depending on the need of that particular Pulsar board.

Most of the functionality is built with three basic components. VMEReadWriteRegister, VMEReadWritePulseDecoder and VMEtribus. These components are existing firmware from CERN. They compare the address on the VME address bus to an address defined in the firmware. If the addresses match the components enable data between the VME data bus and registers, RAM’s or ROM’s.

Inside the VME interface there is a Power-up Reset component, which sends out a synchronous reset signal to all other components at power-up.

FPGA selection

VME address bits 18 and 19 are used to select with which FPGA the user wants to communicate with.

VME address bits 18	19	Selected FPGA
0	0	Control FPGA
1	0	DataIO FPGA 1
1	1	DataIO FPGA 2

Table 1: VME address bits for FPGA selection

Registers

All VME interface components have three status registers (read only), four control registers (read/write) and two pulse registers (write only).These registers are used differently in different firmware, but first two registers are reserved. First status register has a firmware version value in it. First pulse register is used to reset the FPGA. Third register is named DAQ SW version, but it can be used as a normal control register (read/write).

DataIO FPGA 1

Name	Address	Type
Firmware version	YY080000	R
Reset	YY080004	W
DAQ SW version	YY080008	R/W
Control register 1	YY08000C	R/W
Status register 1	YY080010	R
Pulse 1	YY080014	W
Control register 2	YY080018	R/W
Control register 3	YY08001C	R/W
Status register 2	YY080020	R

DataIO FPGA 2

Name	Address	Type
Firmware version	YY0C0000	R
Reset	YY0C0004	W
DAQ SW version	YY0C0008	R/W
Control register 1	YY0C000C	R/W
Status register 1	YY0C0010	R
Pulse 1	YY0C0014	W
Control register 2	YY0C0018	R/W
Control register 3	YY0C001C	R/W
Status register 2	YY0C0020	R

Control FPGA

Name	Address	Type
Firmware version	YY000000	R
Reset	YY000004	W
DAQ SW version	YY000008	R/W
Control register 1	YY00000C	R/W
Status register 1	YY000010	R
Pulse 1	YY000014	W
Control register 2	YY000018	R/W
Control register 3	YY00001C	R/W
Status register 2	YY000020	R

YY = VME address bits 31..24, not used by the firmware.

Table 2: VME interface registers

DAQ RAMs

Each FPGA has two DAQ RAM's. We have chosen this implementation so that the interface to the DAQ RAM's is identical to all Pulsars and to all FPGAs. However in some cases, none, or only one of them, is actually used. For example, in the Pulsar SVT board, the DAQ RAM's in the DATAIO FPGA's are always empty. The SVT board only uses the two DAQ RAM's in the Control FPGA.

We have labeled the first DAQ RAM on an FPGA as DAQ RAM 1 and the second as DAQ RAM 2. Each DAQ RAM is divided into four buffers corresponding to the four L2 DAQ buffers. To each buffer a word count register is associated, which tells the number of words in that buffer.

Each buffer can also be divided into subdivisions. For example one DAQ RAM can have data from eight different inputs. The use of the DAQ RAM's in different firmware is described in more detail later on in this document.

The VME interface component uses the VME address bits to determine which DAQ RAM is being read out and enables that DAQ RAM's output to the VME data bus. Also the word count registers are enabled to the VME data bus the same way.

When reading out DAQ RAM's the VME address bit 17 selects between the DAQ RAM's inside one FPGA.

VME address bit 17	Selected DAQ RAM
0	DAQ RAM 1
1	DAQ RAM 2

Table 3: VME address bit for DAQ RAM selection

When reading out DAQ RAM's VME address bit 23 is set high, bit 22 is set low and bits 20 and 21 select which buffer is read out. This follows CDF speciation for DAQ readout defined in CDF note 2388 [5].

VME address bit				Selected buffer
23	22	20	21	
1	0	0	0	Buffer 0
1	0	0	1	Buffer 1
1	0	1	0	Buffer 2
1	0	1	1	Buffer 3

Table 4: VME address bits for DAQ RAM buffer selection

VME addresses for the word count registers are listed in Table 5 below. Some firmware have more than one word count register per DAQ RAM. For these, look at a more detailed listing of used VME addresses in Appendix 1.

DataIO FPGA 1

DAQ RAM	Buffer #	Address
1	0	YY080800
1	1	YY080900
1	2	YY080A00
1	3	YY080B00
2	0	YY080804
2	1	YY080904
2	2	YY080A04
2	3	YY080B04

DataIO FPGA 2

DAQ RAM	Buffer #	Address
1	0	YY0C0800
1	1	YY0C0900
1	2	YY0C0A00
1	3	YY0C0B00
2	0	YY0C0804
2	1	YY0C0904
2	2	YY0C0A04
2	3	YY0C0B04

Control FPGA

DAQ RAM	Buffer #	Address
1	0	YY000800
1	1	YY000900
1	2	YY000A00
1	3	YY000B00
2	0	YY000804
2	1	YY000904
2	2	YY000A04
2	3	YY000B04

YY = VME address bits 31..24, not used by the firmware.

Table 5: Word count register VME addresses

In the beginning of each buffer on the DAQ RAM 1 in the Control FPGA, there is a DAQ Header Word. The format follows CDF note 2388 [5]. Currently Pulsar firmware does not provide Geographical Address in the DAQ Header Word.

Bit	Description
0..7	Bunch Counter Value
8..12	Geographical Address
13..22	Board Serial Number
23..31	Board Type

Table 6: DAQ Header Word format

IDPROM

All Control FPGA's have a read only memory that contains the values of an IDPROM. IDPROM format is also defined in CDF note 2388 [5].

For all Control FPGA firmware an IDPROM is included in the VME interface. The IDPROM read only memory values are in a memory input file (.mif), which is taken in into compilation when compiling the firmware.

Board type

Each Pulsar board has been assigned with its own Board type. The Board type value is put into the IDPROM, DAQ header word and in the S-LINK data stream as part of a header word. Board type is used to distinguish data from different Pulsar boards in the trigger system.

Board type	**Description**
081	L2 Pulsar Muon/XTRP Rx IIa
083	L2 Pulsar SVT Road Warrior
085	L2 Pulsar Muon/XTRP/L1 Tx or SVT XTRP-emu
086	L2 Pulsar Muon/XTRP/L1 Rx IIb
087	L2 Pulsar SHOWERMAX Tx
088	L2 Pulsar SHOWERMAX Rx
089	L2 Pulsar Cluster/PreFred Tx
090	L2 Pulsar Cluster/PreFred Rx
091	L2 Pulsar SVT Tx
092	L2 Pulsar SVT Rx
093	L2 Pulsar Merger Tx
094	L2 Pulsar Merger Rx
095	L2 Pulsar L2TS Tx
096	L2 Pulsar L2TS
097	L2 Pulsar L1 Scaler
098	L2 Pulsar SVT TF
099	L2 Pulsar test one
100	L2 Pulsar test two
101	L2 Pulsar Stereo Tx
102	L2 Pulsar Stereo Rx
103	SVT Pulsar Hit Buffer

Table 7: Pulsar Board types

Possible future improvements

- Add a read only memory which has a list of addresses of all the registers for the firmware and a description how to use them (This could be used as a kind of a user guide for the board). *

* **Not required for operation.**

2.1.2 Bunch counter

To conform to the CDF standard, we have implemented bunch counter functionality on all the Pulsar boards.

Functionality

The bunch counter component counts the number of CDFCLK clock cycles (132ns) since last B0. The value of the counter is saved on L1A. The values, for events to all four buffers, are saved until the next event occurs for the same buffer.

The Bunch counter value is stamped in the DAQ header word for DAQ readout, and into the S-LINK header word when sending S-LINK formatted data out from the Pulsar board.

Implementation

The counting is done for four buffers independently. There are four different counters and four registers. Upon B0 signal, the counters are reset to zero. When L1A occurs, the value of the counter from one of the four counters is latched into a register. At any time, any one of the four registers can be read.

In the firmware, the bunch counter value can be offset by a value between zero and 256. This offset dictates when the counter is reset after B0. If the offset value is zero, the counter is reset when B0 occurs. Increase the offset by one unit corresponds to a delay of one CDFCLK tick. This offset is used to synchronize the bunch counter value on the Pulsar boards with the rest of CDF. One of the Control registers in the VME interface can be used to set this value. For all L2 trigger system Pulsar boards, Control register 1 is used to set this offset value. By default the value is set to 41.

2.1.3 Latency timer

To be able to study the Level-2 system-wide timing performance, we have implemented a latency timer firmware on the Pulsar boards. The firmware allows us to measure the arrival time of the data packet at different stages in the Pulsar trigger system.

Functionality

The Latency timer starts counting up with a user defined clock when L1A occurs. Any signal, with a Buffer number, inside the FPGA can be used to stop the latency timer. Both the timing information from the current as well as the previous event (unbiased w.r.t L2) are saved. The Latency counter value can be put into the S-LINK header word when sending S-LINK formatted data out from a Pulsar board or it can be put into a diagnostic DAQ RAM at the end of the data.

Implementation

Latency measuring is done independently for the four buffers. The Latency timer has four different counters and eight registers. The counters run on the user defined clock. The Latency timer component remembers the latency value of the previous event, by having two registers for each buffer. When a new value is saved, the previous value is moved to the other register.

On the output of the Latency timer component the counter value is shifted left by two bits, so the resulting accuracy is the user defined clock frequency multiplied by four. If the SLINK40MhzClk clock is used for the timing, then an increase of one on the output value corresponds to 0.1us.

Usage

The Latency timer is used on almost every input and output of a Pulsar board in the L2 Pulsar trigger system. It's also used to measure upstream SVT data timing and to measure Trigger Supervisor handshaking overhead. The current usage of Latency timers in the L2 Pulsar trigger system is show in Figure 5.

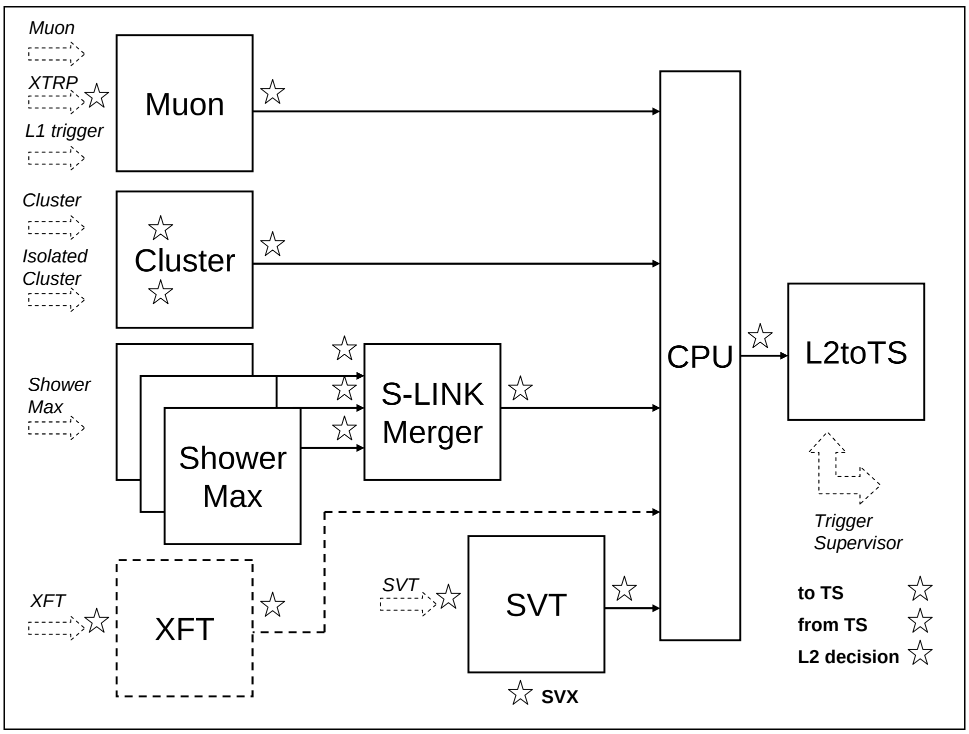

Figure 5: Current usage of Latency timers in the L2 Pulsar trigger system

2.1.4 HRR handling

Functionality

The Pulsar board receives CDF control signals, CDFHalt, CDFRecover, and CDFRun, through the P2 back plane connector. All CDF signals are broadcast to the Control FPGA and to both DataIO FPGA's.

On CDFHalt, all data receiving on all FPGA's is halted. On CDFRecover a reset signal is broadcast inside each of the three FPGA's. On CDFRun the data receiving on the FPGA's is enabled again.

2.1.5 S-LINK formatter

Functionality

Data between the Pulsar boards and to and from the Decision CPU, in the L2 Pulsar trigger system, is sent and received using the S-LINK mezzanine cards. The S-LINK formatter is used to format data to a Pulsar S-LINK format and to send the data out trough an S-LINK output interface. The Pulsar S-LINK format follows the S-LINK specification, and in addition capsulates the data with L2 Pulsar trigger system specific header and trailer words.

Implementation

The S-LINK formatter takes in 32-bit words and encapsulates the data with Pulsar S-LINK header and trailer words and with S-LINK control words. The Pulsar S-LINK header and trailer format is described in Table 8 below.

31…24	23…20	19…18	17…16	15…10	9…2	1…0
Format version	Data source	Region ID	Reserved		Bunch count	Buffer number
Latency value (previous event)				Latency value (current event)		
Data						
Data size				Error flags		

Table 8: Pulsar S-LINK format

The first header word has bits reserved for: Format version, Data source, Region ID, Bunch counter value and Buffer number. Format version is used to recognize different Pulsar S-LINK formats from each other. Currently only one Pulsar S-LINK format exists. Data source and Region ID are used to define the source of the data. Each Pulsar in the L2 trigger system has been assigned with its own Data source. See Appendix 2 for Pulsar Data source values. Region ID can be used to specify a source more accurately on one Pulsar board. On every Pulsar board the Bunch counter value and Buffer number are saved into the outgoing data stream. When data from different Pulsars is merged, this information can be used to make sure that all of the data is from the same event.

In the second header word latency values are saved into the outgoing data stream. Any desired latency value from the board can be put into the second header word. In most cases the latency value from L1A to first word going out is used.

The trailer word has the Data size of the event and Error flags for the event. Error flags are not currently used in the firmware. This feature will soon be added to all firmware.

Data is sent out as long as there is more data on the input of the S-LINK formatter. If there is gaps in the data, the S-LINK formatter pauses sending the data. When End of Event word is received, the S-LINK formatter stops sending data and ends the transmission with the trailer word and with an S-LINK end control word.

The S-LINK formatter has the ability to use flow control. It can stop sending data if the link is full or down. Currently in the L2 Pulsar trigger system the flow control ability is not used.

2.2 Interface Pulsar boards

Interface Pulsar boards are used to interface with different data paths coming to the L2 Pulsar trigger system. These Pulsars receive and record the data they get and in some cases pre-process the data before it goes to the Decision PC.
There are currently four different interface Pulsar boards: Muon, Cluster, ShowerMax and SVT. Their individual functionalities and the firmware implementation are described below.
Another data path, for XFT, to L2 Pulsar trigger system is currently under development. The firmware is already implemented and is also described in this chapter.

2.2.1 Muon

The Muon interface board receives data from Muon, XTRP and L1 trigger interfaces. Muon and XTRP data is recorded into DAQ RAM's. Muon data is zero suppressed and merged with XTRP and L1 trigger data. The merged data is sent out in the Pulsar S-LINK format.

Functionality

Muon data comes into the board from 16 Hotlink fibers. Muon input interface is through the Hotlink receiver mezzanine cards on the Pulsar board. One Hotlink receiver mezzanine card sinks data from four Hotlink fibers. With four mezzanine cards, the board can sink data from all 16 Muon Hotlink fibers. Two mezzanine cards are connected to each DataIO FPGA. Muon input data is saved into DAQ RAM 1 on the DataIO FPGA's. Muon data is zero suppressed on the DataIO FPGA's, and sent to the Control FPGA.

XTRP data is received on the Control FPGA through a SVT input interface. The SVT input interface uses an external FIFO for the incoming data. This FIFO is read by the Control FPGA. XTRP input data is saved into DAQ RAM 2 on the Control FPGA.

L1 trigger data is also received by the Control FPGA. Data comes to the Control FPGA through an LVDS cable connection on the board.

The data from Muon, XTRP and L1 trigger is merged on the Control FPGA and sent out from the P3 connector. The data is sent out in the Pulsar S-LINK format. Outgoing data has the Buffer number, seen by the Muon interface board for the current event, and the Bunch counter value, counted by the board, in the S-LINK header word. Also Data source value is set accordingly to the S-LINK header word (See Appendix 2). The outgoing data is saved into DAQ RAM 1 on the Control FPGA.

From P3 the S-LINK formatted data goes out through the AUX card and the S-LINK LSC mezzanine card.

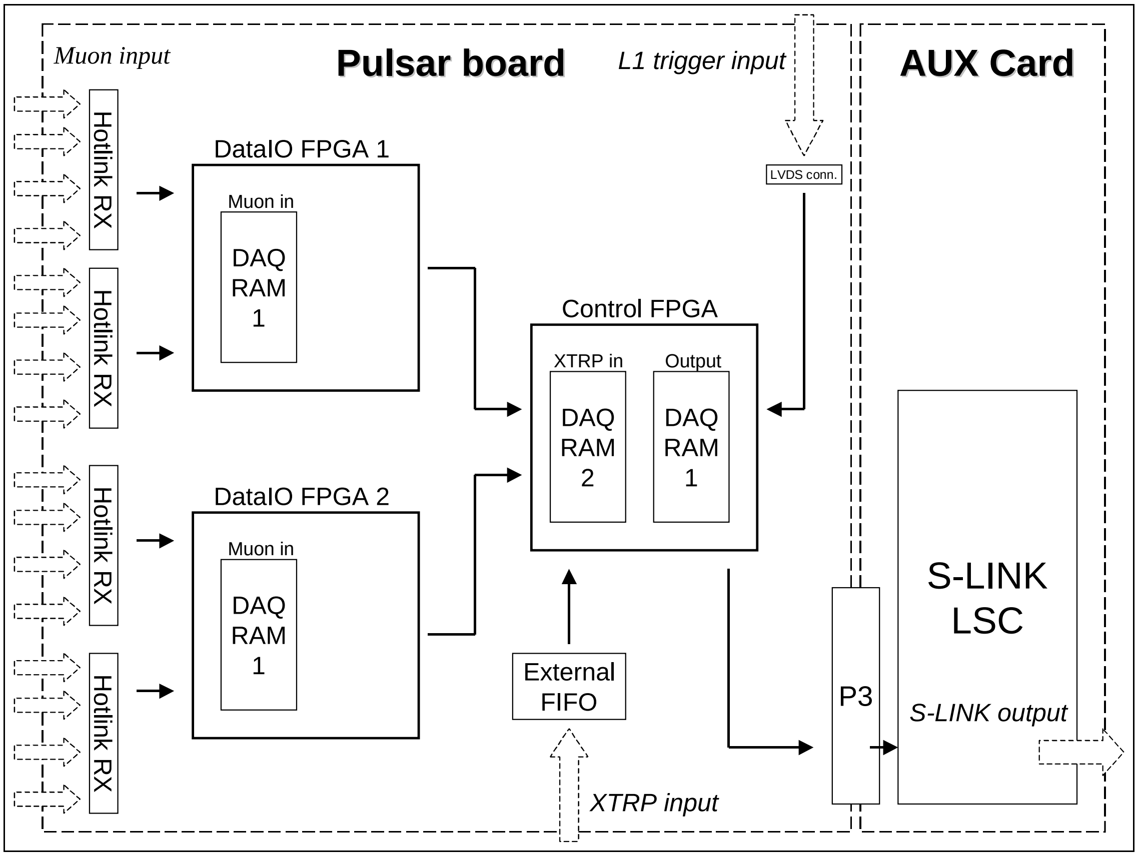

Figure 6: Muon interface board functionality block diagram

Implementation

Firmware on the DataIO FPGA's receive and pre-process the Muon data. Firmware on the Control FPGA receives the Muon data from the DataIO FPGA's. It also receives XTRP and L1 trigger data, merges all of the data together and sends the data out from the S-LINK output in the Pulsar S-LINK format.

DataIO FPGA firmware

The DataIO firmware for both of the DataIO FPGA's of the Muon interface board is identical. The DataIO FPGA's receive Muon data from two Hotlink receiver mezzanine cards. Each mezzanine card has four fiber channel inputs. The two DataIO FPGA's receive a total of 16 Hotlink channels.

Muon input

A 512 words deep 8-bit FIFO is used to receive the Muon data from one channel. With Statemachine 1 (See Appendix 3) and four registers, 32-bit words are formed from the incoming 8-bit words. The 32-bit words are written into a Middle FIFO, which is 128 words deep. A L1A FIFO is used to save the Buffer number for the incoming events. The L1A FIFO is strobed on each L1A. Statemachine 1 controls a counter, which counts the number of incoming words. For an event the Muon input receives a constant number of words. The counter and a comparator is used to create a End of Event mark, which is saved together with the data. Also the Buffer number from the L1A FIFO is saved with the data. Statemachine 2 (See Appendix 3) reads the Middle FIFO and writes the data into a input DAQ RAM. It also controls a write address counter which output is used as part of the write address for the input DAQ RAM. Buffer number controls the upper two bits of the write address. This way the RAM is divided into four buffers. The eight input DAQ RAM's on one DataIO FPGA are mapped to the VME interface so that they look like one DAQ RAM; DAQ RAM 1. First 32 words in the DAQ RAM 1 are from the first input (first channel); next 32 words are from the second input, and so on. Input DAQ RAM's are 128 words deep (32 / buffer).

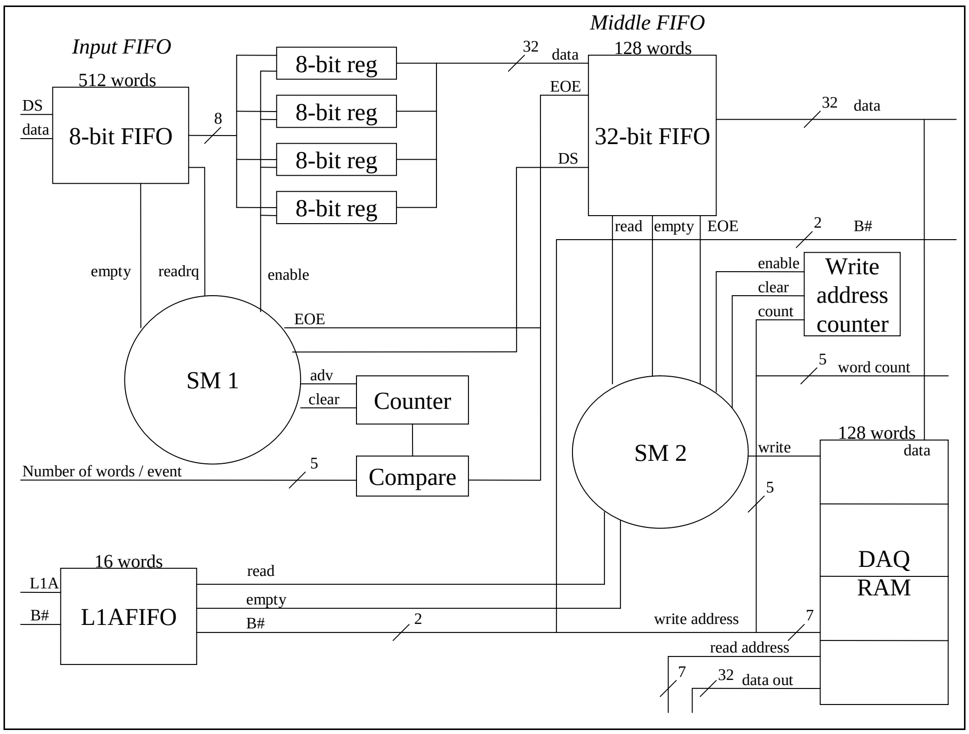

Figure 7: Muon DataIO FPGA input firmware block diagram

Matchbox channels

First three channels on each mezzanine card receive Muon Matchbox data. Data from these channels, for one event, is 124 bytes. Four bytes form a 32-bit Muon word. From the 124 bytes, 31 32-bit words are formed. The last 32-bit word marks the End of Event. In this case the number of words per event constant is set to 31.

Not all 32-bits of Muon data are needed for decision making. 24-bits of the 32-bit words are used. Only the first 18 data words and the end of event word contain meaningful data for the decision making. To any of the first 18 words, which have nonzero data content, 10 bits of address information is added and the words are put into a 34-bit output FIFO. Also the end of event word is put into the output FIFO.

Pre-Matchbox channels

The last channel on each mezzanine card receives Muon Matchbox data. Data from these channels, for one event, is 68 bytes. Four bytes form a 32-bit Muon word. From the 68 bytes, 17 32-bit words are formed. The last 32-bit word marks the End of Event. In this case the number of words per event constant is set to 17.

Like in the case of Matchbox channels, not all 32-bits of Muon data are needed for decision making. But in case of Pre-Matchbox channels all the 17 word per an event contain meaningful data for the decision making. To any of the 17 words, which have nonzero data content, 10 bits of address information is added and the words are put into a 34-bit output FIFO. Also the end of event word is put into the output FIFO.

Formatted Muon word	**Description**
34	EOE
33…32	Buffer number
31	Mezzanine card
29…30	Fiber
28…24	Word count
23…0	Muon data

Table 9: Formatted Muon data word

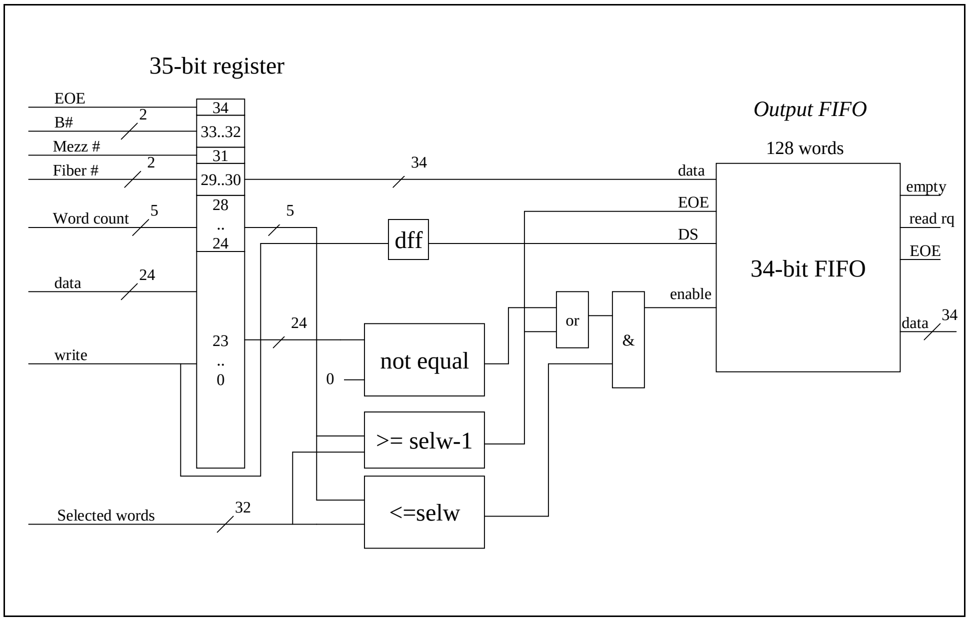

Figure 8: Muon DataIO FPGA zero suppression

Instances of the same input component were used for each of the eight input channels on the DataIO FPGA's to receive and pre-process the Muon data. The output FIFO's for each channel are read by Statemachine 3 (See Appendix 3), which merges the data and sends the data out to the Control FPGA.

Control FPGA firmware

The Control FPGA firmware receives data from the L1 trigger interface, from the XTRP interface and Muon data from the two DataIO FPGA's. The data is merged and send out in the Pulsar S-LINK format.

L1 trigger data comes to the Pulsar board from two 34-line LVDS cables. 64-bits from the total of 68-bits are for data, two bits are used for buffer number and two bits for data strobes (one for each cable). For an event one word from each cable is saved into an input FIFO. Only the data bits are currently saved, buffer number bits are not used.

XTRP data comes to the board first into a 23-bit external FIFO. This external FIFO is read by the Control FPGA firmware and the data is saved into a 23-bit internal FIFO. Bit 22 identifies the last word of an event from the data (XTRP End of Event word). The DAQ RAM 2 on the Control FPGA is used for the incoming SVT data. The upper bits of the 32-bit DAQ RAM are set to zero. The RAM is 2048 words deep (512/buffer).

The zero suppressed Muon data is received from the DataIO FPGA's into two input FIFO's. Each 34-bit word is reformatted into two 23-bit words.

First 23-bit word	Description
22	EOE
21	EOP
20	FPGA
19	Mezzanine card
17…18	Channel
12…16	Word count
0…11	Muon data (0…11)

Second 23-bit word	Description
22	EOE
21	EOP
20	FPGA
19	Mezzanine card
17…18	Channel
12…16	Word count
0…11	Muon data (12…23)

Table 10: Muon output data format

Bit 21, the End of Packet bit is always set high for the Muon data. It is used to distinguish between Muon and XTRP data.

The L1 trigger data, XTRP data and Muon data from the DataIO FPGA's is merged and saved into an output FIFO. 32-bit words are created from the 23-bit Muon words and 24-bit XTRP words, by setting the extra bits to zero.

L1 trigger data bits 0..31
L1 trigger data bits 32..63
XTRP data
Muon data
XTRP EOE word

Table 11: Muon interface board output data format

All the data is merged in to a Output FIFO. The S-LINK formatter starts reading the Output FIFO when ever there is more data. The data is sent out from the S-LINK output in the Pulsar S-LINK format, using the S-LINK formatter component. S-LINK formatted output data is saved into DAQ RAM 1, which is 2048 words deep (512/buffer).

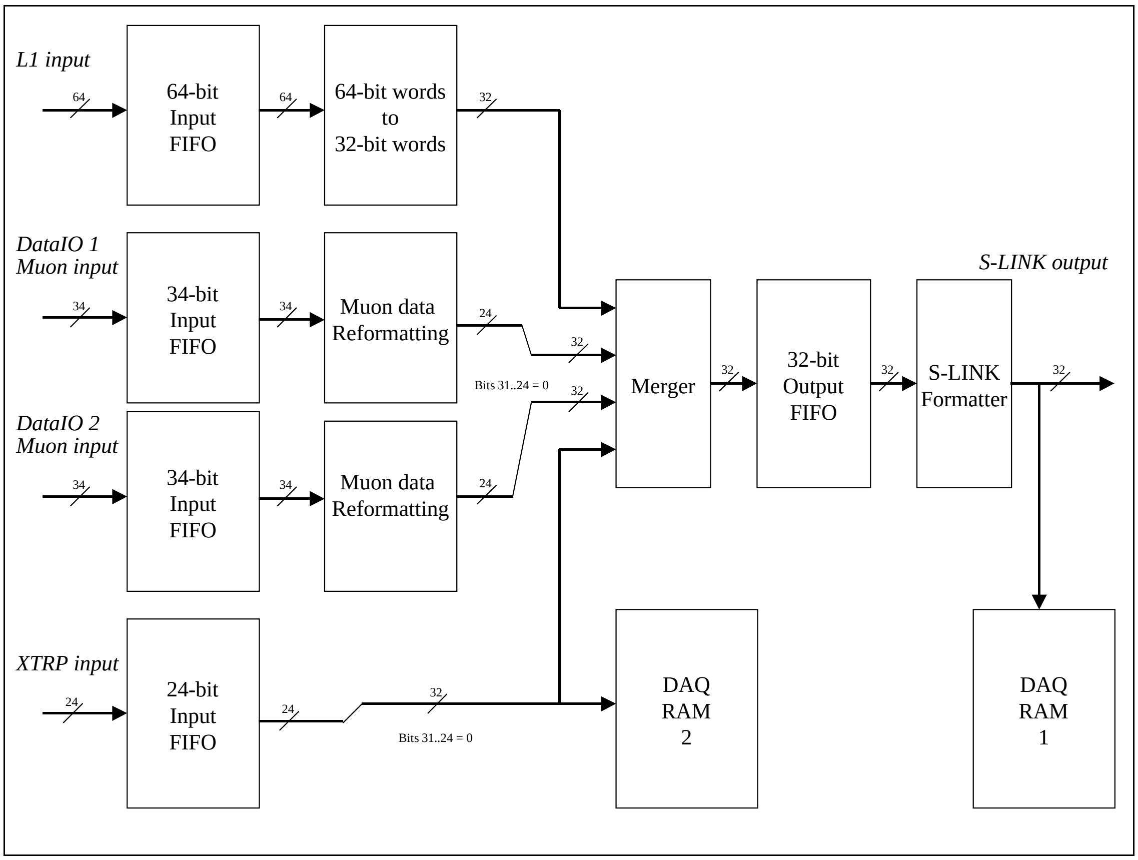

Figure 9: Muon Control FPGA firmware block diagram

Latency timing is measured for the XTRP data arriving on the Control FPGA input and when the XTRP data for one event is completely received. These latency timing values are stored after the XTRP data at the end of DAQ RAM 2 for all four buffers.

RAM and FIFO sizes

DataIO FPGA's

One Muon channel (eight / DataIO FPGA):

- Muon input FIFO: 512 x 8-bits
- Muon middle FIFO: 128 x 32-bits
- Muon output FIFO: 128 x 34-bits
- Muon input DAQ RAM: 128 x 32-bits

Control FPGA

Muon:

- Muon DataIO 1 input FIFO: 512 x 34-bits
- Muon DataIO 2 input FIFO: 512 x 34-bits

XTRP:

- XTRP input FIFO: 512 x 23-bits
- XTRP input DAQ RAM: 2048 x 32-bits

L1:

- L1 input FIFO: 16 x 64-bits

Output:

- Output FIFO: 1024 x 32-bits
- Output DAQ RAM: 2048 x 32-bits

Possible future improvements

DataIO FPGA firmware

- Add latency measurement for the Muon inputs.

Control FPGA firmware

- Add latency measurement for the L1 trigger data.

2.2.2 Cluster

The Cluster interface board receives data from the L2 calorimeter boards (LOCOS, ISOPICK, CLIQUE, ISOCLIQUE) and PreFred trigger interface. The data is recorded into DAQ RAM's. One DataIO FPGA is used to process and record the cluster data while the other is used for iso-cluster data. The data is then sent to the CONTROL FPGA for merging. The PreFred EtSum data is appended here as part of the Slink package. The merged data is sent out in the Pulsar S-LINK format.

Functionality

Cluster data comes into the board through 13 optical fibers and 1 LVDS cable. Six Hotlink fibers come from six LOCOS boards in the L2CAL crates. They carry energy and position information for the cluster. The end of event comes from the CLIQUE board through one LVDS cable. Iso-cluster data (the five energy sums and iso-cluster position) comes through six Taxi fibers from 6 ISOPICK boards. The end of event mark, iso-cluster position and other global information (buffer number, pass number) comes from the ISOCLIQUE board through an additional Taxi fiber.

The information for one cluster is contained in the six cluster fragments. Every cluster fragments has 48 bits of data which is received on the DataIO FPGA 2 and saved in the DAQ RAM 1 while being passed to the algorithm block through a FIFO. The algorithm block then takes the data and performs the following operations:

1. Sums up the HAD and EM energy from the cluster fragments
2. Computes the total number of towers in the cluster
3. Translate the cluster position from local to global coordinates

As for the cluster case, there are six iso-cluster fragments to make up an iso-cluster. The fragment has 88-bits of data which is received on the DataIO1 FPGA and saved into the DAQ RAM 1. Similar with the clusters, there is a FIFO which temporarily holds the data before being passed to the algorithm block where the following operations are performed:

1. Adds up the five Et sums
2. Translate the cluster position from local to global coordinates

In both cases, the latency is measured after the algorithm block operations are completed and before the data is packed into the Pulsar Slink format and sent to the Control FPGA

PreFred EtSum data is also received by the Control FPGA. Data comes to the Control FPGA through an LVDS cable connection on the Pulsar board.

The data from Cluster, Iso-Cluster and PreFred EtSum is merged on the Control FPGA and sent out from the P3 connector. The data is sent out in the Pulsar S-LINK format. Outgoing data has the Buffer number, seen by the Cluster interface board for the current event, and the Bunch counter value, counted by the board, in the S-LINK header word. Also Data source value is set accordingly to the S-LINK header word (See Appendix 2). The outgoing data is saved into DAQ RAM 1 on the Control FPGA.

From P3 the S-LINK formatted data goes out through the AUX card and the S-LINK LSC mezzanine card.

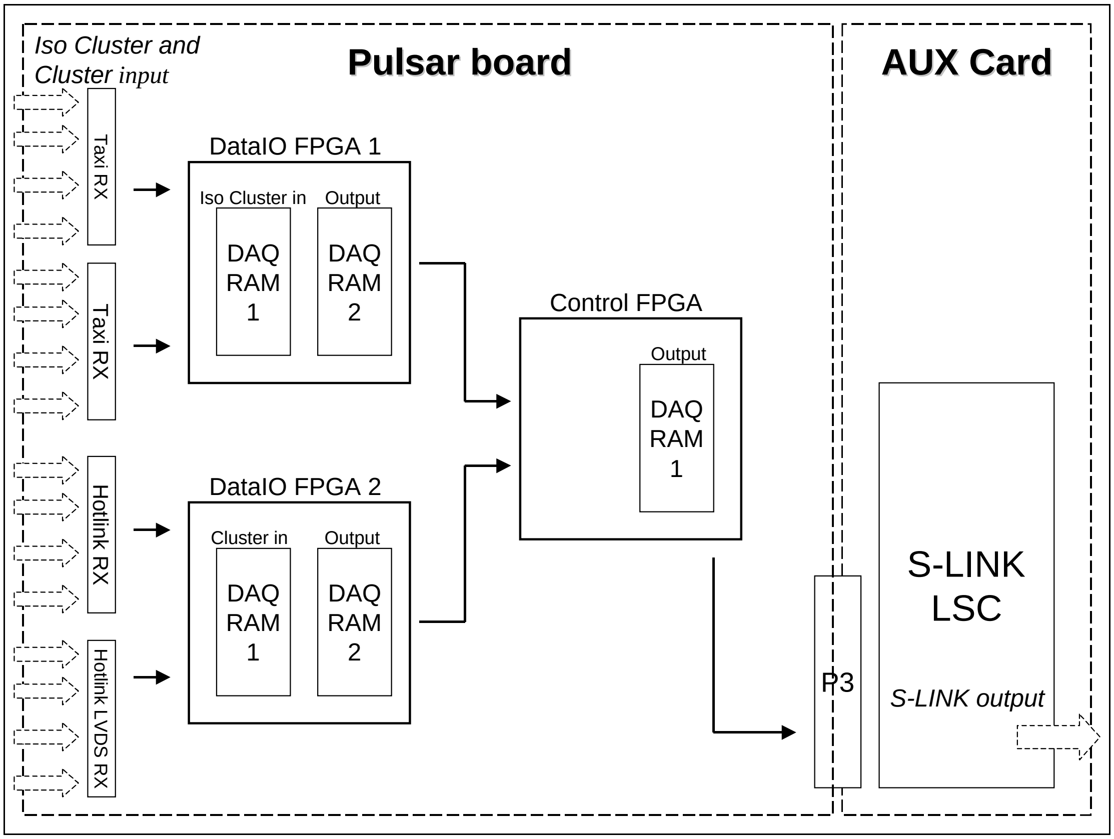

Figure 10: Cluster interface board functionality block diagram

Implementation

Firmware on the DataIO FPGA's receive and pre-process the Cluster and Iso-Cluster data. Firmware on the Control FPGA receives this data from the DataIO FPGA's. It also receives PreFred EtSum data, merges all of it together and sends the data out from the S-LINK output in the Pulsar S-LINK format.

DataIO FPGA firmware

Cluster data from one Hotlink channel and for one cluster fragment is six bytes. From the six bytes two 32-bit words are created. These 32-bit words are saved into the input DAQ RAM. The data is then latched into a 50 bit wide FIFO. The bit 48 of this FIFO is eventually marked as end of event. Similarly, bit 49 is marked as empty event if needed. After all the FIFOs have data, meaning that a whole cluster is present for processing, the algorithm block will read the 6 FIFOs and perform the operations detailed above. A 32-bit wide FIFO holds the data at the output of the algorithm block for S-LINK formatting. A cluster worth of information is contained in two 32-bit words in this FIFO.

Iso-cluster data from one TAXI channel and for one iso-cluster fragment is 11 8-bit words. From these, three 32-bit words are created and saved into the input DAQ RAM. The data is latched into a 96-bit wide FIFO for the algorithm block. In addition, for each iso-cluster, there are also two 8-bit transmissions from ISOCLIQUE. This data is logged into one 32-bit word to be saved into the input DAQ RAM and presented to the inputs of the algorithm block. The input DAQ RAM is 256 words deep (64/buffer). The operations described above are performed on the data and each iso-cluster is packed into four 32-bit words which are latched into a FIFO for further Pulsar S-LINK formatting.

The first word for each input DAQ RAM is the number of words present in that RAM which can be used to determine the number of clusters/iso-clusters in the event. The S-LINK formatted data from both FPGAs is logged into DAQ RAM 2 on each DataIO FPGA.

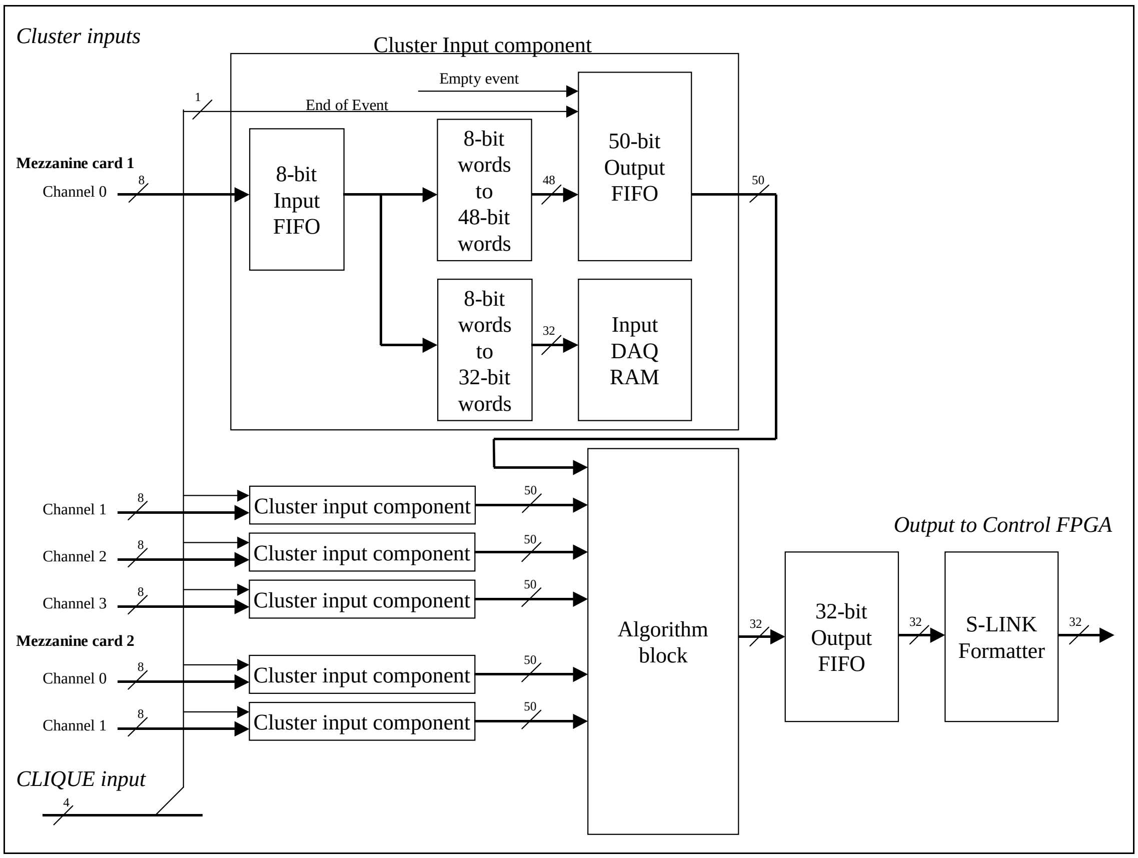

Figure 11: Cluster DataIO FPGA firmware block diagram

The eight input DAQ RAM's on one DataIO FPGA are mapped to the VME interface so that they look like just one DAQ RAM; DAQ RAM 1. First 2 words in the DAQ RAM 1 are from the first input (first channel); next 32 words are from the second input, and so on.

Control FPGA firmware

The Control FPGA firmware receives the Cluster and Iso-cluster data from the two DataIO FPGA's, receives data from the PreFred EtSum, merges the data and sends the data out in the Pulsar S-LINK format.

PreFred EtSum data comes to the Pulsar board from one 34-line LVDS cable. 32-bits from the total of 34-bits are for data and one bit for data strobe. For an event one word from each cable is saved into an input FIFO.
Cluster/Iso-Cluster data from the DataIO FPGA and PreFred EtSum data are merged and saved into an output FIFO.

PreFred EtSum 0..31
Iso-Cluster data
Cluster data

Table 12: Cluster interface board output data format

The S-LINK formatter starts reading the Output FIFO when ever there is more data. The data is sent out from the S-LINK output in the Pulsar S-LINK format, using the S-LINK formatter component. S-LINK formatted output data is saved into DAQ RAM 1, which is 2048 words deep (512/buffer).

RAM and FIFO sizes

DataIO FPGA

ISOLIST:

- Input FIFO: 512 x 8-bits
- ISOPICK FIFO: 8 x 88-bits x 6
- ISOCLIQUE FIFO: 8 x 16-bits
- Output FIFO: 1024 x 32-bits
- Input DAQ RAM: 256 x 32-bits
- Output DAQ RAM: ???

CLIST:

- Input FIFO: 512 x 8-bits
- CLIST FIFO 128 x 50-bits
- Input DAQ RAM: 256 x 32-bits
- Output FIFO: 1024 x 32-bits
- Output DAQ RAM: ???

Control FPGA

- L1 Input FIFO: 16 x 32-bits
- DataIO 1 Input FIFO: 512 x 32-bits
- DataIO 2 Input FIFO: 512 x 32-bits
- Output FIFO 2048 x 32-bits
- Output DAQ RAM: 2048 x 32-bits

2.2.3 ShowerMax

The ShowerMax interface boards receive the data from sent from the Level-1 SMXR boards. ShowerMax data is recorded into DAQ RAM's. One ShowerMax interface board receives one third of the ShowerMax data. In the L2 Pulsar Trigger System, three ShowerMax interface boards are used to receive all of the ShowerMax data. To merge the data from the three ShowerMax boards, a forth Pulsar board, S-LINK Merger board is used.

Functionality

On the ShowerMax interface boards the data is zerosuppressed, merged into one package and sent out in the Pulsar S-LINK format.

ShowerMax data comes into the board from 16 Taxi fibers. ShowerMax input interface is through the Taxi receiver mezzanine cards on the Pulsar board. One Taxi receiver mezzanine card can sink data from four Taxi fibers. With four mezzanine cards, the board can sink data from 16 ShowerMax fibers. Two mezzanine cards are connected to each DataIO FPGA. ShowerMax input data is saved into DAQ RAM 1 on the DataIO FPGA's. Data received by the DataIO FPGA's is merged and sent to the Control FPGA.

On the Control FPGA the data from the DataIO FPGA's is zero suppressed, merged and sent out from the P3 connector. Outgoing data has the Buffer number, seen by the ShowerMax interface board for the current event, and the Bunch counter value, counted by the board, in the S-LINK header word. Also Data source value is set accordingly to the S-LINK header word (See Appendix 2). The outgoing data is saved into DAQ RAM 1 on the Control FPGA.

From P3 the S-LINK formatted data goes out through the AUX card and the S-LINK LSC mezzanine card.

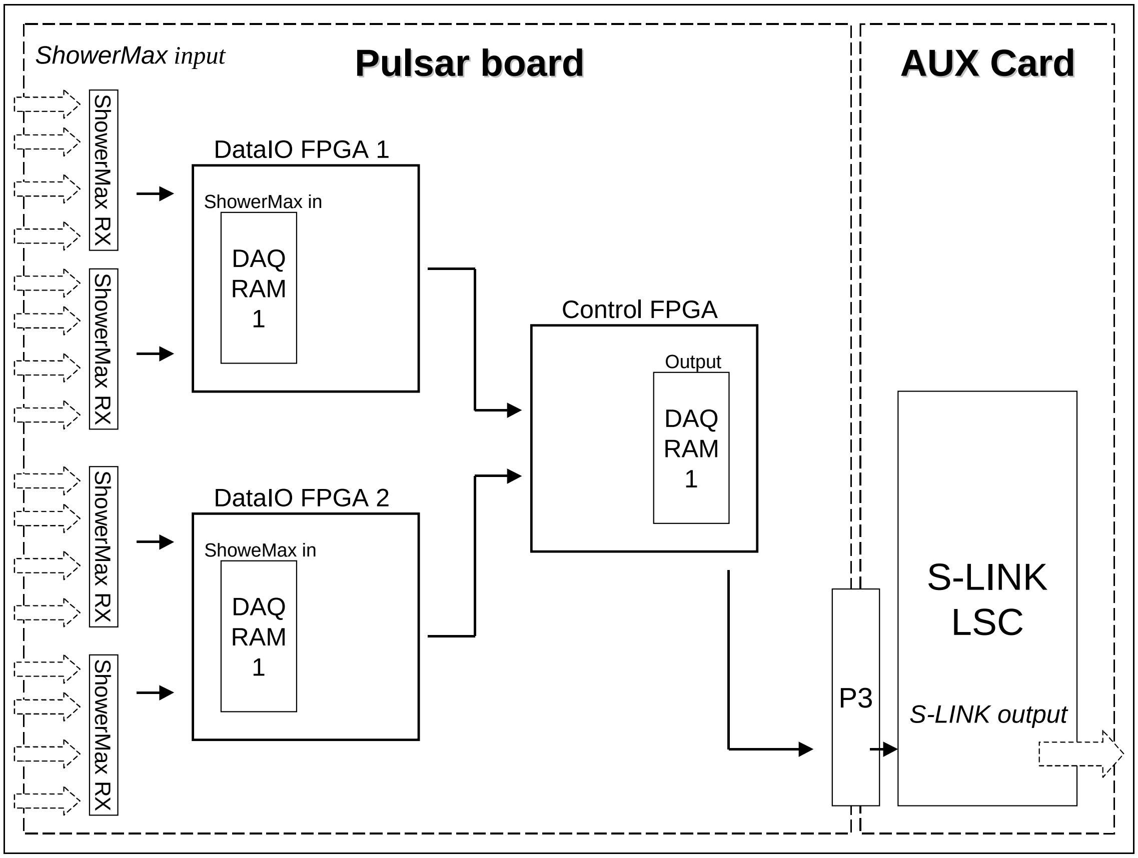

Figure 12: ShowerMax interface board functionality

Implementation

Firmware on the DataIO FPGA's receive the ShowerMax data, merge the data and send the data to the Control FPGA. Firmware on the Control FPGA receives the data from the two DataIO FPGA's, zerosuppresses the data, merges the data together and sends the data out from the S-LINK output in the Pulsar S-LINK format.

DataIO FPGA firmware

The DataIO firmware for both DataIO FPGA's is identical. The DataIO FPGA's receive ShowerMax data from two Taxi receiver mezzanine cards. Each mezzanine card has four fiber inputs. The two DataIO FPGA's receive a total of 16 Taxi channels.

A protection was created to prevent data corruption caused by glitches on the Taxi datastrobe line. Before the Taxi datastrobe signal is used to strobe data into a input FIFO, the signal is synchronized to the local SLINK40MhzClk. To protect against glitches the strobe signal is expected to be atleast two SLINK40MhzClk ticks long (50ns).

A 128 words deep 8-bit FIFO is used to receive the ShowerMax data from one channel. ShowerMax data of from one Taxi channel and for one event is five bytes. Two 32-bit words are created from the five bytes. The first 32-bit word has the data from the last four bytes. In the second 32-bit word the first byte is from the first word, and rest of the bits are set to zero. Statemachine 1 (See Appendix 3) reads the input FIFO and creates the

32-bit words, with 8-bit registers. Statemachine 1 writes the 32-bit words into a 32 words deep middle FIFO. Statemachine 2 (See Appendix 3) reads the middle FIFO and writes the two 32-bit words into an input DAQ RAM, which is 8 words deep. A L1A FIFO is used to save the Buffer number for the incoming events. The L1A FIFO is strobed on each L1A. Statemachine 2 controls a write address counter which output is used as part of the write address for the input DAQ RAM. Buffer number controls the upper two bits of the write address. This way the RAM is divided into four buffers. Statemachine 2 also writes the first 32-bit into a 32 words deep output FIFO.

First 32-bit word	**ShowerMax word**
31…24	5th
23…16	4th
15…8	3rd
0…7	2nd

Second 32-bit word	**ShowerMax word**
31…24	-
23…16	-
15…8	-
0…7	1st

Table 13: 32-bit ShowerMax word format

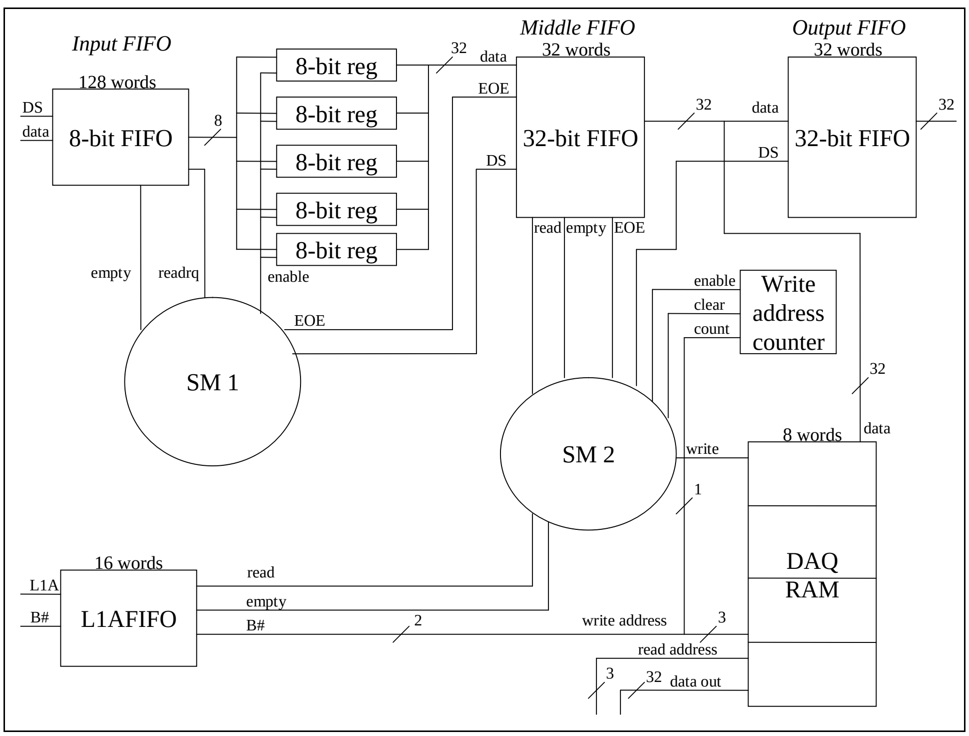

Figure 13: ShowerMax DataIO FPGA input firmware block diagram

Instances of the same input component were used for each of the eight input channels on the DataIO FPGA's to receive the ShowerMax data.

The output FIFO's for each channel are read by Statemachine 3 (See Appendix 3), which merges the data and sends the data out to the Control FPGA. Data from the first input component (first channel on the first mezzanine card) is sent out first, data from the second input component next, and so on.

The eight input DAQ RAM's from the input components on one DataIO FPGA are mapped to the VME interface so that they look like one DAQ RAM; DAQ RAM 1. First two words in the DAQ RAM 1 are from the first input component; next two words are from the second input component, and so on.

Control FPGA firmware

The Control FPGA firmware receives the ShowerMax data from the two DataIO FPGA's. From each of the 16 input channels one 32-bit word is received. Each non-zero word is saved in a 2048 words deep output FIFO. An additional 32-word is added at the end of the data to indicate which of the data words were nonzero. Lower sixteen bits from the 32-bits are used. The lowest bit indicates if the data from DataIO FPGA 1, first mezzanine card and first channel is zero on nonzero. If the bit is high the word is nonzero, and if the bit is low the word is zero. Bit 16 on the last word indicates if the data from DataIO FPGA 2, last mezzanine card and last channel is zero on nonzero. Higher 16-bits are not used. They are set to zero. See Figure 14 below.

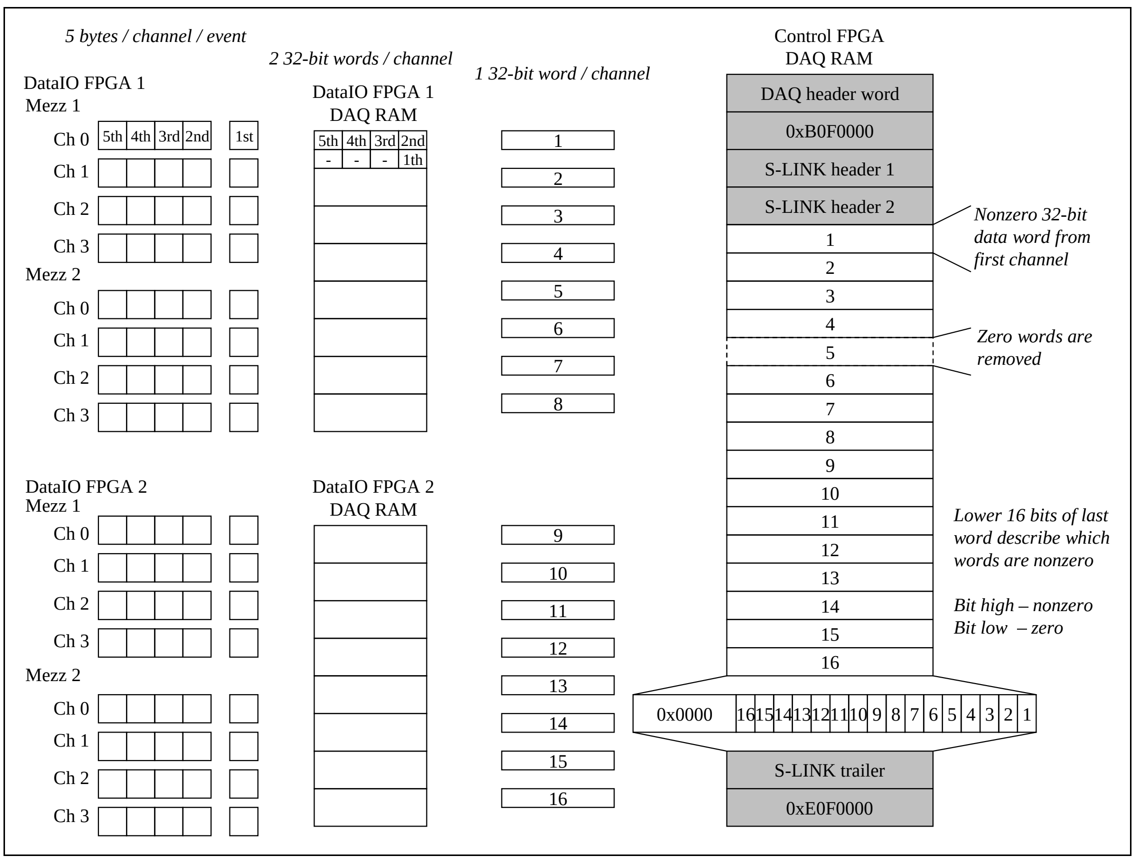

Figure 14: ShowerMax zero suppression

The S-LINK formatter starts reading the Output FIFO when ever there is more data. The data is sent out from the S-LINK output in the Pulsar S-LINK format, using the S-LINK formatter component. S-LINK formatted output data is saved into DAQ RAM 1, which is 2048 words deep (512/buffer).

RAM and FIFO sizes

DataIO FPGA's

One ShowerMax channel (eight / DataIO FPGA):

- Input FIFO: 128 x 8-bits
- Middle FIFO: 32 x 32-bits
- Output FIFO: 32 x 32-bits
- Input DAQ RAM: 8 x 32-bits

Control FPGA

- Output FIFO: 2048 x 32-bits
- Output DAQ RAM: 2048 x 32-bits

Possible future improvements

DataIO FPGA firmware

- Add latency measurement for ShowerMax inputs.

2.2.4 SVT

The SVT interface board receives data from the SVT input interface, pre-processes the data, and sends the data out in the Pulsar S-LINK format.

Functionality

Only the Control FPGA is used on the SVT interface board. SVT data is received through a SVT input interface. The SVT input interface has an external FIFO for the incoming data, which is read by the Control FPGA. SVT input data is saved into DAQ RAM 2.

Received SVT data is reformatted and sent out in the Pulsar S-LINK format. Outgoing data has the Buffer number, seen by the SVT interface board for the current event, and the Bunch counter value, counted by the board, in the S-LINK header word. Also Data source value is set accordingly to the S-LINK header word (see Appendix 2). The outgoing data is saved into DAQ RAM 1.

Input latency from L1A to when the SVT data starts arriving and has completely arrived is measured. Also latency from L1A to when SVX data starts arriving and has completely arrived is measured. Signal for the SVX latency measurements comes from one of the NIM connectors on the AUX card.

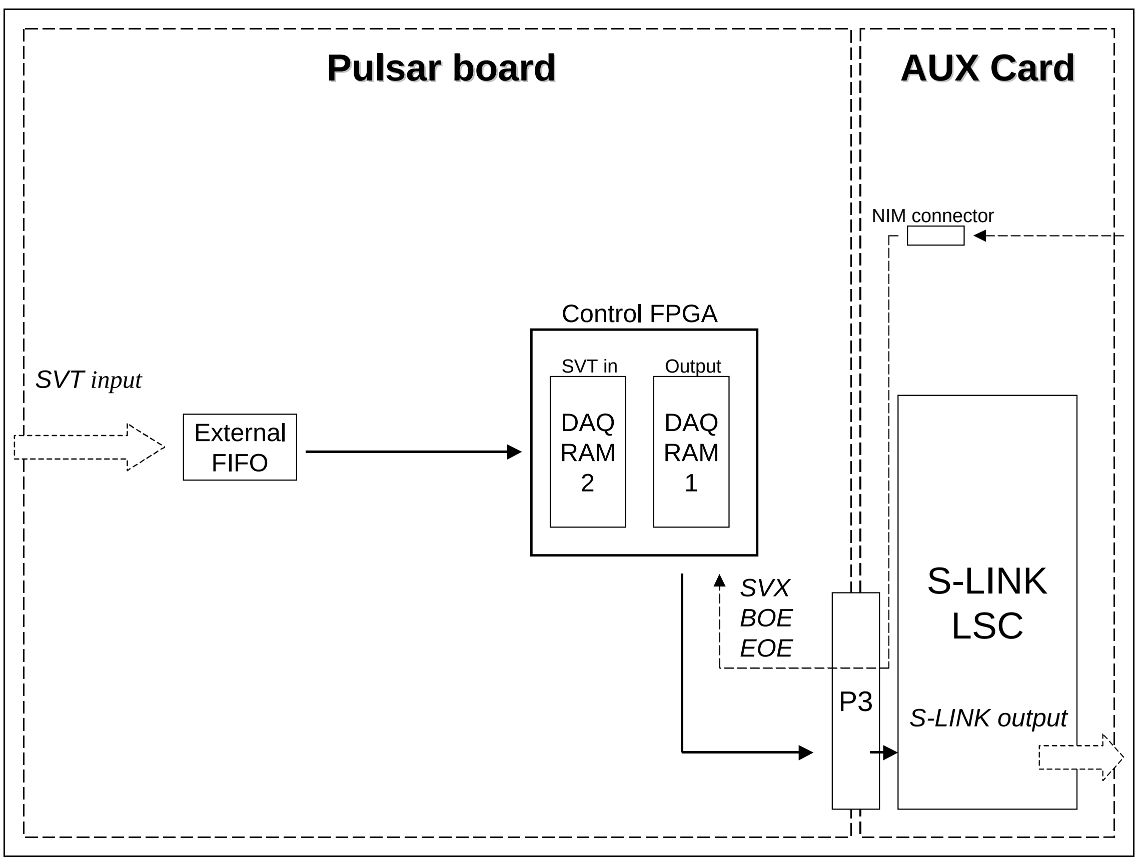

Figure 15: SVT interface board functionality

Implementation

The Control FPGA firmware receives the SVT data, pre-processes the data, and sends the data out from an S-LINK output in Pulsar S-LINK format.

Control FPGA firmware

SVT data comes to the Pulsar board first into a 23-bit external FIFO. A statemachine (See Appendix 3) reads the external FIFO and writes the data into a 512 words deep middle FIFO and to an input DAQ RAM. Bit 22 identifies the last word of an event from the data (SVT End of Event word). A L1A FIFO is used to save the Buffer number for the incoming events. The L1A FIFO is strobed on each L1A. The statemachine controls a write address counter which output is used as part of the write address for the input DAQ RAM. Buffer number controls the upper two bits of the write address. This way the RAM is divided into four buffers. DAQ RAM 2 is used for the incoming SVT data. The RAM is 2048 words deep (512/buffer). The upper bits of the 32-bit words in the DAQ RAM are set to zero.

Latency timing is measured for the SVT data arriving on the Control FPGA input and when the SVT data for one event is completely received. These latency timing values are stored after the SVT data at the end of the input DAQ RAM for all four buffers.

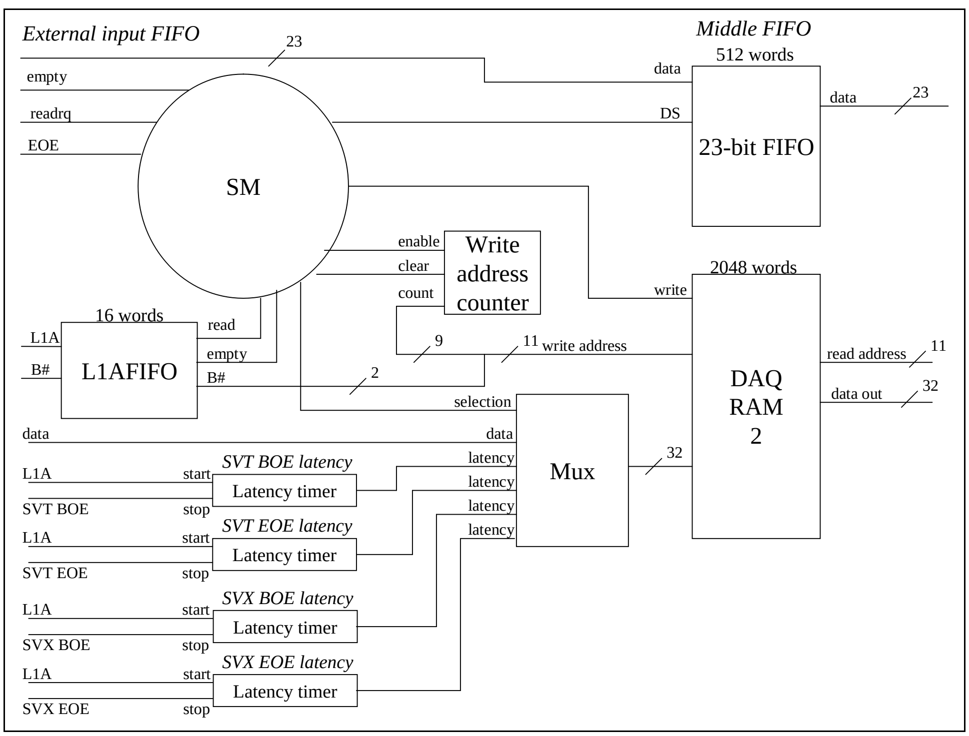

Figure 16: SVT Control FPGA input firmware block diagram

SVT data is in segments of seven 23-bit words. One segment is called a track. An event can have different number of tracks. At the end of an event there is an extra End of Event word. From each track two 32-bit words are created. These 32-bit words and the End of Event word are saved into an output FIFO.

1st 32-bit word	SVT word
31…29	-
28…19	2nd word (0…9)
18…0	1st word (0…18)

2nd 32-bit word	SVT word
31…30	-
29	7th word (20)
28..20	7th word (8…0)
19…9	6th word (20…10)
8…0	2nd word (18…10)

Table 14: 32-bit SVT word format

The S-LINK formatter starts reading the Output FIFO when ever there is more data. The data is sent out from the S-LINK output in the Pulsar S-LINK format, using the S-LINK formatter component. S-LINK formatted output data is saved into DAQ RAM 1, which is 2048 words deep (512/buffer).

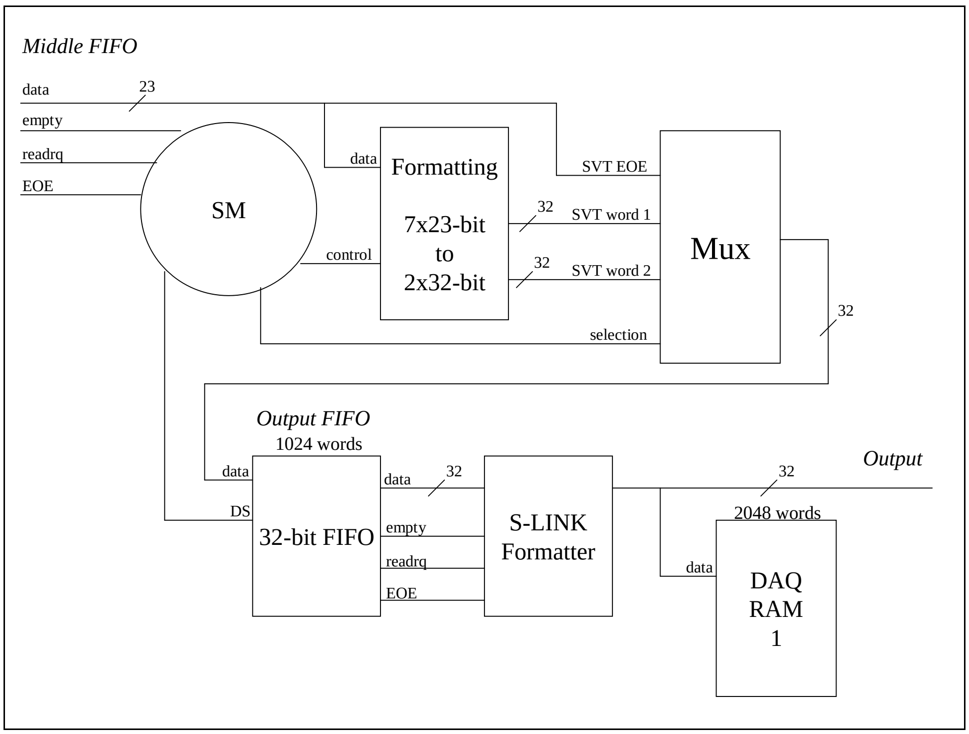

Figure 17: SVT Control FPGA output firmware block diagram

RAM and FIFO sizes

Control FPGA

Input:

- Middle FIFO: 512 x 23-bits
- Input DAQ RAM: 2048 x 32-bits

Output:

- Output FIFO: 2048 x 32-bits

- Output DAQ RAM: 2048 x 32-bits

2.2.5 XFT

The XFT interface board receives the data from sent from the L1 XFT Finder boards. The data is recorded into DAQ RAM's, merged into one package and sent out in the Pulsar S-LINK format.

Functionality

XFT data comes into the board from 12 fibers. XFT input interface is through the XFT receiver mezzanine cards on the Pulsar board. One XFT receiver mezzanine card sinks data from three fibers. With four mezzanine cards, the board can sink data from 12 XFT fibers. Two mezzanine cards are connected to each DataIO FPGA. XFT input data is saved into DAQ RAM 1 on the DataIO FPGA's. Data received by the DataIO FPGA's is merged and sent to the Control FPGA.

On the Control FPGA the data from the DataIO FPGA's is merged again and sent out from the P3 connector. Outgoing data has the Buffer number, seen by the XFT interface board for the current event, and the Bunch counter value, counted by the board, in the S-LINK header word. Also Data source value is set accordingly to the S-LINK header word (See Appendix 2). The outgoing data is saved into DAQ RAM 1 on the Control FPGA.

From P3 the S-LINK formatted data goes out through the AUX card and the S-LINK LSC mezzanine card.

Input latency from L1A to the XFT data arriving on an input is measured for each input.

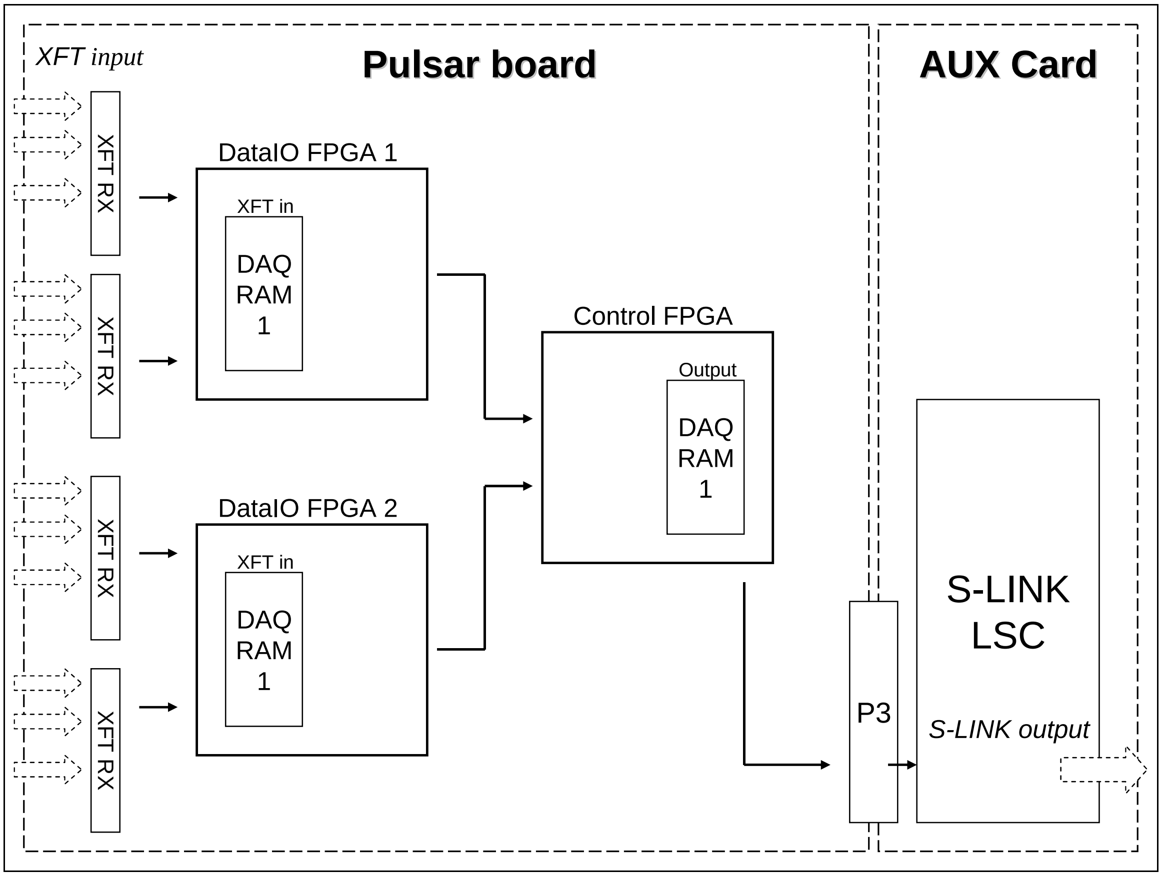

Figure 18: XFT interface board functionality

Implementation

Firmware on the DataIO FPGA's receive the XFT data, merge the data and send the data to the Control FPGA. Firmware on the Control FPGA receives the data from the two DataIO FPGA's, merges the data together and sends the data out from the S-LINK output in the Pulsar S-LINK format.

DataIO FPGA firmware

The DataIO firmware for both DataIO FPGA's is identical. The DataIO FPGA's receive XFT data from two XFT receiver mezzanine cards. XFT receiver mezzanine cards have four fiber inputs. Three out of the four fiber inputs are used. The two DataIO FPGA's receive a total of 12 XFT channels.

A 32 words deep 16-bit input FIFO is used to receive the XFT data from one channel. With Statemachine 1 (See Appendix 3) and two 16-bit registers, 32-bit words are formed from the incoming 16-bit words. Statemachine 1 writes the 32-bit words into a input DAQ RAM and to a output FIFO. A L1A FIFO is used to save the Buffer number for the incoming events. The L1A FIFO is strobed on each L1A. Statemachine 1 also controls a write address counter which output is used as part of the write address for the input DAQ RAM. Buffer number controls the upper two bits of the write address. This way the RAM is divided into four buffers. Input DAQ RAM's are 512 words deep (128 / buffer). See Appendix 1 for details how each input DAQ RAM is mapped to VME.

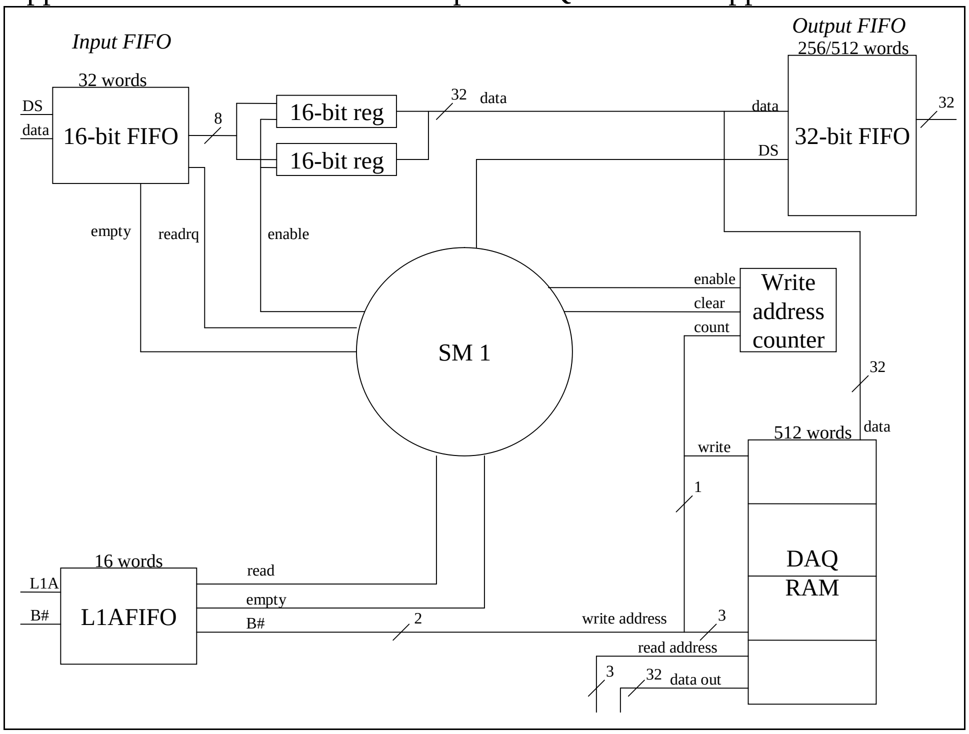

Figure 19: XFT DataIO FPGA input firmware block diagram

Instances of the same input component were used for each of the six input channels on the DataIO FPGA's to receive the XFT data.

The output FIFO's for each channel are read by Statemachine 2 (See Appendix 3), which merges the data and sends the data out to the Control FPGA. One instance of Statemachine 2 merges three channels. Two instances are used to merge all six channels. Data from the first input component (first channel on the first mezzanine card) is sent out first, data from the second input component next, and so on.

The DataIO FPGA firmware uses the latency measuring component to measure latency from L1A to when the data starts arriving at an input and when the whole XFT event is received. SLINK40MhzClk is used for measuring, so an increase of one in the latency value means 100ns. These latency measurement values are stored after the data at the end each channels input DAQ RAM for all four buffers.

Control FPGA firmware

The Control FPGA firmware receives the XFT data from the two DataIO FPGA's. The data is merged and written in a output FIFO. No re-formatting or other manipulation is done to the data.

The S-LINK formatter starts reading the Output FIFO when ever there is more data. The data is sent out from the S-LINK output in the Pulsar S-LINK format, using the S-LINK formatter component. S-LINK formatted output data is saved into DAQ RAM 1, which is 2048 words deep (512/buffer).

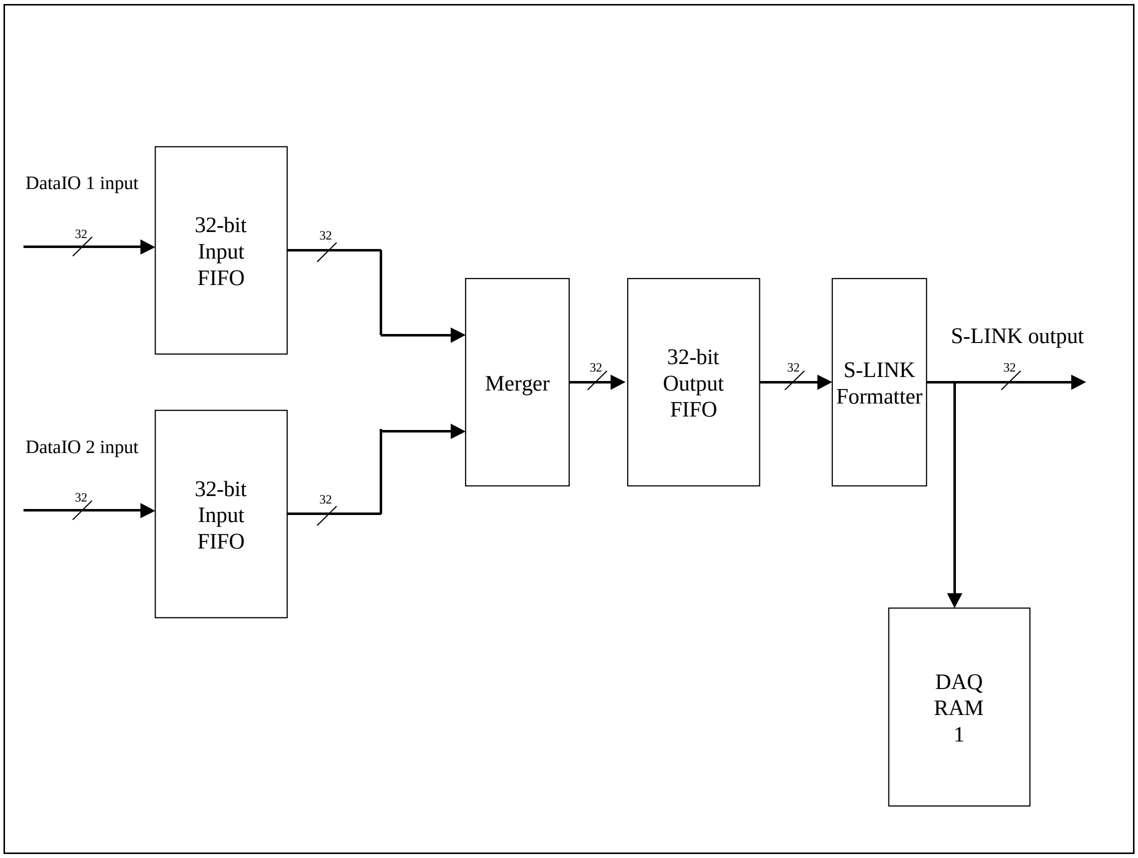

Figure 20: XFT Control FPGA firmware block diagram

RAM and FIFO sizes

DataIO FPGA's

One XFT channel (six / DataIO FPGA):

- Input FIFO: 32 x 16-bits
- Output FIFO:
  - First channel: 256 x 32-bits
  - Other channels: 512 x 32-bits

- Input DAQ RAM: 512 x 32-bits

Control FPGA

- Output FIFO: 2048 x 32-bits
- Output DAQ RAM: 2048 x 32-bits

2.3 S-LINK Merger

The S-LINK Merger board is used, in the L2 Pulsar trigger system, for merging S-LINK formatted data from multiple interface Pulsars into one S-LINK formatted output.

2.3.1 Functionality

The S-LINK input interface is through the S-LINK LDC mezzanine cards on the Pulsar board. One S-LINK LDC mezzanine card can receive one S-LINK channel. Two mezzanine cards are connected to each DataIO FPGA.

S-LINK output interface is through the P3 back plane connection, AUX card, and an S-LINK LSC on the AUX cards mezzanine card connector.

Inputs of the S-LINK Merger board are saved into DAQ RAM's 1 and 2, on both DataIO FPGA's. Output of the S-LINK Merger is saved to DAQ RAM 1 on the Control FPGA.

Number of inputs can be selected by changing a value of a Control register on the Control FPGA. By default all inputs are enabled. Incoming data is recorded to the DAQ RAM's on the DataIO FPGA's even if the inputs are disabled.

Outgoing data is sent in the Pulsar S-LINK format. Outgoing data has the Buffer number, seen by the S-LINK Merger board for the current event, and the Bunch counter value, counted by the S-LINK Merger board, in the S-LINK header word. Also Data source value is set accordingly to the S-LINK header word (See Appendix 2).

Input latency from L1A to the S-LINK data arriving on an input is measured for each input.

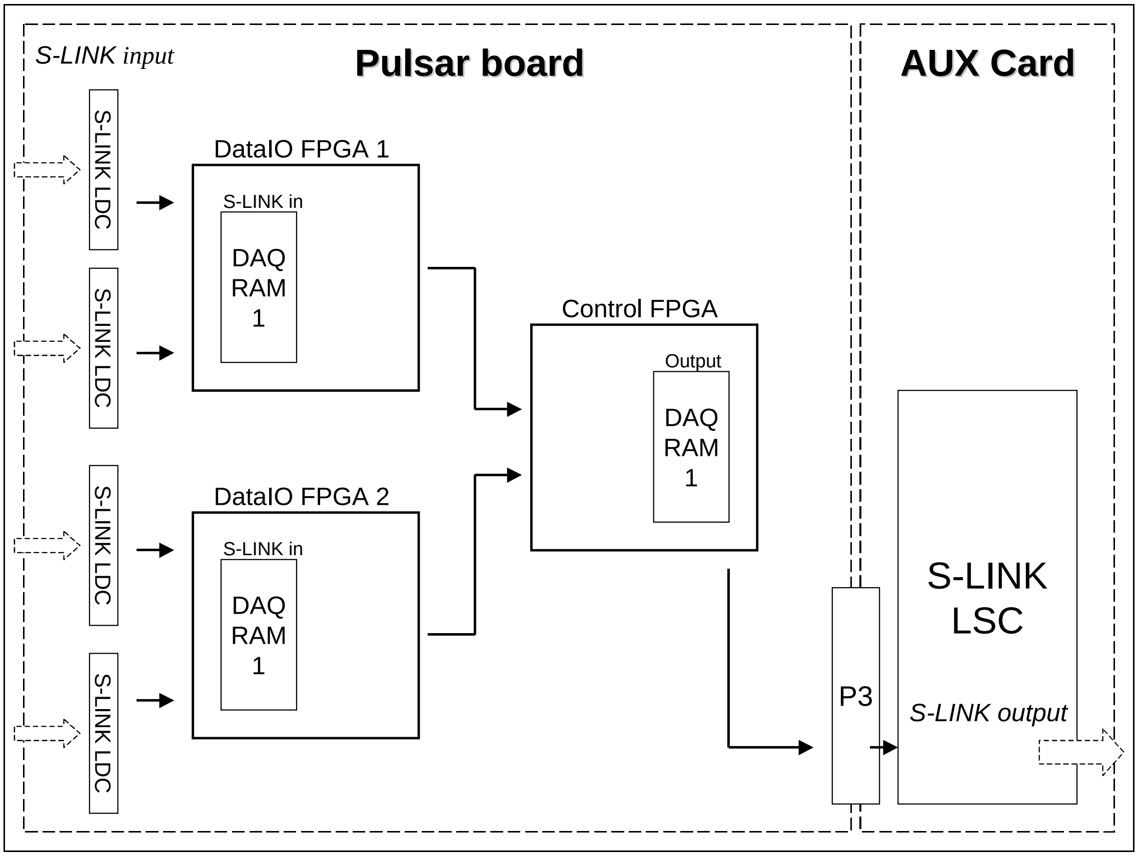

Figure 21: S-LINK Merger board functionality

2.3.2 Implementation

The combination of two DataIO FPGA firmware and the Control FPGA firmware receive S-LINK formatted data from one to four S-LINK inputs. The firmware merge the data from the inputs and send the data out from an S-LINK output in the Pulsar S-LINK format.

DataIO FPGA firmware

The DataIO firmware for both DataIO FPGA's is identical. One DataIO FPGA receives 32-bit S-LINK data from two S-LINK LDC mezzanine cards. Both inputs are saved into separate DAQ RAM's. Data from the first mezzanine card is saved into DAQ RAM 1 and data from the second mezzanine card is saved into DAQ RAM 2.

Incoming data from the two inputs is merged into one output, in a way that the data from the first input goes out first and the second input second. The merged data is sent out to the Control FPGA.

Number of inputs which are enabled can be controlled from the Control register 2 in the Control FPGA VME interface. Depending on the input selection, the DataIO FPGA firmware selects if the data from an S-LINK input is send to the Control FPGA or not.

Even if an input is disabled the incoming data is still saved to the DAQ RAM. The RAM's are 2048 words deep (512/buffer).

The DataIO FPGA firmware uses the latency measuring component to measure latency from L1A to when the data starts arriving at an input and when the whole S-LINK fragment is received. SLINK40MhzClk is used for measuring, so an increase of one in the latency value means 100ns. These latency measurement values are stored after the data at the end each inputs DAQ RAM for all four buffers.

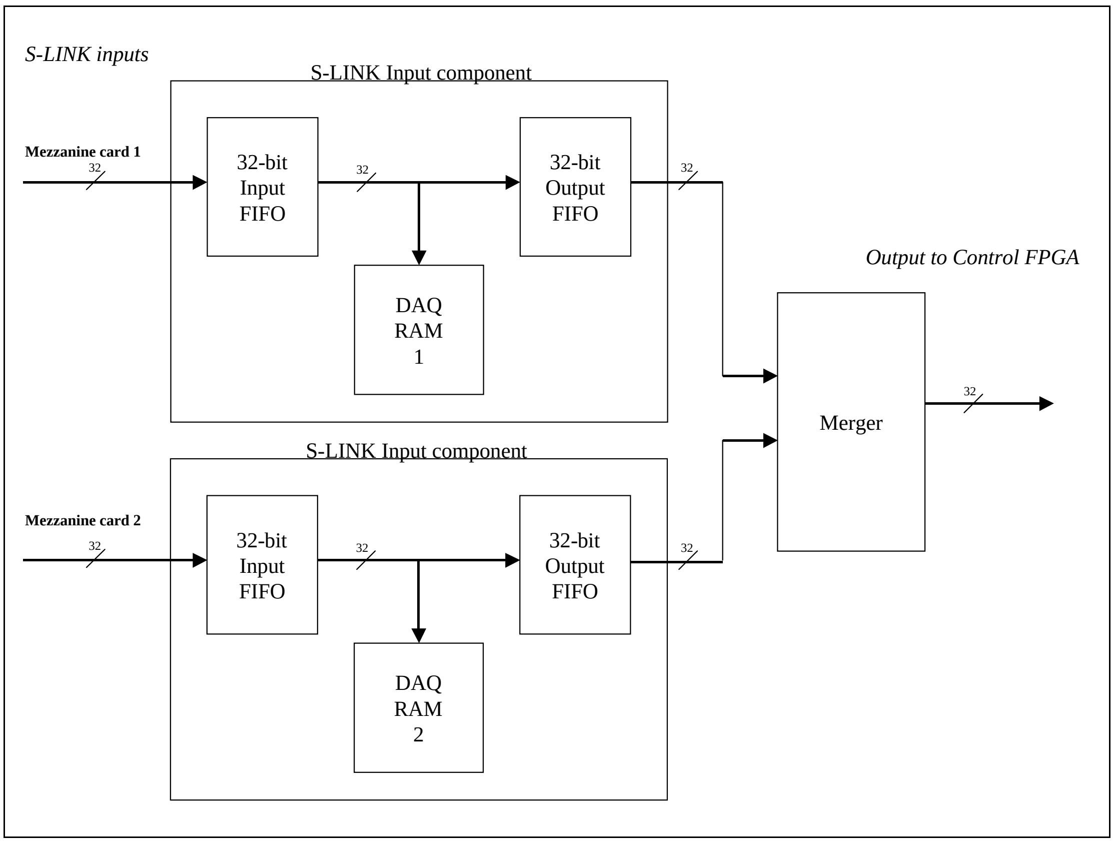

Figure 22: S-LINK Merger DataIO FPGA firmware block diagram

Control FPGA firmware

The Control FPGA receives data from one or both DataIO FPGA's, depending on the input selection

Control FPGA merges the data from the DataIO FPGA's and puts the data to an output FIFO. The S-LINK formatter starts reading the Output FIFO when ever there is more data. The data is sent out from the S-LINK output in the Pulsar S-LINK format, using the S-LINK formatter component. S-LINK formatted output data is saved into DAQ RAM 1, which is 2048 words deep (512/buffer).

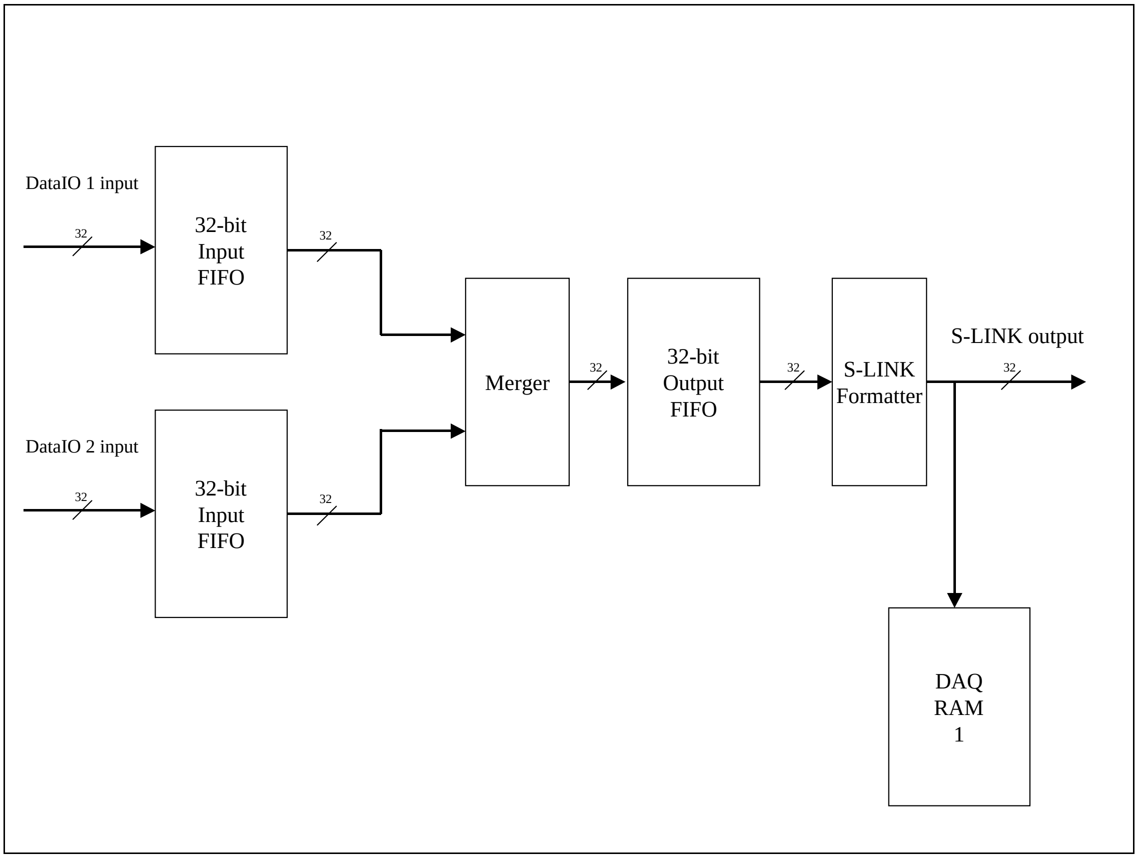

Figure 23: S-LINK Merger Control FPGA firmware block diagram

RAM and FIFO sizes

DataIO FPGA's

One S-LINK channel (two / DataIO FPGA):

- Input FIFO: 256 x 32-bits
- Output FIFO: 512 x 32-bits
- Input DAQ RAM: 2048 x 32-bits

Control FPGA

- DataIO FPGA 1 input FIFO: 512 x 32-bits
- DataIO FPGA 2 input FIFO: 512 x 32-bits
- Output FIFO: 2048 x 32-bits
- Output DAQ RAM: 2048 x 32-bits

2.4 L2toTS interface

The L2toTS board is used, in the L2 Pulsar trigger system, for receiving decisions from the Decision PC, and to negotiate the decisions to the Trigger Supervisor.

Functionality

The decision information comes from the Decision CPU to DataIO FPGA 1 on the Pulsar board, through an S-LINK LDC mezzanine card on the first mezzanine card connector.

On the DataIO FPGA, the data from the Decision CPU is saved into DAQ RAM 1. The first word from the Decision CPU has the information needed to negotiate the decision to the Trigger Supervisor. This word is sent to the Control FPGA.

The Control FPGA sends the decision to the Trigger Supervisor and handles handshaking with the Trigger Supervisor.

A delay can be set to the Control FPGA firmware to delay the start of the handshake. If a delay is set, the firmware will wait before it gives the decision to the TS after it gets the decision from the Decision PC. By default at power-up the value is set to 0, so there is no delay. The delay value can be changed from Control register 4 on the Control FPGA. The unit is 25ns.

For testing purposes, when the Decision CPU cannot be used, it is also possible that the L1 trigger input on the Control FPGA is used as the decision source. Control register 3 on the Control FPGA is used to switch between the decision sources.

Input latency from L1A to the S-LINK data arriving on the DataIO FPGA is measured. The values are saved into DAQ RAM 2 on the DataIO FPGA.
Three different latencies on the Control FPGA are measured:

1. From L1A to L2toTS Pulsar sending decision to the Trigger Supervisor
2. From L1A to Trigger Supervisor sending global decision back to L2toTS Pulsar
3. From L1A to Trigger Supervisor broadcasts L2 decision

The latency values are saved into DAQ RAM 2 on the Control FPGA.

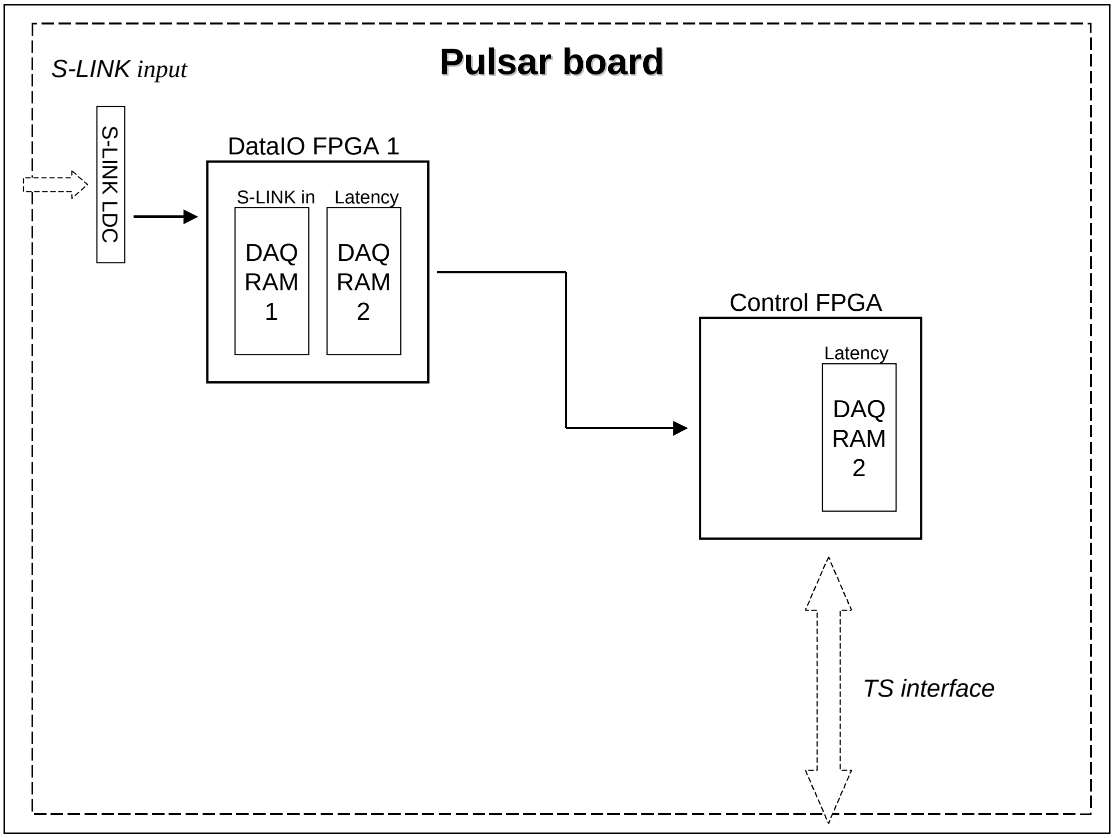

Figure 24: L2toTS board functionality

Implementation

DataIO FPGA firmware

DataIO FPGA 1 receives 32-bit S-LINK data from the S-LINK input interface. The data is saved into DAQ RAM 1. First word, the decision word, from the data is immediately sent to the Control FPGA as it arrives.

The DataIO FPGA firmware uses the latency measuring component to measure latency from L1A to when the data starts arriving at the input and when the whole S-LINK fragment is received. SLINK40MhzClk is used for measuring, so an increase of one in the latency value means 100ns. These latency measurement values are stored into DAQ RAM 2 for all four buffers. The RAM is 4096 words deep (1024/buffer).

Control FPGA firmware

Control FPGA receives the decision word from the DataIO FPGA. The decision word data content is described below.

Bits	Description
31…23	Format version
23…20	Data source
19…18	Region ID
17…16	Spare
15…13	RO List
12	L2 Timeout
11	L2 Reject
10	L2 Accept
9…2	Bunch counter
1…0	Buffer number

Table 15: Decision word format

Bits 12, 11 and 10 are used by the firmware to negotiate the decision to the Trigger Supervisor. Also bits 12 to 15 are send to the Trigger Supervisor. The firmware uses buffer number bits, seen by the L2toTS board, for the current event, in the handshaking (not buffer number bits from the decision word). See Appendix 3 for a diagram of the TS handshake statemachine.

Buffer number bits in the decision word are compared against the buffer number bits seen by the L2toTS board. If there is a mismatch, an error LED (red) is lit. Also bunch counter bits are compared against a Bunch counter value from a Bunch counter component in the Control FPGA firmware. Another error LED (green) is used to indicate this mismatch.

If the L1 trigger interface is selected as a decision source, instead of bits 12, 11 and 10 from the decision word, bit 2 from the L1 trigger input is used in the TShandshake component to send the decision to the Trigger Supervisor. L2 Timeout is not possible when using decision from L1 trigger input. Buffer number and Bunch counter checking is also not possible. This is only used when testing the board without the Decision CPU.

The Control FPGA firmware uses the latency measuring component to measure the latencies described above. These latency measurement values are stored into DAQ RAM 2 for all four buffers.

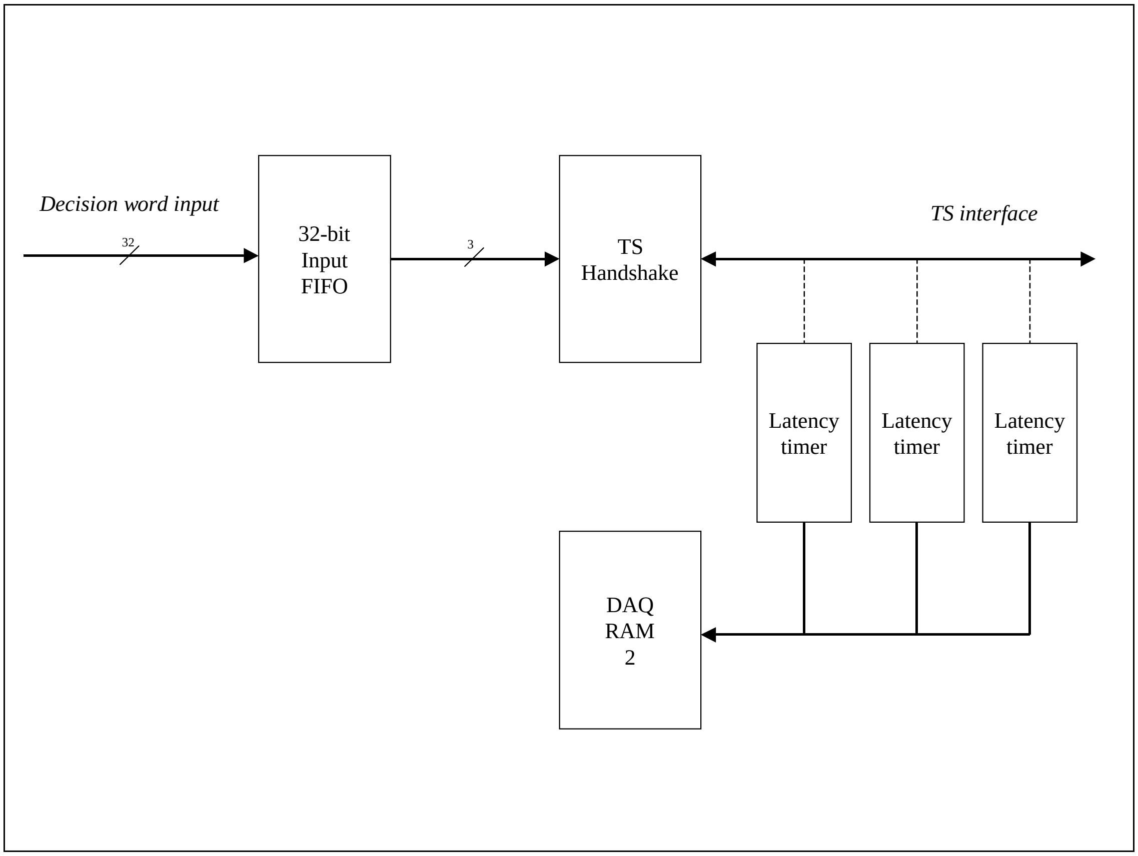

Figure 25: L2toTS board Control FPGA firmware block diagram

RAM and FIFO sizes

DataIO FPGA

- Input FIFO: 512 x 32-bits
- Input DAQ RAM: 4096 x 32-bits
- Latency DAQ RAM: 2 x 32-bits

Control FPGA

- Input FIFO: 16 x 32-bits
- Latency DAQ RAM: 3 x 32-bits

3 Pulsar firmware organization issues

For the L2 trigger system a lot of different firmware has been implemented, not only for the boards that are in the system, but also for boards used in testing the firmware for those boards. The firmware has been constantly improved, so a lot of different versions have been created. To keep different versions of firmware in order, a careful update procedure has to be followed. Also a version control system is required.

3.1 Pulsar firmware development and testing procedures

The Pulsar firmware design is very modular. A lot of the firmware is re-usable, and it is commonly used on different firmware. Before implementing any new features to a firmware, the features and the implementation are planned with a group of people (usually two or three). When the implementation is done, the design is carefully simulated before going for test stand testing. The firmware is first tested in the test stand for millions of events, with real-like data. Pulsars with transmitter firmware are used to emulate real upstream data. Only after the test stand testing is successfully finished, the firmware is tested in the beam with real data. If the firmware passes the initial beam test, it is left in to the system to collect more data. The data from the beam is checked online or later offline. For testing different firmware a dedicated group of people is used. People who know the particular part of the system and know the firmware they are testing.

3.2 Firmware organization and version control

Every time a change is made to a firmware, that firmware gets a new version number, and changes made to the firmware are documented into a change log file. CVS is used for the VHDL source code files. Also other project files and FPGA programming files are saved every time a new version is created.

For each new firmware version, a procedure is followed to minimize the change of forgetting to do something. The procedure is as follows:

- ✓ Change firmware version register value
- ✓ Check pin file
- ✓ Update change log file
- ✓ Copy files to Pulsar firmware configure
    - Local
    - Test stand
- ✓ Configure FPGA
- ✓ Check read only register
- ✓ Configure EPC2s

Firmware versions are divided into two categories; stable and development. Versions that go for test stand testing go into the development tree. When it is decided that a version passes the test stand testing, a stable version is created from it. Stable versions are only updated when added features have been adequately tested in the test stand.

3.2.1 Firmware version register

Firmware version register is used to identify different versions of one firmware, and to identify firmware of different Pulsars and FPGA's. The firmware version register value uses 8-bits for firmware ID, 20-bits for date and 4-bits for version number. For all firmware to different FPGA's and Pulsars, a firmware ID has been assigned.

Firmware version register format

Firmware ID	Date	Version number
8-bits	20-bits	4-bits

Example

DataIO S-LINK Merger 02/20/04 first version Hex: 01402200

Table 16: Firmware version register

ID	Firmware
01	DataIO S-LINK Merger
02	Control S-LINK Merger
03	DataIO Muon RX
04	Control Muon RX
05	Control SVT RX
06	DataIO CLIST RX
07	DataIO ISOLIST RX
08	DataIO ShowerMax RX
09	DataIO L2toTS
0A	Control L2toTS
0B	Control Cluster merger

ID	Firmware
A1	DataIO Muon TX
A2	Control Muon TX
A3	DataIO ShowerMax TX
A4	DataIO ISOLIST TX
A5	DataIO CLIST TX
A6	Control SVT TX

Table 17: Firmware IDs

List of references

[1] **CERN S-LINK homepage**, *http://hsi.web.cern.ch/HSI/s-link/*

[2] **Run IIb Upgrade for CDF L2 Decision Crate,** *CDF note 6259*

[3] **Pulsar Project webpage**, *http://hep.uchicago.edu/~thliu/projects/Pulsar/*

[4] **Description of transmitter firmwares for the Pulsar board,**
http://hep.uchicago.edu/~thliu/projects/Pulsar/Pulsar_doc/notes/TransmitterFirmwares.doc

[5] **Standard Front-End and Trigger VMEbus Based Readout Crate for the CDF Upgrade-The CDF Readout Crate**, *CDF note 2388*

Appendixes

Appendix 1: VME address maps, FIFO and DAQ RAM sizes
Appendix 2: Board types, Data Source values, Firmware ID’s
Appendix 3: Statemachine diagrams