

# L1FPSS

## **Technical Design Report**

**ver 1.01**

**14 November 2003**

**Jamieson Olsen**

**DØ Experiment**

**Fermi National Accelerator Laboratory**

# 1. INTRODUCTION

The Forward Preshower System (FPS) is a sub-system of the Central Track Trigger (CTT) [1]. The FPS system makes trigger decisions and passes these trigger bits onto the trigger framework. Unlike the CFT/CPS axial and CFT stereo detectors, the FPS detectors are positioned in the transverse plane and consist of 16 phi wedges positioned on each of the North and South endcaps.

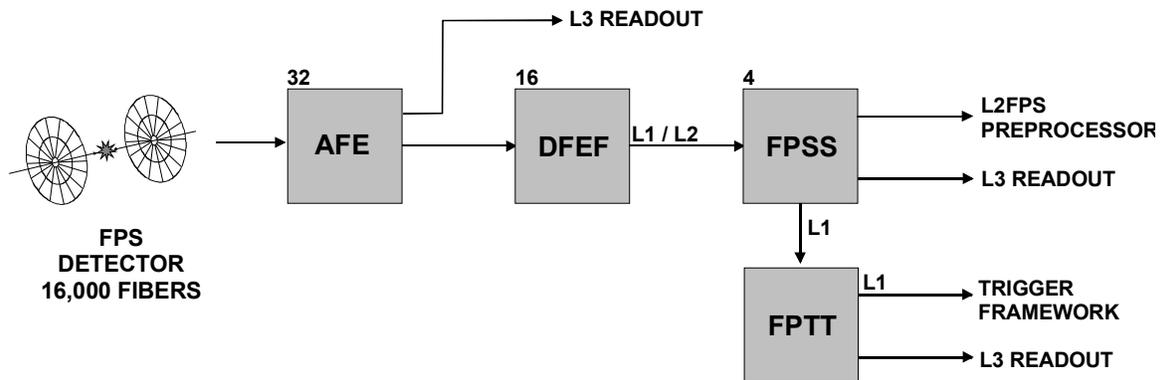
The FPSS is a double-wide daughterboard that is mounted to a Digital Front End Motherboard (DFEM). The FPSS receives L1 and L2 records from the DFEF boards over eight LVDS links. There are a total of four FPSS boards; thus each FPSS board handles one quarter of the FPSS detector (NE, NW, SE, and SW). Incoming L1 records contain counts of FPS clusters; while L2 records contain sorted lists of clusters. The FPSS has three main functions:

- Sum the L1 cluster counts and pass this information on to the FPTT.
- Concatenate the L1 records and send them to the L3 readout crate (0x13).
- Sort, concatenate and truncate the list of L2 clusters and pass it on to the L2FPS Pre-Processor.

L1 and L3 functionality are contained in one FPGA, while the L2 functionality is contained in the other FPGA on the FPSS daughterboard.

*Note: the L2 functionality is described in a separate Technical Design Report [2] and not addressed here.*

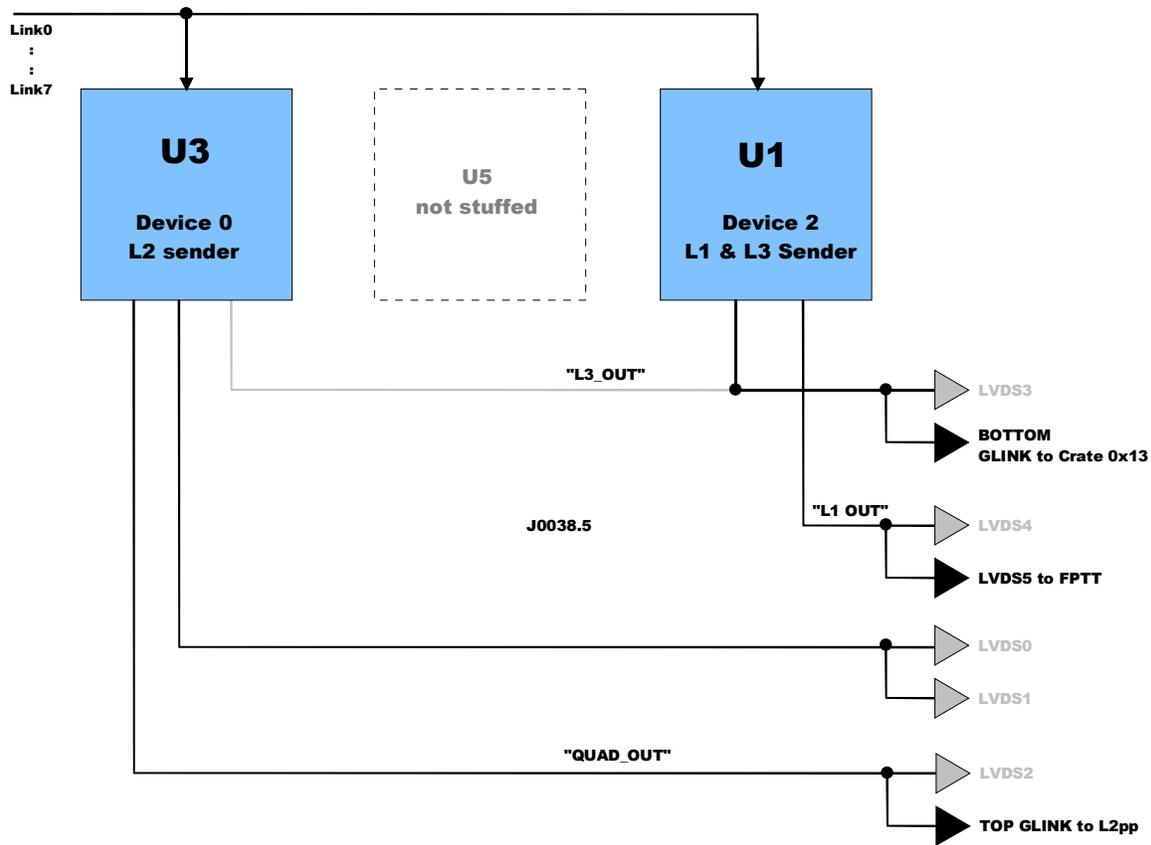
## 2. FPS CHAIN DIAGRAM



There are a total of 32 LVDS links between the DFEF and FPSS boards. Each link corresponds to one FPS wedge. Detailed cable maps are available on the DFE Connections webpage [3]. The FPSS board has three outputs: the L2 and L3 outputs are fiber optic (G-LINK) transmitters, while the L1 link to FPTT is LVDS.

# FPSS BUS STRUCTURE

Eight 28-bit LVDS links feed each FPSS. These ten busses are expected to be aligned to within  $\pm 1$  18.8ns clock tick of each other. Each FPGA on the FPSS sees all eight input links.



The DFE motherboard provides a 53MHz clock and reset; the output busses should be synchronous to this clock. The motherboard also provides a path for FPGA configuration data. After the FPGAs are initialized the motherboard provides a path for setting parameters in the FPGAs and collecting status information.

Some of the output busses go to more than one FPGA – in the diagram above these “extra” data paths are shown in gray lines. These busses should not be used. For example, U3 should not drive the “L3\_OUT” bus.

### 3. RECORD TYPES AND DEFINITIONS

See the Protocols Document version 7.00 [4].

The FPSS expects to see 4 frame L1 records followed by 3 null frames arriving on its inputs. When a L1\_accept occurs the DFEF boards will momentarily interrupt L1 transmission and insert a L2 record into its output datastream. After the L2 record there will be some null frames, then L1 transmission will resume. For details refer to the DFEF Technical Design Report [5].

The incoming L2 records consist of two header frames, followed by 0 to 24 cluster data frames, followed by a trailer.

### 4. L1FPSS FUNCTIONS

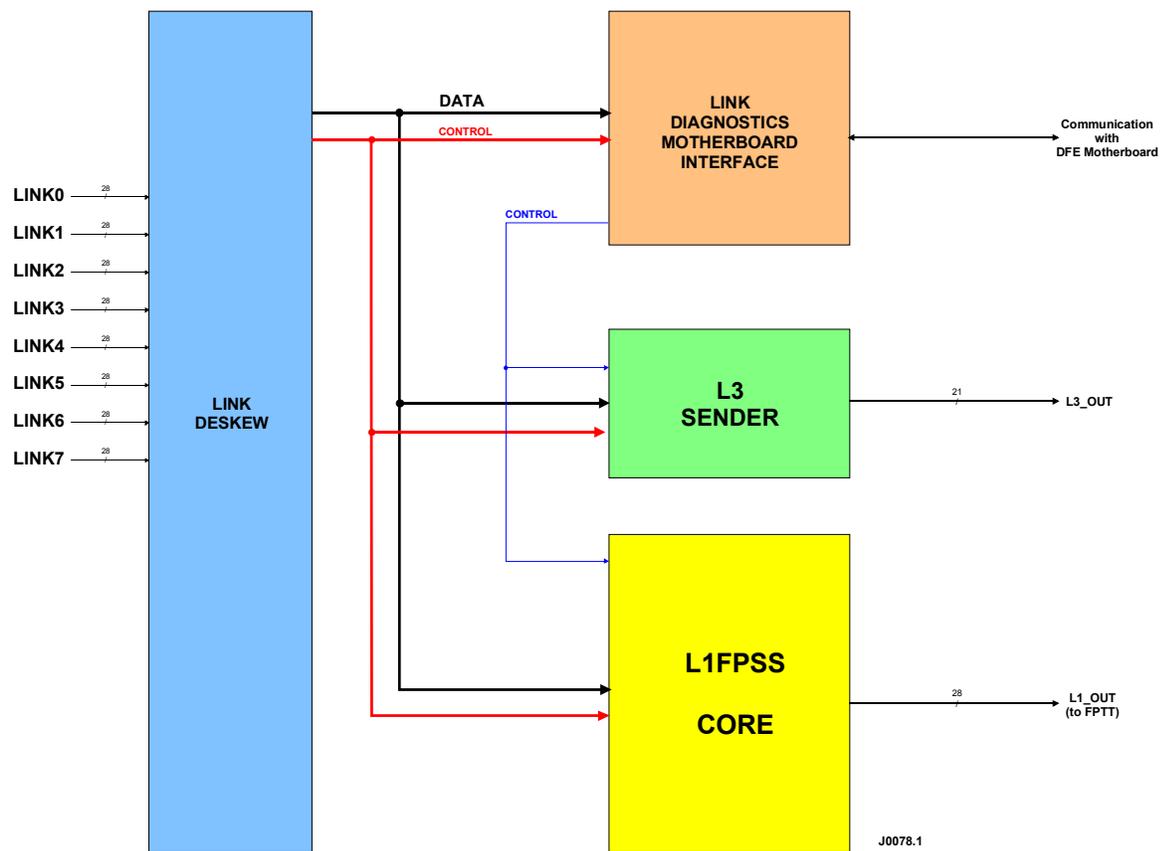
#### 4.1.1. L1 OUTPUT

The L1FPSS must continuously process L1 records sent to it by the DFEF boards. Each L1 record contains four 4-bit counts. The L1FPSS Core module sums and truncates these fields. The result is put into a L1 record and sent to the FPTT board. This L1 path must be optimized for minimal latency.

#### 4.1.2. L3 OUTPUT

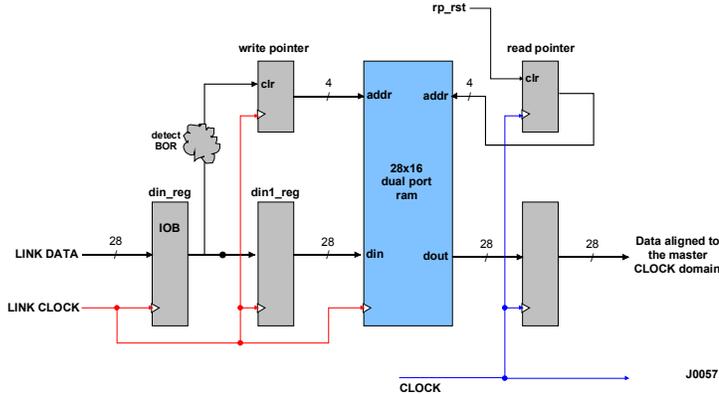
Incoming L1 records are stored in memory (pipeline). When a L2 record arrives at the FPSS it jumps back a  $n$  records in this memory and collects the eight L1 records for the requested event. These eight records are concatenated and wrapped in a L3 record before transmission to the L3 readout crate. The pipeline depth  $n$  is adjustable dynamically after the FPSS is initialized. **NOTE: the vertical parity word in the L3 record trailer is not supported at this time.**

# 5. L1FPSS BLOCK DIAGRAM



## 5.1. LINK DESKEW FRONT END

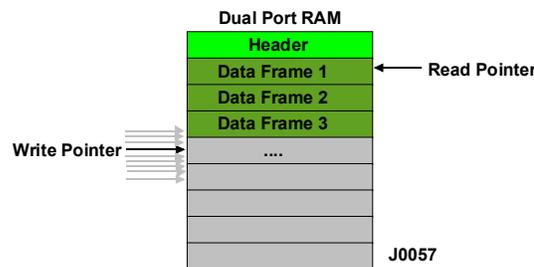
The main purpose of the front end is to cleanly and reliably cross the link clock domains so that the rest of the design operates in the master clock domain. Small dual port rams are used as FIFOs to cross the clock domains. These dual port rams are constructed from distributed RAMs which are smaller than the true dual port BlockRAMs found on the Xilinx FPGAs. As a result, the distributed RAMs can be pushed closer to the input bus pins, thus reducing clock delays and clock skews on non-global clock nets (Links 3-9 use non-global clocks). Below is deskew module for one link:



The registers and write pointer logic in the link clock domain insure that the first frame of a record (L1 or L2) always goes into address 0 of the RAM. The link data is written into the RAM on the rising edge of link\_clock. As the record comes in, subsequent frames are stored in increasing memory locations in the RAM.

After a few frames have been written into the RAM the read pointer is set to zero synchronous to the master clock (called clock from here on out). As the read pointer increments it pulls frames out of the RAM and delivers them to the rest of the design. All link modules have their read pointers reset at the same time.

The read pointer must always trail the write pointer by a few memory locations in order for this to work:



Because the read pointer is trailing a few locations behind the write pointer the design is not sensitive to skew and jitter in the link clock domains. The farther apart the read and write pointers are the more tolerant the design is to link skews, however this also increases latency through the front end. Currently the design separates the pointers by two memory locations, allowing for link skews of about  $\pm 2$  clock ticks. (In reality the skews have measured to less than 5ns.)

With all the ugly clock domain crossing logic hidden in the front end, the rest of the design sees a clean link interface: ten 28-bit busses and a master beginning of record (BOR) marker, all signals are synchronous to the master clock. When BOR is asserted, each of the ten busses should be displaying the first frame of a record (whether it is a L1 or L2 record is doesn't matter). Input link number two was chosen as the master link – this

module's BOR output is derived from input link2. *This also means that link2 must be working properly otherwise the link deskew module will break.*

The front end also checks for missing links. A link is considered missing if a BOR marker is not seen within 64 link clock ticks. A eight bit output bus called `valid_link` provides this information to the rest of the design. For example, if link 3 is not plugged in, `valid_link(7..0)= 1111 0111`.

## 5.2. STATUS AND MOTHERBOARD INTERFACE

**NOTE: everything contained in this section applies only to the L1FPSS FPGA.**  
**The interface to the L2FPSS FPGA is different, and it is described in detail in the L2FPSS Technical Design Report [2].**

This functional block handles all communication with the DFEA motherboard. After the FPSS board is initialized the DFE motherboard basically acts as a conduit for low speed status information and parameter setting. For example, after initialization the user wants to change the L3 pipeline depth. The user runs a `DFE_Ware` macro which sends the parameter to the DFE crate controller (DFEC). The DFEC writes the parameter over the DFE backplane to the specified DFE motherboard. The DFE motherboard serializes the parameter and sends it to the specified FPGA. The FPGA converts the parameter back to parallel and make it available to the L3 sender module – now the pipeline depth has been changed.

This module also sends status information back down to the motherboard one “page” at a time. The motherboard passes this information back to the DFEC over the slow monitor bus. This bus runs continuously and is completely independent of the main DFE backplane bus. The DFEC continuously collects this slow monitor information and makes it readable from the 1553 bus. `DFE_Ware` regularly scans the DFEC and extracts this status information.

After synchronization the input link busses are checked for things like parity errors, synchronization errors, control bit errors, etc. These status bits can be passed down to the motherboard. A counter that keeps track of the turn and crossing numbers is also instantiated in this module.

### 5.2.1. INPUT LINK DIAGNOSTICS

**Missing Link Errors.** If an input link module does not see a beginning of record marker within 64 clock ticks it considers that input link to be missing. This logic is actually contained in the Link Deskew module, but the eight bit bus is sent to the motherboard interface module where the missing link history register is maintained. If a link goes missing – even for a brief moment— the corresponding bit in the missing link history register is set, and remains set until the history is manually cleared. [history\\_reg.vhd](#)

**Parity Errors.** L1 records have both a horizontal and vertical parity bits. The vertical parity bits were added to enable parity detection and correction for the L1 data. Currently the FPSS design ignores these vertical parity bits and simply checks the first 3 frames of the L1 record for horizontal parity errors. **No effort is made to correct parity errors.** If a link has a parity error the corresponding bit in the link parity history register is set, and stays set until manually cleared. [parity\\_check.vhd](#), [parcheck\\_module.vhd](#)

**Synchronization Errors.** The `Link_Dskew` module attempts to align the link data and move everything into the master clock domain. If everything is working correctly then the `Link_Deskew` module asserts the BOR output then all output busses should be displaying a L1 (or L2) header. The synchronization error logic checks for conditions when this isn't the case. Only valid (non-missing) input links will be considered by the synchronization logic. [sync\\_detect.vhd](#)

**Test Pattern Detection.** The DFEF can be placed in a mode where it repeatedly sends a L1 test pattern. This is basically a L1 record with a walking 1's pattern inside of it. Logic in the L1FPSS checks that this pattern is

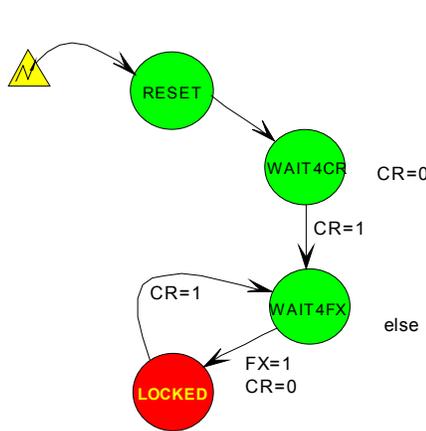
received error free. If not, it sets a bit in the pattern history register. This is only intended to be used for DFEF to FPSS link tests. [pattern\\_checker.vhd](#), [patt8.vhd](#), [patt\\_detect.vhd](#)

### 5.2.2. CONTROL BIT DIAGNOSTICS

The L1FPSS checks the system control bits embedded in the incoming L1 records. The FXX and L1X status bits are XORs of the FX and L1a control bits, respectively. If all L1 inputs are properly aligned then the XOR of these control bits will always be zero.

### 5.2.3. TICK AND TURN COUNTER

The L3 sender needs to know the tick (crossing) and turn numbers so that it can insert these values into the L3 header. Based on the FX and CFT\_RESET (CR) control bits received on link2 the counters are locked onto the incoming data stream. The tick and turn counter logic is based on the simplified state diagram:



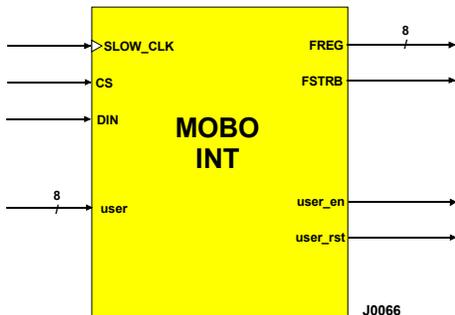
CR happens infrequently (few times per hour) and is often asserted for a second or so. This active high signal is a global system reset (also called SCL\_INIT). On the first FX after the CR goes low the counters are reset to turn=1, tick=7. The tick counter counts up to 158 and is then reset to zero, and the turn counter is incremented. If all is proceeding normally the tick and turn counter module will drive the LOCK output high.

When the tick counter equals 6 it *expects* to see the next FX bit in the next L1 record. The tick and turn counter must be able to “ride out” a missing FX bit without dropping the LOCK bit. If FX comes at an unexpected time, however, the tick and turn module will pulse the BMFX (Bad Master FX) output.

Both LOCK and BMFX status bits are available to the outside world through the slow monitor interface. BMFX is run through a history register – once it goes high it stays high until the history is cleared. The LOCK status bit is passed straight through. [tick\\_turn.vhd](#)

## 5.2.4. DFE MOTHERBOARD INTERFACE

The module *mobo\_int.vhd* forms the core of the DFE Motherboard Interface. It handles the low-level serialization and de-serialization of data between the FPSS FPGA and the DFE motherboard.



When the DFE motherboard writes to the FPSS device the data byte is shifted in on the DIN input MSb first. The CS input is an active low control line that is low for the entire time it takes to shift in the eight bits. Shortly after CS goes high the data byte is driven out on the FREG bus and the FSTRB line is pulsed high for one slow\_clk period. The DIN and CS inputs are sampled by slow\_clk.

In the other direction the eight bits applied to the USER input are continuously serialized and sent to the motherboard over the USER\_EN and USER\_RST lines. The DFE motherboard collects this user byte from the *current device* and sends it over the slow monitor bus as the slow monitor user bits[7..0]. The two devices on the FPSS daughterboard have separate data paths to send their status bytes down to the motherboard, however the motherboard can only read the status information from one device at a time.

### 5.2.4.1. SETTING PARAMETERS AFTER INITIALIZATION

After the L1FPSS device is initialized some additional arguments may be passed to it. These are:

- **L3 Pipeline Depth.** This is adjustable from 0-36 crossings deep. The default is 34.
- **Fake L1 record select bits.** For diagnostic purposes, the FPSS can send fake L1 records to the FPTT. Default is zero.
- **Status Page.** The FPSS has a lot of status information, but it can only be viewed in one byte-wide “page” at a time. Default is page 0.

As previously mentioned, *mobo\_int* supports writing and reading an 8-bit value to/from each device on the FPSS. This does not provide enough bits to do everything the FPSS needs it to do, so a simple protocol is imposed on this interface. The feature register now has four independent registers, selected by bits 7 and 6:

7	6	5	4	3	2	1	0
0	0		CH	Page Select			
0	1	L3 Pipe Depth (0-36)					
1	0						
1	1			Fake L1			

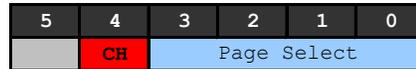
2003-06-10a

For example, to change the L3 Pipeline depth to 25 do the following:

```
set DFE 3
set device 2
# 25 = 0x19 Now OR 0x19 with 0x40 to set bit 6.
cmd writebyte 3 0x59
```

## 5.2.4.2. STATUS PAGES

The L1FPSS supports 16 pages of status information. Each page is a single byte of status information. The current status page becomes the slow monitor user bits for the FPSS board. Previous versions of the status information used logic to stretch error pulses so that spurious errors had a better chance of being sampled by the slow monitor system. However, this method had shortcomings and has been replaced with a status “history” scheme. Status history works like this: after reset all of the FPSS status pages are set to 0x00. On every 53MHz clock edge the contents of the status pages are updated. If a bit goes high, it stays high until the status page history is cleared. Setting Bit 4 of the Page Register clears the status history for the status pages.



- *After power up or board reset the history bits may contain garbage. So it is good practice to clear the history after these conditions.*
- *The history is cleared on the 0 to 1 transition of the CH bit. When clearing the history it's best to explicitly toggle it by writing 0x00, 0x10, and 0x00.*

Below are the 16 status pages specific to the L1FPSS design.

page	S5	S4	S3	S2	S1	S0		
0	FXX	BMFX	L1X	L1A	PE	ML	SE	LOCK
1								
2					Fake_L1			
3					L3_Pipe			
4								
5					0x55			
6					0x55			
7								
8	Miss-link7	Miss-link6	Miss-link5	Miss-link4	Miss-link3	Miss-link2	Miss-link1	Miss-link0
9								
10	Sync-err7	Sync-err6	Sync-err5	Sync-err4	Sync-err3	Sync-err2	Sync-err1	Sync-err0
11								
12	Parity-err7	Parity-err6	Parity-err5	Parity-err4	Parity-err3	Parity-err2	Parity-err1	Parity-err0
13								
14	Patt-err7	Patt-err6	Patt-err5	Patt-err4	Patt-err3	Patt-err2	Patt-err1	Patt-err0
15								

2003-06-10a

**NOTE: the orange cells are cleared by the clear history feature.**

- FXX:** This bit is set if there is a disagreement amongst the FX control bits coming in on the eight input links.
- BMFX:** Bad Master FX. This bit is set if the FX control bit (on Link2) is missing or arrives at an unexpected time.
- L1X:** This bit is set if there is a disagreement amongst the L1 accept control bits coming in on the eight input links.
- L1A:** This bit is set whenever a L2 record comes in on Link2.
- PE:** Parity Error. Logical OR of the Parity-err[7:0].
- ML:** Missing Link. Logical OR of Miss-link[7:0].
- SE:** Synchronization Error. Logical OR of Sync-err[7:0].
- Lock:** Set to indicate that the Tick and Turn counter is happy with the control signals it's receiving from the DFEF, and that the tick and turn counters are incrementing normally.
- Fake\_L1:** Returns the value of the Fake\_L1\_mode register.
- L3\_Pipe:** Returns the value of the L3 sender's pipeline depth.
- Pages 5-6:** 0x55 and 0xAA, respectively. Used for debugging the interface.
- Page 8:** The Missing Link error bit for each input link is listed here. If a link has not seen a Beginning of Record (BOR) transition in at the last 63 clock cycles, the corresponding bit is set and stays set until the history is cleared.
- Page 10:** After the links have been synchronized, a circuit checks that each link's BOR signal is aligned with the master BOR signal. If there is a synchronization problem with a link, the corresponding bit is set and stays set until the history is cleared. Only considers non-missing input links.
- Page 12:** As L1 records come into the L1FPSS, the horizontal parity bits are checked for frames 1-3. If there is a parity error in the first 3 frames of the L1 record then the corresponding bit will be set and stays set until the history is cleared. Only considers non-missing input links.
- Page 14:** The L1FPSS front end contains test record detection circuitry. The DFEF can be placed into a special mode where it sends a special L1 test record to the FPSS. *Used only for DFEF to FPSS LVDS link tests.*

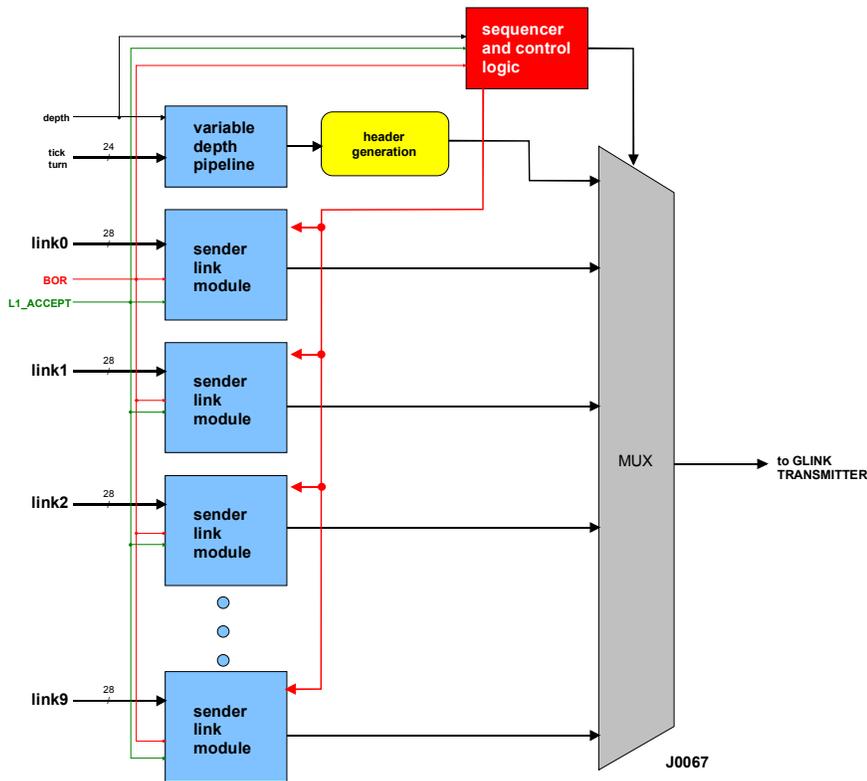
### 5.3. L3 SENDER

The purpose of the L3 sender is to wait for a L1\_Accept to occur, then jump back  $n$  crossings and concatenate all eight L1 records, wrap the raw link data in a L3 record and send it to the L3 processor via a G-LINK fiber optic link. The value of  $n$  is known as the L3 pipeline depth and it can be changed dynamically via the motherboard interface module after initialization. The L3 pipeline depth can range from 0 to 36 crossings deep, with the default being 34.

The FPSS uses the L3 sender originally developed for the CTOC design, where the L1 records are 7 frames long. Since the FPSS incoming L1 records are only 4 frames long the L3 sender will pad out the unused frames with zeros.

#### 5.3.1. L3 SENDER BLOCK DIAGRAM

Below is the block diagram of the L3 sender contained in the L1CTOC design [6]. The L3 sender in the L1FPSS is similar, only there are eight inputs, not ten.



#### 5.3.2. SENDER LINK MODULE

Most of the complexity in the L3 sender is pushed into ten sender link modules. These modules are comprised of two BlockRAMs, some counters, and glue logic. The two block rams are configured as 32x256 for writing, and 16x512 for reading. Normally these modules accept 28-bit link data and write it into the RAM and increment the write pointer on each clock tick. When a L2 header comes in (L1\_ACCEPT occurred) the first seven frames are written into the RAM and then writing is suspended. This is “readout mode”. The read pointer is then calculated as:

$$read\ pointer = write\ pointer - (7 * (depth + 1))$$

The read pointer now points to the header of the target L1 event. Now the sequencer and control logic block is responsible for extracting the link data from each of the sender link modules 16 bits at a time. To the sequencer and control logic block each sender link module looks like a 16x16 RAM. For example, if the control logic drives 0000 on the 4-bit address bus all of the sender link modules will display the L1 header frame bits [27..16]. 0001 = L1 header frame bits [15..0], etc.

A separate pipeline stores the tick and turn numbers for later retrieval. The depth of this pipeline can be dynamically adjusted from 0 to 48 crossings – it is adjusted so that when the sender link modules transition to “readout mode” the turn and crossing number of the target event is present on the outputs of this pipeline.

Most of the fields in the L3 header are constant, except for the tick and turn numbers. The sequencer and control logic controls a large mux that selects which 16-bit word to pass to the G-LINK transmitter. The G-LINK data bus bits[19..16] are used to mark the beginning (0x5) and end (0xA) of the L3 record. The total length of the L3 record is 6 headers + 112 data words + 2 trailers = 120 words, arranged as follows:

```

0101  L3 header1
0000  L3 header2
0000  L3 header3
0000  L3 header4
0000  L3 header5
0000  L3 header6
0000  0000 + link0 header [27..16]
0000  link0 link0 header [15..0]
0000  0000 + link0 data1 [27..16]
0000  link0 data1 [15..0]
0000  0000 + link0 data2 [27..16]
0000  link0 data2 [15..0]
0000  0000 + link0 trailer [27..16]
0000  link0 trailer [15..0]
0000  null 0x0000 <= padding frames
0000  0000 + link1 header[27..16]
0000  link0 link1 header[15..0]
:      : : :
0000  null 0x0000
0000  L3 trailer
0000  L3 parity word (0x0000)

```

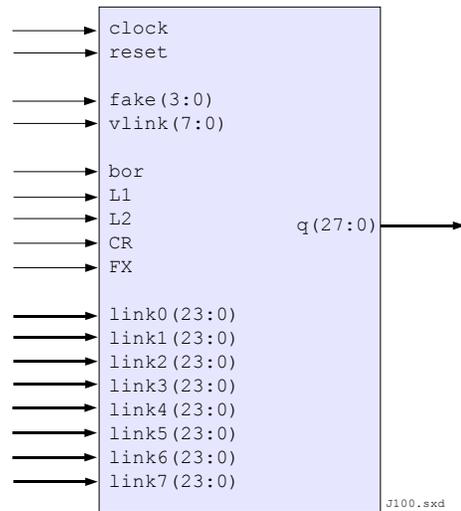
After the complete L3 record has been transmitted, the sequencer and control logic block asserts the RESUME signal, and all of the sender link modules return to “write mode” and continue to store the L1 data in their RAMS. The period L1\_ACCEPT period is equal to the depth of the L3 pipeline – nominally 34 132ns crossings. Also note that in “396 mode” there is a live crossings one out of every three crossings – the other two crossings don’t have any data in them. The FPSS design is 132ns mode compliant – it treats each crossing as through it contains valid data.

### 5.3.3. L3 SENDER TEST MODE

If the L3 depth is set to zero the L3 sender will produce a L3 record but instead of containing eight L1 records it will contain the first 7 frames of the L2 records received on each input link. This is only intended to be used for diagnostic purposes only.

## 5.4. L1FPSS CORE

The L1FPSS core design is quite simple. It receives four 4-bit counts from each DFEF board, truncates the counts at two bits and packs them into a six-frame L1 record which is sent to the FPTT board. The core module is shown below:



All inputs and outputs in core module are synchronous to the system clock. BOR is set to indicate that the first frame of a record is present on the link input busses; the control signals L1 and L2 specify what kind of record is coming in. Two system control bits FX (First Crossing) and CR (CFT\_RESET, aka SCL\_INIT) are supplied to this core design as well.

The valid link bus vlink(7:0) indicates which input link busses are delivering good data; the core design excludes any link bus that doesn't have a corresponding vlink bit set. The fake(3:0) bus controls what kinds of records are sent by the core design, see the next section for details.

The core module sees both incoming L1 and L2 records; L1 records are a fixed length and arrive at regular intervals. However, incoming L2 records are variable length and have some null frames inserted before the next L1 record. So normally the core will observe BOR=1 and L1=1 every 7 clock cycles. When a L2 record comes in then BOR=1 and L2=1, and there will be no more BOR signals for some time. During this time the core design "rides out" the L2 gap, and continues producing empty L1 records and sending them to the FPTT to keep that board synchronized.

While the FPTT needs to have L1 records coming in at regular intervals, it also needs to know that a L1\_ACCEPT occurred so that it can trigger its own L3 sender. To do this, the FPSS core design inserts a fake short L2 record into the L1 datastream. This L2 record is three frames long:

```
F000010    empty L2 header with #obj=0
1123456    static tick and turn number
0123446    parity frame
```

This L2 record is followed by four empty frames, followed by a L1 record. Refer to the Latency section to see a graphical description of the core behavior under these conditions.

## 5.4.1. DIAGNOSTIC OUTPUTS

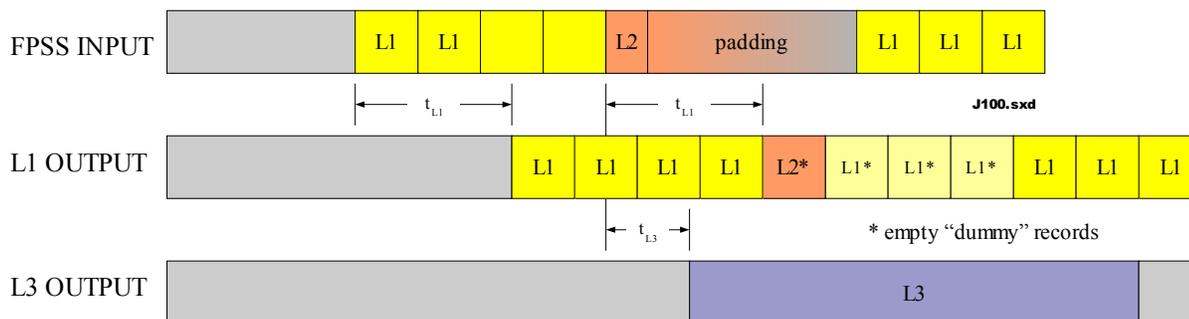
The L1FPSS can send several types of “fake” test records to the FPTT for system debugging:

fake bits	operation
0000	normal
0001	all L1 counts are "00"
0010	all L1 counts are "01"
0011	all L1 counts are "01" on first crossing, "00" otherwise
1111	send link test pattern, suppress L2 record
others	shutdown output link to FPTT

2003-11-14a

Note: The power on reset value is fake(3:0)="0000".

## 6. L1FPSS LATENCY



$t_{L1}$  is the Level-1 latency, 13 clock cycles (244.4ns).

$t_{L3}$  is the Level-3 latency, 9 clock cycles (169.2ns).

## 7. DEVICE RESOURCES

The L1FPSS design fits into a Xilinx Virtex 600 device:

Device utilization summary:

Number of External GCLKIOBs	4 out of 4	100%
Number of External IOBs	291 out of 404	72%
Number of LOCed External IOBs	291 out of 291	100%
Number of BLOCKRAMs	16 out of 24	66%
Number of SLICES	1821 out of 6912	26%
Number of GCLKs	4 out of 4	100%

The design is written entirely in VHDL. It has been functionally simulated in Active-HDL and synthesized using Xilinx XST. The design was placed-and-routed using Xilinx ISE tools.

## 8. REVISION HISTORY

- |      |                  |  |
|------|------------------|--|
| 1.00 | 11 November 2003 | created, leave out L1FPSS core stuff.                        |
| 1.01 | 14 November 2003 | document core section, include latency and device resources. |

## 9. REFERENCES

1. J. Olsen, et al. "The Central Track Trigger," *IEEE Trans. on Nucl. Sci.*, submitted for publication.
2. J. Olsen, "L2FPSS Technical Design Report," available:  
<http://www-d0online.fnal.gov/www/groups/cft/CTT/online/docs/index.html>
3. DFE Connections: <http://www-d0online.fnal.gov/www/groups/cft/CTT/online/docs/index.html#connect>
4. DFE Protocols: <http://www-d0online.fnal.gov/www/groups/cft/CTT/online/docs/index.html#software>
5. J. Olsen, "DFEF Technical Design Report," available:  
<http://www-d0online.fnal.gov/www/groups/cft/CTT/online/docs/index.html>
6. J. Olsen, "L1CTOC Technical Design Report," available:  
<http://www-d0online.fnal.gov/www/groups/cft/CTT/online/docs/index.html>