FERMILAB-PUB-06-064-E

Integrated Upstream Parasitic Event Building Architecture for BTeV Level 1 Pixel Trigger System

Jinyuan Wu, M. Wang, E. Gottschalk, D. Christian, X. Li, Z. Shi, V. Pavlicek and G. Cancelo

Abstract— **Contemporary event building approaches use data switches, either homemade or commercial off-the-shelf ones, to merge data from different channels and distribute them among processor nodes. However, in many trigger and DAQ systems, the merging and distributing functions can often be performed in pre-processing stages. By carefully integrating these functions into the upstream pre-processing stages, the events can be built without dedicated switches. In addition to the cost reducing, extra benefits are gain when the event is built early upstream. In this document, an example of the integrated upstream parasitic event building architecture that has been studied for the BTeV level 1 pixel trigger system is described. Several design considerations that experimentalists of other projects might be interested in are also discussed.**

Index Terms—**Trigger, DAQ, Event Building**

I. Introduction

EVENT building is a necessary process in all DAQ systems as well as many trigger systems in high-energy physics experiments. Event building is to merge data of the same event from several different sub-detectors together. When the operation rate of a detector increases, it is often necessary to distribute data of different events to different post-processors. The merging and distributing functions can be performed by data switches. Frequently, however, the merging and distributing can be done parasitically while the data are flowing through the upstream pre-processing systems, resulting in partial or even full event building.

In the Fermilab BTeV experiment [1], pixel detector data are fed into the level 1 trigger system [2][3] to find detached tracks.

The integrated upstream parasitic event building architecture for the BTeV level 1 pixel trigger system was in the process of being adopted as the new baseline when the BTeV project was terminated in Feb. 2005.

The event building function for the BTeV level 1 pixel trigger system is carefully integrated into several pre-processor stages. In each stage, several input channels are merged together and the pre-processing functions are performed. Upon output from each stage, the data are distributed over several output channels based on the beam cross-over (BCO) number of the particular data. The next stage will perform similar merging and distributing functions. After several stages, data from all input channels with a particular BCO number are merged together in a segment tracker. The full event is processed in the segment tracker that allows hits from different planes to be connected together as track segments. Data with different BCO numbers are distributed to different segment trackers so that the full data bandwidth is shared among all of them.

In this article, details of the design as well as several design considerations that may be of interest to other HEP experiments building similar systems are discussed.

II. The BTeV Level 1 Pixel Trigger System

The original 2000 baseline of the BTeV level 1 pixel trigger system can be divided into several stages as shown in the schematic diagram of Fig. 1.

Data from 60 half planes in 30 pixel detector stations are collected by 120 pixel data combiner boards (PDCB) into 960 serial data channels. The 960 channels, each running at 2.5 Gb/s, are organized as 8 "highways" with 120 channels per highway. In normal operation, the PDCB evenly distribute the whole detector data from a sub-set of BCOs to each highway. For example, a highway may contain hits of the whole detector from the collisions in an accelerator turn, and the next highway contains hits from next accelerator turn. The data are sent to the level 1 pixel trigger system via 960 optical fibers [4][5].

In each highway, data from 120 channels, each representing pixel hits from a fraction of the detector plane, must be pre-processed. The data from the detector are asynchronous and unordered such that a hit from a later BCO may be read out earlier. At the first stage, asynchronous data received from multiple pixel data combiner channels are sorted and ordered within a channel by beam crossing number restoring the time order within each channel in the time stamp ordering (TSO) modules. These data are sent to pixel pre-processors (PP) that

Manuscript received June 2, 2005, revised March 22, 2006. This work was supported in part Operated by Universities Research Association Inc. under Contract No. DE-AC02-76CH03000 with the United States Department of Energy.

Jinyuan Wu, M. Wang, E. Gottschalk, D. Christian, Z. Shi, V. Pavlicek and G. Cancelo are with Fermi National Accelerator Laboratory, Batavia, IL 60510 USA (phone: 630-840-8911; fax: 630-840-2950; e-mail: jywu168@fnal.gov).

X. Li is with Illinois Institute of Technology, Chicago, IL, USA.

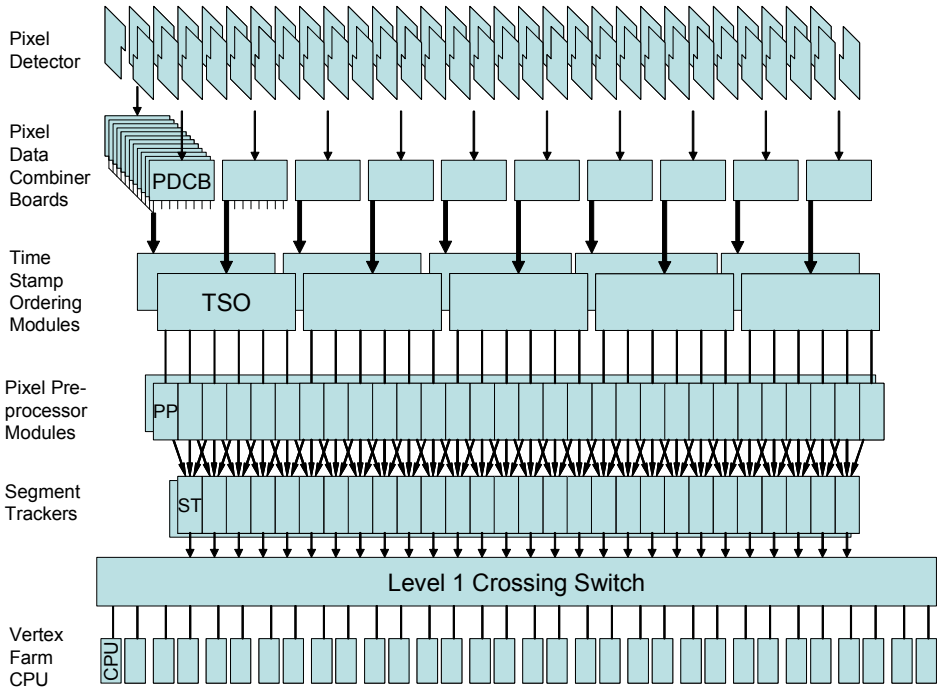

Fig. 1. The original 2000 baseline of the BTeV level 1 pixel trigger system.

find hit clusters and assign xy coordinates to these clusters.

Next, pre-processed coordinate data from three adjacent half planes are sent to one of 56 FPGA (Field Programmable Gate Array) segment trackers (ST) in the following stage that perform the track segment finding phase of the Level 1 trigger algorithm. Note that the data from PP are copied three times here in order to perform the ST function.

To initiate the event building process, the Level 1 switch in the next stage routes packets with identical crossing numbers from 56 different input ports to one of many output ports.

This process is completed once a vertex farm node on the output port receives and assembles all of the data for one crossing thus allowing it to begin the vertex finding phase of the Level 1 trigger algorithm.

It can be seen that the full events needed in the vertex farm nodes are built via the event building switch. We should also mention that in fact, partially built events are needed in the segment trackers already; this is fulfilled with triplication of the links between the PP and the ST stages.

III. The New Architecture

The new 2004 architecture of the BTeV level 1 pixel trigger

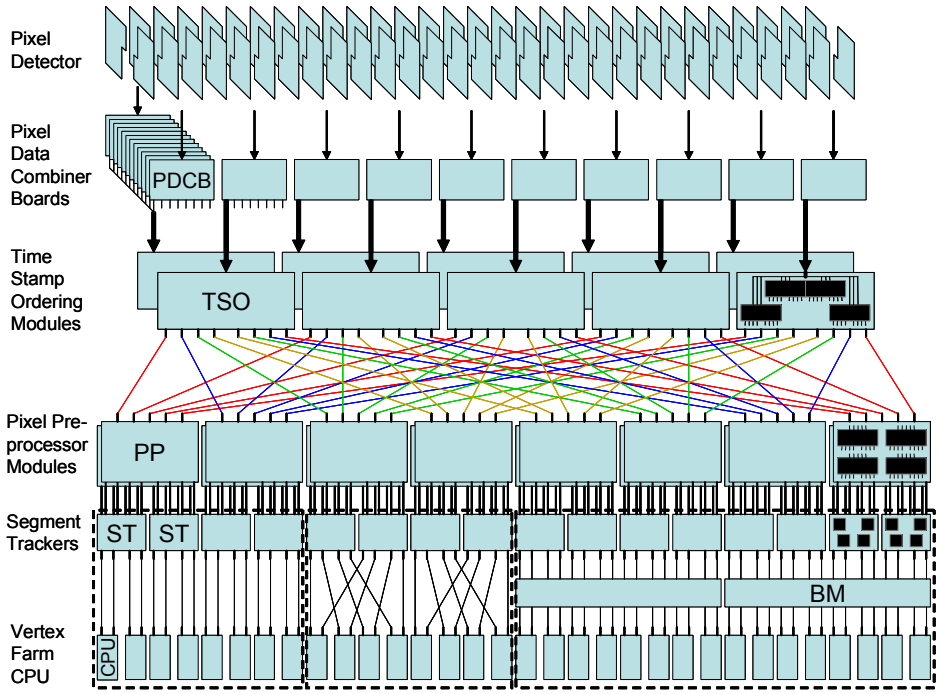

Fig. 2. The new 2004 architecture of the BTeV level 1 pixel trigger system.

system is shown in Fig. 2.

The main feature of this new architecture is that the event building function has been moved up the data path and absorbed into the TSO, PP and ST functions. This architecture eliminates the need for a separate Level 1 switch between the segment trackers and vertex farm CPU nodes in the original baseline. Since the time stamp ordering and pixel pre-processing functions involve operations which sort and order data items by crossing number, they require large amounts of resources like memories and FPGA logic elements. It is natural to implement the additional steps necessary to build complete events along with these functions without significantly increasing the complexity of the hardware. The result is that the data at the outputs of the pixel pre-processors now contain complete data from the full pixel detector instead of mere fragments. Aside from reducing the triplication of interconnections between the PP and ST stages and eliminating the switch between the ST and CPU stages, the new architecture also opens up a myriad of possibilities for other tracking algorithms that need hits from more than 3 adjacent planes in the ST stage.

Naturally there are concerns about the extra complexity in the TSO and the PP stages when the event building functions are combined into these stages. However, through careful planning, the event building functions can actually be absorbed parasitically into the TSO and PP modules.

The design of the TSO and PP modules and their interconnections are described in the following subsections.

A. The Time Stamp Ordering (TSO) Module

The TSO module receives a cable of 12 optical fibers, 2.5Gb/s each. It stores the random hit data according to the BCO number (time stamp) of the hit in temporary memories. After a sufficiently long period of time that the hit data from a BCO are believed to have all arrived, the hit data in a BCO are output together to the pixel pre-processor stage.

To perform the time stamp ordering function, 4 FPGA devices (Altera EP1C6Q240)[6] are use in a TSO module. Data from 3 fibers are handled in one FPGA. Each FPGA is connected to 2 zero bus turnaround (ZBT) synchronous random access memories (SRAM) of size 128K x 32 bits running on a 125 MHz clock. The memories are deep enough to store up to 128 non-empty BCO buckets, which are more than an accelerator turn worth of data. Each FPGA outputs data to 8 differential pairs at a data rate of 375 Mb/s per pair. Each differential pair is routed via the backplane to a PP module. Under normal operation, data from 3 fiber channels in a non-empty BCO are sent to a pre-defined PP module with rotational order.

The data rate of the differential pairs is chosen to be as low as possible to simplify the design of the interconnection and the receivers in the later stages. The reader may note that the total output bandwidth of the TSO module is smaller than the input bandwidth. The justifications are: (1) that due to the 8B/10B conversion, the pay-load data rate is only 80% in the 2.5 Gb/s optical fiber input channel, (2) that the TSO operation is a natural lossless compression process since only one BCO number is needed to be attached to a set of detector hits in the output stream and (3) that the average data input rate is only 12-30% of the input capacity according to the simulation.

Note that parasitic event building is performed here, although barely noticeable. The data merging and redistributing in the TSO module build a partial event of a "sub-detector" with 3 fiber channels.

TABLE I
TIME STAMP ORDERING AND PIXEL PRE-PROCESSOR MODULES

Time Stamp Ordering (TSO) Module	
Number of modules	10/highway, 80 full system
Inputs of module	12 fibers, 2.5 Gb/s per fiber
FPGA on module	4 (EP1C6Q240)
Inputs of each FPGA	3x 16 bits @ 125 MHz
Memories connected to FPGA	2x 32 bits x 128K @ 125 MHz
Outputs of each FPGA	8 differential pairs, 375 Mb/s each One pair to each PP
Outputs of module	32 pairs, 4 pairs to each PP
Pixel Pre-processor (PP) Module	
Number of modules	16/highway, 128 full system
Inputs of module	20 diff. pairs from 5 TSO (40 pairs are actually designed)
FPGA on module	4 (EP1C6Q240)
Inputs of each FPGA	5 pairs, 375 Mb/s (10 pairs are actually designed)
Memories connected to FPGA	2x 32 bits x 128K @ 125 MHz
Outputs of each FPGA	8 differential pairs, 375 Mb/s each
Outputs of module	8 RJ-45 connectors, 4 pairs/connector

Parameters of the TSO and PP modules are listed in Table I.

B. The Pixel Pre-processor (PP) Module

The PP module receives data from the TSO stage via the backplane. Under normal operation, each PP module is connected to 5 TSO modules with 20 input pairs at 375 Mb/s per pair. The PP module is actually designed with 40 pairs connecting to 10 TSO modules. The extra interconnections allow the user to flexibly rescale the system, which will be discussed later.

Each PP module uses 4 FPGA devices. Each device has 10 differential pair inputs from 10 TSO modules (5 in normal operation mode). The data are stored in the ZBT SRAM chips connected to the FPGA. The fired hits of adjacent pixels are grouped together to form a cluster, and the weighted center of the cluster is calculated as the estimate of the position of the charged particle track hit. The coordinates of the cluster center are output to the ST stage to reconstruct the possible track segments.

The outputs from a PP module are organized as 8 front panel RJ-45 connectors each carrying 4 differential pairs at 375 Mb/s per pair. Each PP FPGA outputs 8 pairs, one pair to each connector. The data from a non-empty BCO are routed to an output connector in pre-defined rotational order.

Partial event building again happens in this merging-distributing process. In each output connector differential

pair, data with the same BCO come from a PP FPGA that merges data from 5 TSO modules. Therefore, in the 4 differential pairs of an output connector, data from 4x5x3=60 fiber channels are merged together, which represent half of the pixel detector.

C. *Interconnections between the TSO and the PP Stages*

The TSO and the PP modules are designed in VMEbus 6Ux220mm format. Due to the similarity of the two modules, it is possible to combine them onto a single printed circuit board design. The two modules are assembled by installing different front panel connectors and related components as shown in Fig. 3(a).

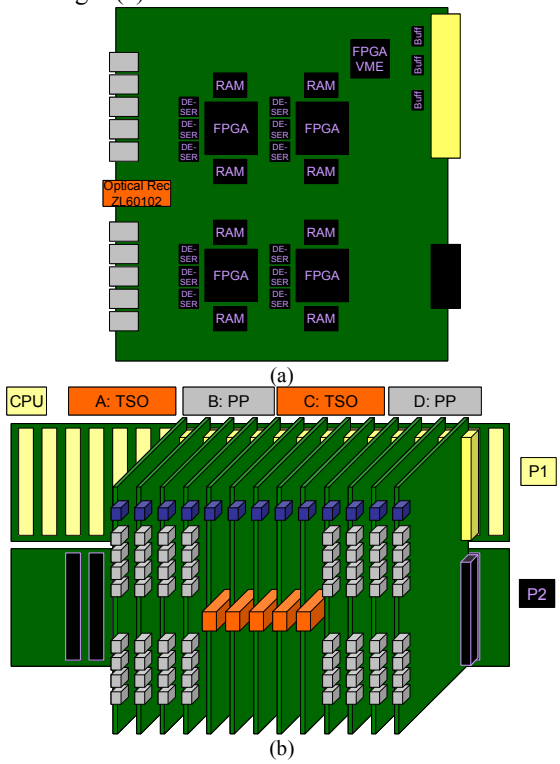

Fig. 3. (a) The printed circuit board for the time stamp ordering module and the pixel pre-processor module. (b) The subsystem.

The subsystem is implemented in a 21-slot 6U VMEbus crate with sections of slots dedicated to the TSO and PP modules as shown in Fig. 3(b). The VMEbus P1 is used for control, setup and debug functions while the P2 backplane carries differential signals interconnecting the TSO and PP modules.

Between the 5 TSO modules in section C and the 4 PP modules to the right in D, there is a 4-out 5-in multi-star routing in the P2 backplane. The multi-star routing is repeated for 4 FPGA devices each in the TSO and PP modules. The same pattern interconnects the TSO modules in section A and the PP modules in B. The mirrored pattern interconnects to TSO modules in C and the PP modules in B.

The P2 plane routing has been studied and can be done in a 10-layer printed circuit boards with 4 ground planes and 6 inner layers organized as 3 differential pairs.

Two crates are used for each highway to house 10 TSO and 16 PP modules.

D. *Segment Tracker (ST) and Vertex Farm Stages*

The 128 cables output from the PP modules are fed to 64 FPGA segment trackers. Each ST receives 2 cables, one from each crate that contains half detector information. The data from the 2 cables, 8 differential pair channels total, are merged together in the ST FPGA to complete the full event building. In Fig. 2, 4 FPGA segment trackers are packed on each ST module and there are 16 ST modules per highway.

Track segments are identified in the ST. Unlike the original architecture where an ST sees the hits of 3 planes from all BCOs, the new segment tracker receives all the hits of the entire detector from some BCOs. Therefore, some processes that could only be performed with the vertex farm CPU in the old scheme, such as matching track segments from different detector plane sets and finding possible vertices, etc. can now be done in the ST with fast firmware, as long as the FPGA resources allow.

The track segments found in this stage are output to the vertex farm nodes. Since the events are fully built, a switch is no longer needed. The ST can directly send data to the CPU as shown in the lower left dash box in Fig. 2. In the real implementation, the ST and CPU are interconnected together to provide "over-switching" using either passive cables or active modules, called "buffer managers" (BM), as shown in the lower middle and lower right dashed boxes in Fig. 2.

The segment tracker/vertex farm subsystem is the topic of another document.

E. *System Rescaling*

The possibility of partial system operation is a very useful feature for error tolerant operation and staged system commissioning. With upstream event building shown in Fig. 2, it is possible to operate the trigger system at a lower beam rate with as few as one PP module, one ST module and one vertex farm node. In the original system shown in Fig. 1, however, all TSO, PP and ST modules must function correctly so that the hit information from a part of the detector will not be missing.

In various stages of the system commissioning, it is possible to operate the system with the following configurations:

- In the very early stage, the users can install 10 TSO modules in sections A and C shown in Fig. 3 while keeping only 1-4 PP modules in B. Every PP module is fed by all of the 10 TSO modules so its outputs contain hits of the whole detector from all 120 input optical fibers.
- In the stage with medium operation rate, all of the 8 PP modules in B and D are inserted that doubles the operating rate.
- In normal operation, two crates are used per highway, with 10 TSO and 16 PP modules to support full operation rate.

F. *Orthogonal Interconnection*

Another option of interconnections between the TSO and the PP modules has also been studied. In this case, the 10

TSO modules and 16 PP modules are mated orthogonally as shown in Fig. 4. Each PP module receives data from all 10 TSO modules and thus builds events from the full detector. This design provides relatively even electrical properties among different interconnections.

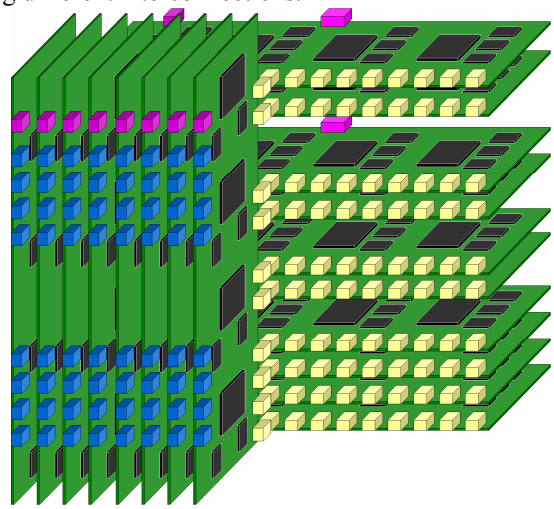

Fig. 4. Orthogonal interconnections between the time stamp ordering (TSO) and the pixel pre-processor (PP) modules: The TSO modules are inserted from back horizontally and the PP modules mate them vertically.

In Fig. 4, the interconnections are drawn as individual blocks to show the logic relationship, but the actual connectors are 3-row x 32-pin (as used in VMEbus) and 3x16 ones interconnected through a backplane as shown in Fig. 5.

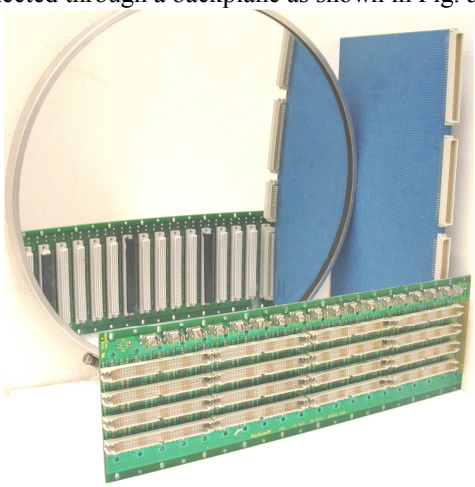

Fig. 5. Photograph of the model vertical card and backplane: The part of the backplane shown provides interconnections between horizontal slots 14-17 with the vertical connector P106.

The backplane shown is constructed using standard 3x32 connectors with 13 mm (.512 in) straight leads. The leads pushed through are trimmed except 4 groups of predefined 3x3 pins in each connector. Standard 3x32 shrouds are installed horizontally to provide insertion alignment for the horizontal cards.

The vertical cards, i.e., the PP modules are in 9U format with VMEbus P1 and P3 connectors at the standard location. We define 2.54 mm (0.1 in) pin grid from top to bottom with the pin 1 of P1 being the pin grid 1. Since the pin 1 of the original VMEbus P3 is at pin grid 106, it is renamed as P106 and we will use this naming scheme throughout the rest of this paper. Another 3x32 connector P42 and a 3x16 connector P82 are placed with pin 1 at pin grid 42 and 82, respectively. Note that P42 is not at the VMEbus P2 location so that the module can never be mistakenly inserted into standard VMEbus crates.

The horizontal slots are defined with 3 rows of pins in a slot with slot pitch of 0.8 in, the same as the vertical slot pitch, so that the card guide rail hardware can be reused. The origin of the slot is chosen so that each 3x32 connector in the vertical card mates with 4 horizontal slots and the 3x16 connector mates 2 as shown in Table II.

TABLE II
ALIGNMENT BETWEEN THE HORIZONTAL SLOTS AND THE VERTICAL CARD CONNECTORS

H. Slots	Pin Grid of Row			V. Connectors			
	C	B	A		Mating Pins		
1	4	5	6	P1	4	5	6
2	12	13	14		12	13	14
3	20	21	22		20	21	22
4	28	29	30		28	29	30
5	36	37	38	Unused			
6	44	45	46	P42	3	4	5
7	52	53	54		11	12	13
8	60	61	62		19	20	21
9	68	69	70		27	28	29
10	76	77	78	Unused			
11	84	85	86	P82	3	4	5
12	92	93	94		11	12	13
13	100	101	102	Unused			
14	108	109	110	P106	3	4	5
15	116	117	118		11	12	13
16	124	125	126		19	20	21
17	132	133	134		27	28	29

Up to 14 horizontal cards can be connected with the 9U vertical cards in this design. In our application, however, only 10 TSO modules are needed, so only slot 6-9, 11-12 and 14-17 are used. The P1 of the vertical cards can be used for regular VMEbus traffic for system control and monitoring functions.

TABLE III
ALIGNMENT BETWEEN THE VERTICAL SLOTS AND THE HORIZONTAL CARD CONNECTORS

V. Slots	Pin Grid of Col			H. Connectors			
	C	B	A		Mating Pins		
20	1	2	3	P1	1	2	3
19	9	10	11		9	10	11
18	17	18	19		17	18	19
17	25	26	27		25	26	27
16	33	34	35	Unused			
15	41	42	43	P39	3	4	5
14	49	50	51		11	12	13
13	57	58	59		19	20	21
12	65	66	67		27	28	29
11	73	74	75	Unused			
10	81	82	83	P78	4	5	6
9	89	90	91		12	13	14
8	97	98	99		20	21	22
7	105	106	107		28	29	30
6	113	114	115	Unused			
5	121	122	123	P116	6	7	8
4	129	130	131		14	15	16
3	137	138	139		22	23	24
2	145	146	147		30	31	32
1	153	154	155	Unused			

The horizontal cards, i.e., the TSO modules are designed similarly as shown in Table III. Each horizontal card has 4

3x32 connectors: P1, P39, P78 and P116 with pin 1 of the connector placed at pin grid 1, 39, 78 and 116, respectively. Note that the horizontal pin grid runs from slot 20 to slot 1.

The interconnections between the horizontal and the vertical cards consist of 3x3 pins that are just enough to carry 4 differential pairs and a common mode return (ground). Note that the differential data signals are fed through the connector pins directly from the TSO to the PP modules, so there are no traces needed on the backplane carrying them.

Orthogonal interconnection scheme is a very attractive option for event building electronics; however, extra work must be done to understand cooling issues for the horizontal cards.

IV. System Design Considerations

Many details have been documented in previous section to help explain system design considerations that experimentalists of other projects might be interested in.

A. Under-switching

The silicon resource usage of switching functions that meet our flexibility requirements is $O(N^2)$, where N is the number of channels to be switched. Therefore it is desirable to split the full event building functions into as many stages as reasonable and let each stage perform a small switching function along with the pre-processing function.

In order to merge 120 fiber channels together in our design, we let the TSO stage merge only 3 channels and the PP stage merge 5. The last factor of 8 merging is in the segment tracker stage.

The segment tracker stage is the stage where the full events are used. However, its input is "under-switched", i.e., it contains 8 data streams that are still to be merged. This merging is not a problem for ST since data buffering are to be done for segment finding processes anyway. However, if this factor of 8 is pushed to the earlier TSO and PP stages, then the TSO and PP stages must perform merging functions of even larger scales such as factors of 12 and 10 each. The purpose of "under-switching" is to minimize impacts on the pre-processing stages.

B. Over-switching

The second consideration can be called "over-switching". Event building is not the only purpose of data switching. In HEP trigger and DAQ systems, data are also switched for load balancing, fault tolerance etc.

In our example shown in Fig. 2, full events are built in the segment tracker and no further event building is necessary after this stage. However, small extra switching abilities between the ST and the CPU stages are still provided. The simple passive cable interconnection shown in the lower middle dashed box in Fig. 2 allows 4 CPU nodes to share the same set of data. Another possibility is to use a set of small interconnection devices buffer managers as shown in lower right dashed box in Fig. 2 that allows data from any of 4 segment tracker modules to be sent to any of 8 CPU nodes.

The buffer manager stage is not used for event building purposes, but the "over-switching" it provides is essential for fault tolerance when dealing with CPU farm node failures, which are likely to happen in large farms. Even under normal operation, a CPU can occasionally run a process into extraordinarily long loop. In this case, other CPU nodes sharing the same load help average out the tail effects.

V. Discussion

We have described how to integrate the event building functions into the pre-processing stages in the BTeV trigger system. Disregarding the specific pre-processor function, each stage can be viewed as a set of switch fabrics each with a few input channels and a few output channels. The entire data-combiner/pre-processor/segment tracker system has a topological structure similar to a multi-stage data switch. However, there is no dedicated switch; the switching functions are parasitically spread over the pre-processor system occupying minimal logic resources in each stage. Since only a few channels are to be merged in each "fabric", and some resources such as serial-to-parallel conversion, memory buffers, I/O pins, etc. are already part of the design of the pre-processing functions, the additional FPGA resource required for the switching function is not large.

Despite elimination of a dedicated switch, the integrated upstream parasitic event building architecture still provides the same benefits as a conventional switch. In fact, since the switching function is moved to the earlier stages, it allows more flexibility with trigger algorithms, more ability to easily rescale hardware resources in response to changing running conditions, and more ability to route data around failed hardware components. For example, with upstream event building each segment tracker receives data from all detector planes; a fault encountered by a segment tracker does not halt the entire system. Instead it will only need to reroute data for some BCO numbers to different segment trackers. At worst, some crossings might be dropped from processing until the fault is resolved.

This paper does not really propose any new cutting edge technology but rather a carefully chosen combination of existing design practices. With this design, less hardware usage is anticipated that results in not only cost saving, but also better and more reliable performance as mentioned above.

References

[1] Kulyavtsev et al., BTeV proposal, Fermilab, May 2000, BTeV-doc-66.

[2] M. Wang, BTeV Level 1 Vertex Trigger Algorithm, BTeV-doc-1179.

[3] E.E. Gottschalk, BTeV detached vertex trigger, Nucl. Instrum. Meth. A 473 (2001) 167.

[4] M. Wang et al., "A Commodity Solution Based High Data Rate Asynchronous Trigger System for Hadron Collider Experiments", Proceedings of the IEEE Real Time Conference 2005, Stockholm, Sweden, June 2005, to be published.

[5] M. Votava et al., "BTeV Trigger/DAQ Innovations", Proceedings of the IEEE Real Time Conference 2005, Stockholm, Sweden, June 2005, to be published.

[6] Altera Corporation, "Cyclone FPGA Family Data Sheet", (2003)