DØ/TRIGGER/XXXX
Version 0.0
September 16, 1999

Specification of the Level 2 Central Tracking Trigger Preprocessor Crate with inputs from the Level 2 Silicon Tracking Trigger

Drew Baden and Dave Toback

University of Maryland

Abstract

We present a technical design report of the Level 2 Central Tracking Trigger (L2CTT) preprocessor crate. This document describes the scenario in which the Level 2 Silicon Tracking Trigger (L2STT) is the source of the input tracks to the L2CTT. It includes a full specification of the inputs and outputs of the crate as well the algorithms used for processing.

1 Introduction

This note is the technical design report for the Level 2 (L2) Central Tracking Trigger (L2CTT) preprocessor. It describes the scenario in which the L2CTT takes in input from the Level 2 Silicon Tracking Trigger (L2STT). It is designed to be complementary to DØNote 3645 [1] which describes the scenario in which the L2CTT receives input from the Level 1 Central Fiber Tracker (L1CFT). It includes a full specification of the inputs and outputs of the crate as well the algorithms used for processing. While this document is designed to be stand-alone from a L2CTT functionality standpoint, it only summarizes the functioning of the general L2 preprocessor crate where relevant to the L2CTT and leaves the details to the references [2, 4].

1.1 Overview

An overview of the L1 and L2 trigger hardware is shown in Figure 1. Within this system, L2 is being designed to handle an average input rate of about 10 kHz, provide a rejection factor of approximately 10, yield a signal efficiency no lower than Run I Level 3 (L3), introduce at most 5% dead-time, and send information to the L3 trigger to assist the software filter decisions.

L2 is implemented mostly using a number of "standard crates", one of which is the centralized decision making crate, the L2 global (L2G), and a number of 'preprocessor' crates, one of which is the L2CTT. In addition to the standard crates, the Silicon Tracking Trigger (L2STT) preprocessor is implemented custom boards. Simulations estimate that the mean processing plus output formatting time for each preprocessor must be less than 50 μsec. The general software and hardware for the L2G and the preprocessors are summarized in the main part of this document and described in detail in [2]. The L2STT is described in more detail in Ref. [3].

The L2CTT will operate in two different scenarios: 1) at the beginning of the run with input tracks from the L1CFT (described in Ref. [**?**])and 2) later in the run with input tracks from the L2STT (which takes in track information from the L1CFT). With direct input from the L2STT the L2CTT performs the following tasks:

- Reads in the track lists from the twelve different ϕ regions of the L2STT.
- Converts the track information into L2 objects for use in L2G ($\mathrm{P_T}$, ϕ_0, ϕ_{EM3}, dE/dX and impact parameter b).
- Removes duplicate tracks where possible.
- Concatenates the track lists into a single list and sorts it by $\mathrm{P_T}$.
- Concatenates the track lists into a second list sorted by track impact parameter.
- Sends the ordered lists of L2 objects to the L2G.
- Maintains L2 Objects for L3 readout on a L2 Accept (L2A) or for min-bias and monitoring.

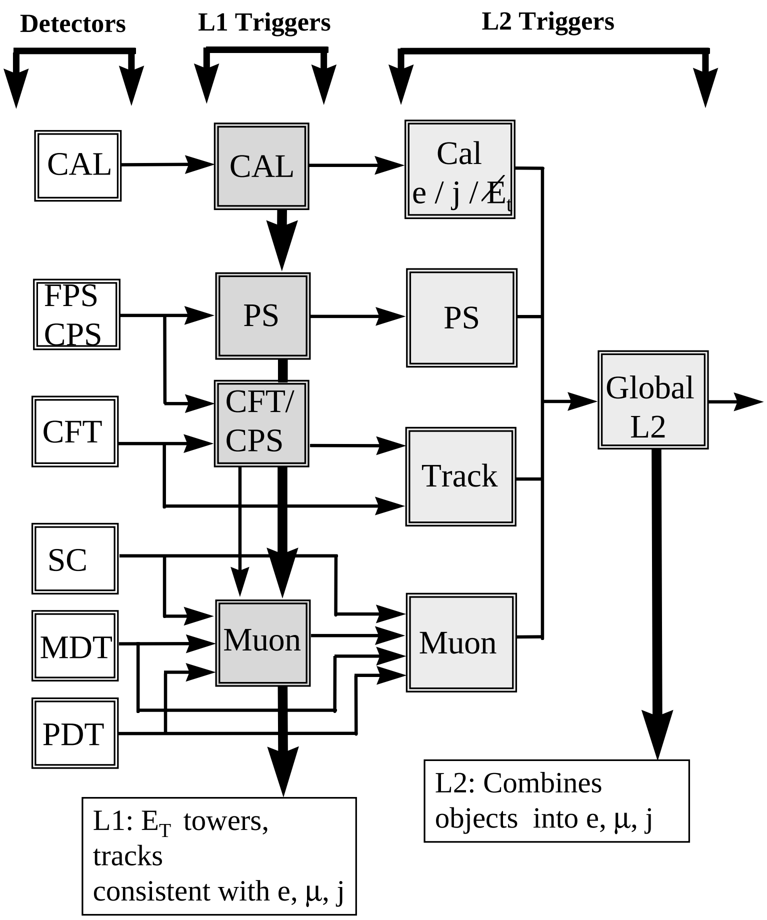

Figure 1: Overview of the L1/L2 hardware trigger system. Note that in this over simplified picture the the "Track" section includes the STT to CTT.

2 Functional Description of the L1 and L2 Tracking Trigger Systems

2.1 The CFT and The L1 Trigger System

The L1CFT tracking trigger performs the following tasks:

- reads out the tracking hits from the CFT and does the track finding.
- reads reads out preshower hits from the Central Preshower (CPS) detector and does cluster finding.
- matches tracks with CPS clusters.
- stores the tracks and CPS clusters for use in the event of a L1 Accept (L1A) and a L2A.
- In the event of a L1A, the track lists are read out to L2. In the event of of a L2A, the track lists are read out to L3.

A few notes about CFT terms are in order. In the ϕ coordinate, the fiber tracker system is sliced into 80 sections, called sectors, each of which subtends 4.5 degrees. The inner (A layer) and outer (H layer) layers are spanned by 16 and 44 fibers per sector respectively[1]

During normal operation the trigger begins by reading in the raw fiber hit information from the CFT using the Front End (FE) boards (1 FE board = 1 sector). All tracks with $\mathrm{P_T} > 1.5$ GeV which cross the outer detector layer ($|\eta_{\mathrm{Detector}}| < 1.5$) are completely contained within one sector (the home sector) or are partly in one of the adjacent sectors (neighbor sectors). To form a seamless trigger, all outer layer hits from the home sector are combined with the hit information from the two neighboring sectors. Track finding uses only the axial layers and is implemented by comparing hit patterns to pre-programmed possible track roads with $\mathrm{P_T} > 1.5$ GeV. In parallel, the CPS does its cluster finding. After track and cluster finding matching is done between each track and cluster candidate [4].

After track finding, each track candidate is referenced using a 32-bit word (including $\mathrm{P_T}$ and ϕ at the H layer as well as the cluster matching information) and sorted into one of four $\mathrm{P_T}$ bins. The highest $\mathrm{P_T}$ bin, Bin 0, is such that $\mathrm{P_T^{Bin0}} > 10$ GeV. Similarly, Bins 1 through 3 are such that 5 GeV$\leq \mathrm{P_T^{Bin\ 1}} \leq 10$ GeV, 3 GeV$\leq \mathrm{P_T^{Bin\ 2}} \leq 5$ GeV and 1.5 GeV$\leq \mathrm{P_T^{Bin\ 3}} \leq$ 3 GeV. See Appendix D for more details on the meaning of the $\mathrm{P_T}$ words.

After binning, the tracks are read out of the front ends into Sextent boards for readout to the L2STT. Each Sextent board reads in tracks with the h-layer fiber number within the sextent. However, tracks which might be from a neighboring sextent at the radius of the SMT must be sent. This is done by sending tracks from FE boards in the 'overlap' region to the Sextent boards [?]. This is implemented using six OVerlayed Region (OV) boards

[1] We note for completeness that the CFT consists of 32 concentric barrel-shaped layers of scintillating fibers arranged in 16 "doublet" layers. The fibers in each layer are at the same radial distance from the z-axis and are such that half of the layers have fibers parallel to the z-axis and half are at an angle to the z-axis to provide stereo tracking reconstruction in L3 and offline.

which decide which additional tracks from the 3 neighboring sectors (using only those with the appropriate sign) are sent. We note that this implies that since there will be duplicate tracks sent to different Sextent boards, there will be duplicate tracks in the L2STT.

A maximum of 48 tracks (6 max per P_T bin for both positive and negative tracks) are sent from each sector to the OV and Sextent boards. After track selection, each OV board can pass a maximum of 48 tracks, selected by P_T bin order (regardless of sign), but not by P_T within the bin, to the six Sextent boards. The Sextent board, which takes in tracks directly from the FE's and the two OV boards, then chooses which tracks to send to L2STT. The Sextent board begins by reading in all the tracks from Bin 0 (regardless of sign). If less than 46 tracks are read in the algorithm reads in tracks from Bin 1 until either 46 tracks are reached or all tracks are entered into the output list. This continues for the rest of the P_T thresholds (Bins 2 and 3) until either all tracks are read in or all 42 track slots are used. The list of tracks, along with corresponding header and trailer information, from the Sextent board to the L2STT via G-links as 16-bit frames at 53 Mhz (1 Gbit/sec).

We note for completeness that there is an inherent ϕ bias in this procedure as tracks with large ϕ could be lost. Similarly, it means that low P_T tracks could be lost. However, this has been studied and the current belief that the bias is negligible [4].

2.2 Level 2 Silicon Tracking Trigger

At Level 2, the tracks are read into the L2STT and processed. For each track the CFT information is associated with data from the Silicon Microstrip Tracker (SMT) and, assuming associated clusters are found in the SMT, a more precise measurement of the track P_T and impact parameter is made. These updated track objects are passed on to the L2CTT. The L2STT performs the following tasks:

- Reads in the track lists from the CFT Sextent boards.
- Uses the ϕ position at the H layer and the P_T word reported by L1CFT to determine a road in the SMT to look for clusters.
- Creates L2STT track objects by associating SMT clusters with CFT tracks and re-fitting. If no clusters are associated with the track, the L2STT object is created with only the original CFT information. Re-fit tracks contain P_T, ϕ_0, χ^2, dE/dX information and impact parameter (b) information.
- Sends all L2STT track objects to the L2CTT along 12 cables.
- Stores information for readout on L2A or for min-bias and monitoring.

2.3 Level 2 Central Tracking Trigger

After the L2STT processing, the tracks are read out of the L2STT and into the L2CTT, where they are processed into final format and passed on to the L2G. In L2G, the tracks are associated with muon, electron and tau candidate information from the rest of the detector and used to make a final L2 decision. The L2CTT performs the following tasks:

- Reads in the track lists from the twelve L2STT output boards.
- Uses the ϕ_0 and the $\mathrm{P_T}$ to determine ϕ_{EM3}which is the ϕ, in global coordinates, at showermax in the EM calorimeter (EM3).
- if there is no $\mathrm{P_T}$ information from the L2STT, it converts the ϕ at the H layer and the $\mathrm{P_T}$ word from the CFT into $\mathrm{P_T}$, and ϕ_0 and ϕ_{EM3}.
- Concatenates the 12 lists of tracks into a single list which is then sorted by $\mathrm{P_T}$.
- Concatenates the 12 lists of tracks into a single list which is then sorted by impact parameter.
- Removes duplicate tracks in the case where both tracks have no SMT matching information.
- Converts the information for each track into L2 track objects.
- Sends out all tracks to L2G sorted by $\mathrm{P_T}$.
- Sends out all tracks to L2G sorted by impact parameter.
- Stores information for readout on L2A or for min-bias and monitoring.

To enhance performance, new events from L2CTT are allowed to flow into buffers in the crate while processing on previous events takes place. The remainder of this document describes the L2CTT.

3 CTT Preprocessor Crate Hardware

The L2CTT is implemented using the components of a standard L2 crate [2]. A schematic of the components in the crate is shown in Figure 2. For a more specific listing of which board is in which slot in the crate see Table 1. We quickly describe the components of the L2CTT:

- Magic Bus Transceiver card (MBT, see Ref. [6]): This board provides the I/O between the L2CTT and the L2STT, the L2G and the Serial Command Link (SCL) Hub [7]. Each MBT allows input from 8 sources. There are two MBT's in the L2CTT crate. The Pilot MBT (in slot 19) is to communicate with the SCL and take in 6 input cables from the L2STT. An additional "assistant" MBT reads in the remaining 6 L2STT inputs and is controlled by the Pilot MBT.
- Magic Bus (MBus, see Ref. [8]): This is a custom 16-byte wide asynchronous bus which resides as the J3 backplane and has 32-bit addressing. It is used to pass information between the MBT and the Administrator and Worker Alphas.

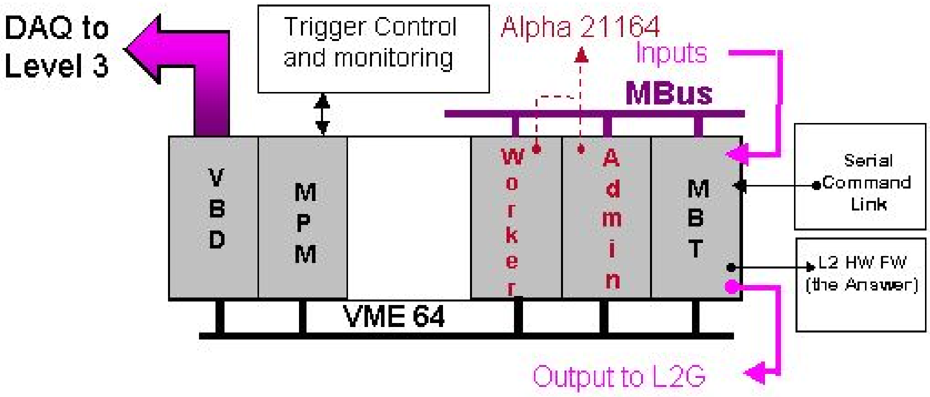

Figure 2: This is schematic drawing of the L2CTT crate. For a more specific mapping of which board goes in which slot see Table 1.

- Worker and Administrator (see Ref. [9]): These are implemented using a general purpose computer (the CPU is a 500 MHz DEC Alpha 21164 on a custom motherboard based on the PC164 design) with VME and Mbus connections. The hardware for both 'Alpha' boards is identical, only software distinguishes the two. The Administrator does the housekeeping tasks (i.e, manages the input and output event buffers, coordinates L3 readout and sends monitoring information to the Trigger Control Computer (TCC)) while the Worker does the track processing. The Worker Alpha resides in slot 7, the Administrator in slot 20.

- Multi-Port Memory (Bit3, see Ref. [10]): This I/O card is used to do the VME controlling of the crate. In addition, the board provides the interface between the TCC and the rest of the crate. The TCC handles run control, downloads run-specific information and collects monitoring information. It resides in slot 1.

- VME Buffer/Driver (VBD, see Ref. [11]): This board provides the readout to L3 and resides in slot 2.

Slot	Card
1	Bit3 Multi-port Memory (Bit3)
2	VBD (readout to L3)
3-6	Spare
7-8	L2 Worker Alpha
9-17	Spare
18	Assistant Magic Bus Transceiver (MBT) Card
19	Pilot Magic Bus Transceiver (MBT) Card
20-21	L2 Administrator Alpha

Table 1: The cards in the L2CTT crate.

4 Inputs to L2CTT

The L2CTT takes inputs from multiple sources. The primary inputs are from:

- L2STT
- TCC
- SCL

4.1 Non-L2STT Inputs

The communication with all inputs other than the L2STT are described in more detail in Refs. [2], [6] and references therein. To summarize, the TCC interactions use the Bit3; the SCL interactions are made through the MBT. See Appendix B for more details about the SCL connections and the information passed.

4.2 Inputs from L2STT

The data transfer between the twelve L2STT boards and the L2CTT is implemented using Cypress Hotlink[2] at a max of 16MBytes/sec. This is read directly into the MBT. The link activity is as follows: Idle...Data-for-one-event...Idle...Data-for-one-event... where Idle is a Hotlink defined state in which no data is sent and pad characters are sent to fill the entire time between data transmissions. The Hotlink protocol is such that the transmitter and receiver are able to decide when data is sent, which means that no information is needed from the TCC for the data to be sent from the L2STT to the MBT. After all the data has been sent, the L2STT transmitter must pad the Data-for-one-event to be a multiple of 4 longwords (4×32 bits = 128 bits). Once all data from all 12 cables are read in, and the Administrator gives the MBT a "ready to start," the MBT puts information from 1 event on the Mbus backplane which is read in by the Worker and put into the input Buffer.

The data passed to the worker from each STT input cable is shown in Table 9 in terms of single longword frames. Each event consists of header information, followed by all track words, then a trailer. The track data is sent such that all CFT tracks are sent before any STT information is sent. Specifically, each event consists of 4 header longwords followed by 1 longword per track containing the CFT information, then 2 additional long words per track containing the STT information (assuming there is associated STT track information[3]). Finally there is one End-of-Record longword. Taking into account the 5 header/trailer words and assuming each CFT track has associated STT information, we find a maximum of 42 tracks $\times$ 3 words/track + 5 header/trailer words $\rightarrow (3 \times 42) + 5 = 131$ longwords. The maximum number of bytes per event is determined using the fact that a maximum of 252 tracks can be sent from L1CFT to L2STT (42 tracks max for each of 6 Sextent boards). A maximum of 816 longwords or 3,264 bytes of data (252 tracks $\times$ 3 words/track + 5

[2]We use the Hotlink protocol running over Unshielded Twisted Pair (UTP). Here Cypress is the company name. The data transfer is using 10-bit frames, 8 bits of which contain data, and 2 bits of which contain control characters. It is these control characters which indicate that data is going to be sent from the L2STT.

[3]there is an index number in the STT info which corresponds to the CFT track number.

Header/trailer words × 12 cables) is passed for each event. We again note that each event is padded (using a repeated End-of-Record) to be a multiple of four longwords.

Header Longword 1
Header Longword 2
Header Longword 3
Header Longword 4
Data Longword 0 (CFT Track 0)
⋮
Data Longword N (CFT Track N)
Data Longword N+1 (STT Track 0, word 0, Associated with CFT Track i)
Data Longword N+2 (STT Track 0, word 1, Associated with CFT Track i)
Data Longword N+2M (STT Track M, word 1, Associated with CFT Track j)
End-of-Record Longword

Table 2: The data longwords sent from the L2STT to the L2CTT. Note that in this example there are N tracks, where N is an integer less than or equal to 42, and M (where M≤N) of these tracks have associated STT information. CFT tracks have one longword per track and all CFT tracks are sent before any STT information is sent. For each CFT track which has associated STT information, there are two long words per track for the STT information. Since the STT track information is not sent in the same order as the CFT track information, each STT track word-pair contains a pointer to its associated CFT track. Note that we have ignored the extra End-of-Record longwords which will be sent to pad the event such that the total data sent is a multiple of 4 longwords.

The bit construction of the header and trailers information sent from the L2STT is given in Tables **??** and **??**. The format of the first three headers and the trailer is identical to that sent to L2G with a few exceptions which are specified in Table **??**. Header 1 contains overview information about the event including the number of tracks and headers sent. Header 2 contains information about which event is being studied. Header 3 contains processing information including status bits (defined in Table 7). Header 4 indicates the number of CFT tracks which have associated STT track information.

For the track data, the bit construction of the CFT and associated STT track information is as specified in Tables **??** and **??**. Within the CFT track word bits 0-4 indicate that if there is a preshower match, where, relative to the track position in phi, the preshower match is (bits 8 and 9 indicate if there is a match). This information is not used in the L2CTT, but is passed on the the L2G for completeness. Bits 5-7 contain some error status information which is described in more detail in Table **??**. Bits 8 and 9 tell whether or not the track is matched to a preshower cluster with 8 being high threshold match, and bit 9 being a low threshold match. Bit 15 contains the sign of the charged particle. Bits 10-14 contain the five bits of the absolute value of P_T. Bits 16-22 contain the FE board number (or sector number) which varies from 0-79. Bit 23 contains a bit which indicates whether or not the track has been sent twice to the L2STT. Bits 24 and 25 contain track isolation information which is not used by the L2CTT but is passed along to the L2G. Bits 26-31 contain the H layer fiber number within the FE sector which varies from 0-43.

There are two data words which contain the STT information which can be associated with each CFT track. 6 bits of χ^2, 11 bits of P_T(Pt is 0.25 GeV/bit -¿ -256 GeV ¡ pt ¡ 256 GeV). 8 bits of Phi0), 8 bits of ϕ_0, 15 bits of b and 2 bits of dE/dX. 6 bits of CFT track track number, 3 bits of Barrel number, 4 bits of error per track (0000 Everything is fine, if bit i is set then there is a hit count overflow in hit layer i).

```
Word 0:

Bits  0-10: Pt
Bits 11-15: Spare
Bits 16-23: Phi0
Bits 24-29: Chi^2
Bits 30-31: dE/dX

Word 1:

Bits  0-14: Impact Parameter
Bit     15: Spare
Bits 16-21: CFT Track Number
Bits 22-23: Spare
Bits 24-27: Error Bits
Bits 28-30: Barrel Number
Bit     31: Spare
```

bit #	Header 1	Header 2	Header 3
31	Header Format(2)	Rotation #(15)	Status Bits(7)
30	Header Format(1)	Rotation #(14)	Status Bits(6)
29	Header Format(0)	Rotation #(13)	Status Bits(5)
28	Object Format(4)	Rotation #(12)	Status Bits(4)
27	Object Format(3)	Rotation #(11)	Status Bits(3)
26	Object Format(2)	Rotation #(10)	Status Bits(2)
25	Object Format(1)	Rotation #(9)	Status Bits(1)
24	Object Format(0)	Rotation #(8)	Status Bits(0)
23	Object Length(7)	Rotation #(7)	Processing Bits(7)
22	Object Length(6)	Rotation #(6)	Processing Bits(6)
21	Object Length(5)	Rotation #(5)	Processing Bits(5)
20	Object Length(4)	Rotation #(4)	Processing Bits(4)
19	Object Length(3)	Rotation #(3)	Processing Bits(3)
18	Object Length(2)	Rotation #(2)	Processing Bits(2)
17	Object Length(1)	Rotation #(1)	Processing Bits(1)
16	Object Length(0)	Rotation #(0)	Processing Bits(0)
15	Header Length(7)	Bunch #(7)	Algorithm Minor Version(7)
14	Header Length(6)	Bunch #(6)	Algorithm Minor Version(6)
13	Header Length(5)	Bunch #(5)	Algorithm Minor Version(5)
12	Header Length(4)	Bunch #(4)	Algorithm Minor Version(4)
11	Header Length(3)	Bunch #(3)	Algorithm Minor Version(3)
10	Header Length(2)	Bunch #(2)	Algorithm Minor Version(2)
9	Header Length(1)	Bunch #(1)	Algorithm Minor Version(1)
8	Header Length(0)	Bunch #(0)	Algorithm Minor Version(0)
7	Number of Objects(7)	Data Type(7)	Algorithm Major Version(7)
6	Number of Objects(6)	Data Type(6)	Algorithm Major Version(6)
5	Number of Objects(5)	Data Type(5)	Algorithm Major Version(5)
4	Number of Objects(4)	Data Type(4)	Algorithm Major Version(4)
3	Number of Objects(3)	Data Type(3)	Algorithm Major Version(3)
2	Number of Objects(2)	Data Type(2)	Algorithm Major Version(2)
1	Number of Objects(1)	Data Type(1)	Algorithm Major Version(1)
0	Number of Objects(0)	Data Type(0)	Algorithm Major Version(0)

Table 3: The bit pattern for the first three header words in the data transfer format. Table **??** contains a list of the values for many of the words given here and itemizes all differences between the values sent to the L2CTT and sent from L2CTT. The Number of Objects is the number of CFT tracks. The number of STT associated tracks is given in Header 4. Bunch # is the 8-bit crossing number generated at the FE, and Rotation # refers to the turn number of the bunches in the accelerator and is a 16-bit value generated in the FE. Both are used for event alignment. The Algorithm Version refer to the software algorithms, the Processing Bits refer to the hardware configuration version. Status is as defined in Table 7.

bit #	Header 4	End-of-Record
31	Spare	Longitudinal Parity(15)
30	Spare	Longitudinal Parity(14)
29	Spare	Longitudinal Parity(13)
28	Spare	Longitudinal Parity(12)
27	Spare	Longitudinal Parity(11)
26	Spare	Longitudinal Parity(10)
25	Spare	Longitudinal Parity(9)
24	Spare	Longitudinal Parity(8)
23	Spare	Longitudinal Parity(7)
22	Spare	Longitudinal Parity(6)
21	Spare	Longitudinal Parity(5)
20	Spare	Longitudinal Parity(4)
19	Spare	Longitudinal Parity(3)
18	Spare	Longitudinal Parity(2)
17	Spare	Longitudinal Parity(1)
16	Spare	Longitudinal Parity(0)
15	Spare	Data Type(7)
14	Spare	Data Type(6)
13	Spare	Data Type(5)
12	Spare	Data Type(4)
11	Spare	Data Type(3)
10	Spare	Data Type(2)
9	Spare	Data Type(1)
8	Spare	Data Type(0)
7	Number of Associated STT Objects(7)	Bunch #(7)
6	Number of Associated STT Objects(6)	Bunch #(6)
5	Number of Associated STT Objects(5)	Bunch #(5)
4	Number of Associated STT Objects(4)	Bunch #(4)
3	Number of Associated STT Objects(3)	Bunch #(3)
2	Number of Associated STT Objects(2)	Bunch #(2)
1	Number of Associated STT Objects(1)	Bunch #(1)
0	Number of Associated STT Objects(0)	Bunch #(0)

Table 4: The bit pattern for the fourth header word and the trailer word. Note that while the trailer word is identical both into and out of the L2CTT, the fourth header word is for use in the L2CTT only. In Header 4, bits 0-7 contain the number of CFT tracks which have associated STT information. In the Trailer word, Bunch Number and Data Type are the same as specified in Tables 3 and **??** respectively.

Value	Meaning
7	Error during processing: Use at own risk
6	No processing Attempted (none required)
5	Object list truncated (any reason)
4	Receiver error on some input physical trailer
3	Spare
2	Spare
1	More data-type info
0	Real Data (1 = Monte Carlo, 0=Normal)

Table 5: This table specifies the meaning of the status word in Header 2 from L2CTT to L2G. Update me???

Header/Trailer Word	Values into L2CTT	Values out of L2CTT
Header Length	4 (in longwords)	3 (in longwords)
Object Length	1? 3?*** (in longwords)	2??? 3???(in longwords)
Object Format	0	0
Header Format	1	1
Data Type	166 (L2STT 0) 167 (L2STT 1) 168 (L2STT 2) 169 (L2STT 3) 170 (L2STT 4) 171 (L2STT 5) 172 (L2STT 6) 173 (L2STT 7) 174 (L2STT 8) 175 (L2STT 9) 176 (L2STT 10) 177 (L2STT 11)	5
Algorithm Major Version	1	1
Algorithm Minor Version	1	1
Processing Bits	0	0
Status Bits	See Table 7	

Table 6: The definition of the header and trailer words sent to L2CTT and from L2CTT. See Tables 3 and 4. Note that only the first three header words are sent to L2G. We note for completeness that the Header Length value sent to L3 from the L1CFT is three, not four.

Value	Meaning
7	Any kind of error during processing: Use at own risk
6	No processing Attempted (none required)
5	Object list truncated (any reason)
4	Receiver error on some input physical trailer
3	Spare
2	Spare
1	More data-type info: 0=MC, 1=Min-bias
0	Real Data (1 = Not Normal, 0=Normal)

Table 7: This table specifies the meaning of the Status Bits word in Header 2. We note for completeness that the Object list truncated value is never 1 in the value sent to L3 from the L1CFT.

bit #	Data Word
31	H Layer Fiber Number(5)
30	H Layer Fiber Number(4)
29	H Layer Fiber Number(3)
28	H Layer Fiber Number(2)
27	H Layer Fiber Number(1)
26	H Layer Fiber Number(0)
25	Isolated CFT Track?
24	Isolated CFT/CPS?
23	L2STT Duplicate Track (0=No/1=Yes)
22	Front End Board Number(6)
21	Front End Board Number(5)
20	Front End Board Number(4)
19	Front End Board Number(3)
18	Front End Board Number(2)
17	Front End Board Number(1)
16	Front End Board Number(0)
15	Sign of the P_T for Track
14	P_TBin(1)
13	P_TBin(0)
12	Extended P_TValue(2)
11	Extended P_TValue(1)
10	Extended P_TValue(0)
9	Match to High Threshold CPS?
8	Match to Low Threshold CPS?
7	ERROR Code(2)
6	ERROR Code(1)
5	ERROR Code(0)
4	Preshower Cluster Sector (0=Same Sector/1=Neighbor Sector)
3	Preshower Cluster Relative Address(3)
2	Preshower Cluster Relative Address(2)
1	Preshower Cluster Relative Address(1)
0	Preshower Cluster Relative Address(0)

Table 8: The bit assignments for the data longwords containing tracks from the L2CFTBC to the L2CTT. Within the data longwords bits 0-3 contain preshower and isolation information which is not used by the L2CTT but is passed along the L2G. Bits 8-15 contain the five bits of P_T including the sign. Bits 17-23 contain the FE board number which varies from 0-79. Bits 27-31 contain the H layer fiber number within the FE sector which varies from 0-43. Update me for In addition, the second data word contains 6 bits of χ^2, 11 bits of P_T, 8 bits of ϕ_0, 15 bits of b and 2 bits of dE/dX. Note that there is also the bit which indicates whether or not there is additional STT info. also is this a duplicate track??

5 Outputs of L2CTT

The outputs of the L2CTT go to:

- L2G
- L3
- TCC
- SCL

5.1 Non-L2G Output

The output to devices other than the L2G are described in more detail in Refs. [2], [6], [12] and references therein. To summarize, stored input and output values of events which are accepted by L2 (L2A) and sent to L3 are implemented using standard techniques with the VBD. The TCC interactions use the Bit3. The SCL interactions are made through the MBT. See Appendix B for more details about the SCL connections and the information passed, and Section 7 for SCL interactions on errors.

5.2 Output to Level 2 Global

The data transfer between the L2CTT and L2G is implemented using the standard MBT-to-MBT connection which sends the sorted track objects out of the front panel of the MBT to the "Pilot" MBT in the L2G crate via unshielded twisted pair (UTP) cable [6]. The data format begins with the standard header followed by the track objects and eventually followed by the standard trailer.

The data format for each event consists of up to 372 longwords of information with 32-bits per longword as shown in Table 9. Each transmission consist of 3 header longwords followed by up to 368 longwords of track data ***UPDATE ME??***(4 L2CFTBC $\times$ 46 tracks (maximum)/concentrator $\times$ 2 words per track) followed by an End-of-Record longword. Each track requires two data longwords. A maximum of 1,488 bytes of data (372 longwords $\times$ 4 bytes/longword) is passed for each event. Each event is padded (using a repeated footer) to be a multiple of 4 longwords.

The format for the header and trailers from L2CTT to L2G is the standard preprocessor to L2G format [2]. The requirements of the data transfer are given in more detail in Appendix C. The bit configuration is given in Tables 3 and 4. The differences between the header and trailers which are sent to the L2CTT and those which are sent from L2CTT to L2G are itemized in Table **??**. The headers contains overall event information, algorithm and transmission information. Header 3 contains the total number of tracks being sent. The fourth header, which contains the number of tracks per P_T bin is only for use within L2CTT and is not sent to L2G.

Table 10 gives the bit assignments for the data longwords which contain tracks. Data Word 0 contains the extrapolated ϕ position at EM3 (ϕ_{EM3}) and the P_T. Word 1 contains the ϕ_0 for the track as well as the isolation and preshower information passed from the L1CFT.

Header Longword 1
Header Longword 2
Header Longword 3
Data Longword 0 (Track 0, Word 0)
Data Longword 1 (Track 0, Word 1)
Data Longword 2 (Track 1, Word 0)
Data Longword 3 (Track 2, Word 1)
$\vdots$
Data Longword 2N-1 (Track N, Word 1)
End-of-Record

Table 9: The data longwords sent from the L2CTT to L2G. Note that in this example there are N tracks, where N is an integer less than or equal to 184.

bit #	Data Word 0	Data Word 1
31	Spare(7)	$\phi_0(7)$
30	Spare(6)	$\phi_0(6)$
29	Spare(5)	$\phi_0(5)$
28	Spare(4)	$\phi_0(4)$
27	Spare(3)	$\phi_0(3)$
26	Spare(2)	$\phi_0(2)$
25	Spare(1)	$\phi_0(1)$
24	Spare(0)	$\phi_0(0)$
23	$\phi_{\mathrm{EM3}}(7)$	$\mathrm{P_T}$ Sign
22	$\phi_{\mathrm{EM3}}(6)$	Spare(2)
21	$\phi_{\mathrm{EM3}}(5)$	Spare(1)
20	$\phi_{\mathrm{EM3}}(4)$	Spare(0)
19	$\phi_{\mathrm{EM3}}(3)$	High Thresh CPS Match
18	$\phi_{\mathrm{EM3}}(2)$	Low Thresh CPS Match
17	$\phi_{\mathrm{EM3}}(1)$	Isolated CFT/CPS
16	$\phi_{\mathrm{EM3}}(0)$	Isolated CFT
15	$\mathrm{P_T}(15)$	Spare(15)
14	$\mathrm{P_T}(14)$	Spare(14)
13	$\mathrm{P_T}(13)$	Spare(13)
12	$\mathrm{P_T}(12)$	Spare(12)
11	$\mathrm{P_T}(11)$	Spare(11)
10	$\mathrm{P_T}(10)$	Spare(10)
9	$\mathrm{P_T}(9)$	Spare(9)
8	$\mathrm{P_T}(8)$	Spare(8)
7	$\mathrm{P_T}(7)$	Spare(7)
6	$\mathrm{P_T}(6)$	Spare(6)
5	$\mathrm{P_T}(5)$	Spare(5)
4	$\mathrm{P_T}(4)$	Spare(4)
3	$\mathrm{P_T}(3)$	Spare(3)
2	$\mathrm{P_T}(2)$	Spare(2)
1	$\mathrm{P_T}(1)$	Spare(1)
0	$\mathrm{P_T}(0)$	Spare(0)

Table 10: This table gives the specification of the track objects output to L2G. The maximum amount of data sent is $\sim$ 185 obj/evt $\rightarrow$ 1,500B/evt. Update me***

6 Internal Functioning of the L2CTT

In this section we describe some of the functioning of the processing done within the L2CTT.

6.1 The L2STT to MBT Connection

The data is sent from the L2STT to the MBT's in the L2CTT via Hotlink. Within the MBT [6] the data are fed into FIFO memories and the end of an event is marked in the FIFO by a control signal. The FIFO can hold 16 events[4].

The data transfer from the MBT to the Administrator and Worker is negotiated between the Administrator and the MBT. The control section of the MBT knows which FIFOs are active and notes when a full event is available in each input FIFO. Data are not transferred out on the MBus until a full event is available. A GO control message from the Administrator signals the MBT that it is ready for event transfer and transmission begins when the MBT cards has an event ready to send. When all FIFO's have sent their event a "DONE" MBus signal is sent.

6.2 Worker Algorithms

The primary functionality of the L2CTT processing occurs on the Worker Alpha which contains a CPU running C++ code. The processing begins on the Worker Alpha after the Administrator has indicated to the Worker that it may process the next event. The algorithm, which is a method of a L2CTT worker class, begins with a call to the 'processBuffer' method with two[5] arguments: 1) a pointer to the four arrays which contain the header, trailer and track information and 2) the location of a buffer which is to contain the output. A fully functional prototype of the Worker code exists (and is compatible with the full trigger simulation effort) and has been used to test the functionality and perform timing studies (see Section 8). The following sections describe the sections of the code which calculate the $\mathrm{P_T}$, ϕ_0 and extrapolated track position ϕ_{EM3}.

This is only correct if there is no L2STT data. If there is SMT data, we only need to calculate the ϕ_{EM3} word which is similar to that listed.

6.2.1 Calculated Results: $\mathrm{P_T}$

The $\mathrm{P_T}$ from L1CFT is passed as a six-bit number. The most significant bit represents the sign of the track and is defined such that a 1 indicates that the trajectory of the particle has $\frac{\Delta\phi}{\Delta r} > 0$. The remaining five-bits represent the absolute value of the $\mathrm{P_T}$. The two most significant bits represent the $\mathrm{P_T}$ Bin of the track[6] and the three least significant bits indicate where within the bin. Before any events are run, the L2CTT initializes itself by placing the L1 to L2 $\mathrm{P_T}$ conversion into a lookup table. For more details on the conversion between the six bit number sent to L2 and the $\mathrm{P_T}$ reported to L2G see Appendix D.

[4]The FIFOs are actually 64KB/channel. But this can easily accommodate 16 events worth of raw tracking data as 16 events at 816 bytes/Event is only 13,056 bytes. FFFIIXXX MMMEEEE******

[5]There may be more in the future.

[6]Recall that $\mathrm{P_T^{Bin\ 3}} \geq 10$ GeV, 5 GeV$\leq \mathrm{P_T^{Bin\ 2}} \leq 10$ GeV, 3 GeV$\leq \mathrm{P_T^{Bin\ 1}} \leq 5$ GeV and 1.5 GeV$\leq \mathrm{P_T^{Bin\ 0}} \leq 3$ GeV.

6.2.2 Calculated Results: Extrapolated ϕ Positions

The L1CFT algorithm reports its ϕ position, not as a best fit ϕ, but rather by indicating which H layer fiber number (0-43) was hit (or more accurately which fiber was used in the fit) in the H layer. Thus, to convert this to global coordinates we must use the FE sector number as well as the fiber number in the H layer within the sector. To extrapolate to the center of EM3 and to the beam line we must use the $\mathrm{P_T}$ information. To convert $\mathrm{P_T}$ to a $\Delta\phi$ we use the equation:

$$\sin(\Delta\phi) = \frac{0.15 \cdot \mathrm{B} \cdot \Delta\mathrm{R}}{\mathrm{P_T}} \tag{1}$$

where B is the magnetic field in Tesla (= 2 Tesla) and ΔR is the change in radius from the H layer to the desired radius in meters ($\mathrm{R_A}$ = 0.2008 m, $\mathrm{R_H}$ = 0.5152 m, $\mathrm{R_{EM3}}$ = 0.916 m).

For illustration purposes we use an example track sent (using the conversion in Table 13) with $\mathrm{P_T}=11101_{\mathrm{binary}}$, H=4 and FE board number = 70. The FE board covers the region $5.50 < \phi < 5.58$. Since there are 44 fibers, and the H fiber hit is number 4, the position at the center of the hit fiber, ϕ^h, is given by:

$$\begin{aligned} \phi^h &= (70 \times \frac{2\pi}{80}) + (\frac{4+0.5}{44} \times \frac{2\pi}{80}) \qquad (2) \\ &= 5.4978 + 0.00803 \\ &= 5.058 \text{ radians.} \end{aligned}$$

To extrapolate for ϕ_{EM3} and ϕ_0 we use:

$$\phi_{\mathrm{EM3}} = \phi^h + \Delta\phi_{\mathrm{EM3}} \tag{3}$$

$$\phi_0 = \phi^h + \Delta\phi_0 \tag{4}$$

where $\Delta\phi_{\mathrm{EM3}}$ and $\Delta\phi_0$ is the change in ϕ from the H layer to EM3 and the beam line respectively.

The distance from the H layer to EM3 has $\Delta\mathrm{R} = \mathrm{R_{EM3}} - \mathrm{R_H}$ = 0.916 m - 0.515 m = 0.40 m. Using Table 13 we find the best guess $\mathrm{P_T}$=+15.96 GeV/c. Plugging into Equation 1 and using Equation 3 we find

$$\begin{aligned} \sin(\Delta\phi_{\mathrm{EM3}}) &= \frac{0.15 \cdot 2 \cdot 0.40}{15.96} = 0.00752 \\ \Delta\phi_{\mathrm{EM3}} &= 0.00752 \\ \phi_{\mathrm{EM3}} &= 5.5058 + 0.00752 \\ &= 5.5133 \end{aligned}$$

Converting this to the one bit of resolution means that we send L2G $\phi_{\mathrm{EM3}} = 140$ ($\mathrm{INT}(5.5133 \cdot \frac{160}{2\pi})$). Similarly we can calculate ϕ_0 using ΔR = 0.0 m - 0.5152 m = -0.5152 m:

$$\begin{aligned} \sin(\Delta\phi_0) &= \frac{0.15 \cdot 2 \cdot (-0.5152)}{15.96} = -0.00968 \\ \Delta\phi_0 &= -0.00968 \\ \phi_0 &= 5.5058 - 0.00968 \\ &= 5.496 \end{aligned}$$

which converts to a value $\phi_0 = 140$ sent to L2G. Note that to correctly extrapolate back to $r = 0$ we have been careful about the sign of ΔR in Equation 1. The resolution of these methods are shown in Figure 3.

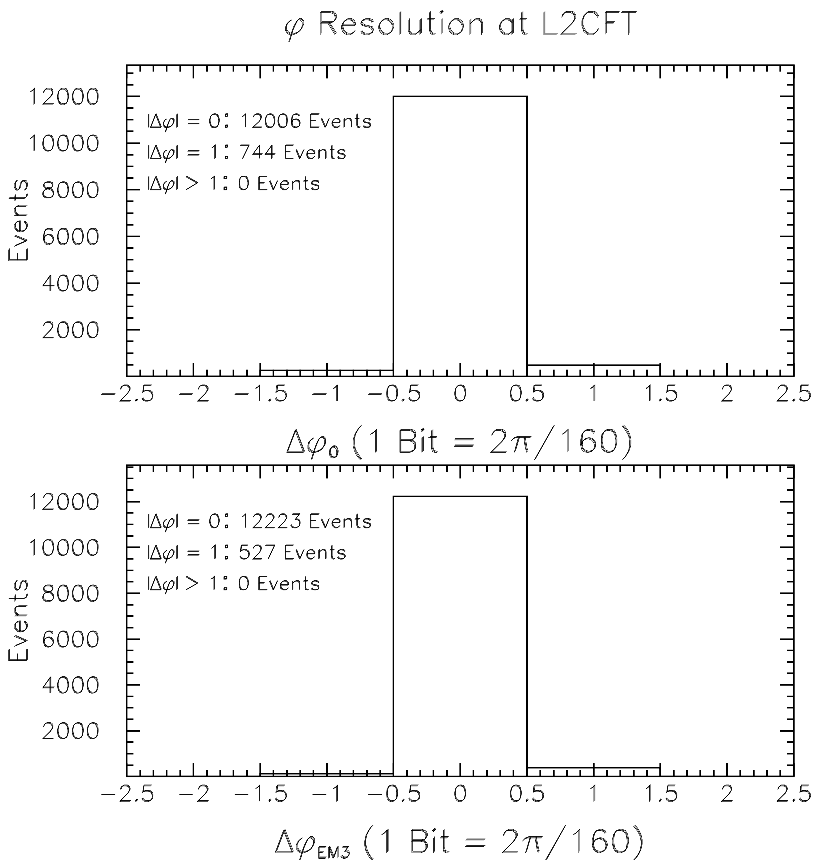

Figure 3: This plot shows the resolution of the ϕ_0 and $\phi_{\rm EM3}$ variables.

6.3 Sorting Algorithm

We use the Insertion Sort [13] algorithm in the Worker Alpha to order the tracks by $\rm P_T$ for output to L2G. Specifically, we create a linked list with each track object containing a pointer to the next highest $\rm P_T$ track in the list.

The algorithm begins by concatenating the tracks from the different L2CFTBC boards. This is done in a special way to help the processing by creating a list which indicates the order in which tracks will be processed. Using the number of tracks in each $\rm P_T$ Bin (Header 4), we place all the Bin 3 tracks from L2CFTBC 0 at the top of the list, followed by the Bin 3 tracks from L2CFTBC 1 and so on until L2CFTBC 3. We then do the same for Bin 2, Bin 1 and Bin 0[7].

[7]We assume for this discussion that the track definition is such that all tracks in bin i have higher $\rm P_T$ than in bin $i+1$.

Once the P_T bin ordered list is in place the algorithm is as follows: We take the first track on the list (assuming there is a track as indicated using Header 3 from each of the L2CFTBC boards) and make its "pointer to the next highest track" (NHT pointer) point to the end of the list (EOL), and the lowest P_T track pointer (LPT pointer) point to its memory location. Next we check to see if there is another track (again using the Header 3 information). If so, we compare the P_T of the new track to the P_T of the track in the lowest P_T track pointer location (our first track). If the P_T is lower, we make the LPT pointer point to the memory location of the new track, and its NHT pointer point to the original track. If the new track has a larger P_T we change the NHT pointer of the original track point to the new track, and the NHT pointer of the new track point to EOL. We repeat until all the tracks in the list are used. In this method the highest P_T tracks fall to the bottom. Thus, after all the Bin 3 tracks are sorted, the Bin 2 tracks need not be compared to any of the Bin 3 tracks in a natural manner which cuts down on processing time.

We next do the same thing for the impact parameter.

6.4 L2 Object Outputting

After all the tracks have been sorted, the L2 objects are created using the format in Table 10. These track objects are then copied, by P_T order, to the output memory location for output to L2G. The output location is specified (passed by reference to the L2CTT object as noted above) by the data movement control section of the Worker as a reference to the beginning of an array.

6.5 Scalar and Monitoring information

Here we specify the list of scalar and monitoring information which will be readout as well as its output format. For now we list the things we need to store: the number of events, errors, total number of tracks seen, tracks per event, number of truncated events, number of saturated events and, the number of synchronization errors. We note that some of these will be stored on the Worker while other will be collected on the Administrator.

7 Errors and Error Handling

There are a number of error conditions which need to be identified by the L2CTT. In addition there are a number of error condition to which the L2CTT has to respond. At the moment the only one which has been discussed is the case of events with mismatched pieces.

The Administrator Alpha looks at all pieces of the event to check if all the pieces are indeed from the same event. If they are not, this causes an error, and the Administrator sends this information to the MBT by writing to a register on the MBT. The SCL Mezzanine card (on the MBT) then notifies the L2 HWFW of the synchronization error (L2ERROR). The Administrator keeps count of these occurrences.

The SCL Hub responds by broadcasting *SCL Initialize* to which MBT responds by raising L1_BUSY. Since any preprocessor node can provoke *SCL Initialize*, Administrator must be notified by MBT. After notification from MBT, the Administrator clears its buffers and requests that the Workers do the same. After clearing their buffers the Workers notify the Administrator. After all Workers have notified the Administrator, the Administrator notifies the MBT to rescind the L1_BUSY.

In the event of all L3 readout buffers being full, or all Wait for L2 Decision Buffers being full, the front end automatically throttles the framework and no more events are sent to L2CTT.

8 Simulations

We'll fix this later....

We have performed a number of simulations designed to test the prototype of the L2CTT worker code. To test the L2CTT object creation and sorting algorithms we have created a set of toy simulations for track generation, the L1CFT reconstruction, and outputs and the L2CTT crate surrounding the Worker Alpha. The simulation uses C++ code. For simplicity the original timing studies are done by compiling and running using Microsoft Visual C++ (Version 6.0) on a 233 MHz Pentium II processor running in a PC running Windows NT. The results were checked using an Alpha running Unix.

The track generator simulation produces events with a very stiff $\mathrm{P_T}$ distribution ($\frac{dN}{d\mathrm{P_T}} \propto e^{-0.1\times \mathrm{P_T}}$) and flat in ϕ_0. For each event a number of tracks are simulated. The L1CFT simulation assumes a perfect detector such that it finds all tracks, has no fakes, and creates L1 track objects, headers and trailers perfectly. There is no smearing, and all information lost is solely due to the resolution of the information transfer protocols. To make conservative timing estimates as well as eliminating any bias due to the $\frac{dN}{d\mathrm{P_T}}$ distribution, each track is placed on their respective L2CFTBC board unsorted by $\mathrm{P_T}$. To simulate the output, each L2CFTBC is given its appropriate headers, data and trailers and outputed into a text file.

The L2CTT simulation mocks up the preprocessor data movement by reading in events from the text file and storing the first 50 events. These events are put into the proper input buffer format, and pointers to the arrays are passed to the Worker prototype code. The Worker algorithm (object creation, sorting and object output) is run 200,000 times on each event and timed. This simulation is repeated for track multiplicities of 0, 1, 2, 4, 8, 16, 32 and 64 tracks for 50 events each. The results are given in Table 11 and shown graphically in Figure 4. For events with less than 40 tracks, the processing time is less than 1 microsecond per track. Note these numbers were checked using an Alpha processor which was found to be a factor of roughly 1.6 faster.

To convert this to a processing time distribution we need the $\mathrm{N_{Track}}$ distribution per event into the L2CTT. This is estimated using a separate technique. The Trigger Simulator Ntuples [14] are used to get a good estimate of the efficiency and fake rates for finding tracks in the CFT. The trigger simulation takes as input fully simulated dijet events from ISAJET in 6 $\mathrm{P_T}$ bins with varying number of additional min-bias events. These events are then run through a full L1 simulation (i.e, calorimetery, tracking and muon). This simulation which does a full track finding gives an estimate which takes into account the track finding efficiency as well as the fake track rate as a function of the number of extra min-bias collisions in the event. In addition it allows a L1 decision based on a straw L1 trigger table. The number of tracks reported into L2CTT is shown in Figure 5. The different distributions represent samples which have a mean number of interactions per crossing[8] of 0.5 and 5.0. We have taken into account the poisson nature of the number of actual interactions in an event given the mean number of interactions. In addition the plots show for comparison the case where all events are investigated (no L1 cut or said differently a min-bias sample) and the case in

[8] The mean number of interactions per crossing is a function of the luminosity, the number of bunch crossings and the bunch spacing.

which events are required to pass L1. Each distribution appears to be fairly well modeled by an exponential. Figure 6 shows the mean number of L1CFT track candidates as a function of the mean number of interactions per collisions.

Figure 7 shows the results of combining the processing time per N_{Track} and the N_{Track} per event distributions to show the fraction of events which will be processed within a specified time. From the figure we see that even with an average of 5 interactions per crossing 95% of the events will be processed in less than 60μs. Even for 10 interactions per crossing, we see less than 35 tracks (on average) into L2CTT. As mentioned above these results should be fairly insensitive to the fact that we used a different P_T spectrum to measure the processing time per event for a given track multiplicity. While these results are clearly not highly accurate as effects such as the noise in the CFT could easily drive the fake track rate higher, we have tried to compensate by not presorting the tracks as done in the L2CFTBC or taking into account the gains from running the code on the Alpha processor (which should be faster than the simulation we have run without optimization). Based on these results we believe the L2CTT, except in extreme situations, should be able to function with minimal deadtime and without additional processing power.

N_{Track}	Make Objects (μSec)	Make and Sort Objects (μSec)	Make, Sort Output (μSec)
0	0.5	0.5	0.8
1	1.1	1.1	1.3
2	1.5	1.5	1.9
4	2.2	2.7	3.1
8	3.6	5.1	5.7
16	6.0	10.8	11.8
32	10.7	25.4	27.4
64	20.7	72.2	78.8

Table 11: The processing time as a function of track multiplicity. Note these numbers were checked using an Alpha processor which was found to be a factor of roughly 1.6 faster. These numbers do not include the data transfer time in and out of the Worker Alpha to L2G.

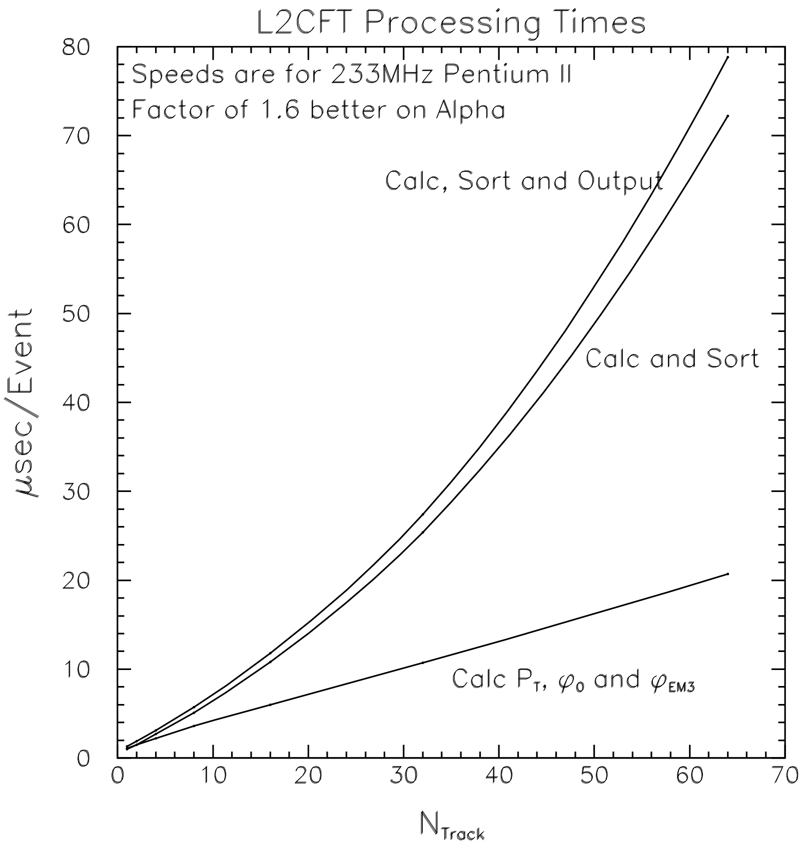

Figure 4: The processing time as a function of the track multiplicity. Note these numbers were checked using an Alpha processor which was found to be a factor of roughly 1.6 faster. These numbers do not include the data transfer time in and out of the Worker Alpha to L2G.

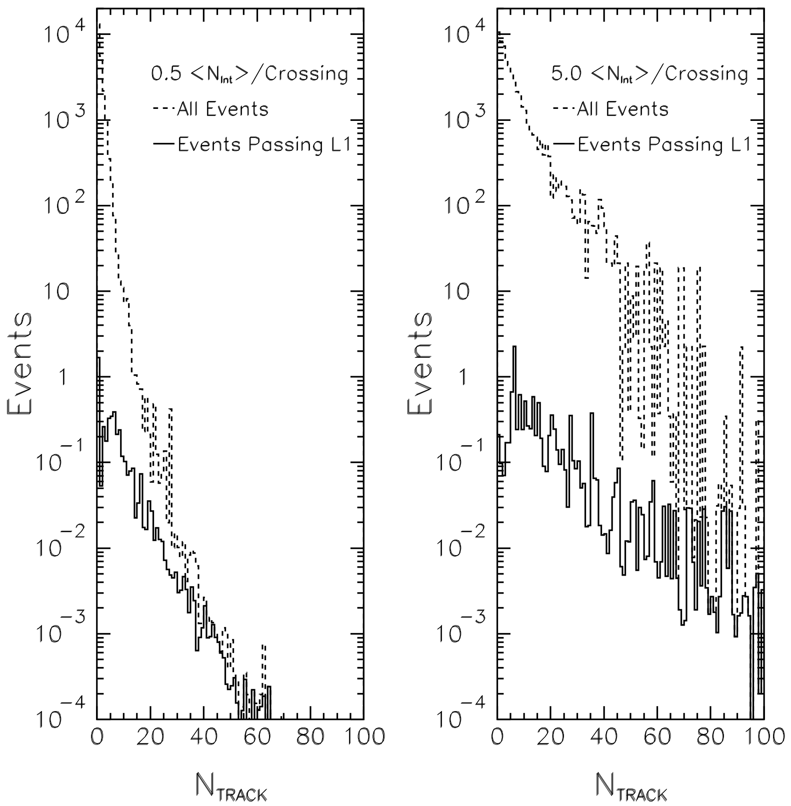

Figure 5: The number of tracks, as reported by the L1CFT, for 0.5 and 5 interactions per crossing using the trigger simulator. The solid line is all events, and the dashed line shows the events which pass the L1 trigger. Note that this takes into account the Poisson variation in the number of min-bias events per collision and the fake track rate.

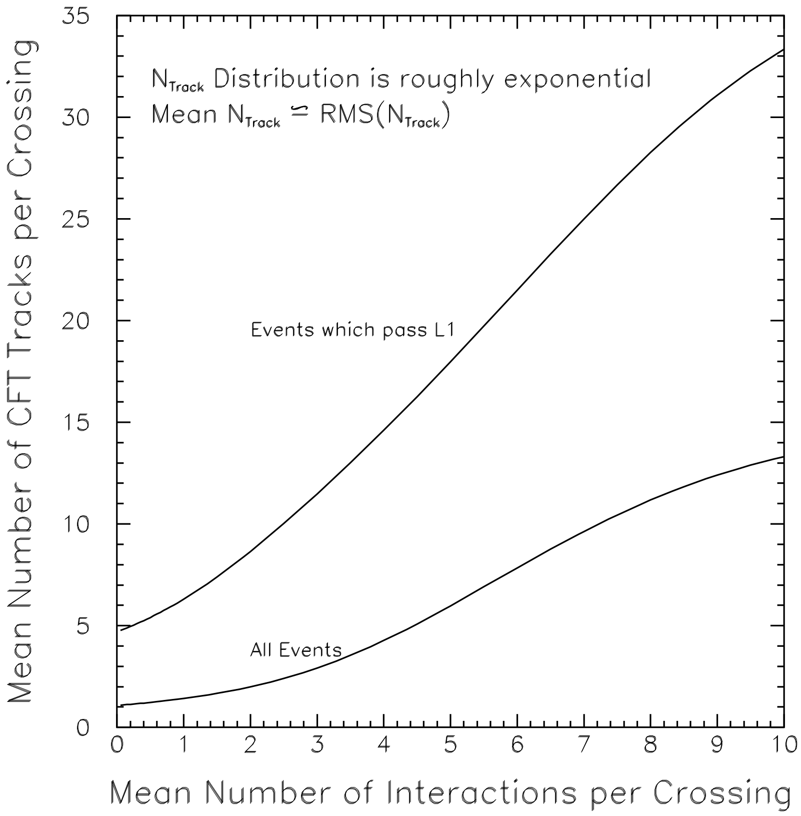

Figure 6: The mean number of tracks, as reported by the L1CFT, as a function of the mean number of events per crossing using the trigger simulator. Note that this takes into account the Poisson variation in the number of min-bias events per collision and the fake track rate.

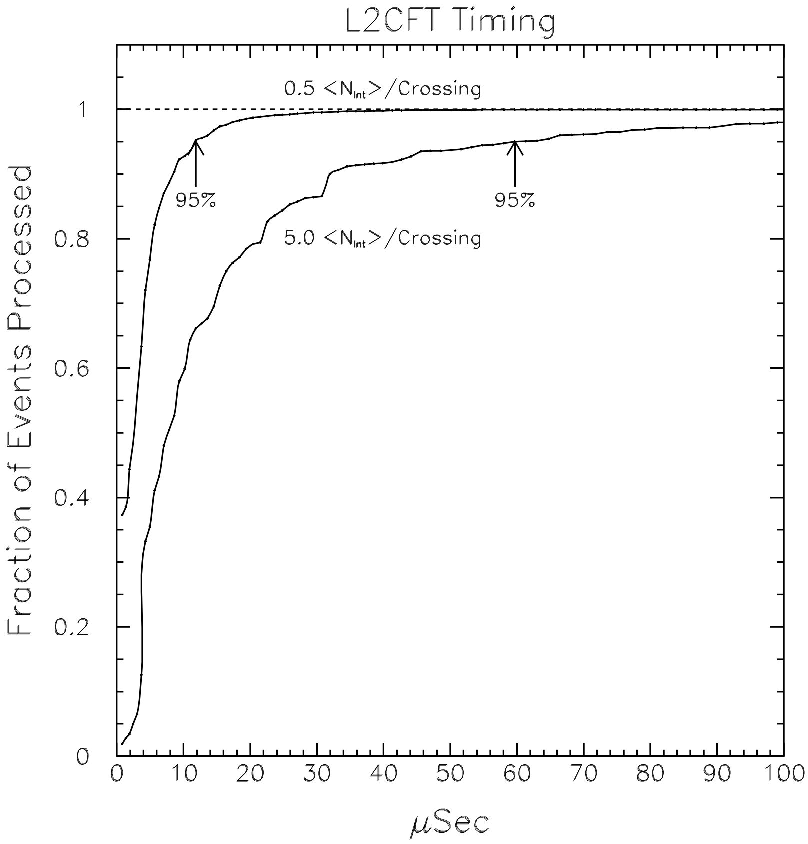

Figure 7: The expected time per event distribution out of the L2CTT. Note that these numbers are conservative in that they use the slower 233 MHz Pentium II processor and have no sorting from L1CFT.

9 High Luminosity Contingency Plans

While studies indicate that there should be enough time to do the full processing and data transfer, there exists the possibility that we may need to cut back on the processing time. This could be the case for running with a large number of tracks sent to L2 per event (e.g. very high luminosity, lots of noise/fakes in the L1CFT etc.). This can be done in a number of ways. For example, we may put a P_T cut requirement to limit the number of tracks sent to L2G. This may mean that we only send information from a certain number of different P_T bins. Another option is to simply limit the number of tracks sent. Either requirement would help insure that we do not get close to the 4kB event size limit (this includes the header and trailer information). A final possibility is to limit the P_T to one Byte and condense the output object from 2 longwords to 1.

10 Conclusions

We have presented a technical design report for the Level 2 Central Tracking Trigger pre-processor crate. The inputs and outputs for the crate are fully specified and appear to be adequate for full functionality. A prototype of the Worker code exists and has been used to do proof of principle and timing studies. The results of the timing studies indicate that the crate should be able to operate with minimal dead-time until only the most extreme of circumstances. There are a number of contingency plans already in place to deal with those scenarios. This design is easily expandable to include the L2STT when it comes online.

11 Acknowledgments

The authors would like to sincerely thank John Hobbs, Manuel Martin, Rob Martin, Roger Moore, and Silvia Tentindo-Repond discussions and careful readings of the text.

References

[1] D. Toback *et al.*, *Specification of the Level 2 Central Tracking Trigger Preprocessor Crate* DØNote 3645. July 1999.

[2] D. Edmunds, S. Gross, P. Laurens, J. Linnemann and R. Moore *Technical Design Report for the Level 2 Global Processor* DØNote 3402, Feb 1998.

[3] This is the L2STT TDR reference.

[4] DØ electronics group (Fred Borcherding *et al.*). *Technical Design Report for the Upgrade L1/L2 Tracking Trigger. Version 2* DØ Note XXXX.

[5] FIC Reference.

[6] D. Baden, *Technical Design Report for the Magic Bus Transceiver Card* DØ Note XXXX.

[7] SCL Reference

[8] Magic bus reference.

[9] Alpha Reference.

[10] Bit3 Reference.

[11] VBD Reference.

[12] L3 Reference

[13] R. Fleischman, Highwind Corp, private communication.

[14] Trigger Simulation Reference

A Level 1 CFT

The CFT consists of 32 concentric barrel-shaped layers of scintillating fibers arranged in 16 "doublet" layers. The fibers in each layer are at the same radial distance from the z-axis and are such that half of the layers have fibers parallel to the z-axis and half are at an angle to the z-axis to provide stereo tracking reconstruction. In the ϕ coordinate, the fiber tracker system is sliced into 80 sections, each of which subtends about 4.5 degrees. The inner (A layer) and outer (H layer) layers are spanned by 16 and 44 fibers per sector respectively.

The L1 tracking trigger performs the following tasks:

- It reads out the tracking hits from the CFT and does the track finding.
- reads reads out preshower hits from the Central Preshower (CPS) detector and does cluster finding.
- Tracks are matched with CPS clusters.
- The tracks and CPS clusters are stored for use in the event of a L1 Accept (L1A) and a L2A.
- In the event of a L1A, the track lists are read out to L2. In the event of a L2A, the track lists are read out to L3.

During normal operation the trigger begins by reading in the raw fiber hit information from the CFT using the Front End (FE) boards (1 FE board covers a $2\pi/80$ degree section). All tracks with $\mathrm{P_T}>$ 1.5 GeV which cross the outer detector layer ($|\eta_{\mathrm{Detector}}| < 1.5$) are completely contained within one sector (the home sector) or are partly in one of the adjacent sectors (neighbor sectors). To form a seamless trigger, all outer layer hits from the home sector are combined with the hit information from the two neighboring sectors. Track finding uses only the axial layers and is implemented by comparing hit patterns to pre-programmed possible track roads with $\mathrm{P_T}>$ 1.5 GeV. In parallel, the CPS does its cluster finding. After track and cluster finding, there is an attempt to match each track and cluster candidates[4].

B Serial Command Link Inputs

The L2CTT crate has a direct connection to the SCL which is used to transfer information from the trigger framework (the SCL Hub) to the crate. The MBT has separate input and output connections which connect directly into the MBT via a mezzinine card. The MBT separates the SCL information by source and makes it appear as the 8th data source on the card. For example, the MBT selects the L2 accept/reject SCL messages buffers and makes them available to the Administrator as needed. In this section we summarize some of the SCL commands as they affect the L2CTT. The commands the SCL sends out are given in Table 12.

The SCL sends notification of a L1A to every geographic section which needs to read out to L3. In the case of L2CTT, the receipt of the L1A SCL message means that at least a header block MUST be sent to L2G for this event and a header block must be prepared for eventual L3 readout in case the event passes L2.

The MBT often stores information from the SCL. This is done by transforming the input into the correct format and directing it to its input FIFO. Each L1A is accompanied by header/trailer information which is placed in a source FIFO and appears to the rest of the system as a standard data source. The relevant L1 information carried by SCL is the 3B crossing number at the time of the L1A and the 2B set of L1 Qualifiers.

There are number of L1 qualifiers. The ones of greatest interest to the L2CTT are the *Unbiased Sample (UBS), Forced Write, L2 Global Needed* and *Collect Status.* The obligation to read out to L3 is announced by receipt of the L1A SCL message, but a separate *L1 Needed* qualifier will be provided to all L2 preprocessors (*L2 Pre_CFT*) and possibly a *L2 Global Needed* may be defined. These qualifiers offer the L2CTT the option to skip execution of its algorithm for an event without its *Needed* qualifier and produce a minimal header-only output to L2G (and preparing a minimal L3 output). This qualifier would be attached to a L1 bit whenever the L2G requires the input of the preprocessor. For example, an electron requirement needs the calorimeter EM preprocessor, the L2CTT, and possibly the L2PS preprocessor.

In the L2CTT, receipt of a *UBS* qualifier will result in readout to L3 of the L2CTT inputs, and possible additional output information beyond that sent to L2G. The standard output to L2G is not changed by the *UBS* bit.

The *Forced Write* qualifier is intended to provide a mechanism for special runs meant to study new L2 triggers. It has exactly the same effect as the *UBS* bit in L2, however, the behavior in L3 is different.

After receipt of an event with a *Collect Status* qualifier, the L2CTT (and all other L2 crates) will capture their scalers and other monitoring information and place it where TCC can retrieve it for serving with monitoring (to be read out via the Bit3). The scalers will be captured after processing of the event is completed, so that scaler information should be exactly matched between the L1 and L2 HWFW, the preprocessors and L2G. The monitoring blocks will be tagged with the L1 crossing number of the event with the *Collect Status* qualifier, so TCC can assemble a consistent set of statistics. *Collect Status* qualifiers will be generated once every 5 seconds (0.2 Hz).

Command	Description
SCL_READY	Serial Command Link Ready Status
SCL_SYNCERROR	Serial Command Link Synchronization Error
SCL_DATAERROR	Serial Command Link Data Error
SCL_ACK	Acknowledge and Clear SCL Error Flags
CLK_53	53 MHz Clock
CLK_7	7.5 MHz Clock
CURRENT_TURN[15..0]	Current Turn Number
CURRENT_BX[7..0]	Current BX Number in this Turn
FIRST_PERIOD	First Period in a Turn Marker
BEAM_PERIOD	Period with Beam Marker
SYNC_GAP	Sync Gap Marker (No L1 Accepts)
COSMIC_GAP	Cosmic Gap Marker (Only Cosmic L1 Accepts)
SPARE_PERIOD	Spare Period Marker
L1_PERIOD	Period with L1 Accept Issued
L1_ACCEPT	L1 Accept in this Geo Section
L1_TURN[15..0]	Level 1 Turn Number
L1_BX[7..0]	Level 1 BX Number in this Turn
L1_Qual[15..0]	L1 Accept Qualifiers/Geo Section L3 Transfer Number
L2_PERIOD	Period with L2 Decision Issue
L2_REJECT	This Geo Section L2 Reject
L2_ACCEPT	This Geo Section L2 Accept
INIT_SECTION	Initialize Geographic Section Flag
L1_BUSY	Busy L1 Status
L2_BUSY	Busy L2 Status
L1_ERROR	Error L1 Flag
L2_ERROR	Error L2 Flag
INIT_ACK	Init_Ack Signal to Hub-End
SYNC_LOST	SCL Receiver Synchronization Lost
SPARE_STATUS[1..0]	Spare Status Signals to Hub-End

Table 12: The commands from the Serial Command Link.

C Appendix: L2G Data Transfer Requirements

We itemize here the requirements for the data transfer between L2CTT and L2G. We point out the specific requirements where appropriate.

- The smallest fragment size for a variable within a track object is 1 byte.
- The track objects's total data size must be an integer number of longwords (1 longword = 4 bytes = 32 bits); a track object is 2 longwords.
- Multi-byte fields do not cross boundaries between longword boundaries.
- The data must prefaced with a header which is an integer number of longwords in length; the L2CTT header is 3 longwords.
- The data transmission will be completed with a trailer which is an integer number of longwords; the L2CTT trailer is 1 longword.
- Some number of bytes will be added after the trailer to make the total number of longwords divisible by 4 since the Magic Bus is a 16B wide bus. No blank objects will be sent to fulfill this criteria.
- Where possible each track object will be identified by rapidity, azimuth and $\mathrm{E_T}$ in that order[9]
- Tracks objects will be sorted in descending $\mathrm{P_T}$ order[10].
- The least significant bit for rapidity will be 0.05, for azimuth $2\pi/160$, and for transverse momentum 0.25 GeV/c.

  Since there are two bytes for $\mathrm{P_T}$ the full scale is 16,383.75 GeV.
- Azimuth will be reported in standard DØ coordinates. Since the charged particles curve in the magnetic field, the reported values of ϕ will be extrapolated to two different interesting radii. These are at $r = 91.6$ cm and $r = 0$. The first radius is the center of EM3 in the central calorimeter, or showermax, (used in electron and muon matching) and will be referred to as ϕ_{EM3}. The extrapolation back to $r = 0$ is useful for for physics object processing (such as J/ψ mass reconstruction) and is referred to as ϕ_0.

[9] Note that this only partially applies to the L2CTT until the STT is used as an input.

[10] Note that when the input is from the STT, there will be two separate lists: one by $\mathrm{P_T}$ and one by impact parameter significance.

D L1 to L2 P_T Conversion Details

There is some discussion about the manner in which the P_T information will be sent to the L2CTT. Initially, it was decided we would use the offset between the A and H layers. This scenario is described in Appendix E. A second solution (the "H and P_T" method) is to send a P_T word which is determined by setting the thresholds as the minimum and maximum reaches of the region and interpolating by 1/Pt per word. The results are given in Table 13, and the mean value of the P_T is what is sent to L2G. The current method is to use a hybrid solution with the P_T information sent for High P_T and offset for low P_T.

Bin #	P_T BinID	P_T^{Min} (GeV/c)	P_T^{Max} (GeV/c)	P_T^{Mean} (GeV/c)
0	0000	1.50	1.71	1.60
0	0001	1.71	2.00	1.85
0	0010	2.00	2.40	2.18
0	0011	2.40	3.00	2.67
1	0100	3.00	3.33	3.16
1	0101	3.33	3.75	3.53
1	0110	3.75	4.29	4.00
1	0111	4.29	5.00	4.62
2	1000	5.00	5.71	5.33
2	1001	5.71	6.67	6.15
2	1010	6.67	8.00	7.27
2	1011	8.00	10.00	8.89
3	1100	10.00	13.32	11.42
3	1101	13.32	19.93	15.96
3	1110	19.93	39.55	26.50
3	1111	39.55	2660.96	77.95

Table 13: The conversion between the 4-bits of P_T information reported by L1CFT and the P_T reported to L2G for the "H and P_T" method. The two most significant bits of P_T BinID represent the Bin number. Note that while the maximum P_T is really infinity, the highest P_T for which an equation exists in L1 is 2660.96 GeV/c.

E $\mathrm{P_T}$ Calculations Using the Offset Method

In this appendix we give details on the Offset to $\mathrm{P_T}$ conversion. To convert from offset to $\mathrm{P_T}$ we use the 4 bits of offset information as well as the H layer fiber number. The 4 offset bits actually specify the $\Delta\phi$, from inner to outer layers of the CFT, measured in number of fiber widths at the inner layer. This is actually subtle. For example, if at the A layer the found track passes through fiber 0 and at the H layer the fiber hit is fiber 4 then the offset will be reported as 1 and the sign will be equal to 1. How did we determine this? H layer fibers of 0, 1 and 2 have an offset of 0 from A layer fiber 0. Similarly H layer fibers 3, 4 and 5 have an offset of 0 from A layer fiber 1 (they have an offset of 1 from A layer fiber 0). To complete the loop, H layer fibers 6 and 7 have an offset of 0 from A=2 and H layer fiber numbers 8, 9 and 10 are associated with an A=3. This repeats every 4 A layer fiber numbers. Thus, given an H layer fiber number and an offset one can determine the $\Delta\phi$ of the particle as it traverses the CFT. The sign of the charge of the particle is such that the curvature makes $\Delta\phi$ positive for increasing r.

To convert the offset (and thus $\Delta\phi$) to a $\mathrm{P_T}$ we use the equation:

$$\mathrm{P_T} = \frac{0.15 \cdot \mathrm{B} \cdot \Delta\mathrm{R}}{\sin(\Delta\phi)} \tag{5}$$

where B is the magnetic field in Tesla (= 2 Tesla), ΔR is the change in radius from the inner layer to the outer layer in meters (R_A = 0.2008 m, R_H = 0.5152 m, ΔR = 0.32 m) and $\Delta\phi$ is the change in ϕ from the inner radius to the outer radius. Using our previous example with offset = 1 and H = 4 and sign = 1, we can deduce that the fibers hit were A=0 and H=4. Since there are 1,280 and 3520 fibers in the A and H layers respectively the fiber widths are 4.91 mrad and 1.78 mrad respectively. Thus, the center of the A=0 fiber is at $\phi(A = 0) = 2.45$ mrad (relative to the sector boundary)and the center of the H=4 fiber is at $\phi(H = 4) = 8.03$ mrad for a $\Delta\phi$= 3.79 mrad. Plugging this into Equation 5 we find:

$$\begin{aligned}
\Delta\phi &= \phi(H=4) - \phi(A=0) \\
&= 0.00803 - 0.00245 \\
&= 0.00558 \\
\mathrm{P_T} &= \frac{0.15 \cdot 2 \cdot 0.32}{0.00558} = 17.21 \text{ GeV}
\end{aligned}$$

Note that the value sent to L2G would be 17.25 GeV since the least significant bit corresponds to 0.25 GeV. Before any events are run, the L2CTT initializes itself by calculating the $\mathrm{P_T}$ for all offset and H combinations and putting them into a lookup table.

Using the Worker prototype code we have run a toy simulation of the L1CFT (see Section 8)in which the parameters of a particle trajectory are converted into FE number, H fiber number, offset, and sign by simply using the A and H layer fiber numbers the track passes through. Figure 8 shows the reported offset as a function of input $\mathrm{P_T}$. Figure 9 shows how the $\mathrm{P_T}$ varies as a function of H for an offset of 3. Figure 10 shows the $\mathrm{P_T}$ resolution for different values of $\mathrm{P_T}\dot{\mathrm{S}}$imilarly Figure 3 shows the ϕ_0 and ϕ_{EM3} resolutions.

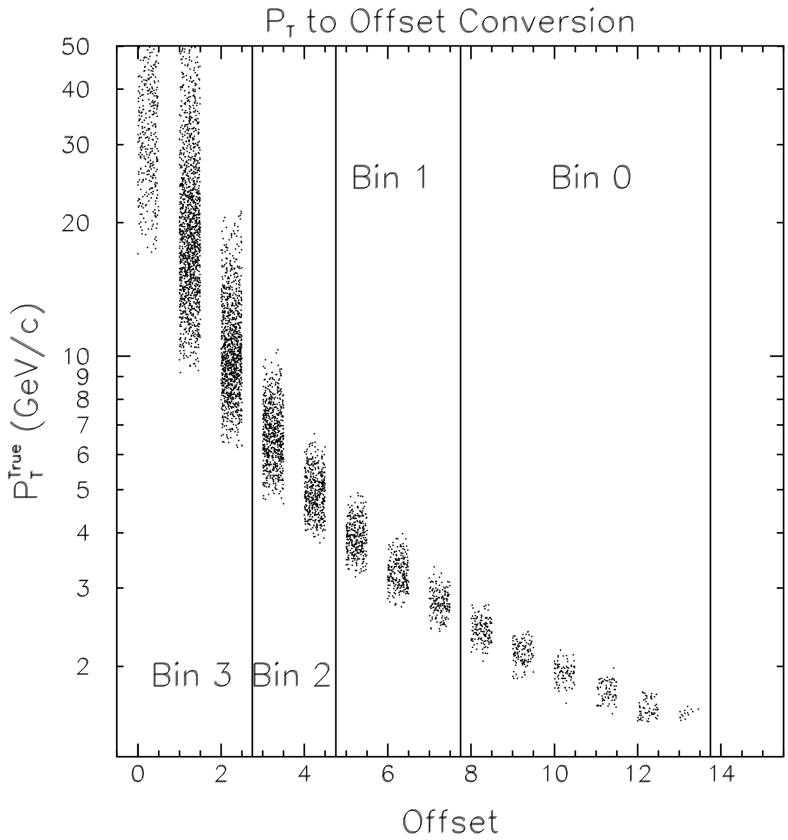

Figure 8: P_T vs. Offset for the toy simulation of L1CFT. Note that the P_T is not a function of offset. Said differently, not all tracks with an offset of 2 are of higher P_T than have an offset of 3.

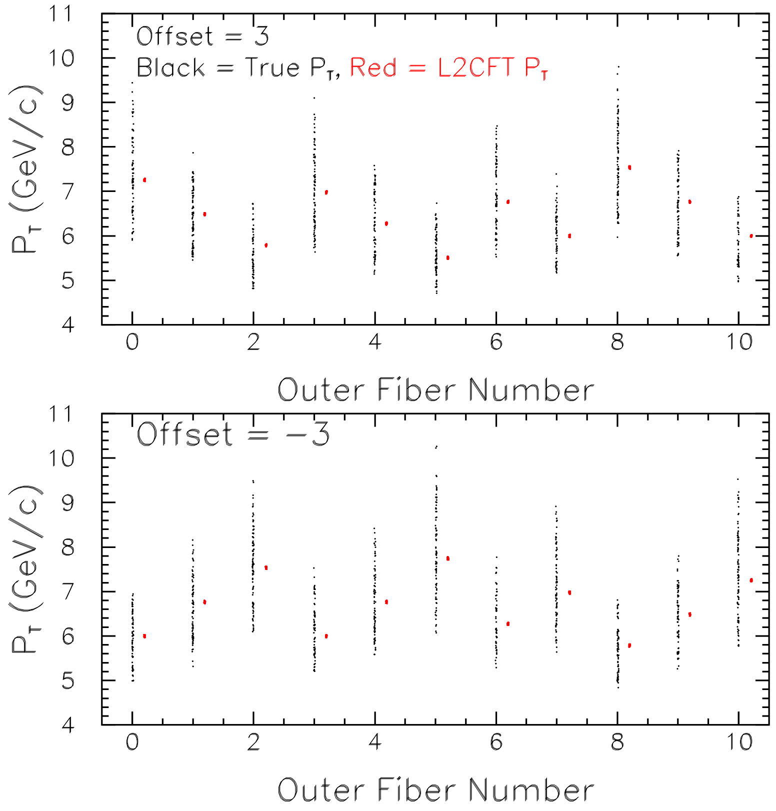

Figure 9: This plot shows how the P_T varies as a function of the H layer fiber number for an offset of 3. The dots just to the right of the main distributions show the best guess P_T using Equation 5.

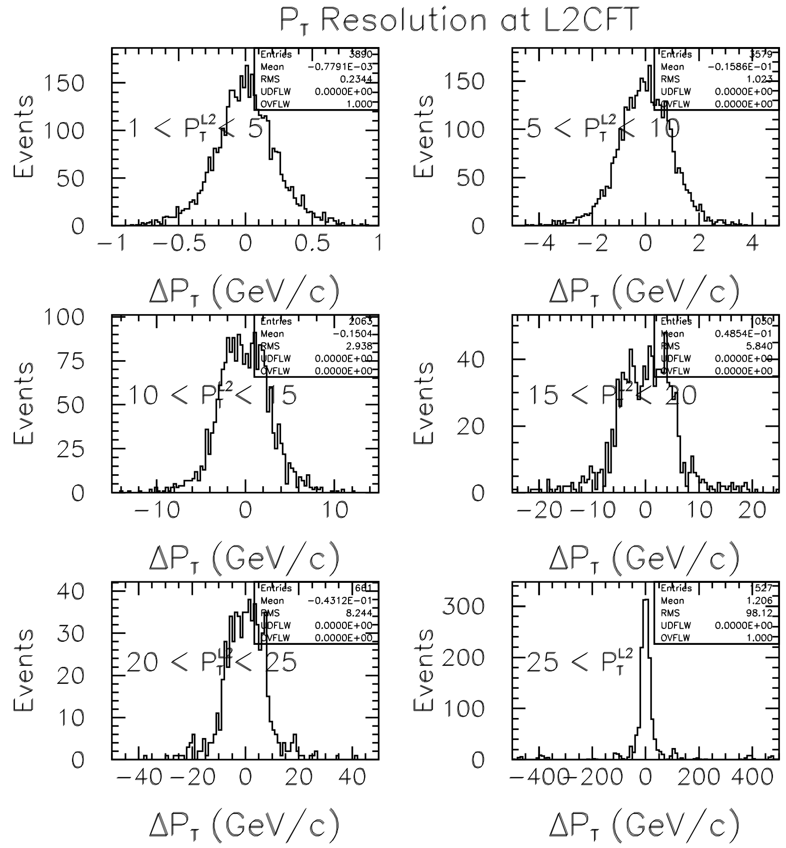

Figure 10: This plot shows the P_T resolution as P_T varies. Note that the resolution worsens as P_T increases and becomes unreliable above about 25 GeV.

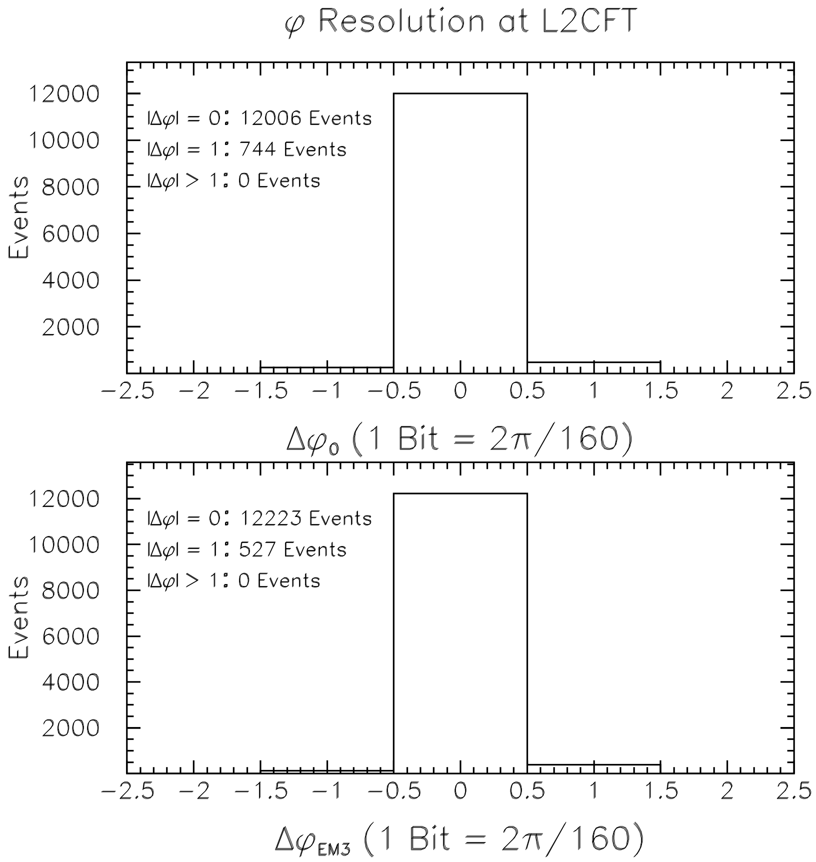

Figure 11: This plot shows the resolution of the ϕ_0 and ϕ_{EM3} variables using the worker prototype.