CDF/DOC/CDF/PUBLIC/4152

Description of the CDF Data Structure for Run II

W. Badgett, J. Bellinger, K. Biery, J. Boudreau, F. Chlebana, D. Dagenhart, C. Gay, G. Feild, L. Gerchtein, P. Gris, E. James, M. Jones, R. Kennedy, S. Kuhlmann, T. LeCompte, J. Lewis, C.-J. Lin, T. Liu, A. Meyer, J. Nachtman, S. Nahn, C. Paus, C. Pilcher, K. Pitts, J. Proudfoot, B. Reisert, M. Schmidt, L. Sexton-Kennedy, R. Snider, D. Stuart, M. Tecchio, K. Terashi, M. Wang, B. Winer

Edited by
D. Dagenhart (wdd@fnal.gov) and F. Chlebana

December 4, 2003

Abstract

This note describes the data structure for Run II. A brief description is given of each detector in order to motivate the choice of the data format and to illustrate numbering conventions. An overview of the DAQ and trigger is given as well as a description of how the Front End Crates are populated. Guidelines are given for defining online banks.

http://www-cdf.fnal.gov/internal/upgrades/computing/data_format/data_format.html

Contents

1 Introduction

This note specifies the data format for the raw data banks for Run II. A brief description of each detector component is included in order to help motivate the choice of the data format. An up to date version of this document is linked to the "Data Format WWW Page" at:

`http://www-cdf.fnal.gov/internal/upgrades/computing/data_format/data_format.html`

This note specifies

- The meaning of each word in all raw data banks.
- The bit fields in each word.
- The definition of channel ID's, module ID's, and similar indices.

The intention of this note is to provide enough information so that one could interpret a hex dump of the raw data.

Each detector group is responsible for assuring that:

- The detector elements, boards, crates, and related software for their detector create data blocks with the format described in this note.
- The reconstruction code for their detector can process raw data with the format specified in this note. The bank classes will be written by an expert provided by the offline, but individual reconstruction groups are responsible for testing the bank classes and using them to access the data.
- The simulation produces the raw data banks with the format specified in this note. The simulation and data formats should be identical.
- The sections of this note relevant to their detector are correct and up to date. The detector groups are responsible for writing the latex files for the initial version and significant revisions of this note (put the file on cdfsga and send the filename to wdd@fnal.gov). The current latex files are available at the web site above.
- Coordinating changes between online and reconstruction.

The editor of this note:

- Edits the latex files submitted by the detector groups. Maintains the files and the web site.
- Consults with the detector groups to help them establish data formats. In particular, every effort will be made to ensure that data formats from different detectors follow a few common patterns.
- Helps to coordinate and write new raw data bank classes. Also provides documentation on how to use the bank classes.
- Attempts to find and resolve problems.

Please report any errors found in this document. Please update sections any time data formats change. There is no way the editor can keep track of all the formats for all the detectors in the experiment, unless the detector groups provide the information.

2 Detector Overview

A schematic of the CDF detector is presented in Figure 1. The detector is partitioned into west and east halves. The main inner tracking components include the Silicon Vertex Detector (SVXII), Intermediate Silicon Layers (ISL) and Central Outer Tracker (COT). The calorimeter consists of the Central, Plug and Endwall Calorimeters.

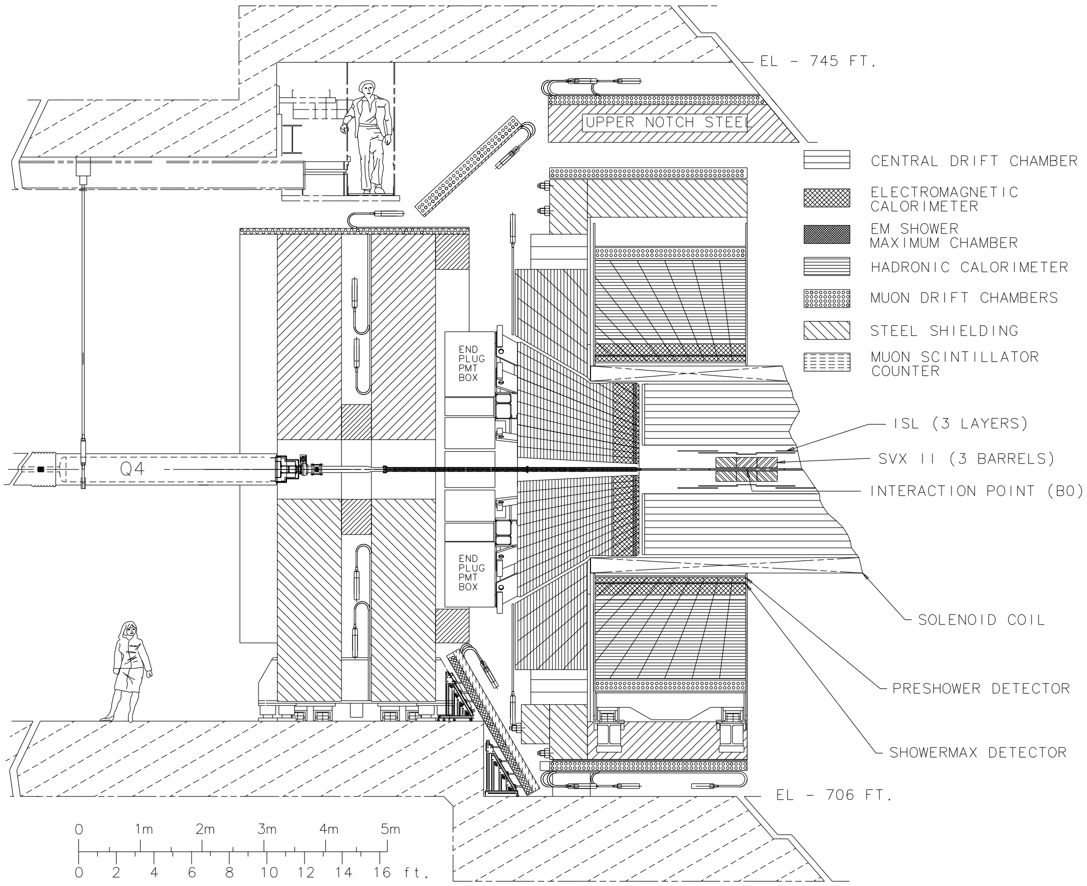

Figure 1: *Schematic view of the CDF detector as configured for Run II.*

In the Standard CDF coordinate system, the positive z axis is in the direction of the protons (roughly towards the east), the positive y axis points up, and the positive x axis points out of the ring. The polar angle θ is zero at the z axis and ϕ is zero along the x axis and increases from the positive x axis to the positive y axis.

3 DAQ Overview

3.1 Front End Crates

Data from the detector is sent to cards (sometimes called modules or boards). An example would be a TDC card or an ADMEM card. The cards reside inside front end crates.

Usually, each card is associated with a specific detector component. (A component is defined to be a specific subdetector. For example, COT, CMU, CES are components.) Within a crate, the data from all cards related to a specific component are put into one data block. Usually, there is one block per crate for each component with cards in the crate. (Although there are exceptions to the one block per crate rule and sometimes one card will handle more than one component.)

Then, the front end crate creates a minibank out of each block. A block is made into a minibank by adding 5 header words to the beginning of the block. The header words are a bank name, bank number, version number, bank length, and bank type. This is exactly the YBOS/TRYBOS bank structure described in Section 4, with one exception. The total number of minibanks for a given component is added to the upper 16 bits of the "Bank Number" field in the minibank header. The lower 16 bits hold a bank number for the minibank. This number usually runs from 1 to the total number of minibanks for this component (although there are a few exceptions to this rule).

3.2 The Event Builder and Level 3

Data from the Front End Crates is formatted as minibanks and sent to VRB modules sitting in the Data Concentrator Crates (DCC). Each VRB module has 10 input links and will take all the data from up to 10 front end crates. Each Data Concentrator Crate holds several VRBs and one SCPU. All the VRBs send their data to the SCPU in their crate. The SCPUs send the data through a 16×16 ATM switch to a converter node in the Level 3 PC farm. For a single event, all the data from all the SCPUs is sent to a single converter node in the Level 3 PC Farm (although there are many converter nodes and different events will be spread around the different converter nodes). From this point forward the data for an event will be kept together.

The converter then sends all the data for an event to a processor node. The processor node runs two processes on the event. First it runs the reformatter, then it runs the Level 3 filter. Events which pass the filter are sent to a Level 3 output node. The output node sends the events to the Consumer Server Logger (CSL), which directs the events to tape and/or disk as appropriate.

The bank format does not change from the output of the front end crate to the input of the reformatter process. Between these points, the data is always formatted as minibanks. The minibanks are simply passed from one place to the next.

The reformatter takes all the minibanks for a component (for example COT) and converts them to a single bank (for example COTD). The reformatter takes all the data blocks in the minibanks and concatenates them in order of the minibank numbers. The reformatter creates a block pointer section in front of the datablocks. It also puts a new bank header section in front of the block pointers. The reformatter output is in YBOS/TRYBOS format.

The numbering conventions for the minibanks, blocks, and crates are a little complicated. The data blocks are always numbered in the order they appear in the final bank array starting from 0. This is a possible source of confusion, because the minibank numbers usually start at one (a bad design choice, probably too late to change it). Usually, the minibank number is equal to the corresponding block number PLUS ONE. There is also a crate number that identifies the hardware crate where the minibanks are created. The convention for numbering crates is different for each subdetector. The most common situation is that crate number is the same as minibank number, but there are many exceptions to this. Be careful.

Since all the information describing the banks has been added in the Front End Crate, the Reformatter process is completely data driven. The reformatter code is the same for all banks.

Of course there are exceptions. The silicon DAQ is different than all the other DAQ. There are also a few exceptional raw data banks that are not built from minibanks (like LRIH and TSID). These banks either do not have block pointer sections or the block pointer sections are different than for standard raw data banks.

The second process running on a Level 3 processor node is the Level 3 filter. This process runs offline reconstruction code and selects events passing the Level 3 criteria. The input to the Level 3 process is in YBOS/TRYBOS format. On input, the format is converted to the EDM2 format. The EDM2 format is used in the reconstruction in Level 3, in offline reconstruction, and in the data files saved to disk and tape.

The data flow is illustrated in Figure 2.

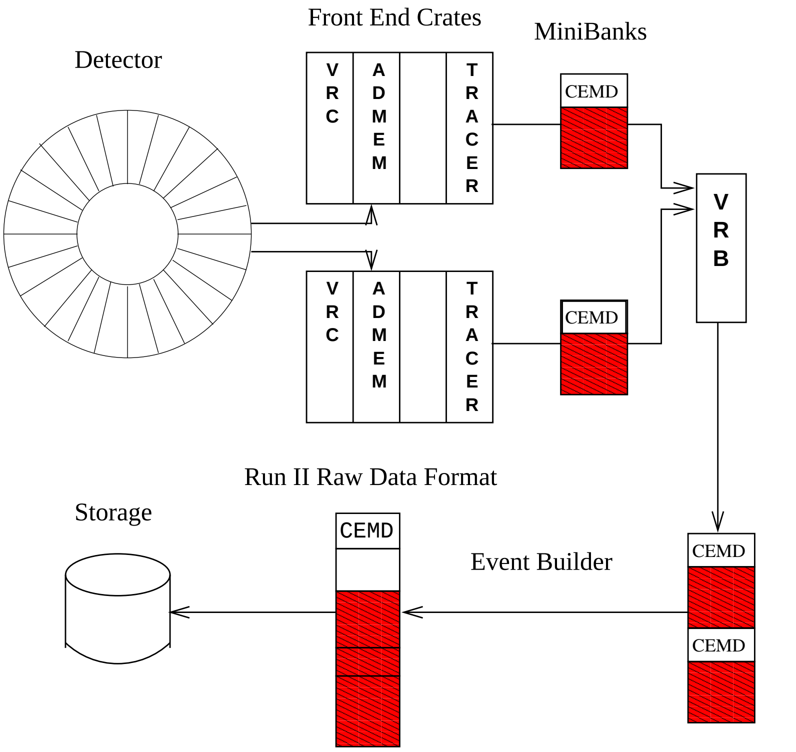

Figure 2: *The data from from the front end crates to storage.*

3.3 Software Event Builder

The software event builder is an alternative to the hardware event builder. The VRB modules, SCPUs, ATM switch, and Level 3 PC farm are not used by the software event builder. Data goes straight from the front end crates to Software Event Builder then straight to the Consumer Server Logger (CSL). The software event builder converts the minibanks into YBOS/TRYBOS format and that is output to CSL. One of the first things the CSL does is to spawn an AC++ process that converts the YBOS/TRYBOS format to EDM2 format. Then the data is stored on disk or tape in EDM2 format. The Level 3 filter is not run on events built with the Software Event Builder. This is the state of the Software Event Builder as of 5 November 2001.

4 YBOS Data Structure

This section describes the features of the YBOS/TRYBOS format that are common to all bank types. The YBOS/TRYBOS bank structure is shown in Table 1. Except for the version number, this is identical to the run 1 format. Further information on the run 1 formats can be found in [2]. In run 2, the data is in this YBOS/TRYBOS format only for a brief time between the reformatter output and the input of the Level 3 filter. The differences between the YBOS/TRYBOS format and the minibank and EDM2 formats are described in Section 5 and Section 6. Note that this document describes all the individual bank type formats in terms of the YBOS/TRYBOS format, mostly for historical reasons.

The first word in the bank is its name. The first 3 characters of the bank name identify what data is stored in the bank (frequently a component like COT or CES). The last character for raw data banks is usually a D. There are two exceptions. For calibration banks, the last character is a C. For pedestal banks, the last character is a P.

The bank number is not used in run 2 raw data. It's purpose is to identify different banks with the same name. In the run 2 raw data, there is usually at most one bank for each possible name.

The third word is the bank version. This is discussed in Section 7.

The bank length is the number of 4 byte words in the bank, excluding the header words (the first four words), but including the type word, the words in the block pointer section, and the words in the data blocks. This is a little confusing because the bank length counts 4 byte words even in a bank that holds I*2 data words. On the other hand, the block pointers count words based on the type, either 2 byte or 4 byte words.

Bank Name (4 ASCII Characters)
Bank Number
Bank Version
Bank Length
Bank Type
Group 1 Type (if appropriate)
...
Group n Type (if appropriate)
Bank Data Word 0
...
Bank Data Word n

Table 1: *Structure of a YBOS bank. Except for the data words, each line represents a 32 bit word. The data word size is given by the "Bank Type" word, usually 16 bit or 32 bit.*

Raw data banks are all monotype banks. This means that there are not any "group type" words. Usually the "bank type" word is simply 1 or 3 to indicate that the bank data members are type I*2 or I*4. The definition of the bitfields in the "bank type" words can be found in Appendix D of [2] and is given in Table 2. The definition of the "Type ID" value is given in Table 3.

0-7	Bank/Group Type ID
8-15	all zero
16-31	No. of Group Type Words which follow

Table 2: *The bit assignment of the type word.*

Bank/Group Type	Description
1	16 bit Integer (I*2)
2	8 bit ASCII Characters
3	32 bit Integer (I*4)
4	32 bit Real (R*4)
5	64 bit Real (R*8)

Table 3: *Coding for the Bank Type field.*

4.1 Block Pointer Section

Immediately following the header and type sections is a block pointer section. The block pointer section is part of the bank data and is the same type as specified in the data type word. Most raw data banks will either be I*2 or I*4 monotype banks. The format of the pointer section is specified in Table 4. More details about the Block Pointer section can be

Number of Blocks (=N)
Pointer to Block 0
...
Pointer to Block N-1
Pointer to End of Data

Table 4: *Format of the Block Pointer section. The data type is determined by the Type word in the header section. Each row can be either a 16 bit word (I*2) or 32 bit word (I*4).*

found in the sections describing the named banks.

Pointers are specified as offsets in word counts starting from the beginning of the data section (which is the block pointer section). The words are counted based on the size indicated in the Bank Type word, usually either 2 or 4 bytes. In general, data blocks can be empty. The size of a block is the difference between the next pointer and the current pointer. The 'End of Data' pointer is the index to the word just past the last word in the bank. An example of a correctly filled pointer section is given in Table 5. The example corresponds to a bank having four blocks with Block 0 having five data words, Block 1 is empty, Block 2 has two data words and Block 3 is empty.

There are a few raw data banks that do not have a block pointer section, but most raw data banks have it. For example, LRIH and TSID do not. The block pointer section for the silicon banks is different. These unique features are described in the sections for those bank types.

Offset	Description	Value
0	Number of Blocks	4
1	Pointer to Block 0	6
2	Pointer to Block 1	11
3	Pointer to Block 2	11
4	Pointer to Block 3	13
5	Pointer to End of Data	13
6	Block 0 data 0	
7	Block 0 data 1	
8	Block 0 data 2	
9	Block 0 data 3	
10	Block 0 data 4	
11	Block 2 data 0	
12	Block 2 data 1	

Table 5: *Example of the Block pointer section and data section.*

There is an additional complication for I*2 banks. All blocks are required to start on 32 bit word boundaries. If a block pointer section or a block has an odd number of words, then an extra padding word is inserted into the block or block pointer section. The padding word is placed at the end of the block or block pointer section.

5 Minibanks

Minibanks are the same as normal YBOS banks, with a couple of exceptions. The "Bank Number" word is divided into two bitfields. The upper 16 bits contain the total number of minibanks for a component. The lower 16 bits contain the bank number associated with the minibank. Usually, these bank numbers run from 1 up to the total number of minibanks for the component (although there are exceptions to this rule).

In a minibank, there is only one block of data and it starts immediately after the "Bank Type" word. There are no block pointers. See figure 6.

The reformatter in Level 3 or the software event builder takes all the minibanks for a component, strips and deletes the minibank headers, concatenates the blocks in order of minibank number, creates a block point section and adds that to the beginning, and adds the YBOS header and type words to start the new bank. As was noted earlier, minibank bank numbers start at 1, but blocks are numbered starting at 0.

Bank Name (4 ASCII Characters)
Bits 0:15 Bank Number. Bits 16:31 Total Number of Minibanks
Bank Version
Bank Length
Bank Type
Block Data Word 0
...
Block Data Word n

Table 6: *Structure of a minibank.*

6 EDM2

EDM2 is the format that is used offline and also online after the input module of the Level 3 filter. The data saved to tape and disk is almost always in this format. As of 5 November 2001, all data and simulation is stored in EDM2 format (or at least it is supposed to be). Most data has been in EDM2 format since before the commissioning run in the fall of 2000.

The EDM2 bank format is very similar to the YBOS/TRYBOS bank format. In the C++ code, each bank with a different name has its own class. For example, COTD_StorableBank is the class that holds banks named "COTD". In memory, the StorableBanks classes contain an array that is identical to the YBOS/TRYBOS array except that the array starts with the type word. The bank name, bank number, bank version, and bank length are also stored in the StorableBanks class, just not in the array. They are separate data elements. They are all accessible and their values can be set when creating a new bank.

The streamer function in the StorableBanks base class passes all these data elements (the array and the other elements from the YBOS header) to a ROOT buffer. ROOT persistently stores all data on disk or tape. ROOT can restore the data from disk or tape to memory. A detailed discussion of how ROOT does this is beyond the scope of this document.

In EDM2, the banks are usually identified by their type, not the 4 character bank name. One usually searches for a COTD_StorableBank class, instead of searching for a bank with the 4 character string "COTD" as a name. The 4 character string is used when converting from YBOS format to EDM2 format in the level 3 filter. After that the name string is saved and accessible, but usually it is not used.

In EDM2, the bank number is not usually used. The StorableBank class inherits from the StorableObject class. StorableObject has additional data members that hold a description string and process name as well. The description string and process name do not exist in the YBOS/TRYBOS format. They are added when the conversion to EDM2 occurs. For raw data, the process name is set to "RAW". There is usually only one raw data bank of each type per event. In simulation, the process name and description can be set to anything. The description and process name are normally used to differentiate between raw data banks of the same type. The bank number is saved and it is possible to access it. One just normally does not.

7 Bank Versions

Banks are versioned. In the minibank format, the version number is the third word in the header of each minibank. When the reformatter combines them into one bank, the same version number is copied into the third word of the YBOS/TRYBOS bank array (all the minibanks used to create one YBOS bank should have the same version). When the conversion from YBOS/TRYBOS to EDM2 is made, this value is copied from the bank array to a variable called _bank_version in the StorableBank base class.

The bank version is accessed offline using the bank_version() accessor function in the StorableBank base class. All the raw data bank classes inherit from the StorableBank class. There is another function called class_version() in the StorableBanks class. class_version() is not related to the bank version and not something most people need to know about or use.

The bank version is incremented when the format of the words in the bank array changes. It should be incremented so that people accessing the data can write code to access all versions or at least generate some kind of error for nonsupported versions. As of 5 November 2001, all the bank version numbers are still at 0. In this document, the sections on the individual bank types will specify the version number if and only if there exist nonzero versions.

The version number for a bank in the data is changed by modifying the code that runs in the front end crate and builds the minibank. In simulation, the set_bank_version(version) function in the StorableBank class should be called when creating the bank. When a bank version number changes many other things should also be done. Version changes should be documented in this note (CDF 4152). All interesting versions that exist in data on tape or disk should be documented. Also, when the format of a bank changes all the software that accesses and uses the bank must be appropriately modified. The simulation software should also be updated.

8 Event Record Banks

8.1 LRIH - Logical Record Identifier/Header Bank

The LRIH bank is the first bank in any logical record and has a fixed size. The format of the bank is specified in Table 7

"LRIH"
Bank Number
Bank Version
Bank Length
Bank Type (I*4)
Run Number
Run Type
Record Number
Trigger Number
Run Section
Record Type
Creating Process?
Output Stream
Creation Time Stamp

Table 7: *Structure of the LRIH bank.*

- Run Number
  The run number is unique and increases monatonically.

- Run Type
  The definition of the Run Type word is specified in Table 8 Valid values for the Experiment Type, and Run Type fields are defined in the Banks product in the LRID_Bank_Utils.hh header file. *Note that the above mentioned file should be coordinated with Jim Pangburn's definitions of run_type. Following discussions 14-JAN-98 TWG meeting, the format of this word was changed to add the bunch ID here instead of in the EVCD bank.*

Bit	Field
31 - 28	Unused
27 - 20	Bunch ID
19 - 16	Experiment Type
15 - 12	Partition Number
11 - 0	Run Type

Table 8: *Structure of the Run Type word.*

- Record Number
  This records index into the current file

- Trigger Number Also know as the event number, this is the unique number given to the event in the online system. If the record is not an event record, this field is set to zero.*Comment made at 14-JAN-98 TWG meeting, This field may need to be 64 bits in run2*

- Run Section
  Specifies a period of constand integrated luminosity.

- Record Type
  Identifies the logical record as an Event, Calibration Record, ... Valid values for the record type field are defined in the Banks product in the LRID_Bank_Utils.hh header file.

- Creating Process
  Identifies the process which created this record. *What should this be? In run 1a it contained info. like the process number of the data logger wich created it. In 1b it also contained the version numbers of A_C and offline that were used. Do we still want to keep it? A more useful bit of info. would be a data format identifier. For example PAD, DST, RAW, BPAD or we could use the run1 scheme for file extensions (psia_7b). If sofware version numbers are to be kept on the data then should it be the numbers of the software for the datalogger (meaningless if you ask me), or should it be the version numbers of the last software that wrote something to the record? If we store a data set identifier then this field will change at each step of the processing chain.*

- Creation Time Stamp
  Date of creation seconds since 00:00:00 UTC, January 1, 1970.
  Is there anything else desired here? Would a micro-time stamp be usefull?

8.2 TSID - Trigger Supervisor Data Bank

The TSID bank contains the Trigger Supervisor data for each event. The format of the bank is specified in Table 9. The TSID bank is formed from the message sent from the Trigger Manager to the Scanner Manager. The Trigger Manager creates the contents of the message from the contents of the Trigger Supervisor Scan FIFO.

"TSID"
Bank Number
Bank Version
Bank Length
Bank Type (I*4)
Trigger Supervisor Word 0
Trigger Supervisor Word 1
Trigger Supervisor Word 2
Trigger Supervisor Word 3

Table 9: *Structure of the TSID bank.*

- Trigger Supervisor Word 0

  The definition of Trigger Supervisor Word 0 is given in Table 10.

  Note that the 8 MSB of the Event ID are placed in Trigger Supervisor Word 3. The Event ID starts at zero (0) for each run, and is incremented by one (1) for each Level 2 Accept broadcasted by the Trigger Supervisor to the partition (i.e. when the DONE handshake is initiated.) The 4 LSB of the Event ID are sent with the broadcasted Level 2 Accept and are expected to appear in the data stream from each front-end or trigger crate.

  The Message ID should equal 1, indicating a successfully read out event (i.e. no DONE TIMEOUT).

Bit	Field
31 - 28	Message ID
27	0
26 - 24	Partition Number
23 - 0	Event ID (24 LSB)

Table 10: *Structure of Trigger Supervisor Word 0.*

- Trigger Supervisor Word 1

  The definition of Trigger Supervisor Word 1 is given in Table 11.

  The Bunch Counter and Board Header follow the definitions laid out in CDF notes 3145 and 2388.

Bit	Field
31 - 8	Board Header
7 - 0	Bunch Counter

Table 11: *Structure of Trigger Supervisor Word 1.*

- Trigger Supervisor Word 2

  Trigger Supervisor Word 2 contains a 32 bit Turn Counter. The Turn Counter starts at zero (0) for each run and is incremented by one (1) for each passing of the Bunch Zero (B0) marker.

- Trigger Supervisor Word 3

  The definition of Trigger Supervisor Word 3 is given in Table 12.

  The Markers indicate the status at the time of collision for the accepted event.

  The Calibration Trigger Flag is set for events accepted for purposes of calibration; the Calibration Type still needs to be defined. The Calibration Type equals the value sent by the Trigger Supervisor to the front end when initiating a calibration event.

  The Event Sequence Number will be 0, 1, 2, 3 (in order) when a set of four events is taken in consecutive 132 nsec periods for each Level 1 Accept (‘Myron Mode’). This field is only relevant when running in this mode, which must be determined elsewhere, by Run Condition settings.

The Level 2 Buffer Number is the value that was sent by the Trigger Supervisor to the front end when broadcasting the Level 2 Accept.

The Readout/Scan List Type needs to be defined to indicate if a more or less verbose readout of data is to be expected. Zero (0) indicates the default readout.

Note that the 24 LSB of the Event ID are in Trigger Supervisor Word 0.

Bit	Field
31 - 24	Event ID (8 MSB)
23 - 19	0
18 - 16	Readout/Scan List Type (0 is default)
15 - 14	Level 2 Buffer Number
13 - 12	Event Sequence Number (in 'Myron' Mode)
11 - 5	Calibration Type
4	Calibration Trigger Flag
3	ABORT Marker
2	Trigger Inhibit Marker
1	Bunch Crossing Marker
0	Bunch Zero Marker

Table 12: *Structure of Trigger Supervisor Word 3.*

9 Banks with Common Formats

Components that share the same readout electronics have the same data format. Table 13 shows the four major data format categories. There are some unique banks that do not fall into any of the categories, but most do. Every attempt is made to make all raw data bank formats as similar as reasonably possible to one of these formats. For example, most trigger banks have a unique format, but they are given a data format that has a pointer structure like the TDC banks.

Readout	Detector	Bank Name	Banks Type
QIE	Central CAL Plug CAL Endwall	CEMD, CHAD PEMD, PHAD WHAD	I*2
TDC	MUON COT	CMUD, CMXD CMPD, BMUD COTD	I*4
SVXII	ISL SVXII	ISLD SIXD	I*2
Calibration	Calorimeters	CHAC PEMC	I*4

Table 13: The four major data format categories for the raw data banks along with some banks in each category. This is not a complete list of all banks in each category, just a few of each type.

9.1 QIE Data Format

The QIE banks have a header and block pointers as described in Sections 4 and 4.1. This section describes the content of the blocks.

The format of a QIE data block is specified in Table 14. These are I*2 type banks (they hold 16 bit integer words). Each row in the table represents two 16 bit words, first a Channel ID and then a data word. The Channel ID is added by the VME Readout Controller (VRC). The bit assignment for the Channel ID and the meaning of the data word vary. They are specified later in the sections that describe the data format for each individual detector.

Channel ID	Data 0
Channel ID	Data 1
...	
Channel ID	Data n

Table 14: *Structure of the block of data from QIEs coming from one Front End Crate. One row represents two 16 bit words, first a Channel ID and then a data word.*

There is an alternate format. This alternate format uses clusters. A cluster is simply a group of data words that have consecutive Channel IDs. For a cluster, space is saved by only writing a channel ID for the first channel. Then as one looks at subsequent data words, one increments the channel ID associated with the previous data word by one. For all QIE detectors, the upper three bits of the channel ID are reserved to hold the number of channels in the cluster. This is also called the cluster width. This places a couple restrictions on the clusters. Three bits hold a number in the range 0 to 7. The largest cluster can have 7 data words. This also requires that the Channel IDs in a cluster be sequential. But some of the Channel IDs are defined as bitfields and there are gaps in the numbers. Clusters cannot span these gaps.

If the upper three bits of the Channel ID are zero, then only one data word follows. Otherwise, the cluster width gives the number of data words that follow. Data blocks are expected to be aligned on a 32 bit words. Padding is added for blocks with an odd number of words. Table 15 shows this clustering format. The offline code is being written to support either this format or the one above transparently (at least offline code that uses the StorableBanks packages). As of this writing, this format is not produced by either the online or the simulation.

There is yet another alternate format. This format is not supported by offline code and should never be written into physics data. Online groups might use it for debugging and testing. In this format, there are no channel IDs and a data word is written for all possible channels. Data words are associated to channels by their location in the block. See Table 16.

4|ChID	data
data	data
data	0|ChID
data	2|ChID
data	data
1|ChID	data
2|ChID	data
data	PAD

Table 15: *Clustering example. The numbers in the table indicate the cluster width stored in the top 3 bits of the Channel ID. The channel width indicates the number of consecutive data words which follow (except 0 indicates that one data word follows). Padding is inserted so that the next data block will be aligned on a 32 bit word.*

Data 0	Data 1
...	
Data n-1	Data n

Table 16: *An alternate format used only in the online for testing and debugging. This should never appear in physics data and the offline code cannot interpret it. In this format, a data word is written for every possible channel. No channel IDs are written. Channels are associated with data by their location.*

9.2 TDC Data Format

The TDC banks have a header and block pointers as described in Sections 4 and 4.1. This section describes only the content of the blocks.

The same TDC is used for the COT, CAL, and MUON detectors. Only the common elements of the format of data from these detectors are described in this section. Later, there are sections describing the format of each type of TDC bank.

There are two versions of the format of data from a TDC. Version 0 was used up until October 2003. The plan is to start using version 1 sometime around November 2003. Online, the bank version will be set in the bank header. Offline, one can distinguish the versions using the bank_version() accessor in the StorableBank class. In both versions, the bank format is composed of 32 bit integer words (in other words I*4 monotype).

The format for version 0 is specified in Table 17. The format for version 1 is specified in Table 18. In both versions, the first word specifies the number of TDCs contained in the block. Next comes a set of pointers giving the index of the start of the data for each TDC. Each pointer specifies the number of 32 bit words from the "Number of TDCs" word to the first word for that TDC. The last pointer gives the index of the word just beyond the last word in the block. The format of the header word is specified in Table 19 and is the same in both versions. The format of the TDC hit word is specified in Table 20.

In version 0, there are never pad words in the bank. In version 1, the pad words all have the hex value 0xFFFFFFFF. In version 1, the pad words may or may not be present in the bank. They are allowed, but not required. Code accessing the bank should work equally well whether the pad words are present or not present.

The version 1 format was added to this note on 13 October 2003. Offline software that existed on that date will fail when it tries to read the version 1 format. The format changes are related to modifications to the TDC DSP code that are designed to speed up data readout. One change involved using "Dual-ported-FIFO" readout mode. To use this mode, the hardware requires that the number of words in a TDC section be a multiple of 8 bytes. Therefore, one pad word will added as necessary. These pad words may not be present in simulation or data banks that have been compressed and uncompressed. In addition, the word order is changed and the version 0 first flag bit is always set to zero in version 1.

Each TDC chip records tracking chamber wire transitions, either low to high (as 0 1) or high to low (as 1 0). These leading or trailing edges are stored along with the time they passed out of the TDC digitizing buffer relative to a L1 Accept, up to a maximum time of 2047 ns. The width of a pulse is determined from the time of a trailing edge relative to the preceding leading edge. If the TDC recorded an unpaired leading edge the DSP fills bits (21 11) with 1s, if the TDC recorded an unpaired trailing edge, the DSP sets the pulse width to 0.

- *First hit flag*: In version 0, this bit is on (=1) for the hit with the earliest leading edge time in each channel. The bit is always set to 0 in version 1.
- *Unpaired leading edge*: leading edge = leading edge time; width = 2047.
- *Unpaired trailing edge*: leading edge = trailing edge time; width = 0.

The CDF clock applies a bunch zero marker to one of the 159 bunch positions of the Tevatron. Data resulting from an interaction involving this bunch will be tagged in the data with a bunch ID of zero. For collisions involving subsequent bunches, the bunch ID value is

Number of TDCs
Pointer to first TDC
...
Pointer to last TDC
Pointer to End of Block
Header Word TDC 0
Hit Word 0
Hit Word 1
...
Hit Word n
Header Word TDC 1
Hit Word 0
Hit Word 1
...
Hit Word m
...
...
Header Word Last TDC
Hit Word 0
Hit Word 1
...
Hit Word k

Table 17: *Structure of one block of data coming from TDCs located in one Front End Crate. A TDC bank can contain multiple blocks. This is the bank version 0 format.*

incremented appropriately. The bunch ID is utilized to keep all of the readout electronics in synchronization.

The L2 accept value is simply a value between 0 and 3 which indicates which L2 buffer was utilized for a particular event. Scan type can be used to indicate different readout formats. The trigger supervisor requests a specific readout format for each event depending on the type of run (normal run, calibration, full diagnositc mode, etc...). At the moment the different scan list types have not been well defined.

All data words for a channel will be together. There will be nothing between the first and last data words of a channel other than data words from that channel.

Nothing else can be assumed about the format of TDC banks. Code written to access TDC banks should not depend on anything not explicitly specified above. There should not even be any subtle dependences. The TDC bank format for a given detector may vary in the following ways:

- The number of blocks may vary.
- The order of blocks may vary.
- Empty blocks may or may not have a block pointer.
- Assignment of TDCs to blocks may vary.

Number of TDCs
Pointer to first TDC
...
Pointer to last TDC
Pointer to End of Block
Hit Word 0
Hit Word 1
...
Hit Word n
Pad Word (may or may not exist)
Header Word TDC 0
Hit Word 0
Hit Word 1
...
Hit Word m
Pad Word (may or may not exist)
Header Word TDC 1
...
...
Hit Word 0
Hit Word 1
...
Hit Word k
Pad Word (may or may not exist)
Header Word Last TDC

Table 18: *Structure of one block of data coming from TDCs located in one Front End Crate. A TDC bank can contain multiple blocks. This is the bank version 1 format. See the text for comments related to the pad words.*

- The order of TDC modules may vary.
- For TDC modules that are not readout, there will be no pointer, header word, or data words. For TDC modules that are readout, there will always be a pointer and header word, but frequently there will be no data words.
- The order of channels in a TDC may vary.
- The time order of data words in a channel may vary. This means the location of the first flag bit may also vary.
- Channels in one TDC module may be split to more than one module with all given the same module ID.

It is a requirement that all code that reads TDC banks be able to handle all the variations listed above for each bank. This flexibility is needed to optimize the online DAQ system. One can find information in other places that specifies some of the things listed above.

Bit	TDC Header Word
31:23	Module ID
22:20	Scan type
19:18	L2 accept
17:8	Number of TDC hit words following this word
7:0	Bunch ID

Table 19: *Bit assignment for the TDC header words. The module ID is defined differently for each detector. These definitions are in the sections that describe the unique features of specific TDC banks. Note that bits 8:17 contains the number of hit words only (not header or pad words).*

Bit	TDC Hit Word
31	First hit flag (version 0 only)
30:22	TDC Channel ID
21:11	Width (ns)
10:0	Leading edge (ns)

Table 20: *Bit assignment for the TDC hit word. In version 1, bit 31 is always assigned the value 0. The channel ID is defined differently for each detector. These definitions are in the sections that describe the unique features of specific TDC banks.*

There are CDF notes. The online hardware database contains many of these details. These things should be viewed as a current view, not as a permanent requirement on the TDC bank formats. These things may change over time, but the software that accesses the TDC banks should not need to be changed.

The items listed above may vary for many reasons. Over time, the electronics configuration may change. There are different modes of operation of the DAQ system that cause some of the variations. In addition, there will be no attempt to keep the order and number of blocks, modules, channels, and data words the same in the data and simulation. The simulation code will be written to create the bank in some plausible order. Once that order is set, the simulation will not be modified when the variations listed above occur in the data.

9.3 SVX3 Data Format

9.3.1 HDI block

Since SVXII, ISL, and Layer 00 use the same readout chips (SVX3) and DAQ, the banks produced from these components have similar underlying data structures which are described here. The basic data block corresponds to a single HDI (High Density Interconnect). The organization of an HDI block is shown in Table 21.

The data begins with an 8 bit *FIB ID0* which is common to all HDIs within a FIB (Fiber Interface Board). It is followed by a second 8 bit *FIB ID1* indicating which of the 10 pipeline processors in that FIB handled the data for this HDI. The two FIB IDs are downloadable and will be set to encode the Barrel number in ID0, and the port card and HDI number in ID1. The third byte is the bunch crossing number. The fourth is the back end state. The back end state byte has its upper six bits set high and its lower two bits specify one of up to four states the chip was in when the data was acquired. The number of chip states with significantly different behaviour needs to be studied with the full detector before a final choice for this parameter is set.

Next comes the 8 bit *Chip ID* where the upper two bits are used to identify the byte. Next is a 8 bit *Status* byte, which is the Cell Id of the capacitor used for this event by this chip. Next come pairs of bytes specifying the *Channel ID* and the associated data. This repeats for all chips attached to the HDI. A Chip ID and status byte are generated for all chips, even chips for which no data is present. The last chip in the HDI chain is programmed such that it will always read out its last channel and whatever the data happened to be for that event. Finally, there are two end of record (EOR=1100XXXX) bytes and bytes of 0s for padding until the block size is a multiple of 64 bits.

FIB ID0	FIB ID1
Bunch Crossing #	BE State
Chip ID	Status
Channel ID	Data
...	
Channel ID	Data
Chip ID	Status
Channel ID	Data
Channel ID	Data
...	
Channel ID (0x7f)	Data
EOR	EOR
00	00

Table 21: *Structure of an HDI Block, the basic block of SVX3 data. Each line in the table represents a 16 bit word, with the most significant byte first (left). The total data block is padded to a multiple of 64 bits by bytes set to zero.*

- FIB ID0
  One byte containing the detector ID, barrel number and ladder end. The bit assignments are specified in Table 22 and noted in Fig. 3. For SVXII and ISL, the barrel

number bits [2:1] and ladder end bit [0:0] can also be thought of as 3 bits [2:0] counting bulkheads (or equivalently "halfbarrels") moving West to East in z.

Bit Range	Definition
7:5	= 111 to indicate FIB ID
4:3	0 for SVXII, 2 for ISL, 3 for L00
2:1	Barrel number (0=West, 1=Central, 2=East SVX & ISL) (0 = West, 1 = East L00)
0:0	Ladder end (0=West, 1=East, 0 for L00)

Table 22: *Bit assignments for the FIB ID0 byte.*

- FIB ID1
  One byte specifying the portcard number and HDI number which are used to encode the phi and radial location of the ladder producing the data. The encoding differs between SVXII, ISL, and Layer 00. It is straight forward for SVXII, but it is complicated for ISL and Layer 00 since those detectors were designed after the DAQ layout was finalized and have very different geometery than SVXII. The bit assignments for FIB ID1 are shown in Fig. 3. The conversions to phi address and layer number are specified in the figure.

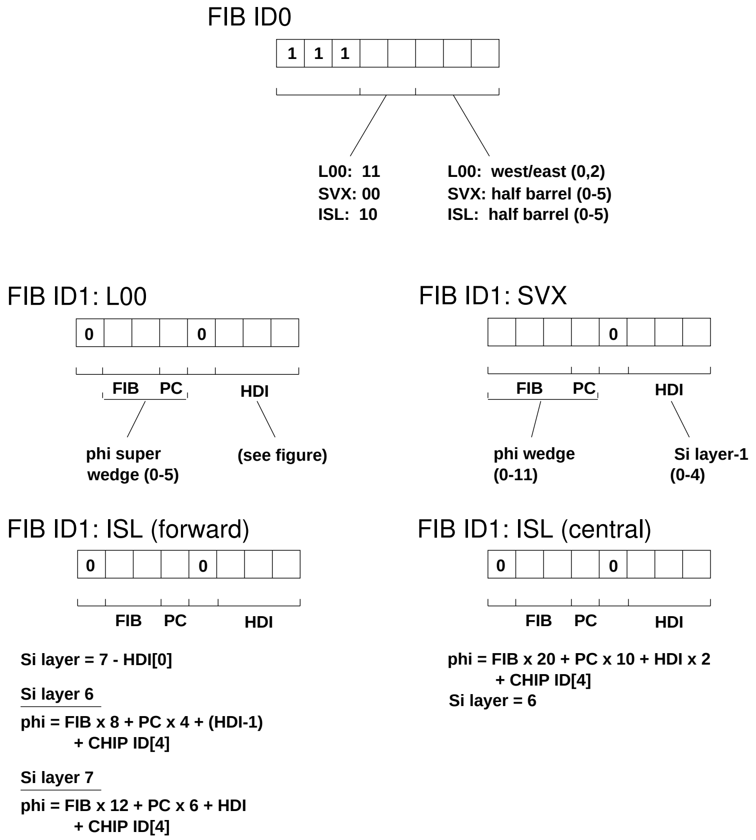

Figure 3: *Bit assignments for the FIB ID0 and FIB ID1 bytes are specified for each of the detectors; L00, SVXII, ISL.*

- Chip ID
  One byte containing the Chip ID number and identifying itself as a Chip ID byte. See Table 23.

Bit Range	Definition
7:6	= 10 to indicate Chip ID
5:5	$r - \phi$ or $r - z$ Chip Bit
4:0	Chip Identifier

Table 23: *Bit assignments for the chip ID byte.*

  The lower nibble of the chip id simply counts down from the nChips-1 to zero as the readout in an HDI progresses. The offline does not use the id to determine chip location. Instead, it relies on position within the readout stream since this is hardwired into the readout logic and cannot change, whereas for hardware reasons (stuck bits etc) the chip id can change.

- Status
  One byte of information following the Chip ID indicating the pipeline capacitor this chip read out for this event (0-45)

- Channel ID
  One byte containing the channel number and identifying itself as a Channel ID byte. See Table 24.

Bit Range	Definition
7:7	= 0 to indicate Channel ID
6:0	Channel # (0-127)

Table 24: *Bit assignments for the Channel ID byte.*

- Data
  One byte of pulse height information for this channel in ADC counts. Un-Gray Coding and pedestal subtraction have already been performed by the FIB. In order to utilize most of the dynamic range of the SVX3 chip (0-255 ADC counts), but also allow for a valid representation of (some) negative values (after pedestal subtraction, of course), the following data representation was chosen. Values within [0,239] ([0x0,0xEF]) are mapped onto themselves, i.e. treated as unsigned integers. Channels for which the data was within [0,239] but exceed 239 after pedestal subtraction (negative pedestals are possible in DPS ON mode) are mapped to 239. Since pedestals in DPS ON mode typically have small absolute values, this does not introduce a large error on the pulse height from that channel.

  The last 16 values, with the all bits in the upper nibble set ([240,255], i.e. [0xF0,0xFF]), are interpreted as either negative (regular 2's complement representation), or special cases listed in Table 25. Thus checking if the upper 4 bits (rather than just the upper bit in 2's complement) is an easy way to check if the data has any chance of passing a threshold cut.

Of the 16 values above 0xF0, 12 are used as negative data-pedestal values for study of pedestal stability. 0xF0 represents a saturated channel, for which the true ADC count is unknown. 0xF1 represents an out-of-range ADC count returned from the chip, which indicates a hardware error. 0xF2 marks the channel as unusable because it is either on the dead/hot channel list, or an underflow (< -12 after pedestal subtraction). 0xF3 is marks a channel which returned 0 ADC counts in DPS ON mode only. The true ADC count for such a channel cannot be determined, and so is mapped to this value.

To summarize, channels with data in the range [-12,239] can be used as-is, -13 and 240 can be used with care, and -14 and -15 are unuseable channels.

Unsigned value	Interpretation
0 (0x0)	0 (WYSYWIG pass-through)
1 (0x1)	1
...	...
238 (0xEE)	238
239 (0xEF)	Data was valid (within [0,counter mod-1]) but pedestal subtraction (of a negative pedestal) resulted in an overflow (> 239)
240 (0xF0)	Channel was saturated (at counter modulo) before pedestal subtraction
241 (0xF1)	Error: Data from chip greater than counter mod
242 (0xF2)	Channel killed by FIB (i.e. marked dead or hot) or underflow (≤ -13 after pedestal subtraction)
243 (0xF3)	Data from chip = 0 in DPS ON mode
244 (0xF4)	-12 (WYSYWIG pass-through, two's-complement representation)
245 (0xF5)	-11
...	...
255 (0xFF)	-1

Table 25: *Mapping between FIB output and ADC count/error codes for SVX data.*

- EOR and Other
  There are five types of EOR bytes defined, all of the form 1100XXXX. Other byte types are not currently written into the D banks, but are reserved for future use.

  11000000 Invalid - EOR
  11000001 Normal - EOR
  11000010 Truncated - EOR
  11000100 End Fill - Should never be seen - EOR
  11001000 Abort - EOR
  11010000 Interchip Fill - Should never be seen
  11100000 Reserved
  11xxxxxx Reserved

9.3.2 VRB block

The second level of SVX3 data formatting is in the VME Readout Buffer (VRB). Each VRB receives data from 10 HDIs in one FIB. A VRB outputs that data to the reformatter in a VRB block as shown in Table 26. The HDI blocks are ordered with increasing FIB ID1 i.e., increasing PC and HDI number.

The VRB Block header contains global event information and information pertaining to the particular VRB. The "Total byte count" is contained in the highest 16 bits of the first word, and includes the 32 header bytes and the 64 bit padding for each HDI Block. The "Time Since L1A" in bits 8-15 and the "TS Counter" in bits 0-7 of that word come from the SRC and should be the same for all Silicon VRB Blocks. Two Error words are next. An Error word is 6 bits for each of 5 channels reporting whether the VRB detected a particular error in that channel, with Channels 0-4 in Error word 0 and 5-9 in Error word 1. Within an Error word, the mapping is Chan $x \rightarrow$ Bits $26 - 6 \times x$ to $31 - 6 \times x$. Starting with the most significant bit, the 6 bits mean the following: Glink Synch Error, Glink Frame Error, Event Synchronization Error, Data Format Error, Invalid SRC Command, and Internal VRB Error. The lowest two bits of each Error Word are unused and should be zero. Following the error words are 10 16 bit fields containing individual HDI Block byte counts, for the HDI blocks. The HDI Block byte counts do *not* include the padding. Missing HDI blocks are indicated by a byte count of zero.

Total byte count	Time Since L1A	TS Count
Error Reporting Word 0		
Error Reporting Word 1		
Byte count of HDI Block 0	Byte count of HDI Block 1	
Byte count of HDI Block 2	Byte count of HDI Block 3	
Byte count of HDI Block 4	Byte count of HDI Block 5	
Byte count of HDI Block 6	Byte count of HDI Block 7	
Byte count of HDI Block 8	Byte count of HDI Block 9	
HDI Block 0 ...		
HDI Block 1 ...		
HDI Block 2 ...		
HDI Block 3 ...		
HDI Block 4 ...		
HDI Block 5 ...		
HDI Block 6 ...		
HDI Block 7 ...		
HDI Block 8 ...		
HDI Block 9 ...		

Table 26: *Structure of a VRB Block. Each line in the table represents a 32 bit word. The details of an "HDI Block" are specified in Table 21.*

10 Calorimeter Banks

10.1 Channel ID and Other Common Features

The calorimeter banks (CHAD, CEMD, WHAD, PEMD, and PHAD) have many common features. These are described in this section.

These banks are all QIE banks and have the format described in Section 9.1.

These banks have the same channel ID bitfields. These are shown in Table 27. Rapidity segmentation is defined by counting towers on one side, starting with 0 for the tower nearest the center of the detector and incrementing by 1 as one counts each tower in order of the absolute value of the rapidity. For the central calorimeters, rapidity segmentation ranges from 0 to 9. For the endwall calorimeter, it ranges from 6 to 11. For the plug calorimeters, it ranges from 10 to 21. The PMT/Tile bit is discussed separately for each detector below.

Normally these banks are written with zero suppression on, which means channels without a significant signal are deleted from the bank. In debugging mode, all channels will be written into the bank. In this case, the 12th bit in the channel ID is set when the channel would have failed the online zero suppression cut.

The cluster width is discussed in Section 9.1.

The blocks in these banks are all written in the same order. First the blocks on the west side are written in phi order, then the blocks on the east side are written in phi order.

The format of these banks is shown in Table 28.

The QIE data word consists of a 16 bit calibrated, linearized number. The upper bit is used as a range bit. When set, the lower 15 bits should be multiplied by 8 to get the correct value. The value is an energy in units of counts.

Bit	Definition
15:13	Cluster width
12	0=Passed, 1=Failed zero suppression
11	West = 0; East = 1
10:6	Azimuthal Segmentation (0-23)
5:1	Rapidity Segmentation (0-21)
0	PMT/Tile

Table 27: *Bit assignment for the calorimeter Channel ID. The azimuthal segmentation is also called wedge number or phi. The rapidity segmentation is also known as tower or eta. This is the Channel ID in the CHAD, CEMD, WHAD, PEMD, and PHAD banks.*

10.2 CHAD and CEMD - Central Calorimeter Banks

The Central Calorimeter consists of 15^o wedges assembled into 4 arches. A north arch and a south arch on the west side. Also a north arch and a south arch on the east side. Each wedge is segmented into towers corresponding to a pseudorapidity interval of about 0.11. Each tower is read out by two photomultiplier tubes. The wedges are further segmented longitudinally into electromagnetic (EMC) and hadronic (HAC) sections. The EMC section consists of alternating layers of scintillator and lead. A two dimensional readout strip chamber is located at the shower maximum (see Section 11). The HAC section consists of alternating layers of iron and scintillator.

<table>
<tr><td colspan="2">“CEMD” or “PHAD” or ...</td></tr>
<tr><td colspan="2">Bank Number</td></tr>
<tr><td colspan="2">Bank Version</td></tr>
<tr><td colspan="2">Bank Length</td></tr>
<tr><td colspan="2">Bank Type (I*2)</td></tr>
<tr><td>Number of Blocks (=n)</td><td>Pointer to Block 0</td></tr>
<tr><td>Pointer to Block 1</td><td>Pointer to Block 2</td></tr>
<tr><td>...</td><td>...</td></tr>
<tr><td>Pointer to Block (n - 1)</td><td>Pointer to End of Data</td></tr>
<tr><td>Channel ID</td><td>Block 0 Data 0</td></tr>
<tr><td colspan="2">...</td></tr>
<tr><td>Channel ID</td><td>Block 0 Data m</td></tr>
<tr><td colspan="2">...</td></tr>
<tr><td>Channel ID</td><td>Block (n - 1) Data 0</td></tr>
<tr><td colspan="2">...</td></tr>
<tr><td>Channel ID</td><td>Block (n - 1) Data j</td></tr>
</table>

Table 28: *Structure of the CHAD, CEMD, WHAD, PEMD, and PHAD banks. This is without clustering. The alternative clustering format is described in Section 9.1. In the table, each line shows two 16 bit words.(except for the bank header and type words which are 32 bit words).*

Data from the Hadronic section are contained in the CHAD bank and the data from the Electromagnetic section are contained in the CEMD bank. Both banks have the same structure. These banks have many things in common with all the other calorimeter banks. For example, all the calorimeter banks have the same Channel ID bitfields. The channel IDs and other common format details are specified in Section 10.1.

Both CEMD and CHAD have 16 data blocks numbered from 0 to 15. Each Block of data contains data from three adjacent Central Calorimeter wedges. Block 0 has wedges 0 to 2, block 1 has wedges 3 to 5, ... The blocks of data are organised in increasing ϕ. The first set of eight blocks contain data from the West side of the detector and the following set of eight blocks contain data from the East side. The data within a block is organized in increasing phi number, η number, and pmt number.

```
phi 0 eta 0 pmt 0
phi 0 eta 0 pmt 1
phi 0 eta 1 pmt 0
phi 0 eta 1 pmt 1
...
phi 0 eta 9 pmt 0
phi 0 eta 9 pmt 1
phi 1 eta 0 pmt 0
...
```

There are two photomultiplier tubes for each tower. Data from these are differentiated via the pmt bit in the channel ID. This bit is 0 for the tube at lower phi and 1 for the tube at higher phi.

10.3 WHAD - Endwall Hadronic Calorimeter

The Endwall Calorimeter covers the polar angles between 30^o to 45^o and 135^o to 150^o. There are a total of 48 15^o wedges each having 6 towers. Light is collected by two PMTs per tower for a total of 576 PMTs reading out the entire Endwall Calorimeter. The Endwall Calorimeter overlaps with the both the Central Calorimeter and the Plug Calorimeter, see Figure 1.

Data from the Endwall calorimeter is stored in the WHAD bank. This bank has many things in common with all the other calorimeter banks. For example, all the calorimeter banks have the same Channel ID bitfields. The channel IDs and other common format details are specified in Section 10.1, Table 27, and Table 28.

There are 8 blocks in the WHAD bank. Each Block of data contains data from six adjacent wedges. Block 0 has wedges 0 to 5, block 1 has wedges 6 to 11, ... The blocks of data are organised in increasing ϕ. The first set of four blocks contain data from the West side of the detector and the following set of four blocks contain data from the East side.

The data within a block is organized in increasing phi number, η number, and pmt number.

```
phi 0 eta 6 pmt 0
phi 0 eta 6 pmt 1
phi 0 eta 7 pmt 0
phi 0 eta 7 pmt 1
...
phi 0 eta 11 pmt 0
phi 0 eta 11 pmt 1
phi 1 eta  0 pmt 0
...
```

There are two photomultiplier tubes for each tower. Data from these are differentiated via the pmt bit in the channel ID. This bit is 0 for the tube at lower phi and 1 for the tube at higher phi.

10.4 PHAD and PEMD - Plug Hadronic and Electromagnetic Calorimeter Banks

The Plug Calorimeter consists of an electromagnetic (EM) and hadronic (HA) section. The EM section has a shower position detector which is described in Section 14. Timing information is available for the hadronic section, see Section 17.2. A side view of the Plug Calorimeter is shown in Figure 4.

Data from the plug hadronic calorimeter is stored in the PHAD bank. Data from the plug electromagnetic calorimeter is stored in the PEMD bank. These banks have many things in common with all the other calorimeter banks. For example, all the calorimeter banks have the same Channel ID bitfields. The channel IDs and other common format details are specified in Section 10.1, Table 27, and Table 28.

A total of twelve crates are used to read out the Plug Calorimeters. Each crate reads out four adjacent 15^o wedges. There is one data block for each crate in each bank. The data blocks are numbered 0 to 11. Block 0 has wedges 23, 0, 1, and 2 on the west side. Block 1 has wedges 3 to 6 on the west side. This pattern continues to block 5, which contains

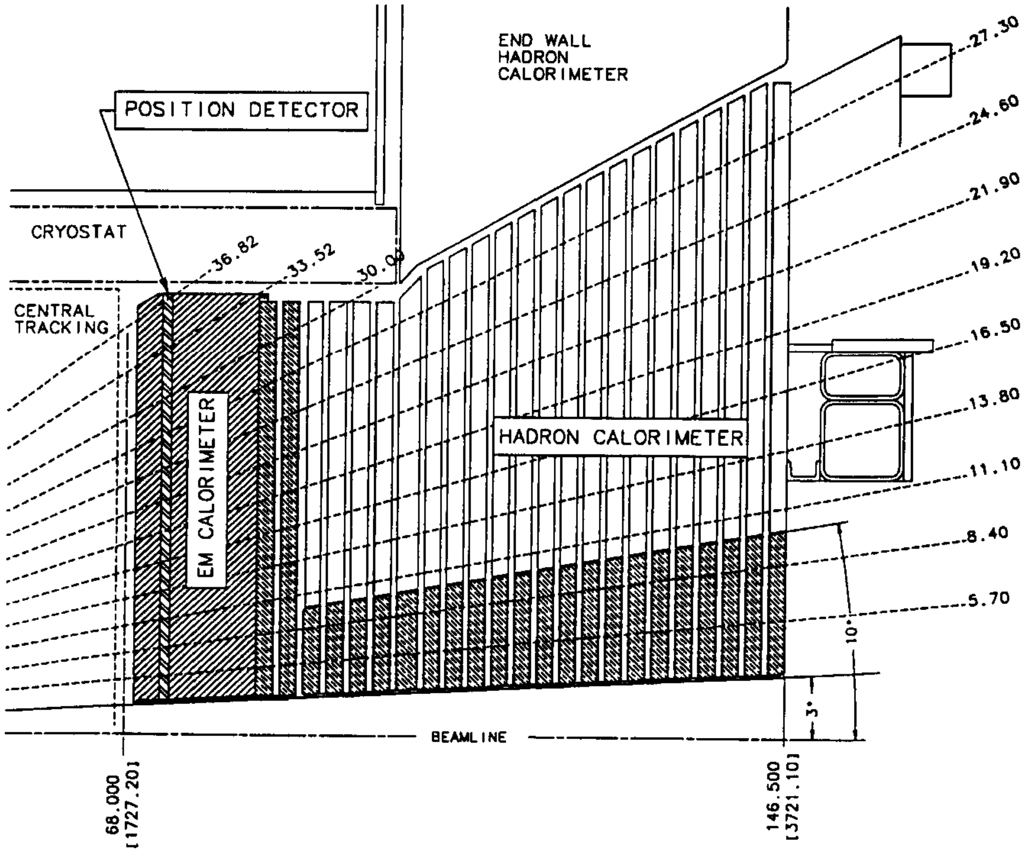

Figure 4: *Cross section of the Plug Calorimeter.*

wedges 19 to 22 on the west side. The pattern repeats for blocks 6 to 11 on the east side. Block 6 has wedges 23, 0, 1, and 2 ....

The data within a block is organized in increasing phi number, η number, and pmt number. There is one exception to this rule. For blocks 0 and 6, data from wedge 23 preceeds the data from wedge 0, 1, and 2.

```
phi 23 eta 10 tile 0
phi 23 eta 10 tile 1
phi 23 eta 11 tile 0
phi 23 eta 11 tile 1
...
phi 23 eta 20 tile 0
phi 23 eta 21 tile 0
phi 0  eta 10 tile 0
phi 0  eta 10 tile 1
...
```

For the plug, the term tower can have two meanings so one must be careful. Sometimes the rapidity segmentation field in the channel ID is called tower number. In this case, it varies from 10 to 21. For the plug, there are some values of rapidity where there are two towers in a wedge and the towers are numbered in a different fashion. This numbering scheme is shown in Figure 5. The two towers in a wedge are differentiated in the channel ID using the PMT/tile bit. It is numbered 0 for the tower at lower phi and 1 for the one at higher phi. For four towers at the highest rapidity, there is only one tower and the bit is always zero.

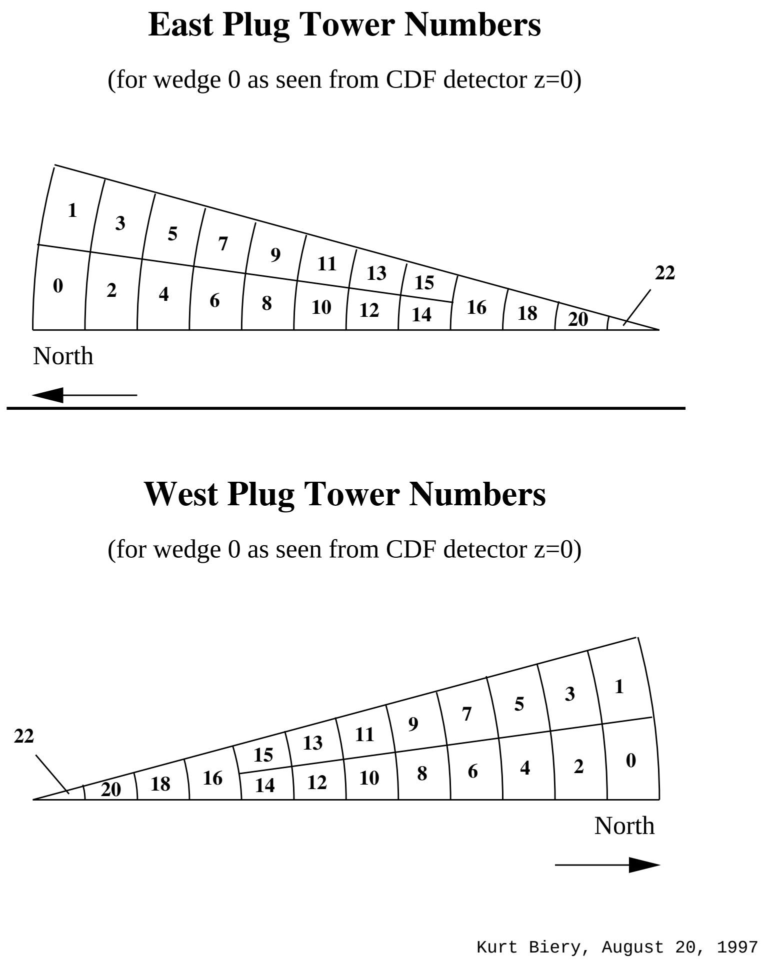

Figure 5: *The physical towers of the first* 15^o *wedges of the East and West halves of the Plug Calorimeter. The tower numbers are shown. The wedge numbers increase with increasing* ϕ *for each plug. These tower numbers are calculated from the channel ID by taking the rapidity segmentation, subtracting 10, multiplying by 2, and adding the PMT/Tile bit.*

11 Central Calorimeter Shower Maximum Detector

The Central Calorimeter Shower Maximum Detector (CES) is used to determine shower position and tranverse development at the shower maximum by measuring charge deposition on orthogonal strip and wire views. These chambers are located at a depth of 6 radiation lengths within the Central EM Calorimeter. Each chamber subtends 15o in ϕ and covers calorimeter towers 0 thru 9. There are 32 wire channels in the $r - \phi$ view. These are split at the boundary of scintillator towers 5 and 6 and high voltage is fed to each half separately to give a total of 64 wire channels per wedge. The strip channels are segmented into 1.67cm cells up to the boundary of scintillator towers 5 and 6 and segmented into 2.01 cells after that to give a total of 128 strip cells.

Each 15^o wedge will be read out by shower max digitizer boards (SMD). There are a total of eight SMD, each having 32 channels, in one 6U VME crate with one crate reading out one 15^o wedge. Two SMD cards read out 64 wire channels, 4 SMD cards read out 128 strip channels, one SMD card reads out a central preradiator chamber and one SMD card reads out a central crack chamber. Data from the eight SMD cards is sent to the shower maximum readout card (SMXR) located in the Central Calorimeter Crate. The SMXR receives a total of 256 channels. Three wedges are read out by one Central Calorimeter Crate thus requiring three SMXR cards.

11.1 Channel ID - Central Calorimeter Shower Max

The bit assignment for the Central Calorimeter Shower Max Detector is specified in Table 29.

Bit	Channel ID
15:13	Cluster width
12:8	Module (0-23)
7	Strip (0) or Wire (1)
6:0	Strip (0-127), Wire (0-63)

Table 29: *Bit assignment for the Central Calorimeter Shower Max Channel ID*

11.2 Channel Mapping - Central Calorimeter Shower Max

One block of data contains information from three calorimeter wedges. The data within a wedge is organized such that the strip information from the module with the lowest phi index comes first followed by the wire information, followed by data from the second then the third module.

Module 0, strip 0
...
Module 0, strip 127
Module 0, wire 0
...
Moduel 0, wire 63
...
Module 2, strip 0
...
Module 2, strip 127
Module 2, wire 0
...
Moduel 2, wire 63

11.3 CESD - Central Calorimeter Shower Max

The structure of the CESD bank is specified in Table 30. The data is organized in blocks with three wedges per block.

"CESD"	
Bank Number	
Bank Version	
Bank Length	
Bank Type (I*2)	
Number of Blocks (=16)	Pointer to West Block 0
...	...
Pointer to West Block 7	Pointer to East Block 0
...	...
Pointer to East Block 7	Pointer to End of Data
Channel ID	West Block 0 Data 0
...	
Channel ID	West Block 0 Data n
...	
Channel ID	East Block 7 Data 0
...	
Channel ID	East Block 7 Data n

Table 30: *Structure of the CESD bank.*

12 Central Calorimeter Preradiator

The Central Preradiator Detector (CPR) is a wire proportional chamber. These chambers are located at the front face of the Central EM Calorimeter. One chamber subtends 15o in ϕ and covers half of the calorimeter towers 0 thru 9 so that each calorimeter wedge is instrumented with 2 CPR chambers. Each chamber has 16 wire channels in the $r - \phi$ view to give a total of 32 CPR channels per wedge.

12.1 Channel ID - Central Calorimeter Preradiator

The Channel ID of the Central Calorimeter Preradiator is specified in Table 31.

Bit	Definition
15:13	Cluster width
12:8	Module number (0 - 23)
7:6	Unused
5	West = 0; East = 1
4:0	Wire (0-31)

Table 31: *Channel ID for the Central Calorimeter Preradiator.*

12.2 CPRD - Central Preradiator

The structure of the CPRD bank is specified in Table 32. The data is organized in blocks with three wedges per block.

"CPRD"	
Bank Number	
Bank Version	
Bank Length	
Bank Type (I*2)	
Number of Blocks (=16)	Pointer to West Block 0
...	...
Pointer to West Block 7	Pointer to East Block 0
...	...
Pointer to East Block 7	Pointer to End of Data
Channel ID	West Block 0 Data 0
...	
Channel ID	West Block 0 Data n
...	
Channel ID	East Block 7 Data 0
...	
Channel ID	East Block 7 Data n

Table 32: *Structure of the CPRD bank.*

13 Central Calorimeter Crack Chambers

The Central Calorimetor Crack chambers (CCR) are strip readouts of wire chambers located behind 12 X0 of tungsten. The CCR are 3.5 cm wide and area located in front of the gap between wedge modules. There are a total of 10 strips per chamber which correspond to CEM towers.

13.1 Channel ID - Central Calorimeter Crack Chambers

Bit	Channel ID
15:13	Cluster width
12:8	Module (0-23)
7:5	Unused
4	West = 0 ; East = 1
3:0	Pad number (0-9)

Table 33: *Channel ID for the Central Calorimeter Crack Chambers.*

13.2 CCRD - Central Crack Chambers

The structure of the CCRD bank is specified in Table 34. The data is organized in blocks with three wedges per block.

"CCRD"	
Bank Number	
Bank Version	
Bank Length	
Bank Type (I*2)	
Number of Blocks (=16)	Pointer to West Block 0
...	...
Pointer to West Block 7	Pointer to East Block 0
...	...
Pointer to East Block 7	Pointer to End of Data
Channel ID	West Block 0 Data 0
...	
Channel ID	West Block 0 Data n
...	
Channel ID	East Block 7 Data 0
...	
Channel ID	East Block 7 Data n

Table 34: *Structure of the CCRD bank.*

14 PESD - Plug Calorimeter Shower Maximum Detector

The Plug Shower Maximum Detector (SMD) is located 6 radiation lengths inside the EM section of the Plug Calorimeter and provides a measurement of the position of showering particles. On each side, the SMD is segmented into eight 45^o wedges (not standard CDF 15^o wedges). Each wedge has two planes containing 200 scintillating strips each. Data from this detector is stored in the PESD bank.

The PESD bank is a QIE bank and has the format described in Section 9.1. The general structure is the same as shown in Table 28 for the calorimeter banks, but there are many details that are different from the calorimeter bank format. Those details are described below.

The PESD channel ID is shown in Table 35. On each side, the SMD is segmented into eight 45^o wedges numbered from 0 to 7 in ϕ order. Wedge 0 lies between $\phi = 0^o$ and $\phi = 45^o$.

Bit	Channel ID
15:13	Cluster width
12	Side: West = 0, East = 1
11:9	Wedge (0:7)
8	Layer (0:1)
7:0	Strip (1:200)

Table 35: *Channel ID in the PESD bank.*

Each plane in a wedge is shaped like a slice of a pie (like a slice in a thin pie that lies perpendicular to the beam and has a hole in the middle around the beam). Each wedge has two edges that lie at constant ϕ, we will call them the high ϕ and low ϕ edges. The scintillator strips in a single plane of a single wedge are all oriented parallel to one of these edges. In layer 0, the strips are parallel to the high ϕ edge. In layer 1, the strips are parallel to the low ϕ edge.

On the west side, layer 0 is the layer closer to the interaction region, which is the standard CDF convention. But on the east side the convention is reversed. On the east, layer 1 is closer to the interaction region and layer 0 is farther from the interaction region. There is a second way to label the layers. Each layer is labeled U or V. On both sides, the U layer is closer to the interaction region and the V layer is farther from the interaction region.

The strips are numbered based on their distance from the boundary of the wedge parallel to the strips. This sounds simple, but there is a complication. In each layer of each wedge, there are 30 instances where two strips lie along the same line at the same distance from the wedge boundary. The pair of strips do not overlap. One strip is at high pseudorapidity and one strip at low pseudorapidity. These 30 pairs of strips are the strips nearest the boundary of the wedge parallel to the strips. The other 140 strips all are located at different distances from the boundary of the wedge parallel to the strips. Strips 1 to 30 are the low pseudorapidity strips near the boundary and they are numbered in order of distance from the boundary. Strips 31 to 170 are strips where there is only one strip at each distance, also numbered in order of distance from the boundary. Strips 171 to 200 are the high pseudorapidity strips near the boundary, also numbered in order of distance from the boundary. So strips 1 and 171, 2 and 172, ..., and strips 30 and 200 lie along a common line.

The top bit in the data word is an error bit. The lower 15 bits are interpreted as a simple integer that contains the energy in units of counts.

The correspondence between wedges and data blocks is shown in Table 36. The order of channels within the blocks is not specified. Reconstruction code that accesses the bank should be able to handle any order of channels. The order should not affect reconstruction results.

Block	Side and Wedges
0	West Wedges 1 and 2
1	West Wedges 3 and 4
2	West Wedges 5 and 6
3	West Wedges 7 and 0
4	East Wedges 1 and 2
5	East Wedges 3 and 4
6	East Wedges 5 and 6
7	East Wedges 7 and 0

Table 36: *Correspondence between blocks and wedges in the PESD bank.*

15 PPRD - Plug Calorimeter Preradiator

The PPRD bank format is very similar to the format of the calorimeter banks. The channel ID is the same and is defined in Table 27. Except for a couple of things discussed explicitly below, everything in Section 10.1 also applies to the PPRD bank format. The bank structure shown in Table 28 also applies to the PPRD bank.

There are a few things that are different from the calorimeter bank format. The top bit in the data word is an error bit. The lower 15 bits are interpreted as a simple integer that contains the energy in units of counts.

The lowest bit in the channel ID has the same meaning as for the PEMD and PHAD banks. Within a wedge, there are two towers at the same pseudorapidity for many values of pseudorapidity. The PMT/Tile bit is zero for the tower at lower ϕ and 1 for the tower at higher ϕ.

There are 8 blocks in the PPRD bank. The first four blocks contain channels from the west side. The next four blocks contain channels from the east side. Beyond that, the assignment of channels to blocks is not specified. The order of channels within a block in PPRD is not specified. Reconstruction code that accesses the bank should be able to handle any order of channels and any assignment of channels to blocks. The ordering should not affect reconstruction results.

16 SMXR Data Error and Pedestal Shift Banks

The SMXR readout is one of the few systems in the CDF front end system in which the digitizing electronics do not reside in a front end crate. As a result, it is necessary to provide banks to report the many possible error conditions which might occur in the data path from the front-end electronics to the Level 2 buffer. Error conditions are signaled in two basic locations: the header word, which is read from each SMXR channel independently, and error bits included in the data stream sent by the SMQIE (analogous to the case for the ADMEMs). We have chosen to define these banks as hardware-oriented in terms of their channel addressing to minimise the number of such banks which need to be defined to accomodate the many component banks read out by SMXR's.

Errors reported in the header read out for each SMXR channel are reported in the CSHD bank for central crates and the PSHD bank for plug crates (these are referred to below as the "header error" bank.) Errors reported in the data stream are recorded in the CSED and PSED banks for central and plug crates respectively (these are referred to below as the "data error" bank).

The rule for an entry into the header error bank is quite straightforward. The header for an SMXR channel will be added to the header error bank if one of the bits in the header signifying a data transmission error are set (the endcycle, dtr, or SMD missing data bits as described in CDF5186.)

The data error bank is designed to be the (hardware) complement of the raw data bank. That is to say that data recorded in the data error bank WILL NOT be recorded in the raw data bank. The rules for data entry into the data error bank are quite involved given the several error conditions that would make such an entry necessary.

1 If an SMD missing data bit is set in the header, then all data for that SMXR channel (i.e. a single SMD board) will be recorded in the data error bank.

2 The endcycle and dtr error bits both signal fatal errors requiring CDFreset. In this case all data read from the SMXR Level 2 buffer is invalid. If this data is recorded in the data stream, it will be in the data error bank (more likely, it will be simply dropped entirely).

3 If one of the data error bits is set in the Level 2 buffer on a single detector channel then the data for that single detector channel will be recorded in the data error bank.

4 If one of the data error bits is set in the Level 2 buffer for multiple detector channels, but from the same SQUID, then the data for these detector channels will be recorded in the data error bank. All other data is considered good and will appear in the raw data bank.

5 If one of the data error bits is set in the Level 2 buffer for multiple detector channels and from different SQUIDs, then the data for the entire SMXR readout channel is recorded in the data error bank.

6 If the wordcount error bit is set in the Level 2 buffer, then the data for the entire SMXR readout channel is recorded in the data error bank.

Finally, coherent shifts have been observed in the 4 consecutive timeslices, which are summed to form the linearised data for the central detectors (CES, CCR, CPR). An algorithm will be run in the front end crate to measure and correct this shift. The CSSD bank contains the shifted pedestals used for this correction on an event by event basis. At the present time, we envisage monitoring these shifts in YMON and downstream, therefore this bank will appear in the data stream for all events.

16.0.1 General Structure

This applies to the five defined hardware banks: CSHD, CSED, CSSD, PSHD and PSED. The bit assignment for the Channel ID is specified to be similar to the Channel ID specified for calibrations to facilitate re-use of accessor code. The block structure is identical to that defined for the raw data banks and is shown in Table 37 for the central detector banks (CSHD, CSED, CSSD) and in Table 38 for plug detector banks (PSHD and PSED).

In the central, there are a total of 16 crates, numbered 1 to 16. In each bank, there is one data block per crate. Block numbers run from 0 to 15. Block number equals crate number minus one. Each block handles 3 wedges. Block 0 handles wedges 0 to 2 on the west side. Block 1 handles wedges 3 to 5. This pattern continues in phi order on the west side then repeats on the east side.

In the plug, there are 8 crates, 4 east and west. Crate numbers run from 1 to 8 with block numbers running from 0 to 7.

“CSHD” or “CSED” or “CSSD”
Bank Number
Bank Version
Bank Length
Bank Type = 3 (I*4 monotype)
Number of Blocks (=16)
Pointer to Block 0 (West)
...
Pointer to Block 7 (West)
Pointer to Block 8 (East)
...
Pointer to Block 15 (East)
Pointer to End of Data
Start of Data
...
...

Table 37: *Block Structure of the CSHD, CSED and CSSD banks.*

Each crate contains 3 SMXR cards. Each SMXR card contains 8 separate channels, 6 of which are the data for 1 wedge of Central Shower Max (CES), 1 channel for the data for 1 wedge of Central Preshower (CPR), and 1 channel for the data for 1 wedge of Central Crack Chambers (CCR). Each SMXR channel in the central (plug) detector reads out 32 (48) QIE channels.

"PSHD" or "PSED"
Bank Number
Bank Version
Bank Length
Bank Type = 3 (I*4 monotype)
Number of Blocks (=8)
Pointer to Block 0 (West)
Pointer to Block 1 (West)
...
Pointer to Block 3 (West)
Pointer to Block 4 (East)
...
Pointer to Block 7 (East)
Pointer to End of Data
Start of Data
...

Table 38: *Block Structure of the PSHD, PSED banks.*

These banks may be written out with some channels not present. The order that the channels are written into the bank is not specified. The map from (slot, SMXR channel, QIE number) to the hardware channels is given elsewhere.

16.0.2 CSHD, PSHD - SMXR Header Banks

The block structure for this bank is shown above in Table 37 for CSHD and in Table 38 for PSHD.

Channel ID (I*4)
SMXR Channel Header (I*4)
Channel ID (I*4)
SMXR Channel Header (I*4)
...
...

Table 39: *Format of a data block in a CSHD or PSHD bank.*

Bit	Channel ID
31:24	Crate Number (1:16)
23:19	Slot Number

Table 40: *Channel ID Bit assignment for the SMXR header bank (CSHD, PSHD). Note that the SMXR channel number can be determined from geographical address in the header word.*

The bank data format is given in Table 39. For each data element, there is one 32-bit word containing the SMXR Channel ID as specified in Table 40 and one data word. The data word is a 32-bit integer containing the header read from the corresponding Level 2 buffer. The header word is bit packed. Its format is given in CDF5186 and is repeated here in Table 41 for the ease of the user. The reader is directed to CDF5186 to determine the meaning of these bits with regard to the operation of the hardware. In general, the data for any SMXR readout channel is only added to this bank if one of the bits indicating a data transmission error is set.

Bit	Header Word
0:7	BunchID (aka bunch crossing number)
8:12	Geographical Address
13:20	SMD Missing data bit map (SMXR chan 0-7)
21	DTR function bit
22	End Cycle
23:31	Board Type

Table 41: *Header Word Bit assignment (see CDF5186)*

16.0.3 CSSD - Central SMXR Shifted Pedestal Bank

The block structure for this bank is shown above in Table 37. The bank data format is given in Table 42. For each data element, there is one 32-bit word containing the SMXR Channel ID as specified in Table 43 and one 32-bit data word. The data word contains the shifted pedestals (to avoid having to deal with negative numbers). Bits 0-15 contain the shifted pedestal used for detector channels 0-15. Bits 16-31 contains the shifted pedestal for detector channels 16-31. There is no corresponding bank for the plug readout since we do not expect this coherent shift.

Channel ID (I*4)
Shifted Pedestal for Det Channels 0-15 (bits 0:15 of the data word)
Shifted Pedestal for Det Channels 16-31 (bits 16:31 of the data word)
Channel ID (I*4)
Shifted Pedestal for Det Channels 0-15 (bits 0:15 of the data word)
Shifted Pedestal for Det Channels 16-31 (bits 16:31 of the data word)
...
...

Table 42: *Format of a data block in a CSSD bank.*

Bit	Channel ID
31:24	Crate Number (1:16)
23:19	Slot Number
18:16	SMXR Channel

Table 43: *Channel ID Bit assignment for the SMXR shifted pedestal bank (CSSD).*

16.0.4 CSED, PSED - SMXR Data Error Banks

The block structure for this bank is shown above in Table 37 for CSED and in 38 for PSED.

Channel ID (I*4)
Level 2 Det. Channel Data (I*4)
Channel ID (I*4)
Level 2 Det. Channel Data (I*4)
...
...

Table 44: *Format of a data block in a CSED or PSED bank.*

The bank data format is given in Table 44. For each data element, there is one 32-bit word containing the SMXR Channel ID as specified in Table 45. Note that this only differs from the Channel ID in the header bank in that QIE number is also part of the field. The data word is a 32-bit integer. The lower 16 bits of this word contain the 16 bit data read from the Level 2 buffer for this QIE number. The upper 16 bits are not defined at present but are required by the DAQ system. 3 error bits are returned in the level 2 buffer for plug readout and 1 for central readout as described in CDF5186. For ease of the user, the bit assignment is repeated here. **Note that this describes the bit format for only data mode operation of the SMXR.** Table 46 gives the bit assigment for the CSED bank and Table 47 the bit assignment for the PSED bank (the difference is due to features of the SMQIE, summing of timeslices and the limitation of the data output to 16 bits per detector channel from the Level 2 buffer.))

Bit	Channel ID
31:24	Crate Number (1:16)
23:19	Slot Number
18:16	SMXR Channel
15:8	QIE Number

Table 45: *Channel ID Bit assignment for the SMXR data error bank (CSED, PSED).*

Bit	Channel ID
16:31	Not Defined
15	Error Bit (compound)
0:14	Linearized Data (from LUT)

Table 46: *CSED Data Bit assignment.*

Bit	Channel ID
16:31	Not Defined
15	Wordcount Error
14	Capacitor ID sequence Error
13	Parity Error
0:12	Linearized Data (from LUT)

Table 47: *PSED Data Bit assignment.*

17 Calorimeter TDC Banks

17.1 EMTD - Electromagnetic Calorimeter TDC Bank

The EMTD bank is a standard TDC bank. Most of the format details for TDC banks are specified in Section 9.2.

The EMTD bank holds TDC data from the electromagnetic Central and Plug calorimeters. The TDCs are the same as used for the COT and muon detectors. There are 6 TDCs for the CEM calorimeter, each TDC covers 8 wedges and spans towers from 0 to 9 inclusively. There are 4 TDCs for the PEM calorimeter, each TDC covers 12 wedges and spans towers from 10 to 17 inclusively. The four PEM towers with highest rapidity are not equipped with TDCs. Each TDC module, both in CEM and PEM, services either the east side or the west side exclusively.

The module ID for the EMTD bank is defined in Table 48. The channel ID for the EMTD bank is defined in Table 49. The following formulas are used to combine the information from the channel ID and module ID.

$$\text{Rapidity} = (\text{Rapidity Low or High Bit}) * 10 + (\text{Rapidity Segmentation}).$$

$$\text{Wedge} = (\text{Lowest Wedge}) + (\text{Wedge in TDC}).$$

Then if and only if the resulting wedge number is greater than 23

$$\text{Wedge} = \text{Wedge} - 24.$$

Note that the values that the "Lowest Wedge" bitfield will hold are not specified. Only the way to use them to calculate wedge is specified. The "Lowest Wedge" number is guaranteed to be no greater than 23, while "Lowest Wedge + Wedge in TDC" can be greater than 23.

There are a number of unused TDC channels in the CEM system. All those channels are assigned distinct channel IDs for debugging purposes even though they will be masked off and data from the unused channels in normal operation mode should not propagate into the bank. We are assigning the "Wedge within TDC" bitfield values outside the specified range for CEM (0:7) in order to create unique channel IDs for the unused channels.

While the system is being tested and prototyped, the TDC modules will be configured and cabled in a different way than is described above. The data format described above will not work in this special configuration, because there will be PEM and CEM channels cabled to the same TDC. The CEM channels will be exclusively tower 9 CEM channels. The standard format is extended as follows to support this configuration. The value 3 will be written into the "Rapidity Low or High" bitfield in the module ID. If the "Rapidity segmentation modulo 10" bitfield in the channel ID is not 9, then this is a PEM channel and 10 is added to this value to get the tower number (same as usual). If the "Rapidity segmentation modulo 10" bitfield in the channel ID is 9, then this is a CEM channel and it is tower 9 (not 19). In both cases, the wedge is calculated the same as always. The only difference is that the "Wedge within TDC" bitfield may hold values between 0 and 11 for the CEM channels instead of the normal range of 0 to 7. Side is extracted from the module ID as always. The offline bank class supports both this test format and the normal format in a way that should be transparent to most users.

Bits	Description
8:6	Rapidity Low or High (0 = towers 0:9 (CEM), 1 = towers 10:17 (PEM))
5:5	Side (0=West, 1=East)
4:0	Lowest Wedge in TDC

Table 48: *EMTD module ID.*

Bits	Description
8:4	Wedge within TDC (0:7 for low rapidity, 0:11 for high rapidity)
3:0	Rapidity segmentation modulo 10 (0:9 for low rapidity, 0:7 for high rapidity)

Table 49: *EMTD channel ID.*

17.2 HATD - Hadronic Calorimeter TDC Bank

The HATD bank is a standard TDC bank. Most of the format details for TDC banks are specified in Section 9.2.

The HATD bank holds TDC data from the hadron Central, Wall and Plug calorimeters. The TDCs are the same as used for the COT and muon detectors. There will be 16 TDCs. 12 TDCs handle channels for rapidity rings 0-15 corresponding to $|\eta| < 1.78$. These TDCs handle 2 wedges per TDC. The other four TDCs cover the remaining six η rings per side (16-21). These TDCs handle 6 wedges per TDC.

The module ID for the HATD bank is defined in Table 50. The module IDs will have the values 0,2,4,6,...,20,22 for the low rapidity TDCs and 32, 38, 44, and 50 for the high rapidity TDCs.

The channel ID for the HATD bank is defined in Table 51. Bit 0 in the channel ID is defined in more detail in Table 52. The following two formulas are used to combine the information from the channel ID and module ID:

$$\text{Wedge} = (\text{Lowest Wedge}) + (\text{Wedge in TDC}).$$

$$\text{Rapidity} = (\text{Rapidity Low or High Bit}) * 16 + (\text{Rapidity Segmentation}).$$

Bits	Description
6:8	Unused
5:5	Rapidity Low or High (0 = towers 0:15, 1 = towers 16:21)
4:0	Lowest Wedge in TDC

Table 50: *HATD module ID.*

Bits	Description
8:8	Side (0=West, 1=East)
7:5	Wedge within TDC (0:1 for low rapidity, 0:5 for high rapidity)
4:1	Rapidity segmentation modulo 16 (0:15 for low rapidity, 0:5 for high rapidity)
0	Depth Central/Wall or Plug Tile (See below)

Table 51: *HATD channel ID.*

Rapidity Region	Rapidity Segments	Bit Definition
Low	00 - 05	bit=0 (always central)
Low	06 - 07	bit=0 for central, bit = 1 for endwall
Low	08 - 11	bit=1 (always endwall)
Low	12 - 15	bit=0 for low ϕ tile, bit = 1 for high ϕ tile
High	00 - 01	bit=0 for low ϕ tile, bit = 1 for high ϕ tile
High	02 - 05	bit=0 (one tile per rapidity segmentation)

Table 52: *Bit 0 in the HATD channel ID.*

18 LEDD and XEFD - CEM Light Emitting Diode/Xenon Flashing Banks

The format of these banks in shown in Table 53. These banks have a header and block pointers as described in Sections 4 and 4.1. Inside the blocks, three 16-bit words are written for each channel. The first word is a channel ID. The bit assignment for the channel ID is shown in Table 54. The second word is the number of counts for the PIN diode or the TAC. The third word is defined as the ratio of the PIN diode counts.

- For PIN0 channel, the ratio is $\frac{\text{PIN0 counts}}{\text{PIN1 counts}}$
- For LED PIN1 channel, the ratio is $\frac{\text{PIN1 counts}}{\text{PIN0 counts}}$
- For XEF PIN1 channel, the ratio is $\frac{\text{PIN1 counts}}{\text{PIN2 counts}}$
- For XEF PIN2 channel, the ratio is $\frac{\text{PIN2 counts}}{\text{PIN0 counts}}$
- LED PIN2 does not exist
- For the TAC, the third word is always set to zero.

Both data words are 16 bit integers. Note that if there are an odd number of 16 bit words in a block, the reformatter will insert a padding word at the end of the block. All blocks must start on 32 bit boundaries.

"LEDD/XEFD"	
Bank Number	
Bank Version	
Bank Length	
Bank Type (I*2)	
Number of Blocks (=16)	Pointer to Block 0
Pointer to Block 1	Pointer to Block 2
...	...
Pointer to East Block 15	Pointer to End Of Data
Channel ID	number of counts
PIN ratio	Channel ID
number of counts	Pin ratio
...	...
...	...
Channel ID	number of counts
PIN ratio	Channel ID
number of counts	Pin ratio

Table 53: *Structure of the LEDD/XEFD banks. Each block contains data from three wedges of the Central Electromagnetic Calorimeter. The first eight blocks include data from the West side of the calorimeter in ϕ order. The next 8 blocks have data from the East side also in ϕ order.*

Bit	Definition	
15:13	all zero (reserved)	
12	zero (unused)	
11	West = 0; East = 1	
10:6	Azimuthal segmentation (0:23)	
5:1	Calcard channel ID(0:3)	0: PIN diode 0 1: PIN diode 1 2 : PIN diode 2 3 : TAC
0	0	

Table 54: *Bit assignment for the channel ID of the LEDD and XEFD banks. Azimuthal segmentation is also known as wedge number. Bits 13:15 must be zero since the bank accessor code will interpret nonzero values as a cluster width.*

19 Laser Reference Banks

19.1 Central and Endwall Calorimeters

Data from the laser reference crate for the central hadronic calorimeter and the endwall calorimeter is stored in two banks, the LSTD and LSAD banks. Both banks share a common channel ID which is defined in Table 55.

There are 26 channels in each bank. There are two types of channel: pin diode and photomultiplier (PM). There are two pin diode channels. The group number is 0 for the west pin diode channel and 1 for the east pin diode channel. There are 24 PM channels. The PM groups are defined in Table 56. There are four PM's in a group. These are numbered 0 to 3. These four PM's measure the same quantity and give redundancy and more statistical significance to the measurement.

Bits	Definition
0	Type: Pin diode = 0, PM = 1
1:3	Group (0:5)
4:5	PM number (0:3)

Table 55: *Bit assignment for the laser reference bank channel IDs. This is used in both the LSTD and LSAD bank. The "PM number" field is not used for pin diode channels. Bits 6 and higher are not used.*

Group Number	Refers to
Pin Diodes	
0	West
1	East
PMs	
0	Endwall West
1	Endwall East
2	CHA NW
3	CHA SW
4	CHA NE
5	CHA SE

Table 56: *Meaning of the group number for the laser reference banks.*

19.1.1 LSTD

The LSTD bank is a standard TDC bank. Most of the format details for TDC banks are specified in Section 9.2.

There is only one TDC module used, therefore the module ID is not necessary and left undefined. The channel ID was defined in the previous section.

19.1.2 LSAD

The LSAD bank is a QIE banks and has the format described in Section 9.1. The only difference is that this bank cannot be clustered.

The QIE data word consists of a 16 bit calibrated, linearized number. The upper bit is used as a range bit. When set, the lower 15 bits should be multiplied by 8 to get the correct value.

The order the channels are written into the LSAD bank is not specified.

The channel ID for LSAD is defined above.

19.2 LSPD - Plug Calorimeter Pin Diode Data

Laser pulses are sent to the photomultiplier tubes (PMTs) on the plug calorimeter and plug shower max detector for test purposes. There is one laser for each side. Each laser sends pulses to a primary distributor (PD). The PD splits the light and sends a light signal to each of the 12 cow's on the same side. Each cow in turn splits its input into many different light signals that are sent to the photomultiplier tubes (PMTs). There is one cow mounted between each pair of adjacent PMT boxes on the plug. Half of the cows distribute light to hadronic PMTs and half distribute light to electromagnetic PMTs. As one goes around the plug in the phi direction, the cows alternate between hadronic and electromagnetic. Each cow sends signals to the 4 nearest wedges. Each wedge receives laser light from two nearest cows. For example, the cow betweeen wedges 0 and 1 on the west side sends light to the electromagnetic PMTs in wedges 23, 0, 1, and 2. The cow between wedges 2 and 3 on the west side sends light to the hadronic PMTs in wedges 1, 2, 3, and 4. This pattern continues around the detector. The east side follows the same pattern, except the positions of the electromagnetic and hadronic cows are swapped.

Pin diodes are used to monitor the intensity of light input to the primary distributor and each of the cows. There is one pin diode for each of the two primary distributors. These diodes both have “pin number” 0. Inside each cow, there are 3 pin diodes. These are redundant sensors. All three measure the same thing. They are given “pin numbers” 1, 2, and 3 for identification purposes.

There are extra ADMEMs in the electronics associated with the plug calorimeter. Logically, these extra ADMEMs would be used for the eta = 10 towers of the plug hadronic calorimeters, but these towers do not exist. The signals from the laser pin diodes are digitized by these extra ADMEMs.

The data from the pin diodes is stored in the LSPD bank. The LSPD bank is a QIE bank and has the format described in Section 9.1. The bit fields in the channel ID for the LSPD bank are specified in Table 57.

The bitfields are defined so that it is possible to reuse the online code that creates the PHAD bank to make the LSPD bank with very few modifications. For a given LSPD channel, the side, wedge, and type bit fields are filled as if they were the side, wedge, and PMT/Tile bit fields for the PHAD channel that would have used the same ADMEM channel (if tower 10 existed). Bit 0 (Type) is specified in exactly the same way as is done for bit 0 (PMT) of the PHAD channel ID (0 for the lower phi channel and 1 for higher phi within the wedge) and bits 2:6 (Wedge) correspond to the azimuthal segmentation. The value in the PHAD eta bit field is not needed.

The LSPD bank is structured like the PHAD bank described in Section 10.4 (one block per crate where each crate reads out four adjacent wedges, for a total of 12 blocks). As for the PHAD bank, the data is ordered in increasing phi (wedge) number,

For the west plug, the 3 pin diodes of each cow end up in the two wedges adjacent to the cow at higher phi (the wedge closest to the cow is always odd). If the wedge number is odd, type = 0 corresponds to pin3 and type = 1 corresponds to to an empty channel. If the wedge number is even, type = 0 corresponds to pin 1 and type = 1 corresponds to pin 2 For example, the channel with wedge = 2 and type = 0 corresponds to pin 1 of the (e.m.) cow between wedges 0 and 1, and the channel with wedge = 3 and type = 0 corresponds to pin 3 of the (had.) cow between wedges 2 and 3.

The situation is different for the east plug. The 3 pin diodes of each cow end up in the two wedges adjacent to the cow at lower phi (the wedge closest to the cow is always even). If the wedge is even, type = 1 corresponds to an empty channel and type = 0 corresponds to pin 3. If the wedge is odd, type = 1 corresponds to pin 2 and type = 0 corresponds to pin 1. For example, the channel with wedge = 3 and type = 0, corresponds to pin 1 of the (had.) cow between wedges 4 and 5.

There are also two pin diodes which monitor the light from the primary distributors. These are assigned pin number 0. The one on the west side, pin diode 0 is read out of the channel with wedge number 1 with type 1 in the channel ID. On the east side, pin diode 0 is readout of the channel with wedge number 0 and type 1.

Given any specific calorimeter channel (of the PHAD or PEMD) banks, one then needs to establish which cow is responsible for filling that channel. This may seem a little complicated because there are 4 wedges associated with each cow and two cows associated with each wedge. In fact, one only has to remember that each cow fills four adjacent wedges (the two adjacent to it at lower phi and the two adjacent to it at higher phi) and that had. cows and e.m. cows alternate. On the west side, the cow between wedges 0 and 1 is an e.m. cow while on the East side, the cow between wedges 0 and 1 is a had. cow.

Bits	Definition
0	Type: 0 or 1
1	Side: West = 0, East = 1
2:6	Wedge (0:23)
7:12	Not used
13:15	Must be 0 or Cluster Width

Table 57: *Bit fields in the LSPD bank channel IDs. See the text for a description of how to determine the pin number and cow from the channel ID. (The name "Type" is somewhat meaningless. There are not two diode types, this bit field just holds a number needed to calculate the PIN number.)*

The QIE data word consists of a 16 bit word that is interpreted as a simple unsigned integer.

20 Silicon Vertex Detector (SVX II) Component Banks

The Silicon Vertex Detector (SVXII) consists of 3 barrels which are readout on each end. Each barrel end has 12 phi wedges. Each wedge has 5 HDIs (one per layer) and is readout by a Port Card (PC). There are a total of 72 PCs (3 barrels $\times$ 2 ends $\times$ 12 wedges). The PCs are located radially outside the wedge. The data for each HDI on a PC is transmitted as a digital optical signal from the PC to the Fiber Interface Board (FIB). Two PCs send data to one FIB. The FIBs are located in VME crates outside the central detector. The FIBs perform pedestal subtraction and transfer the data through glinks to the event buffers of the VRBs in the first floor counting room where it is stored pending a Level–2 decision. Each VRB reads data from a single FIB. The VRBs will be located in 6 VME crates with 6 VRBs per crate. Data is transferred in parallel between each crate and the Event Builder. A crate's data then consists of 6 VRB blocks. Like all other subdetectors each VRB block consists of a header and 10 data channels which in this case each contain the data of one HDI. These HDI and VRB data blocks are described in section 9.3. The raw data from the SVXII is stored in the SIXD bank which is described below.

20.1 SIXD - Silicon Vertex Detector

The SIXD bank is a concatenation of 360 HDI data blocks into a single bank. The HDI blocks are ordered in groups corresponding to the 6 barrels, moving west to east. Within each barrel group, the ordering is by increasing phi wedges, and within each wedge, the ordering is by increasing layer. The structure of the SIXD bank is given in Table 58.

The SIXD data (as well as the ISLD and L00D data described later) are stored in a I*2 bank since this is more natural. Due to the large size of the bank this nevertheless constitutes a problem when creating the pointer section to jump to the HDI data units, as an absolute pointer to the last HDI Block will not generally fit in one I*2. To rectify this problem each pointer is broken into two I*2 words, with the least significant 16 bits in the first word and the most significant 16 bits in the second. Thus, the absolute pointer to HDI block i is $ptr[2*i] + (ptr[2*i+1] << 16)$.

Following the pointer section is one I*2 containing global event information stripped from the VRB header, namely the "Time Since Level 1" and "TS Turn Counter" bytes. This information may prove useful in confirming event synchronization and offline pulse height corrections.

"SIXD"	
Bank Number (=1)	
Bank Version	
Bank Length	
Bank Type (I*2)	
Number of Blocks (=360)	
Padding	
Pointer to Block 0 Bits 15:0 (West Barrel)	
Pointer to Block 0 Bits 31:16(West Barrel)	
...	
Pointer to Block 120 Bits 15:0 (Central Barrel)	
Pointer to Block 120 Bits 31:16(Central Barrel)	
...	
Pointer to Block 240 Bits 15:0 (East Barrel)	
Pointer to Block 240 Bits 31:16(East Barrel)	
...	
Pointer to End of Data Bits 15:0	
Pointer to End of Data Bits 31:16	
Time Since L1 Acc	TS Turn Count
Padding	
HDI Block 0 ...	
HDI Block 1 ...	
... ...	
HDI Block 359 ...	

Table 58: *Structure of the SIXD bank. The first set of 120 pointers (240 "half pointer") point to the West Barrel of the SVXII, the next to the Central Barrel and the final 120 point to the East Barrel. The details of a "HDI Block" are described in Table 21. In some simulation files, the pad word before the first HDI block has been moved to after the last HDI block. Offline software should be able to handle both possibilities without error.*

20.2 SVXII Specific ID Encoding

As specified in section 9.3, the conversion between the hardware quantities "portcard" and "HDI" number and the offline quantities "phi-address" and "layer" number differs between SVXII, ISL, and L00. In the case of SVXII, the conversion is straight forward. The phi address (0-11) is the upper nibble of FIB ID1. The layer is the lower nibble of FIB ID1.

Further information regarding the numbering conventions for the offline addresses is described in CDF note 4103 including how these addresses can be packed into a single address word.

Note that the term "phi address" is used for what has historically been called a "wedge number." This is to maintain consistency with ISL which does not adhere to a wedge geometry. Nonetheless, a wedge of SVXII ladders are all at the same phi address. Note that the Phi_address is the phi of the middle of the wedge.

20.3 SIXS - SVXII Status Bank

The VRB header containing error stati and global event information is discarded by the reformatter. The global event information is retained in the SIXD bank, but potential error information is not. Thus, a status bank for storing this information is forseen: the "SIXS" bank. Only non-zero raw VRB Block Error words will be added here, with the accompanying channel identifier (Section 9.3.2). Since errors are expected to occur with a low frequency this means considerable empty space will be contracted per event. Separating the error codes from the data bank has the advantage that in the (hopeful) case of no errors, the bank will not even be created, and should errors occur the bank with the error codes can be easily dropped at a later stage of the production when it is not needed.

<table>
<tr><td colspan="3">"SIXS"</td></tr>
<tr><td colspan="3">Bank Number (=1)</td></tr>
<tr><td colspan="3">Bank Version</td></tr>
<tr><td colspan="3">Bank Length</td></tr>
<tr><td colspan="3">Bank Type (I*4)</td></tr>
<tr><td colspan="3">Number of Blocks (1)</td></tr>
<tr><td colspan="3">Pointer to First Block (3)</td></tr>
<tr><td colspan="3">Pointer to End of Bank</td></tr>
<tr><td>FIB ID0</td><td>FIB ID1</td><td>Error Byte</td></tr>
<tr><td>FIB′ ID0</td><td>FIB′ ID1</td><td>Error Byte</td></tr>
<tr><td colspan="3">...</td></tr>
</table>

Table 59: *Structure of the SIXS bank.*

21 Layer00 (L00) Component Banks

Layer 00 (L00) adds a low mass, high precision measurement layer inside of SVXII to improve impact parameter resolution and radiation hardness. It is made of single-sided silicon, r-phi only. An r-phi view of the L00 detector layout is shown in fig. 6. An r-z view of a single 60° phi region is shown in fig. 7. As in SVXII and ISL, the fundamental data structure of L00 is the HDI block described in 9.3. However, the tight space constraints and the desire to use L00 data in SVT lead to a complicated readout structure that differs from the SVXII format. The L00 readout is segmented into 6 "readout barrels" as in SVXII, but they are not physical barrels. (Indeed, the only reason for this segmentation is to maintain compatibility with SVXII's structure for SVT.) L00 reads out 60 degrees of azimuth for these 6 z segments in a single portcard. (By comparison, an SVXII portcard reads 30 degrees and one z segment.) Only 4 of the 5 HDIs in a portcard are used by L00. The data path from HDIs to VRBs is the same as in SVXII (Section 20). Six L00 VRBs reside in one VRB crate.

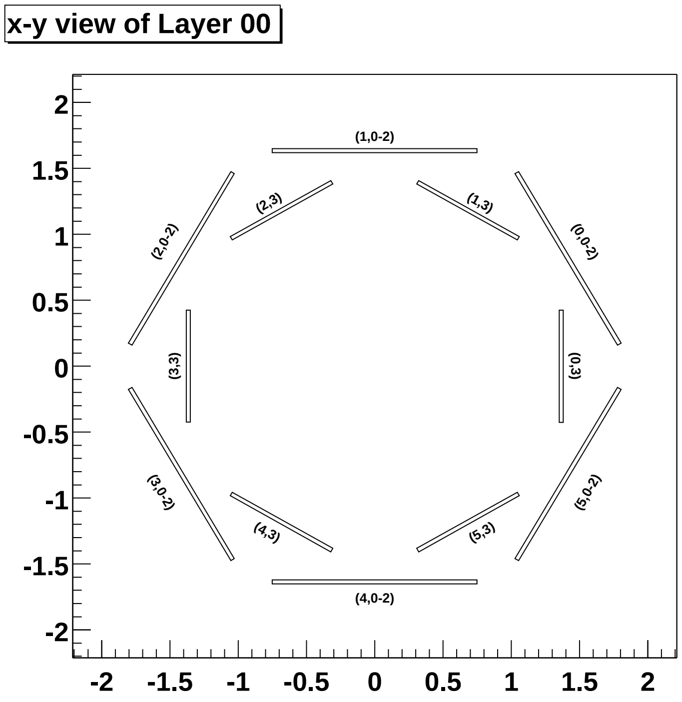

Figure 6: *Schematic r-phi view of the detector locations in Layer 00. The numeric labels, (P,H), represent the portcard number and HDI number(s) that correspond to each detector region.*

21.1 Layer00 Raw Data

The L00 raw data resides in the ISLD bank which is described in section 22. The L00 HDI blocks are simply appended after the ISL HDI blocks.

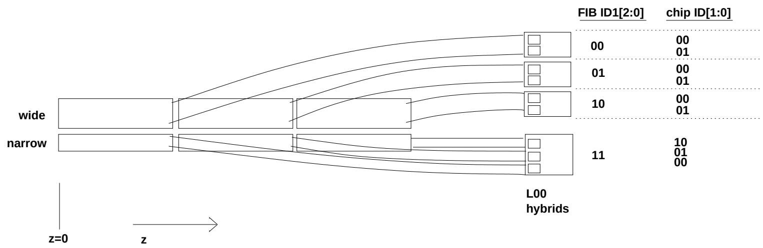

Figure 7: *Schematic r-z view of the detector locations in Layer 00. This is a view of the East end of the detector. Note that the FIB ID1 values are set to increase with increasing z. The West end of the detector also has FIB ID1 values increase with increasing z, not with —z—.*

21.2 L00 Specific ID Encoding

As specified in section 9.3, the conversion between the hardware quantities "portcard" and "HDI" number and the offline quantities "phi-address" and "layer" number differs between SVXII and L00. This is specified in Fig. 3. Note that the chip id is also needed to completely specify the z segment.

22 Intermediate Silicon Layers (ISL) Component Banks

ISL adds a sixth layer of silicon at intermediate radius ($r = 20$ cm) covering $|\eta| \lesssim 2$ and a seventh layer of silicon at ($r = 28$ cm) covering $1 \lesssim |\eta| \lesssim 2$ as shown in fig 8.

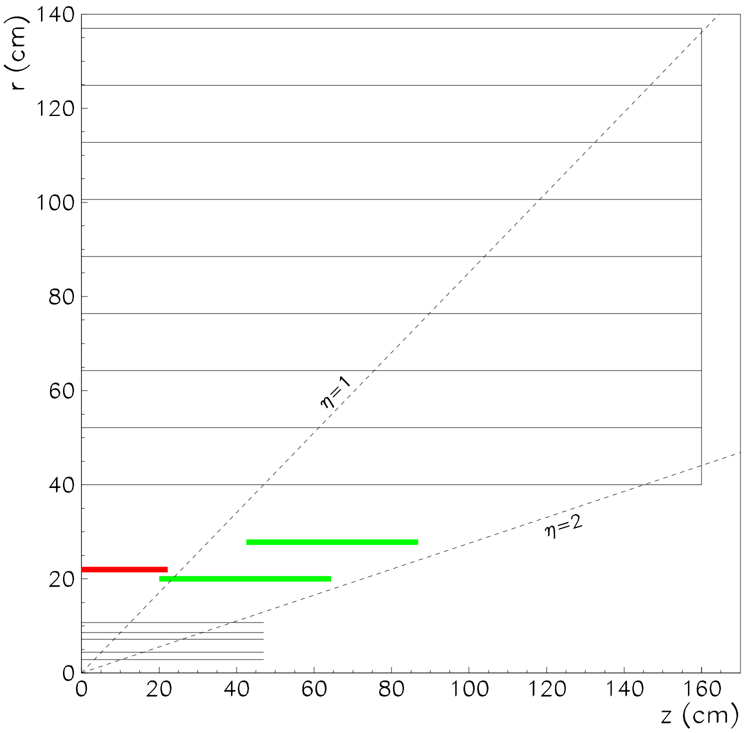

Figure 8: *A schematic view of the location of the two ISL layers between the SVXII and COT.*

The interface between the detector and DAQ is slightly different between SVXII and ISL. In SVXII, each HDI is connected to a single ladder which is one layer within a wedge. In ISL, each HDI is connected to two ladders which are adjacent in ϕ. This added complication allows substantial DAQ cost savings.

The layout differs between the central barrel of ISL and the forward barrels. The term "wedge" is not appropriate for ISL as it is in SVXII. Instead, "Phi address" is used to refer to the ladders arranged in increasing phi within a layer. The ladders are arranged as shown in fig. 9. In the forward region ($|\eta| > 1$) the sixth layer contains 24 ladders, and the seventh layer contains 36 ladders. In the central region ($|\eta| < 1$), the sixth layer contains 28 ladders and there is no seventh layer.

The data path from HDIs to VRBs is the same as in SVXII (section 20). Sixteen ISL VRBs reside in two VRB crates with 8 VRBs per crate. There are actually only 30 port cards so only 7.5 VRBs are needed for each end of the detector. However, to maintain symmetry 16 VRBs are used. Three of the 5 extra channels in the extra FIB/VRB on the west side will be used to read out the data from the *Silicon Outside the COT* (SOC) ladders described in CDF note 5333.

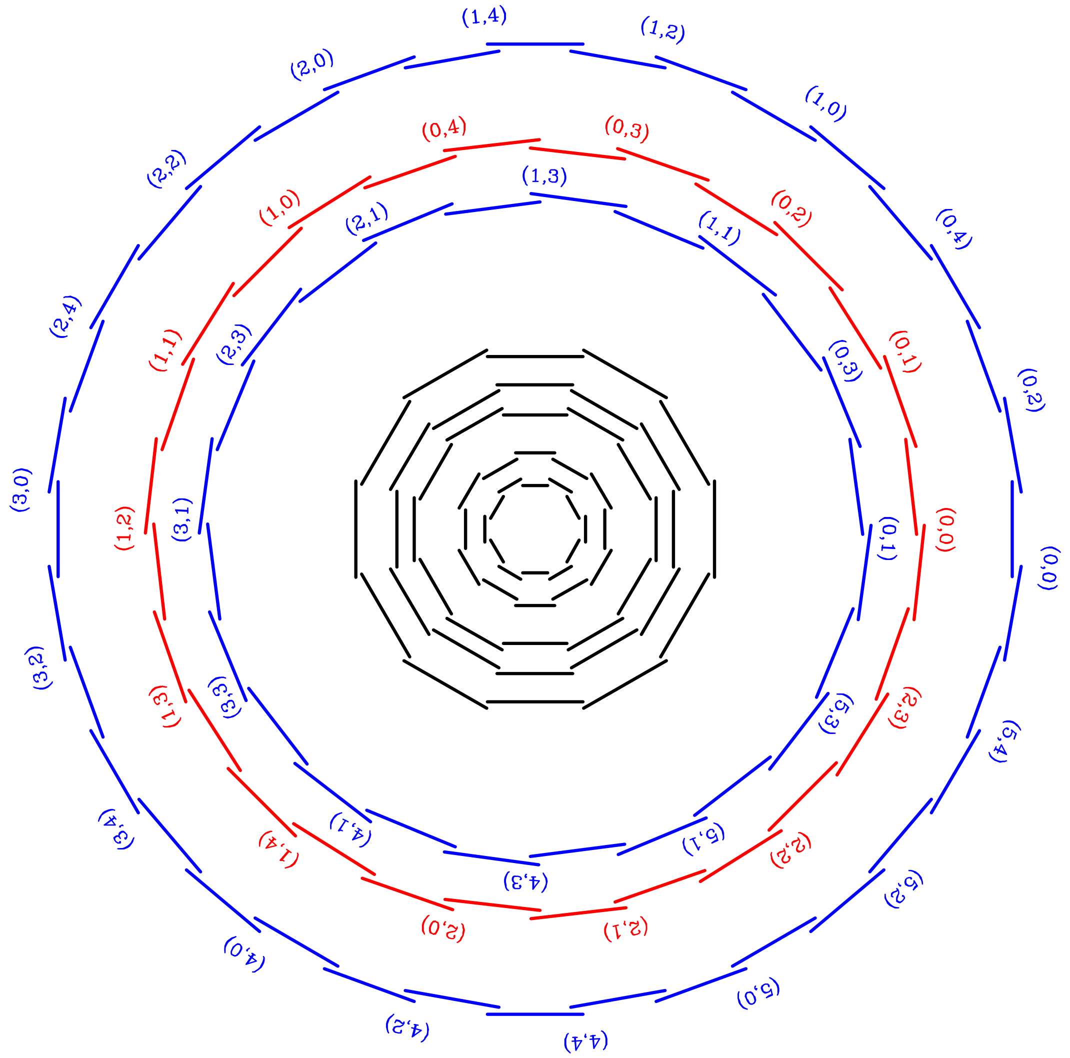

Figure 9: *A schematic view of the HDI connections to ladders in ISL. The numbers near each pair of ladders indicate the portcard number and HDI number (PC,HDI) which are attached to that ladder pair.*

22.1 ISLD - Intermediate Silicon Layers

The ISLD bank closely follows the structure of the SIXD bank (Section 20.1) except for the number of HDI blocks. There are 148 HDI blocks from the ISL itself, 60 from each of the 2 forward barrels and 28 from the central barrel. Furthermore, the ISLD bank contains the data from the Layer00 detector, 48 HDI blocks which are appended after the ISL HDI blocks. Finally, the 3 HDI Blocks from SOC will be appended the end of the bank, and will be present but empty when this detector is not being read out. They will have FIB IDs of 0xf230, 0xf231, and 0xf232 corresponding to 3 HDIs on the 4th Portcard Of the ISL Central West barrel. This gives a total of 199 HDI blocks in the ISLD bank. Within each group above, the HDI blocks are ordered in groups corresponding to the barrels. Within each barrel group, the ordering is by increasing phi. The structure of the ISLD bank is given in Table 60.

"ISLD"	
Bank Number (=1)	
Bank Version	
Bank Length	
Bank Type (I*2)	
Number of Blocks (=199)	
Padding	
Pointer to Block 0 Bits 15:0 (ISL West Barrel)	
Pointer to Block 0 Bits 31:16(ISL West Barrel)	
...	
Pointer to Block 60 Bits 15:0 (ISL Central Barrel)	
Pointer to Block 60 Bits 31:16(ISL Central Barrel)	
...	
Pointer to Block 88 Bits 15:0 (ISL East Barrel)	
Pointer to Block 88 Bits 31:16(ISL East Barrel)	
...	
Pointer to Block 148 Bits 15:0 (L00 West End)	
Pointer to Block 148 Bits 31:16 (L00 West End)	
...	
Pointer to Block 172 Bits 15:0 (L00 East End)	
Pointer to Block 172 Bits 31:16(L00 East End)	
...	
Pointer to Block 196 Bits 15:0 (SOC 1)	
Pointer to Block 196 Bits 31:16(SOC 1)	
...	
Pointer to End of Data Bits 15:0	
Pointer to End of Data Bits 31:16	
Time Since L1 Acc	TS Turn Count
Padding	
HDI Block 0 (First ISL) ...	
... ...	
HDI Block 148 (First L00) ...	
... ...	
HDI Block 196 (First SOC) ...	
... ...	

Table 60: *Structure of the ISLD bank. The first set pointers point to the West Barrel of the ISL, the next to the Central Barrel, then the East Barrel. Layer 00's HDI blocks are appended afterward with the West end of Layer00 first and the East end second. Finally SOCs 3 HDI Blocks are appended. The details of a "HDI Block" are described in Table 21. In some simulation files, the pad word before the first HDI block has been moved to after the last HDI block. Offline software should be able to handle both possibilities without error.*

22.2 ISL Specific ID Encoding

As specified in section 9.3, the conversion between the hardware quantities "portcard" and "HDI" number and the offline quantities "phi-address" and "layer" number differs between SVXII and ISL. This is specified in Fig. 3. Note that the chip id is also needed to completely specify the phi address since bit 4 of the chip id differentiates between the two ladders within an HDI.

22.3 ISLS - ISL Status Bank

As described in section 20 the VRB Block header is stripped off the event and only the HDI data blocks are kept in the data stream. Since the header contains error stati and global event information, for the ISL (and L00 data) a special bank is used for storing this information, the "ISLS" bank. This ISLS parallels exactly the structure of the SIXS bank, described in section 20.3. The only change is the Bank name.

"ISLS"		
Bank Number (=1)		
Bank Version		
Bank Length		
Bank Type (I*4)		
FIB ID0	FIB ID1	Error Byte
FIB′ ID0	FIB′ ID1	Error Byte
...	...	...

Table 61: *Structure of the ISLS bank.*

23 Muon Detector Component Banks

23.1 CMUD - Central Muon Detector

The CMUD bank is a standard TDC bank. Most of the format details for TDC banks are specified in Section 9.2.

In Run II, the Central Muon Detector (CMU) uses the original muon chambers from Run I and provides a coverage of $|\eta| \leq 0.6$. The CMU is built from drift chambers that have a single sense wire oriented roughly parallel to the beamline. The chambers surround the Central Calorimeter in a cylindrical arrangement four layers deep.

The CMUD module ID is defined in Table 62. The CMUD channel ID is defined in Table 63. Side and wedge follow the standard CDF definitions. One calculates wedge from the IDs in two steps. First,

$$\text{wedge} = \text{lowest wedge} + \text{wedge in TDC}.$$

Second, if wedge is greater than 23,

$$\text{wedge} = \text{wedge} - 24.$$

This implies that a TDC will only have channels from two adjacent wedges.

For a given chamber, the values in the "lowest wedge" and "wedge in TDC" bit fields may vary in different CMUD banks. It is only the wedge number calculated from them that is required to always be the same. This gives us extra flexibility when cabling the TDCs.

The four layers are numbered from 0 to 3 in order of distance from the interaction point. Within one layer of one wedge, there are twelve chambers. These chambers are numbered in ϕ order from 0 to 11. This is called the stack number.

Bit	Definition
6:8	Unused
5	Side, West = 0, East = 1
4:0	Index of lowest wedge

Table 62: *CMUD module ID definition.*

Bit	Definition
7:8	Unused
6	Wedge within TDC
5:2	Stack (0 - 11)
1:0	Layer (0 - 3)

Table 63: *CMUD channel ID definition.*

23.2 CMPD - Central Muon Upgrade

The CMPD bank is a standard TDC bank. Most of the format details for TDC banks are specified in Section 9.2.

The Central Muon Upgrade (CMP) consists of a set of muon chambers covering the area where $55^o < \theta < 125^o$. They lie farther from the interaction point than CMU and are behind more shielding. The CMP chambers are rectangular single wire drift tubes. They are all aligned so that the sense wires lie roughly parallel to the beam.

The CMP can be divided into 4 regions: north, top, south, and bottom. These regions are numbered from 0 to 3 in that order. In the north and south regions, the chambers are all oriented so the shortest dimension of the chambers lies horizontal very roughly parallel to a line running from north to south. In the top and bottom regions, the chambers are all oriented so that the shortest dimension of the chambers lies parallel to vertical.

There are 4 layers of chambers. The layers are numbered from 0 to 3 in order of distance from the interaction point.

Each region can be also divided into parts. Within a part all the chambers in a layer are lined up and spaced at regular intervals. This is not true when one goes from a chamber in one part to a chamber in an adjacent part. In both the north and south regions, there is one small part near the bottom with a few chambers and larger part that covers most of the area. The bottom region is broken into three parts. Two small parts at either end and one large part that covers most of the bottom. The top region is broken into six parts. Two small parts at either end. One part over the north yoke and one part over the south yoke. There is a part in the center suspended from the ceiling. There is also another part suspended higher above the north end of the north yoke.

A stack is formed by four chambers, one from each layer. A part is formed from stacks lined up in in a long straight row.

The CMPD module ID is defined in Table 64. The CMPD channel ID is defined in Table 65. The map from the IDs to a geometrical index is given in Table 66. Note that although the definition of the bitfields in the module ID and channel ID were INITIALLY motivated by the electronic channel map, these definitions are not linked to the electronic map. When possible, cabling changes will be accounted for by changing the hardware database (the IDs written into CMPD for a particular chamber should not change).

Bit	Definition
8:4	Unused
3:2	Region
1:0	TDC within region

Table 64: *CMPD module ID definition.*

Bit	Definition
8:7	Unused
6:2	Physical Stack (0 - 23)
1:0	Layer (0 - 3)

Table 65: *CMPD channel ID definition*

Region	TDC in Region	Physical Stack	Logical Stack ϕ ordered index in region	Part
0 North	0	0:6	0:6	0
		7:8	None	
		9:23	7:21	1
	1	0:23	22:45	1
	2	0:15	46:61	1
		16:23	None	
1 Top	0	0:4	0:4	0
		5	None	
		6:14	5:13	1
		15:17	None	
		18:23	14:19	2
	1	0:11	20:31	2
		12:23	32:43	3
	2	0:3	44:47	3
		4:5	None	
		6:23	48:65	4
	3	0:4	66:70	4
		5	None	
		6:11	71:76	5
		12:23	None	
2 South	0	0	None	
		1:5	62:58	1
		6:7	57:56	1
		8:9	None	
		10:23	55:42	0
	1	0:23	41:18	0
	2	0:17	17:0	0
		18:23	None	
3 Bottom	0	0:2	0:2	0
		3:23	3:23	1
	1	0:23	24:47	1
	2	0:13	48:61	1
		14:16	62:64	2
		17:23	None	

Table 66: *This table defines the relationship between the geometric positions of the CMP chambers and the bit fields in the channel ID and the module ID written into the CMPD bank. For example, the 3rd line below the column headings tells one that physical stack 9 is logical stack 7, that physical stack 10 is logical stack 8, and so forth. In the south region, the logical stack numbers are in reverse order. This is significant and intentional. The south region is cabled in reverse order. In the north region, "ϕ order" means bottom to top. In the top region, "ϕ order" means you take all chambers in a part before moving onto the next part. There are some small overlaps between the parts that make this not exactly ϕ order, but it is so much simpler this way that we ignore that small detail. None means that a chamber does not exist with that physical stack number.*

23.3 CSPD - Central Muon Upgrade Scintillation Counters

The CSPD bank is a standard TDC bank. Most of the format details for TDC banks are specified in Section 9.2.

A layer of scintillation counters (CSP) is attached to the outside surface of the CMP Drift Chambers. The CSP counters measure $2.5 \times 15 \times 320\text{cm}^3$, and each counter covers two CMP chambers in width and half of a chamber in length. The number of channels in each region is shown in Table 67.

The CSPD channel ID is defined in Table 68. In each of the 4 regions on each side, the scintillator counters are numbered in ϕ order. The scintillation counter at the lowest value of ϕ is counter 0. In the north region, "ϕ order" means from the bottom to the top. There are no gaps in the counter numbers (as one counts from 0 up, all numbers are associated with a counter that exists until one reaches the counter at highest ϕ). The module IDs in the CSPD bank are not necessary to identify channels and their contents are not specified here. There is sufficient information in the channel IDs to identify all the channels.

Region	West	East
0 North	34	34
1 Top	41	41
2 South	34	34
3 Bottom	34	34

Table 67: *Number of CSP channels per region.*

Bit	Definition
7:8	Region: North = 0, Top = 1, South = 2, Bottom = 3
6	Side: West = 0, East = 1
5:0	Counter

Table 68: *CSPD channel ID definition*

23.4 CMXD - Central Muon Extension Drift Tubes

The CMXD bank is a standard TDC bank. Most of the format details for TDC banks are specified in Section 9.2.

The Central Muon Extension consists of drift tubes (CMX) and scintillation counters (CSX) and provides coverage in the range $0.6 \leq |\eta| \leq 1.0$. CMXD holds data from the drift chambers. Each CMX drift chamber wire lies along a constant value of ϕ. There are two drift chambers at each value of ϕ. Adjacent pairs of drift chambers are separated by 0.625^o in ϕ. The chambers are arranged in 8 layers. The chambers in different layers overlap. A straight trajectory from the primary interaction region through the CMX detector can strike a chamber in all 8 layers, although it will most often strike 6 or 4 chambers.

The CMX looks complicated. The top three quarters of the CMX looks very different from the bottom quarter, but on both top and bottom the pattern described in the previous paragraph is followed. The arches and keystone cover ϕ from roughly -45^o to 225^o. On the arches and keystone each layer lies on the surface of a cone whose axis of symmetry is the beamline. The miniskirt covers ϕ from roughly 225° to 315°. Each layer on the miniskirt lies in a plane. The plane orientation is set by taking a plane perpendicular to the beam axis and rotating it by ±30° about the x axis.

Four geometric indices are used to identify drift chambers.

- side(0:1), wedge(0:23), layer(0:7), wire(0:5)

Side equals 0 on the west side, side equals 1 on the east side. Wedge is the standard CDF wedge number. Wedges are numbered from 0 to 23 in ϕ order around the detector. A wedge is a 15° section. Wedge 0 goes from $\phi = 0^o$ to $\phi = 15^o$. The layers are numbered from 0 to 7 in order of distance from the primary interaction region. Within one layer of one wedge, wires are numbered from 0 to 5 in ϕ order.

A few chambers do not exist, but are assigned indices to keep the numbering scheme simple.

The CMXD module ID is defined in Table 69. The CMXD channel ID is defined in Table 70. One calculates wedge from the IDs in two steps. First,

$$\text{wedge} = \text{lowest wedge} + \text{wedge in TDC}.$$

Second, if wedge is greater than 23,

$$\text{wedge} = \text{wedge} - 24.$$

This implies that a TDC will only have channels from two adjacent wedges.

For a given chamber, the values in the "lowest wedge" and "wedge in TDC" bit fields may vary in different CMXD banks. It is only the wedge number calculated from them that is required to always be the same. This gives us extra flexibility when cabling the TDCs.

Bit	Definition
6:8	Unused
5	Side, West = 0, East = 1
4:0	Index of lowest wedge

Table 69: *CMXD module ID definition.*

Bit	Definition
7:8	Unused
6	Wedge in TDC
3:5	Wire (0:5)
2:0	Layer (0:7)

Table 70: *CMXD channel ID definition.*

23.5 CSXD - Central Muon Extension Scintillation Counters

The CSXD bank is a standard TDC bank. Most of the format details for TDC banks are specified in Section 9.2.

The Central Muon Extension consists of drift tubes (CMX) and scintillation counters (CSX) and provides coverage in the range $0.6 \leq |\eta| \leq 1.0$. CSXD holds data from the scintillation counters. The centerline of each CSX counter lies along a constant value of ϕ. There are significant differences in the configuration of the counters between the arch and miniskirt counters. On the arches and keystone, the scintillation counters (CSX) are attached to the both the inside and outside surfaces of the drift chamber assembly. On the miniskirt, the scintillation counters are only attached to the inner surface of the drift chambers. On the arches and keystone, there is one photomultiplier tube (PMT) per counter. On the miniskirt, there are two PMTs per counter. One on the end closer to the beamline and one on the end farther from the beamline.

The scintillation counters are enumerated using three indices.

- side (0:1), layer(0:1), counter(1:107)

Side equals zero on the west side. Side equals 1 on the east side. The inner layer is layer 0 (inner means closer to the primary interaction region). The outer layer is layer 1. Note that all the miniskirt counters are in layer 0. The counters in one layer on one side are numbered from 1 to 107 in ϕ order, except that counters are counted starting from the bottom of the north arch. The bottom edge of counter 1 is at approximately $\phi = -45^{o}$.

Counters 1 to 73 lie on the arches and keystone. In each layer on the arches, there is a boundary between adjacent counters every 3.75^{o} in ϕ. The nomenclature inherited from run 1 looks like I23W (internal counter 23 on the west side) or X44E (external counter 44 on the east side). Internal means layer 0. External means layer 1.

Counters I01W, X73W, X01E, and I73E have been cut to 1/4 the normal counter width in ϕ. Counters X01W, I73W, I01E, and X73E have been cut to 3/4 the normal counter width in ϕ. The lowest edge of counter 1 lies at roughly $\phi = -45^{o}$ (there are four counters numbered counter 1, one in each of two layers on two sides). This makes the inner and outer layers offset by half a counter width and improves the ϕ resolution.

The counters on the very top of the east side are assigned counter numbers even though they do not exist. This makes the numbering similar on both sides.

The counters on the miniskirt are numbered from 74 to 107. On the miniskirt, the centers of adjacent counters are separated by 2.5^{o} in ϕ. There is a gap of 2.5^{o} between the arch counters and the miniskirt counters. The upper edge of counter 74 lies at roughy $\phi = 227.5^{o}$ and the upper edge of counter 107 lies at roughly $\phi = -47.5^{o}$. Counters 90 and 91 do not exist but are assigned indices (they were removed to make room for SVX cooling pipes).

There are two types of channel. RAW channels use data from a single PMT. Meantime channels include an average time from two PMTs that were hit in coincidence. On the arches, the coincidence is always between two PMTs that are mounted on counters in different layers. On the miniskirt, the coincidence is between two PMTs mounted on the same counter.

The number and configuration of TDCs is not yet known, but module and channel IDs sufficient to identify every channel are defined below. One thing is assumed about the assignment of channels to TDC modules. A TDC module will have channels from only one side. The module ID is defined in Table 71. The channel ID is defined in Table 72

Bit	Definition
6:8	Unused
5	Side, West = 0, East = 1
4:0	TBD

Table 71: *CSXD module ID definition. The bit field "TBD" is to be determined. It is not necessary to identify channels and will be used only for debugging purposes to uniquely identify TDC modules. There is enough information in the channel ID and the side bit to identify all the channels.*

Bit	Definition
8	Coincidence type (0=RAW, 1=mean-timed)
7	End(0:1) or Layer (0:1)
6:0	Counter (1:107)

Table 72: *CSXD channel ID definition. For RAW miniskirt hits, bit 7 gives the "End". For the PMT nearer the beam, "End" is zero. For the PMT farther from the beam, "End" is one. For all arch hits, bit 7 gives the layer. For miniskirt meantime hits, bit 7 is undefined. For meantime arch hits, the counter and layer bitfields always identify the counter at the lower value of ϕ.*

23.6 Intermediate Muon Detector

The Intermediate Muon Detector consists of 3 detectors corresponding to three raw data banks: BMUD, BSUD, and TSUD. The detectors are all read out using the same TDCs so the basic data word structure is the same for each of these systems. The channel counts for the various systems can be summarized as follows:

- Barrel Drift Chambers (BMU), 864 channels per side
- Barrel Counters (BSU), 216 channels per side
- Toroid Counters (TSU), 144 channels per side

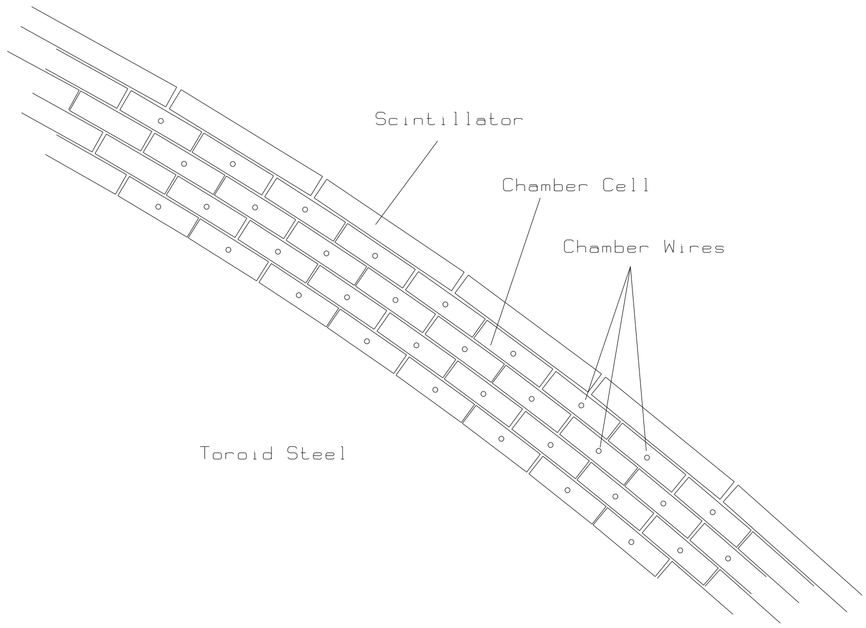

Figure 10: *View of the Intermediate Muon Barrel Detector. The detector consists of four layers of wire chambers followed by a layer of scintillation counters. The detector is arranged in a cylindrical formation and surrounds the toroids.*

23.6.1 BMUD - Intermediate Muon Barrel Drift Chambers

The BMUD bank is a standard TDC bank. Most of the format details for TDC banks are specified in Section 9.2.

The Barrel Drift Chambers (BMU) form a cylindrical arrangement of four layers surrounding the forward toroids (see Figure 10).

The BMUD module ID is defined in Table 73. The BMUD channel ID is defined in Table 74.

Bit	BMUD TDC Header
6:8	Unused
5	Side, West = 0, East = 1
4:0	Index of lowest wedge

Table 73: *BMUD module ID definition.*

Bit	BMUD TDC Data
7:8	Unused
6	Wedge within TDC
5:2	Stack (0 - 11)
1:0	Layer (0 - 3)

Table 74: *BMUD channel ID definition.*

Side and wedge follow the standard CDF definitions. One calculates wedge from the

IDs in two steps. First,

$$\text{wedge} = \text{lowest wedge} + \text{wedge in TDC}.$$

Second, if wedge is greater than 23,

$$\text{wedge} = \text{wedge} - 24.$$

For a given chamber, the values in the "lowest wedge" and "wedge in TDC" bit fields may vary in different BMUD banks. It is only the wedge number calculated from them that is required to always be the same. This gives us extra flexibility when cabling the TDCs.

The four layers are numbered from 0 to 3 in order of distance from the interaction point. Within one layer of one wedge, there are twelve chambers. These chambers are numbered in ϕ order from 0 to 11. This is called the stack number.

Some chambers described by the above numbering system do not exist, but are assigned indices to keep the numbering scheme simple. In particular, all chambers in wedges 15 to 20 do not exist. There might be a small number of additional missing chambers.

23.6.2 BSUD - Intermediate Muon Barrel Scintillation Counters

The BSUD bank is a standard TDC bank. Most of the format details for TDC banks are specified in Section 9.2.

A total of 432 Barrel Scintillation Counters (BSU) are mounted on the outside of the BMU. A combined layout of the BMU and the BSU is shown in Figure 10.

The BSUD module ID is defined in Table 75. The BSUD channel ID is defined in Table 76.

Bit	Definition
8:6	Unused
5	Side, West = 0, East = 1
4:0	Index of lowest wedge (0 - 23)

Table 75: *BSUD module ID definition.*

Bit	Definition
8:7	Unused
6:4	Wedge within TDC (0 - 5)
3	Layer, inner=0, outer=1
2:0	Counter (0 - 5)

Table 76: *BSUD channel ID definition.*

Side and wedge follow the standard CDF definitions. One calculates wedge from the IDs in two steps. First,

$$\text{wedge} = \text{lowest wedge} + \text{wedge in TDC}.$$

Second, if wedge is greater than 23,

$$\text{wedge} = \text{wedge} - 24.$$

For a given chamber, the values in the "lowest wedge" and "wedge in TDC" bit fields may vary in different BSUD banks. It is only the wedge number calculated from them that is required to always be the same. This gives us extra flexibility when cabling the TDCs.

The BSU counters are twice as wide as the BMU drift chambers in the ϕ direction. In the η direction, there are two counters for each for each chamber. On each side, one set of counters covers $|\eta|$ between 1.0 and 1.25 and the other set of counters covers $|\eta|$ between 1.25 and 1.5. The set at lower $|\eta|$ is at a slightly smaller radius to allow it to fit inside the CMX. There is also a slight overlap between the counters at low and high $|\eta|$. The counters at lower $|\eta|$ are layer 0 and the counters at higher $|\eta|$ are layer 1.

Within one layer of one wedge the counters are numbered from 0 to 5 in ϕ order.

Some counters described by the above numbering system do not exist, but are assigned indices to keep the numbering scheme simple. In particular, all counters in wedges 15 to 20 do not exist. There might be a small number of additional missing counters.

23.6.3 TSUD - Intermediate Muon Toroid Scintillation Counters

The TSUD bank is a standard TDC bank. Most of the format details for TDC banks are specified in Section 9.2.

The TSUD module ID is defined in Table 77. The TSUD channel ID is defined in Table 78.

Bit	Definition
5	Side, West = 0, East = 1
4:0	Index of lowest wedge (0 - 23)

Table 77: *TSUD module ID definition.*

Bit	Definition
6:3	Wedge within TDC (0 - 11)
2	Layer, inner=1, outer=0
1:0	Counter (0 - 2)

Table 78: *TSUD channel ID definition.*

Side and wedge follow the standard CDF definitions. One calculates wedge from the IDs in two steps. First,

$$\text{wedge} = \text{lowest wedge} + \text{wedge in TDC}.$$

Second, if wedge is greater than 23,

$$\text{wedge} = \text{wedge} - 24.$$

For a given chamber, the values in the "lowest wedge" and "wedge in TDC" bit fields may vary in different TSUD banks. It is only the wedge number calculated from them that is required to always be the same. This gives us extra flexibility when cabling the TDCs.

The toroid counters are located in between the toroids. There are two sets of these counters on each side. Each set covers a different range of η. The outer set of toroid counters covers $1.3 \leq |\eta| \leq 1.5$, while the inner set of counters has a smaller radius and covers $1.5 \leq |\eta| \leq 2.0$. The inner set is defined to be layer 1. The outer set is defined to be layer 0. (Note the nonstandard numbering of layers, usually they are numbered from inner to outer. This was not the original plan, but a lot of data was recorded this way before anyone noticed the mistake and we decided to keep all the data consistent and not change the DAQ.)

As of May 2002, the inner TSU layer does not exist. It is not currently known when the inner layer will be built. It is possible that it will never be built.

Each counter covers 5 degrees in ϕ. Within one layer of one wedge the counters are numbered from 0 to 2 in ϕ order.

A small number of counters described by the above numbering system may not exist, but are assigned indices to keep the numbering scheme simple.

24 COT Central Outer Tracker

24.1 COTD - TDC data

The COTD bank is a standard TDC bank. Most of the format details for TDC banks are specified in Section 9.2.

The COT is a cylindrical drift chamber with 8 superlayers. Half the superlayers have sense wires that run parallel to the axis of the cylinder. These are called axial superlayers. The rest have wires oriented at a small angle to the axis of the cylinder to allow for 3 dimensional tracking. These are called stereo superlayers. The superlayers are numbered from 0 to 7 in the order of their distance from the cylinder axis. (Unfortunately, in some places the superlayers are numbered from 1 to 8 instead of 0 to 7. 0 to 7 is used in this document and in most offline documentation and code. 1 to 8 is used in most online documentation and code. It is probably too late to fix this. Be careful.)

Within a superlayer, there are 12 layers. A cell is defined to include one wire from each of the layers. The wires in a cell are numbered from 0 to 11, also in order of radius. The cells within a superlayer are numbered in ϕ order. The x axis crosses cell number 0. The number of cells per superlayer is different for each superlayer as shown in Table 79. From this, one can calculate that the COT has 30,240 channels; 16,128 in four axial superlayers and 14,112 in four stereo superlayers.

One TDC module reads out all the channels on eight cells. The eight cells read out by one TDC module always lie in a single superlayer. In superlayers 0, 2, 4, 5, and 6, there is one TDC module that handles cells 0 to 7, the next TDC handles cells 8 to 15, the next TDC handles cells 16 to 23. This pattern continues to the end of each superlayer. There is a similar pattern in the other superlayers (1, 3, and 7), except the pattern starts at cell 4 instead of cell 0. In these superlayers, there is a TDC that handles cells 4 to 11, the next TDC handles cells 12 to 19, the next TDC handles cells 20 to 27. This pattern also continues around each superlayer. For these superlayers, the last TDC holds the last four channels in the superlayer and cells 0 to 3. This gives a total of 315 TDC modules. Table 79 shows the number of TDC modules per superlayer. After the split wire upgrade, there will be two channels for each wire in superlayers 0 and 1. There will be twice as many TDC modules for those layers, but on each side the assignment of channels to TDCs will be the same.

Superlayer	# cells	TDC modules
0	168	21(42)
1	192	24(48)
2	240	30
3	288	36
4	336	42
5	384	48
6	432	54
7	480	60

Table 79: For each superlayer, the total number of cells and TDC modules is given. Numbers in parentheses are the number of modules required for the "split-wire" upgrade.

A unique address for the module is contained in the module ID in the TDC header word.

The module ID is defined in Table 80. Each data word contains a channel ID. The channel ID is defined in Table 81.

Bit	Definition
8:6	Superlayer (0 - 7)
5:0	TDC Module (0 - 59)

Table 80: *Definition of the Module ID in the TDC header word. The "TDC Module" number referred to in this table is equal to the lowest cell number handled by the TDC divided by 8 and always rounding down to the nearest integer. The modules that include the four lowest and four highest cells in a superlayer are a special case. In this case, one uses the lowest cell number out of the four highest cells in the superlayer to calculate "TDC Module" number. Same as in the normal case, divide by 8 and round down. After the split wire upgrade, the west sides of the first two superlayers will be numbered as above. For the east first superlayer split wire upgrade modules, 21 is added to differentiate the east modules from the west modules. Similarly, for the east second superlayer split wire upgrade modules, 24 is added.*

Bit	Definition
6:4	cell offset (0 - 7)
3:0	wire (0 - 11)

Table 81: *Bit assignment for the channel ID. The cell offset is 0 for the cell at lowest ϕ in this TDC module and is incremented by one as one progresses to each adjacent cell at higher ϕ. In the special case where the TDC module handles the four highest and four lowest cells in a superlayer, the four highest cells get cell offsets 0 to 3. The four lowest cells (cells 0 to 3) are have a "cell offset" from 4 to 7.*

For superlayers 0, 2, 4, 5, and 6

$$\text{cell} = (\text{cell offset}) + 8 * (\text{TDC Module}).$$

For most channels in superlayers 1, 3, and 7

$$\text{cell} = 4 + (\text{cell offset}) + 8 * (\text{TDC Module}).$$

For cell offset greater than 3 in the last TDC module in superlayers 1, 3, and 7

$$\text{cell} = (\text{cell offset}) - 4.$$

CDF Note 4507 contains many interesting details about the mapping to electronic channels for the COT. Reconstruction software that accesses the COTD bank should not depend on any details in CDF 4507 that are not also specified in this note. Reconstruction software should be written so that changes to the DAQ electronics do not effect reconstruction results. This is discussed in more detail Section 9.2.

24.2 CTMD - COT Pressures, Voltages, Currents

In each event, the CTMD bank stores the readings from several COT monitors: pressure, high voltage on the sense wires, high voltage on the potential wires, current on the sense wires, and possibly temperatures on several points (electronics or the drift chamber itself) in the future. These quanties affect the COT dE/dx resolution and therefore are critical for the dE/dx corrections. A 64-channel ADC VME card, sitting in the COT pulser crate, is the interface between run control and the monitors.

The CTMD banks have a header and block pointers as described in

Sections 4 and 4.1. These are I*2 type banks (the data words and block pointers are 16 bit integer words). Each CTMD bank has only one block.

The format of a CTMD bank is specified in Table 82. Users must keep in mind that this format might be changed in the future. The conversion of the short integer to the physical value is specified in Table 83.

The format for this bank was not stable before run 142535 (5 April 2002). The hardware was also not fully installed, so CTMD banks before that run contain nonsense.

"CTMD"	
Bank Number	
Bank Version (=0)	
Bank Length	
Bank Type (I*2)	
Number of Blocks (=1)	
Pointer to Block 0	
Pointer to End of Data	
Padding	
0	Average sense voltage, Superlayer 0
1	Average sense voltage, Superlayer 1
...	...
7	Average sense voltage, Superlayer 7
8-11	Pressures inside the COT
12	Average pressure (truncated mean as used for feedback)
13	Not used
14	Average potential voltage, Superlayer 0
15	Average potential voltage, Superlayer 1
...	...
21	Average potential voltage, Superlayer 7
22	Average sense current, excluding S11, Superlayer 0
23	Average sense current, excluding S11, Superlayer 1
...	...
29	Average sense current, excluding S11, Superlayer 7
30-63	Unused

Table 82: *CTMD data format. The number in the left column is an index from the start of the data block and also the ADC channel number. The 4 pressure measurements are from transducers that make redundant measurements of the pressure in the COT.*

Physical Units	conversion
pressure:psi	$p = 14.5 + (1.0/0x1000) * adc$
voltage: %	$v = (128/4096) * (adc + 2048)$
current: μA	$i = .01 * exp(log(400000.)/0x1000 * (adc + 0x800))$

Table 83: *Conversions from 16 bit integers that are the data words in the bank to physical quantities. In the formulas, 'adc' represents the value in the data word. Pressure is in pounds per square inch (psi). The voltages are measured as a percentage of the nominal voltage. '0x' indicates the number that follows is a hexidecimal number.*

24.3 CTAD - COT Flash ADC Data

COT FADC data come from the COT analog monitor system, which provides analog signals corresponding to individual COT sense wires. It does not provide signals for every sense wire, but rather for 96 sense wires distributed evenly among superlayers and quadrants.[1] These signals represent the input to the COT discriminator electronics (ASDQ cards) which are read by the COT TDC modules. The purpose of this system is to improve tracking through studies of chamber noise and ASDQ performance. This is a diagnostic bank used by COT experts. It is not designed to be used in a physics analysis or during track reconstruction.

The CTAD bank format includes a header section, block pointer section, and card pointer sections which have exactly the same formats as in TDC banks. Section 9.2 describes the TDC bank format. However, the content inside the card data sections is in a completely different format. The format of the card data section is shown in Table 85.

The bank type is I*4, the bank is composed of 32-bit words. These words hold voltage measurements. The voltages are recorded every 2 ns. Each of these 2 ns time periods is called a "tick". Table 85 shows how these voltages are stored into the bank. The voltage is stored in an 8 bit integer that has a value between 0 and 255 in units of counts. The scale of the FADC output is 2.56 counts/mV.

There are no channel IDs or card IDs stored in the bank. The data is identified by its location in the bank. The data blocks are numbered starting at 0 and in the order they appear in the bank. Cards within a block are numbered in the order they appear in the block starting at 0 within each block. Note that blocks and cards are counted in this numbering scheme if the block or card pointer is present even if the corresponding block or card data section is empty. Channels are numbered in the order they appear in a card starting at 0. One has to know the block number, card number, and channel number corresponding to each sense wire to use this bank. This correspondence is not fixed and will vary from run to run. One must consult a COT expert to get the correct correspondence for the run one is interested in. Specifically, the number of blocks, the order of blocks, number of cards, order of cards, the order of channels and the channels in a card may all vary.

24 sense wires are connected to each of 4 primary FADC modules. There is also a test module. Each FADC module can be configured to read out any 6 of the 24 channels connected to it. In the current configuration (October 2002), each of these 5 modules is in a different crate and so appears in the bank in a different block, and each block contains one card. Table 84 shows the region of the COT containing the sense wires attached to each of the modules. The configuration of modules is not permanent and will be modified in the future. One must contact a COT expert to know the configuration at a particular time.

FADC Module number	Superlayers	Quadrants
0	1,4,5, or 8	A or D
1	2,3,6, or 7	B or C
2	1,4,5, or 8	B or C
3	2,3,6, or 7	A or D
4	test module	

Table 84: COT regions covered by CTAD modules as of October 2002.

[1] See Section 24.1 for a further description of the Central Outer Tracker.

word	bits 0:7	bits 8:15	bits 16:23	bits 24:31
0	Channel 0 Header (0xFADCFADC)			
1	V. at tick 11	V. at tick 13	V. at tick 15	V. at tick 17
2	V. at tick 8	V. at tick 10	V. at tick 12	V. at tick 14
3	V. at tick 19	V. at tick 21	V. at tick 23	V. at tick 25
4	V. at tick 16	V. at tick 18	V. at tick 20	V. at tick 22
5	V. at tick 27	V. at tick 29	V. at tick 31	V. at tick 33
6	V. at tick 24	V. at tick 26	V. at tick 28	V. at tick 30
...				
60	V. at tick 240	V. at tick 242	V. at tick 244	V. at tick 246
61	V. at tick 251	V. at tick 253	V. at tick 255	V. at tick 257
62	V. at tick 248	V. at tick 250	V. at tick 252	V. at tick 254
63	V. at tick 259	V. at tick 261	V. at tick 263	V. at tick 265
64	Channel 1 Header (0xFADCFADC)			
65	V. at tick 11	V. at tick 13	V. at tick 15	V. at tick 17
66	V. at tick 8	V. at tick 10	V. at tick 12	V. at tick 14
...(Same structure as channel 0)				
126	V. at tick 248	V. at tick 250	V. at tick 252	V. at tick 254
127	V. at tick 259	V. at tick 261	V. at tick 263	V. at tick 265
...				
320	Channel 5 Header (0xFADCFADC)			
321	V. at tick 11	V. at tick 13	V. at tick 15	V. at tick 17
322	V. at tick 8	V. at tick 10	V. at tick 12	V. at tick 14
...(Same structure as channel 0)				
382	V. at tick 248	V. at tick 250	V. at tick 252	V. at tick 254
383	V. at tick 259	V. at tick 261	V. at tick 263	V. at tick 265

Table 85: *Format of the data from 1 card in a CTAD bank. There are 6 channels per card and 64 words per channel. The first word in each channel a header word. All the header words have the same value, 0xFADCFADC. Four voltage readings are stored in each of the data words following the header. Voltages measured at even ticks are stored in time order in the even numbered words. Voltage measured at odd ticks are stored in time order in odd numbered words. The even ticks are stored starting with tick 8 in word 2. The odd ticks are stored starting at tick 11 in word 1. Note this leaves some gaps, ticks 0 to 7, 9, 256, 258, 260, 262, and 264 are not stored in the bank. One tick represents 2 ns and indicates the time. Each voltage is stored in 8 bits and packed into a 32-bit word as shown above.*

25 TOFD - Time of Flight Raw Data Format

The Time-of-Flight subdetector consists of 216 bars of scintillator located between the COT and the cryostat. Photomultiplier tubes are attached at both ends of every scintillator bar. Each PMT is instrumented with a preamplifier that drives the signals out of the detector on shielded twisted pair cables.

The PMT pulses are received by the front end electronics, located on transition boards in the endwall VME crates. Each channel of electronics contains a discriminator that starts a time-to-amplitude converter (TAC) and a gated current switch to generate a current proportional to the input pulse height. Both the output of the TAC and the current switch are digitized using the ADMEM board connected via the P3 connector on the backplane. Each PMT channel is associated with two channels on the ADMEM board: one used for the timing measurement, the other used for a measure of the pulse height. We do not anticipate the ability to digitize multiple hits in a single bunch crossing at any time in the foreseeable future.

The required timing precision of the electronics is of the order of 25 ps and the time interval over which hits are expected is about 50 ns. Hence, the amplitude of the TAC must be digitized using at least an 11 bit ADC. The ADMEM board uses CAFE cards to integrate and digitize PMT pulses from the calorimeters. Although these have a dynamic range of 13 bits, they only use a 10 bit ADC and therefore do not provide the precision required by Time of Flight. Instead, the channels used for timing on the TOF ADMEM boards are instrumented using “deCAF” cards which provide 12 bits of precision using an AD9225 ADC. These cards do not use QIE chips, nor do they require the flash RAM used on the CAFE cards.

For measuring the charge of the PMT pulses, unmodified CAFE cards are used to digitize the output of the integrator. This information will be used primarily for performing slewing corrections to the times digitized by the deCAF cards.

25.1 Channel Mapping

The 216 bars of scintillator are numbered 0 to 215 in increasing azimuthal angle, ϕ. The boundary between bars 0 and 215 lies in the x-z plane. The *nominal* azimuthal positions of the bars are calculated assuming each bar subtends an angle of $\Delta\phi = 360°/216$. Thus, the angular extent of bar i is from $\phi = i\Delta\phi$ to $(i+1)\Delta\phi$. The *actual* locations of the bars will be determined by a survey which will measure their position in space relative to the COT end-plates.

The electronics channels are associated with the 432 PMT's attached to the bars of scintillator. PMT's 0 to 215 are associated with the PMT's on the east end of the detector and PMT's 216 to 431 are on the west end. These channels are read out by front end electronics located in crates mounted on the end-walls of the detector. Each crate instruments the PMT's at one end of the detector in a quadrant in ϕ as described in Table 86.

25.2 TOFD Bank Format

Although the TOF electronics uses ADMEM boards for digitizing the time and charge of PMT pulses, the bank format differs from that used by the other QIE-based subdetectors. This difference lies in the fact that there are two ADMEM channels associated with each

PMT number	Bars	ϕ range	End-wall crate
$0-53$	$0-53$	$0-90°$	Upper North East
$54-107$	$54-107$	$90-180°$	Upper South East
$108-161$	$108-161$	$180-270°$	Lower South East
$162-215$	$162-215$	$270-360°$	Lower North East
$216-269$	$0-53$	$0-90°$	Upper North West
$270-323$	$54-107$	$90-180°$	Upper South West
$324-377$	$108-161$	$180-270°$	Lower South West
$378-431$	$162-215$	$270-360°$	Lower North West

Table 86: Mapping between PMT numbers, nominal positions of the scintillator bars and the associated front-end electronics crates.

PMT which will never be read out separately. Furthermore, the channels used for timing do not use QIE chips and are instead implemented using a 12 bit ADC. Thus, although standard ADMEM boards are used to read out the TOF front end electronics, the TOF data bank is unique to the TOF system.

Table 87 shows the structure of the TOFD bank in memory. The TOFD bank has bank header words and block pointers as described in Sections 4 and 4.1. Except for the bitfields in the header and data words, the format of the TOFD bank is identical to the TDC bank format. We do not specify the order in which the blocks are written nor do we specify the order in which data from the modules are written into each block. The association of a particular module with a group of channels is made based on the module ID field in the module header word, which is described on the page following the bank structure table.

The maximum data volume, in which the data from all 432 channels is stored, is 2216 bytes (including 10 words in the block pointer section, 64 words in the module pointer sections, 48 ADMEM header words, and 432 data words, not including the bank header words and type words).

	''TOFD''	
	Bank number	
	Index to next bank with same name (=0)	
	Bank length	
	Bank type (I*4)	
	Block count word (=8)	
	Pointer to block 0	
	Pointer to block 1	
	...	
	Pointer to block 7	
	Pointer to end-of-data	
	Number of TOF ADMEM boards, block 0 (=6)	
	Pointer to first TOF ADMEM	
	...	
	Pointer to last TOF ADMEM	
	Pointer to end of block 0	
*	TOF ADMEM header word	
*	TDC data	ADC data
	...	
	Number of TOF ADMEM boards, block 7 (=6)	
	Pointer to first TOF ADMEM	
	...	
	Pointer to last TOF ADMEM	
	Pointer to end of block 7	
*	TOF ADMEM header word	
*	TDC data	ADC data
	...	
	...	

Table 87: Structure of the TOFD bank. Fields identified with * are described in following tables.

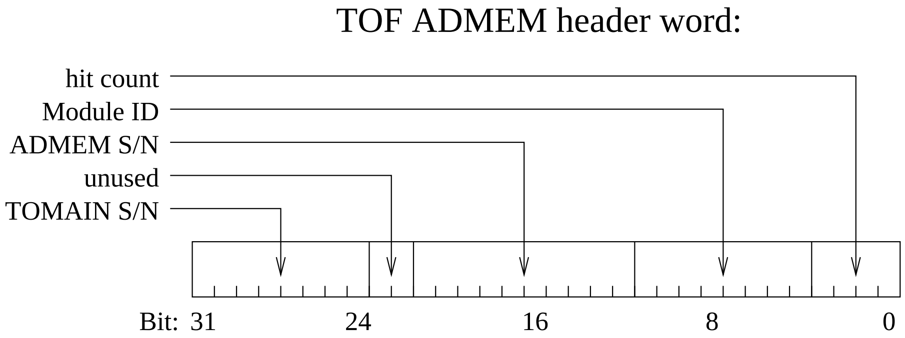

Table 88: Bit fields in the TOF ADMEM module header word.

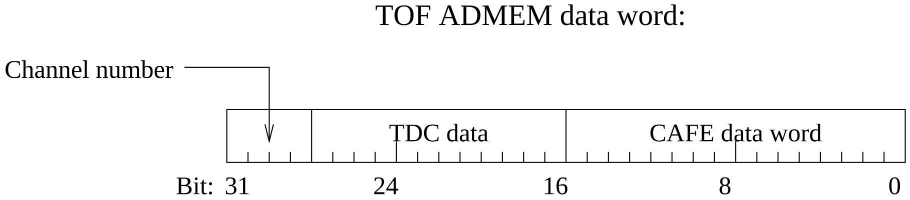

Table 89: Bit fields in the TOF ADMEM data word.

The data blocks from each end-wall crate contain the data from each of the six ADMEM boards used by TOF. The data from each board will consist of a header word followed by hit data, if any. There will always be a single header word present even if there are no hits registered by a particular ADMEM board. The format of this header word is shown in Table 88. Since only 9 PMT channels will be instrumented by a single transition/ADMEM board combination, only 4 bits are required for the hit count in the module header word.

The module header word consists of a 4-bit hit-count field that contains the number of data words to follow. The module ID is stored in bits 4..11, followed by the ADMEM serial number in bits 12..21 and the transition board serial number in bits 24..31. Bits 22 and 23 are currently unused. The module ID number is used to associate hit data with the physical channel number. It remains unchanged even when a particular ADMEM or transition board is replaced. The association between the module ID's and groups of channels is described in the following paragraph.

The format of each data word, corresponding to a hit channel, is shown in Table 89. The lower 16-bits correspond exactly to the data read from the CAFE cards on the ADMEM board for channels which digitize the charge measurement. The TDC data corresponds to the data from the 12-bit ADC which digitizes the TAC. In this scheme, there are 4 bits left over which can be used to identify which local channel on the module was hit. Using both the channel number from this word and the module ID from the module header word, the PMT channel can be determined as follows:

$$\begin{aligned} \texttt{MODULE\_ID} &= \texttt{PMT\_NUMBER}/9 \\ \texttt{CHANNEL\_NUMBER} &= \mathrm{MOD}(\texttt{PMT\_NUMBER}, 9) \\ \texttt{PMT\_NUMBER} &= \texttt{MODULE\_ID} * 9 + \texttt{CHANNEL\_NUMBER}. \end{aligned}$$

The signals from a triplet of three PMT channels are carried out of the detector on a cable consisting of three sets of shielded twisted pairs in a common jacket. These are attached to a PC board which plugs onto a set of three adjacent PMT's in ϕ. Because the

PMT channel	East ADMEM channels	West ADMEM channels
i	$0,1$	$4,5$
$i+1$	$2,3$	$2,3$
$i+2$	$4,5$	$0,1$
$i+3$	$6,7$	$10,11$
$i+4$	$8,9$	$8,9$
$i+5$	$10,11$	$6,7$
$i+6$	$12,13$	$16,17$
$i+7$	$14,15$	$14,15$
$i+8$	$16,17$	$12,13$

Table 90: Association between PMT channel numbers and ADC and TDC channels on the TOF ADMEM boards for the east and west ends of the detector.

same PC board and cable assembly is used for east and west ends of the detector, the order in which the PMT channels are associated with the channels on the ADMEM board are different for each end. This association is shown in Table 90.

26 Cherenkov Luminosity Counter Component Banks

The luminosity monitor for CDF in Run II is an array of gaseous Cherenkov Luminosity Counters (CLC) [5]. The detector is composed of two subdetectors. Each subdetector is positioned within the three-degree hole between the endplug calorimeter and the beam pipe. There are 48 counters in each subdetector, and these counters are arranged into three rings (16 counters per ring). Each counter in the shape of a cone (one or two meters in length) is made up of aluminized mylar and is filled with Isobutane gas. These cones are oriented in a way such they are pointing towards the interaction point (IP), with the smaller end of the cone closer to IP. Cherenkov photons produced by charged particles in each cone are collected at the other end (further away from IP) into a photomultiplier tube (PMT). Figure 11 shows the side view of one of the two CLC subdetectors.

The signals from PMT of each counter are sent to transition boards that are attached to the back of the CLC readout crate. Each input signal is split into three, one for amplitude measurement, one for timing measurement, and another one is sent to the CDF online luminosity crate for an independent online luminosity measurement. The amplitude measurement is performed in the ADMEM/CAFE board (designed for the CDF calorimeters). Each ADMEM/CAFE board will be used to handle 16 CLC counters (counters in one ring), and therefore there will be six of these boards in the CLC front end crate to handle all of the 96 counters. The amplitude measurements will be stored in the CLAD bank.

It is planned to use the Michigan TDC together with a time stretcher to do the timing measurement for CLC. This TDC is the same as the ones that are used for COT and MUON detectors. All the 96 counters will be handled by one TDC board. The timing measurement will be stored in the CLTD bank.

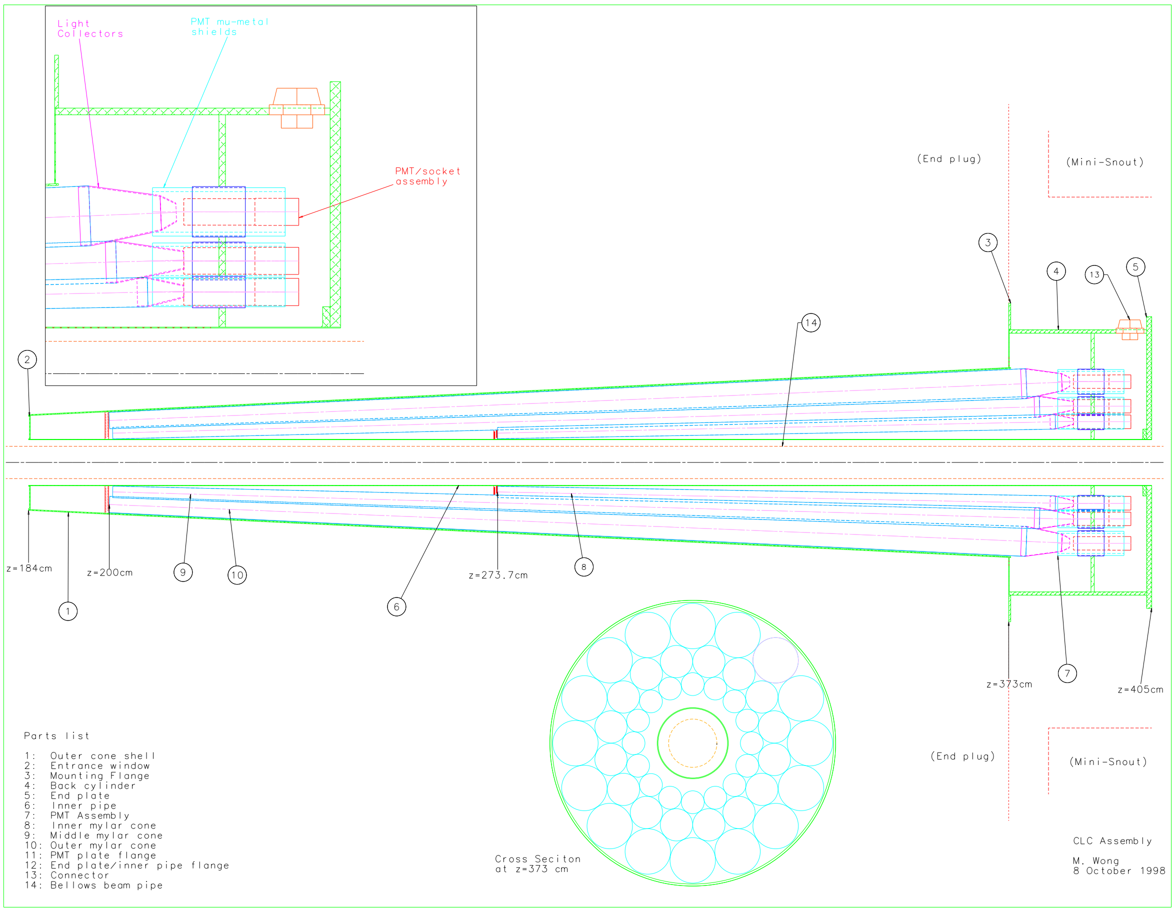

Figure 11: *The side view of one of the two CLC subdetectors.*

26.1 CLAD - CLC Amplitude Bank

The bank structure of the “CLAD” bank is specified in Table 91. The block consists of repeated instances of a 16 bit channel ID followed by a 16 bit data word. The bit assignments for the data word are shown in Table 92. In each 16 bit data word, ten bits are used for amplitude measurement, and six bits are used for the luminosity calculated by the Xilinx FPGA (Field Programmable Gate Array) in the ADMEM board [6]. The channels are numbered from 0 to 95, with 0 to 47 (48 to 95) for the counters in the subdetector positioned at the west (east) end of CDF. Figure 12 shows the correspondence of the channel ID to the counters for the subdetectors located at the west and east ends of CDF.

“CLAD”	
Bank Number	
Bank Version	
Bank Length	
Bank Type (I*2)	
Number of Blocks (=1)	Pointer to Block 0
Pointer to End of Data	Padding
Channel ID	Data Word 0
...	
Channel ID	Data Word 95

Table 91: *Structure of the CLAD bank. There is only one block.*

Bit	Data Word
10:15	Luminosity from FPGA
0:9	Amplitude

Table 92: *Bit assignment for the CLAD data word.*

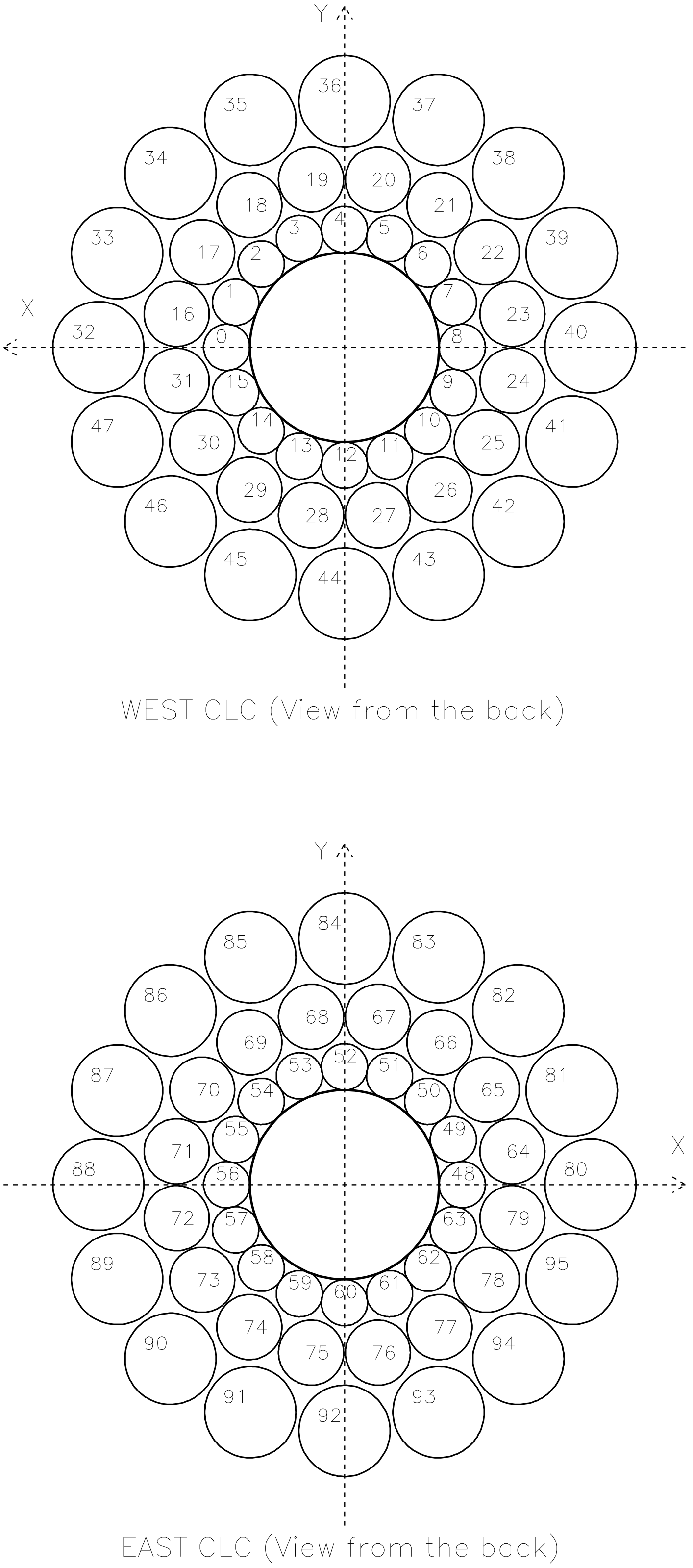

Figure 12: *The correspondence of the channel ID to the counters for the CLC subdetector located at the west and east ends of CDF. In both figures, the arrow on the x axis points roughly north.*

26.2 CLTD - CLC Timing Bank

The bank structure of the "CLTD" bank is specified in Table 93. The structure (including the bitfields in the TDC header word and each data word) are exactly the same as for all other TDC banks (see Section 9.2). The bits in the TDC data word that are used for Channel ID will be defined in the same way as in the CLAD bank. The Channel ID will go from 0 to 95. There is only one module and one block in the CLTD bank.

"CLTD"
Bank Number
Bank Version
Bank Length
Bank Type (I*4)
Number of Blocks (=1)
Pointer to Block 0
Pointer to End of Data
Number of TDCs (=1), Block 0
Pointer to first TDC
Pointer to End of Block
Header Word, TDC 0
TDC Data
...

Table 93: *Structure of the CLTD bank.*

26.3 CLLD - CLC Online Luminosity Bank

The format is valid starting at run 114627. Older data may contain CLLD banks with a different format.

The luminosity monitor provides instantaneous and integrated luminosity for both CDF (online and offline) and the Tevatron (online). To determine luminosity for online consumption, scalers are implemented in the Xilinx FPGA chips on the ADMEM board. Each FPGA handles four CLC channels. The scalers are incremented at every bunch crossing by the number of hits (or particles) measured in these four channels. The counts accumulated in these scalers, which are stored in the CLAD bank, the measured amplitudes of a few channels through the ADMEM Level-1 outputs, the GLIVE and bunch-zero (B0) signals are sent to a memory module in the CDF online luminosity crate.

Signals from the transition boards (in the CLC readout crate) are sent to a scaler module in the online luminosity crate for a more straight forward albeit rudimentary luminosity measurement. The cpu in the online luminosity crate takes in the information from the memory and scaler modules and calculates :

- $\mathrm{L}_{\mathrm{inst}}^{\mathrm{GLIVE}}$ instantaneous luminosity bunch by bunch (GLIVE is present)
- $\mathrm{L}_{\mathrm{total}}^{\mathrm{GLIVE}}$ integrated luminosity bunch by bunch (GLIVE is present)
- $\mathrm{L}_{\mathrm{inst}}$ instantaneous luminosity bunch by bunch

- L_{total} integrated luminosity bunch by bunch
- $< L_{inst}^{GLIVE} >$ average instantaneous luminosity (GLIVE is present)
- $< L_{total}^{GLIVE} >$ average integrated luminosity (GLIVE is present)
- $< L_{inst} >$ average instantaneous luminosity
- $< L_{inst} >$ average integrated luminosity
- $< L_{inst}^{scaler} >$ average instantaneous luminosity (from scaler)
- $< L_{inst}^{scaler} >$ average integrated luminosity (from scaler)

The luminosity measurements calculated in the online luminosity crate cpu are in floating point type. They are then encoded into the 32-bit integer words which are stored in the CLLD bank. Table 94 defines the bit fields in the 32-bit words:

31 30 5	4	3 2 1 0
Mantissa	Sign (+/-)	Power′

Table 94: *Bit field definitions for the 32-bit words holding the luminosity measurements in CLLD.*

These bit fields are converted back to floats as follows:

bit 4 = 0 ⇒ sign = +1

bit 4 = 1 ⇒ sign = -1

Power = sign * Power′ + 25

$L = \text{Mantissa} * 10^{\text{Power}}$

These calculated luminosity measurements are sent to the Tevatron via ACNET for online monitoring. These measurements are also sent back to the CLC readout crate into a memory module, and are read out by the CDF data acquisition system and stored into the CLLD bank, and can be accessed by online consumers such as XMON and LUMMON. The bank structure for the CLLD bank is specified in table 95. The two 32-bit words for time stamp are used to store the time when the luminosity measurements were calculated by the cpu.

"CLLD"
Bank Number
Bank Version
Bank Length
Bank Type (I*4)
Number of Blocks (=1)
Pointer to Block 0
Pointer to End of Data
L_{inst}^{GLIVE} (bunch 0)
L_{inst}^{GLIVE} (bunch 1)
...
L_{inst}^{GLIVE} (bunch 34)
L_{inst}^{GLIVE} (bunch 35)
L_{total}^{GLIVE} (bunch 0)
L_{total}^{GLIVE} (bunch 1)
...
L_{total}^{GLIVE} (bunch 34)
L_{total}^{GLIVE} (bunch 35)
L_{inst} (bunch 0)
L_{inst} (bunch 1)
...
L_{inst} (bunch 34)
L_{inst} (bunch 35)
L_{total} (bunch 0)
L_{total} (bunch 1)
...
L_{total} (bunch 34)
L_{total} (bunch 35)
$< L_{inst}^{GLIVE} >$
$< L_{total}^{GLIVE} >$
$< L_{inst} >$
$< L_{total} >$
$< L_{inst}^{scalar} >$
$< L_{total}^{scalar} >$
Time Stamp 0
Time Stamp 1

Table 95: *Structure of the CLLD bank.*

27 Forward Detectors

Forward detectors include the **Roman Pot Spectrometer** (RPS), the **Beam Shower Counters** (BSC), and the **MiniPlug Calorimeters** (MP). Table 96 shows the number of readout channels for each subdetector and the name of corresponding bank. Figure 13 shows the location of the forward detectors along $\overline{p}$ direction on the west side of the CDF central detector.

Subdetector	Bank name	Number of channels		
		East	West	Total
BSC-1	FDLD	4	4	8
BSC-1	FDAD	4	4	8
BSC-2	FDAD	2	2	4
BSC-3	FDAD	2	2	4
BSC-4	FDAD	-	2	2
RPS	RPAD	-	80+80+80	240
RPS triggers	FDAD	-	3	3
MP	MPAD	84	84	168
MP triggers	MPAD	18	18	36

Table 96: *The number of readout channels for each subdetector and the name of corresponding bank.*

27.1 Roman Pot Spectrometer (RPS)

The **Roman Pot Spectrometer** will be used to detect the tracks from the antiprotons traversing the detector. The RPS consists of three Roman Pot stations located at about 57 meters to the west from the Interaction Point (IP), approximately one meter apart from each other. The detectors used in the pots comprise 240 scintillator fiber channels and three scintillator counters for triggering. Each pot has 80 fiber channels divided into 4 planes: 2 for X readout and 2 for Y readout (the X readout fibers are vertical and measure X, the Y readout fibers are horizontal and measure Y). The data will be stored in the RPAD bank.

The signals from PMT of the three RPS trigger counters will be used for triggering. These measurements will be stored in the FDAD bank.

27.2 Beam Shower Counters (BSC)

The **Beam Shower Counters** detect particles traveling in either direction from the IP along and near the beam-pipe ($5.5 < |\eta| < 7.5$) and will be used as a rapidity gap tagger for single diffractive and double-pomeron exchange processes. The BSC can be also used as luminosity and beam loss monitors as well.

There are four BSC stations on the West side and three on the East side of the IP. All stations are located along the beam-pipe, at increasing distances from the IP as one goes from BSC-1 to BSC-4 (see Figure 13). BSC-1, 2 and 3 consist of two stations each, positioned symmetrically with respect to the IP, whereas BSC-4 consists of a single station on the West side. Stations are made up of two scintillation counters, except for the BSC-1

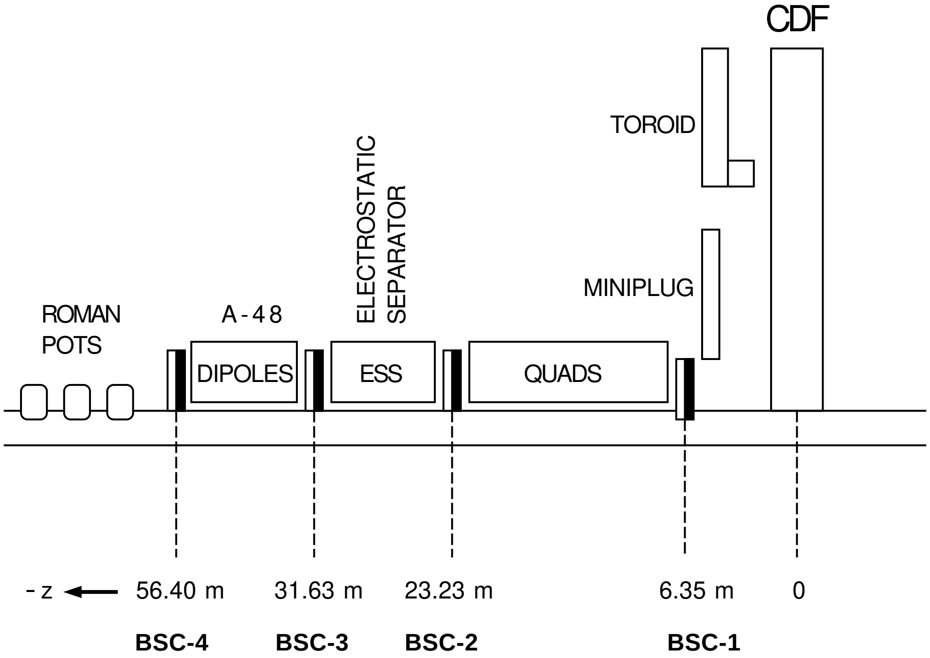

Figure 13: Location of the forward detectors along the $\bar{p}$ direction on the west side of the CDF central detector (not to scale). This is a view from the bottom looking up at the southwest quadrant of the detector. The Roman Pot Spectrometers lie to the south of the beam. The Beam Shower Counters and MiniPlug circle all the way around the beam. The east side is similar, except for two things. On the east side, only the first three BSC stations will be installed, as there is no room for BSC-4. There are no Roman Pots on the east side.

stations, which have four counters. Since each counter is connected to its own PMT, the entire system consists of 18 signal channels, 10 from the West and 8 from the East. Signals from BSC will be stored in the FDAD bank. Also, signals from BSC-1 for beam loss and collision measurements will be stored in the FDLD bank.

27.2.1 Beam Shower Counters for Beam Loss and Collision Monitoring

In the "Collider Run", the luminosity will be measured using the CLC. The BSC-1 counters will be used to monitor Tevatron beam losses and measure in-time east-west coincidence rates. We will store this information into the CDF DAQ system as "in-time" and "out-of-time" crossing rates in the counters. In the following, we will refer to these rates as "Collisions" and "Losses", respectively.

The signals from BSC-1 go to a transition board of one ADMEM board which provides one analog and two discriminated outputs. The analog signal goes to the ADMEM board, where it is digitized and it is used for diffractive physics. The discriminated outputs are directed through a second transition board to a Tevatron-dedicated 6U VME crate for "Losses" and "Collisions" measurements. For timing purposes, the earlier signal due to beam losses is delayed by about 40 nsec. A Dual Port Memory receives the information serially and sends it to the crate controller which determines the following quantities on beam loss and collision:

$\mathrm{BC_{inst}}$	Bunch by bunch instantaneous beam collisions,
$\mathrm{BL_{inst}^{p(\bar{p})}}$	Bunch by bunch instantaneous beam (anti)proton losses,
$\langle \mathrm{BC_{inst}} \rangle$	Average instantaneous beam collision,
$\langle \mathrm{BC_{total}} \rangle$	Average integrated beam collisions
$\langle \mathrm{BL_{inst}^{p(\bar{p})}} \rangle$	Average instantaneous beam (anti)proton losses,
$\langle \mathrm{BL_{total}^{p(\bar{p})}} \rangle$	Average integrated beam (anti)proton losses.

For the bunch by bunch beam losses $\mathrm{BL_{inst}^{p(\bar{p})}}$, the loss data for each of 4 counters will be stored in the bank, as well as the loss for all 4 counters.

All this information is then sent back to a second Dual Port Memory in the 9U crate, and read out by the CDF DAQ system. All these quantities are 16 bit words, and they will be stored, according to the bunch number, in the upper or lower 16 bits of the 32 bit data words in the FDLD bank.

27.3 MiniPlug Calorimeters

MiniPlug (MP) calorimeters are designed to measure charged and neutral hadron transverse energy as well as pseudorapidity covering rapidity region $3.5 < |\eta| < 5.5$.

The two MP detectors will be mounted on each side of the IP within the holes of the toroids. MP calorimeter consists of alternating layers of lead plates and liquid scintillator read out by WaveLength–Shifting fibers. The fibers are then read out by Multi–Channel Photomultiplier Tubes (MCPMTs). Each MCPMT has 17 outputs: 16 anode signals (not all of them used) and one "sum output" signal which comes from the last dynode. We use 18 MCPMTs on each MP detector. Each MP detector has a total of 252 channels and 18 channels from the sum output signals. We combine the anode signals from three channels of each PMT into a single electronic channel (or "tower"). After grouping the channels together, each MP has 84 anode signal outputs. The total number of output channels is 2 x 84 + 2 x 18. Signals from MP will be stored in the MPAD bank.

27.4 FDLD - BSC Beam Loss and Collision Bank

The bank structure of the "FDLD" bank is specified in Table 97. There is one block in the bank and the block consists of 32 bit data words. The data words contain two 16 bit fields. For the recording of the loss and collision data relative to individual bunches, the upper and lower 16 bits are used for bunches with even and odd IDs, respectively. For the loss data for each of 4 counters, the upper and lower 16 bits are used for counters 1 or 3 and 2 or 4, respectively. The last two 32 bit words are used to store the time and date when the beam collision and loss measurements are calculated by the CPU, respectively.

The 16 bit fields used to store all the data in this bank (except the time stamp) are not simply integers. They contain three smaller bitfields: base is in bits 6 to 15, bit 5 is a signbit, and bits 0 to 4 are powerbits. These must be decoded using the following formulas:

$$\text{power} = -1^{\text{signbit}} * \text{powerbits} + 21$$

$$\text{value} = \text{base} * 10^{\text{power}}$$

The format described in this note was first used in run 110026. Before that run the format of this bank was not stable.

27.5 FDAD - RPS Trigger Counters and BSC Bank

The bank structure of the "FDAD" bank is specified in Table 98. The block consists of repeated instances of a 16 bit channel ID followed by a 16 bit data word. The channels are numbered from 0 to 20; Table 99 shows the correspondence of the channel ID to the subdetector channels. For each subdetector, channels are counted in this sequence: counter inside the ring, counter outside the ring for BSC-2, BSC-3, BSC-4; upper counter inside the ring, lower counter inside the ring, upper counter outside the ring, lower counter outside the ring for BSC-1.

"FDLD"	
Bank Number	
Bank Version	
Bank Length	
Bank Type (I*4)	
Number of Blocks (=1)	
Pointer to Block 0	
Pointer to End of Data	
BC_{inst} (bunch 0)	BC_{inst} (bunch 1)
BC_{inst} (bunch 2)	BC_{inst} (bunch 3)
...	
BC_{inst} (bunch 34)	BC_{inst} (bunch 35)
BL^{p}_{inst} (bunch 0)	BL^{p}_{inst} (bunch 1)
BL^{p}_{inst} (bunch 2)	BL^{p}_{inst} (bunch 3)
...	
BL^{p}_{inst} (bunch 34)	BL^{p}_{inst} (bunch 35)
$BL^{\bar{p}}_{inst}$ (bunch 0)	$BL^{\bar{p}}_{inst}$ (bunch 1)
$BL^{\bar{p}}_{inst}$ (bunch 2)	$BL^{\bar{p}}_{inst}$ (bunch 3)
...	
$BL^{\bar{p}}_{inst}$ (bunch 34)	$BL^{\bar{p}}_{inst}$ (bunch 35)
BL^{p}_{inst} (bunch 0, counter 1) BL^{p}_{inst} (bunch 0, counter 3)	BL^{p}_{inst} (bunch 0, counter 2) BL^{p}_{inst} (bunch 0, counter 4)
BL^{p}_{inst} (bunch 1, counter 1) BL^{p}_{inst} (bunch 1, counter 3)	BL^{p}_{inst} (bunch 1, counter 2) BL^{p}_{inst} (bunch 1, counter 4)
...	
BL^{p}_{inst} (bunch 35, counter 1) BL^{p}_{inst} (bunch 35, counter 3)	BL^{p}_{inst} (bunch 35, counter 2) BL^{p}_{inst} (bunch 35, counter 4)
$BL^{\bar{p}}_{inst}$ (bunch 0, counter 1) $BL^{\bar{p}}_{inst}$ (bunch 0, counter 3)	$BL^{\bar{p}}_{inst}$ (bunch 0, counter 2) $BL^{\bar{p}}_{inst}$ (bunch 0, counter 4)
$BL^{\bar{p}}_{inst}$ (bunch 1, counter 1) $BL^{\bar{p}}_{inst}$ (bunch 1, counter 3)	$BL^{\bar{p}}_{inst}$ (bunch 1, counter 2) $BL^{\bar{p}}_{inst}$ (bunch 1, counter 4)
...	
$BL^{\bar{p}}_{inst}$ (bunch 35, counter 1) $BL^{\bar{p}}_{inst}$ (bunch 35, counter 3)	$BL^{\bar{p}}_{inst}$ (bunch 35, counter 2) $BL^{\bar{p}}_{inst}$ (bunch 35, counter 4)
$\langle BC_{inst} \rangle$	$\langle BC_{total} \rangle$
$\langle BL^{p}_{inst} \rangle$	$\langle BL^{p}_{total} \rangle$
$\langle BL^{\bar{p}}_{inst} \rangle$	$\langle BL^{\bar{p}}_{total} \rangle$
Time Stamp	
Time Stamp	

Table 97: *Structure of the FDLD bank. There is only one block. Each line represents one of the 32 bit words that form the bank. Where a line is divided into two columns, the left (right) column gives the contents of bits 16:31 (0:15). This is a fixed size bank.*

"FDAD"	
Bank Number	
Bank Version	
Bank Length	
Bank Type (I*2)	
Number of Blocks (=1)	Pointer to Block 0
Pointer to End of Data	Padding
Channel ID	Data Word 0
...	
Channel ID	Last Data Word

Table 98: *Structure of the FDAD bank. There is only one block. The order channels are written into the bank is not specified. Banks may be written without data from all 21 channels. The bank data (everything past the bank type word) is composed of 16 bit words. Two are shown on each line.*

Subdetector	Channel IDs
BSC-1 East	0,1,2,3
BSC-2 East	4,5
BSC-3 East	6,7
BSC-1 West	8,9,10,11
BSC-2 West	12,13
BSC-3 West	14,15
BSC-4 West	16,17
RP	18,19,20

Table 99: *The correspondence of the channel ID to the subdetector channels in the FDAD bank. For each subdetector, channel IDs are in this sequence: counter inside the ring, counter outside the ring for BSC-2, BSC-3, BSC-4; upper counter inside the ring, lower counter inside the ring, upper counter outside the ring, lower counter outside the ring for BSC-1. The RP channel IDs are in order of distance from the Interaction Point.*

27.6 MPAD - MiniPlug Calorimeter Bank

The bank structure of the "MPAD" bank is specified in Table 100. The bit fields in the channel ID are defined in Table 101.

Each MiniPlug contains 84 anode signal outputs or "towers", which are composed of three adjacent channels of each PMT. Figure 14 shows the numbering of the towers in X - Y plane of each MiniPlug, viewed from east to west. The tower number in this figure forms the lower 7 bits of the channel ID for the towers.

"Sum channels" are formed by summing the output of several individual towers. Each sum channel covers a sector of 60 degrees in ϕ. Each sector is radially divided in three concentric rings: inner, middle and outer. The number of sum channels is shown in Figure 15 and it is written into the lower 7 bits of the channel ID. There are 5 individual towers in each of the inner (0 through 5) or outer (12 through 17) sum channels; there are only 4 individual towers in each of the middle (6 through 11) sum channels. For example, sum channel 0 is the sum for towers 0 to 4, sum channel 6 is the sum for towers 5 to 8, and sum channel 12 is the sum for towers 9 to 13.

"MPAD"	
Bank Number	
Bank Version	
Bank Length	
Bank Type (I*2)	
Number of Blocks (=2)	Pointer to West MP Block
Pointer to East MP Block	Pointer to End of Data
Channel ID	West MP Block Data Word 0
...	
Channel ID	West MP Block Last Data Word
Channel ID	East MP Block Data Word 0
...	
Channel ID	East MP Block Last Data Word

Table 100: *Structure of the MPAD bank. There are two blocks and the order of blocks is shown. All channels will not always be present in the bank. The bank data (everything past the bank type word) is composed of 16 bit words. Two are shown on each line.*

Bit	Definition
13:15	Must be 0
9:12	Unused
8	Side: West = 0; East = 1
7	Type: Sum = 0; Tower = 1
6:0	Sum (0:17) or Tower (0:83)

Table 101: *Bit assignment for the MPAD channel ID.*

The order of channels written in the bank is specified in Table 102. The data is organized such that the sum output information from the west MiniPlug comes first followed by the tower information, which are followed by the east Miniplug.

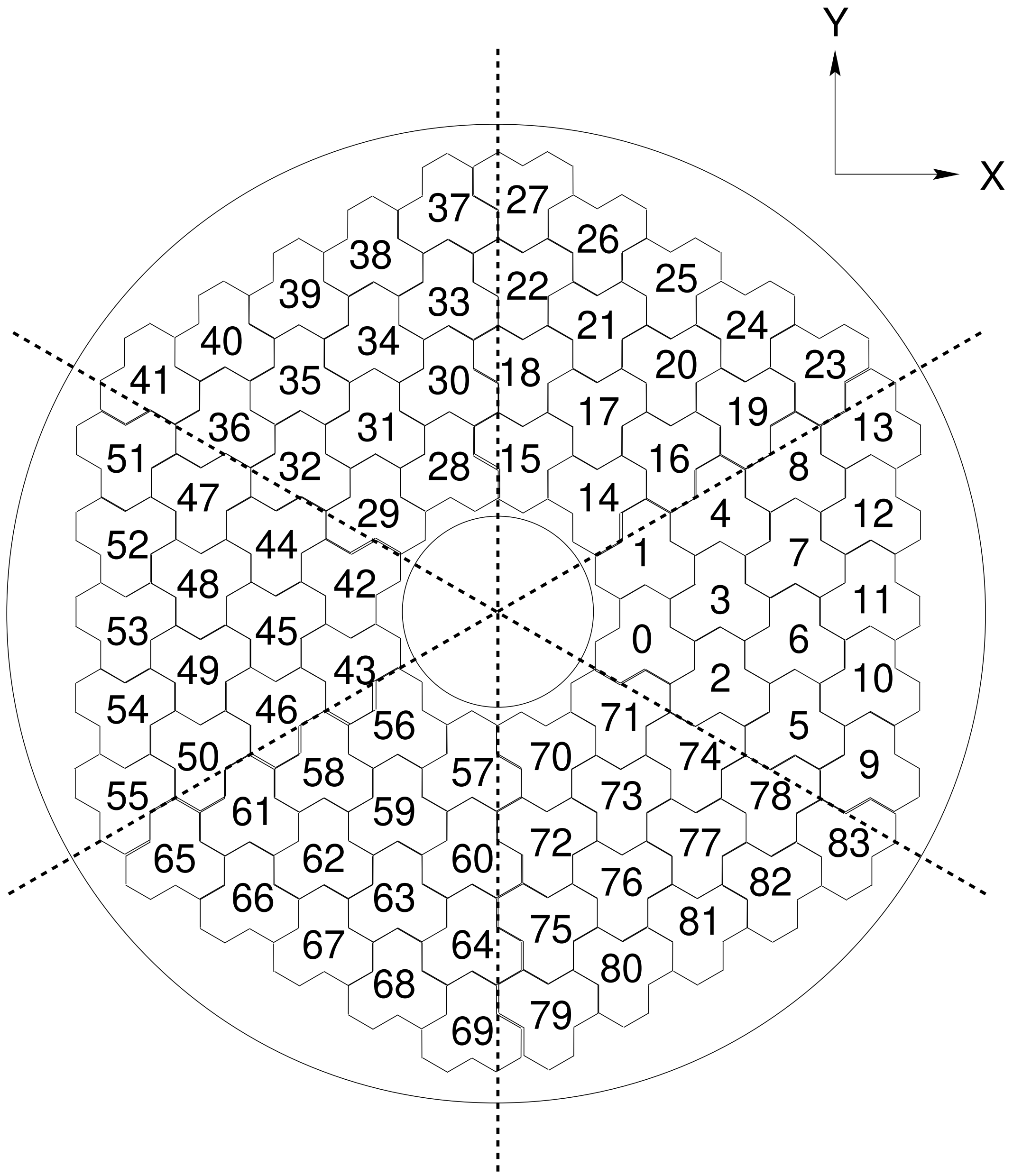

Figure 14: Numbering of the towers for each Miniplug as viewed from the east towards the west. Three adjacent hexagons form a "tower" and the corresponding PMT channels are combined into a single channel. The tower number in the figure is written in the lower 7 bits of the channel ID.

West MP, Sum 0
...
West MP, Sum 17
West MP, Tower 0
...
West MP, Tower 83
East MP, Sum 0
...
East MP, Sum 17
East MP, Tower 0
...
East MP, Tower 83

Table 102: *Order of channels written in the MPAD bank.*

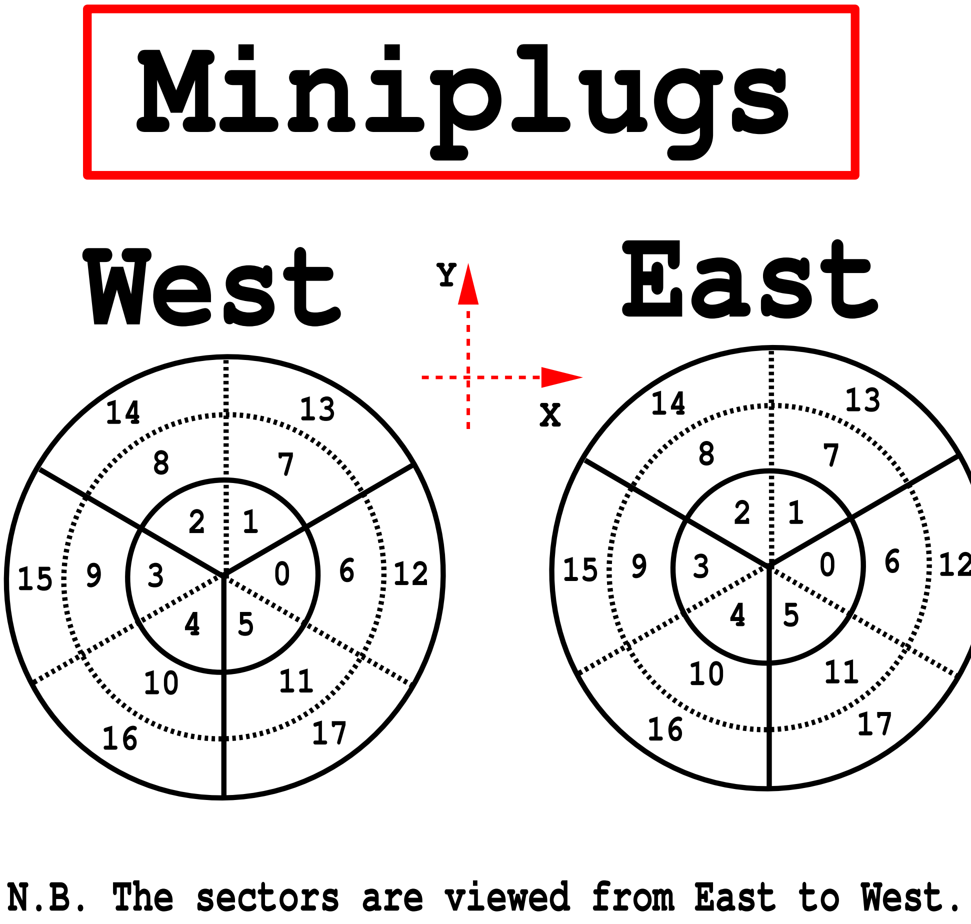

Figure 15: Numbering of the MCPMTs which read out the two MiniPlugs (East and West). This is also the numbering of the sum channels that is written into the lower seven bits of the channel ID. Each MiniPlug is viewed from east to west.

27.7 RPAD - Roman Pot Spectrometer Bank

The bank structure of the "RPAD" bank is specified in Table 103. The block consists of repeated instances of a 16 bit channel ID followed by a 16 bit data word. The bit fields in the channel ID are specified in Table 104. The three numbers in the channel ID are defined as follows. The Roman Pots are numbered from 1 to 3 in order of increasing distance from the Interaction Point (IP). The X readout fibers (the vertical fibers) have the type bit set to 0. The Y readout fibers (the horizontal fibers) have their type bit set to 1. In each type of readout within each Roman Pot, the fibers are numbered from 0 to 39 and are grouped in two planes. The plane containing X (Y) fibers with even (odd) numbers is closest to the IP; the other plane contains odd (even) fibers. The X fibers are numbered from 0 to 39 in order of increasing distance from the beam; the Y fibers are numbered from 0 to 39 from bottom to top. This scheme is illustrated in Figure 16.

<table>
<tr><td colspan="2">"RPAD"</td></tr>
<tr><td colspan="2">Bank Number</td></tr>
<tr><td colspan="2">Bank Version</td></tr>
<tr><td colspan="2">Bank Length</td></tr>
<tr><td colspan="2">Bank Type (I*2)</td></tr>
<tr><td>Number of Blocks (=1)</td><td>Pointer to Block 0</td></tr>
<tr><td>Pointer to End of Data</td><td>Padding</td></tr>
<tr><td>Channel ID</td><td>Data Word 0</td></tr>
<tr><td colspan="2">...</td></tr>
<tr><td>Channel ID</td><td>Last Data Word</td></tr>
</table>

Table 103: *Structure of the RPAD bank. There is only one block. All channels may not always be present. The bank data (everything past the bank type word) is composed of 16 bit words. Two are shown on each line.*

Bit	Definition
13:15	Must be 0
9:12	Unused
8:7	Roman Pot: (1:3)
6	Type: X readout = 0; Y readout = 1
5:0	Fiber (0:39)

Table 104: *Bit assignment for the RPAD channel ID.*

The order of channels written in the bank is specified in Table 105. The data is organized such that the X fiber information from the Roman Pot 1 comes first followed by the Y fiber information, which are followed by the data from the Pot 2, then the Pot 3.

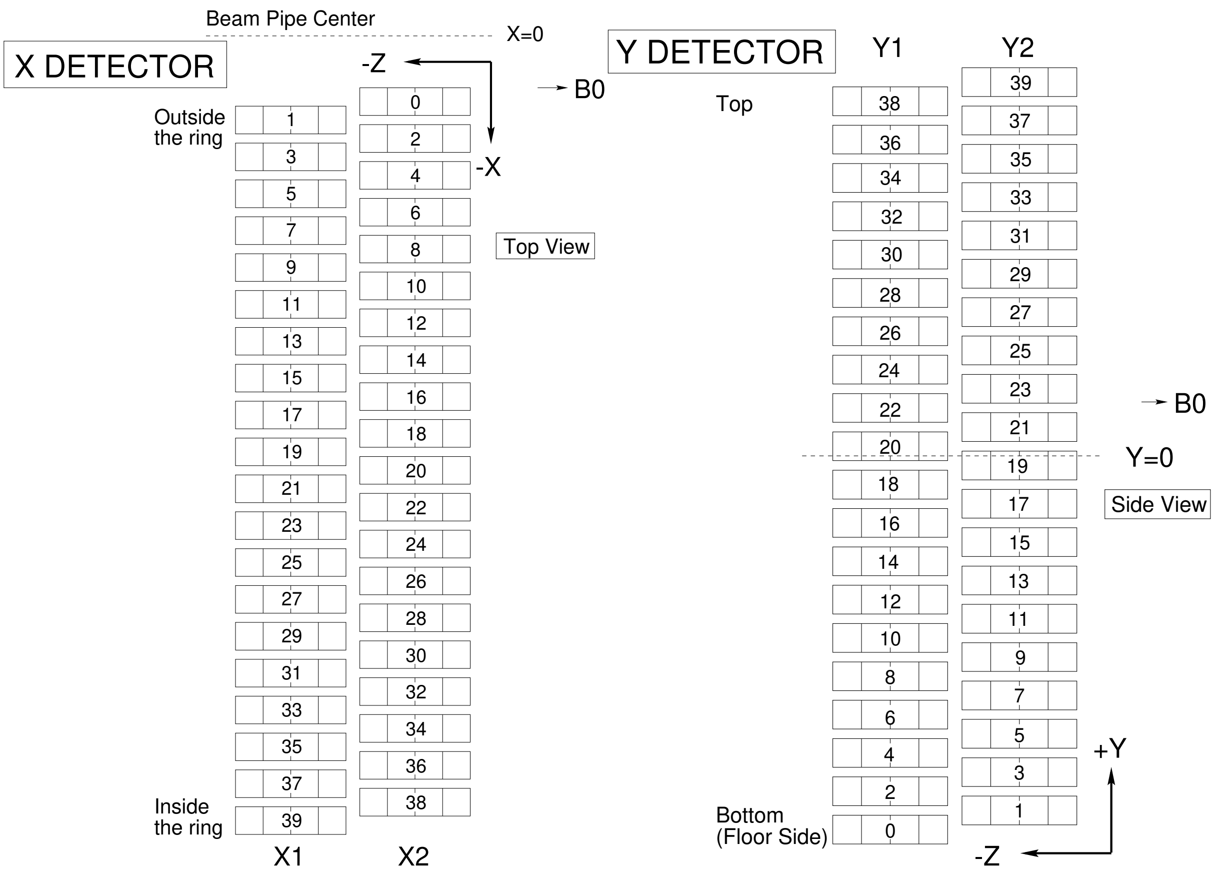

Figure 16: Numbering of the fibers for X (left) and Y (right) readouts. The X fibers are viewed from top to bottom; the Y fibers are viewed from side. The interaction point is on the right side for both X and Y fibers, so the track from the antiproton in diffractive events passes through the fibers from right to left.

Pot 1, X fiber 0
...
Pot 1, X fiber 39
Pot 1, Y fiber 0
...
Pot 1, Y fiber 39
...
Pot 3, X fiber 0
...
Pot 3, X fiber 39
Pot 3, Y fiber 0
...
Pot 3, Y fiber 39

Table 105: *Order of channels written in the RPAD bank.*

28 BFLD - Magnetic Field Bank

The BFLD bank provides information on the magnetic field. The format of the bank is specified in Table 106 The field is controlled with a precision NMR probe and a precision DC current transformer. For each of these there is a second device for cross calibration and to act as a hot spare. There is also a Hall probe for calibration and monitoring. These devices are read periodically by the magnet controls system which adjusts the current to keep the field stable. The controls system sends the information and a time stamp to the CDF DAQ system on a serial line. It is buffered in a Camac FIFO that is periodically polled by one of the VME CPUs. Because the NMR may not be locked or one of the redundant devices may be broken, there is an integer data quality word to indicate if the data are valid and if they are used for the feedback logic. The data quality word is described in Table 107.

"BFLD"
Bank Number
Bank Version
Bank Length
Bank Type (I*4)
Number of Blocks (= 0)
Pointer to Block 0
Pointer to End of Data
Data Quality (I*4)
Readout Time Stamp
NMR0 Field (Gauss)
NMR1 Field (Gauss)
Hall Probe Value (Gauss)
DCCT0 Current (Units of 0.1 Amps)
DCCT1 Current (Units of 0.1 Amps)

Table 106: *Structure of the BFLD bank.*

Bit	Field
0	NMR0 valid
1	NMR0 used for feedback
2	NMR1 valid
3	NMR1 used for feedback
4	Hall probe valid
5	Hall probe used for feedback
6	DCCT0 valid
7	DCCT0 used for feedback
8	DCCT1 valid
9	DCCT1 used for feedback
10-15	Unused
16-31	Bank version (=100)

Table 107: *Structure of the data quality word.*

29 Trigger Banks

A three level trigger system is used to collect events. Level 1 (L1) is a hardware trigger used to find physics objects using a subset of detector elements. The local detector triggers are sent to a global L1 trigger [7] which makes a decision based on the local triggers. Limited event reconstruction is performed at Level 2 (L2). The local trigger information is passed to the global L2 trigger which makes the decision on whether to accept or reject the event. An overview of the Run II trigger system is given in Figure 17.

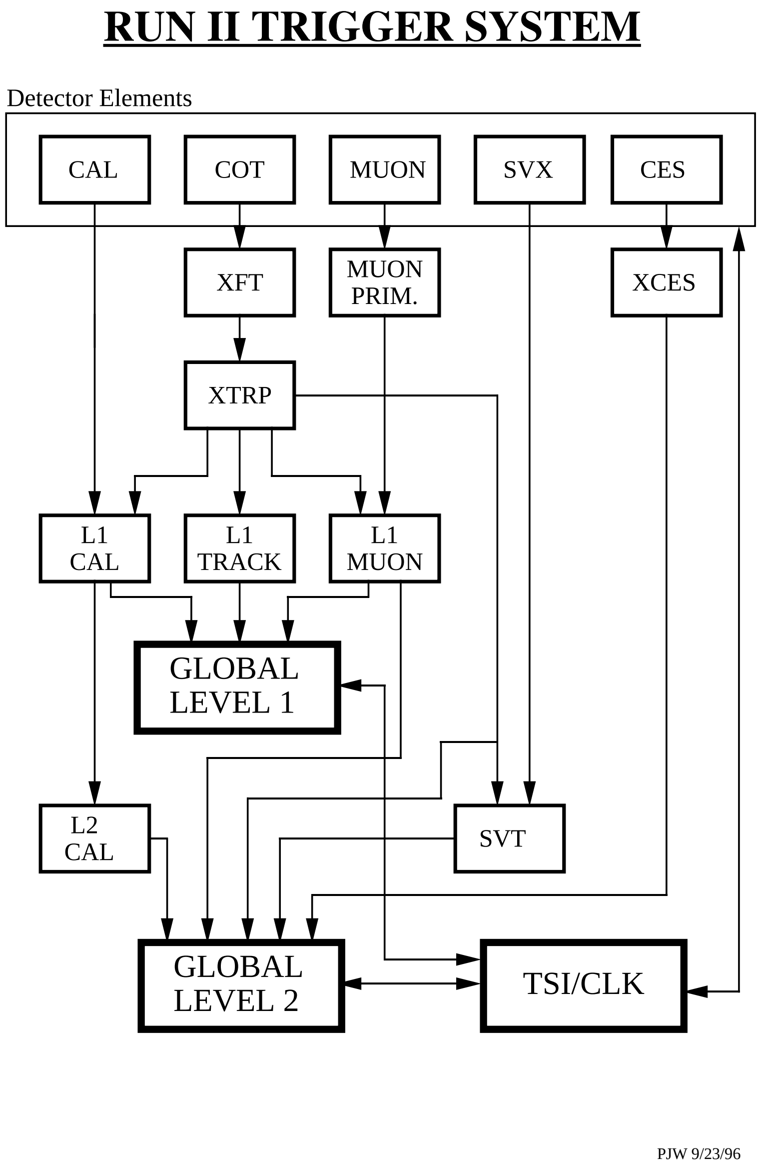

Figure 17: *Overview of the Run II trigger system.*

The L1 Calorimeter (L1CAL) trigger selects events based on energy deposits with signatures which appear to come from electrons, photons or jets. These object triggers are formed by applying thresholds to individual trigger towers. In addition global energy sums are performed such as the total event transverse energy and the missing transverse energy which is constructed from the sum of the trigger towers.

Although all banks could be read out whenever an event is accepted by Level 2, in practice, most of the information in the banks described in this section is for diagnostic purposes. The full trigger readout will be done on a prescaled basis using a special diagnostic readout list. Only the default banks listed below are expected to be read out in each Level 2 accept. This procedure will save time and space on tape.

- Default banks:
  `TCMD, TL1D, TFRD, SL1D, SVTD, TL2D`

- Diagnostic banks:
  `XFFD, XFLD, XTRD, TC1D, TC2D, T2PD`

The readout and reformatting of the trigger data makes use of the same front end code and thus we wish to use a consistant structure for the trigger banks. The reformatter code which assembles the minibanks into the final bank format is data driven and assembles the final bank based on information in the header section. The reformatting process will follow a set of rules driven by the contents of the header section of the minibank. We also need to allow the flexibility for the bank contents to change, for example change the number of cards that are read out or the number of data words for each card. The bank type that gives us this flexibility is for the data block to have offsets to the number of cards. This bank type is one of the basic types and is the same as the TDC type banks.

29.1 XFFD - XFT Finder Bank

The XFT Finder boards find track segments in the axial superlayers of the COT. In XFT nomenclature, the superlayers are numbered 1-4. The output segment locations are refered to as pixels. The finders report 12 pixels with no pointing information for the 2 inner layers and 6 pixels each for left- and right-pointing segments in the outer two layers. There are two types of finders: the 2/4 type finds segments in 12 cells of layer 2 and 20 cells of layer 4; the 1/3 type finds segments in 16 cells of layer 3 and 8 cells of layer 1. Layer 1 has redundant electronics to serve split wires if they are installed. The two copies are known as layer 1 and layer $1'$. The finders get their input from the COT TDC modules in the pit. A mezzanine card on the TDC makes a crude digitization and reports two bits per wire to the finder. Hits consistent with being near a sense wire are called “prompt”; those far from the wire are “delayed.” There are a total of 48 finders. There are 8 finders in each of 6 blocks. There are 48 integer words read from each finder board, yielding 384 words per block. The structure of the XFFD bank is given in Table 108, and the structure within each board is given in Table 109.

"XFFD"
Bank Number
Bank Version
Bank Length
Bank Type (I*4)
Number of Blocks (=6)
Pointer to Block 0: (Layer 1/3, $270° < \phi < 30°$)
Pointer to Block 1: (Layer 2/4, $270° < \phi < 30°$)
Pointer to Block 2: (Layer 1/3, $30° < \phi < 150°$)
Pointer to Block 3: (Layer 2/4, $30° < \phi < 150°$)
Pointer to Block 4: (Layer 1/3, $150° < \phi < 270°$)
Pointer to Block 5: (Layer 2/4, $150° < \phi < 270°$)
Pointer to End of Data
Block 0: Number of Cards (=8)
Pointer to Card 0
...
Pointer to Card 7
Pointer to End of Block
Finder Data
...
Finder Data
...
Block 5: Number of Cards (=8)
Pointer to Card 0
...
Pointer to Card 7
Pointer to End of Block
Finder Data
...
Finder Data

Table 108: *Structure of the XFFD bank. Nominally there are 8 cards per block, but it is possible for cards to be missing. Card order may vary. One should identify the cards in the bank using the slot ID and block position.*

Offset	Data Type	1/3 Board		2/4 Board	
		Layer	Cell	Layer	Cell
$P_m + Q_n + 0$	Header				
$P_m + Q_n + 1$	Segment	3	0, 1	4	0, 1
$P_m + Q_n + 2$	Segment	3	2, 3	4	2, 3
$P_m + Q_n + 3$	Segment	3	4, 5	4	4, 5
$P_m + Q_n + 4$	Segment	3	6, 7	4	6, 7
$P_m + Q_n + 5$	Segment	3	8, 9	4	8, 9
$P_m + Q_n + 6$	Segment	3	10,11	4	10,11
$P_m + Q_n + 7$	Segment	3	12,13	4	12,13
$P_m + Q_n + 8$	Segment	3	14,15	4	14,15
$P_m + Q_n + 9$	Segment	1	0, 1	4	16,17
$P_m + Q_n + 10$	Segment	1	2, 3	4	18,19
$P_m + Q_n + 11$	Segment	1	4, 5	2	0, 1
$P_m + Q_n + 12$	Segment	1	6, 7	2	2, 3
$P_m + Q_n + 13$	Segment	$1'$	0, 1	2	4, 5
$P_m + Q_n + 14$	Segment	$1'$	2, 3	2	6, 7
$P_m + Q_n + 15$	Segment	$1'$	4, 5	2	8, 9
$P_m + Q_n + 16$	Segment	$1'$	6, 7	2	10,11
$P_m + Q_n + 17$	Wire	3	0	4	0
$P_m + Q_n + 18$	Wire	3	1	4	1
$P_m + Q_n + 19$	Wire	3	2	4	2
$P_m + Q_n + 20$	Wire	3	3	4	3
$P_m + Q_n + 21$	Wire	3	4	4	4
$P_m + Q_n + 22$	Wire	3	5	4	5
$P_m + Q_n + 23$	Wire	3	6	4	6
$P_m + Q_n + 24$	Wire	3	7	4	7
$P_m + Q_n + 25$	Wire	3	8	4	8
$P_m + Q_n + 26$	Wire	3	9	4	9
$P_m + Q_n + 27$	Wire	3	10	4	10
$P_m + Q_n + 28$	Wire	3	11	4	11
$P_m + Q_n + 29$	Wire	3	12	4	12
$P_m + Q_n + 30$	Wire	3	13	4	13
$P_m + Q_n + 31$	Wire	3	14	4	14
$P_m + Q_n + 32$	Wire	3	15	4	15
$P_m + Q_n + 33$	Wire	1	0	4	16
$P_m + Q_n + 34$	Wire	1	1	4	17
$P_m + Q_n + 35$	Wire	1	2	4	18
$P_m + Q_n + 36$	Wire	1	3	4	19
$P_m + Q_n + 37$	Wire	1	4	2	0
$P_m + Q_n + 38$	Wire	1	5	2	1
$P_m + Q_n + 39$	Wire	1	6	2	2
$P_m + Q_n + 40$	Wire	1	7	2	3
$P_m + Q_n + 41$	Wire	$1'$	0	2	4
$P_m + Q_n + 42$	Wire	$1'$	1	2	5
$P_m + Q_n + 43$	Wire	$1'$	2	2	6
$P_m + Q_n + 44$	Wire	$1'$	3	2	7
$P_m + Q_n + 45$	Wire	$1'$	4	2	8
$P_m + Q_n + 46$	Wire	$1'$	5	2	9
$P_m + Q_n + 47$	Wire	$1'$	6	2	10
$P_m + Q_n + 48$	Wire	$1'$	7	2	11

Table 109: *Structure of data for a single finder board in the XFFD bank. P_m is the offset to block m. Q_n is the offset the board n within the block.*

Structure of the XFFD VME Header Word for the Finder Boards

Bit	Function
7:0	Bunch Counter
8:12	Geographical Address (Slot)
13:22	Module Serial Number
23:31	Module Type

There are two kinds of data in the XFFD readout, pixel data and wire data. The structure of pixel data word is:

Bit	Function
11:0	Cell 0 - Pixel Data(11:0)
12	Cell 0 - B0 Marker
13	Cell 0 - W0 Marker
14	Cell 0 - Error Marker
15	Undefined
27:16	Cell 1 - Pixel Data(11:0)
28	Cell 1 - B0 Marker
29	Cell 1 - W0 Marker
30	Cell 1 - Error Marker
31	Undefined

And the structure of a wire data word is:

Bit	Function
11:0	Wire Data Prompt (11:0)
23:12	Wire Data Delay (11:0)
24	B0 Marker
25	W0 Marker
26	Error Marker
31:37	Undefined

The slot mapping of the 48 Finder Boards in XFFD:

Block 0			Block 1	
Crate	Slot	COT Cell Range	Slot	COT Cell Range
0	4	SL2: 140-147; SL6: 280-295	13	SL4: 212-223; SL8: 356-375
0	5	SL2: 148-155; SL6: 296-311	14	SL4: 224-235; SL8: 376-395
0	6	SL2: 156-163; SL6: 312-327	15	SL4: 236-247; SL8: 396-415
0	7	SL2: 164-171; SL6: 328-343	16	SL4: 248-259; SL8: 416-435
0	8	SL2: 172-179; SL6: 344-359	17	SL4: 260-271; SL8: 436-455
0	9	SL2: 180-187; SL6: 360-375	18	SL4: 272-283; SL8: 456-475
0	10	SL2: 188-003; SL6: 376-007	19	SL4: 284-007; SL8: 476-015
0	11	SL2: 004-011; SL6: 008-023	20	SL4: 008-019; SL8: 016-035
Block 2			Block 3	
Crate	Slot	COT Cell Range	Slot	COT Cell Range
1	4	SL2: 012-019; SL6: 024-039	13	SL4: 020-031; SL8: 036-055
1	5	SL2: 020-027; SL6: 040-055	14	SL4: 032-045; SL8: 056-075
1	6	SL2: 028-035; SL6: 056-071	15	SL4: 044-055; SL8: 076-095
1	7	SL2: 036-043; SL6: 072-087	16	SL4: 056-067; SL8: 096-115
1	8	SL2: 044-051; SL6: 088-103	17	SL4: 068-079; SL8: 116-135
1	9	SL2: 052-059; SL6: 104-119	18	SL4: 080-091; SL8: 136-155
1	10	SL2: 060-067; SL6: 120-135	19	SL4: 092-103; SL8: 156-175
1	11	SL2: 068-075; SL6: 136-151	20	SL4: 104-115; SL8: 176-195
Block 4			Block 5	
Crate	Slot	COT Cell Range	Slot	COT Cell Range
0	4	SL2: 076-083; SL6: 152-167	13	SL4: 116-127; SL8: 196-215
0	5	SL2: 084-091; SL6: 168-183	14	SL4: 128-139; SL8: 216-235
0	6	SL2: 092-099; SL6: 184-199	15	SL4: 140-151; SL8: 236-255
0	7	SL2: 100-107; SL6: 200-215	16	SL4: 152-163; SL8: 256-275
0	8	SL2: 108-115; SL6: 216-231	17	SL4: 164-175; SL8: 276-295
0	9	SL2: 116-123; SL6: 232-247	18	SL4: 176-187; SL8: 296-315
0	10	SL2: 124-131; SL6: 248-263	19	SL4: 188-199; SL8: 316-335
0	11	SL2: 132-139; SL6: 264-279	20	SL4: 200-211; SL8: 336-355

29.2 XFLD - XFT Linker Bank

Data from two SL1/3 finder boards and two SL2/4 are sent to a given linker board which forms tracks from the list of pixels. The data are divided into 3 blocks corresponding to the 3 VME crates of linker cards, 8 per crate. The structure of the XFLD bank is given in Table 110, and the format of the data for each linker board is given in Table 111.

Each linker board reports the pixel data input to each of the input formatter chips on the board. Each finder chip reports pixels for 4 cells, and 5 of the 6 input formatters receive data from 4 finder chips. The format of the pixel data word is:

Bit	Function
11:0	Finder 0, cell i
23:12	Finder 1, cell i

Each linker board contains 12 linker chips which report the best track (if any) in each 1.25° section of azimuth. Each linker board on readout reports the track found in each linker. The format of the first 4 Linker output words is:

"XFLD"
Bank Number
Bank Version
Bank Length
Bank Type (I*4)
Number of Blocks (=3)
Pointer to Block 0 – $270° < \phi < 30°$
Pointer to Block 1 – $30° < \phi < 150°$
Pointer to Block 2 – $150° < \phi < 270°$
Pointer to End of Data
Block 0: Number of Cards (=8)
Pointer to Card 0
...
Pointer to Card 7
Pointer to End of Block
Linker Data
...
Block 1: Number of Cards (=8)
Pointer to Card 0
...
Pointer to Card 7
Pointer to End of Block
Linker Data
...
Block 2: Number of Cards (=8)
Pointer to Card 0
...
Pointer to Card 7
Pointer to End of Block
Linker Data
...

Table 110: *Structure of the XFLD bank. Nominally there are 8 cards per block, but it is possible for cards to be missing. Card order may vary. One should identify the cards in the bank using the slot ID and block position.*

Offset	Description
$P_m + Q_n + 0$	Input Formatter 0 pixel info: Finder 0&1, cell 0
$P_m + Q_n + 1$	Input Formatter 0 pixel info: Finder 2&3, cell 0
$P_m + Q_n + 2$	Input Formatter 0 pixel info: Finder 0&1, cell 1
$P_m + Q_n + 3$	Input Formatter 0 pixel info: Finder 2&3, cell 1
$P_m + Q_n + 4$	Input Formatter 0 pixel info: Finder 0&1, cell 2
$P_m + Q_n + 5$	Input Formatter 0 pixel info: Finder 2&3, cell 2
$P_m + Q_n + 6$	Input Formatter 0 pixel info: Finder 0&1, cell 3
$P_m + Q_n + 7$	Input Formatter 0 pixel info: Finder 2&3, cell 3
$P_m + Q_n + 8$	Input Formatter 1 pixel info: Finder 0&1, cell 0
. . .	
$P_m + Q_n + 40$	Input Formatter 5 pixel info Finder 0&1, cell 0 pixels
$P_m + Q_n + 41$	Input Formatter 5 Error Word
$P_m + Q_n + 42$	Input Formatter 5 pixel info Finder 0&1, cell 1 pixels
$P_m + Q_n + 43$	Input Formatter 5 Error Word
$P_m + Q_n + 44$	Input Formatter 5 pixel info Finder 0&1, cell 2 pixels
$P_m + Q_n + 45$	Input Formatter 5 Error Word
$P_m + Q_n + 46$	Input Formatter 5 pixel info Finder 0&1, cell 3 pixels
$P_m + Q_n + 47$	Input Formatter 5 Error Word
$P_m + Q_n + 48$	Linker Output Word 0
$P_m + Q_n + 49$	Linker Output Word 1
$P_m + Q_n + 50$	Linker Output Word 2
$P_m + Q_n + 51$	Linker Output Word 3
$P_m + Q_n + 52$	Linker Output Word 4
$P_m + Q_n + 53$	Linker Output Word 5
$P_m + Q_n + 54$	Linker Output Word 6
$P_m + Q_n + 55$	Linker Output Word 7
$P_m + Q_n + 56$	VME Header Word 0
$P_m + Q_n + 57$	VME Header Word 1
$P_m + Q_n + 58$	VME Header Word 2
$P_m + Q_n + 59$	VME Header Word 3

Table 111: *Structure of data for a single linker board in the XFLD bank. P_m is the offset to block m. Q_n is the offset the board n within the block.*

Word	Bit	Function
0	1:0	Word Marker
0	4:2	Phi (Linker 0)
0	5	Short Track (Linker 0)
0	12:6	Pt Bin (Linker 0)
0	15:13	Phi (Linker 1)
0	16	Short Track (Linker 1)
0	23:17	Pt Bin (Linker 1)
1	1:0	Word Marker
1	4:2	Phi (Linker 4)
1	5	Short Track (Linker 4)
1	12:6	Pt Bin (Linker 4)
1	15:13	Phi (Linker 5)
1	16	Short Track (Linker 5)
1	23:17	Pt Bin (Linker 5)
2	1:0	Word Marker
2	4:2	Phi (Linker 2)
2	5	Short Track (Linker 2)
2	12:6	Pt Bin (Linker 2)
2	15:13	Phi (Linker 3)
2	16	Short Track (Linker 3)
2	23:17	Pt Bin (Linker 3)
3	1:0	Word Marker
3	7:2	Duplicate Trk Rem (Linker 0-5)
3	8	Beam Zero Marker
3	9	Linker Error Flag
3	10	Isolation (Linker 0)
3	11	Isolation (Linker 1)
3	12	Isolation (Linker 2)
3	13	Isolation (Linker 3)
3	14	Isolation (Linker 4)
3	15	Isolation (Linker 5)
3	23:16	Beam Crossing Counter

The Linker Output Words 4-7 are identical in structure except that they contain the information for Linker chips 6-11. The Pt word has a range of 0-95 with Pt Bin 0 being low pt negatively charged track and 95 being a low Pt positively charged track. Pt bins 47 and 48 represent high Pt ("infinite") for negatively and positively charged tracks, respectively. A Pt bin value of 124 (0x7C) is an invalid Pt and indicates no track is present in the linker chip. It is possible for two linker chips to find the same track in adjacent pixels. Duplicate tracks are removed on the linker board when the tracks are found in linker chips 0-5 or 6-11 on the same board. Duplicate tracks found on neighboring boards and between linker chips 5 and 6 are removed by the XTRP. The duplicate track flag is set on with a duplicate track has been removed.

The sixth input formatter receives data from two finder chips, but also takes care of the error handling on the board. The structure of the XFLD error word is:

Bit	Function
6:0	Error count for board
7	Error present (This event)
8	Error on cable 0 from Start bit
9	Error on cable 0 from word 0 bit
10	Error on cable 0 from Beam0 bit
11	Error on cable 0 from Finder Error Bit
12	Error on cable 1 from Start bit
13	Error on cable 1 from word 0 bit
14	Error on cable 1 from Beam0 bit
15	Error on cable 1 from Finder Error Bit
16	Error on cable 2 from Start bit
17	Error on cable 2 from word 0 bit
18	Error on cable 2 from Beam0 bit
19	Error on cable 2 from Finder Error Bit
20	Error on cable 3 from Start bit
21	Error on cable 3 from word 0 bit
22	Error on cable 3 from Beam0 bit
23	Error on cable 3 from Finder Error Bit

Four words are also read out on the VME interface chip on the Linker board. The structure of that words is as follows.

Bit	Function
7:0	Beam Crossing Counter
12:8	Geographical Address (Slot)
13	Beam 0 Marker
19:14	Brd Serial Number
31:20	Brd Type
31:0	VME and Linker Versions
31:0	Input and Output Version
0	Current Error Flag
1	Error Flag for L2 Event
23:2	Spare
31:24	Error Counter

Here is the slot mapping of the 24 Linker Boards in XFLD:

Linker Board	Block	Slot	Phi Range
0	0	4	270-285 deg
1	0	6	285-300 deg
2	0	8	300-315 deg
3	0	10	315-330 deg
4	0	12	330-345 deg
5	0	14	345-360 deg
6	0	16	0-15 deg
7	0	18	15-30 deg
8	1	4	30-45 deg
9	1	6	45-60 deg
10	1	8	60-75 deg
11	1	10	75-90 deg
12	1	12	90-105 deg
13	1	14	105-120 deg
14	1	16	120-135 deg
15	1	18	135-150 deg
16	2	4	150-165 deg
17	2	6	165-180 deg
18	2	8	180-195 deg
19	2	10	195-210 deg
20	2	12	210-225 deg
21	2	14	225-240 deg
22	2	16	240-255 deg
23	2	18	255-270 deg

29.3 XTRD - Extrapolation Unit Bank

The XTRP extrapolation unit receives the track information from the XFT linker cards and performs several functions:

- Send 2 bits of track-match data to the muon trigger boards for 6 muon towers per wedge separately for the CMU, CMX and IMU detectors
- Send 8 bits of track match data to each wedge in the DIRAC boards.
- Sends a list of tracks to the Two-Track trigger board which performs the two-track trigger algorithms.
- Send a sparsified track list to the SVT and the Level 2 processor crate

The first three functions are performed with table lookups. The muon match information is readout in TCMD. The track list and DIRAC-track map are included for each XTRP data

"XTRD"
Bank Number
Bank Version
Bank Length
Bank Type (I*4)
Number of Blocks (=2)
Pointer to Block 0
Pointer to Block 1
Pointer to end-of-data
Block 0: Number of cards (=n)
Pointer to Card 0
Pointer to Card 1
. . .
Pointer to Card (n - 1)
Pointer end of block
Card 0, Word 0
. . .
Card (n - 1), Word 26
Block 1: Number of cards (=1)
Pointer to Card 0
Pointer end of block
Track Trigger Header Word
Track Trigger Data Word

Table 112: *Structure of the XTRD bank.* Block 0 contains the XTRP Data boards; Block 1 contains the Track Trigger board. There are important comments at the end of this section related to variations in the format caused by card order and missing cards.

Offset	Description
$P_n + 0$	Header
$P_n + 1$	Track Data, Wedge 0, XFT Linker 0
. . .	. . .
$P_n + 12$	Track Data, Wedge 0, XFT Linker 11
$P_n + 13$	Track Data, Wedge 1, XFT Linker 0
. . .	. . .
$P_n + 24$	Track Data, Wedge 1, XFT Linker 11
$P_n + 25$	Calorimeter Extrapolation Data, Wedge 0
$P_n + 26$	Calorimeter Extrapolation Data, Wedge 1

Table 113: *XTRD Data board data block.*

board in XTRD. Diagnostic information is read out for the two-track trigger. In addition, the track trigger decision bits sent to and read-out through Level 1 and included in the TL1D bank. The structure of the XTRD bank is given in Table 112. There are 12 XTRP Data boards, each one serving 2 wedges. Data board 0 (XTRP slot 6) serves wedges 0 & 1; Data board 1 (XTRP slot 7) serves wedges 2 & 3; etc.
The format of the data from each board is given in Table 113. The structure of the track data words is:

Bit Range	Definition
2:0	local ϕ
3	Short-track bit
10:4	P_T
11	Isolation bit
12	Reserved
13	Bunch Zero marker
31:14	Not used

where the P_T bits are defined by the XFT.
The format of the DIRAC data word is:

Bit Range	Definition
7:0	XTRP/Calorimeter Bus
8	Bunch Zero Marker
31:9	Not used

For the Track Trigger block, the format of the Track Trigger data word is:

Bit Range	Definition
15:0	track trigger bits
16	bunch 0
31:17	Not used

The header words for the cards in blocks 0 & 1 have the following format:

Bit Range	Definition
7:0	bunch counter
12:8	Geographic Address (Slot)
22:13	Board serial number
31:23	Board Type

When all cards are present, there will be twelve cards in block 0. When a card is missing no data words will be written into the bank for the card. The pointer corresponding to a missing card may or may not be present. The order the cards appear in the bank may vary. Either of the two blocks may be empty. The card data section in block 1 may also be empty. Code which reads this bank must be able to deal with all these possibilities. It must check for empty blocks and card sections. It must use the slot number (geographic address) to identify card data in the bank. It cannot use the position of cards in the bank. The slot numbers are defined as follows.

- slot 6 = wedges 0 and 1
- slot 7 = wedges 2 and 3
- ...
- slot 17 = wedges 22 and 23

29.4 TC1D - Level 1 Calorimeter Trigger Bank

The TC1D bank contains data read out from the DIRAC and CRATESUM boards of the Level 1 calorimeter trigger. Each crate covers 2 calorimeter wedges (30°) on both the east and west sides for both the central and plug calorimeters. There are four DIRAC cards per wedge and two wedges per CRATESUM board. Each set of four DIRAC boards cover, with increasing slot number in the crate, East Plug, East Central, West Central, and West Plug ("Central" in the DIRAC boards goes to $|\eta| = 1.2$ including first plug tower). The bank is broken up into 12 blocks each containing data from 1 CRATESUM board and the corresponding 8 DIRAC boards that provide it data. There is one block of data coming from each crate. Table 114 describes the structure of the TC1D bank header and block pointer sections.

Table 115 describes the structure of the nth block of the TC1D bank. This structure may vary in two ways. First, the card order may vary. Second, cards and the corresponding pointer and data may be missing entirely. When accessing the bank, card data should be identified using the slot number. The slot number is in the header word at the start of the data for each of the DIRAC cards and the CRATESUM board. The bitfields in the header words are defined in Table 116. The meaning of the slot numbers is defined in the second column of Table 115. Note that the slots are numbered in different ways for odd and even numbered blocks.

For each DIRAC board, three words are written after the header word. The words contain one byte for the EM threshold and one byte for the EM+HAD (TOT) threshold for 6 channels. These three words are described in Table 117.

Each CRATESUM board writes nine words into the corresponding block: a header word followed by eight data words that are described in Table 118. The first two data words contain the data sent from the CRATESUM to the SUMET PreFRED and Tower Trigger PreFRED. The remaining six words contain the data received by the CRATESUM from eight DIRAC cards it services.

"TC1D"
Bank Number
Bank Version
Bank Length
Bank Type (I*4)
Number of Blocks (=12)
Pointer to Wedge 0-1 Block (=P0)
Pointer to Wedge 2-3 Block (=P1)
Pointer to Wedge 4-5 Block (=P2)
Pointer to Wedge 6-7 Block (=P3)
Pointer to Wedge 8-9 Block (=P4)
Pointer to Wedge 10-11 Block (=P5)
Pointer to Wedge 12-13 Block (=P6)
Pointer to Wedge 14-15 Block (=P7)
Pointer to Wedge 16-17 Block (=P8)
Pointer to Wedge 18-19 Block (=P9)
Pointer to Wedge 20-21 Block (=P10)
Pointer to Wedge 22-23 Block (=P11)
Pointer to End of Data

Table 114: *Structure of the TC1D bank header and block pointer sections.*

Displacement	Board / Slot: Pn=even (odd)	Data Description
Pn+0		Number of cards (=9)
Pn+1		Pointer to card 0 (=C0)
...		
Pn+9		Pointer to card 8 (=C8)
Pn+10		Pointer to End of Block
Pn+C0+0	DIRAC / 4 (13)	Wedge 2n, East Plug DIRAC Header
Pn+C0+1	”	Wedge 2n, East Plug DIRAC Word 1
Pn+C0+2	”	Wedge 2n, East Plug DIRAC Word 2
Pn+C0+3	”	Wedge 2n, East Plug DIRAC Word 3
Pn+C1+0	DIRAC / 5 (14)	Wedge 2n, East Central DIRAC Header
Pn+C1+1	”	Wedge 2n, East Central DIRAC Word 1
Pn+C1+2	”	Wedge 2n, East Central DIRAC Word 2
Pn+C1+3	”	Wedge 2n, East Central DIRAC Word 3
Pn+C2+0	DIRAC / 6 (15)	Wedge 2n, West Central DIRAC Header
Pn+C2+1	”	Wedge 2n, West Central DIRAC Word 1
Pn+C2+2	”	Wedge 2n, West Central DIRAC Word 2
Pn+C2+3	”	Wedge 2n, West Central DIRAC Word 3
Pn+C3+0	DIRAC / 7 (16)	Wedge 2n, West Plug DIRAC Header
Pn+C3+1	”	Wedge 2n, West Plug DIRAC Word 1
Pn+C3+2	”	Wedge 2n, West Plug DIRAC Word 2
Pn+C3+3	”	Wedge 2n, West Plug DIRAC Word 3
Pn+C4+0	DIRAC / 9 (18)	Wedge 2n+1, East Plug DIRAC Header
Pn+C4+1	”	Wedge 2n+1, East Plug DIRAC Word 1
Pn+C4+2	”	Wedge 2n+1, East Plug DIRAC Word 2
Pn+C4+3	”	Wedge 2n+1, East Plug DIRAC Word 3
Pn+C5+0	DIRAC / 10 (19)	Wedge 2n+1, East Central DIRAC Header
Pn+C5+1	”	Wedge 2n+1, East Central DIRAC Word 1
Pn+C5+2	”	Wedge 2n+1, East Central DIRAC Word 2
Pn+C5+3	”	Wedge 2n+1, East Central DIRAC Word 3
Pn+C6+0	DIRAC / 11 (20)	Wedge 2n+1, West Central DIRAC Header
Pn+C6+1	”	Wedge 2n+1, West Central DIRAC Word 1
Pn+C6+2	”	Wedge 2n+1, West Central DIRAC Word 2
Pn+C6+3	”	Wedge 2n+1, West Central DIRAC Word 3
Pn+C7+0	DIRAC / 12 (21)	Wedge 2n+1, West Plug DIRAC Header
Pn+C7+1	”	Wedge 2n+1, West Plug DIRAC Word 1
Pn+C7+2	”	Wedge 2n+1, West Plug DIRAC Word 2
Pn+C7+3	”	Wedge 2n+1, West Plug DIRAC Word 3
Pn+C8+0	CRATESUM / 8 (17)	CRATESUM Header
Pn+C8+1	”	CRATESUM Wedge Level ΣE_t Sums
Pn+C8+2	”	CRATESUM Trigger Summary
Pn+C8+3	”	Wedge 2n, DIRAC ΣE_t Sums
Pn+C8+4	”	Wedge 2n+1, DIRAC ΣE_t Sums
Pn+C8+5	”	Wedge 2n, East DIRAC Trigger Summaries
Pn+C8+6	”	Wedge 2n, West DIRAC Trigger Summaries
Pn+C8+7	”	Wedge 2n+1, East DIRAC Trigger Summaries
Pn+C8+8	”	Wedge 2n+1, West DIRAC Trigger Summaries

Table 115: *Nominal structure of the nth TC1D block. This structure may vary as described in the text!!!*

Bit	Header Words
31:23	Board Type (0x5 for DIRAC)
22:13	Serial Number
12:8	Geographic Address
7:0	Bunch Counter

Table 116: *Bit assignment for the header words for the DIRAC cards and the CRATESUM card.*

Word	Bits(31:24)	Bits(23:16)	Bits(15:8)	Bits(7:0)
1	Channel 1, TOT	Channel 1, EM	Channel 0, TOT	Channel 0, EM
2	Channel 3, TOT	Channel 3, EM	Channel 2, TOT	Channel 2, EM
3	Channel 5, TOT	Channel 5, EM	Channel 4, TOT	Channel 4, EM

Table 117: Definition of trigger decision words from DIRAC boards.

Word	Description			
Wedge ΣE_t Sums	Unused(31:20),Wedge 2n+1(19:10), Wedge 2n(9:0)			
Trigger Summary	Unused(31:20),Spare(19:16),2-bit Trig(15:10), 1-bit Trig(9:0)			
DIRAC ΣE_t Sums	Bits(31:24)	Bits(23:16)	Bits(15:8)	Bits(7:0)
	2n W Plug	2n W Cent.	2n E Cent.	2n E Plug
	2n+1 W Plug	2n+1 W Cent.	2n+1 E Cent.	2n+1 E Plug
DIRAC Trigger	Bits(31:16)		Bits(15:0)	
Summaries	2n E Cent.		2n E Plug	
	2n W Plug		2n W Cent.	
	2n+1 E Cent.		2n+1 E Plug	
	2n+1 W Plug		2n+1 W Cent.	

Table 118: Definition of ΣE_T and trigger words from CRATESUM boards.

29.5 TCMD - Central Muon Matchbox Trigger Bank

The format of the TCMD bank is given in Table 119. The bank consists of three Blocks of data. At the start of each Block of data are a set of pointers which point to the data coming from one card.

"TCMD"
Bank Number
Bank Version
Bank Length (variable)
Bank Type (I*4)
Number of Blocks (=3)
Pointer to Block 0
Pointer to Block 1
Pointer to Block 2
Pointer to End of Data
Block 0: Number of Cards (=12)
Pointer to Card 0
...
Pointer to Card 11
Pointer to End of Block
Matchbox Data
...
Matchbox Data
Block 1: Number of Cards (=4)
Pointer to Card 0
...
Pointer to Card 3
Pointer to End of Block
Prematch Data
...
Prematch Data
Block 2: Number of Cards (=1)
Pointer to Card 0
Pointer to End of Block
MTSC Data
...
MTSC Data

Table 119: *Structure of the TCMD bank.*

29.5.1 Matchbox Data

Each Matchbox card collects muon trigger primitives over an entire 30 degree slice of the detector and matches them with track trigger data received from the XTRP modules. There are a total of twelve matchbox cards in the system to provide complete 360 degree trigger coverage. The initial plan is that there will be twelve cards that are read out in order of increasing phi starting with the card covering the region between 0 and 30 degrees and finishing with the card covering the region between 330 and 360 degrees. Software that reads this bank should be written to handle cards that are missing or read out in another order. The header word of each card has a geographical address (slot number) that allows one to identify a card without using its position in the block. The slot numbers run sequentially from 7 to 18, also in phi order. For example, slot number 7 corresponds to the card that handles the region between 0 and 30 degrees.

The data words output from each Matchbox card are defined below. Note that the bit fields contained in these words are not documented yet (the plan is that this will be done in a separate note referenced by this note, it will be a long and complicated description).

Displacement	**Data Description**
P0	Header Word
P0+1	FPGA Version Numbers
P0+2	Matchbox Output Bits
P0+3	CMU East - High Pt Bits
P0+4	CMU East - Low Pt Bits
P0+5	CMU West - High Pt Bits
P0+6	CMU West - Low Pt Bits
P0+7	CMX East - High Pt Bits
P0+8	CMX East - Low Pt Bits
P0+9	CMX West - High Pt Bits
P0+10	CMX West - Low Pt Bits
P0+11	CSX and HAD East Bits
P0+12	CSX and HAD West Bits
P0+13	BMU East - High Pt Bits
P0+14	BMU East - Low Pt Bits
P0+15	BMU West - High Pt Bits
P0+16	BMU West - Low Pt Bits
P0+17	BSU East Bits
P0+18	BSU West Bits
P0+19	TSU and HAD East Bits
P0+20	TSU and HAD West Bits
P0+21	Spare Word A
P0+22	Spare Word B
P0+23	TOF Bits
P0+24	CMU East Diagnostic Bits
P0+25	CMU West Diagnostic Bits
P0+26	Eta Gap Diagnostic Bits
P0+27	CMX East Diagnostic Bits
P0+28	CMX West Diagnostic Bits
P0+29	IMA East Diagnostic Bits
P0+30	IMA West Diagnostic Bits
P0+31	IMB East Diagnostic Bits
P0+32	IMB West Diagnostic Bits
P0+33	XTRP CMU Map Bits
P0+34	XTRP CMX Map Bits
P0+35	XTRP BMU Map Bits

29.5.2 Prematch Data

Prematch Cards collect CMP and CSP trigger primitives and match them in their natural rectangular geometry. Matches are converted into an azimuthal map to match the rest of the detector and passed to the appropriate matchbox cards. Four Prematch cards are included in the muon trigger readout. Each of the cards covers one full wall of the detector. The initial plan is that the cards are readout in the following order: Bottom, South, Top, North. Software that accesses this bank should be able to handle cases where a card is missing or read out in some other order. These cards also will have a geographic address (slot number) in the header word to identify them. The slot numbers will be 3 = bottom, 4 = south, 5 = top, and 6 = north.

The data words output from each Prematch card are defined as follows:

Displacement	**Data Description**
P0	Header Word
P0+1	FPGA Version Numbers
P0+2	CMP Primitives for Stacks 00-23
P0+3	CMP Primitives for Stacks 24-47
P0+4	CMP Primitives for Stacks 47-71
P0+5	CMP Primitives for Stacks 72-95
P0+6	CSP East Bits for Counters 00-23
P0+7	CSP East Bits for Counters 24-47
P0+8	CSP West Bits for Counters 00-23
P0+9	CSP West Bits for Counters 24-47
P0+10	CMP/CSP East Matches for Stacks 00-23
P0+11	CMP/CSP East Matches for Stacks 24-47
P0+12	CMP/CSP East Matches for Stacks 48-71
P0+13	CMP/CSP East Matches for Stacks 72-95
P0+14	CMP/CSP West Matches for Stacks 00-23
P0+15	CMP/CSP West Matches for Stacks 24-47
P0+16	CMP/CSP West Matches for Stacks 48-71
P0+17	CMP/CSP West Matches for Stacks 72-95

29.5.3 MTSC Data

The MTSC collects trigger decisions from each of the 30 degree Matchbox cards and produces the final muon trigger bits which are passed to FRED. The data words output from the MTSC card are defined as follows:

Displacement	**Data Description**
P0	Header Word
P0+1	Trigger Output Bits 00-19
P0+2	Trigger Output Bits 20-39

29.6 TL1D - Level 1 Trigger Bank

The Level 1 trigger information from the PreFRED modules [8, 7] is stored in the TL1D bank. The PreFRED modules will only be read out for certain values of the Readout List as determined by Level 2 [9, 10, 11] and thus the TL1D bank will not be available for every event. The format of the TL1D bank is given in Table 121. The card header words in the TL1D bank are defined in Table 120. The geographical addresses (slot numbers) and contents of the other data words in TL1D are defined in Table 122.

Bit	Function
7:0	Bunch Counter
8:12	Geographical Address (Slot)
13:22	Serial Number
23:31	Module ID (PREFRED)

Table 120: *Header word definition in TL1D. The header word is the first word in every card data section in the bank.*

"TL1D"
Bank Number
Bank Version
Bank Length
Bank Type (I*4)
Number of Blocks (=1)
Pointer to Block 0
Pointer to End of Data
Number of Cards (variable, usually 6)
Pointer to Card 0
Pointer to Card 1
Pointer to Card 2
Pointer to Card 3
Pointer to Card 4
Pointer to Card 5
Pointer to End of Block
Card 0: SUMET PreFRED data word 0
Card 0: SUMET PreFRED data word 1
Card 0: SUMET PreFRED data word 2
Card 0: SUMET PreFRED data word 3
Card 1: CALTRG PreFRED data word 0
Card 1: CALTRG PreFRED data word 1
Card 1: CALTRG PreFRED data word 2
Card 2: MUTRG PreFRED data word 0
Card 2: MUTRG PreFRED data word 1
Card 2: MUTRG PreFRED data word 2
Card 3: TRKTRG PreFRED data word 0
Card 3: TRKTRG PreFRED data word 1
Card 3: TRKTRG PreFRED data word 2
Card 4: BSCTRG PreFRED data word 0
Card 4: BSCTRG PreFRED data word 1
Card 4: BSCTRG PreFRED data word 2
Card 5: Multi-purpose PreFRED data word 0
Card 5: Multi-purpose PreFRED data word 1
Card 5: Multi-purpose PreFRED data word 2

Table 121: *Structure of the TL1D bank. All six cards may not always be present in the bank. When a card is missing, there will usually be no card pointer (or the card data section will be empty). Card order may vary. This means that when accessing the bank one must use the geographic address (GA or also called slot number) to identify the data in each section of the bank.*

Pfred Type	Word	Bits	Definition
Sumet (GA 4)	0		Header Word
	1	31:24	Unused
		23:22	MET^2 Trigger bits
		21:20	E_T Trigger bits
		19:16	FRED-delayed Trigger bits (for hardware diagnostic)
		15:0	MET^2 $1GeV^2$ units
	2	31	FRED-delayed B0 (for hardware diagnostic)
		30:21	ΣE_y - bits 29:21 = abs. value, bit 30 sign bit
		20:11	ΣE_x - bits 19:11 = abs. value, bit 20 sign bit
		10:0	ΣE_T 1GeV units
	3	31:11	Unused
		10:0	ΣE_T from previous crossing
Caltrg (GA 5)	0		Header Word
	1	31:20	Unused
		19:0	FRED-delayed Trigger bits
	2	31	FRED-delayed B0 (for hardware diagnostic)
		30:16	Unused
		15:11	Double bit triggers (3 pairs implemented)
		10	Unused
		9:0	Single bit triggers (10)
MUtrg (GA 6)	0		Header Word
	1	31:20	Unused
		19:0	FRED-delayed Trigger bits
	2	31	FRED-delayed B0 (for hardware diagnostic)
		19:0	Triggers bits
TRKtrg (GA 7)	0		Header Word
	1	31:20	Unused
		19:0	FRED-delayed Trigger bits
	2	31	FRED-delayed B0 (for hardware diagnostic)
		30:20	Unused
		19:0	Triggers bits
BSCtrg (GA 10)	0		Header Word
	1	31:20	Unused
		19:0	FRED-delayed Trigger bits
	2	31	FRED-delayed B0 (for hardware diagnostic)
		30:20	Unused
		19:0	Triggers bits
Multi (GA 11)	0		Header Word
	1	31:20	Unused
		19:0	FRED-delayed Trigger bits
	2	31	FRED-delayed B0 (for hardware diagnostic)
		30:20	Unused
		19:0	Triggers bits

Table 122: *Contents of the data words in the TL1D bank. In the first column, the value of the geographical address (GA or slot) needed to identify each card is shown.*

29.7 TFRD - Global Level 1 Decision Bank

The TFRD bank contains the data from the Global Level 1 Decision Module (a.k.a. FRED) [7]. The format of the TFRD bank is given in Table 123. The second and third words in the card section contain bits representing the trigger primitives provided by the PreFRED modules [8]. The next two words contain bits representing the 64 Level 1 trigger decisions before prescaling. The next two words contain the bits designating the final 64 Level 1 trigger decisions. The exact definition of each bit will be trigger table dependent.

The next two words contain the count of the number of Bunch Crossing triggers that have occurred since the beginning of the current run. The value of the Bunch Crossing trigger counter is used by the Data Logger in incrementing the Run Section number [6], thus this trigger will be generated and counted regardless of whether the DAQ chain is "live" [12]. For more information on Run Sections and the luminosity calculation in Run II, see reference [6]. The last two words in the TFRD bank represent the number of triggers accrued for one of the higher rate processes (*e.g.* $B \to \pi\pi$). The function of this counter is to provide a precise determination of one of the trigger rates being counted by the Level 1 scalers which are described in Section 29.8. The high rate trigger will only be counted when the DAQ system is live.

"TFRD"
Bank Number
Bank Version
Bank Length
Bank Type (I*4)
Number of Blocks (=1)
Pointer to Block 0
Pointer to End of Data
Number of Cards (=1)
Pointer to Card 0
Pointer to End of Block
Card 0: Header Word
Card 0: FRED trigger inputs [31:0]
Card 0: FRED trigger inputs [63:32]
Card 0: Unprescaled trigger decisions [31:0]
Card 0: Unprescaled trigger decisions [63:32]
Card 0: Prescaled trigger decisions [31:0]
Card 0: Prescaled trigger decisions [63:32]
Card 0: Bunch Crossing trigger counter word [31:0]
Card 0: Bunch Crossing trigger counter word [63:32]
Card 0: High rate trigger counter word 1 [31:0]
Card 0: High rate trigger counter word 2 [63:62]

Table 123: *Structure of the TFRD bank.*

29.8 SL1D - Level 1 Scalar Bank

The SL1D bank contains data from scaler cards which count the individual trigger decisions and associated information from the Global Level 1 Decision module [7] and various rates for the Physics partition as determined in the Trigger Supervisor crate.

The scaler modules are read out each time a Level 2 Accept message is broadcasted to the front-end and trigger crates by the Trigger Supervisor. The scalers employed are commercially available VME multiscalers from Struck with firmware tailored to CDF's needs (model SIS3805). Note that the scalers are not pipelined devices and **do not** report counts accumulated only up to the time of the Level 1 Accept that led to the Level 2 Accept forcing the readout. Instead the counts are from the sampling which occurs when the Level 2 Accept message arrives at the scaler crate and this occurs at a time on the order milliseconds after the relevant Level 1 Accept. This time is not fixed but determined by the performance of the DAQ readout system.

The firmware implementation used on the scalers is the (version 2) shadow register type, in which the sampling consists of an asynchronous parallel copying of the scaler counts into a FIFO register for subsequent readout. (The sampling message is sent to all modules in the Scaler crate via a VME Broadcast message). The result of this asynchronous copying means that the least significant 6 bits of the count may be inaccurate (for a high rate input) for a particular readout; however no counts are lost.

The contents of each scaler channel are reported in a pair of consecutive 32 bit words. The dynamic range of the scaler count data is 48 bits, with the LSB 32 data bits reported in the second ('odd') word of each pair. The MSB 16 data bits are reported in the lower two bytes of the first ('even') word; the upper byte of this word contains the channel number (0-31) on the scaler module. (Note also that the scalers do not have separate registers for data associated with the 4 Level 2 buffers).

The format of the SL1D bank is given in detail in Tables 124, 125, 126, 127, 128, 129, 130, and 131. The data is organized as follows: first the trigger rates from Global Level 1 (aka FRED); second, global decision counts and livetime, deadtime and other diagnostic information from the Trigger Supervisor for the Physics Partition; third, the L1 Accepts per bunch crossing; and fourth, the livetime per bunch crossing.

FRED Scalers

From the Global Level 1 Decision Module (aka FRED), the counts for each of the 64 trigger primitives, the 64 unprescaled trigger decisions and the 64 final trigger decisions are scaled. The scaled signals are gated on the FRED module by GFRED = TS_RUN * $\overline{\text{TS\_PAUSE}}$ * $\overline{\text{INHIBIT}}$, which is supplied by the Trigger Supervisor in the Level 1 Accept message. [9, 12]

Trigger Supervisor Scalers

Scaler data from one Trigger Supervisor (typically the one serving the Physics partition) can also be recorded. This data includes information pertaining to the Runtime, Livetime, and a tally of Deadtime due to different sources [12]. Also included is information pertaining to the usage of the four Level 2 buffers and the frequency with which they are found in the EMPTY, FULL, PENDing or BUSY states.

The number of Global Level 1 decisions are scaled in various forms: as seen by all crates in the partition (CDF_L1A), per Level 2 buffer, as sent by FRED to the TS, as well as the FRED decisions gated by GRUN [12].

The number of Global Level 2 decisions are also scaled in various forms: as seen by all crates in the partition (CDF_L2A, CDF_L2R), per Level 2 buffer, as exchanged between the Level 2 Processors and the Trigger Supervisor via the logic on the Global Level 2 module.

Some additional information is also kept including the number of B0's and BC's observed, the time spent in HALT and in the ABORT gaps, etc.

Note: The quantities recorded in the Trigger Supervisor scalers include some elements that should be treated as counts and some elements that can be converted to times.

SL1D_TS[0] through [15] and SL1D_TS[32] through [39] are all quantities reflecting simple counts, e.g. number of Level 1 Accepts send to the front-end crates (CDF_L1A) or for particular L2 Buffer (e.g. CDF_L1A_Buffer_1), number of times an event was rejected by L2 for a particular L2 buffer (e.g. CDF_L2R_Buffer_3) etc.

SL1D_TS[16] through [31], SL1D_TS[43] and SL1D_TS[49] through [63] are all quantities reflecting time spent in a particular state as defined to allow for a proper accounting of livetime and assigning blame for deadtime [12]. Each count represents 132 nsec of time. **However all of these signals are gated by GRUN [12] which requires BC to be asserted. To 'correct' to an effective 'wall clock' time one needs to inflate the count by a factor of 159/(number of BC in one complete turn).** The number of BC's per turn can and should be determined by CDF_BC/CDF_B0 (i.e. SL1D_TS[45]/SL1D_TS[44]).

SL1D_TS[27] counts Synchronization Errors and is non-zero if a glitch has been observed in the CDF CLOCK on the TS crate (number of 132 nsec intervals between successive B0 markers does not equal the distance around the ring).

SL1D_TS[43] records any under- or over-flow (Miscount) in the logic used to form the buffer state occupancy information recorded in SL1D_TS[49] through [63].

SL1D_TS[40], [41], [47] and [48] all reflect time in a pariticular state, each count representing 132 nsec. These signals have not been gated by GRUN, so no dependence on BC and no 'correction' factor is needed.

SL1D_TS[42], [44], [45] and [46] all reflect simple counts, e.g. the number of times HALT or RECOVER was issued.

Finally the quantities in SL1D_TS[12] through [15], SL1D_TS[43] and SL1D_TS[49] through [63] are meaningful only if a Level 2 Processor (or an emulator on the Global Level 2 module) is communicating with the Trigger Supervisor which has the Scaler crate in the same partition.

Per Bunch Accounting

The Level 1 Accepts per bunch (i.e. in coincidence with BC occurring outside of the Abort Gap) are recorded for up to 159 possible bunches with an additional channel being used to scale any occurrances of synchronization errors. When the number of bunches is 36, then 37 scaler channels will be employed. The first channel will contain the Level 1 Accepts for Bunch 0 (the first BC), the second will contain the the Level 1 Accepts for the next (second) BC, etc. The last, 37th channel should read 0, unless there are synchronization problems.

If we were ever to move beyond 36 bunches, the format will be extended in the obvious way through the use of the card pointers.

The Livetime per bunch is handled in a manner similar to the Level 1 Accepts per bunch, described above.

Note: The per bunch information is presented only for locations around the ring (out of 159) marked with BC. This means that a map is required to associate per bunch information given in this bank with the Bunch ID as recorded with each event.

Note: Since the scaler cards support 32 channels, the number of channels needed to cover all locations marked with BC will not necessarily be an integral multiple of 32. This means that for both Level 1 Accepts and Livetime per bunch data, some cards will only need to report data for the first several channels and trailing cards will report no data. In this case the card pointers will be set accordingly. However be forwarned that in the early stages the data for these unused channels may instead be filled with all zero words (again with the card pointers set accordingly). The online group will decided when to suppress the unused channels.

"SL1D"
Bank Number
Bank Version
Bank Length
Bank Type (I*4)
Number of Blocks (=1)
Pointer to Block 0
Pointer to End of Data
Number of Cards (=18)
Pointer to Card 0
Pointer to Card 1
Pointer to Card 2
Pointer to Card 3
Pointer to Card 4
Pointer to Card 5
Pointer to Card 6
Pointer to Card 7
Pointer to Card 8
Pointer to Card 9
Pointer to Card 10
Pointer to Card 11
Pointer to Card 12
Pointer to Card 13
Pointer to Card 14
Pointer to Card 15
Pointer to Card 16
Pointer to Card 17
Pointer to End of Block

Table 124: *Structure of the SL1D bank (Header/Pointer Section). Note that the number of pointers is fixed. There will always be 18 pointers even if cards are missing. The data section for a given card may be filled with zeroes or empty, but the card pointer will always be present.*

Card 0: Word00	FRED trigger input 0 [63:32]
Card 0: Word01	FRED trigger input 0 [31:0]
. . .	
Card 0: Word62	FRED trigger input 31 [63:32]
Card 0: Word63	FRED trigger input 31 [31:0]
Card 1: Word00	FRED trigger input 32 [63:32]
Card 1: Word01	FRED trigger input 32 [31:0]
. . .	
Card 1: Word62	FRED trigger input 63 [63:32]
Card 1: Word63	FRED trigger input 63 [31:0]
Card 2: Word00	Unprescaled trigger decision 0 [63:32]
Card 2: Word01	Unprescaled trigger decision 0 [31:0]
. . .	
Card 2: Word62	Unprescaled trigger decision 31 [63:32]
Card 2: Word63	Unprescaled trigger decision 31 [31:0]
Card 3: Word00	Unprescaled trigger decision 32 [63:32]
Card 3: Word01	Unprescaled trigger decision 32 [31:0]
. . .	
Card 3: Word62	Unprescaled trigger decision 63 [63:32]
Card 3: Word63	Unprescaled trigger decision 63 [31:0]
Card 4: Word00	Prescaled trigger decision 0 [63:32]
Card 4: Word01	Prescaled trigger decision 0 [31:0]
. . .	
Card 4: Word62	Prescaled trigger decision 31 [63:32]
Card 4: Word63	Prescaled trigger decision 31 [31:0]
Card 5: Word00	Prescaled trigger decision 32 [63:32]
Card 5: Word01	Prescaled trigger decision 32 [31:0]
. . .	
Card 5: Word62	Prescaled trigger decision 63 [63:32]
Card 5: Word63	Prescaled trigger decision 63 [31:0]

Table 125: *Structure of the SL1D bank (FRED Section).*

SL1D_TS[00]	Card 6: Word00	CDF_L2A [63:32]
SL1D_TS[00]	Card 6: Word01	CDF_L2A [31:0]
SL1D_TS[01]	Card 6: Word02	CDF_L2R [63:32]
SL1D_TS[01]	Card 6: Word03	CDF_L2R [31:0]
SL1D_TS[02]	Card 6: Word04	FRED_L1A [63:32]
SL1D_TS[02]	Card 6: Word05	FRED_L1A [31:0]
SL1D_TS[03]	Card 6: Word06	CALIBRATION_ENABLE [63:32]
SL1D_TS[03]	Card 6: Word07	CALIBRATION_ENABLE [31:0]
SL1D_TS[04]	Card 6: Word08	CDF_L1A_Buffer_3 [63:32]
SL1D_TS[04]	Card 6: Word09	CDF_L1A_Buffer_3 [31:0]
SL1D_TS[05]	Card 6: Word10	CDF_L1A_Buffer_2 [63:32]
SL1D_TS[05]	Card 6: Word11	CDF_L1A_Buffer_2 [31:0]
SL1D_TS[06]	Card 6: Word12	CDF_L1A_Buffer_1 [63:32]
SL1D_TS[06]	Card 6: Word13	CDF_L1A_Buffer_1 [31:0]
SL1D_TS[07]	Card 6: Word14	CDF_L1A_Buffer_0 [63:32]
SL1D_TS[07]	Card 6: Word15	CDF_L1A_Buffer_0 [31:0]
SL1D_TS[08]	Card 6: Word16	FRED.AND.CALIB_L1A [63:32]
SL1D_TS[08]	Card 6: Word17	FRED.AND.CALIB_L1A [31:0]
SL1D_TS[09]	Card 6: Word18	CDF_L1A [63:32]
SL1D_TS[09]	Card 6: Word19	CDF_L1A [31:0]
SL1D_TS[10]	Card 6: Word20	CDF_L1A.AND.FRED_L1A [63:32]
SL1D_TS[10]	Card 6: Word21	CDF_L1A.AND.FRED_L1A [31:0]
SL1D_TS[11]	Card 6: Word22	CDF_L1A.AND.CALIB_L1A [63:32]
SL1D_TS[11]	Card 6: Word23	CDF_L1A.AND.CALIB_L1A [31:0]
SL1D_TS[12]	Card 6: Word24	L2A_L2P_to_TS [63:32]
SL1D_TS[12]	Card 6: Word25	L2A_L2P_to_TS [31:0]
SL1D_TS[13]	Card 6: Word26	L2R_L2P_to_TS [63:32]
SL1D_TS[13]	Card 6: Word27	L2R_L2P_to_TS [31:0]
SL1D_TS[14]	Card 6: Word28	L2A_TS_to_L2P [63:32]
SL1D_TS[14]	Card 6: Word29	L2A_TS_to_L2P [31:0]
SL1D_TS[15]	Card 6: Word30	L2R_TS_to_L2P [63:32]
SL1D_TS[15]	Card 6: Word31	L2R_TS_to_L2P [31:0]

Table 126: *Structure of the SL1D bank, continued (Global Trigger Section).*

SL1D_TS[16]	Card 6: Word32	≥1 L2 buffers Empty [63:32]
SL1D_TS[16]	Card 6: Word33	≥1 L2 buffers Empty [31:0]
SL1D_TS[17]	Card 6: Word34	≥1 L2 buffers Full [63:32]
SL1D_TS[17]	Card 6: Word35	≥1 L2 buffers Full [31:0]
SL1D_TS[18]	Card 6: Word36	≥1 L2 buffers Pending [63:32]
SL1D_TS[18]	Card 6: Word37	≥1 L2 buffers Pending [31:0]
SL1D_TS[19]	Card 6: Word38	1 L2 buffer Busy [63:32]
SL1D_TS[19]	Card 6: Word39	1 L2 buffer Busy [31:0]
SL1D_TS[20]	Card 6: Word40	Inhibit Deadtime [63:32]
SL1D_TS[20]	Card 6: Word41	Inhibit Deadtime [31:0]
SL1D_TS[21]	Card 6: Word42	WAIT/BUSY Deadtime [63:32]
SL1D_TS[21]	Card 6: Word43	WAIT/BUSY Deadtime [31:0]
SL1D_TS[22]	Card 6: Word44	L2_or_Readout Deadtime [63:32]
SL1D_TS[22]	Card 6: Word45	L2_or_Readout Deadtime [31:0]
SL1D_TS[23]	Card 6: Word46	Readout Deadtime [63:32]
SL1D_TS[23]	Card 6: Word47	Readout Deadtime [31:0]
SL1D_TS[24]	Card 6: Word48	L1_Done Deadtime [63:32]
SL1D_TS[24]	Card 6: Word49	L1_Done Deadtime [31:0]
SL1D_TS[25]	Card 6: Word50	L2 Deadtime [63:32]
SL1D_TS[25]	Card 6: Word51	L2 Deadtime [31:0]
SL1D_TS[26]	Card 6: Word52	TS Bookkeeping Deadtime [63:32]
SL1D_TS[26]	Card 6: Word53	TS Bookkeeping Deadtime [31:0]
SL1D_TS[27]	Card 6: Word54	Synchronization Errors [63:32]
SL1D_TS[27]	Card 6: Word55	Synchronization Errors [31:0]
SL1D_TS[28]	Card 6: Word56	GLIVE_TIME [63:32]
SL1D_TS[28]	Card 6: Word57	GLIVE_TIME [31:0]
SL1D_TS[29]	Card 6: Word58	GRUN_TIME [63:32]
SL1D_TS[29]	Card 6: Word59	GRUN_TIME [31:0]
SL1D_TS[30]	Card 6: Word60	FRED_L1A.AND.GRUN [63:32]
SL1D_TS[30]	Card 6: Word61	FRED_L1A.AND.GRUN [31:0]
SL1D_TS[31]	Card 6: Word62	FRED_L1R.AND.GRUN [63:32]
SL1D_TS[31]	Card 6: Word63	FRED_L1R.AND.GRUN [31:0]

Table 127: *Structure of the SL1D bank, continued (Livetime & Deadtime Section).*

SL1D_TS[32]	Card 7: Word00	CDF_L2A_Buffer_3 [63:32]
SL1D_TS[32]	Card 7: Word01	CDF_L2A_Buffer_3 [31:0]
SL1D_TS[33]	Card 7: Word02	CDF_L2A_Buffer_2 [63:32]
SL1D_TS[33]	Card 7: Word03	CDF_L2A_Buffer_2 [31:0]
SL1D_TS[34]	Card 7: Word04	CDF_L2A_Buffer_1 [63:32]
SL1D_TS[34]	Card 7: Word05	CDF_L2A_Buffer_1 [31:0]
SL1D_TS[35]	Card 7: Word06	CDF_L2A_Buffer_0 [63:32]
SL1D_TS[35]	Card 7: Word07	CDF_L2A_Buffer_0 [31:0]
SL1D_TS[36]	Card 7: Word08	CDF_L2R_Buffer_3 [63:32]
SL1D_TS[36]	Card 7: Word09	CDF_L2R_Buffer_3 [31:0]
SL1D_TS[37]	Card 7: Word10	CDF_L2R_Buffer_2 [63:32]
SL1D_TS[37]	Card 7: Word11	CDF_L2R_Buffer_2 [31:0]
SL1D_TS[38]	Card 7: Word12	CDF_L2R_Buffer_1 [63:32]
SL1D_TS[38]	Card 7: Word13	CDF_L2R_Buffer_1 [31:0]
SL1D_TS[39]	Card 7: Word14	CDF_L2R_Buffer_0 [63:32]
SL1D_TS[39]	Card 7: Word15	CDF_L2R_Buffer_0 [31:0]
SL1D_TS[40]	Card 7: Word16	TOTAL_TIME [63:32]
SL1D_TS[40]	Card 7: Word17	TOTAL_TIME [31:0]
SL1D_TS[41]	Card 7: Word18	CDF_HALT_TIME [63:32]
SL1D_TS[41]	Card 7: Word19	CDF_HALT_TIME [31:0]
SL1D_TS[42]	Card 7: Word20	CDF_RECOVER_or_RUN [63:32]
SL1D_TS[42]	Card 7: Word21	CDF_RECOVER_or_RUN [31:0]
SL1D_TS[43]	Card 7: Word22	Miscount [63:32]
SL1D_TS[43]	Card 7: Word23	Miscount [31:0]
SL1D_TS[44]	Card 7: Word24	CDF_B0 [63:32]
SL1D_TS[44]	Card 7: Word25	CDF_B0 [31:0]
SL1D_TS[45]	Card 7: Word26	CDF_BC [63:32]
SL1D_TS[45]	Card 7: Word27	CDF_BC [31:0]
SL1D_TS[46]	Card 7: Word28	CDF_BC_PRIME [63:32]
SL1D_TS[46]	Card 7: Word29	CDF_BC_PRIME [31:0]
SL1D_TS[47]	Card 7: Word30	CDF_ABORT_TIME [63:32]
SL1D_TS[47]	Card 7: Word31	CDF_ABORT_TIME [31:0]

Table 128: *Structure of the SL1D bank, continued (Diagnostic Section).*

SL1D_TS[48]	Card 7: Word32	GFRED_TIME [63:32]
SL1D_TS[48]	Card 7: Word33	GFRED_TIME [31:0]
SL1D_TS[49]	Card 7: Word34	0 Empty Level 2 Buffers [63:32]
SL1D_TS[49]	Card 7: Word35	0 Empty Level 2 Buffers [31:0]
SL1D_TS[50]	Card 7: Word36	1 Empty Level 2 Buffer [63:32]
SL1D_TS[50]	Card 7: Word37	1 Empty Level 2 Buffer [31:0]
SL1D_TS[51]	Card 7: Word38	2 Empty Level 2 Buffers [63:32]
SL1D_TS[51]	Card 7: Word39	2 Empty Level 2 Buffers [31:0]
SL1D_TS[52]	Card 7: Word40	3 Empty Level 2 Buffers [63:32]
SL1D_TS[52]	Card 7: Word41	3 Empty Level 2 Buffers [31:0]
SL1D_TS[53]	Card 7: Word42	4 Empty Level 2 Buffers [63:32]
SL1D_TS[53]	Card 7: Word43	4 Empty Level 2 Buffers [31:0]
SL1D_TS[54]	Card 7: Word44	0 Full Level 2 Buffers [63:32]
SL1D_TS[54]	Card 7: Word45	0 Full Level 2 Buffers [31:0]
SL1D_TS[55]	Card 7: Word46	1 Full Level 2 Buffer [63:32]
SL1D_TS[55]	Card 7: Word47	1 Full Level 2 Buffer [31:0]
SL1D_TS[56]	Card 7: Word48	2 Full Level 2 Buffers [63:32]
SL1D_TS[56]	Card 7: Word49	2 Full Level 2 Buffers [31:0]
SL1D_TS[57]	Card 7: Word50	3 Full Level 2 Buffers [63:32]
SL1D_TS[57]	Card 7: Word51	3 Full Level 2 Buffers [31:0]
SL1D_TS[58]	Card 7: Word52	4 Full Level 2 Buffers [63:32]
SL1D_TS[58]	Card 7: Word53	4 Full Level 2 Buffers [31:0]
SL1D_TS[59]	Card 7: Word54	0 Pending Level 2 Buffers [63:32]
SL1D_TS[59]	Card 7: Word55	0 Pending Level 2 Buffers [31:0]
SL1D_TS[60]	Card 7: Word56	1 Pending Level 2 Buffer [63:32]
SL1D_TS[60]	Card 7: Word57	1 Pending Level 2 Buffer [31:0]
SL1D_TS[61]	Card 7: Word58	2 Pending Level 2 Buffers [63:32]
SL1D_TS[61]	Card 7: Word59	2 Pending Level 2 Buffers [31:0]
SL1D_TS[62]	Card 7: Word60	3 Pending Level 2 Buffers [63:32]
SL1D_TS[62]	Card 7: Word61	3 Pending Level 2 Buffers [31:0]
SL1D_TS[63]	Card 7: Word62	4 Pending Level 2 Buffers [63:32]
SL1D_TS[63]	Card 7: Word63	4 Pending Level 2 Buffers [31:0]

Table 129: *Structure of the SL1D bank, continued (more Diagnostic Section).*

Card 8: Word00	Level 1 Accepts for Bunch 0 [63:32]
Card 8: Word01	Level 1 Accepts for Bunch 0 [31:0]
. . .	
Card 8: Word62	Level 1 Accepts for Bunch 31 [63:32]
Card 8: Word63	Level 1 Accepts for Bunch 31 [31:0]
Card 9: Word00	Level 1 Accepts for Bunch 32 [63:32]
Card 9: Word01	Level 1 Accepts for Bunch 32 [31:0]
. . .	
Card 9: Word06	Level 1 Accepts for Bunch 35 [63:32]
Card 9: Word07	Level 1 Accepts for Bunch 35 [31:0]
Card 9: Word08	Synchronization Errors [63:32]
Card 9: Word09	Synchronization Errors [31:0]

Table 130: *Structure of the SL1D bank, continued (L1A per Bunch Section).*

Card 13: Word00	Livetime for Bunch 0 [63:32]
Card 13: Word01	Livetime for Bunch 0 [31:0]
. . .	
Card 13: Word62	Livetime for Bunch 31 [63:32]
Card 13: Word63	Livetime for Bunch 31 [31:0]
Card 14: Word00	Livetime for Bunch 32 [63:32]
Card 14: Word01	Livetime for Bunch 32 [31:0]
. . .	
Card 14: Word06	Livetime for Bunch 35 [63:32]
Card 14: Word07	Livetime for Bunch 35 [31:0]
Card 14: Word08	Synchronization Errors [63:32]
Card 14: Word09	Synchronization Errors [31:0]

Table 131: *Structure of the SL1D bank, continued (Livetime per Bunch Section).*

29.9 TC2D - Level 2 Calorimeter Diagnostic Bank

The format of the TC2D bank is given in Table 132. The bank consists of 6 blocks, one for each L2CAL crate. Each L2CAL crate houses 12 DCAS cards, arranged in four groups of three cards each. Each group of three cards covers one 15^o wedge. One card for trigger towers 0 to 7, one card for trigger towers 8 to 15, and one for trigger towers 16 to 23.

Table 133 shows the format of the data written into the bank for each card. There is a header word at the start of each card. Then for each of the 8 trigger towers, there is an energy word and a tower usage word. The header word is defined in Table 134. The energy word is defined in Table 135. The tower usage word is defined in Table 136.

Cards appear in the blocks in the order they are read out. This order may vary. Sometimes a card will not be read out. In this case, both the card data and the card pointer will not be present. One must use the block number and the slot ID in the header word to identify cards. Table 137 specifies the correspondence between block/slot and wedge/trigger towers.

"TC2D"
Bank Number
Bank Version
Bank Length (fixed)
Bank Type (I*4)
Number of Blocks (=6)
Pointer to L2CAL Block 0
Pointer to L2CAL Block 1
Pointer to L2CAL Block 2
Pointer to L2CAL Block 3
Pointer to L2CAL Block 4
Pointer to L2CAL Block 5
Pointer to End of Data
Block 0: Number of Cards (12)
Pointer to Card 0 (=P0)
...
Pointer to Card 11
Pointer to End of Block
L2CAL Data for Card 0 Block 0
...
L2CAL Data for Card 1 Block 0
...
...
L2CAL Data for Card 11 Block 0
...
Block 5: Number of Cards (12)
Pointer to Card 0
...
Pointer to Card 11
Pointer to End of Block
L2CAL Data for Card 0 Block 5
...
L2CAL Data for Card 1 Block 5
...
L2CAL Data for Card 11 Block 5

Table 132: *Structure of the TC2D bank. Normally, there will be 12 card sections in each block, but this will not always be true.*

Displacement	**Data Description**
P0	Header word
P0+1	Energy word for Trigger Tower 0
P0+2	Tower usage word Trigger Tower 0
P0+3	Energy word for Trigger Tower 1
...	
P0+16	Tower usage word Trigger Tower 7

Table 133: *Format of the data section written into TC2D for each DCAS card. The towers appear in order of trigger tower number. For some cards it would be towers 8 to 15 or 16 to 23, instead of 0 to 7.*

Bit	Function
7:0	Bunch Counter
8:12	Geographical Address (Slot ID)
13:22	Serial Number
23:31	Module ID

Table 134: *TC2D header word definition. The header word is the first word in every card data section in the bank.*

Bit	Definition
31:26	unused
25:16	EM Energy
15:10	unused
9:0	HAD energy

Table 135: *TC2D energy word definition.*

Bit	Definition
31:8	unused
7	=1 if tower is SEED for threshold pass 3, 0 otherwise
6	=1 if tower is SEED for threshold pass 2, 0 otherwise
5	=1 if tower is SEED for threshold pass 1, 0 otherwise
4	=1 if tower is SEED for threshold pass 0, 0 otherwise
3	=1 if tower is CLUSTER MEMBER for thr. pass 3, 0 otherwise
2	=1 if tower is CLUSTER MEMBER for thr. pass 2, 0 otherwise
1	=1 if tower is CLUSTER MEMBER for thr. pass 1, 0 otherwise
0	=1 if tower is CLUSTER MEMBER for thr. pass 0, 0 otherwise

Table 136: *TC2D tower usage word definition.*

Block	Slot ID	Wedge	Trigger Towers
n	4	4n	16 to 23
n	5	4n	8 to 15
n	6	4n	0 to 7
n	9	4n+1	16 to 23
n	10	4n+1	8 to 15
n	11	4n+1	0 to 7
n	13	4n+2	16 to 23
n	14	4n+2	8 to 15
n	15	4n+2	0 to 7
n	18	4n+3	16 to 23
n	19	4n+3	8 to 15
n	20	4n+3	0 to 7

Table 137: *How to identify cards by block and slot in TC2D. The block number (n), can have any value from 0 to 5 and refers to the order the blocks appear in the bank. The slot ID is written into the card header word. Wedges are standard CDF wedges. The trigger towers are defined in detail in CDF note 2045.*

29.10 SVTD – SVT Interface Bank

SVT also sends a track list to an interface board in the Level 2 processor crate. Because SVT itself does not normally write data into the DAQ event stream, the only information about SVT in the data stream comes via the Level 2 crate. The interface card holds the data until requested by a processor. There are 7 words per track; however, only 21 bits are used in each word. The last word in the bank is used to report any errors found by SVT. The format of the track data is given in Table 139. The End Event word format is given in Table 141.

The structure of the SVTD bank is:

"SVTD"
Bank Number
Bank Version
Bank Length
Bank Type (I*4)
Number of Blocks (=1)
Pointer to Block 0
Pointer to End of Data
Number of Cards (=1)
Pointer to Card 0
Pointer to End of Card
Card 0 Header Word
Track 0, Word 0
...
Track 0, Word 6
...
Track $N-1$, Word 0
...
Track $N-1$, Word 6
End Event Word

Table 138: *Structure of the SVTD bank.*

Word	Bit	Function
0	12:0	ϕ
	15:13	z_{in}
	18:16	z_{out}
	19	0
	20	1
	22:21	0
	31:23	Not assigned
1	8:0	Absolute value of Impact parameter
	9	Sign of impact parameter
	17:10	Absolute value of Curvature (Pt) Bin
	18	Sign of Curvature
	22:19	0
	31:23	Not assigned
2	14:0	Road ID
	16:15	0
	20:17	ϕ sector
	22:21	0
	31:23	Not assigned
3	9:0	x_0 – Hit coordinates in layer 0
	19:10	x_1 – Hit coordinates in layer 1
	20	Spare
	22:21	0
	31:23	Not assigned
4	9:0	x_2 – Hit coordinates in layer 2
	19:10	x_3 – Hit coordinates in layer 3
	20	Spare
	22:21	0
	31:23	Not assigned
5	9:0	x_4 – Hit coordinates in layer 4
	20:10	χ^2 of fit
	22:21	0
	31:23	Not assigned
6	8:0	XFT linker ID
	20:9	Track-fitter status
	21	1 (End of Packet)
	22	0
	31:23	Not assigned

Table 139: *SVT track data block. Note that this format does not accomodate layer 00 and may need to be modified in the future.*

- The Card 0 Header Word contains the bunch crossing number in bits 0:7, copied from the End Event word.
- Bits 21 and 22 of the data words have the following meaning: bit 21 is 1 if the word is an End of Packet word (last word of a track data block); bit 22 is 1 if the word is an End Event word.
- ϕ in word 0 is ϕ of the track as computed by the TF. Units are approximately 0.767 mrad ($2\pi/8192$), the range is 0-6.28.
- z in word 0: $z_{out} = z$ ID of outer hit, z_{in} = same for inner hit.
- Word 1 has 9 bits of signed curvature (c) and 10 of signed impact parameter d, most significant bit is 1 to mean negative sign.
- The curvature is in units of (0.003/256) cm^{-1}. The range is $\pm$0.003 cm^{-1}.
- The impact parameter is in units of 10 μm. The range is $\pm$0.511 cm.
- The curvature sign bit is 1 for negative charged tracks.
- Word 2 contains SVT road-id and ϕ sector and is a copy of the ROAD word received from the AMS via the HB. Only the road ID bits that are actually used by the TF (15 LS bits) are passed on.
- Words 3, 4, and 5 contain the χ^2 of the fit and the coordinates of the hits in the 5 silicon layers (AM layers 0-4).
- The maximum value of the χ^2 is 1024. The MSB is the overflow bit.
- The SVX hit coordinates are reported in abbreviated form (8 Least Significant Bits of the coordinates). The full information (14 bits) can be reconstructed from this coordinate and the road number in word 2. The 2 MS bits of each x_i field are the long cluster flag (bit 8) and the hit existance flag (bit 9).
- The SVX his coordinates are in units of $\Delta s/16$, where Δs is the strip size. The strip size in the five SVX layers is 60, 62, 60, 60, 65 μm for layers 0-4 respectively.
- The XFT linker ID in the last word is simply the 9 most significant ϕ bits of the XFT track as delivered to SVT by XTRP i.e. the XFT wedge number. The size of the XFT wedge is 22 mrad. Since XFT only reports one track per wedge at most, this number uniquely associate the SVT track to the original XFT one.
- The meaning of the Track Fitter Status word is fully described in CDF Note 5026: "Hardware Design and Specification of the SVT Track Fitter", section 2.1. Bit allocation is reported in Table 140.

Bit	Function
3:0	Fit quality code
4	0
5	Hit overflow
6	Layer overflow
7	Combination overflow
8	Invalid data
9	Fitting result overflowed
10	FIFO overflow
11	Error summary (=OR of the above 6 bits)

Table 140: *Track fitter status word.*

Bit	Function
7:0	Event Tag – bunch crossing number
8	parity bit
9	parity error
10	lost synch error
11	FIFO overflow error
12	invalid data error
13	internal overflow error
14	truncated output error
15	G-link lost lock error
16	Parity error in cable to Level 2
18:17	Level 1 trigger info
20:19	Level 2 Buffer number
21	1 (End of Packet)
22	1 (End Event)
31:23	Unused

Table 141: *SVTD End Event word.*

- The L1 trigger info bits are a condensed summary of the L1 trigger decision. They are computed starting from the 64-bit L1 trigger word in the XTF-A board. The actual meaning of these two bits is described in the XFT documentation.
- The Parity bit is computed taking the bitwise XOR of all data words in the event (23 bits/word) excluding the End Event word.
- The End Event word is a packet by itself and therefore has EP=EE=1 (bits 21 and 22).

29.11 SVDD – SVT Internal Diagnostic Bank

Occasionally SVT crates will add internal diagnostic information into the DAQ event stream. This may be done in special runs or even in every run, but in the latter case with a very limited number of words. This information will be filled by the VME crate controllers either reading some VME physical registers (e.g. local error bits) or retrieving summary information from the on-line SVT monitor program. The SVDD bank is a flexible tool for subtle problem diagnosis by SVT experts. It is not for public usage. The meaning and format of each data block may change without notice at any time.

We will use the bank version number and the card header word to keep track of what each data block means. The bank version will be of the form:

$$\text{SVDD–version–number} = \text{Major–version} * 1024 + \text{Minor–version}$$

It is expected that offline code that reads the bank may have to change when the major version number changes, while changes in minor version number will have to be backward compatible (like defining a new type of card header).

The structure of the SVDD bank is shown in Table 142. The first word of each Card is a Header Word containing information that specifies the kind of data to follow. This information will be specified as needed during the SVT commissioning and operation.

"SVDD"
Bank Number
Bank Version Number
Bank Length
Bank Type (I*4)
Number of Blocks (=8)
Pointer to Block 0
Pointer to Block 1
...
Pointer to Block 7
Pointer to end-of-data
Block 0: Number of cards (=n0)
Pointer to Card 0
Pointer to Card 1
. . .
Pointer to Card (n0 - 1)
Pointer to end of block
Card 0, Word 0
. . .
Card (n0 - 1), Word 0
...
Card (n0 - 1), Last Word
Block 1: Number of cards (=n1)
Pointer to Card 0
...
Pointer to Card (n1 - 1)
Pointer to end of block
Card 0, Word 0
. . .
Card (n1 - 1), Last Word
...
...
Block 7: Number of cards (=n7)
Pointer to Card 0
...
Pointer to Card (n7 - 1)
Pointer end of block
Card 0, Word 0
. . .
Card (n7 - 1), Last Word

Table 142: *Structure of the SVDD bank.* There is one block for each SVT crate (8 crates). Each block will be filled with diagnostic information for use by SVT experts to debug particular specific problems and content will be defined at time of usage. The number of card sections in each block is variable.

29.12 T2PD - Muon Level 2 PULSAR Trigger Bank

The Level-2 trigger information from the Pulsar boards is stored in the T2PD bank. For the Pulsar based Level-2 muon trigger system, the muon data from the 12 matchbox fibers and 4 pre-match fibers are sent to a dedicated Muon Pulsar board for processing. In addition, the Muon Pulsar board will receive a copy of the XTRP data. The Muon Pulsar board performs zero-suppression on the raw muon data and sends the results downstream to the Level-2 global decision CPU along with the XTRP track list. The raw Level-1 muon trigger data received by the Pulsar board from the 16 hotlink fibers and the outputs of the Pulsar board (zero-suppressed muon data and XTRP trklist) are stored in the Level-2 buffers for DAQ readout. The format of the T2PD_StorableBank is given in Table 143, and the format within the cards is given in Tables 144, 145, 146, 147, 148, 149, 150, and 151.

"T2PD"
Bank Number
Bank Version
Bank Length (variable)
Bank Type (I*4)
Number of Blocks (=1)
Pointer to Block 0
Pointer to End of Data
Block 0: Number of Cards (=3)
Pointer to Card 0 (P0)
Pointer to Card 1 (P1)
Pointer to Card 2 (P2)
Pointer to End of Block
Card 0: Pulsar Summary Data
...
Card 1: XTRP TrkList Data
...
Card 2: Muon Matchbox/PreMatch Data
...

Table 143: Structure of the T2PD bank. The bank is composed of 32 bit data words.

29.12.1 Pulsar Summary Data

The Pulsar summary data contain a copy of the XTRP track list and the zero-suppressed muon data. The format of this data is described in Table 144. The first word is a header word whose bits are defined in Table 145. The format of the XTRP words is defined in Table 146. The format of the end of event word is defined in Table 147.

The T2PD muon words come in pairs. The format of these words is defined in Table 148. Each pair of T2PD muon words corresponds to one non-zero muon trigger word in the TCMD bank. The TCMD trigger words have their 24 lowest bits filled with trigger bits. In T2PD, bits 0:11 of muon word 0 contain the lowest 12 trigger bits from the corresponding TCMD trigger word. In T2PD, bits 0:11 of muon word 1 contain bits 12:23 of the same TCMD trigger word. See the description of the TCMD bank for a description of these bits (Section 29.5).

There is additional information in the T2PD muon words which can be used to determine which TCMD trigger word corresponds to the pair of T2PD words. There are two parts to this. First, one must determine the corresponding matchbox or prematch card. There are many TCMD words for each card. So second, one must determine which of these TCMD words corresponds to the pair of T2PD words.

The Pulsar FPGA number, mezzanine card number, and channel number can be used to calculate whether the data is from a prematch card or a matchbox card and the prematch number or matchbox number. Channel # 3 is always from a prematch card. The prematch card number is given by the expression: $(2 \times \mathrm{FPGA\#}) + (\mathrm{mezzanine\ card\#})$. Channels 0 to 2 are for matchbox cards. The matchbox number is given by the expression: $[(6 \times \mathrm{FPGA\#}) + (3 \times \mathrm{mezzanine\ card\#}) + \mathrm{channel\ \#}]$. The matchboxes are numbered sequentially in ϕ order starting with matchbox 0 covering ϕ from 0 to 30 degrees. The prematch cards are numbered as follows: bottom=0, south=1, top=2, and north=3.

For each card, there are many TCMD words. The tables in Section 29.5.1 and Section 29.5.2 describe these words. In the tables, there is an offset for each word from the beginning of the card. One can calculate these offsets using the "TCMD Word Number" which is stored in bits 12:16 of the T2PD muon data words. For matchbox data, the offset from the beginning of the TCMD card is $3 +$ "TCMD Word Number". For prematch data, the offset from the beginning of the TCMD card is $2 +$ "TCMD Word Number". For example if it is matchbox data, "TCMD Word 0" corresponds to "CMU East - High Pt Bits", and "TCMD Word 29" corresponds to "IMB West Diagnostic Bits". For example if it is the prematch, "TCMD Word 0" corresponds to "CMP Primitives for Stacks 00-23", "TCMD Word 15" corresponds to "CMP/CSP West Matches for Stacks 72-95".

Displacement	Data Description
P0	Header Word
P0+1	Number of Data Words ($N_{TRK} + 2 * N_\mu + 1$)
P0+2	Track 0 XTRP Word
...	...
P0+1+N_{TRK}	Track $N_{TRK} - 1$ XTRP Word
P0+2+N_{TRK}	Muon Pair 0 Word 0
P0+3+N_{TRK}	Muon Pair 0 Word 1
...	...
P0+N_{TRK}+2*N_μ	Muon Pair $N_\mu - 1$ Word 0
P0+1+N_{TRK}+2*N_μ	Muon Pair $N_\mu - 1$ Word 1
P0+2+N_{TRK}+2*N_μ	End of Event Word

Table 144: Format of Card 0. N_{TRK} is the number of XTRP tracks. N_μ is the number of pairs of muon data words.

Bit	Definition
7:0	Bunch Counter
12:8	Geographic Address (slot #)
22:13	Serial Number
31:23	Board Type

Table 145: Header word for the Pulsar Summary Data card.

Bit	Definition
11:0	Track ϕ
18:12	Track Curvature Bin
19	Not Used
20	SL6/SL8 Bit (Short Track Bit)
21	EP Bit (always 1)
22	EE Bit (always 0)
31..23	Not Used

Table 146: Bit assignments for the card 0 XTRP data words.

Bit	Definition
18:0	Reserved for Error Flags and Event ID Tag
20:19	Level-2 Buffer Number
21	EP Bit = 1
22	EE Bit = 1
31..23	Not Used

Table 147: Bit assignments for Card 0 End of Event Word.

Bit	Word 0	Word 1
11..0	Trigger Bits	Trigger Bits
12..16	TCMD Word Number (0-29 for matchbox data, 0-15 for pre-matchbox data)	(Same as word 0)
17..18	Pulsar Channel # (0-3)	(Same as word 0)
19	Pulsar Mezzanine card # (0-1)	(Same as word 0)
20	Pulsar FPGA # (0-1)	(Same as word 0)
21	EP Bit (always 0)	EP Bit (always 0)
22	EE bit (always 0)	EE bit (always 0)
31..23	Not Used	Not Used

Table 148: Bit assignments for card 0 zero-suppressed muon data words. See text for more details.

29.12.2 XTRP TrkList Card

Card 1 contains a copy of the XFT track data sent to the Pulsar board from the Level-1 XTRP trigger. The format of card 1 is shown in Table 149. The XTRP data words and end of event word have the same format as in card 0.

Displacement	Data Description
P1	Number of Data Words (N_{TRK}+1)
P1+1	Reserved
P1+2	Track 0 XTRP Data
...	...
P1+1+N_{TRK}	Track N_{TRK} - 1 XTRP Data
P1+2+N_{TRK}	End of Event Word

Table 149: Format of card 1. The bit assignments for the XTRP Data Word and the End of Event Word are described in Table 146 and Table 147.

29.12.3 Muon Matchbox/PreMatch Card

Card 2 contains the muon trigger data sent to the Pulsar board from the Level-1 muon trigger. The data content of card 2 is essentially a copy of the TCMD bank block 0 and 1. However, there are several differences.

In the TCMD bank, the words contain trigger bits in bits 0:23 and the remaining bits are not used. In card 2 of T2PD, the words contain the same 24 bits, but 8 control bits are inserted in between these bits. The trigger bits are shifted up as necessary to make room for the control bits. The control bits are inserted at bits 7, 15, 23, 27, 28, 29, 30, and 31. These control bits are used for checking transmission errors.

In TCMD, there is a separate card section in the bank for each matchbox and prematch card. In T2PD, all the words are put into one card (card 2). 30 sequential words are written into the T2PD card 2, for each matchbox or prematch card. Table 150 specifies the 30 words written into card 2 for each matchbox card. Table 151 specifies the 30 words written into card 2 for each prematch card. Other than the control bits and related shifting these words are identical to the words written into TCMD as described in Section 29.5.1 and Section 29.5.2.

The 30 word groups are written into card 2 in order of the fiber number. The tables give the offset in card 2 where n is the number of the fiber used to send data from the prematch or matchbox card to the Pulsar board. The fibers are numbered as follows:

$$\text{fiber number} = (8 \times \text{FPGA \#}) + (4 \times \text{mezzanine card \#}) + \text{channel \#}.$$

Displacement	**Data Description**
P2+(30×n)	CMU East - High Pt Bits
P2+(30×n)+1	CMU East - Low Pt Bits
P2+(30×n)+2	CMU West - High Pt Bits
P2+(30×n)+3	CMU West - Low Pt Bits
P2+(30×n)+4	CMX East - High Pt Bits
P2+(30×n)+5	CMX East - Low Pt Bits
P2+(30×n)+6	CMX West - High Pt Bits
P2+(30×n)+7	CMX West - Low Pt Bits
P2+(30×n)+8	CSX and HAD East Bits
P2+(30×n)+9	CSX and HAD West Bits
P2+(30×n)+10	BMU East - High Pt Bits
P2+(30×n)+11	BMU East - Low Pt Bits
P2+(30×n)+12	BMU West - High Pt Bits
P2+(30×n)+13	BMU West - Low Pt Bits
P2+(30×n)+14	BSU East Bits
P2+(30×n)+15	BSU West Bits
P2+(30×n)+16	TSU and HAD East Bits
P2+(30×n)+17	TSU and HAD West Bits
P2+(30×n)+18	Spare Word A
P2+(30×n)+19	Spare Word B
P2+(30×n)+20	TOF Bits
P2+(30×n)+21	CMU East Diagnostic Bits
P2+(30×n)+22	CMU West Diagnostic Bits
P2+(30×n)+23	Eta Gap Diagnostic Bits
P2+(30×n)+24	CMX East Diagnostic Bits
P2+(30×n)+25	CMX West Diagnostic Bits
P2+(30×n)+26	IMA East Diagnostic Bits
P2+(30×n)+27	IMA West Diagnostic Bits
P2+(30×n)+28	IMB East Diagnostic Bits
P2+(30×n)+29	IMB West Diagnostic Bits

Table 150: Format of 30 word group for matchbox trigger data stored in Card 2 of T2PD. P2 is the pointer to the beginnning of card 2, and n is the fiber number (see text).

Displacement	**Data Description**
P2+(30×n)	CMP Primitives for Stacks 00-23
P2+(30×n)+1	CMP Primitives for Stacks 24-47
P2+(30×n)+2	CMP Primitives for Stacks 47-71
P2+(30×n)+3	CMP Primitives for Stacks 72-95
P2+(30×n)+4	CSP East Bits for counters 00-23
P2+(30×n)+5	CSP East Bits for counters 24-47
P2+(30×n)+6	CSP West Bits for counters 00-23
P2+(30×n)+7	CSP West Bits for counters 24-47
P2+(30×n)+8	CMP/CSP East Matches for Stacks 00-23
P2+(30×n)+9	CMP/CSP East Matches for Stacks 24-47
P2+(30×n)+10	CMP/CSP East Matches for Stacks 48-71
P2+(30×n)+11	CMP/CSP East Matches for Stacks 72-95
P2+(30×n)+12	CMP/CSP West Matches for Stacks 00-23
P2+(30×n)+13	CMP/CSP West Matches for Stacks 24-47
P2+(30×n)+14	CMP/CSP West Matches for Stacks 48-71
P2+(30×n)+15	CMP/CSP West Matches for Stacks 72-95
P2+(30×n)+16	Reserved
...	...
P2+(30×n)+30	Reserved

Table 151: Format of 30 word group for prematch trigger data stored in Card 2 of T2PD. P2 is the pointer to the beginnning of card 2, and n is the fiber number (see text).

29.13 TL2D - Level 2 Trigger Bank

The format of the TL2D bank is given in Table 152. The format of the data for each card is specified in the following sections.

"TL2D"
Bank Number
Bank Version
Bank Length (variable)
Bank Type (I*4)
Number of Blocks (=1)
Pointer to Block 0
Pointer to End of Data
Number of Cards (=12)
Pointer to L1 Trigger Card (=P0)
Pointer to L1 Scaler Card (=P1)
Pointer to MUON Card (=P2)
Pointer to XTRP Track List Card (=P3)
Pointer to SVT Track List Card (=P4)
Pointer to Cluster List Card (=P5)
Pointer to Isolation Cluster List Card (=P6)
Pointer to XCES Hits Card (=P7)
Pointer to MUON Decision Card (=P8)
Pointer to L2 Trigger Card (=P9)
Pointer to L2 Scalar Card (=P10)
Pointer to L2 Diagnostic Card (=P11)
Pointer to End of Block

Table 152: *Structure of the TL2D bank.*

29.13.1 L1 Trigger Card

The L1 Trigger card section has the following format:

Displacement	**Data Description**
P0	L2 Buffer Number (31:16), Number of L1 Trigger Words (=3) (15:0)
P0+1	Level 1 Decision bits 0-31
P0+2	Level 1 Decision bits 32-63
P0+3	ΣE_T Data

The ΣE_T word is defined as:

Bit	Definition
31	Unused
30:21	$(\Sigma E_T)_y$
20:11	$(\Sigma E_T)_x$
10:0	ΣE_T

29.13.2 L1 Scaler Card

The structure of the L1 Scaler card section is:

Displacement	Data Description
P1	Number of L1 triggers in Scaler card, N (variable)
P1+1	Event count for L1 trigger bit 0
...	....
P1+N	Event count for L1 trigger bit N-1

29.13.3 Muon Header Card

The structure of the Muon card section is:

Displacement	Data Description
P2	Number of MUON Flag Words (=4)
P2+1	Muon Matchbox Mask $0° < \phi < 120°$
P2+2	Muon Matchbox Mask $120° < \phi < 240°$
P2+3	Muon Matchbox Mask $240° < \phi < 360°$
P2+4	Muon Pre-Match Mask

In normal Run Mode, only the 128-bit mask loaded into the L2 Processor to indicate the presence of non-zero muon trigger data for a given event will be included in the output stream. The mask is made of four bits of information for each 15° azimuthal detector slice (24 total). One bit indicates the presence of central detector data, a second indicates the presence of IMU data, and a third indicates the presence of L1 trigger data. The fourth bit is unused. These are followed by mask bits for the muon Pre-match board. There are 4 bits each for the Top, North Wall, Bottom and South Wall.

29.13.4 XTRP Card

The structure of the XTRP card section is:

Displacement	Data Description
P3	Number of XTRP Tracks N_{TRK} (variable)
P3+1	Track 1 XTRP Data
P3+2	Track 1 Match Data
P3+3	Track 2 XTRP Data
P3+4	Track 2 Match Data
...	...
P3+2*N_{TRK}-1	Track N_{TRK} XTRP Data
P3+2*N_{TRK}	Track N_{TRK} Match Data
P3+2*N_{TRK}+1	Track End of Event Word

Each track has two words associated with it, the XTRP and Match words. The XTRP word is defined as:

XTRP Bit	Definition
31:22	Unused
21	Spare bit
20	SL6/SL8 bit (Short track bit)
19	Isolation bit
18:12	Track curvature bin
11:0	Track phi

The last bit field in the word above is “Track phi”. This gives the XFT bin of the track. These bins are numbered from 0 to 2303 in order of global phi at superlayer 6 of the COT. (I do not know whether bin 0 is centered where phi equals 0 or the lower bin edge is where phi equals 0 or something else???) If one right shifts the “Track phi” value by 3 bits (or equivalently divide by 8 and round down), then one gets the “XFT Linker ID” that identifies the XFT chip. Since an XFT chip outputs at most one track per event, this uniquely identifies an XFT track within the event. The “XFT Linker ID” runs from 0 to 287.

The Match word is defined as:

Match Bit	Definition
31:30	Unused
29:22	XCES East/West Low/High z Low/High Threshold
21:20	CMP East/West
19:16	Phi/Eta East/West Gaps
15:12	IMB East/West Low/High P_T
11:8	IMA East/West Low/High P_T
7:4	CMX East/West Low/High P_T
3:0	CMU East/West Low/High P_T

The Track End of Event word is defined as:

Match Bit	Definition
31:23	Unused
22	End of Event bit
21	Unused
20:19	L2 Buffer number
18:9	Error flags
8	Reserved
7:0	Event Tag

The muon/electron matching data word are produced by the L2 Processor itself based on the presence of any hit in the outer detectors encountered by a track while it escapes from the interaction point.

29.13.5 SVT Card

The SVT card section is defined as:

Displacement	**Data Description**
P4	Number of SVT Tracks N_{SVT} (variable)
P4+1	Track 1, Word 1
P4+2	Track 1, Word 2
P4+3	Track 2, Word 1
...	...
P4+2*N_{SVT}-1	Track N_{SVT}, Word 1
P4+2*N_{SVT}	Track N_{SVT}, Word 2
P4+2*N_{SVT}+1	SVT Error Word (see table 141)

Each track has two associated words which are a subset of the information from SVTD and are defined as:

Word	Bit	Function
1	12:0	ϕ
	15:13	z_{in}
	18:16	z_{out}
	27:19	Absolute Value of Impact Parameter
	28	Sign of Impact Parameter
	31:29	Not assigned
2	7:0	Absolute Value of Curvature (Pt) Bin
	8	Sign of Curvature (Pt) bin
	19:9	χ^2 of fit
	28:20	XFT Linker ID
	29	Track-fitter error summary
	31:30	Not assigned

These bit fields have exactly the same meaning and hold the same values as the corresponding bit fields in the SVTD bank. See Section 29.10 and Table 139 for a more detailed description of their contents. Bit 29 of the second word is the same as bit 11 in the error summary word defined in Table 140. The “XFT Linker ID” is also the same as the top 9 bits of “Track phi” bitfield in the XTRP card of the TL2D bank (see the previous section).

29.13.6 Cluster Card

The Cluster card section is defined as:

Displacement	**Data Description**
P5	Number of Clusters N_{CLS} (variable)
P5+1	Cluster 1 Part-1 Data
P5+2	Cluster 1 Part-2 Data
P5+3	Cluster 2 Part-1 Data
...	...
P5+2*N_{CLS}-1	Cluster N_{CLS} Part-1 Data
P5+2*N_{CLS}	Cluster N_{CLS} Part-2 Data

Each cluster has two words which are specified as

Word	Bit	Definition
1	0:11	EM Energy Sum
	12:14	Spare
	15	EM Energy Overflow Bit
	16:27	Hadronic Energy Sum
	28:30	Spare
	31	Hadronic Energy Overflow Bit
2	0:9	Number of Towers in Cluster
	10:14	Cluster ϕ
	15:19	Cluster η
	20:21	L2 Buffer
	22:23	Clustering Threshold Pass
	24:30	Unused
	31	Bit is 1 for valid clusters

29.13.7 Isolation Cluster Card

The Isolation Cluster card section contains the six 15-bit energy sums per cluster made by the isolation board. The sums are for total energy and have the 0.125 GeV least count of the Level 2 clustering. The card section is specified as:

Displacement	**Data Description**
P6	Number of Isolation Clusters N_{ISO} (variable)
P6+1	Cluster 1 Part-1 Data
P6+2	Cluster 1 Part-2 Data
P6+3	Cluster 1 Part-3 Data
P6+4	Cluster 1 Part-4 Data
P6+5	Cluster 2 Part-1 Data
...	...
P6+4*N_{ISO}-1	Cluster N_{ISO} Part-3 Data
P6+4*N_{ISO}	Cluster N_{ISO} Part-4 Data

The four words associated with each isolation cluster are defined as:

Word	Bit	Definition
1	0:1	Clustering Pass
	2:6	ϕ
	7	Data Bit (1 = Diagnostic, 0 = Real Data)
	8:9	Buffer Number
	10:14	η
	15	Spare
	16:20	Overflow bits for the 5 Et Sums (bit 16 is for Et Sum 1)
	21:22	Spare
	23	HiLow, 0 = 0:4 sums, 1 = 5:9 sums
	24:31	Spare
2	0:15	Et Sum 1
	16:31	Et Sum 2
3	0:15	Et Sum 3
	16:31	Et Sum 4
4	0:15	Et Sum 5
	16:31	Spare

29.13.8 XCES Trigger Hits Card

The XCES card section has fixed size (49 words) and the following format:

Displacement	Data Description
P7	Number of XCES Trigger Channels (=48)
P7+1	Trigger bits for wedge west phi=0
P7+2	Trigger bits for wedge west phi=1
...	....
P7+24	Trigger bits for wedge west phi=23
P7+25	Trigger bits for wedge east phi=0
...	....
P7+48	Trigger bits for wedge east phi=23

The bit assignment of the words is:

high η module $\vert z\vert > 121$ cm		
Bit	wire	threshold
0	0–3	low
1	0–3	high
2	4–7	low
3	4–7	high
4	8–11	low
5	8–11	high
6	12–15	low
7	12–15	high
8	16–19	low
9	16–19	high
10	20–23	low
11	20–23	high
12	24–27	low
13	24–27	high
14	28–31	low
15	28–31	high

low η module $\vert z\vert < 121$ cm		
Bit	wire	threshold
16	0–3	low
17	0–3	high
18	4–7	low
19	4–7	high
20	8–11	low
21	8–11	high
22	12–15	low
23	12–15	high
24	16–19	low
25	16–19	high
26	20–23	low
27	20–23	high
28	24–27	low
29	24–27	high
30	28–31	low
31	28–31	high

If the ADC sum of the 4 CES wires exceeds the threshold (high and low) the bit is set to 1.

29.13.9 Muon Decision Data Card → Pulsar Muon Data Card

The Pulsar muon data card contains Level-1 muon data sent from the Pulsar board to the global Level-2 decision processor.

Displacement	Data Description
P8	Number of Pairs of Muon Words (N_μ)
P8+1	Muon Pair 0 Word 0
P8+2	Muon Pair 0 Word 1
P8+3	Muon Pair 1 Word 0
P8+4	Muon Pair 1 Word 1
...	...
P8+2*N_μ - 1	Muon Pair $N_\mu - 1$ Word 0
P8+2*N_μ	Muon Pair $N_\mu - 1$ Word 1

Table 153: L2 Pulsar muon data card. The bit assignment for the muon data words is the same as in card 0 of the T2PD bank (see Section 29.12.1, Table 144 and Table 148)

29.13.10 L2 Trigger Card

The L2 Trigger Card section is specified as:

Displacement	Data Description
P9	Number of L2 Triggers N_{L2} (variable)
P9+1	bit 0-31 L2 trigger mask
P9+2	bit 32-63 L2 trigger mask
P9+3	bit 64-95 L2 trigger mask
P9+4	bit 96-127 L2 trigger mask
P9+5	bit 0-31 L2 trigger mask before prescale
P9+6	bit 32-63 L2 trigger mask before prescale
P9+7	bit 64-95 L2 trigger mask before prescale
P9+8	bit 96-127 L2 trigger mask before prescale
P9+9+0	L2 Trigger Word 0
P9+9+1	L2 Trigger Word 1
...	...
P9+9+N_{L2} - 1	L2 Trigger Word N_{L2} - 1

Table 154: *The L2 Trigger Card.*

The contents of the L2 Trigger Words are not defined yet.

29.13.11 L2 Scaler Card

The L2 Scaler card section is specified as:

Displacement	Data Description
P10	Number of L2 Triggers N_{L2} (variable)
P10+1	Prescaling factor for L2 trigger 1
...	...
P10+N_{L2}	Prescaling factor for L2 trigger N_{L2}
P10+N_{L2}+1	Number of events passing prescaled trigger 1
...	...
P10+2*N_{L2}	Number of events passing prescaled trigger N_{L2}
P10+2*N_{L2}+1	Remainder of L2 trigger 1 inputs to prescale
...	...
P10+3*N_{L2}	Remainder of L2 trigger N_{L2} inputs to prescale
P10+3*N_{L2}+1	Count of L2 trigger 1 before prescale
...	...
P10+4*N_{L2}	Count of L2 trigger N_{L2} before prescale

Table 155: *The L2 Scaler Card.*

where the number of L2 triggers and their prescaling factors are as specified in the Trigger Table for the given run.

29.13.12 L2 Diagnostic Card

The contents of this card section are not defined yet.

Displacement	Data Description
P11	Number of L2 Diagnostic Words N_{DIAG} (variable)
P11 + 1 to P11 + N_{DIAG}	Not Specified Yet

Table 156: *The L2 Diagnostic Card.*

29.14 INHD - Trigger Inhibit Bank

The INHD bank contains information about the Trigger Inhibit system. The format of the bank is specified in Table 157. The bank is organized in standard block format, currently with one block, $N = 1$. We reserve the right to add additional blocks at a later time for additional trigger inhibit related information, but the Block 0 format shall not change.

The *Inhibit Enable Mask* is a bit mask of source components that are allowed to contribute to the global trigger inhibit signal. When a bit is set to one in the *Inhibit Enable Mask* field, then the corresponding component was a possible source of global trigger inhibits at that particular time during the run.

Table 158 lists the mapping from bit to source component as of 17 July 2001. Note that not all bits are used. The actual mapping used by the DAQ system is found in the *Inhibits* table of the Hardware Database (see reference [13]), and may change at any time. The dynamically constructed value of the *Inhibit Enable Mask* will depend on the detector components included in the run and the instantaneous status of a particular source component. Changes in the mapping should generally be limited to the addition of new sources of inhibit signals. However, any changes in the mapping of the bits can be found in the *InhibitUpdates* table (also in the hardware database [13]) if they occur.

The *Inhibit Signals Mask* represents the state of a particular source component's inhibit signal at event readout time. The same bit mapping from table 158 applies to this mask. When a bit is set to one, then the corresponding component is requesting that triggers be inhibited. However, this mask data is read *before* the enable mask is applied, so this particular component may not have actually caused the DAQ system to stop taking triggers.

The I*4 format of the bank allows up to 32 sources of trigger inhibit, for future expansion. Due to hardware limitations, expansion beyond 16 bits is unlikely.

Table 158 also indicates the *Global* flag for each source. When *true*, these inhibit sources affect all data taking, regardless of components booked in a run. When *false*, the inhibit sources map onto selectively bookable detector components. The *Inhibit* column of the *Components* table in the hardware database [13] maps readout (bank producing) components with inhibit components.

"INHD"	
Bank Number	
Bank Version	
Bank Length	
Bank Type (I*4)	
Number of blocks ($= N = 1$)	
Pointer to Block 0 ($= P0$)	
...	
Pointer to Block N-1	
Pointer to End of Data	
$P0$	Inhibit Enable Mask
$P0 + 1$	Inhibit Signals Mask

Table 157: *Structure the Trigger Inhibit bank INHD*

Bit	Name	Description	Global
0	BFLD	Solenoid	true
1	SVX	Silicon Tracking Detector	false
2	ISL	Intermediate Silicon Tracking	false
3	COT	Central Outer Tracker	false
4	TOF	Time-of-Flight	false
5	CCAL	Central Calorimeter	false
6	PCAL	Plug Calorimeter	false
7	CMU	Central Muon (CMU,CMX,CMP)	false
8	IMU	Intermediate Muon	false
9	CLC	Cerenkov Luminosity Counter	false
10	TEV	Tevatron Events	true
12	RACKS	Rack Protection	true
14	VME	VME Power Supplies	true

Table 158: Trigger Inhibit bit mapping as of 17 July 2001. IMPORTANT: This bit mapping will change and one must consult the hardware database to get the mapping used at the time any particular bank was created. The last column indicates whether a source is associated with a specific detector component or can only be associated with the entire detector.

29.15 TFTD - TOF Trigger Diagnostic Bank

TFTD
Bank Number
Bank Version
Bank Length
Bank Type
Number of Blocks (=1)
Pointer to Block 0
Pointer to End of Data
Number of Cards (=4)
Pointer to Card 0
Pointer to Card 1
Pointer to Card 2
Pointer to Card 3
Pointer to End of Block
TOTRIB 0 Header
TOTRIB 0 Word 0
...
TOTRIB 0 Word 9
...
TOTRIB 3 Header
TOTRIB 3 Word 0
...
TOTRIB 3 Word 9

Table 159: TFTD Storable Bank Description

The Time-of-Flight (TOF) L1 Trigger uses charge measurements from the TOF system to provide a highly ionizing particle (HIP) trigger, a high multiplicity minimum ionizing particle (MIP) trigger, and a cosmic trigger.

The TOF trigger is instrumented in pairs of bars. Since there is significant attenuation in the TOF bars, a hit close to a single PMT is always large. Requiring a large pulse in both the east and west of a pair of TOF bars leads to a more uniform z response. The TOF Trigger Board (TOTRIB) makes this coincidence calculation. The TFTD bank contains the TOTRIB data, which is useful for monitoring and diagnosing the trigger.

The TFTD bank format is described in Table 159. One crate containing four TOTRIBs instruments the entire detector, so there is one block containing four cards in the TFTD bank. Each TOTRIB L2 buffer is eleven words long. The first word is a header, with fields specified in Table 160. The bunch counter is not specified since the TOTRIB cannot keep track of it. The remaining 10 words are data. (12/19/02 - the header has not yet been implemented in firmware.)

The TOTRIB contains four FPGAs: a VME unit handling the VME interface and three coincidence units (CUs) handling the coincidence calculations. Because the connection

bits	description
0-7	0
8-15	Slot
16-23	Serial Number
24-31	Module ID

Table 160: TOTRIB card header

bits	description
0-7	CU 0 (i*8)th - (i*8+7)th bit
8-15	CU 1 (i*8)th - (i*8+7)th bit
16-23	CU 2 (i*8)th - (i*8+7)th bit
24-31	unused

Table 161: TOTRIB ith word

between the CUs and the VME interface is slow, one third of the L2 buffer is stored on each CU. As each word is readout, each CU is polled in turn for its next 8 bits of data. So each non-header word of the TFTD bank contains 8 bits from each CU and 8 unused bits, as shown in Table 161.

bits	num	description
0-7	8 bits	sum
8-11	4 bits	hip pattern
12-15	4 bits	mip pattern
16-35	20 bits	mip/hip coincidence
36-55	20 bits	mip/hip east
56-75	20 bits	mip/hip west
76-79	4 bits	unused

Table 162: TOTRIB coincidence unit data

Each CU provides 80 bits to the level 2 buffer, containing the input to the TOTRIB, the coincidence output, the sum output, and the pattern output. There is one HIP bit and one MIP bit of input for each pair of bars from both the east and west of the detector, as described in CDF note 5724. If a pulse has passed the MIP or HIP threshold, the corresponding bit is high. The coincidence output is just the logical *and* of these bits. The sum is calculated by adding the MIP coincidences. Finally, coarser resolution scatter patterns are calculated from the coincidence output. The arrangement of this data is shown in Tables 162, 163, and 164.

bits	num	description
0 -4	5 bits	hip1
5 -9	5 bits	mip1
10-14	5 bits	hip0
15-19	5 bits	mip0

Table 163: mip/hip coincidence data

bits	num	description
0 -4	5 bits	mip1
5 -9	5 bits	hip1
10-14	5 bits	mip0
15-19	5 bits	hip0

Table 164: mip/hip east/west data

30 TMPD - DAQ Diagnostic Bank

The TMPD bank holds information useful for DAQ troubleshooting and optimization. It has a structure similar to a TDC bank. Most of the format details for TDC banks are specified in Section 9.2. There is one block for each crate in the system, and the block number corresponds to the Reset Channel number +1. Which blocks are present in a given run or event is arbitrary.

A data block coming from a crate has the format specified in Table 165. The format of the data may and will change, but we will always try to append additional data at the end. The number of data sections may increase from its current value of 1.

- The details of the TRACER event header register are described in Table 166.
- Since a bunch counter mismatch between cards in a crate is detected by the readout software, and in general halts data taking, the bunch counter in the bank should be valid for all cards in the crate.
- Various (currently 4) times that characterize the performance of the readout and DAQ in general are contained in the bank. All times are measured for the *previous* event.
- The error code can be valuable in diagnosing problems; its possible values are specified in readout code header files. Since some errors can only be detected very late in the readout process, and thus only be put in the TMPD bank for the next event, the event number corresponding to an error is also put into the bank.

Besides DAQ diagnostics, the TMPD bank is also used to send fake data through the system. An arbitrary number of words can be present, composed of repeated patterns of 0xffffffff, 0xa5a5a5a5, 0x5a5a5a5a, 0x00000000, 0xffffffff, etc.

Number of data sections (= 1)
Pointer to first data section
...
Pointer to last data section
Pointer to End of Block
IP address of the crate processor
TRACER event header register
Bunch counter from one of the cards that are read out
Readout time [ns]: ISR call → r/o (including TDCs + formatting) finished
Readout time [ns]: ISR call → after all r/o and formatting+output done
Readout time [ns]: ISR call → r/o except TDCs / SVT-monitoring finished
Time [ns]: Previous event completely finished → ISR call
Size of data from this crate, in Bytes
Error code of last error
Event number the error code corresponds to
Test pattern (any length)
(0xffffffff 0xa5a5a5a5 0x5a5a5a5a 0x00000000 ...)
...

Table 165: *Structure of one block of data coming from one Front End Crate.*

Bit	TRACER event header
31:25	Calibration number (from TS)
24:22	Scan list number (from TS)
21:20	L2 buffer number (from TS)
19:16	Event ID (from TS)
15:11	Undefined
10:8	Partition number
7:0	Crate ID (= reset channel +1)

Table 166: *Bit assignment for the TRACER event header word. More details can be found in the TRACER and TSI documentation.*

31 Calibration and Pedestal Banks

The calibration/pedestal banks have one common feature that is a little ugly. The type string in the YBOS header indicates that these are all I*4 monotype banks (filled with only 32 bit integers). Actually, they contain both I*4's and R*4's (32 bit integers and 32 bit float's). This should cause no problems on any platforms currently supported for run 2, although it would cause problems if there was a need to return to VMS or use some new machine architecture that had a different representation of float's and int's (probably never). No one knows quite where this started, but it is everywhere in the calibration banks and would be difficult to fix.

31.1 Calorimeter Calibration Banks

31.1.1 Channel ID and Other Common Features

The bit assignment for the Channel ID is specified in Table 167. This applies to CEMC, CHAC, WHAC, PEMC, and PHAC banks. There are five bit fields which define the crate number, slot number, QIE number, capacitor number, and current range.

Each bank is filled with data from crates. For the CEMC and CHAC, there are 16 crates. For the WHAC, there are 8 crates. For the PEMC and PHAC, there are 12 crates. The crates are numbered sequentially starting at 1. The data from each crate are written into 1 block in the bank. The blocks are numbered sequentially starting from 0 (block number equals crate number minus one). Each crate holds ADMEM cards. Each card has QIE chips. The number of cards per crate and chips per card is detector dependent. These numbers and the map between the "electronic address" and the "geometric address" are described below for each bank individually. Each QIE chip has four capacitors and there are eight current ranges per QIE chip. Therefore, there are 32 channel ID's associated with each QIE chip.

These banks may be written out with some channels missing. The order channels are written into the bank is not specified.

These banks have a block pointer section as described in Section 4.1.

The bank format is specified in Table 168. For each data element, five 32 bits words are written in to the bank. There is an integer that is a channel ID. Then there are four floats. A block of data contains repeated occurences of the five word data elements and nothing else. It is the same for all five bank classes.

Bit	Channel ID
31:24	Crate Number
23:16	Slot Number
15:8	QIE Number
7:4	Capacitor Number (0:3)
3:0	Current Range (0:7)

Table 167: *Bit assignment for the Calorimeter Calibration Channel ID (CEMC, CHAC, WHAC, PEMC, and PHAC).*

"CEMC" or "CHAC" or ...
Bank Number
Bank Version
Bank Length
Bank Type (=3)
Number of Blocks (=n)
Pointer to Block 0
...
Pointer to Block (n - 1)
Pointer to End of Data
Channel ID
Slope
Offset
Slope Error
Offset Error
Channel ID
Slope
Offset
Slope Error
Offset Error
...

Table 168: *Structure of the calorimeter calibration banks (CEMC, CHAC, WHAC, PEMC, PHAC).*

31.1.2 CEMC - Central Electromagnetic Calorimeter Calibration Data

See Section 31.1.1 for a description of the bank format and channel IDs. The map from the "electronic address" [crate, slot, QIE number] to the "geometric address" has not been documented yet.

31.1.3 CHAC - Central Hadronic Calorimeter Calibration Data

See Section 31.1.1 for a description of the bank format and channel IDs. The map from the "electronic address" [crate, slot, QIE number] to the "geometric address" has not been documented yet.

31.1.4 WHAC - Wall Hadronic Calorimeter Calibration Data

See Section 31.1.1 for a description of the bank format and channel IDs. The map from the "electronic address" [crate, slot, QIE number] to the "geometric address" has not been documented yet.

31.1.5 PEMC - Plug Electromagnetic Calorimeter Calibration Data

See Section 31.1.1 for a description of the bank format and channel IDs. The map from the "electronic address" [crate, slot, QIE number] to the "geometric address" has not been documented yet.

31.1.6 PHAC - Plug Hadronic Calorimeter Calibration Data

See Section 31.1.1 for a description of the bank format and channel IDs. The map from the “electronic address” [crate, slot, QIE number] to the “geometric address” has not been documented yet.

31.2 Calorimeter Pedestal Banks (CEMP, CHAP, PEMP, PHAP, WHAP)

The QIE pedestal banks are used to store raw pedestal information for the capacitors of all the QIE channels. Together with the QIE calibration information (see Section 31.1), these data are used to calculate the frontend lookup tables. The raw data banks, *e.g.* CEMD, already have the lookup table correction applied.

The calorimeter QIE pedestal banks are:

- **CEMP:** Central EM Calorimeter Pedestal Data.
- **CHAP:** Central Hadronic Calorimeter Pedestal Data.
- **PEMP:** Plug EM Calorimeter Pedestal Data.
- **PHAP:** Plug Hadronic Calorimeter Pedestal Data .
- **WHAP:** Wall Hadronic Calorimeter Pedestal Data.

The data format is shown in Table 169. All five banks have identical structure. There is one data element per capacitor and four capacitors per QIE chip. Therefore, there are four data elements per QIE channel. For each data element, there are three 32-bit words. They are the channel ID, pedestal value, and the RMS of the pedestal. The first one is an integer and the other two are floats. A block of data contains repeated occurrences of the three-word data elements.

The bit assignment for the Channel ID of the pedestal bank is the same as the one for the corresponding calibration bank, except bits 0-3 are not used (see Table 170). The map between the “electronic address” [crate, slot, QIE chip] and the “geometric address” is identical to the map for the corresponding calibration banks (see Section 31.1).

These banks may be written out with some channels missing. The order channels are written into the bank is not specified.

These banks have a block pointer section as described in Section 4.1.

Bank Name (“CEMP” or “CHAP” or . . .)
Bank Number
Bank Version
Bank Length
Bank Type (=3)
Number of Blocks (=n)
Pointer to Block 0
...
Pointer to Block (n - 1)
Pointer to End of Data
Channel ID
Pedestal Mean
RMS of the Pedestal
Channel ID
Pedestal Mean
RMS of the Pedestal
...

Table 169: Structure of calorimeter QIE pedestal banks (CEMP, CHAP, PEMP, PHAP, WHAP).

Bit	Channel ID
31:24	Crate Number
23:16	Slot Number
15:8	QIE Number
7:4	Capacitor Number (0:3)
3:0	Not Used

Table 170: Bit assignment for the calorimeter pedestal bank channel ID.

31.3 CLAC, CLAP - Cherenkov Luminosity Counter Calibration Banks

The format of the CLAC bank is very similar to the format of the calorimeter calibration banks. Everything in Section 31.1.1 also applies to the CLAC bank. The bitfields in the channel ID are identical (see Table 167). The data words are also identical (see Table 168). The only difference is that there is only one block in the CLAC bank.

The format of the CLAP bank is very similar to the format of the calorimeter pedestal banks. Everything in Section 31.2 also applies to the CLAP bank. The bitfields in the channel ID are identical (see Table 170). The data words are also identical (see Table 169). The only difference is that there is only one block in the CLAP bank.

There is general information regarding the Cherenkov Luminosity Counter in Section 26. The geometry, location, and numbering of the photomultipliers are described in that section. The map from the "electronic addresses" in the CLAC/CLAP channel IDs to the "geometric addresses" is not documented yet. This map is the same for CLAC and CLAP.

31.4 TOFC, TOFP - Time of Flight Calibration Banks

The format of the TOFC bank is very similar to the format of the calorimeter calibration banks. Everything in Section 31.1.1 also applies to the TOFC bank. The bitfields in the channel ID are identical (see Table 167). The data words are also identical (see Table 168). The only difference is that there are eight blocks in the TOFC bank.

The format of the TOFP bank is very similar to the format of the calorimeter pedestal banks. Everything in Section 31.2 also applies to the TOFP bank. The bitfields in the channel ID are identical (see Table 170). The data words are also identical (see Table 169). The only difference is that there are only eight blocks in the TOFP bank.

There is general information regarding the Time of Flight system in Section 25. The geometry, location, and numbering of the photomultipliers are described in that section. The map from the "electronic addresses" in the TOFC/TOFP channel IDs to the "geometric addresses" will be documented elsewhere. This map is the same for TOFC and TOFP.

31.5 FDAC, FDAP - Beam Shower Counter Calibration Banks

The format of the FDAC bank is very similar to the format of the calorimeter calibration banks. Everything in Section 31.1.1 also applies to the FDAC bank. The bitfields in the channel ID are identical (see Table 167). The data words are also identical (see Table 168). The only difference is that there is only one block in the FDAC bank.

The format of the FDAP bank is very similar to the format of the calorimeter pedestal banks. Everything in Section 31.2 also applies to the FDAP bank. The bitfields in the channel ID are identical (see Table 170). The data words are also identical (see Table 169). The only difference is that there is only one block in the FDAP bank.

There is general information regarding the Beam Shower Counter in Section 27.2. The geometry, location, and numbering of the photomultipliers are described in that section. The map from the "electronic addresses" in the FDAC/FDAP channel IDs to the "geometric addresses" is not documented yet. This map is the same for FDAC and FDAP.

31.6 MPAC, MPAP - Miniplug Calibration Banks

The format of the MPAC bank is very similar to the format of the calorimeter calibration banks. Everything in Sec. 31.1.1 also applies to the MPAC bank. The bitfields in the channel ID are identical (see Table 167). The data words are identical (see Table 168).

The format of the MPAP bank is very similar to the format of the calorimeter pedestal banks. Everything in Sec. 31.2 also applies to the MPAP bank. The bitfields in the channel ID are identical (see Table 170). The data words are also identical (see Table 169).

The only difference in bank structure between the miniplug banks and the other calorimeter calibration/pedestal banks is that MPAC and MPAP both have 4 blocks. The channels are divided among the blocks as shown in Table 171. More information about the miniplug can be found in Section 27 where the MPAD bank is described. The full map from the "electronic addresses" (crate, slot, QIE number) to the "geometric addresses" is not documented yet.

Block 0	W sum
Block 1	E sum
Block 2	W tower
Block 3	E tower

Table 171: *Blocks in the MPAC and MPAP.*

31.7 Shower Maximum Calibration Banks

31.7.1 Channel IDs and Other Common Features

The bit assignment for the Channel ID is specified in Table 172. This applies to the CESC, CESP, CPRC, CPRP, CCRC, and CCRP banks. There are a total of 16 crates, numbered 1 to 16. In each bank, there is one data block per crate. Block numbers run from 0 to 15. Block number equals crate number minus one. Each block handles 3 wedges. Block 0 handles wedges 0 to 2 on the west side. Block 1 handles wedges 3 to 5. This pattern continues in phi order on the west side then repeats on the east side.

Each crate contains 3 SMXR cards. Each SMXR card contains 8 separate channels, 6 of which are the data for 1 wedge of Central Shower Max (CES), 1 channel for the data for 1 wedge of Central Preshower (CPR), and 1 channel for the data for 1 wedge of Central Crack Chambers (CCR). Each SMXR channel reads out 32 QIE channels. Each QIE chip has 4 different capacitors numbered 0 to 3. Each capacitor has 8 different ranges, numbered 0 to 7.

These banks may be written out with some channels not present. The order that the channels are written into the bank is not specified. The map from (slot, SMXR channel, QIE number) to the hardware channels is described for each bank individually.

These banks have a block pointer section as described in Section 4.1.

Bit	Channel ID
31:24	Crate Number (1:16)
23:19	Slot Number
18:16	SMXR Channel
15:8	QIE Number
7:4	Capacitor Number (0:3)
3:0	Current Range (0:7)

Table 172: *Bit assignment for the Central Shower Maximum Calibration Bank Channel ID. (CESC, CESP, CPRC, CPRP, CCRC, and CCRP). For the pedestal banks, the "Current Range" field has a different meaning than in the calibration banks. In the pedestal banks, it is an "average range".*

31.7.2 CESC - Central Electromagnetic Shower Max Calibration Data

The bank format is specified in Table 173. For each data element there are six 32-bit words. First, an integer gives the channel ID. Then there are five floats. A block of data contains repeated occurrences of the six word data elements and nothing else.

Section 31.7.1 describes the channel ID and other things in the CESC data format that are the same as other Shower Max raw data banks.

The map from (slot, SMXR channel, QIE number) to the hardware channels is not documented yet.

31.7.3 CCRC - Central Crack Chamber Calibration Data

The bank format is specified in Table 173. For each data element there are six 32-bit words. First, an integer gives the channel ID. Then there are five floats. A block of data contains repeated occurrences of the six word data elements and nothing else.

Section 31.7.1 describes the channel ID and other things in the CCRC data format that are the same as other Shower Max raw data banks.

The map from (slot, SMXR channel, QIE number) to the hardware channels is not documented yet.

31.7.4 CPRC - Central Preradiator Calibration Data

The bank format is specified in Table 173. For each data element there are six 32-bit words. First, an integer gives the channel ID. Then there are five floats. A block of data contains repeated occurrences of the six word data elements and nothing else.

Section 31.7.1 describes the channel ID and other things in the CPRC data format that are the same as other Shower Max raw data banks.

The map from (slot, SMXR channel, QIE number) to the hardware channels is not documented yet.

"CESC" or "CPRC" or "CCRC"
Bank Number
Bank Version
Bank Length
Bank Type = 3 (I*4 monotype)
Number of Blocks (=16)
Pointer to Block 0 (West)
...
Pointer to Block 7 (West)
Pointer to Block 8 (East)
...
Pointer to Block 15 (East)
Pointer to End of Data
Channel ID (I*4)
Slope (R*4)
Offset (R*4)
Slope Error (R*4)
Offset Error (R*4)
Fit Chisq (R*4)
Channel ID (I*4)
Slope (R*4)
Offset (R*4)
Slope Error (R*4)
Offset Error (R*4)
Fit Chisq (R*4)
...

Table 173: *Structure of the CESC, CPRC, and CCRC banks. The format is identical for all three banks. Only the bank names and the map to hardware channels are different.*

31.7.5 CESP - Central Electromagnetic Shower Max Pedestals

The bank format is specified in Table 174. For each data element there are three 32-bit words. First, an integer gives the channel ID. Then there are two floats. A block of data contains repeated occurrences of the three word data elements and nothing else.

Section 31.7.1 describes the channel ID and other things in the CESP data format that are the same as other Shower Max raw data banks.

The map from (slot, SMXR channel, QIE number) to the hardware channels is the same as for CESC (see Section 31.7.2).

31.7.6 CCRP - Central Crack Chamber Pedestals

The bank format is specified in Table 174. For each data element there are three 32-bit words. First, an integer gives the channel ID. Then there are two floats. A block of data contains repeated occurrences of the three word data elements and nothing else.

Section 31.7.1 describes the channel ID and other things in the CCRP data format that are the same as other Shower Max raw data banks.

The map from (slot, SMXR channel, QIE number) to the hardware channels is the same as for CCRC (see Section 31.7.3).

31.7.7 CPRP - Central Preradiator Pedestals

The bank format is specified in Table 174. For each data element there are three 32-bit words. First, an integer gives the channel ID. Then there are two floats. A block of data contains repeated occurrences of the three word data elements and nothing else.

Section 31.7.1 describes the channel ID and other things in the CPRP data format that are the same as other Shower Max raw data banks.

The map from (slot, SMXR channel, QIE number) to the hardware channels is the same as for CPRC (see Section 31.7.4).

"CESP" or "CPRP" or "CCRP"
Bank Number
Bank Version
Bank Length
Bank Type = 3 (I*4 monotype)
Number of Blocks (=16)
Pointer to Block 0 (West)
...
Pointer to Block 7 (West)
Pointer to Block 8 (East)
...
Pointer to Block 15 (East)
Pointer to End of Data
Channel ID (I*4)
Pedestal Mean (R*4)
Pedestal RMS (R*4)
Channel ID (I*4)
Pedestal Mean (R*4)
Pedestal RMS (R*4)
...
...

Table 174: *Structure of the CESP, CPRP, and CCRP banks. The format is identical for all three banks. Only the bank names and the map to hardware channels are different.*

31.8 RPAC, RPAP - Roman Pot Calibration Banks

The format of the RPAC (RPAP) banks are very similar to the format of the Shower Max calibration (pedestal) banks as described in Sec. 31.7.1. The bitfields in the channel ID are identical (see Table 172). Each SMXR reads out in the mode used for the Plug Shower Max (48 QIE channels), with channels 0-39 corresponding to x or y fibers 0-39 of a given Roman pot. The data words are also identical, but there are six blocks. The structure of the calibration and pedestal banks is shown in Table 175 and Table 176, respectively.

More information about the Roman pot detector can be found in Section 27 where the RPAD bank is described. The full map from the "electronic addresses" (slot, SMXR channel, QIE number) to the "geometric addresses" is not documented yet.

"RPAC"
Bank Number
Bank Version
Bank Length
Bank Type = 3 (I*4 monotype)
Number of Blocks (=6)
Pointer to Block 0 (1x)
Pointer to Block 1 (1y)
...
Pointer to Block 5 (3y)
Pointer to End of Data
Channel ID (I*4)
Slope (R*4)
Offset (R*4)
Slope Error (R*4)
Offset Error (R*4)
Fit Chisq (R*4)
Channel ID (I*4)
Slope (R*4)
Offset (R*4)
Slope Error (R*4)
Offset Error (R*4)
Fit Chisq (R*4)
...

Table 175: *Structure of the Roman pot calibration bank (RPAC).*

"RPAP"
Bank Number
Bank Version
Bank Length
Bank Type = 3 (I*4 monotype)
Number of Blocks (=6)
Pointer to Block 0 (1x)
Pointer to Block 1 (1y)
...
Pointer to Block 5 (3y)
Pointer to End of Data
Channel ID (I*4)
Pedestal Mean (R*4)
Pedestal RMS (R*4)
Channel ID (I*4)
Pedestal Mean (R*4)
Pedestal RMS (R*4)
...

Table 176: *Structure of the Roman pot pedestal bank (RPAP).*

31.9 PPRC, PPRP, PESC, PESP - Plug Preradiator and Shower Max Calibration Banks

The formats of the PPRC, PPRP, PESC, and PESP banks are very similar to the formats of the central Shower Max calibration and pedestal banks as described in Sec. 31.7.1. The bitfields in the channel ID are identical (see Table 172). The data words are also identical. The only difference is that there are eight blocks in these banks. The structures of these calibration and pedestal banks are shown in Tables 177 and Table 178, respectively.

More information about the plug preradiator and shower max detectors can be found in Sections 14 and 15 where the PPRD and PESD banks are described. The full map from the "electronic addresses" (slot, SMXR channel, QIE number) to the "geometric addresses" is not documented yet.

"PPRC" or "PESC"
Bank Number
Bank Version
Bank Length
Bank Type = 3 (I*4 monotype)
Number of Blocks (=8)
Pointer to Block 0 (West)
Pointer to Block 1 (West)
...
Pointer to Block 3 (West)
Pointer to Block 4 (East)
...
Pointer to Block 7 (East)
Pointer to End of Data
Channel ID (I*4)
Slope (R*4)
Offset (R*4)
Slope Error (R*4)
Offset Error (R*4)
Fit Chisq (R*4)
Channel ID (I*4)
Slope (R*4)
Offset (R*4)
Slope Error (R*4)
Offset Error (R*4)
Fit Chisq (R*4)
...

Table 177: *Structure of the PPRC and PESC banks.*

“PPRP” or “PESP”
Bank Number
Bank Version
Bank Length
Bank Type = 3 (I*4 monotype)
Number of Blocks (=8)
Pointer to Block 0 (West)
Pointer to Block 1 (West)
...
Pointer to Block 3 (West)
Pointer to Block 4 (East)
...
Pointer to Block 7 (East)
Pointer to End of Data
Channel ID (I*4)
Pedestal Mean (R*4)
Pedestal RMS (R*4)
Channel ID (I*4)
Pedestal Mean (R*4)
Pedestal RMS (R*4)
...

Table 178: *Structure of the PPRP and PESP banks.*

31.10 LEDC and XEFC - CEM Light Emitting Diode/Xenon Flashing Calibration Banks

The bit assignment for the channel ID is specified in Table 179. Three bit fields identify the channel. A fourth bit field gives the number of events used to calculate the data words. The crate, slot, and QIE numbering are the same as for CEMC bank (see Section 31.1.2) with the following exception. If the QIE number is greater than 20, then it is not a QIE channel; it is a Calcard channel.

- QIE number 20 refers Calcard data for PIN diode 0.
- QIE number 21 refers Calcard data for PIN diode 1.
- QIE number 22 refers Calcard data for PIN diode 2 (XEF only).
- QIE number 23 refers Calcard data for the TAC.

Bit	Channel ID
31:27	Crate Number (1:16)
26:20	Slot Number (0:21)
19:12	QIE Number (0:19) Calcard Word Number (20:23)
11:0	Number of Events

Table 179: *Bit assignment for the channel ID of the LEDC and XEFC banks.*

There are a total of 16 crates for the Central Electromagnetic calorimeter. The crates are numbered 1 to 16. Data from one crate is put into one data block. The data blocks are numbered 0 to 15 (block number equals crate number minus one). The blocks are written into the bank from the west side first in ϕ order. Then the blocks from the east side are written into the bank in ϕ order.

Each crate holds 3 EM ADMEM cards (1 ADMEM per wedge) and one calibration card (Calcard) calibrating 3 wedges with Light Emitting Diode (LED) or Xenon Flashing (XEF) systems. There are 3 LEDs and 1 XEF per wedge. The light output of these systems is monitored by PIN diodes. There are 2 PIN diodes per wedge for the LED system and 3 PIN diodes per wedge for the XEF.

The format of the banks is shown in Table 180. There are five 32 bit words for each channel. The first word is an integer. It is the channel ID described above. Then there are four floats. For the QIE channels these four data words are: the mean value and the RMS of the PMT ADC counts, the mean value and the RMS of the ratio number of PMT counts divided by the number of counts of the corresponding PIN diode (computation on an event-by-event basis). For the PIN diodes, the results of interest are the mean and RMS of the PIN diode counts and the mean and RMS of the ratio of the PIN diode counts (computed on an event-by-event basis). The mean value and the RMS of the TAC counts are also retained.

Channel ID (QIE 0)
$< \text{PMT counts} >$ $\sigma(\text{PMT counts})$ $< \frac{\text{PMT counts}}{\text{PIN diode counts}} >$ $\sigma(\frac{\text{PMT counts}}{\text{PIN diode counts}})$
.....
Channel ID (QIE 19)
$< \text{PMT counts} >$ $\sigma(\text{PMT counts})$ $< \frac{\text{PMT counts}}{\text{PIN diode counts}} >$ $\sigma(\frac{\text{PMT counts}}{\text{PIN diode counts}})$
Channel ID (Calcard PIN diode 0)
$< \text{PIN diode 0 counts} >$ $\sigma(\text{PIN diode 0 counts})$ $< \frac{\text{PIN 0 counts}}{\text{PIN 1 counts}} >$ $\sigma(\frac{\text{PIN 0 counts}}{\text{PIN 1 counts}})$
....
Channel ID (Calcard PIN diode 2)
$< \text{PIN diode 2 counts} >$ $\sigma(\text{PIN diode 2 counts})$ $< \frac{\text{PIN 2 counts}}{\text{PIN 0 counts}} >$ $\sigma(\frac{\text{PIN 2 counts}}{\text{PIN 0 counts}})$
Channel ID (Calcard TAC)
$< \text{TAC counts} >$ $\sigma(\text{TAC counts})$ 0 0
... repeat for each wedge in this block

Table 180: *Structure of one LEDC/XEFC data block. There are 16 of these blocks in a bank. The bank also includes a bank header and block pointers as described in Sections 4 and 4.1. The <> symbols indicate a mean value. σ() indicates an RMS value.*

31.11 COT Calibration Banks

Unlike the calorimeter calibration banks, the COT calibration banks contain only unsigned integers. This state of affairs was brought about by the need to pack several pieces of information onto one 32 bit word in the front end crate. This also means that these banks should be immune to the unlikely possibility that some future architecture will have a different representation of float's and int's.

31.11.1 Channel ID and other Common Features

The bit assignment for the COT channel IDs is given is Table 181. This applies to both the CTTC and CTIC banks. Note that the channel ID gives the "geometric address" for a given channel rather than the "electronic address". The upper 3 bits give the superlayer, the next 6 highest bits give the TDC module number, the next 3 bits give the cell number, and finally the next 4 bits give the wire number. The lowest 16 bits in the Channel ID word are not used. The TDC module number is redundant. It can be calculated from superlayer, cell, and wire. These geometric indices are described in a little more detail in Section 24.1, where the COTD format is described. The map from the "geometric address" to "electronic address" is also described in Section 24.1.

These banks may be written out with some channels missing. The order channels are written into the bank is not specified.

These banks have a block pointer section as described in Section 4.1.

Bit	Channel ID
31:29	Superlayer (0:7)
28:23	TDC Module (0:59)
22:20	Cell Number (0:7)
19:16	Wire number (0:11)
15:0	Undefined

Table 181: Bit assignment for the COT Calibration Channel ID (CTTC and CTIC).

31.11.2 CTTC - COT T_0 Offset Calibration Data

The format of the T_0 calibration data is given in Table 182. Three 32 bit data words (integers) are written for each channel. The first word is the channel ID, followed by the T_0 offset, and a diagnostic word. The diagnostic word is composed of 2 1-byte words and 1 2-byte word; the two upper bytes give the rms's of the start times and the widths output by the TDC for the given channel, while the two lowest bytes give the slope of the T_0 offset (see Table 183). It should be noted that the start time distribution is what the front-end code uses to obtain the T_0 offset, whereas the width distribution is used only for diagnostic purposes. The slope measures how much the output start time changes as the input pulse time changes (ideally, the output time should track the input time exactly).

"CTTC"
Bank Number
Bank Version
Bank Length
Bank Type (=3)
Number of Blocks(=20)
Pointer to Block 0
...
Pointer to Block 19
Pointer to End of Data
Channel ID
Timing Constant
T_0 diagnostic word
Channel ID
Timing Constant
T_0 diagnostic word
...

Table 182: Structure of the CTTC banks.

Bit	Diagnostic Word
31:24	start time rms
23:16	width rms
15:0	T_0 slope

Table 183: Bit assignment for the CTTC diagnostic word

31.11.3 CTIC - COT dE/dx Calibration Data

The form of the dE/dx calibration data is given in Table 184. Four 32 bit words (integers) are written for each channel. The first word is the channel ID; this is followed by a word which gives dE/dx parameter 1, a word which gives dE/dx parameters 2 and 3, and a diagnostic word. The upper 2 bytes of the third word are used for dE/dx parameter 2 while the lower 2 bytes are used for dE/dx parameter 3. The diagnostic word is composed of 2 2-byte words; the two upper bytes give the rms's of the widths output by the TDC for the given channel, while the two lowest bytes give the χ^2 of the dE/dx fit used to get the dE/dx parameters (Table 185).

"CTIC"
Bank Number
Bank Version
Bank Length
Bank Type (=3)
Number of Blocks(=20)
Pointer to Block 0
...
Pointer to Block 19
Pointer to End of Data
Channel ID
dE/dx parameter 1
bits 16:31 dE/dx parameter 2; bits 0:15 dE/dx parameter 3
dE/dx diagnostic word
Channel ID
dE/dx parameter 1
bits 16:31 dE/dx parameter 2; bits 0:15 dE/dx parameter 3
dE/dx diagnostic word
...

Table 184: Structure of the CTIC banks.

Bit	Diagnostic Word
31:16	width rms
15:0	fit χ^2

Table 185: Bit assignment for the CTIC diagnostic word

32 Silicon (SVX/ISL+L00) Calibration Banks

Since all Silicon detectors are read out in the same manner, it makes sense to treat all X mode calibration banks the same. In X Mode, the Vrbs are read out and analyzed in the VME crate, and only after a certain number of events, the final calculations are done and the results are sent to the Reformatter to be concatenated into the final calibration bank(s).

The simplest way to accomplish this is to make the output minibanks mimic the format of Vrb Blocks (See Section 9.3.2) with a key so that the Reformatter knows to assign the proper name to the bank, but otherwise formats what it receives in precisely the same way as the Raw Data Banks (Sections 20.1 and 22.1).

The HDI Blocks (Section 9.3) of the calibration banks will start with the same FIB ID headers, but will write the top 6 bits of the Back End State Byte to signal the type of calibration bank. In this way the Reformatter which examines FIB ID0 to decide which subdetector this data is from simply uses information in the same 32 bits to decide what the name of the bank is. Everything else at the reformatter level is identical to reformatting raw data.

As the amount of information is different for each type of callibration, the format of the HDI block past the fib headers is modified according to the type of calibration. The basic idea is to re-use all the same header information as the raw data format, and replace raw data information (channel byte, pulseheight byte) with that relevant for the calibration (eg. channel byte, n bytes of calib info) where the amount and sequence of information is calibration dependent. In addition, information which is not channel dependent such as # of Events, will be placed before the HDI Block. The HDI Block will still be padded to an 8 byte boundary. The main guidelines for “designing” the banks are

- Minimize size of information due to the large quantity of data (401 kChan for SVX, 303 + 14 kChan for ISL + L00) so the banks don't get too big
- Provide reasonable digital precision of information
- Use some reasonable encoding scheme (i.e, always use an even # of bytes)

32.1 Pedestal Calibration: SIXP/ISLP banks

For X Mode pedestals, the analysis produces per channel pedestals, noise and differential noise relative to the neighboring channel, for each of the back-end states of the SVX3 Chip tracked by the SRC. In addition, pedestals can be taken either with Dynamic Pedestal Subtraction (DPS) on, or off. The modifications to the format are as follows:

- Back End State Byte[7:2] == 0. This defines the bank as a pedestal bank.
- Back End State Byte[1:0] == {0,1}, if the data was taken with DPS off, or on, respectively.
- Extra Information: 2 Bytes for the Number of back-end states defined + 2 Bytes of # Events for each of the defined Back End states
- Channel Info: 14 Bytes:
  Channel, (Pedestal, Noise, Differential Noise) x #BE states + padding byte
  Each Pedestal in in the range [0, 127.5] in units of 0.5 ADC counts, and overflow

going into the last bin. In DPS OFF mode, the Noise and Differential Noise are in the range [0, 31.875] in units of 0.125 ADC counts, again with overflow going into the top bin. The padding byte is optional, depending on the number of BE states kept track of by the SRC. Initially this number will be 4, but may be reduced once we gain more experience running the detector.

In DPS ON mode, the noise is as above, however the locations reserved for the channel's DNoise are instead filled with the fraction of events with non-zero data on that channel. The data is in the range [0,1.] in units of 1/255., i.e. 255 corresponds to every event having non-zero data on that channel.

With this structure the total size of the SIXP and ISLP banks sum to ≈ 14 Bytes $\times\, 7.25 \times 10^5$Channels ≈ 11 MBytes.

The pedestal, noise and dnoise for a given back-end state can be accessed as follows:

- Channels occur at intervals of int((3*# BE states + 2)/2)
- Within a channel, the data (pedestal, noise, dnoise) for back-end state s (up to # BE states) is at byte $b = 1 + 3 * s + \text{off}$, where off={0,1,2} for pedestal, noise, dnoise respectively.
- The data is then at word int(b/2), and low/high byte given by (b+1)%2.

An example HDI Block for a Pedestal Calibration is given in Table 186.

32.2 Threshold Scan: SIXT/ISLT banks

In the SVX3 chip, one threshold is applied to all channels, before FIB pedestal subtraction, regardless of Back End State, to maintain a reasonable noise occupancy while not losing any signal efficiency. The required threshold setting for a given noise occupancy can be calculated from the pedestal and noise averaged across all channels, given that the noise is gaussian, which is expected. As such, a threshold scan where the noise occupancy is measured for various threshold settings is essentially a check of this calculation.

The required bank format for such a calibration is simply for each chip the threshold setting and the chip occupancy for several points on the occupancy curve. Due to the fact that threshold units (1 ADC count) are similar to noise, high precision is required in the occupancy information. The current plan is to scan ~ 7 threshold points from $t_{nom} - 4$ to $t_{nom} + 2$, where t_{nom} is the nominal threshold for that chip. The bank would be sent to a consumer to evaluate the differences between calculated occupancy and measured occupancy at each point, and put results in the database.

For a Threshold scan of n points, the HDI Block modifications are:

- Back End State Byte[7:2] == 1
- Extra Information: Number of point, n, $n \times 2$ Bytes # Events for Point i
- Chip Info: $4 \times n + 2$ Bytes:
  Chip ID, n, $n\times$ (1 byte Setting, 3 bytes Occupancy)
  Threshold is in the Range [0,255] and occupancy is in the range [0, 1] in units of 2^{-24}.Blecchh.

An example of a SIXT HDI block is given in Table 187. For 7 threshold points, the total size of the SIXT and ISLT banks sum to ≈ 30 Bytes $\times$ 5644Chips ≈ 170 kBytes.

32.3 Gain Scan: SIXC/ISLC banks

Although not directly applicable in the online framework (contrasted with thresholds and pedestals) the gain of the Silicon system can be measured using charge injection from the chip, and the resulting numbers may be useful offline and will be useful for monitoring and setting the optimal chip parameters. To determine the gain again a scan is used, with the charge injected being the variable scanned. The result is ideally a straight line fit of charge injected vs average pulseheight per channel.

Unfortunately, the online tunable parameter, the DAC setting, does not map directly to charge injected, and the conversion depends on information not available in the VME crates. As such, the output of an X mode calibration will be a set of SIXC banks, on per DAC setting point, with that DAC Setting (per HDI) and the average pulse height for each channel. The final conversion of DAC setting to charge injected and the fit and resulting slope will be left to the online consumer. For each SIXC bank the HDI Block modifications are:

- Back End State Byte[7:2] == 2
- Bunch Crossing Byte $\Rightarrow$ Calibration point n
- Extra Information: 16 Bits of # of Events
- Chip Info: 2 bytes:
  Chip ID, DAC Setting for this point
- Channel Info: 2 bytes:
  Channel ID, Average Pulseheight [0:255] in units of 1 ADC count

An example of a SIXC HDI block is given in Table 188. For each DAC setting point, the size of the SIXC and ISLC banks sum to ≈ 2 Bytes $\times\ 7.25 \times 10^5$Channels ≈ 1.45 MBytes.

FIB ID0	FIB ID1
Reserved	BE State == 0/1
# BE States == 4	
# Events BE 0	
# Events BE 1	
# Events BE 2	
# Events BE 3	
Chip ID	Status
Channel ID	Ped 0
Noise 0	DNoise 0
Ped 1	Noise 1
DNoise 1	Ped 2
Noise 2	DNoise 2
Ped 3	Noise 3
DNoise 3	00
...	
Channel 0x7f	Ped 0
Noise 0	DNoise 0
Ped 1	Noise 1
DNoise 1	Ped 2
Noise 2	DNoise 2
Ped 3	Noise 3
DNoise 3	00
...	
Chip ID	Status
Channel ID	Ped 0
...	
Channel 0x7f	Ped 0
...	
DNoise 3	00
EOR	EOR
00	00

Table 186: *Decoding for an HDI Block from a X Mode Pedestal Calibration. Each line in the table represents a 16 bit word, with the most significant byte first (left). The total data block is padded to a multiple of 64 bits by bytes of 0X00*

FIB ID0	FIB ID1
Reserved	BE State == 4
n points	
# Events Point 1	
...	
# Events Point n	
Chip ID t_1 $Occ_1[15:8]$	Reserved $Occ_1[23:16]$ $Occ_1[7:0]$
...	
t_n $Occ_n[15:8]$	$Occ_n[23:16]$ $Occ_n[7:0]$
...	
EOR	EOR
00	00

Table 187: *Example of HDI Block for SIXT Bank*

FIB ID0	FIB ID1
i^{th} point	BE State == 8
# Events	
Chip ID Channel Channel	DAC value Ave ADC Ave ADC
...	
Chip ID Channel	DAC value Ave ADC
...	
...	
EOR	EOR
00	00

Table 188: *Example of HDI Block for SIXC Bank*

References

[1] The CDF Upgrade, June 14 1995.

[2] YBOS PROGRAMMERS REFERENCE MANUAL, CDF Note No. 156, Version 4.00, 10 July 1992. http://www-cdf.fnal.gov/offline/ybos/murat/ybos.01.html

[3] CDF Note No. 4103, Numbering and Naming Convention for SVX-II and ISL, J. Conway et al.

[4] ESE-SVX-951010, SVXII Silicon Strip Detector Upgrade Project Fiber Interface Board Specification, K. Woodbury.

[5] "The CDF RunII Luminosity Monitor", S. Kimenko et al. , CDF Note 4330.

[6] "Luminosity Information in RunII", J. Konigsberg et al. , CDF Note 4831.

[7] "Conceptual Design of CDF FRED for Run II", J. Thurston Chandler, Greg Feild, Karen Ohl, Michael Schmidt, CDF Note 4470.

[8] "Specification and Design of the Level 1 Calorimeter Trigger PreFRED Modules for the Run II Upgrade", A. Amadon, M. Bogdan and P. Wilson, CDF Note 4340.

[9] "Trigger Supervisor Protocols for CDF II", R. Greg Feild, Andrew J. Martin and Michael P. Schmidt, CDF Note 4469.

[10] "The Global Level 2 Decision Module and the TS $\leftrightarrow$ Level 2 Protocol", R. G. Feild, A. J. Martin and M. P. Schmidt, J. Yu, M. K. Campbell, S. Miller, CDF Note 4519.

[11] "Specification of Calibration Control Capabilities of the CDF II Trigger Supervisor", A. J. Martin and M. P. Schmidt, CDF Note 4547.

[12] "Deadtime/Livetime Accounting and the Scalers for the Run II Trigger/DAQ Upgrade", R. Greg Feild, Andrew J. Martin, Karen E. Ohl and Michael P. Schmidt, CDF Note 4480.

[13] "The CDF Run II Hardware Database and Online Java API", W. Badgett, CDF Note 5681.