CDF/DOC/TRIGGER/4225
Version 2.2
Sept 26, 2005

DIRAC — The FINAL Specification

John Wahl and Pedro Amaral
University of Chicago
Update to V2.2 by Peter Wilson
Fermilab

Abstract

This note describes the Digital Information Receive and Compare (DIRAC) board. DIRAC is the Run II replacement for the Run I RAW/CAS boards. DIRAC is the core processor for the Level 1 calorimetry trigger. The expected inputs and outputs of the board are specified. The sequence of operations performed on the input signals to produce the output signals is discussed. The VMEbus addressing of the DIRAC board is specified. Appendix B specifies all the info that needs to be downloaded before a Cold Start.

Note revision 2.2 made to correct description of HAD_EM_SEL in Control_1 which has opposite behavior from previous documentation. Also add details describing Sum E_T path including two new tables and bug in implementation of two tower sum (Section 3.10 and Appendix C).

1 Purpose

Currently, collisions occur approximately every 3 years at the B0 collision hall. With the completion of the Main Injector upgrade, a collision may occur every 132ns. The previous level 1 trigger was not able to function at this speed. Therefore, a new level 1 trigger must be designed to meet this challenge. A general knowledge of the RUN II trigger system is assumed. If anything is unclear refer to the documents referenced in the confusing section. If there are no documents referenced, pray.

2 Level 1 DIRAC Inputs and Outputs

2.1 Detector Information

Run II is the first collider run at the new bunch crossing time of 132ns. For Run II the Front-end electronics (FE) and the level 1 (L1) trigger are being replaced. (Among many others) The new L1 trigger expects digital signals from the FE ADMEM boards on the detector.[1] DIRAC accepts a 10 bit word for each trigger tower ($15°\phi \times 0.2\eta$). Each card processes 6 EM and 6 HAD trigger towers from a quarter of the same wedge in ϕ.[2] Therefore, each card receives 120 bits from the detector. There are 16 DIRAC cards in a crate, so each crate receives 1920 signals from 4 wedges of the detector. (Figure 1)

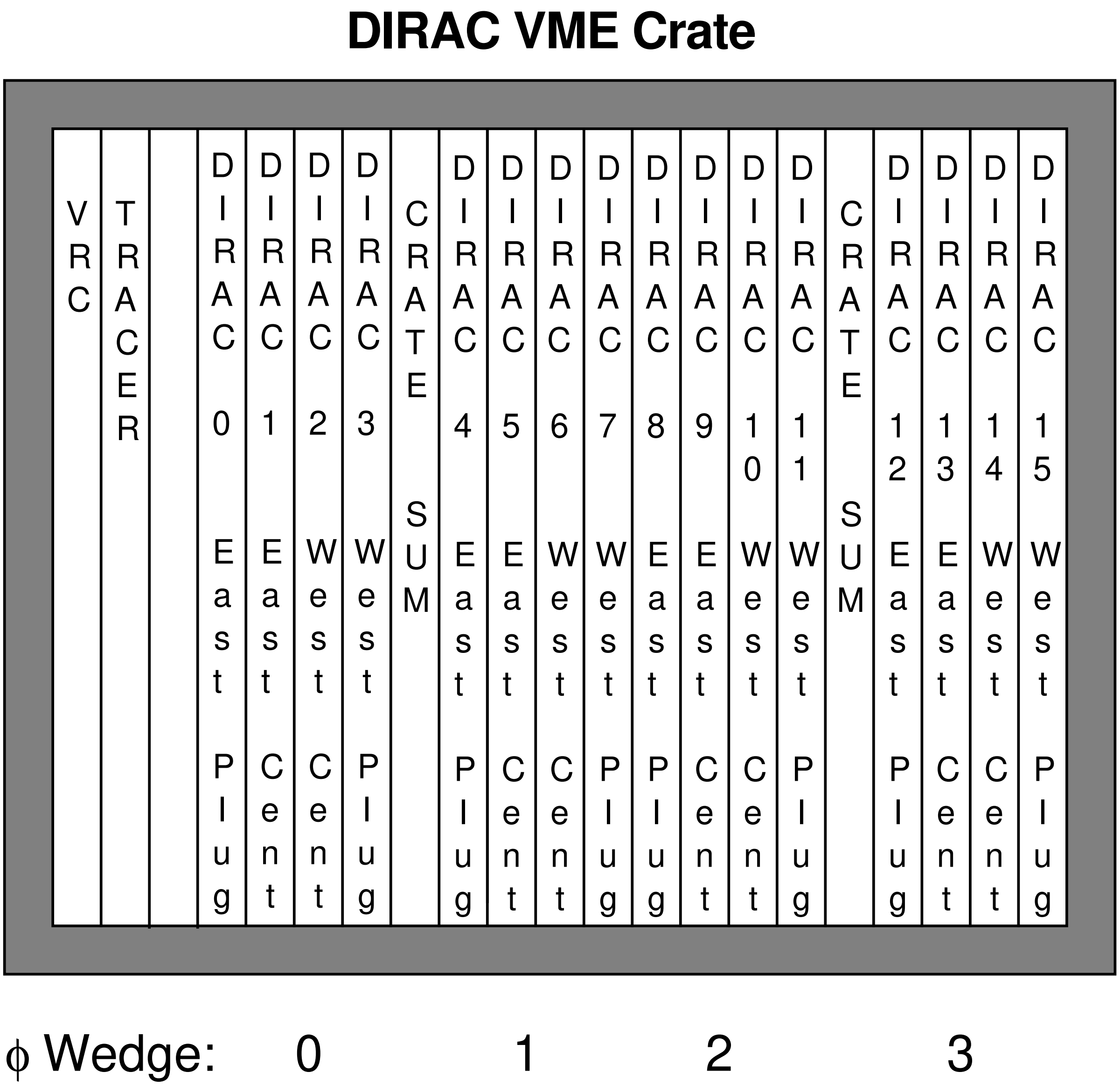

Figure 1: L1 Calorimetry Crate for DIRAC and Crate Sum

2.2 The Crate

There are 2 backplanes in each crate. The upper backplane is a CDF standard backplane.[3] On the upper backplane are 3 connectors for each slot. The top connector is the standard 5-row VME-P P1.[4] The middle connector on the upper backplane is a special VME-P power connector, P0.[4] The lower connector, P2, is a hybrid. The middle row of signals on P2 are the standard VME lines. The outer four rows carry global CDF control signals and extra power on pins unused by VME.[5] The DIRAC boards are designed to respond to the VMEbus protocol for downloading information at the beginning of a run, DAQ readout and diagnostics. The second backplane, P3, is custom. An auxillary card (AUX) which receives the 120 differential detector signals(for a total of 240) plugs into P3 from the rear of the crate.[6] The AUX card converts the 120 differential signals to single ended signals and passes them to DIRAC through P3. All trigger tower signals are required to arrive at DIRAC at the same time.

There is one Crate Sum (CS) board per eight DIRAC boards.[7] CS receives an 8 bit ΣE_T and 16 bit trigger summary from each DIRAC card across the P3 backplane. This means that the P3 backplane must be at least 192 bits (192 signals from the 8 DIRAC boards) wide for CS and at least 144 bits (24 signals to CS + 120 signals from the AUX card) for DIRAC(this does not include ground pins). The pin configuration for DIRAC's P3 is shown in Table 1. For the CS's pin configuration on P3 see Trigger note 91.[7]

Table 1: P3 Backplane Pin Assignments

	Column				
Row	A	B	C	D	E
1	ground	TRSUM(0)	TRSUM(1)	TRSUM(2)	TRSUM(3)
2	NC	TRSUM(4)	TRSUM(5)	ground	TRSUM(6)
3	NC	TRSUM(7)	ground	TRSUM(8)	TRSUM(9)
4	NC	TRSUM(10)	TRSUM(11)	ground	TRSUM(12)
5	ground	TRSUM(13)	ground	TRSUM(14)	TRSUM(15)
6	NC	TOWIN(0)	TOWIN(1)	TOWIN(2)	TOWIN(3)
7	NC	TOWIN(4)	TOWIN(5)	ground	TOWIN(6)
8	NC	TOWIN(7)	TOWIN(8)	TOWIN(9)	TOWIN(10)
9	ground	TOWIN(11)	ground	TOWIN(12)	TOWIN(13)
10	NC	TOWIN(14)	TOWIN(15)	TOWIN(16)	TOWIN(17)
11	NC	TOWIN(18)	TOWIN(19)	ground	TOWIN(20)
⋮	⋮				
38	NC	TOWIN(112)	TOWIN(113)	TOWIN(114)	TOWIN(115)
39	NC	TOWIN(116)	TOWIN(117)	ground	TOWIN(118)
40	NC	TOWIN(119)	AUX_SUM_EN0	ET_SUM(0)	ET_SUM(1)
41	ground	ET_SUM(2)	AUX_SUM_EN1	ET_SUM(3)	ET_SUM(4)
42	NC	ET_SUM(5)	ET_SUM(6)	ground	ET_SUM(7)
43	NC	NC	AUX_ENABLE	NC	NC
44	NC	NC	NC	NC	NC

Table 2: Front Panel Connector 4 – Tracking Bits.

Pin	Signal	Pin	Signal
1	Bit 0-	2	Bit 0+
3	Bit 1-	4	Bit 1+
5	Bit 2-	6	Bit 2+
7	Bit 3-	8	Bit 3+
9	Bit 4-	10	Bit 4+
11	Bit 5-	12	Bit 5+
13	Bit 6-	14	Bit 6+
15	Bit 7-	16	Bit 7+

2.3 Tracking Input

We allow information from the Extra Fast Tracking (XFT) system to be used in a L1 calorimetry decision.[8] Eight bits per 15° in ϕ are received differentially through a front panel connector from XFT via the eXTRaPolation (XTRP) boards (see Table 2). These 8 bits are refer to 8 thresholds defined on the XTRP boards. The 8 bits are sent to 3 SRAM's that decode the information into 3 bits per trigger tower. DIRAC can decode the information regardless of the meaning of the triggers used to set the 8 bits. Only the DIRAC boards examining the Central and Plug calorimeter data receive tracking data. The same data is sent to each DIRAC card in a ϕ slice. For a crate there are 32 (4 ϕ slices $\times$ 8) bits from the XTRP. This information arrives at time t_{Tracking} after the bunch crossing.

Table 3: Front Panel Connector 5 – Bit Blaster

	Column	
Row	A	B
1	TCK	GND
2	TDO	VCC
3	TMS	NC
4	NC	NC
5	TDI	GND

2.4 Bit/Byte Blaster Input

To facilitate reprogramming of the EPLDs resident on Dirac, we have implemented Altera's In System Programmablility (ISP).[9] This allows most, but not all the Altera EPLDs on Dirac to be soldered and still be reprogrammed at a future date. See the detailed Dirac document for details on reprogramming.[10] The Bit Blaster (or Byte Blaster) connector is a 10 pin connector on Dirac's front panel and the definition of the bits is shown in table 3.[11]

3 Board Modules

Figure 2 shows a block level diagram of DIRAC. The module **Towers** contains 6 copies of the L1 decision electronics and as shown in figure 3. Objects in bold face within this document are names of modules or signals in figure 2 or 3.

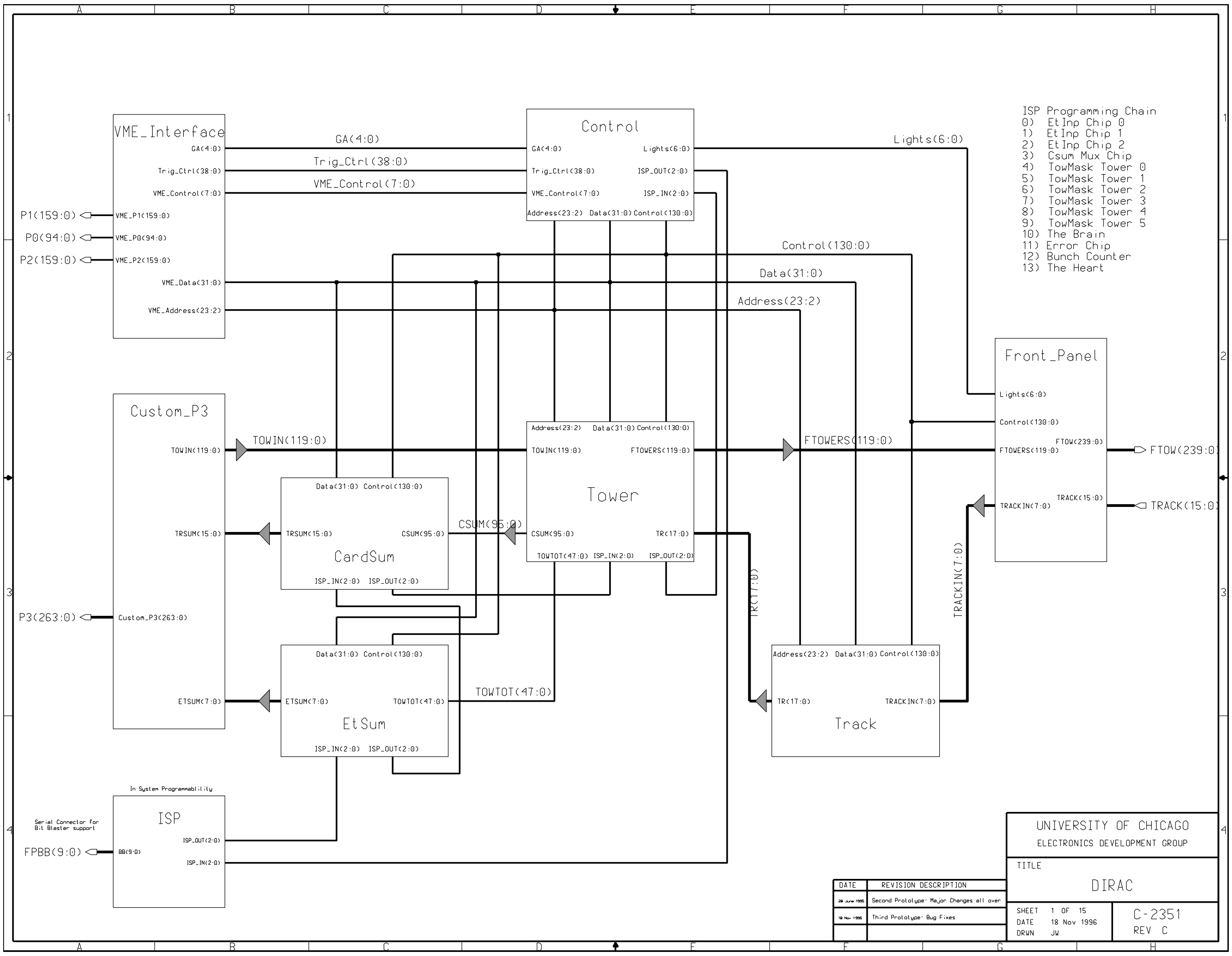

Figure 2: Top Level Diagram for DIRAC

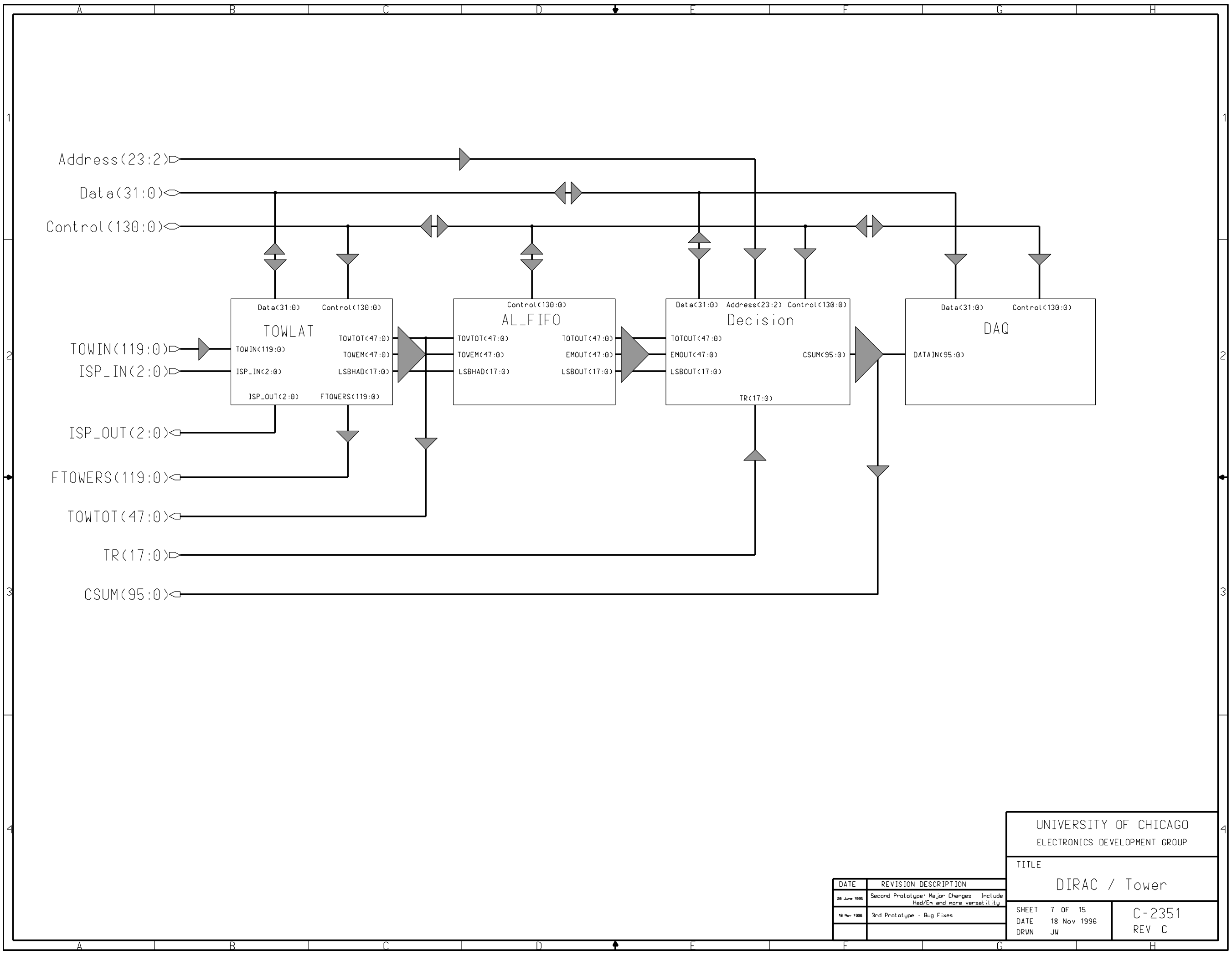

Figure 3: Tower Modules — 6 on each DIRAC Board

3.1 Global CDF Control Signals

All global[1] control signals are routed along P2 from the Tracer module in slot 2.[12] The following is an incomplete list of CDF reserved signals on the outer rows of P2. This list contains only those signals used by DIRAC.

- **CDF_L1A*** — Global L1 Accept
- **CDF_L1R*** — Global L1 Reject
- **CDF_B0*** — Bunch Zero
- **CDF_CLK** — Differential PECL Clock Signal (132 ns)
- **CDF_RESET*** — Master Reset
- **CDF_L1BA0*** — L1 Buffer Address Bit 0
- **CDF_L1BA1*** — L1 Buffer Address Bit 1
- **CDF_HALT*** — Stop Processing Events
- **CDF_ERROR*** — Tell the Trigger Supervisor Interface (TSI) that an Error occurred on DIRAC

Two other signals are relavent to DIRAC but not used directly.

- **CDF_L2A*** — Global L2 Accept
- **CDF_L2R*** — Global L2 Reject

CDF_L1A*, **CDF_L1R***, **CDF_L2A*** and **CDF_L2R*** are the global trigger decisions from the Trigger Supervisor Interface (TSI).[13][14] With a **CDF_L1A*** the TSI sends **CDF_L1B0*** and **CDF_L1B1*** designating in which of 4 buffers to store the event while waiting for a L2 decision. On a **CDF_L2A***, the VME Readout Controller (VRC) in every crate will readout the appropriate L2 buffer via VME. On a **CDF_L2R***, nothing is done on DIRAC, the buffer in question is merely available for another event. The TSI keeps track of the status of the buffers. The **CDF_CLK** signal has a period of 132 ns, regardless of the actual bunch spacing. The **CDF_RESET*** signal, generated by the TSI, tells DIRAC to reset its FIFOs. **CDF_RESET*** serves two purposes: 1) a method to re–synchronize the system without stopping the run and 2) initiating a run.[15] The **CDF_ERROR*** signal tells TSI that an error occurred on DIRAC. An error for the DIRAC board is sent only if one of the FIFOs on DIRAC becomes full. This could indicate either a loss of sync or a clock signal that did not get properly distributed on DIRAC.

[1]Global refers to all CDF Trigger and DAQ control signals.

3.2 VME_Interface

The **VME_Interface** allows DIRAC to communicate with other cards in a VME crate as a slave only using A32/D32/D32(BLT) protocol defined by the VME standard.[16] As specified in the VME64-P standard, Dirac recognizes only addresses corresponding to its slot number. For more information about the DIRAC VME interface please refer to appendix A.

3.3 Custom_P3 and The TowLat Chip

The 20 bits of energy in each tower come through the **CUSTOM_P3** module. The energies arrive at DIRAC as transverse energy and with the proper pedastal already subtracted. These 20 bits are divided into 10 bits of hadron transverse Energy (E_T) and 10 bits of electromagnetic E_T. One count corresponds to 0.125GeV for a full scale of 127.875GeV. These signals are grouped into a 120 bit bus called **TOWIN(119:0)**. This bus is fanned out in six 20 bit segments and routed to the **TOWLAT** section within the **Towers** module.(see Figure 3)

The **TOWLAT** section contains 6 EPLDs. These chips latch the input energies at a time $t_{\text{Calorimetry}}$. Each chip represents one trigger tower, receiving 10 bits of hadron and 10 bits of electromagnetic data. The functionality of the Tower Latch chip is fully described in a separate document.[10] The ability to mask off any individual section of a tower is provided. The masked towers will contribute 0 energy to the sum for that tower and will output a value of 0 to all other outputs. The two 10-bit energies are echoed (after masking and latching) to the **FTOWERS(119:0)** bus. The **FTOWERS(119:0)** bus is then sent to the Level 2(L2) DCAS boards (described later).[17] Eight of the 10 electromagnetic energy bits are used in later L1 trigger decisions. These 8 bits are bundled into the **TOWEM(47:0)** bus. The choice of which 8 bits are used is determined by the state the Tower Granularity bit in the control registers. One can either choose to have 0.250GeV or 0.500GeV resolution on both the **TOWEM** and **TOWTOT** energies. Finally a crude measure of the hadron energy is provided by **LSBHAD(17:0)**. This bus contains 3 bits per trigger tower that are either the 3 least significant bits (LSBs) or the next to 3 LSBs. The choice is made via a bit in the control registers. These bits are used to make a rough cut on hadron energy in a tower for electron triggers. The two 10-bit data words input to the **TOWLAT** section are summed to produce an 8-bit number that is the total energy in each trigger tower. In the event of an overflow in the sum, all bits are turned on, i.e. the energy is set to full–scale. The default resolution of the summed energy is 0.250GeV with a full scale of 63.75GeV, as mentioned earlier the resolution can be changed. The bus **TOWTOT(47:0)** is the collection of all 6 chips total energies.

Note that if one uses a granularity of 0.250 GeV (0.500 GeV) here in the **TOWLAT** section, to form the Dirac decision, then for the ETSUM section, the granularity of the 8 bit ETSUM word sent to CrateSum is 0.500 GeV (1.000 GeV), respectively. This change was done recently, anticipating that the (previous) full range on ETSUM still would not be comfortable for Level 2. This change was done on the EtInp chips, U31, U33, U34, program bigadder.gdf [2].

3.4 Alignment Fifo

The **TOWTOT(47:0)**, **TOWEM(47:0)** and **LSBHAD(17:0)** buses are fed to the **AL_FIFO** section for time aligning with the tracking trigger bits. The purpose of this FIFO is to make sure that the tracking information and the tower energy bits being examined are from the same event. For example, two parts of one event can get out of time because the tower energy is processed faster than the tracking information. This FIFO prevents such an occurence, by waiting a time t_{Tracking} after the bunch crossing to release the event for further processing. The outputs of this section are called **TOTOUT(47:0)**, **EMOUT(47:0)** and **LSBOUT(17:0)** and are sent to the **Decision** section.

3.5 Decisions, Decisions. . .

The **Decision** section receives the outputs of the **AL_FIFO** section. In addition, **Decision** accepts an 18 bit bus, called **TR(17:0)** from the **Track** section. Twelve (EM + TOT) SRAMs receive these 4 busses and make the trigger decision. For each trigger tower, one SRAM looks at the **TOWTOT** and **TR** busses and generates the 8 jet trigger decisions. The other SRAM examines the **TOWEM**, **LSBHAD** and **TR** busses and makes the 8 electromagnetic trigger decisions. These decisions are combined into the **CSUM(95:0)** bus. Note: There are 16 decisions per trigger tower for a total of 96 bits. However, there are only 13(ten 1–bit and three 2–bit) L1 triggers available for the global decision. The reduction from 16 to 13 is performed in the **CSUM** section. The **CSUM** bus is sent to two modules **CardSum** and **DAQ**.

A brief discussion of the physics behind the **Decision** section is in order. The jet decisions are based on the total energy in a given trigger tower. In addition, the existence of a track above a threshold is available on an individual tower basis. This allows, for the first time at CDF, the existence of a true jet trigger at L1. The electromagnetic decision is equally revolutionary. The electromagnetic (EM) decision is based on the amount of EM energy in the tower, the existence of a track pointing to a given trigger tower and the amount of hadronic energy in the tower. The crude hadronic energy information allows either a factor of 8 or 16 cut on Em/Had at level 1. The **LSBHAD** data is 3 bits from the hadronic energy for the tower. As described earlier it is either the 3 LSBs or the next to 3 LSBs, thus providing a measure of the amount of hadronic energy with a maximum of either $\frac{7}{8}$GeV or $1\frac{3}{4}$GeV and resolution of 0.125 or 0.25GeV, respectively (again a programmable parameter). The basic jet and electron triggers are easy to envision. A new interesting trigger for level 1 calorimetry may be for muons. One can cut on a window of energy (minimum ionizing) around a few GeV and ask for a track

[2] These chips can be reprogrammed, if needed.

Table 4: Front Panel Connector 1 – FP1

	Column	
Row	A	B
1	EM Tow 0 E_T0	EM Tow 1 E_T0
2	EM Tow 0 E_T1*	EM Tow 1 E_T1*
3	EM Tow 0 E_T2	EM Tow 1 E_T2
⋮	⋮	
38	HAD Tow 0 E_T8*	HAD Tow 1 E_T8*
39	HAD Tow 0 E_T9	HAD Tow 1 E_T9
40	HAD Tow 0 E_T9*	HAD Tow 1 E_T9*

pointing at this tower. This combined with the normal muon triggers available is likely to give an even stronger handle on muon triggering at level 1.

3.6 Data Acquisition (DAQ)

DAQ stores the event while waiting for a L2 decision. This section contains a L1 FIFO and a set of 4 L2 buffers for each set of 8 trigger decisions. The L1 FIFO is 256 deep, easily accomadating the maximum L1 decision time of 42 crossings (42×132 ns $= 5.544\ \mu$s). If a **CDF_L1R*** is received by Dirac, the next event in the L1 FIFO is strobed and ignored. If a **CDF_L1A*** is received, the next event is strobed out of the L1 FIFO and read into one of the 4 L2 buffers based on the state of **CDF_L1BA0*** and **CDF_L1BA1***. If a Level 2 reject is issued, nothing is done in the crate. The buffer corresponding to this event is marked as open by the TSI.[13][14] If a Level 2 accept is issued, the VRC contained in each crate accesses the appropriate L2 buffer via VME. The L2 buffers on Dirac are placed at the proper place in the memory map for access during readout by the VRC and TRACER.[3]

3.7 DIRAC to L2 connection

The module **Front_Panel** takes **FTOWERS(119:0)** and sends the 120 signals to the L2 DCAS boards via front panel connectors.[17] The signals are sent differentially, requiring 240 pins. This is done through three 80 pin Honda connectors located on the front panel (FP1, FP2, FP3). Tables 4, 5 and 6 show the pin assignments for these connectors, the number listed next to E_T is the bit number in **FTOW**. (Note that facing the connector in the board, pin 1 is the lower one and pin 40 the upper one). At time $t_{\mathrm{Calorimetry}} + 37$ns after the bunch crossing, the tower E_T information is valid on the cable to DCAS.

Table 5: Front Panel Connector 2 – FP2

	Column	
Row	A	B
1	EM Tow 2 E_T0	EM Tow 3 E_T0
2	EM Tow 2 E_T1*	EM Tow 3 E_T1*
3	EM Tow 2 E_T2	EM Tow 3 E_T2
⋮	⋮	
38	HAD Tow 2 E_T8*	HAD Tow 3 E_T8*
39	HAD Tow 2 E_T9	HAD Tow 3 E_T9
40	HAD Tow 2 E_T9*	HAD Tow 3 E_T9*

Table 6: Front Panel Connector 3 – FP3

	Column	
Row	A	B
1	EM Tow 4 E_T0	EM Tow 5 E_T0
2	EM Tow 4 E_T1*	EM Tow 5 E_T1*
3	EM Tow 4 E_T2	EM Tow 5 E_T2
⋮	⋮	
38	HAD Tow 4 E_T8*	HAD Tow 5 E_T8*
39	HAD Tow 4 E_T9	HAD Tow 5 E_T9
40	HAD Tow 4 E_T9*	HAD Tow 5 E_T9*

3.8 Tracking Information

As discussed earlier, XTRP sends 8 differential bits representing tracks found by the XFT in the 15° wedge covered by each DIRAC board. These differential signals are received by **Front_Panel** through front panel connector 4 (FP4). **Front Panel** then outputs the 8 single ended signals as **TRACKIN(7:0)**. **TRACKIN** is latched at time t_{Tracking} after the bunch crossing by the module **TRACK**. The module **TRACK** contains three 256×8 SRAMs that take in the 8 bits as an address and output which trigger towers pass each tracking trigger on the **TR(17:0)** bus. Since SRAMs are used for the tracking bits and in the **Decision** section, the final decision can be an arbitrary combination of the 8 tracking bits and the energy bits. This allows for a change in meaning of the 8 tracking bits. The mapping is downloaded at the beginning of each run.

3.9 Trigger Summary

The module **CardSum** accepts the 8+8(EM+Jet) trigger bits from each of the 6 towers on the **CSUM(95:0)** bus. This module consists of 5 Erasable Programmable Logic Devices (EPLD). The 96 bits are sent to 4 EPLDs which produce 2 bits output for each trigger type, for a total of 32 bits. There are 16 different triggers at this summary stage, 8 2–bit jet and 8 2–bit EM triggers. Each trigger summary tells whether 0, 1 , 2 or greater than 2 towers passed each trigger type. For trigger 0, bits 0 and 1 tell how many of the 6 towers passed that trigger, bits 2 and 3 tell how many passed trigger 1, etc.... Table 7 shows the meaning of each set of two bits.

These 32 bits are then sent to the CSUM_MUX EPLD. This chip takes the 8+8 2–bit triggers and passes 10 1–bit and 3 2–bit triggers to the CardSum backplane drivers. The CSUM_MUX Internal register in the memory map determines which of the 8+8 triggers are used. In the current configuration, the first 10 bits sent over the backplane to CrateSum(CS) are the 10 1–bit triggers and the next 6 bits are the 3 2–bit triggers, for a total of 16 bits, **CSUMOUT(15:0)**. The CSUM_MUX internal register is a 17–bit register and table 8 shows how the individual bits are multiplexed. Each 1–bit trigger now tells whether 0 trigger towers pass or at least one trigger tower passed the particular trigger. Some explanation for table 8 is neccessary. The first column represents the CSUM_MUX internal register bit. The second column shows which input is used for the On state of the bit from column 1. The third column is the same but for the Off state of the bit. Column four shows which of the 16 output lines, **CSUMOUT**, is driven by the selected input. There are two sets of triggers that are a selection out of 8 triggers to drive one output line. These are shown as a combination of columns 2 and 3 where the phrase "select xx" is used. This means that the triggers xx are selected based on the state of the 3 bits(an 8-1 multiplexer). If 000 (binary) is written to CSUM Register bits 8-10, then trigger Jet0 is selected on TRSUM bit8. If 011, then trigger Jet3. If 111, trigger EM3.

The 16–bit trigger summary word, **TRSUM(15:0)**, is driven on the backplane at

Table 7: Meaning of the 2 Bits Representing How Many Towers Passes a Trigger

Bit		How Many
1	0	Towers Passed
0	0	0
0	1	1
1	0	2
1	1	> 2

$t_{\text{Tracking}} + 215$ns after the bunch crossing, since tracking data arrives after Calorimeter Data. **TRSUM(15:0)** remains valid there for 1 clock cycle after that. **TRSUM** is sent to **Custom_P3** and out over the P3 backplane to CS. A minor timing requirement should be noted at this time. **TRSUM** has 49ns (2*132ns-215ns) to traverse the backplane and be latched by CS. The backplane propagation time should be under 5ns, leaving CS with 45ns to latch the data. This is easily accomplished with current technology.

Table 8: Interpretation of the CSUM_MUX Register

CSUM Register Bit	Bit State On	Bit State Off	**TRSUM** Bit Driven
0	Jet 0	EM 0	0
1	Jet 1	EM 1	1
2	Jet 2	EM 2	2
3	Jet 3	EM 3	3
4	Jet 4	EM 4	4
5	Jet 5	EM 5	5
6	Jet 6	EM 6	6
7	Jet 7	EM 7	7
$8-10$	Select EM 0-3, Jet 0-3		8
$11-13$	Select EM 4-7, Jet 4-7		9
14	Jet 5	EM 5	$10-11$
15	Jet 6	EM 6	$12-13$
16	Jet 7	EM 7	$14-15$

3.10 Sum E_T

EtSum accepts **TOWTOT(47:0)** and adds the six 8 bit tower E_Ts into one 8 bit E_T. This sum represents the total energy deposited in all 12 towers. If TOW_GRAN bit of Control1 is OFF, then if ΣE_T is greater than 128GeV (8 bits with a resolution of 0.5GeV), then all 8 bits are asserted to represent an overflow.[3] If TOW_GRAN bit of Control1 is ON, then if ΣE_T is greater than 256GeV (8 bits with a resolution of 1GeV), then all 8 bits are asserted to represent an overflow. Again, the meaning of the 8 bits sent to Cratesum depends on the TOW_GRAN bit of Control_1:

If TOW_GRAN is ON, the default choice, the 8 bits have a full scale of 256 GeV, resolution of 1 GeV.

If TOW_GRAN is OFF, the 8 bits have a full scale of 128 GeV, resolution of .5 GeV.

Four EPLDs are used to perform this summing in a bi-level process. The 3 EtInp chips sum 2 towers each and apply a threshold to all towers. The input energies into these chips have a LSB of 0.5 (0.25) GeV, for TOW_GRAN ON(OFF). This threshold is 0, 1, 2 or 3 bit shifts as defined in the 2–bit ET_THRESH register in Control register 1. For example, if the threshold is set to 2 bits, then any tower with energy below the 2nd bit(<4 counts) is not included in the sum. This would represent a threshold of 4×0.5 [4]GeV = 2GeV. This way one can correct for pedestals of 0, .5, 1 or 2 GeV. Table 9 lists the possible tower thresholds for the two possible values of tower granularity. The output of these chips into the EtOut chip is shifted up by one, for a LSB of 1 (0.5) GeV, for TOW_GRAN ON(OFF). The EtOut chip sums the outputs of the EtInp chips into the final 8–bit number. Again, the final output into CrateSum will have a LSB of 1 (0.5) GeV, for TOW_GRAN ON(OFF). **ETSUM** is latched $t_{\mathrm{Calorimetry}} + 264$ns after the bunch crossing by the backplane latch. **ETSUM** passes through **Custom_P3** and is valid on the backplane 14ns later, leaving 132ns-14ns=118ns to traverse the backplane and get latched on CS. Table Sum Granularity summarizes the granularity and full scale vales for each stage of the Sum E_T calculation for the two possible values of the tower granularity. The ranges for Cratesum and PreFRED are included for convenience of the reader.

Again, read the important note at the end of section 3.3.

3.11 In System Programmability

The ability to reprogram the functionality on a board of sufficient complexity is a must for a system to endure over a long period of time. We provide this functionality by allowing most of the EPLDs to be reprogrammed through the Bit Blaster (or Byte Blaster [5])

[3] The version of the summing firmware currently available (and used so far during the run) does not carry forward the saturation of a single tower through the summing (3/16/05 PJW).

[4] Again depending on the TOW_GRAN bit. This case is for a granularity of 0.50 GeV, bit ON.

[5] Altera's Byte Blaster can (and was) also be used to program the chain. It's MUCH faster then the Bit Blaster.

Table 9: Tower Thresholds (ET_THRESH) for Sum E_T Calculation

bit		Granularity (GeV)	
		0.25	0.50
		TOW_GRAN (Control 1)	
		0	1
1	0	E_T(GeV)$\geq$	
0	0	0.0	0.0
0	1	0.5	1.0
1	0	1.0	2.0
1	1	2.0	4.0

Table 10: Granularity and Full scale (in GeV) for stages of Sum E_T Calculation

		Granularity (GeV)			
		0.25		0.50	
		TOW_GRAN (Control 1)			
		0		1	
	Bits	lsb	FS	lsb	FS
DIRAC Input	10	0.125	127.875	0.125	127.875
Tower Sum (TOWTOT)	8	0.25	63.75	0.50	127.5
Two Tower (ETINP)	8	0.50	127.50	1.00	255.0
Six Tower (ETINP)	8	0.50	127.50	1.00	255.0
Wedge Sum (Cratesum)	10	0.50	511.50	1.00	1023.0
SUM E_T (PreFRED)	11	1.0	2047.0	2.0	4094.0
E_{Tx}/E_{Ty} (PreFRED)	10	1.0	1023.0	2.0	2046.0
$\not{E}_T{}^2$ GeV^2 (PreFRED)	16	$(1.0)^2$	$(255.0)^2$	$(2.0)^2$	$(511.0)^2$

interface. [11] [6] The **ISP** section contains the 10–pin Bit Blaster connecter on the front panel. This interface is a serial line strung through all chips in the chain. There are 4 signals, TDI, TDO, TMS and TCK involved in the ISP protocol.

- TDI - Test Data In, is the input data
- TDO - Test Data Out, is the output data(input to next chip)
- TMS - Test Mode Select
- TCK - Test Clock, is the clock to strobe data through the chain

Two busses are routed through many sections of the schematics, **ISP_IN(2:0)** and **ISP_OUT(2:0)**. **ISP_IN** is always the input lines to the chip(TDI, TMS, TCK). Likewise **ISP_OUT** are the outputs of the chip. The outputs of one chip are tied to the inputs of the next chip, hence the serial chain. Bits 1 and 2(TMS and TCK) are tied together for the whole board. Bit 0 is TDI for input and TDO for output. The programming chain is shown in the upper right corner of figure 2.

4 Addressing

Throughout the rest of the paper, the C language numerical base identification standard is used. Any number prefixed by “0x” is a hexidecimal number. Any number without a prefix is decimal.

Figure 4 shows the division of VME address space for DIRAC. Each DIRAC card uses a 16 Mbyte slice from the “Defined space” section. Bits A23:A02 address DIRAC’s internal memory. A23:A20 are decoded into 16 “Block”s of memory. Each “Block” is 256K 32–bit(Quad) words addressed sequentially by A19:A02. Block transfers are limited by VME specification IEEE 1014–1987 to 256 bytes(64 Quad words) and performed by incrementing, with every data strobe, A07:A02 from a pre–loaded base address on DIRAC. In hexadecimal, a full 32–bit address looks like **YYZAAAAA**, where

- **YY** — A31:A24 = <GA4:GA0>000 — Zone selector. 32 Zones in the address space. This selects a slot in the crate, <GA4:GA0>.
- **Z** — A23:A20 — “Block” selector. 16 “Blocks” on DIRAC.
- **AAAAA** — A19:A00 — Word selector. 256K Quad words per “Block”.

[6]The EtOut and Csum chips are not re-programmable, but they contain only simple summing logic.

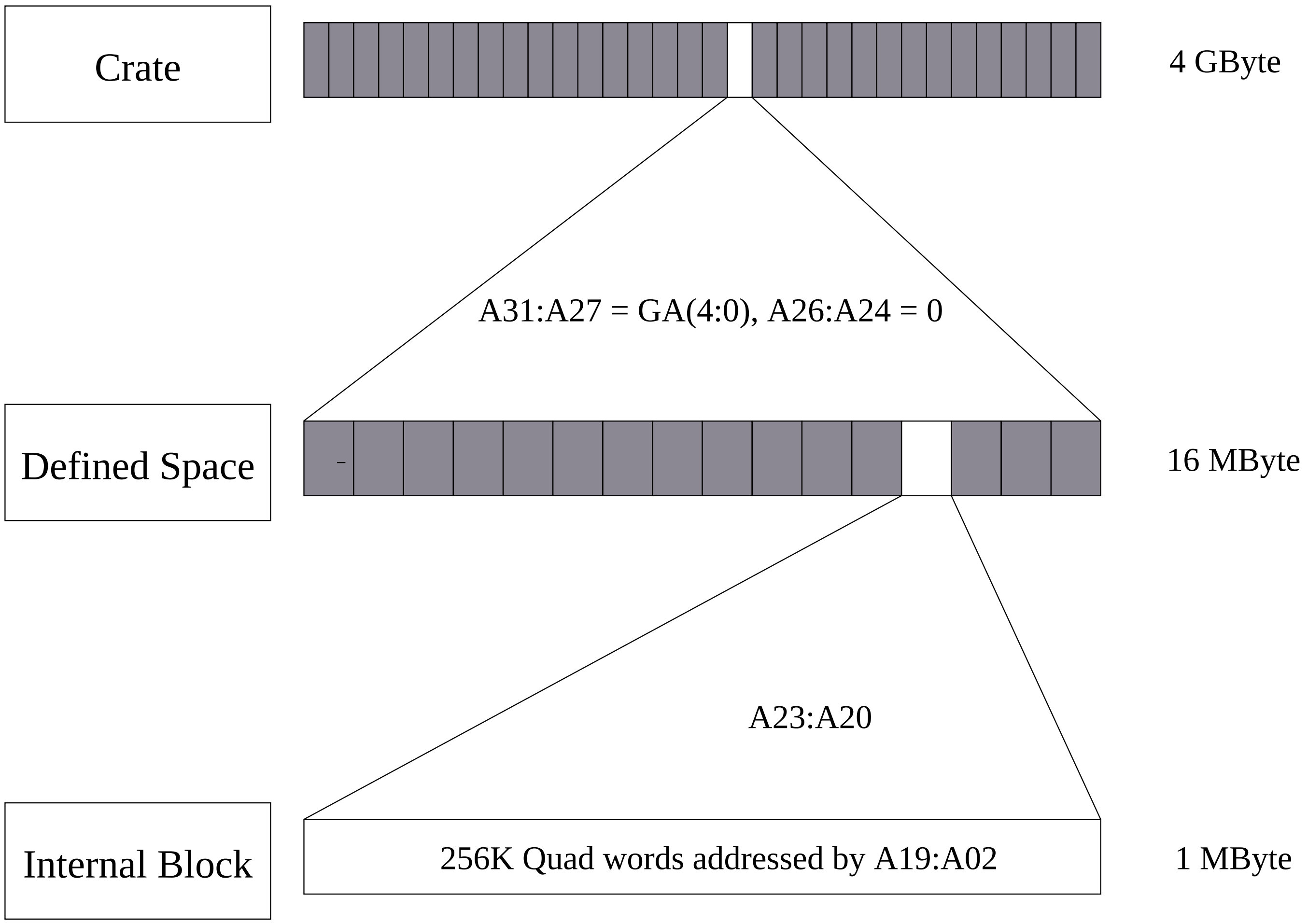

Figure 4: VME Address Space

Table 11: DIRAC Memory Look–Up Units

Memory	Towers	Size	“Block”	Addresses(A23:A00)	R/W
MLU_T_T	0–1	32K × 32	4	0x41FFFC : 0x400000	RW
MLU_T_T	2–3	32K × 32	5	0x51FFFC : 0x500000	RW
MLU_T_T	4–5	32K × 32	6	0x61FFFC : 0x600000	RW
TR_FAN_OUT	all	256 × 24	3	0x3003FC : 0x300000	RW

4.1 DIRAC MLUs

There are numerous Memory Look Ups(MLUs) on DIRAC. The location of the MLUs in DIRAC address space is specified in table 11. MLU_T_T refers to SRAMs in the **Decision** section and TR_FAN_OUT to SRAMs in the **Track** section. Each of these SRAMs must be downloaded at the beginning of every run with trigger decisions or tracking fan out information. Each address is readable for diagnostics. Writing to the memory is only allowed when in **LOAD** mode, and bits 6 and 7 of CONTROL1 are OFF (0).(**LOAD** mode is described later in this document)

The final decisions for which triggers pass given a tower energy threshold and tracking configuration is loaded into the MLU_T_T SRAMs. These are 6 towers $\times$ 2 types (HAD&EM) $\times$ 32K $\times$ 8 bits and organized as three 32K $\times$ 32 bits sections. Special note: Bits 8:15 and 24:31 of each section really only occupy 4K independent addresses. This is because the jet MLUs do not require the extra 3 bits for **LSB_HAD** and the contents must be the same for all values of these 3 bits. Hence the factor of 8. Therefore, when downloading the memories, care must be taken to repeat the 4K data for the whole 16K section of memory otherwise invalid data would be downloaded to the Jet MLUs. For Example, if you want to download 0x55 to tower 0(EM) and 0x45 to tower 1(Jet) for location 0x0 within the memory, you would write 0x4555 to location 0x400000. Now you move along and want to write 0x66 to location 0x402000, so you write 0x4566 to location 0x402000. Thus you overwrite the location in the Jet SRAM with the same value. If you were to write 0x0066 to location 0x402000, you would overwrite the Jet MLU with 0x00.

The mapping information for tracking triggers is loaded into the TR_FAN_OUT SRAMs. This memory is 6 towers $\times$ 256 $\times$ 4 bits and is organized as a 256 $\times$ 24 bits space within "Block" 3. This mapping depends explicitly on the meaning of the bits coming from the XTRP boards. This is programmable on the XTRP board and here as well.

4.2 DIRAC DAQ Memory

According to the CDF specification for readout, the first long word of every DAQ readout specifies the origination and bunch number for the event data.[18] This word consists of the 19 bits (the MSBits) specifying the board type, 5–bit geographical address, and 8 bits (the LSBits) for the bunch number.[18] Dirac follows this word with 3 long words of data containing the 16 trigger decisions for all 6 towers. The second readout word contains the trigger decisions for towers 0 and 1. The word is divided into 4 8 bit words (from LSB to MSB) containing the EM then HAD decisions for tower 0 followed by tower 1 in the same order. The remaining two words contain the information for towers 2–3 and towers 4–5, in the same configuration. This memory is divided into 4 distinct sections, one for each L2 buffer. As per the CDF specification, the locations of these buffers are shown in table 12.[3] During DAQ readout the buffer containing the L2 event is accessed using D32(BLT) (Note: this is only 4 long words in length)

.

Table 12: DIRAC DAQ Memory

DAQ Buffer	Size	"Block"	Addresses(A23:A00)	R/W
0	4×32	8	0x80000C : 0x800000	RO
1	4×32	9	0x90000C : 0x900000	RO
2	4×32	A	0xA0000C : 0xA00000	RO
3	4×32	B	0xB0000C : 0xB00000	RO

4.3 Data Path Access

These sections of VME memory are used to test the functionality of DIRAC. It allows access to virtually all points along the data path. The tests themselves are beyond the scope of this paper, but a description of the part on the board that the memory effects is given below. In **LOAD** mode, data path access is unlimited and any operation can be performed. For example, we can load all the towers with some energies and propagate the data through Dirac. In **RUN** mode, access is limited to reading from the data path only (spying). DIRAC will not allow you to write data that might interfere with normal data taking.

Table 13 summarizes the memory allocation for data path access to the Tower Latches. Table 14 describes the memory map for all other data path access.

Table 13: Tower Latch Data Path Access Map

Memory	Towers	Size	"Block"	Addresses(A23:A00)	Data Mask	R/W
LSB EM Tow	0–1	1	2	0x200000	0x001F001F	WO
MSB EM Tow	0–1	1	2	0x200004	0x001F001F	WO
LSB EM Tow	2–3	1	2	0x200008	0x001F001F	WO
MSB EM Tow	2–3	1	2	0x20000C	0x001F001F	WO
LSB EM Tow	4–5	1	2	0x200010	0x001F001F	WO
MSB EM Tow	4–5	1	2	0x200014	0x001F001F	WO
LSB HAD Tow	0–1	1	2	0x200020	0x001F001F	WO
MSB HAD Tow	0–1	1	2	0x200024	0x001F001F	WO
LSB HAD Tow	2–3	1	2	0x200028	0x001F001F	WO
MSB HAD Tow	2–3	1	2	0x20002C	0x001F001F	WO
LSB HAD Tow	4–5	1	2	0x200030	0x001F001F	WO
MSB HAD Tow	4–5	1	2	0x200034	0x001F001F	WO
LSB L2 EM Tow	0–1	1	2	0x200000	0x001F001F	RO
MSB L2 EM Tow	0–1	1	2	0x200004	0x001F001F	RO
LSB L2 EM Tow	2–3	1	2	0x200008	0x001F001F	RO
MSB L2 EM Tow	2–3	1	2	0x20000C	0x001F001F	RO
LSB L2 EM Tow	4–5	1	2	0x200010	0x001F001F	RO
MSB L2 EM Tow	4–5	1	2	0x200014	0x001F001F	RO
LSB L2 HAD Tow	0–1	1	2	0x200020	0x001F001F	RO
MSB L2 HAD Tow	0–1	1	2	0x200024	0x001F001F	RO
LSB L2 HAD Tow	2–3	1	2	0x200028	0x001F001F	RO
MSB L2 HAD Tow	2–3	1	2	0x20002C	0x001F001F	RO
LSB L2 HAD Tow	4–5	1	2	0x200030	0x001F001F	RO
MSB L2 HAD Tow	4–5	1	2	0x200034	0x001F001F	RO
LSB L1 EM Tow	0–1	1	2	0x200060	0x001F001F	RO
MSB L1 EM Tow	0–1	1	2	0x200064	0x000F000F	RO
LSB L1 EM Tow	2–3	1	2	0x200068	0x001F001F	RO
MSB L1 EM Tow	2–3	1	2	0x20006C	0x000F000F	RO
LSB L1 EM Tow	4–5	1	2	0x200070	0x001F001F	RO
MSB L1 EM Tow	4–5	1	2	0x200074	0x000F000F	RO
LSB L1 Tot Ene	0–1	1	2	0x200040	0x001F001F	RO
MSB L1 Tot Ene	0–1	1	2	0x200044	0x001F001F	RO
LSB L1 Tot Ene	2–3	1	2	0x200048	0x001F001F	RO
MSB L1 Tot Ene	2–3	1	2	0x20004C	0x001F001F	RO
LSB L1 Tot Ene	4–5	1	2	0x200050	0x001F001F	RO
MSB L1 Tot Ene	4–5	1	2	0x200054	0x001F001F	RO

Table 14: Other Data Path Access Map

Memory	Towers	Size	“Block”	Addresses(A23:A00)	Data Mask	R/W
VME_Loop	–	256×8	3	0x3003FC : 0x30000	0x000000FF	RO
TR_DIAG	–	1×8	7	0x700000	0x000000FF	WO
TRIN_FIFO	–	1×8	7	0x700000	0x000000FF	RO
L1_FIFO	0–1	256×32	2	0x2013FC : 0x201000	0xFFFFFFFF	WO
L1_FIFO	2–3	256×32	2	0x2023FC : 0x202000	0xFFFFFFFF	WO
L1_FIFO	4–5	256×32	2	0x2033FC : 0x203000	0xFFFFFFFF	WO
Static AL_FIFO	0–1	1×32	2	0x204000	0xFFFFFFFF	RO
Static AL_FIFO	2–3	1×32	2	0x205000	0xFFFFFFFF	RO
Static AL_FIFO	4–5	1×32	2	0x206000	0xFFFFFFFF	RO
Strobe AL_FIFO	0–1	256×32	2	0x2073FC : 0x207000	0xFFFFFFFF	RO
Strobe AL_FIFO	2–3	256×32	2	0x2083FC : 0x208000	0xFFFFFFFF	RO
Strobe AL_FIFO	4–5	256×32	2	0x2093FC : 0x209000	0xFFFFFFFF	RO
Csum	–	1×32	2	0x700004	0xFFFFFFFF	RO
Csum MUX Int Reg	–	1×17	2	0x700008	0x0001FFFF	WO
Csum MUX Ouputs	–	1×16	2	0x70000C	0x0000FFFF	RO
EtSum	–	1×8	2	0x700010	0x000000FF	RW

The Tower Latch memory is not straight–forward. This section controls data access to the Tower Latch chips in **TOWLAT**. A user is allowed to write data into the latches via the write–only(WO) registers **EM** and **HAD Tow**. By writing data to these registers, one can provide artificial data to the rest of the L1 trigger as well as the L2 data path. The LSB and MSB refer to the least and most significant 5 bits. The tower energies are all 10 bit numbers, so to write a value to the EM tower, one must write the lower and upper 5 bits separately. This idea is the same for all aspects of the Tower Latch chips. Each set of 5 bits has its own clock, so it is possible to download unique data to all towers. The register titles **L2 EM** and **HAD Tow** are read–only(RO) and allow access to the data going to L2 from the Tower Latches. The **L1 EM Tow** registers allow reading the Dirac EM energies. Finally, the **L1 Tot Ene** section contains the total energy for a tower, what has been referred to as the jet energies. The **LSBHAD** data is also packed into both the **L1 EM Tow** and the **L1 Tot Ene**. The packing of this data is discussed in reference [10].

The rest of the memory is much easier to understand, so you can relax now, right after you re–read the previous paragraph a few times. **TR_Diag** writes to the diagnostic register in the **Track** section. This register lets the user provide fake input to the tracking section, bypassing the tracking front panel connector. **TRIN_FIFO** reads out the tracking FIFO. **VME_Loop** allows you to check a pair of buffers used within **Track**. **L1_FIFO** provides the ability to write data into the L1 FIFO's. Only 2 FIFO's can be written to at any given time, since the FIFO clock is a global clock. This is sufficient to test these chips.

There are 2 methods to read the alignment FIFO's in the **AL_FIFO** section. One strobes the data out of the FIFO and the other merely spys on the outputs of the FIFO. The corresponding registers in table 14 are obvious. There are no data buffers directly connected to the FIFOs in **AL_FIFO**. In order to read data from these FIFOs, Dirac uses the data buffers in the **Decision** block. This means the Decision(**MLU_T_T**) SRAMs must be downloaded with a decodable pattern to determine the state of the **AL_FIFO** outputs. The most common pattern to use should be an echo of the lower 8 address bits (In the memory count from 0-255 then repeat to fill the memory up).

The **Csum** register is dual–purpose. When reading from this register, you read from the trigger summing chips, thus getting the output of the board. When writing, you write into a set of input latches in the Csum Multiplexer chip. This allows the user to fake data going off Dirac and to test the Csum Multiplexer chip. The two **Csum MUX** registers allow manipulation of the Csum Multiplexer chip. The **Csum MUX Int Reg** determines how the chip multiplexes the 16 2–bit trigger summaries to the 10 1–bit and 3 2–bit summaries that actually are sent to CrateSum. The actual decoding of this register was previously described in the trigger summary section. Of course, one is allowed to spy on the outputs of the multiplexer through **Csum MUX Outputs**. Finally, the **EtSum** register allows reading the output of the final energy summing chip. Additionally, one can write data to the backplane latch that drives data to CrateSum.

4.4 Control and Strobe Registers

This memory is used to control the overall operation of DIRAC. A listing of all control registers is contained in table 15. The descriptions of the various bits of the control and strobe words are contained in tables 16, 17, 18 and 19. The **Module ID** memory is for an ID prom that will contain a "special" ASCII code to identify this board as a DIRAC board. The **Bunch Offset** contains a 3 bit value designating the postition of Dirac in the L1 pipeline. This value is only 3 bits because the bunch counter chips in the **Control** section must be programmed within 8 CDF clocks of the real position. The **Full** and **Empty Flag** registers access data in the Error chip on Dirac. This chip monitors the empty or full state of all Dirac FIFOs. If a chip becomes full, an error signal is sent to the master control chip (Brain) and an appropriate action is taken. These registers are read–writable, allowing the user to mask off any FIFO's Full or Empty status. This masking ability is neccessary to allow Dirac to function with a "Bad" channel. The Empty flags are fed to the global timing chip(Heart) to handle the special latency timing of the Dirac FIFOs. There are 6 L1 FIFOs storing trigger decisions and 1 L1 FIFO for the bunch counter; the extra bit is a spare. There are also 6 alignment FIFOs storing energy data and 1 storing **LSBHAD** data. The **Track** FIFO flags are the 8th bit of the **AL** flags. The packing of the data during reading of the flags is discussed in note 9999.[10] The **EM** and **HAD Masks** registers allow the user to mask off any Dirac tower energies. The value output from the **TOWLAT** section for the masked tower is zero and does not contribute to the total energy output of Dirac.

Regarding Control_0 note that to write to the MLU memories, one must be in LOAD mode and Bit 6 and 7 of Control_0 must be off (0). Also to RESET Dirac via VME, that is using the bit0 of Control0 strobe, the board must also be in LOAD mode. (Of course in normal operation the board receives the CDF reset signal while in RUN mode, and acts appropriately, by clearing error flags and fifos).

Table 15: Control and Strobe Words

Memory	Size	“Block”	Addresses(A23:A00)	Data Mask	Offset	R/W
Control_0	1×10	0	0x000000	0xFFC00000	22	RW
Control_1	1×10	0	0x000004	0xFFC00000	22	RW
Strobes	1×9	0	0x000008	0x7FC00000	23	WO
Delays	1×10	0	0x00000C	0xFFC00000	22	RW
Module ID	32×8	1	0x10007C : 0x100000	0xFF000000	24	RO
Bunch Offset	1×3	7	0x700020	0x00000007	0	WO
L1 Full Flags	1×8	7	0x700030	0x000000FF	0	RW
AL Full Flags	1×8	7	0x700034	0x000000FF	0	RW
L1 Empty Flags	1×8	7	0x700038	0x000000FF	0	RW
AL Empty Flags	1×8	7	0x70003C	0x000000FF	0	RW
EM Masks	1×6	7	0x000010	0x0FC00000	22	RW
HAD Masks	1×6	7	0x000014	0x0FC00000	22	RW

Table 16: Control_0 Register

Bit				
Rel	Abs	Bit Name	Description	RW
0	22	RESET	Board level reset	WO
1	23	LOAD	ON – **LOAD** mode—Diagnostic mode OFF – **RUN** mode—Data Taking mode	RW
2	24	CLK_MODE	ON — use CDF clock from P2 OFF — use VME SYSCLK(halved)	RW
3	25	FTOW_EN	Enable front panel L2 drivers	RW
4	26	TRACK_SRC	ON — use front panel connector as the source of tracking information OFF — use tracking diagnostics register to fake the front panel input	RW
5	27	SPARE	Spare DIRAC Bit	RW
6	28	MLU_FIFO_EN	Enable all MLUs and FIFOs while in **LOAD** mode	RW
7	29	MLU_READ_MODE	Place all MLUs in Read Mode while in **LOAD** mode	RW
8	30	AUX_SUM_EN0	Enable AUX Endwall Sum 5+12=5	RW
9	31	AUX_SUM_EN1	Enable AUX Endwall Sum 7+12=7	RW

Table 17: Control_1 Register

Bit				
Rel	Abs	Bit Name	Description	R/W
0	22	L1_ACC	Data is written from L1 FIFO to L2 DAQ buffer specified by **L1_ADDR**(see below) This is used to fake a **L1A** during diagnostics	WO
2:1	24:23	L1_ADDR	Address to store event for a fake L1 Accept	RW
4:3	26:25	ET_THRESH	Energy threshold for EtSum If a towers energy is below this value in bits, then the tower does not contribute to the sum. (See Table 9 for details)	RW
5	27	ETSUM_EN	Enable Backplane drivers for **EtSum**	RW
6	28	CSUM_EN	Enable Backplane drivers for **CardSum**	RW
7	29	HAD_EM_SEL	Select which bits determine the **LSBHAD** Energy. ON — bits (3:1) of HAD Energy (0.250 GeV lsb) OFF — bits (2:0) of HAD Energy (0.125 GeV lsb)	RW
8	30	TOW_GRAN	Set the Tower Energy Granularity for L1 Decisions. ON — 0.500 GeV Least Count OFF — 0.250 GeV Least Count	RW
9	31	AUX_ENABLE	Enable Tower Data coming from AUX	RW

The **Strobes** word (table 18) is used to issue a clock signal to a specific section of Dirac. This is accomplished by writing to the "Brain" chip which passes the data to the "Heart" chip. The "Heart" chip then issues the appropriate clock(s). Usually a single clock is issued, but it is possible to clock multiple sections of Dirac. The **Delays** word (table 19) specifies the delay from the interaction for data to arrive at Dirac. Two times are needed, one for calorimetry energies and another for tracking information ($t_{\mathrm{Calorimetry}}$ and t_{Tracking}, respectively). The exact delays can be determined from the following formulae,

$$\begin{aligned} t_{\mathrm{Calorimetry}}\,(\mathrm{ns}) &= (\mathbf{CALOR\_DEL} + 12) * 132 + (\mathbf{CALOR\_TAP}) * 25 + \tau_{\mathrm{PLD}} \\ t_{\mathrm{Tracking}}\,(\mathrm{ns}) &= (\mathbf{TRACK\_DEL} + 15) * 132 + (\mathbf{TRACK\_TAP}) * 25 + \tau_{\mathrm{PLD}} \end{aligned}$$

where τ_{PLD} is the delay imposed by the timing control logic and is approximately 32 ns for a EPM7128LC84–10(the chip used for the "Heart"). The range of t_{Tracking} is 2012 to 3011 ns and $t_{\mathrm{Calorimetry}}$ is 1616 to 2615 ns.

With these assignments Dirac obeys the specifications of the current L1 timing diagram. Note that Tracking Data must arrive at Dirac at least 3 clocks AFTER Tower Data. Therefore when adjusting the previous delays, it's understood that this condition is always met. Failure to do so will cause Tower and Tracking data misalignments in Dirac. A final practical note on adjusting B0 Offset: Currently this should be set equal to the the number of TRACK_DEL, so that the first event processed by Dirac, presumably B0 after the previous timing adjustments, will have the 00 marker for the B0 counter.

Table 18: Strobes Register

Bit				
Rel	abs	Bit Name	Description	R/W
0	23	TOWER_IN	Latch the data into the Tower Latch chip from the AUX card	WO
1	24	ALFIFO_IN	Strobe data into **AL_FIFO** and input latches of **EtSum**	WO
2	25	ETSUM_IN	Strobe data into of **EtSum** section	WO
3	26	ETSUM_OUT	Strobe data out of **EtSum** and across P3 to CS	WO
4	27	ALFIFO_OUT	Strobe data out of **AL_FIFO**	WO
5	28	TRACK_IN	Clock the data into the tracking FIFO from either the front panel or the diagnostic register. Source is determined by the state of the **TRACK_SRC** bit of **Control_0**	WO
6	29	L1FIFO_IN	Strobe data into **DAQ** L1 FIFO	WO
7	30	CSUM_IN	Strobe data into **CardSum**	WO
8	31	L1FIFO_OUT	Strobe data out of the **DAQ** L1 FIFO	WO

Table 19: Delays Register

Bit				
Rel	Abs	Bit Name	Description	R/W
2:0	24:22	CALOR_DEL	Interaction to $t_{\text{Calorimetry}}$ delay — coarse	R/W
4:3	26:25	CALOR_TAP	Interaction to $t_{\text{Calorimetry}}$ delay — fine	R/W
7:5	29:27	TRACK_DEL	Interaction to t_{Track} delay — coarse	R/W
9:8	31:30	TRACK_TAP	Interaction to t_{Track} delay — fine	R/W

Table 20: Address Modifiers

AM(Hex)	Address Bits	Transfer Type
09	32	Extended non–privileged data access
0B	32	Extended non–privileged block transfer
0D	32	Extended supervisory data access
0F	32	Extended supervisory block transfer

APPENDIX

A DIRAC VME Interface

DIRAC provides the circuitry for A32 addressing, D32 and D32(BLT) data transfers through the VME bus protocol. DIRAC allows address pipelining and responds to both supervisory and non-privileged accesses. The address modifier bits and the transfer types supported by DIRAC are summarized in table 20. Since all transfers are quad byte and unaligned data transfers are not supported, LWORD* is low, address bit A01 is ignored and data strobes DS0* and DS1* are ORed together. A module is selected in the VME crate by using a 5–bit geographical address GA(4:0) hardwired on the P1 backplane. The address defines a 128 Mbyte zone from the full 4 Gbyte VME address space (please refer to figure 4). This zone is addressed if A31:A27 equals GA4:GA0. DIRAC uses only 16 Mbyte(A26:A24 = 0) of the 128 Mbyte zone.

B Run Startup: Cold Start

Before **EVERY** Cold Start, the following registers must be configured with valid data. For this the board needs to be set into LOAD mode (bit 1 of Control0 ON) and bits 6 and 7 of Control1 OFF, and a VME RESET issued (bit 0 of Control0). (This last one is just to make sure that previous error conditions are erased).

- MLU_T_T (All 3 sets of SRAMs) — The Trigger Tables. Each SRAM has 32Kx(32 bits), for a total of 3x32K words, each word a 4 byte one. So a total of 384 Kbytes/board.
- TR_FAN_OUT — The Tracking Fan–Out Information. 256x(24 bits).
- Csum MUX Int Reg — The CSUM MUX Internal Register to determine which trigger bits get passed to CrateSum. 17 bits.
- Control_0 — Various DIRAC State Settings. 10 bits.
- Control_1 — Various DIRAC State Settings. 10 bits.
- Delays — The Timing Delays for Pipeline Stages. 10 bits.
- Bunch Offset — The Number of Bunches DIRAC is in the Pipeline. 7 bits.
- EM Masks — Masking Register for EM Energies. 6 bits
- HAD Masks — Masking Register for HAD Energies. 6 bits

If any of the above registers are neglected at run startup, any data taken can be considered invalid.

A total of about 400 Kbytes/board needs to be downloaded. If within a crate there is no Φ dependence and the only η dependence is the difference between PLUG and CENTRAL, then a total of about 800 Kbytes/crate needs to be downloaded.

The following registers are for masking bad channels inside Dirac. In a normally GOOD working Dirac, they assume default values such that no L1 or AL Fifos are masked. So for a normally working GOOD board, they need NOT to be downloaded.

- L1 Full Flags — Masking Register for FIFO Full Flags.
- AL Full Flags — Masking Register for FIFO Full Flags.
- L1 Empty Flags — Masking Register for FIFO Empty Flags.
- AL Empty Flags — Masking Register for FIFO Empty Flags.

C Known "features"

The following "features" were added after "careful" consideration:

- Module ID.

  When Addresses 0x10007C:0x100000 are read, the upper 8 bits of data of each of these addresses have the ASCII equivalents of: 00XX 005 DIRAC. Beautiful, ahn ? The only thing is that instead of reading the board serial number, you get XX. But when you read the DAQ buffers, the serial number indeed is there, in its right place. Getting the serial number right also in the module ID would have required to route more input lines into one chip (Control) that did not have more input pins available. Either way, who needs the serial numbers on the board ID, anyway?

- Serial Number switches.

  When setting the switches that set the board serial number (switches S1), the following happens: to set bit 6 (staring counting at 0) must set switch number 9 (starting counting at 1). To set bit 7, must set switch number 7; to set bit 8, switch number 8. This happened because on the schematics the switch counting goes as 1 2 3 4 5 6 9 7 8 10, instead of 1 2 3 4 5 6 7 8 9 10. The reasons for this are "too long to explain here"...

- Sum Saturation in Sum E_T Calculation (3/16/05)

  The current version of the firmware for the ETINP chip (in use since the begining of the run) does not correctly handle the case of a single tower at full scale. In each stage of the summing, if the input or the sum is at full scale, the output is supposed to be set to full scale. However, the first summing stage of the Sum E_T path which adds two towers together does not handle the case of one of the single tower inputs being at full scale. This is the result of dropping the lsb out of the sum and shifting the full scale up. This bug is independent of selected tower granularity. In principle this is simple to fix but it may not be practical because of the age of the EPLDs used and the required software to modify the firmware.

References

[1] Shaw, Teresa M. *Specification for ADC/Memory Module.* Particle Physics Division. CDF Upgrade Project. 6/11/97.

[2] Frisch, H.J. and Wilson, P.J. *Trigger tower organization and summing in η–ϕ space for Run II and beyond.* CDF/DOC/TRIGGER/CDFR/2045.

[3] Shaw, T and Sullivan, Greg. *A Standard Front–End and Trigger VMEBus Based Readout Crate for the CDF Upgrade.* CDF/DOC/TRIGGER/CDFR/2388.

[4] VIPA *VME-P Document.* NIM/VME-P 9612.

[5] Wilson, P.J. *J3 Backplane and Auxiliary Boards for the Level1 Calorimeter Trigger Upgrade (Run II).* CDF Trigger Note 92.

[6] The P3 Auxilliary(AUX) card was designed and built by Peter Wilson.

[7] Toback, Dave. *Crate Sum - Upgrade for Run II.* CDF Trigger Note 91.

[8] Freeman, Jim, et al. *The Run II eXtrememly Fast Tracker (XFT)* http://cdfsga.fnal.gov/upgrades/daq_trig/xft/xft_home.html.

[9] Altera Corporation. *Designing for In–System Programmability in MAX 7000S Devices.* June 1995, Application Brief 145.

[10] Wahl, John E. *The Anatomy of a Dirac.* CDF Note, CDF/DOC/TRIGGER/4231.

[11] Altera Corporation. *Bit Blaster Serial Download Cable Data Sheet.* Altera Corporation Product Data Sheet.

[12] Shaw, Teresa M. *Specification for TRigger And Clock + Event Readout Module (TRACER)* 8/5/1996, Fermilab Experiment Electronic Systems Group.

[13] CDF Group, Yale University. *Trigger Supervisor Protocols for Run II* August 18, 1996, Version 1.2.

[14] CDF DAQ Group. *The CDF DAQ Manager's Handbook.* August 18, 1996, Draft #3.

[15] Wahl, John and Wilson, Peter J. *Digital Receive And Compare (DIRAC) Board Startup Timing Diagram.* June 11, 1995, CDF Trigger Note 95.

[16] *The VMEbus Specification.* Revision C.1, Motorola, Inc. IEEE P1014 and IEC 47b.

[17] Sarah Truitt is designing the Digital Compare and Sum (DCAS) boards for the Level 2 trigger.

[18] Ohl, K, Wahl, J, and Wilson, P. *Specification of DAQ/Trigger Synchronization Checking Protocol and VME board Header word.* Version 2.0. CDF/DOC/TRIGGER/CDFR/3145.