Draft
TGN-98
CDF/TRIGGER/CDFR/xxx
Version 1.2
November 25, 1998

Error Detection and Handling for the Calorimeter Trigger

A. Amadon, R. Culbertson, D. Toback, J. Wahl
Univ. of Chicago
T. Shaw, P.J. Wilson
Fermilab
M. Tecchio, S. Truitt
Univ. of Michigan

1 Introduction

This note describes the error handling of the calorimeter trigger path. For the Level 1 trigger this includes the ADMEM, DIRAC (L1AUX), Cratesum, and PreFRED modules. The Level 2 portion consists of the DCAS, LOCOS, (CLIQUE), and Cluster List boards. The errors are broken down into those detected in hardware on the boards and those detected by monitoring software (TRIGMON).

2 Hardware detected errors

The ADMEM, DCAS, LOCOS, CLIQUE, and Cluster List boards have no hardware error detection. None of these boards makes a connection to the CDF_ERROR backplane signal.

The DIRAC, Cratesum and PreFRED all have only one hardware detected error condition. This is if any of the L1 storage FIFOs or data alignment FIFOs has a Full Flag (FF) set. In all cases the state of each FIFO FF is latched into a register with each CDF_CLK. The output of these registers is ORed together and sets a FLIP-FLOP which sets a board error register (1bit) and the CDF_ERROR line on the backplane. To protect against recurring problems on one DIRAC channel, the output of the FF registers can be masked to stop the generation of a board error and subsequent CDF_ERROR. The Flip-Flop is reset by the CDF_RECOVER signal from the P2 backplane. It can also be reset via VME write to a control register. Table 1 summarizes the hardware error detection.

If the FIFO FF comes on for one clock cycle this will set the Board Error register until the system resets but the FIFO FF Register may have reset on the next clock cycle. If the FF stays on then the Board register and the FIFO FF register will be set. Note that if the FIFO FF comes on then 256 clock cycles of data have been written to the FIFO without a read so up to 33usec of data may be effected. If any of these have been passed by L2 the

Table 1: Hardware Errors

Board	Error	CDF_ERROR	FIFO Registers	Board Register
DIRAC	FIFO Full	Maskable	latched on CDF_CLK	Reset by CDF_RECOVER
CS	FIFO Full	Maskable	latched on CDF_CLK	Reset by CDF_RECOVER
PreFRED	FIFO Full	Maskable	latched on CDF_CLK	Reset by CDF_RECOVER

trigger decision can be checked in software. However, we believe that any data which has not been read out in the case of this error should be pitched.

3 Software Error Detection

Most errors in the calorimeter trigger data path will be detected by online monitoring by comparing the data readout from the modules along the path with the results of simulation. Most events will not have the full readout list, some events will (see below). Figure 1 shows a block diagram of the available calorimeter readout and trigger path including the ADMEM, DIRAC, Cratesum, PreFRED, and XTRP for the Level 1 path. For Level 2, the DCAS, LOCOS, and Cluster List boards are shown, the CLIQUE controls the cluster finding but none of the cluster data passes through this board. In each board the box labeled *DAQ* indicates where data will be intercepted for readout through a L1 storage pipeline (or FIFO) and L2 storage buffers. Boxes labeled *PROC* indicate some processing is taking place.

The error handling for the L1 chain will be handled in 5 tests for the ΣE_T decision path and a parallel 4 tests on the Tower Trigger based path.The following software tests are possible in the chain. Test 1 validates the data for all the L1 and L2 triggers. Tests 2-5 test the L1 ΣE_T path,tests 6-9 test the L1 Tower trigger path and tests 10-14 test the L2 cluster finder.

1. ADMEM to DIRAC to DCAS: compare full 15 bit ADMEM calorimeter tower readout with MASKED trigger tower readout from DCAS card. Verifies trigger input data and DIRAC masking.

2. DIRAC ΣE_T algorithm: compare simulation run on data readout from DCAS with board level ΣE_T readout from Cratesum inputs. Verifies 10 to 8 bit masking, DIRAC summing and backplane connection.

3. Cratesum ΣE_T algorithm: compare simulation run on CRATESUM input data with CRATESUM output data.

4. PreFRED ΣE_T algorithm: compare simulation run on CRATESUM output data with PreFRED output data. Also tests CRATESUM to PreFRED cable connection. Can test PreFRED ΣE_T and $\not{E}_T$ algorithm plus threshold application.

5. Test PreFRED to FRED connection by comparing 4 ΣE_T and $\not{E}_T$ trigger bits from PreFRED readout with trigger bits from FRED readout.

6. ADMEM to DIRAC to DCAS: same as above.

7. DIRAC Tower Trigger algorithm: compare simulation run on data readout from DCAS and XTRP (the XTRP track list and the DIRAC tracking bits are read out on every event) with tower level decision information readout from the DIRAC card. Verifies 10 to 8 bit masking, HAD/EM bits, trigger decision look ups.

8. Cratesum Tower Trigger algorithm: compare simulation run on DIRAC tower level decisions with CRATESUM output data. Tests DIRAC tower trigger summary logic, Backplane connection in addition to CRATESUM logic.

9. PreFRED Tower Trigger algorithm: compare simulation run on CRATESUM output data with PreFRED output data. Also tests CRATESUM to PreFRED cable connection.

10. Test PreFRED to FRED connection by comparing 20 tower trigger bits from PreFRED readout with trigger bits from FRED readout.

11. Compare DCAS partial cluster sums, readout from LOCOS, with simulation of cluster finding run on trigger tower data readout from DCAS board.

12. Compare LOCOS crate cluster sums, readout from LOCOS, with simulation of cluster run on partial sums from input of LOCOS.

13. Compare cluster list, readout from Cluster List board, with simulation based on LOCOS crate level sums.

14. Compare cluster list readout from Cluster List board with simulation of the full cluster finding run on trigger tower list readout from DCAS cards.

The cluster-finder tests might be executed by first comparing the final cluster list to simulation. If this passes move on the the next event. If it fails go back to the board level tests to determine what is at fault.

Table 2 is a non-exhaustive list of errors including type of detection, severity of error and recovery. The severity is broken into the case of single occurrence and multiple occurrences in the same board(s) in a short period of time (probably a few minutes). The severity level uses the following definitions:

WARNING Record the error, count how many per run,

MINOR Initiate HALT-RECOVER-RUN sequence,

SEVERE Initiate HALT-test-(RECOVER or End Run and fix),

FATAL End run and fix the problem.

Table 2: Partial list of errors for the Calorimeter Trigger Chain. The severity refers to response to a single occurrence of the error or multiple occurrences within a few minutes. The key to the severity levels appears in the text. The responses are what to do in the Severe or Fatal case.

Error	Hardware Tested	Detection Test	Severity Single	Severity Multiple	Recovery
Full FIFO	DIRAC,CS,PreFRED	Hardware	Minor	Severe	Replace card?
Trigger Data Bit(s) error (1chan)	ADMEM, CABLE, DIRAC, DCAS	Software 1	Warning	Severe or Fatal	Mask Tower
Trigger Data Bit(s) error (many chan)	ADMEM, CABLE, DIRAC, DCAS	Software 1	Warning	Fatal	Mask Towers or Replace Board
DIRAC ΣE_T Stuck all On	DIRAC	Software 2	Warning	Fatal	Replace DIRAC Card
Cratesum ΣE_T Stuck all On	CRATESUM	Software 3	Warning	Fatal	Replace Cratesum Card
PreFRED ΣE_T Stuck all On	PreFRED	Software 4	Warning	Fatal	Replace PreFRED Card or Cable
DCAS Partial Cluster Sum Bad	DCAS	Software 11	Minor	Fatal	Replace DCAS
Cluster List disagrees with LOCOS output	Cluster List	Software 13	Minor	Fatal	Replace Cluster or Cable
etc					

4 Diagnostic Readout List

The current plan is to read out the full list only on the events triggered for the purpose of trigger debugging. These events will not be rejected by any stage of the trigger and can be directed to the monitor program as a priority. Which boards are read out on every event can be changed with little effort. Two readout lists will be active during data taking. The normal physics list includes the information that is recorded on each event. The diagnostic readout list is used on a small fraction of diagnostic triggers. These events are marked but not rejected by the trigger. They can be directed to the monitoring programs with priority over normal events.

Table 3 lists what is read out from each of the boards in the calorimeter readout and trigger chain. In the case of the CLUSTER List, DCAS, and PreFRED modules the readout list is not expected to change for diagnostic readout. The current plan is that DIRAC and Cratesum will only be on the diagnostic readout list. The LOCOS list is preliminary (Monica doesn't even know about it) and is a good candidate for diagnostic readout list. The readout lists can be changed with little effort.

The ADMEM has a special readout for diagnostic purposes. If a special calibration code is sent to the ADMEM board, the following clock cycle is taken in "hiccup" mode. In this mode the data from the QIE-FADC passes first through the standard FRAM calibration table to linearize the data. Then on the following clock cycle the SAME data is passed through the FRAM in pass through mode. The result is that the L1 storage pipeline will have the normal data in the cell corresponding to the correct bunch crossing and in the following cell will contain the raw data. The TS would then issue 4 successive L1 accepts to enable readout of both the raw and calibrated data. Note that the data in the next few clock clock cycles may be seriously affected by this "hiccup" mode since the FADC has to be stopped and restarted to execute this cycle.

Table 3: Readout list for Calorimeter readout and Trigger modules. All also have a 4 byte header word.

Description	Bytes/Channel	Channels	Byte/Board	Total
ADMEM				
Tower Data	2	20	40	
"Hiccup" Raw Tower Data	2	20	40	
DIRAC				
Trigger Tower Decisions	2	6	12	1152
Total (includes headers)			16	1536
Cratesum				
DIRAC ΣE_T	1	8	8	96
DIRAC Trig summary	2	8	16	192
Wedge ΣE_T	2	2	4	48
Two Wedge Tower Decision Summary	4	1	4	36
Total (includes headers)			20	372
SUMET PreFRED				
ΣE_T, from the previous CDF_CLK cycle	2	1	2	2
$\not{E}_T$ and ΣE_T trigger decision bits	1	1	1	1
$\not{E}_T{}^2$	2	1	2	2
ΣE_T, ΣE_x, ΣE_y	4	1	4	4
Total (includes header)			13	13
Tower Trigger PreFRED				
Final trigger decision bits	3	1	3	3
Total (includes header)			7	7
DCAS				
Trigger Tower Data	2	16	32	2304
Seed and Inclusion Maps	2?	16	32	2304
Total (includes headers)			68	4896
L2 Processors				
Complete Cluster list	4	1	4	4

5 Physics Monitoring

The bit-level verification of a large number of events is a powerful method of detecting errors. However it is possible that some errors will not be simple to detect such as a logical flaw in the downloaded RAM contents. These errors will have to be detected at the physics level.

Also, conceptually the physics level is the final word since the quality of events on tape is the goal of the trigger. There are several methods of detecting errors.

Clearly there will be a monitor with the functions of LUMMON. One of the first lines of defense is the trigger rate. Since the rejection is so large, a small error can cause a large increase in the trigger rate which can easily be spotted. This is also sensitive to subtle interplay between components of the trigger such as bunch crossing labelled in sync while truly being out of sync or a misalignment of XTRP and calorimeter data due to a miscabling or a failure of some bits. Also ratios of triggers may be more sensitive to failures. For example a 20% increase in an isolated trigger might be considered insignificant considering all the run conditions but when ratioed to the unisolated trigger, it becomes obvious there is a problem.

As well as the bit-level checking, TRIGMON will be making the several basic histogram of trigger performance. The most powerful is to record the efficiency of the trigger by finding trigger objects in unbiased samples and checking that the trigger identified them. The rate at which good events are rejected can be recorded for each trigger. A carefully structured set of triggers is needed to provide the best statistics and to cover all triggers. In some cases each component of the trigger needs to be checked with a different samples. For example, the track attached to the electron might be tested by a prescaled EM cluster trigger while the EM threshold would be checked with lower-E_T trigger with a track. The process of identifying these monitoring methods is an important task since the same method can be used to measure trigger efficiency for final physics results. Efficiencies can be recorded as a function of ϕ and η to look for local failures. The ΣE_T from the previous crossing can be checked for a correlation with the triggered crossing. Of course this was worked through for Run I so the place to start would be with the old TRIGMON.

At a higher level, there is the functionality of SPY which nominally monitors L3 but also monitors physics quantities in general. This process will be able to look at a purified physics sample, events that have already satisfied the trigger at many levels. This is the point where the sample can be compared against the physics expectation with high statistics. For example, the W, Z and ψ cross sections can be measured in a crude way. The geometric acceptance and kinematic properties of these events are predictable and can be checked.

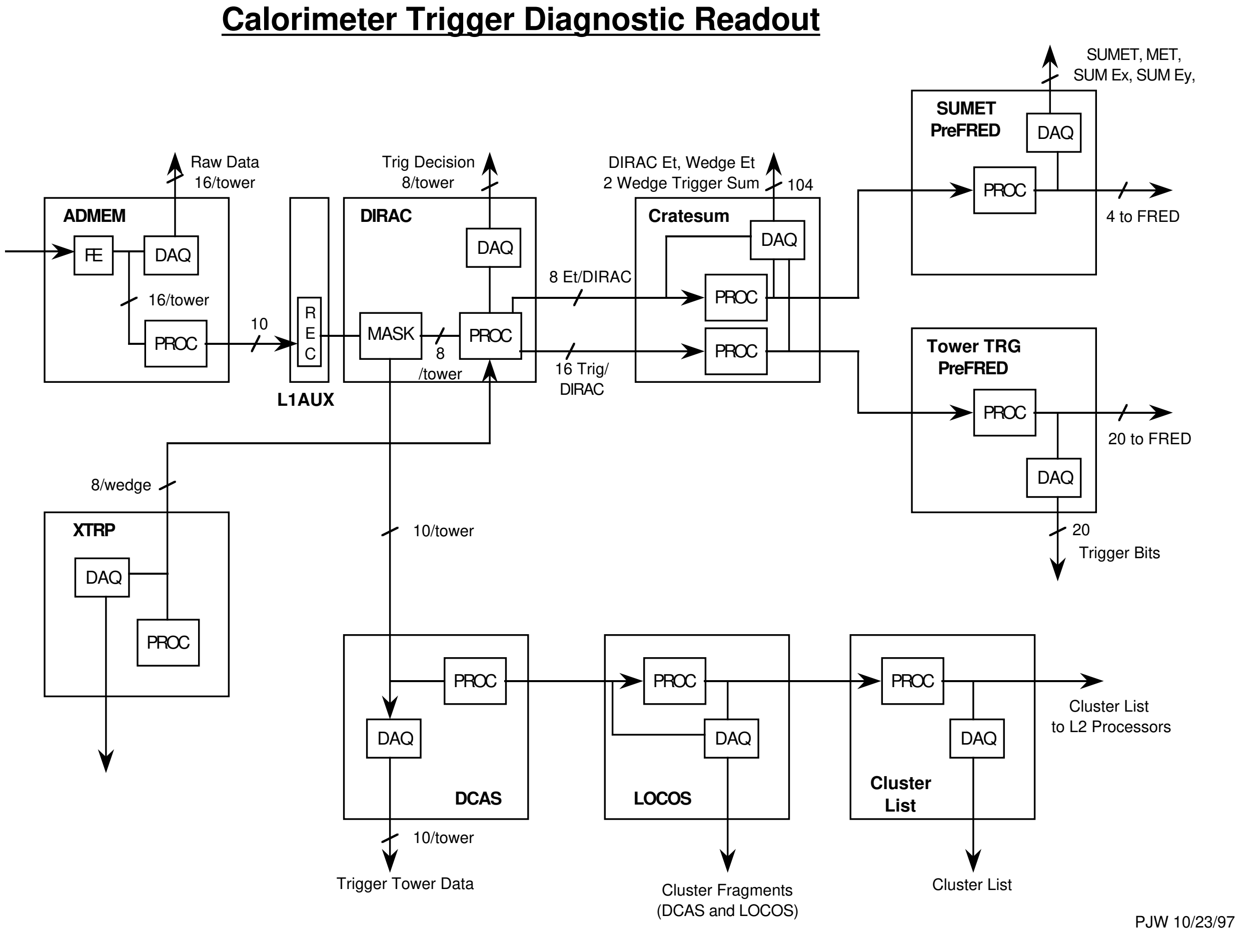

Figure 1: Block diagram of the Calorimeter trigger chain showing where processing takes place and where diagnostic data is readout into the event stream.