Project Monitor Form

Project: CMS FED **Date:** Monday 3 March-2008	**PMF number: 70** **Sheet:** 1 **of** 2

Project Implementation phase.

FED Production

Full details in Production spreadsheet.

FEDs

System at CMS is performing well in integration tests with full DAQ systems.

Tracker was successfully installed at CMS in December. Fibre cabling is in progress.

Total of 5 boards are currently at RAL for investigation and repair. Return is not urgent.

Firmware and Software

Problems with overflows in Raw Data special 904 MiniDAQ runs at high random rates. FE buffers will overflow if short bursts of back to back frames. FED freezes and stops Tracker DAQ.

Report from Geoff.. December 2007
Slawek reports some crashes recently at high occupancy and trigger rate, eg when pedestals for some reason vary. It is believe the crashes are caused by FE FPGA buffer overflows. I thought the FE buffers could not overflow and we protected the back end by the busy and other FMM inhibits, and then we would send empty frames if required.

Reply
The way the firmware has worked up to now is that if there is an overflow of any of the many buffers, either in the FE FPGA in general due to too much data in ZS mode running or in the BE due to too many L1 Accept signals (in Virgin Raw running), then the entire FED logic will immediately go into freeze which stops all further data being accepted from the FEs but does allow all the events in the BE up to that instant to be readout.

This allows all the buffer and other status registers to be read at the instant it happened to ascertain the source of the fault . This has proved the most effective way of understanding 'crashes'. Allowing the FED to continue further would likely hide the original cause of overflow.

Of course the Busy signals from the FE and BE are passed to the FMM to prevent this happening. But it may be that there are too many events in the APV pipeline already to prevent eventual overflow. One could try reducing the Busy level if we are on the edge.

It was originally the intention to implement a mode to allow the logic to degrade more gracefully eg by putting out empty events if the FE signalled it was approaching overflow. We have discussed options for this but implementing it has not been tried yet as it is quite complicated. This might also be done by the FE flagging a channel had overflowed in the data for instance. In cases where a few channels were occassionally overflowing due to real data fluctuations this would be useful. However in many abnormal conditions I suspect it may be of limited use

But in short the FE FPGA buffers will currently overflow if occupancies or rates are beyond limits.

Plan of action to remove the freezing of FEDs when the FE FPGA buffers overflow:

Phase 1:
Oz to modify the FE FPGA f/w in order for the Full Flags to be true full flags.
Saeed to modify the BE FPGA f/w in order to:
Assert BUSY when the buffers are partial full (about 75%) to signal to the trigger source to slow down or stop further triggers If FE FPGA Full FLAG is then asserted, Mark all event fragments (from this FE FPGA) in the Tracker Header as corrupt from this point onwards so that offline software can ignore the data Release BUSY when there are no more events pending and all buffers are empty . FED will then recover and remove the corrupt flags in the header and will continue reading out good events without any user intervention.
This should ensure continuous FED operation in the event of FE overflows across all fibres running in Virgin Raw modes (due to excessive trigger rates). And also in the case of extreme data fluctuations on individual fibres running in ZS mode.

Phase 2:

In addition to Phase 1, Oz to investigate mechanisms for flagging overflows at the fibre level within the FE FPGA f/w. Specifically useful for ZS mode.

New BE code

This is a new BE FPGA firmware.
IN this version, once the TTS BUSY status is asserted, if the FE FULL Flag is set, then the BUSY signal is kept asserted.
All events from this point is flagged as corrupt by the way of FE FULL Flags in the header words (latched in BE).

The system does not assert OVERFLOW unless one of the BE FPGA buffers overflows.
The system then waits for all buffers to be empty before releasing the BUSY signal.

At this point the normal operation would continue.

Tested by Jonathan Feb 2008

> Hi guys,
>
> Today I found time to test the new fed firmware. I have deliberately
> overflowed the fed back end buffers... And then readout the events and
> have found that the new firmware seems to work.

Counters Reset

Saeed added register for (yet another non standard) way of resetting the event counters for special DAQ runs.

Currently these counters are only reset via TTC Channel B commands: RESYNC+ BX rst + L1A rst (of course this would reset the system at the same time, if a register is implemented then it would have to be done on a FED by FED basis

```
which could take longer!)

If an additional reset is required, then it can be implemented, but beware of
the potential hazard. (please see below)
```

sLHC

PPRP meeting was held at Glasgow in December.
Visit by PPRP panel to Imperial in January.

Actions from the previous PMF			
Action	**Status**	**Who**	**Original Target date**
Release Draft of Fed User Document	In Progress	JC	30-11-07
Provide comments for new FE algorithm proposal	Done	JC	

Actions outstanding and new actions		
Action	**Who**	**Target Date**
Release Draft of Fed User Document	JC	01-07-08
Update BE doc with new throttling description	ST	01-06-08
Review outstanding firmware user requirements for Jon	JC,ST	01-06-08
Examine 5 boards returned from cern	IC	01-05-08
Distribute project close out documents	JC	01-06-08

Project Monitor Form- milestones

Project: CMS FED		**PMF number: 70**		
Project Manager: J. Coughlan **Date:** Monday 3 March-2008		**Sheet: 2 of** 2		
	Milestones from **P**roject **M**anagement **P**lan Version**:1.6**	**date due in PMP**	**predicted date**	**date done**
1	User Requirements Document	30.07.01		26.09.01
2	Project Spec sign off	21.12.01		05.02.02
3	Board Level Preliminary Review	14.01.02		16.01.02
4	FE Analogue Channel Feasibility Review	31.01.02		21.03.02
5	FE Module Feasibility Review	28.02.02		08.05.02
6	Board Level Feasibility Review	25.02.02		25.02.02
7	Delay FPGA Interim Review	11.03.02		27.03.02
8	Front End FPGA Interim Review	28.02.02		12.08.02
9	Back End FPGA Interim Review	04.03.02		17.12.02
10	FE Module Final Review	18.06.02		25.06.02
11	BE Module Interim Review	28.06.02		15.08.02
12	Schematics finalised	05.08.02		22.08.02
13	Layout & Routing done	16.09.02		29.10.02
14	Full Board FEDv1 Design Final Review	23.09.02		06.11.02
15	IDR Customer Production sign off &PCB Tape Out	07.10.02		06.12.02
16	Batch 1 (2 off) Non-Opto Assembled FEDv1s at RAL	11.11.02		22.01.03
17	Old version OptoRx for Batch 0 in UK	26.08.02		28.01.03
18	Batch 2 (3 off incl 1 Opto) Assembled boards at RAL	20.06.03		27.06.03
19	New version OptoRx at RAL	01.04.03		21.07.03
20	FEDv1 Interim Review	08.09.03		11.09.03
21	Batch 3 (6 off all opto) Assembled boards at RAL	30.09.03		08.10.03
22	Ship 1st FEDv1 to CERN.	30.09.03		03.11.03
23	Ship 2nd FEDv1 to CERN.	28.11.03		19.12.03
24	Batch 4 (6 off DDi) Assembled boards at RAL	01.03.04		22.03.04
25	Finalise design changes for FEDv2	01.04.04		25.03.04
26	Design Review FEDv2	18.06.04		04.06.04
27	FEDv2 tape-out	16.07.04		28.06.04
28	First FEDv2 boards at RAL	08.10.04		02.08.04
29	Dispatch calls for Tender	26.08.04		13.09.04
30	Sign Tender contract	09.02.05		24.06.05
31	FEDv3 tape-out	06.04.05		08.06.05
32	First FEDv3 boards at RAL	13.07.05		28.07.05
33	Production of 500 FEDv3 starts	08.09.05		01.08.05
34	First FEDv3 at B904 Prevessin	30.11.05		27.10.05
35	First FED installed at USC55	17.11.05		19.10.06
36	Last FED installed at USC55	26.07.06		20.07.07
37	Power on Tracker at USC55	01.08.06	01.07.07	
38	Readout test with Tracker at USC55	01.10.06	01.08.07	
39	LHC test run	02.04.07	01.11.07	