	DIF task force meeting : minutes **FW common repository** **Sharing effort on FW** **Inspection of some DIF functionalities**	Draft 0

Revision status	Comments	Date
Draft 0	First version by RC. Distributed to participants for comments	22/10/08

DIF task force meeting: minutes

FW common repository

sharing Effort on FW

Held on Wednesday October 22

The meeting started at 18:05 CEST.

List of the participants: Mathias Reineke (DESY), Bart Hommels (Cambridge), Remi Cornat (LLR)

Julie Prast is excused

Agenda on Indico : http://ilcagenda.linearcollider.org/conferenceDisplay.py?confId=3101

1) **Common repository for the DIF firmware**

The following demands were expressed :

- Use of the existing web site(s)
- Avoid "complicated procedures", simple management
- Compressed archive of each blocs/design with an "official" name and a version number could be attached to the page (up/downloaded form/to the site or links to the designer's pages)

As we are at early development stage, we do not need a more sophisticated system.

We all agreed to create a wiki page on the calice wiki (Bart or Remi could create the first page).

2) **Sharing the effort on DIF firmware development**

Mathias started to work on the firmware from LAL design (test bench of ROCs) : existing code reference for the DIF-SLAB interface.

We agreed on the need to define a few top level blocs and a structural architecture. The top level blocs are defined according to the major functions of the DIF and include all the related sub functionalities to provide a clear object that could be easily integrated by the end user.

Work to be done by one month : define top level blocs entities.

First list of top level entities :

- DIF-LDA itf manager, including 8b/10b MAC + fifo + LUT for commands +…

- USB itf, including "LDA emulator"
- DIF-SLAB itf

Other blocs : data flow MUX (selection of USB or LDA or adjacent DIF) + see Bart's bloc diagram

3) **DIF functionalities**

Commands : Mathias proposed to use a register to store the occurrence of each command then the register is reset when the command is executed (equivalent to an acknowledge). We agreed on that.

Commands are issued using K characters. About 32 K characters are available; each comes with 16 data bits. The K character can be specific to a command or to a command family. A Look up table could be used to decode the commands and to write the corresponding registers.

When the commands call for an answer (eg. update of temperature sensors register), the answer is sent to the LDA using SC block transfer mode. Spontaneously or after another command (eg. Read temp sensors register) ?

Commands are decoded using a LUT (command mode of the LDA-DIF link): specific to the detectors, useful in early development stages (debug commands can be set).

Slow control : it uses block transfer mode. Limitation due to the packet size + packet interrupt (if command need to be sent). Implies that the LDA/ODR must store the packets if the data stream is divided into more than one packet. May be a problem with CRC.

Local buffering of packets ? the SC clock on ROC side can be interrupted if needed, on the flow R/W operations seem possible.

Clock: the DIF should be designed for a minimum clock frequency. If the actual frequency is higher, the DIF should continue to work. We agreed on the following range : 40-120 MHz with a default value set at 80 MHz.

Which clock to use when the DIF is controlled through USB (and the LDA cable is not plugged)? USB module includes a 6 MHz crystal + frequency multiplier.

USB: use of a DIF-LDA "emulator" to provide the same entity ports than the DIF-LDA itf. It should allow an easy multiplexing of these two itf.

"real time" functionalities will be degraded when using USB

Power pulsing: for the moment it seen as a VETO on some specific signals. The power pulsing bloc comes in parallel of all other blocs.

Conclusion: Agreement to put FW archives on a wiki page on the calice wiki
Forthcoming work focused on the definition of top level functional blocs, structural architecture by the end of November.