AHCAL DIF.

Status and Outlook

Mathias Reinecke
for the DESY AHCAL developers
EUDET Electronics/DAQ meeting
London, June 9th, 2009

Outline

> Reference Documents for interfacing the DIFs

> USB DIF Setup for AHCAL and in general

- Purpose of USB Interface

- Protocols for Simple Commands and Large Amounts of Data (e.g. readout)

> Switch to CALICE DAQ / DOOCS (AHCAL)

Please understand these slides as invitation for discussions!

Reference Documents (the DIF developers)

'**Format of the Readout Data of the DIF**'
Version 1.1.0

DIF – Operating Manual (May 18th, 2009):
http://adweb.desy.de/~reinecke/DIF_Firmware_vers1_13.pdf
(same as version 1.12 except for some AHCAL specific commands).

'All' necessary command- and address-definitions
for DOOCS development.

But: USB protocols not included here! Still undefined!

DIF USB interface

> Purposes of the USB-to-DIF interface

- Debugging interface in 'final environment':
  error in CALICE DAQ communication

- Commissioning of the Prototype Detectors.

- New: Spy interface (in parallel to CALICE DAQ).

- Not: Synchronous operation of full detector.

> USB interface emulates CALICE DAQ interface as much as possible.

Purposes define protocol complexity – maybe.

DIF USB – Communication Protocol

Remi's proposal (1 byte per line), from PC to DIF:

USB Data	Explanation	in AHCAL up to now (prelim.)
START	0xAA	0xCC (we can easily change)
Header	2b config, 5b address, 1b r/w	type_modifier (significant byte)
Size	Size data block MSB	Size data block MSB
Size	Size data block LSB	Size data block LSB
Data	Data to DIF	Data to DIF
...	...	...
Data	Data to DIF	Data to DIF
STOP	STOP or RESTART	

> 'Data' is most often 2-bytes command data (see command manual)

> Do we need STOP byte? (could be in data / we have data-length info)

> Missing in AHCAL 'header': address for partition/slab/ROchain (3b). Could be in 'Size MSB' or 'Data MSB' ??

DIF USB – Communication Protocol II

In AHCAL: DIF answers on commands with an Acknowledge (6 bytes):

USB Data	in AHCAL: Acknowledge
START	0xCC (we can easily change)
Header	type_modifier (significant byte)
Size	Size data block MSB
Size	Size data block LSB
Data	Command Data Word MSB
Data	Command Data Word LSB

> Simple Commands: Acknowledge is echo.

> Readout of settings (temperature, registers, voltages, …):
  DIF sends data first and Acknowledge afterwards

> FAST Commands: echo.

Field	SubField	Comments
PACKETTYE (16b)		DIFBT_PKT_DATA = 0x"0001"
PACKETID (16b)		Arbitrary ID (automatic increment)
TYPEMODIFIER (16b)		0x000e
DATALENGTH (16b)		Number of 16b words in the DATA field
DATA	localDIFID (6b)+ ROpacketID (10b)	Read-out data specific header
	ROLastPacket(1b) + ROChainID (3b) + ROSequenceID (12b)	Read-out data specific header
	ROCData 1 to 505 x 16b words	Data slice from the ROCs, w or w/o empty chips (a slice of **DIF_MaxROPacket_Size** – 7 words max)
CRC	16b	Inserted by 8b/10b itf

Readout packet structure for CALICE DAQ
(see 'reference documents')

In AHCAL up to now:

> No CRC.

> Data Section 100bytes maximum.

> Some fields filled with dummy data up to now.

OK, or something more simple for USB?

DESY

DIF USB – Slow Control Data

Field	Subfield	Comments
PACKETTYPE (16b)		
PACKETID (16b)		
TYPEMODIFIER (16b)		
DATALENGTH (16b)		
DATA	localDIFID (6b) + ROpacketID (10b)	16 bit
	SCDataSet (1b) + ROChainID (3b) + ASICAddress (12b)	16 bit
	SCData	1 to 505 16bit-words
CRC (16b)		

Table 26: Block Transfer packet structure for the transfer of Slow-Control data

SC packet structure for CALICE DAQ
(cf. command manual)

In AHCAL up to now: 1byte header:
2b partition + 5bit ROCaddress + 1bit SCSet

SC-data is transferred ROC-wise.

Will be changed if we agree on CALICE-DAQ scheme for USB!

Switch to CALICE DAQ (to be discussed here)

> What is needed to prepare DOOCS, ODR/LDA/DCC firmware(s).

  - Commands can be integrated into DOOCS one after the other.

> Time-schedule for AHCAL to switch to CALICE DAQ?

Conclusions and Outlook

> First communication Labview-USB-DIF established.

> Preliminary USB protocol used, needs clarification.

> Development of DOOCs firmware needs closer cooperation (input from our side!)