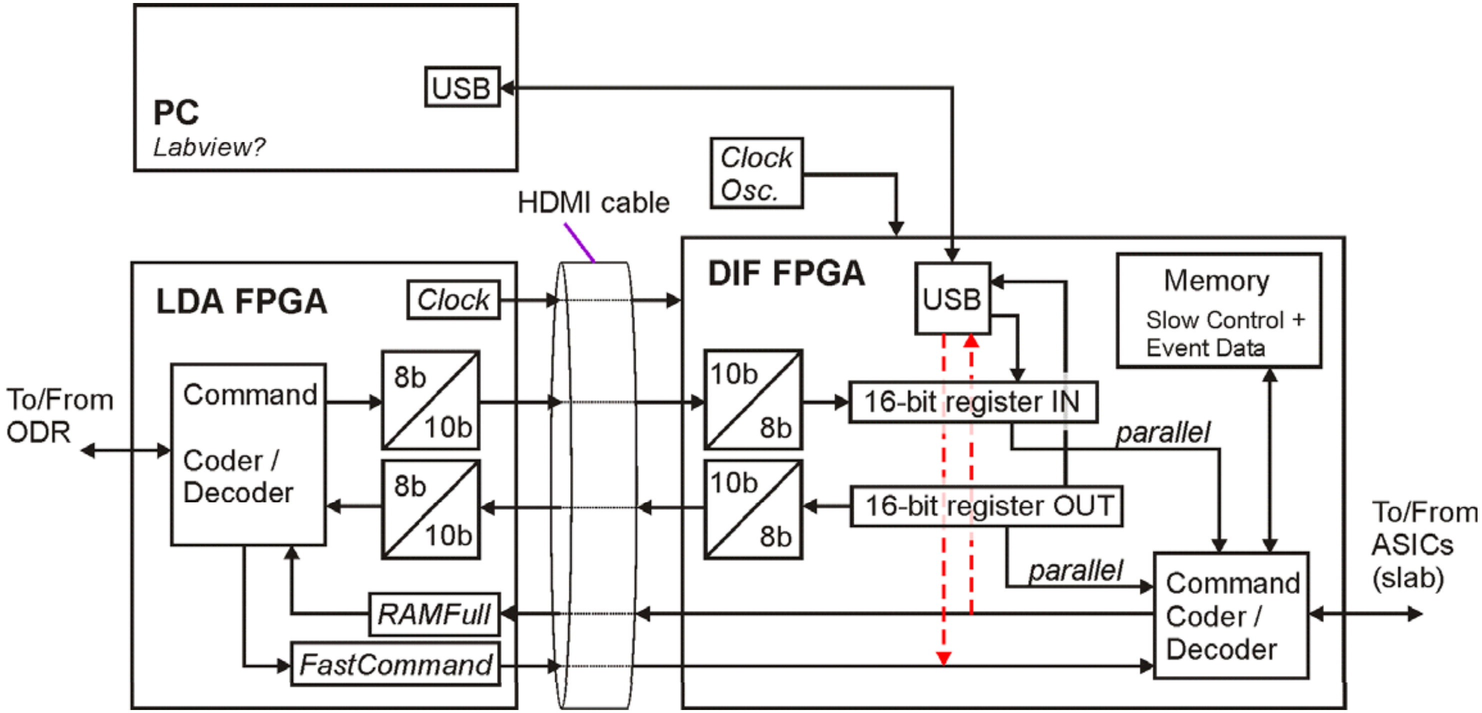
DIF-LDA command interface

The CALICE DIF developers

DIF-LDA interface



DIF clock (from LDA): def. 80MHz (40-120MHz)
Standard data transfer: 8b/10b
Fast Commands: uncoded

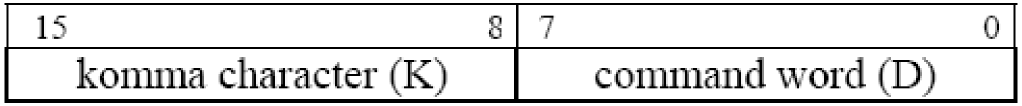
USB interface emulates LDA interface (clock-source: free of choice)

Standard data transfer LDA-DIF

FE

1. Command Frame (16-bit length):

DIF-DIF link
Bit0 ???

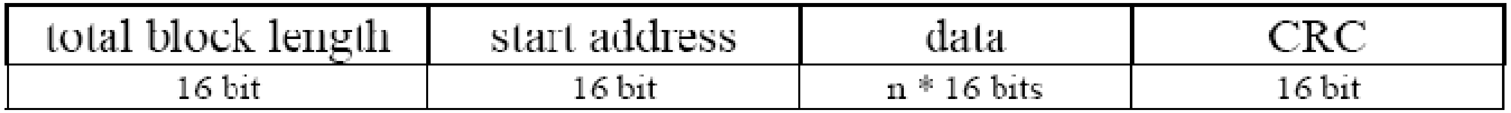


15 8	7 0
komma character (K)	command word (D)

The komma character K and the command word D are referenced to by *KX.Y* and *DX.Y*, respectively. E.g.:
K28.1 is the 8-bit sequence 11100 001.

Several Ks and „special sequences“ are predefined. See
http://www.hep.manchester.ac.uk/u/mpkelly/calice/lda/Calice_LDA_Overview.html

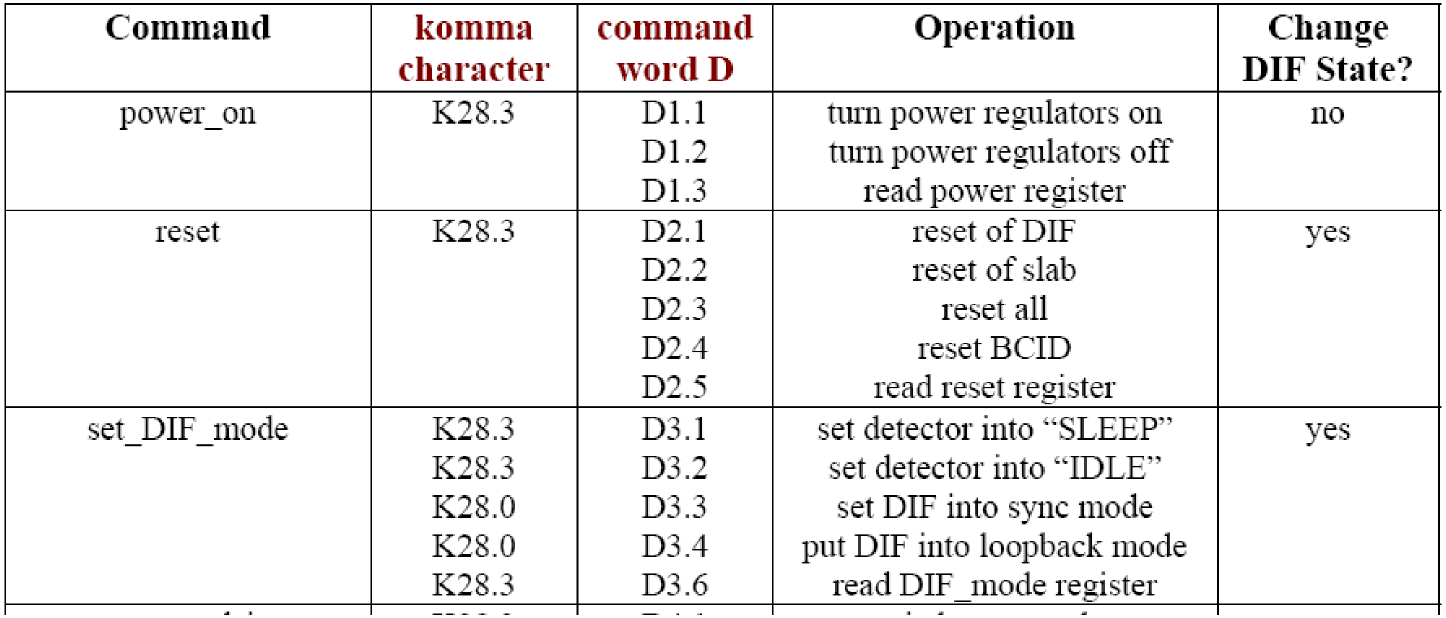
2. Block Transfer



total block length	start address	data	CRC
16 bit	16 bit	n * 16 bits	16 bit

LDA to DIF commands

Preliminary command list available:

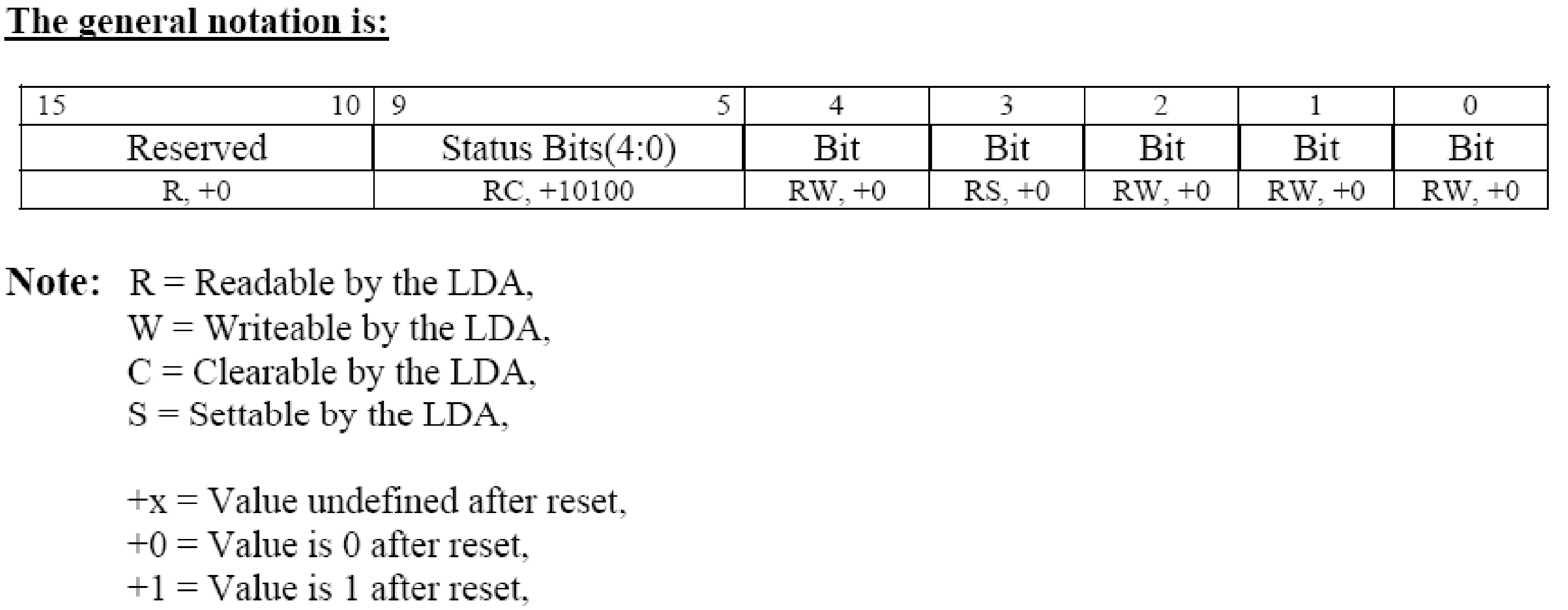


Command	komma character	command word D	Operation	Change DIF State?
power_on	K28.3	D1.1 D1.2 D1.3	turn power regulators on turn power regulators off read power register	no
reset	K28.3	D2.1 D2.2 D2.3 D2.4 D2.5	reset of DIF reset of slab reset all reset BCID read reset register	yes
set_DIF_mode	K28.3 K28.3 K28.0 K28.0 K28.3	D3.1 D3.2 D3.3 D3.4 D3.6	set detector into “SLEEP” set detector into “IDLE” set DIF into sync mode put DIF into loopback mode read DIF_mode register	yes

Command list has 4 sections: General commands,
ECAL specific commands,
DHCAL specific commands,
AHCAL specific commands.

Command Registers in DIF

For each command (LDA to DIF) the DIF has a dedicated **command register**.
The address of this **command register** is defined by the X in the incoming DX.Y command word.
Command registers are 16-bit, and can be subdivided for several functional purposes.



The general notation is:

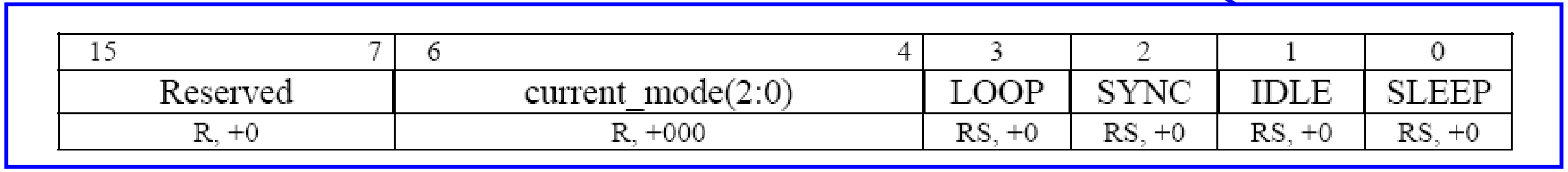
15 10	9 5	4	3	2	1	0
Reserved	Status Bits(4:0)	Bit	Bit	Bit	Bit	Bit
R, +0	RC, +10100	RW, +0	RS, +0	RW, +0	RW, +0	RW, +0

Note: R = Readable by the LDA,
W = Writeable by the LDA,
C = Clearable by the LDA,
S = Settable by the LDA,

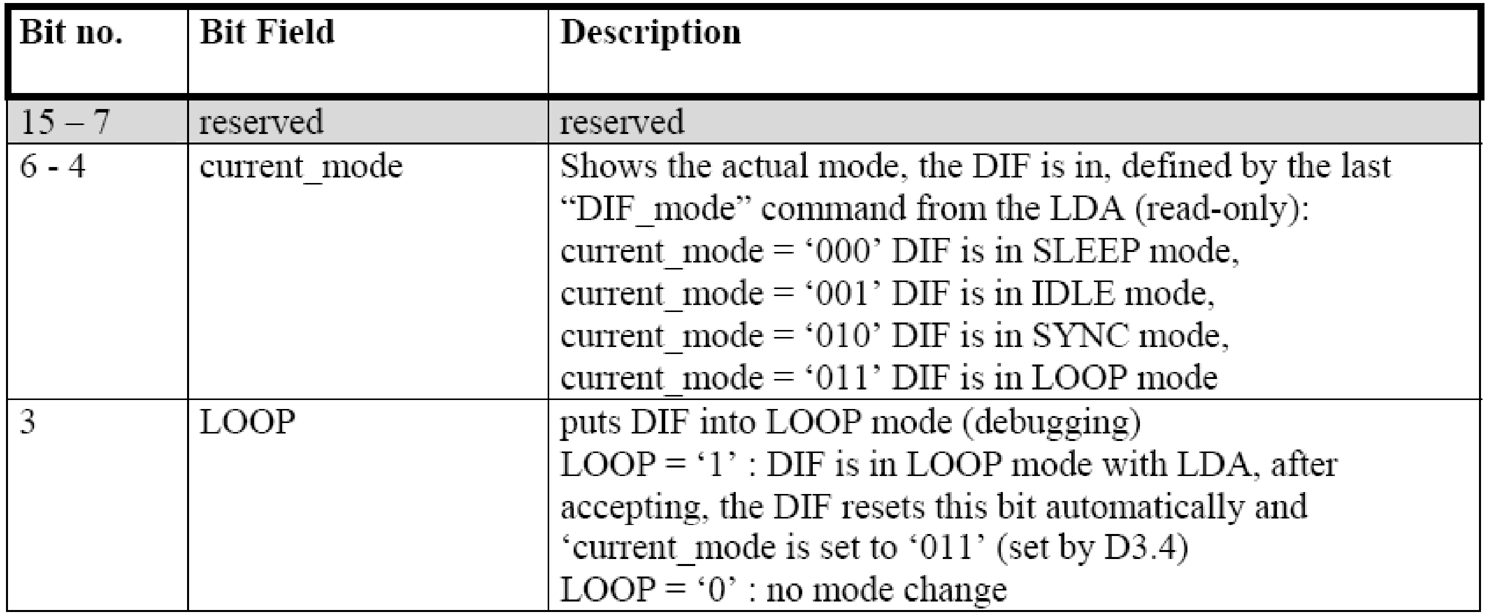
+x = Value undefined after reset,
+0 = Value is 0 after reset,
+1 = Value is 1 after reset,

DIF command Register: Example

2.1.3 DIF_mode register (command and address D3.Y)



15 7	6 4	3	2	1	0
Reserved	current_mode(2:0)	LOOP	SYNC	IDLE	SLEEP
R, +0	R, +000	RS, +0	RS, +0	RS, +0	RS, +0



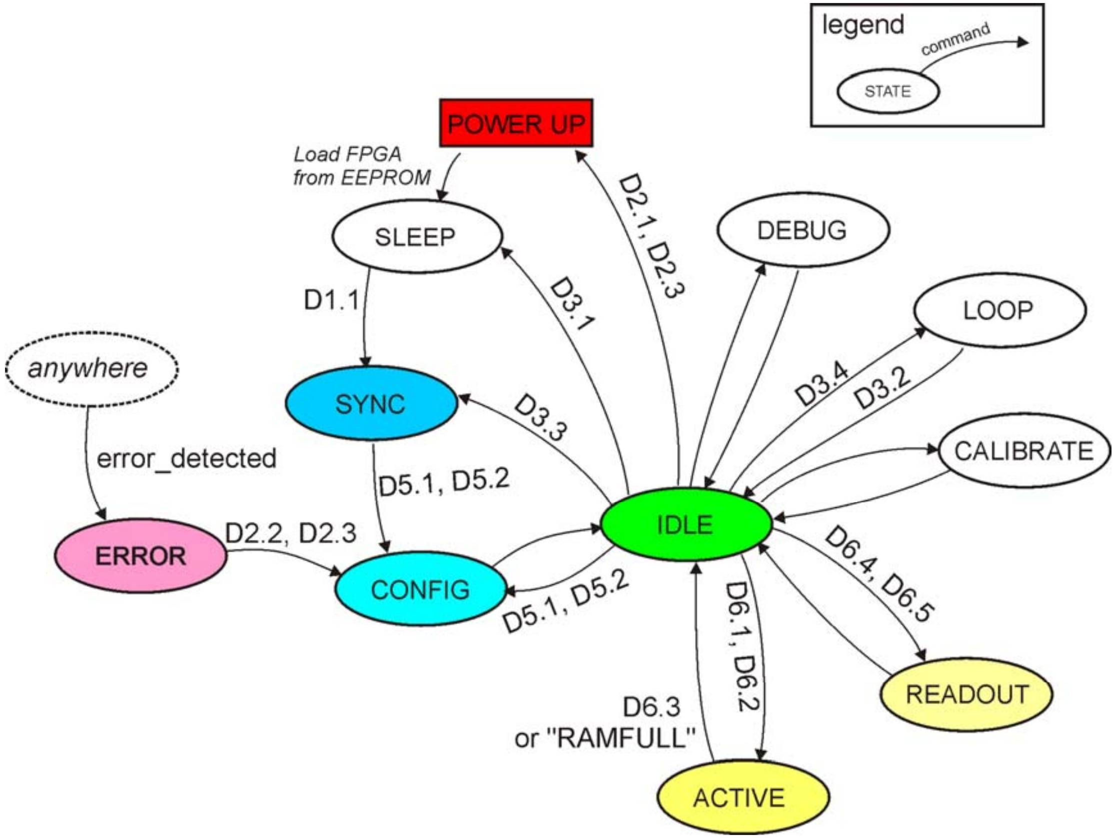
Bit no.	**Bit Field**	**Description**
15 – 7	reserved	reserved

6 - 4	current_mode	Shows the actual mode, the DIF is in, defined by the last “DIF_mode” command from the LDA (read-only):
		current_mode = ‘000’ DIF is in SLEEP mode,
		current_mode = ‘001’ DIF is in IDLE mode,
		current_mode = ‘010’ DIF is in SYNC mode,
		current_mode = ‘011’ DIF is in LOOP mode

3	LOOP	puts DIF into LOOP mode (debugging) LOOP = ‘1’ : DIF is in LOOP mode with LDA, after accepting, the DIF resets this bit automatically and ‘current_mode is set to ‘011’ (set by D3.4) LOOP = ‘0’ : no mode change

Bit 2, 1 and 0 description not shown

DIF states



Change DIF State (last column) means: If the DIF changes its state from „IDLE“ to any other state by a received command, the respective operation should not be interrupted by following commands except for emergencies or resets. After completion of the tasks, the DIF changes back to “IDLE” automatically and is ready for new commands.

Conclusions

Ideas about the DIF command interface are preliminary and by no means fixed.

We should discuss the architecture (DIF-LDA interface) and agree on a concept soon.

Reference Document: „DIF Firmware“ version 1.3, 5th Nov. 2008
Download via http://ilcagenda.linearcollider.org/conferenceDisplay.py?confId=3132