

Trigger Fast Control Commands and Data Formats

This document is recreated based on the original Fast Control (FC) command definition document by Krista Marks, and updated for the 2002 DCT upgrade modifications. This command protocol is used for all DCT and GLT modules. The definitions for the Control and Status Register (CSR) formats are also included.

1 Global (Run Time) Commands

1.1 CLINK Format

All run time CLINK commands are 12 bit instructions with no additional data to follow. The format for CLINK command is (LSB first):

zsccccdddd
z = one leading bit = 0
s = one start bit = 1
c = five command bits, LSB first
d = five data bits, LSB first

1.2 DLINK Format

When a “Read Event” command is received, all “oldest” data stored in the DAQ buffers are sent on the DLINK. A header 32 bit DAQ header is sent first followed by the actual DAQ data. Each trigger module sends a fixed number of bytes for any given DAQ data format.

Header (4 bytes)	
bit	description
0:0	Start bit = 1
1:1	Event data = 1
2:6	Trigger tag (0:4) (L1A command data)
7:7	Reserved - byte alignment
8:12	Trigger counter (0:4) (Local clock counter)
13:14	DAQ buffer number (1:0)
15:15	Reserved - byte alignment
16:31	CSR1 status value (15:0)
Data: N 16-bit words	

Note that the header and data are sent with LSB first, but header DAQ buffer number and the CSR1 values have the reverse bit order with MSB going out first - a historic feature of the FC firmware code on DCT/GLT boards.

1.3 Command Definitions

Op-code	Command	Data	Description
0 ₁₆	No operation	00000	No operation.
1 ₁₆	Clear Readout	Reserved	Reset address pointer to DAQ buffers.
2 ₁₆	Sync	Reserved	Align counters for generating lower frequency clocks.
3 ₁₆	L1 Trigger Accept	trigger tag	Transfer event data to next available DAQ buffer.
4 ₁₆	Read Event	Reserved	Start transfer data in “oldest” DAQ buffer to ROM.
5 ₁₆	Calibration Strobe	Reserved	Undefined for trigger.
6 ₁₆	Start Playback	Reserved	Start playback/record for diagnostic memories.
7 ₁₆ -B ₁₆	Reserved	Reserved	

2 Subsystem Specific (Non Run Time) Commands

2.1 CLINK Format

Non run time CLINK commands also start with a 12 bit command header, but can have various different types of additional command data to follow. The general format for CLINK command is (LSB first):

zsccccacaaaadd...

z = one leading bit = 0

s = one start bit = 1

c = five command bits, LSB first

a = five sub-command bits, LSB first

d = data bits (lengths dependent on command), LSB first

2.2 DLINK Format

The non-event DLINK data from frontend have the following general format (LSB first):

Header (2 bytes)	
bit	description
0:0	Start bit = 1
1:1	Register data = 0
2:6	Command Op-code (0:4)
7:7	Reserved - byte alignment
8:12	Sub-command (0:4)
13:15	Reserved - byte alignment
Data: N 16 or 32 bit words	

2.3 Command Definitions

Op-code	Command	Sub-command	Data
1E ₁₆	User Reset	Reserved	None
1D ₁₆	Read CSR	CSR address	None
1C ₁₆	Write CSR	CSR address	16 bit CSR value
1B ₁₆	Write Block Address	Reserved	16 bit block address value
1A ₁₆	Write Address	Reserved	16 bit address value
19 ₁₆	Read Memory	0/1=16/32 bits	None
18 ₁₆	Read Memory & increment address	0/1=16/32 bits	None
17 ₁₆	Write Memory	0/1=16/32 bits	16 bit address + 2/4 bytes data
16* ₁₆	Block Write Memory (fixed 2Kb)	0/1=16/32 bits	[16/32-bit start address] [512 32-bit data]
15* ₁₆	Block Read Memory (fixed 2Kb)	0/1=16/32 bits	None
14 ₁₆	TSF Reframing	Reserved	None
13† ₁₆	Block Write Memory (variable size)	0/1=16/32 bits	[16-bit start address] [16-bit word count=N] [N data words (each 16/32 bits long)]
12† ₁₆	Block Read Memory (variable size)	0/1=16/32 bits	None
11† ₁₆	Block Read Setup (variable size)	0/1=16/32 bits	[16-bit start address] [16-bit count=N 16/32 words to read]

* **Commands 16₁₆, 15₁₆ only applicable to old TSF,PTD LUT and GLT.**

† **Commands 13₁₆, 12₁₆, 11₁₆ only applicable to new upgrade TSF, ZPD.**

Op-code	Description
1E ₁₆	Reset any user specified control registers and local pointers.
1D ₁₆	Readout Control Status Register (CSR) specified by subcommand.
1C ₁₆	Set Control Status Register (CSR) to attached data value.
1B ₁₆	Set block address (upper 16 bit of 32 bit address).
1A ₁₆	Set current register/memory address (upper 16 bit of 32 bit address).
19 ₁₆	Read memory data at current address. Data length defined by subcommand.
18 ₁₆	Read memory data at current address followed by increment of current address.
17 ₁₆	Write memory at specified address with 16 or 32 bit values.
16 ₁₆	Memory block-write for 512 addresses.
15 ₁₆	Memory block-read for 512 addresses.
14 ₁₆	Resync slower clocks and clear TSF input FIFO.
13 ₁₆	Write a block of memory of specified address range.
12 ₁₆	Read a block of memory as specified by command 11.
11 ₁₆	Setup block-read start address and word count.

The change of the block-transfer protocol for the new DCT upgrade is very significant. The original fast control protocol did not have the block-transfer commands and the commands 16,15 were only added later in 98. Because the ORCA 2C10 FC FPGA was very close to be full already the block-transfer implementation had some serious limitations. These commands were implemented as fixed block-transfers for 512 addresses. The maximum single block transfer size is 4Kbytes, which is the size of the frontend cache for the ROM. However, because that includes the starting address, the actual maximum number data long words is only 1023 which is awkward to divide memories with so that the actual transfers are limited to a maximum of 512 long words and using a 513 word buffer in the ROM to hold the starting address in addition. This limitation still holds for the new DCT system. The shortcomings of this old implementation were:

- Cannot use this for memories smaller than 512 addresses. Any accidental use for small ranges will cause memory overwrites beyond the range.
- The 32 bit block-write was naturally implemented with start address occupying 32 bits (only lower 16 bit used, upper 16 always zero) to ensure full 32 bit data alignment. The 16 bit block-write was on the other hand not a real 16 bit transfer, but 16 bit data followed a padding of 16 bits of zeros. This was probably due to the difficulty of increment address and write data within the same clock-4 so that the transfer had to be slowed down by a factor of two. Because the ROM code still have to pack everything into 32 bit long words, the actual packing was such that the first long word has first data word on upper 16 and starting address on lower 16 bits, while the remaining data words have real data on upper 16 and padded zeros on lower 16.
- The block-transfer OP code was only implemented for TSF,PTD LUT and all GLT memories, but not for BLT or TSF,PTD diagnostic memories.

The new DCT upgrade implementation for block transfers have added the word count control for variable size control. This also naturally symmetrized 32 and 16 bit transfers so that both types of transfers have first long word being start-address and word count, while the rest of data are naturally 32 bit aligned. The 16 bit transfers are now also running at full speed and no need for padded zeros. It should be noted that the word count mean number of 16 (32) bit words to follow in case of 16 (32) bit transfers.

Because of the incompatibility of the block-transfer protocols between new and old systems, care must be taken in the online code not apply the new transfer code to old boards and vice versa.

Here are a few specific issues worth some attention:

- The new block-read operation now takes two command to complete. Firstly, command 11 needs to be executed to setup the starting address and transfer length. This is then followed by command 12 to carry out the actual read transfer. This two step procedure is necessary similar to the single word memory read commands, as the read commands

cannot carry any command data due to a hardware limitation of the ROM. It is strongly recommended that commands 11,12 are always executed in pairs with no other operations in between and the subcommand (tag) for commands 11 and 12 must be the same to consistently indicate 16bit or 32bit transfers.

- For the upgrade FC, the CSR read command (18) for CSR3,CSR4 (current address and block address) only operates correctly for normal memory R/W (including block R/W), but these readback can be wrong after an DAQ read-event command. Write block address and write address command need to be executed after read-event to restore the validity of the CSR3,CSR4 read. This is because the DAQ readout is controlled by the OP section (board dependent) through same memory R/W mechanism by redirecting the block address and address internally without letting the board independent FC section knowing it. Given that the use of these commands are limited, it was deemed that the significant complication to completely implement CSR3,4 read is not justified. Users should therefore use these commands with caution and care.
- The upgrade version of the new FC code did not even implement the old block transfer commands 16,15. However, to protect accidental use of block-write command for which the data can be misinterpreted commands to completely mess up the board, any command 16 accidentally applied to new DCT boards will cause the FC to blank out any command interpretation for 512 words .

3 CSR Definitions

3.1 Write CSR (Op-code 1C₁₆)

The Control Status Registers are selected by choosing the appropriate sub-command, which is effectively its address. The following is a table of bit field definitions for the old and new system:

CSR1 (sub-command=1): Operation Mode		
Bits	Old system	New system
0	0/1=Non-run/Run time commands	0/1=Non-run/Run time commands
1	writable control spare	writable control spare
2	writable control spare	Non-writable
CSR2 (sub-command=2): DAQ Data Format		
Bits	Old and new systems	
0:1	User defined DAQ format number (differ between boards)	
CSR3 (sub-command=3): Playback/Record Memory Enables		
Bits	Old system	New system
0	0/1=disable/enable input memory	0/1=disable/enable mem(0)
1	0/1=input memory record/play	0/1=record/play mem(0)
2	0/1=disable/enable output memory	0/1=disable/enable mem(1)
3	0/1=output memory record/play	0/1=record/play mem(1)
4:5		0/1=disable/enable mem(2:3)
6:9		0/1=record/play mem(2:5)
CSR4 (sub-command=4): Playback/Record Mode		
Bits	Old and new systems	
0	0=cycle continuously; 1=single shot	
CSR5 (sub-command=5): Software LEDs		
Bits	Old and new systems	
0:3	4 bit LED pattern	

3.2 Read CSR (Op-code 1D₁₆)

The readback CSR address/sub-command have very different meaning from those for CSR write. A new CSR5 readback is added for new DCT system with more algorithm diagnostic memories. The CSR1 format was kept as close to the old format as possible, which causes a rather ugly split of e.g. the new Board ID which needs 3 bits. One also has to hope that we haven't and will never set the 2nd control spare bit in the old boards to cause any board ID confusion. The diagnostic memories are not explicitly names here for the new system, but index 0,1 should still correspond to the traditional input/output memories, while the additional memories can have different meanings within ZPD and TSF. Each new algorithm

memory also has its own play/record mode control, while a group of them may share the same enable line so that indices ≥ 2 have different meaning between enable and play/record.

CSR1 (sub-command=1): Register Summary		
Bits	Old system	New system
0	Run mode	Run mode
1	control spare	Control spare
2	control spare	Board ID(2)
3:4	DAQ format	DAQ format
5	Enable input memory	Enable mem(0)
6	Play/record input memory	Play/record mem(0)
7	Enable output memory	Enable mem(1)
8	Play/record output memory	Play/record mem(1)
9	Play/record mode	Play/record mode
10	GLINK ready	GLINK ready
11	GLINK synchronized	GLINK synchronized
12	Input memory active	Mem(0) active
13	Output memory active	Mem(1) active
14:15	Board ID(0:1)	Board ID(0:1)
CSR2 (sub-command=2): Software LEDs		
Bits	Old and new systems	
0:3	4 bit LED pattern	
CSR3* (sub-command=3): Read Address		
Bits	Old and new systems	
0:15	Read the lower 16 bits of the current 32 bit FC address	
CSR4* (sub-command=4): Read Block Address		
Bits	Old and new systems	
0:15	Read the upper 16 bits of the current 32 bit FC address	
CSR5 (sub-command=5): Algorithm Memory Status		
Bits	Non-existent for old system	New system
0:1		Enable mem(2:3)
2:5		Play/record mem(2:5)
6		mem(2) active
7		=1 to flag new DCT system

* Note: for new DCT FC, the CSR3,CSR4 read only operates correctly for normal memory R/W command. DAQ read-event will disrupt the validity and write block-address and write address commands are needed to restore.

4 Board ID Convention

The original board ID system had only 2 bits. To distinguish the new DCT boards, we expanded the board ID system to 3 bits (have to hope we never set the 2nd control spare bits on the old boards).

0	000	Old TSF
1	001	BLT
2	010	PTD
3	011	GLT
4	100	New TSF-X
5	101	New TSF-Y
6	110	ZPD