EDOC211

EUROGAM DATA ACQUISITION SYSTEM

Register Definitions for the FDT32 module

Edition 1.0

April 1995

Nuclear Physics Support Group

Central Laboratory of the Research Councils
Daresbury Laboratory

Registers for the fdt32 module

This document should be read in conjuction with the document **EDOC210** which describes the **Register Server** implementation for **CAMAC**.

all register attributes include -c (crate address) and -n (station address)

Registers using the **xCAMAC** class of the **CAMAC** Register Server

control register write by f(16) a(0)

The control register is 5 bits long and contains

bit 4	qerbit	value of data bit d30 on the DT32 bus - should normally be 0
bit 3	go_adc	enables data to be written by the ADCs to be FERA bus
bit 2	unused	
bit 1	test	enables TEST mode - should normally be 0
bit 0	en_roco	enables the FERA bus to write to the fdt32 data FIFO and hence to the DT32 bus

fdt32#*.control	attributes -a 0 -f 16 -w 16 -p wo -l 0 -b 0 -i 0
fdt32#*.control.go_adc	attributes -a 0 -f 16 -w 16 -p wo -l 1 -b 3 -i 0
fdt32#*.control.test	attributes -a 0 -f 16 -w 16 -p wo -l 1 -b 1 -i 0
fdt32#*.control.en_roco	attributes -a 0 -f 16 -w 16 -p wo -l 1 -b 0 -i 0

status register read by f(0) a(0)

The status register is 9 bits long and contains

bit 8	en_roco	copy of control register bit 0
bit 7	dtpause	status of DTPAUSE signal on the DT32 bus
bit 6	dtstop	status of DTSTOP signal on the DT32 bus
bit 5	pass_fera	status of PASS signal from FERA bus
bit 4	dtren	status of DTREN signal from the DT32 bus
bit 3	notfull	status of FULL flag from FIFO (low active)
bit 2	notempty	status of EMPTY flag from FIFO (low active)
bit 1	nothfull	status of HALF FULL flag from FIFO (low active)
bit 0	dtpass	status of DTPASS signal on the DT32 bus

fdt32#*.status	attributes -a 0 -f 0 -w 16 -p ro -l 0 -b 0 -z x
fdt32#*.status.en_roco	attributes -a 0 -f 0 -w 16 -p ro -l 1 -b 8 -z d
fdt32#*.status.dtpause	attributes -a 0 -f 0 -w 16 -p ro -l 1 -b 7 -z d
fdt32#*.status.dtstop	attributes -a 0 -f 0 -w 16 -p ro -l 1 -b 6 -z d
fdt32#*.status.pass_fera	attributes -a 0 -f 0 -w 16 -p ro -l 1 -b 5 -z d
fdt32#*.status.dtren	attributes -a 0 -f 0 -w 16 -p ro -l 1 -b 4 -z d

fdt32#*.status.notfull attributes -a 0 -f 0 -w 16 -p ro -l 1 -b 3 -z d
fdt32#*.status.notempty attributes -a 0 -f 0 -w 16 -p ro -l 1 -b 2 -z d
fdt32#*.status.nothfull attributes -a 0 -f 0 -w 16 -p ro -l 1 -b 1 -z d
fdt32#*.status.dtpass attributes -a 0 -f 0 -w 16 -p ro -l 1 -b 0 -z d

write gate delay DAC write by f(16) a(6)

time between VALIDATION and start of GATE signal to the ADCs
adjustable between 120ns and 23us

fdt32#*.gate.delay attributes -a 6 -f 16 -w 16 -p wo -l 0 -b 0
a sensible initial value is to be determined

write gate length DAC write by f(16) a(7)

length of GATE signal to the ADCs
adjustable between 50ns and 7us

fdt32#*.gate.length attributes -a 7 -f 16 -w 16 -p wo -l 0 -b 0
a sensible initial value is to be determined

write FT to validation delay DAC write by f(16) a(8)

time between Fast Trigger and Validation signal
adjustable between 130ns and 20us

fdt32#*.val.delay attributes -a 8 -f 16 -w 16 -p wo -l 0 -b 0
a sensible initial value is to be determined

write request to REN delay DAC write by f(16) a(9)

time between REQUEST and start of ADC readout
adjustable between 75ns and 11us

fdt32#*.ren.delay attributes -a 9 -f 16 -w 16 -p wo -l 0 -b 0
a sensible initial value is to be determined

select test output # 1 write by f(17) a(0)

various internal signals can be routed for inspection to the TEST OUT 1 lemo socket

data	0	gate
	1	fast trigger (FT)
	2	FT to validation delay
	3	REQ
	4	validation
	5	CLI
	6	trigger OK
	7	WAK
	15	WST from ADC
	23	PASS from ADC
	31	inhibit request

fdt32#*.test.#1 attributes -a 0 -f 17 -w 16 -p wo -l 0 -b 0

select test output # 2 write by f(17) a(1)

various internal signals can be routed for inspection to the TEST OUT 2 lemo socket

data	1	validok
	2	endsubevent
	3	/OPBM
	4	/OPMBR
	5	dtpause
	6	dtren
	7	dtpass

fdt32#*.test.#2 attributes -a 1 -f 17 -w 16 -p wo -l 3 -b 0

Note - this is a 3 bit register occupying the least significant bits of the data word

select test output # 3 write by f(17) a(1)

various internal signals can be routed for inspection to the TEST OUT 3 lemo socket

data	1	/OE
	2	/WE
	3	/RD
	4	/WR
	5	/HF
	6	/FULL
	7	/EMPTY

fdt32#*.test.#3 attributes -a 1 -f 17 -w 16 -p wo -l 3 -b 3

Note this is a 3 bit register which shares **CAMAC** access with the select test output # 2 function

Write LUT item write by f(16) a(1)

Write a single data word to the fdt32 Look Up Table.

fdt32#*.writelutitem attributes -a 1 -f 16 -w 16 -p wo -l 0 -b 0

Read LUT item read by f(0) a(1)

Read a single data word from the fdt32 Look Up Table.

fdt32#*.readlutitem attributes -a 1 -f 0 -w 16 -p ro -l 0 -b 0 -z x

Read FIFO item read by f(0) a(2)

Read a single data word from the fdt32 data FIFO.

fdt32#*.readfifoitem attributes -a 2 -f 0 -w 24 -p ro -l 0 -b 0 -z x

Registers using the **cCAMAC** class of the **CAMAC** Register Server

These registers do not require data and have no initial value. However read-register operations may return Q and X as data.

reset write or read by f(9) a(0)

module reset

fdt32#*.reset attributes -a 0 -f 9

clear address counter write or read by f(10) a(0)

set address used for access to the LUT to zero

fdt32#*.clearaddr attributes -a 0 -f 10

clear FIFO write or read by f(9) a(2)

clear data FIFO

fdt32#*.clearfifo attributes -a 2 -f 9

enable LAM write or read by f(26) a(0)

enable module LAM

fdt32#*.enablelam attributes -a 0 -f 26

disable LAM write or read by f(24) a(0)

disable module LAM

fdt32#*.disablelam attributes -a 0 -f 24

test LAM read by f(8) a(0)

returns Q=1 if the module LAM is set

fdt32#*.testlam attributes -a 0 -f 8

Registers using the **qCAMAC** class of the **CAMAC** Register Server

These registers transfer data using Q repeat mode between a file supplied by the write-register command and the fdt32 module. The write-register (or initialise-register) command is used to trigger execution of the **CAMAC** data transfer. The read-register command does not cause any **CAMAC** data transfer but just returns the file name supplied by the last write-register (or initialise-register) command.

Write LUT write by f(16) a(1)

The fdt32 Look Up Table is 8K words of 16 bits.

fdt32#*.writelut attributes -a 1 -f 16 -w 16 -p wo -l 8192
-i /eg/experiments/fdt32/LUT.default

Read LUT read by f(0) a(1)

The fdt32 Look Up Table is 8K words of 16 bits.

fdt32#*.readlut attributes -a 1 -f 0 -w 16 -p ro -l 8192
-i /eg/experiments/fdt32/LUT.current

Read FIFO read by f(0) a(2)

The fdt32 data FIFO is 4K words of 24 bits.

fdt32#*.readfifo attributes -a 2 -f 0 -w 24 -p ro -l 4096
-i /eg/experiments/fdt32/FIFO.current