MAJORANA MAIN FPGA REGISTERS

When reading registers that were not defined they will return the value xFFFFFFFF.

Description	Address	Access	Width
Board ID	**0x00**	**R**	**32**
Value is the board serial number that is given by the switch configuration.		**Bit Value**	
		1	**0**
Bits[11: 0]: Board Id (Pins 8 down to 4 show Geographical Addressing, all other pins are set to zero)		Value	
Bits[19:12]: FIRMWARE SUBVERSION			
Bits[31:20]: FIRMWARE VERSION			

Description	Address	Access	Width
Programming Done	**0x04**	**WR**	**32**
This gives the programming status of the VHDL module as follows.		**Bit Value**	
		1	**0**
Bits[9: 0]: Programming status of the channels			
Bit 10: Programming status of the DAC module			
Bit 11: Programming status of the front bus (slave) module			
Bit 12: Programming status of the master logic module			
Bit 13: Programming status of the Debug module			
Bit 14: Programming status of the FIFO module			
Bit 15: Programming status of the Self Trigger module			
Bits[19:16]: TBD			
Bit 20: FIFO 0 EF (empty flag) flag			
Bit 21: FIFO 1 EF (empty flag) flag			
Bit 22: FIFO 0 PAE (partially empty flag)flag (there is less then 2k words in the fifo)			
Bit 23: FIFO 0 HF (half full flag)flag (there is more then 16k words in the fifo)			
Bit 24: FIFO 0 PAF (partially full flag)flag (there is more then 30k words in the FIFO, the algorithm will stop putting data in the FIFO if this flag is asserted to avoid having incomplete data packages in the FIFO)			
Bit 25: FIFO 0 FF (full flag)flag			
Bit 26: FIFO 1 FF (full flag)flag			
Bit 27: FIFO reset.		reset	Nop
Bits[31:28]: Sets the down samples module (0 no down sampling, 1 for 2 points, 2 for 4 points 3 for 8 points and 4 for 16 points). Warning changing this value alters the behavior of all the DSP modules since the clock frequency gests divided by 2, 4, 8 or 16.			

- Note1: Bits 15 down to 0 are self clearing bits. After a VME command is sent to program on of these modules those bits are going to be set to one. Upon completion of the command those bits are going to be set back to zero. If an error occur this error will be reported by one of these bits staying at 1.
- Note 2: In the digitizer board there are two 18 bit FIFO working in parallel to generate a 36 bit wide FIFO (see schematics) Bits 26 downto 20 are the status bits that come from those two components and are reported here.

Description	Address	Access	Width
External Window	**0x08**	**RW**	**32**
External validation window length in clock cycles. Value at reset is 0x0190 or 4us.		**Bit Value**	
		1	**0**
Bits[10: 0]: External window length		Value	
Bits[31:11]: TBD			

Description	Address	Access	Width
Pileup Window	**0x0C**	**RW**	**32**
Pileup window length. Value at reset 0x0400 (10us).		**Bit Value**	
		1	**0**
Bits[15: 0]: Pileup window length		Value	
Bits[31:16]: Base line restored delay time		Value	

Description	Address	Access	Width
Noise Window	**0x10**	**RW**	**32**
Noise window length. Value at reset 0x0040 (640ns).		**Bit Value**	
		1	**0**
Bits[6: 0]: Noise window length		Value	
Bits[31:11]: TBD			

Description	Address	Access	Width
External trigger sliding length	**0x14**	**RW**	**32**
Length before we read the energy when we operate in external trigger. Value at reset 0x0190 (4.0us). In TTCS mode this register is also used but the external signal is defined to be the master board CC_LED.		**Bit Value**	
		1	**0**
Bits[10: 0]: External trigger sliding length		Value	
Bits[31:11]: TBD			

Description	Address	Access	Width
Collection time	**0x18**	**RW**	**32**
Collection time maximum length (length of the flat top in the trapezoid, typically set to 0x0020 or 320ns). Value at reset 0x01C2 (4.5us)		**Bit Value**	
		1	**0**
Bits[9: 0]: Collection time		Value	
Bits[31:9]: TBD			

Description	Address	Access	Width
Integration time	**0x1C**	**RW**	**32**
Integration time length (length of one side of the trapezoid). Value at reset 0x01C2 (4.5us)		**Bit Value**	
		1	**0**
Bits[9: 0]: Integration time		Value	
Bits[31:9]: TBD			

Description	Address	Access	Width
Hardaware status	**0x20**	**RW**	**32**
		Bit Value	
		1	**0**
Bits[0]: DCM Front Bus Lock signal (50MHz)		**Locked**	
Bits[1]: DCM Front Bus Lock signal (100MHz)			
Bits[1]: DCM VME CLK Lock signal		**Locked**	
Bits[2]: SD Lock signal		**Locked**	
Bits[15: 3]: TBD		Value	
Bits[31:16]: TBD			

Description	Address	Access	Width
Data package user defined data	**0x24**	**RW**	**32**
The data written here will show up at the data package header		**Bit Value**	
		1	**0**
Bits[11: 0]: User defined data		Value	
Bits[31:12]: TBD			

Description	Address	Access	Width
Collection time low resolution channel	**0x28**	**RW**	**32**

	Bit Value	
Collection time maximum length (length of the flat top in the trapezoid, typically set to 0x0020 or 320ns). Value at reset 0x020 (0.32us)	1	0
Bits[9: 0]: Collection time	Value	
Bits[31:9]: TBD		

Description	Address	Access	Width
Integration time low resolution channel	**0x2C**	**RW**	**32**

	Bit Value	
Integration time length (length of one side of the trapezoid). Value at reset 0x0040 (.64us)	1	0
Bits[9: 0]: Integration time	Value	
Bits[31:9]: TBD		

Description	Address	Access	Width
External FIFO Monitor	**0x30**	**R**	**32**

	Bit Value	
Monitors the occupancy of the external FIFO using internal logic.	1	0
Bits[17: 0]: FIFO occupancy	Value	
Bits[19:18]: TBD		
Bits[20]: FIFO full flag		
Bits[21]: FIFO empty flag		
Bits[22]: FIFO programmable almost full flag (Hard coded to 1023)		
Bits[23]: FIFO programmable almost empty flag (Hard coded to 261120)		
Bits[31:24]: TBD		

Description	Address	Access	Width
Control/Status channel #Number	*	**RW**	**32**

	Bit Value	
This register address is the only one that is read/write.	1	0
Bit 0: START/STOP. This bit must be set to one for the channel to operate. Value at reset is 0. *Read/Write.*		
Bit 1: Debug Mode. This bit when set to 1 puts the channel in debug mode operation using internal data. Value at reset 0. (*Read/Write)*		
Bit 2: Pile-up drop-out enable. When this bit is set to one, the channel drops any event that occurs in a pileup window else it just notifies pileup through a flag in the event header. Value at reset 1. *Read/Write.*		
Bits [3]: Channel Pre-sum Pre-sum all ADC data coming into the channel. Value a reset: 0. *Read/Write.*	**Pre-sum**	**Normal**
Bits [4]: Trigger mode. We have the following configuration: 0: Internal mode 1: <External> NOT IMPLEMENTED YET Value a reset: 0. *Read/Write.*	**External Trigger <TBD>**	**Internal Trigger**
Bit 5: CFD Tap delay setup. If set to 0, the tap delay is filling-up and no event can be acquired. *Read-only.*		
Bit 6: Tap delay 1 setup. If set to 0, the tap delay is filling-up and no event can be acquired. *Read-only.*		
Bit 7: Tap delay 2 setup. If set to 0, the tap delay is filling-up and no event can be acquired. *Read-only.*		
Bit 8: Tap delay 3 setup. If set to 0, the tap delay is filling-up and no event can be acquired. *Read-only.*		
Bit 9: Tap delay 4 setup. If set to 0, the tap delay is filling-up and no event can be acquired. *Read-only.*		

Bits [11:10]: Polarity validation. Value at reset 11. 01: only positive trigger considered 10: only negative trigger considered 11: both triggers considered 00: no trigger considered (the LED still fires externally though, as opposed to START/STOP) *Read/Write.*		
Bit 12: Select CFD. (not implemented)		
Bit 13: Select Pole-zero		
Bit 14: Enable pole-zero data to be sent as trace	**PZ output**	**Raw data**
Bit 15: Pre-buffer Ready flag		
Bit 16: VALIDATE_TRIG; timming SM block		
Bit 17: VALIDATE; timming SM block		
Bit 18: Proc_LEDsig; timming SM block		
Bit 19: CC_LED; timming SM block		
Bits[21:20]: STATE; timming SM block		
Bit 22: Base line restorer enable.	**Enabled**	**Disabled**
Bits[31:23]: TBD		

- There is one of this registers for each channels and their address are specified by the next table

Description	Address	Access	Width
Control/Status channel 0	**0x40**	**RW**	**32**
Control/Status channel 1	**0x44**	**RW**	**32**
Control/Status channel 2	**0x48**	**RW**	**32**
Control/Status channel 3	**0x4C**	**RW**	**32**
Control/Status channel 4	**0x50**	**RW**	**32**
Control/Status channel 5	**0x54**	**RW**	**32**
Control/Status channel 6	**0x58**	**RW**	**32**
Control/Status channel 7	**0x5C**	**RW**	**32**
Control/Status channel 8	**0x60**	**RW**	**32**
Control/Status channel 9	**0x64**	**RW**	**32**
Control/Status channel central contact low resolution (CCLR)	**0x68**	**RW**	**32**

Description	Address	Access	Width
LED Threshold & PZ multiplier	*****	**RW**	**32**
This is the Leading Edge Discriminator (LED) threshold. Only bits 17 to 0 are used. This is an unsigned value and the three lower bits are extra precision bit. (The dot is between bit 3 and 2). The value at reset is 0x1FFFF (full range so that no event should trigger). It is internally converted to a signed value depending on how the sign of the current sample is. The Pole zero multiplier default value is 1536 (600h). Pole zero time constant is given by t = (2^23/PoleZeroMultiplier)*10ns = 54.6us. The time constant increments in steps of (0.57/16)us.	**Bit Value**		
	1	**0**	
Bits[16: 0]: LED Threshold	Value		
Bits[31:20]: Pole Zero multiplier			

* There is one of this registers for each channels and their address are specified by the next table.

Description	Address	Access	Width
LED Threshold 0	**0x80**	**RW**	**32**
LED Threshold 1	**0x84**	**RW**	**32**

LED Threshold 2	**0x88**	**RW**	**32**
LED Threshold 3	**0x8C**	**RW**	**32**
LED Threshold 4	**0x90**	**RW**	**32**
LED Threshold 5	**0x94**	**RW**	**32**
LED Threshold 6	**0x98**	**RW**	**32**
LED Threshold 7	**0x9C**	**RW**	**32**
LED Threshold 8	**0XA0**	**RW**	**32**
LED Threshold 9	**0xA4**	**RW**	**32**
LED Threshold central contact low resolution	**0xA8**	**RW**	**32**

Description	**Address**	**Access**	**Width**
CFD Parameters	*****	**RW**	**32**
Bit 12-7: CFD delay. Value at reset is 0x3F giving a delay of 630ns. For the algorithm to perform correctly this delay must be smaller then the collection time. Bit 6-5: CFD fraction. It is a value indicating what fraction is used (see table 9 for values). Value at reset is “00” giving a fraction of 0.5. Bit 4-0: CFD threshold. It is an unsigned value of the CFD threshold. The dot is after bit 0. Value at reset is 0x10000 (160kev).	**Bit Value**		
	1	**0**	
Bits[4: 0]: CFD threshold	Value		
Bits[6: 5]: CFD fraction	Value		
Bits[12: 7]: CFD delay	Value		
Bits[31:13]: TBD			

- There is one of this registers for each channels and their address are specified by the next table.
- We may need some more parameters such as an integration time for the pulse going into the CFD (DCR, 8/31/2006)

Description	**Address**	**Access**	**Width**
CFD Parameters 0	**0XC0**	**RW**	**32**
CFD Parameters 1	**0xC4**	**RW**	**32**
CFD Parameters 2	**0xC8**	**RW**	**32**
CFD Parameters 3	**0xCC**	**RW**	**32**
CFD Parameters 4	**0xD0**	**RW**	**32**
CFD Parameters 5	**0xD4**	**RW**	**32**
CFD Parameters 6	**0xD8**	**RW**	**32**
CFD Parameters 7	**0xDC**	**RW**	**32**
CFD Parameters 8	**0xE0**	**RW**	**32**
CFD Parameters 9	**0xE4**	**RW**	**32**
CFD Parameters central contact low resolution	**0xE8**	**RW**	**32**

Description	**Address**	**Access**	**Width**
Window Timing Register	*****	**RW**	**32**
	Bit Value		
	1	**0**	
Bits[3: 0]: Channel pre-sum divider value Enabled with pre-sum enable bit X”0”: Divide by 1. (Default) X”1”: Divide by 2. X”2”: Divide by 4. X”3”: Divide by 8.	Value		
Bits[7: 4]: Channel pre-sum rate value Enabled with pre-sum enable bit X”0”: Sum 2 samples. (Default) X”1”: Sum 4 samples. X”2”: Sum 8 samples. X”3”: Sum 10 samples.	Value		
Bits[11: 8]: Multi-rate pre-sum divider value Active in the baseline and flat top data windows X”0”: Divide by 1. (Default) X”1”: Divide by 2. X”2”: Divide by 4. X”3”: Divide by 8.	Value		

Bits[15: 12]: Multi-rate pre-sum rate value Active in the baseline and flat top data windows X”0”: Sum 2 samples. (Default) X”1”: Sum 4 samples. X”2”: Sum 8 samples. X”3”: Sum 10 samples.	Value
Bits[26: 16]: Flat top window counter value Number of samples in the flat top window +4. Default: 256	Value
Bits[31:27]: TBD	

- There is one of this registers for each channels and their address are specified by the next table.

Description	Address	Access	Width
Window Timing Register 0	**0x100**	**RW**	**32**
Window Timing Register 1	**0x104**	**RW**	**32**
Window Timing Register 2	**0x108**	**RW**	**32**
Window Timing Register 3	**0x10C**	**RW**	**32**
Window Timing Register 4	**0x110**	**RW**	**32**
Window Timing Register 5	**0x114**	**RW**	**32**
Window Timing Register 6	**0x118**	**RW**	**32**
Window Timing Register 7	**0x11C**	**RW**	**32**
Window Timing Register 8	**0x120**	**RW**	**32**
Window Timing Register 9	**0x124**	**RW**	**32**
Window Timing Register central contact low resolution	**0x128**	**RW**	**32**

Description	Address	Access	Width
Rising Edge Window Register	*****	**RW**	**32**
		Bit Value	
		1	**0**
Bits[10: 0]: Post Rising Edge Window Counter Value Number of samples in the post rising edge window -18. Default: 512		Value	
Bits[22: 12]: Pre Rising Edge Window Counter Value Number of samples in the pre rising edge window +13. Default: 512		Value	
Bits[31:23]: TBD			

- There is one of this registers for each channels and their address are specified by the next table.

Description	Address	Access	Width
Rising edge window 0	**0x140**	**RW**	**32**
Rising edge window 1	**0x144**	**RW**	**32**
Rising edge window 2	**0x148**	**RW**	**32**
Rising edge window 3	**0x14C**	**RW**	**32**
Rising edge window 4	**0x150**	**RW**	**32**
Rising edge window 5	**0x154**	**RW**	**32**
Rising edge window 6	**0x158**	**RW**	**32**
Rising edge window 7	**0x15C**	**RW**	**32**
Rising edge window 8	**0x160**	**RW**	**32**
Rising edge window 9	**0x164**	**RW**	**32**
Rising edge window central contact low resolution	**0x168**	**RW**	**32**

Description	Address	Access	Width
Hit rate counter	*****	**R**	**32**
Counter that shows the hit rate on each channel. The rate is updated every second		**Bit Value**	
		1	**0**
Bits[31:0]: Hit rate		Value	

- There is one of this registers for each channels and their address are specified by the next table.

Description	Address	Access	Width
Hit rate counter ch 0	**0x180**	**RW**	**32**

Hit rate counter ch 1	**0x184**	**RW**	**32**
Hit rate counter ch 2	**0x188**	**RW**	**32**
Hit rate counter ch 3	**0x18C**	**RW**	**32**
Hit rate counter ch 4	**0x190**	**RW**	**32**
Hit rate counter ch 5	**0x194**	**RW**	**32**
Hit rate counter ch 6	**0x198**	**RW**	**32**
Hit rate counter ch 7	**0x19C**	**RW**	**32**
Hit rate counter ch 8	**0x1A0**	**RW**	**32**
Hit rate counter ch 9	**0x1A4**	**RW**	**32**
Hit rate counter central contact low resolution	**0x1A8**	**RW**	**32**

Description	**Address**	**Access**	**Width**
DAC	**0x400**	**RW**	**32**
Update This register defines which signal is going to be written into the dac output at every clock cycle.		**Bit Value**	
		1	**0**
Bit [3 : 0]: Mux selection : value 0 to 9 DAC outputs the LSB of the digitizer channels 0 to 9 respectively value A DAC outputs x7F value B DAC outputs x4A value C DAC outputs x15 value D DAC outputs xEB value E DAC outputs counter up value F DAC outputs bits 11 : 4 of this same register		Value	
Bits[11: 4]: Data to be written by the DAC			
Bits[31:12]: TBD		Value	

Description	**Address**	**Access**	**Width**
Slave front bus status (MSB)/ Control (LSB)	**0x480**	**RW**	**32**
This register contains several controls for the slave front bus (bits 15..0) and status for the same module (31..16).		**Bit Value**	
		1	**0**
Bits[0]: slave front bus enable. If disabled the module will ignore all the commands received by the front bus. Front bus clock is not affected by this control.		Enabled	Dis.
Bits[1]: Header memory validate overwrite. When set all data packages that are built in the pre buffer memory are going to be validated and readout. Basically this will be a trigger that happens when the CC fires all channels are readout automatically if they are set to TTCS mode (see channel control status register for more info in trigger modes).		Value	
Bits[2]: Reset Debug counters		Value	
Bits[3]: Reset counter for Sync commands received by the Front bus slave logic.			
Bits[4]: Reset counter for Imperative Sync commands received by the Front bus slave logic.			
Bits[5]: Reset counter for Latch status commands received by the Front bus slave logic.			
Bits[6]: Reset counter for Header memory validate commands received by the Front bus slave logic.			
Bits[7]: Reset counter for Header memory read slow data (hit pattern)commands received by the Front bus slave logic.			
Bits[8]: Reset counter for front end reset commands received by the Front bus slave logic.			
Bits[9]: Reset counter for front end calibration inject commands received by the Front bus slave logic.			
Bits[17:16]: slave board ID (00 Master board; 01,10 and 11 slave boards). They are set by the LSB of the G.A.		Read only	
Bits[31: 3]: TBD			

Description	**Address**	**Access**	**Width**

Latch status command central contact (channel 9) time stamp LSB	**0x484**	**RW**	**32**
After a latch command the TS on channel zero will be latched in this register.	**Bit Value**	**1**	**0**
Bits[31: 0]: TS [31:0]	Value		

Description	Address	Access	Width
Latch status command central contact (channel 9) time stamp MSB	**0x488**	**RW**	**32**
After a latch command the TS on channel zero will be latched in this register.	**Bit Value**	**1**	**0**
Bits[15: 0]: TS [47:32]	Value		
Bits[31:16]: TBD			

Send box18

Description	Address	Access	Width
Central contact (channel 9) time stamp LSB	**0x48C**	**RW**	**32**
This register shows the LSB of the central contact (channel 9) time stamp.	**Bit Value**	**1**	**0**
Bits[31: 0]: TS [31:0]	Value		

Description	Address	Access	Width
Central contact (channel 9) time stamp MSB	**0x490**	**RW**	**32**
This register shows the MSB of the central contact (channel 9) time stamp.	**Bit Value**	**1**	**0**
Bits[15: 0]: TS [47:32]	Value		
Bits[31:16]: TBD			

Description	Address	Access	Width
Sync counter	**0x494**	**RW**	**32**
This counter is increased by one every time the slave front bus logic received a sync command.	**Bit Value**	**1**	**0**
Bits[31: 0]: Counter	Value		

Description	Address	Access	Width
Imperative Sync counter	**0x498**	**RW**	**32**
This counter is increased by one every time the slave front bus logic received a imperative sync command.			**0**
Bits[31: 0]: Counter			

Description	Address	Access	Width
Latch status counter	**0x49C**	**RW**	**32**
This counter is increased by one every time the slave front bus logic received a latch command.			**0**
Bits[31: 0]: Counter			

Description	Address	Access	Width
Header memory validate counter	**0x4A0**	**RW**	**32**
This counter is increased by one every time the slave front bus logic received a header memory validate command. One data package should be written to the digitizer FIFO.	**Bit Value**	**1**	**0**
Bits[31: 0]: Counter	Value		

Description	Address	Access	Width
Header memory read slow data counter	**0x4A4**	**RW**	**32**
This counter is increased by one every time the slave front bus logic received a read slow data (hit pattern) command.			**0**
Bits[31: 0]: Counter			

Description	Address	Access	Width

Front end reset and calibration inject counters	**0x4A8**	**RW**	**32**
This counter is increased by one every time the slave front bus logic received a front end reset (MSB) or calibration inject (LSB) command.			**0**
Bits[15: 0]: calibration inject counter			
Bits[31:16]: front end reset counter			

Description	Address	Access	Width
Slave Front Bus Send Box 10	**0x4AC**	**RW**	**32**
Default value xDEADF00D.		**Bit Value**	
		1	**0**
Bits[31: 0]: TBD		Value	

Description	Address	Access	Width
Slave Front Bus Send Box 9	**0x4B0**	**RW**	**32**
Default value xDEADF00D.		**Bit Value**	
		1	**0**
Bits[31: 0]: TBD		Value	

Description	Address	Access	Width
Slave Front Bus Send Box 8	**0x4B4**	**RW**	**32**
Default value xDEADF00D.		**Bit Value**	
		1	**0**
Bits[31: 0]: TBD		Value	

Description	Address	Access	Width
Slave Front Bus Send Box 7	**0x4B8**	**RW**	**32**
Default value xDEADF00D.		**Bit Value**	
		1	**0**
Bits[31: 0]: TBD		Value	

Description	Address	Access	Width
Slave Front Bus Send Box 6	**0x4BC**	**RW**	**32**
Default value xDEADF00D.		**Bit Value**	
		1	**0**
Bits[31: 0]: TBD		Value	

Description	Address	Access	Width
Slave Front Bus Send Box 5	**0x4C0**	**RW**	**32**
Default value xDEADF00D.		**Bit Value**	
		1	**0**
Bits[31: 0]: TBD		Value	

Description	Address	Access	Width
Slave Front Bus Send Box 4	**0x4C4**	**RW**	**32**
Default value xDEADF00D.		**Bit Value**	
		1	**0**
Bits[31: 0]: TBD		Value	

Description	Address	Access	Width
Slave Front Bus Send Box 3	**0x4C8**	**RW**	**32**
Default value xDEADF00D.		**Bit Value**	
		1	**0**
Bits[31: 0]: TBD		Value	

Description	Address	Access	Width
Slave Front Bus Send Box 2	**0x4CC**	**RW**	**32**

Default value xDEADF00D.	Bit Value	
	1	0
Bits[31: 0]: TBD	Value	

Description	Address	Access	Width
Slave Front Bus Send Box 1	**0x4D0**	**RW**	**32**
Default value xDEADF00D.		**Bit Value**	
		1	**0**
Bits[31: 0]: TBD		Value	

Description	Address	Access	Width
Slave Front bus register 00	**0x4D4**	**RW**	**32**
Bits zero to nine are commands that are going to be executed by the slave front bus. They self clear after the execution of the command.		**Bit Value**	
		1	**0**
Bits[0]: Sync command		Value	
Bits[1]: Front End Calibration Inject		Value	
Bits[2]: Latch Status		Value	
Bits[3]: Front End reset		Value	
Bits[4]: Imperative sync		Value	
Bits[5]: HM validate		Value	
Bits[6]: HM read slow data command			
Bits[9: 6]: TBD		Value	
Bits[10: 19]: FB register 1		Value	
Bits[20: 29]: FB register 2		Value	
Bits[31: 30]: TBD		Value	

Description	Address	Access	Width
Slave Front bus register 01	**0x4D8**	**R**	**32**
Default value xDEADF00D.		**Bit Value**	
		1	**0**
Bits[9: 0]: Channels hit pattern		Value	
Bits[19:10]: FB register 1			
Bits[29:20]: FB register 2			
Bits[31:30]: TBD			

Description	Address	Access	Width
Slave Front bus register 02	**0x4DC**	**R**	**32**
Default value xDEADF00D.		**Bit Value**	
		1	**0**
Bits[9: 0]: FB register 0		Value	
Bits[19:10]: FB register 1			
Bits[29:20]: FB register 2			
Bits[31:30]: TBD			

Description	Address	Access	Width
Slave Front bus register 03	**0x4E0**	**R**	**32**
Default value xDEADF00D.		**Bit Value**	
		1	**0**
Bits[9: 0]: FB register 0		Value	
Bits[19:10]: FB register 1			
Bits[29:20]: FB register 2			
Bits[31:30]: TBD			
Description	**Address**	**Access**	**Width**
Slave Front bus register 04	**0x4E4**	**R**	**32**
Default value xDEADF00D.		**Bit Value**	
		1	**0**
Bits[9: 0]: FB register 0		Value	
Bits[19:10]: FB register 1			

Bits[29:20]: FB register 2	
Bits[31:30]: TBD	

Description	**Address**	**Access**	**Width**
Slave Front bus register 05	**0x4E8**	**R**	**32**
Default value xDEADF00D.		**Bit Value**	
		1	**0**
Bits[9: 0]: FB register 0		Value	
Bits[19:10]: FB register 1			
Bits[29:20]: FB register 2			
Bits[31:30]: TBD			

Description	**Address**	**Access**	**Width**
Slave Front bus register 06	**0x4EC**	**R**	**32**
Default value xDEADF00D.		**Bit Value**	
		1	**0**
Bits[9: 0]: FB register 0		Value	
Bits[19:10]: FB register 1			
Bits[29:20]: FB register 2			
Bits[31:30]: TBD			

Description	**Address**	**Access**	**Width**
Slave Front bus register 07	**0x4F0**	**R**	**32**
Default value xDEADF00D.		**Bit Value**	
		1	**0**
Bits[9: 0]: FB register 0		Value	
Bits[19:10]: FB register 1			
Bits[29:20]: FB register 2			
Bits[31:30]: TBD			

Description	**Address**	**Access**	**Width**
Slave Front bus register 08	**0x4F4**	**R**	**32**
Default value xDEADF00D.		**Bit Value**	
		1	**0**
Bits[9: 0]: FB register 0		Value	
Bits[19:10]: FB register 1			
Bits[29:20]: FB register 2			
Bits[31:30]: TBD			

Description	**Address**	**Access**	**Width**
Slave Front bus register 09	**0x4F8**	**R**	**32**
Default value xDEADF00D.		**Bit Value**	
		1	**0**
Bits[9: 0]: FB register 0		Value	
Bits[19:10]: FB register 1			
Bits[29:20]: FB register 2			
Bits[31:30]: TBD			

Description	**Address**	**Access**	**Width**
Slave Front bus register 10	**0x4FC**	**R**	**32**
Default value xDEADF00D.		**Bit Value**	
		1	**0**
Bits[9: 0]: FB register 0		Value	
Bits[19:10]: FB register 1			
Bits[29:20]: FB register 2			
Bits[31:30]: TBD			

Description	**Address**	**Access**	**Width**
Master Logic Control/Status 1	**0x500**	**RW**	**32**
		Bit Value	

	1	0

Bits[0]: Master Logic Enabled	Enabled	Disabled
Bits[1]: Start Snap Shot	Value	
Bits[2]: SD_Local_LE	Value	
Bits[3]: SD_Line_LE	Value	
Bits[4]: SD_PEM(0)	Value	
Bits[5]: SD_PEM(1)	Value	
Bits[6]: FB_Debug_Command (1 data transfer only)	Value	
Bits[7]: FB_Debug_Command (continuous data transfer)	Value	
Bits[8]: Reset Serdes lost lock flag	If 1 keeps the flag on reset.	
Bits[9]: Snap Shot reset		
Bits[10]: Snap Shot read data select	SD_TX	SD_RX
Bits [11]: Snap Shot trigger on CC_LED event	Value	
Bits [12]: Enable Serdes Tx dc balance component	Value	
Bits [13]: Enable central contact low resolution data package readout.	Enabled	Disabled
Bits [14]: Enable master header readout.	Enabled	Disabled
Bits [15]: Enable slave header readout.	Enabled	Disabled
Bits [16]: FIFO Full	Value	
Bits [17]: SD_Lock	Value	
Bits [18]: HasDataFlag	Value	
Bits [19]: FB_Busy high level logic	Value	
Bits [20]: FB_Busy low level logic	Value	
Bits [21]: Snap Shot Busy RX	Value	
Bits [22]: Snap Shot Empty RX	Value	
Bits [23]: Serdes RX SM in sync with TTCL system	Value	
Bits [24]: Serdes lost lock flag	Value	
Bits [25]: Snap Shot Busy TX	Value	
Bits [26]: Snap Shot Empty TX	Value	
Bits[31:27]: TBD	Value	

Description	**Address**	**Access**	**Width**
Master Logic Control/Status 2	**0x504**	**RW**	**32**
		Bit Value	
		1	**0**
Bits[15: 0]: Number of clock cycles that the CC_LED flag will stay high. The CC_LED flag signal is sent to the TTCL through one of the serdes auxiliary IOs.		Value	
Bits[16]: Overwrite time stamp comparison. When a trigger decision is made and this bit is enabled then every trigger decision will generate a HM validate command to all HM in the digitizers.			
Bits[17]: Reset master logic (main and front bus) debug counters			
Bits[18]: Reset counter for Sync commands received by the Front bus slave logic.			
Bits[19]: Reset counter for Imperative Sync commands received by the Front bus slave logic.			
Bits[20]: Reset counter for Latch status commands received by the Front bus slave logic.			
Bits[21]: Reset counter for Header memory validate commands received by the Front bus slave logic.			
Bits[22]: Reset counter for Header memory read slow data (hit pattern)commands received by the Front bus slave logic.			
Bits[23]: Reset counter for front end reset commands received by the Front bus slave logic.			
Bits[24]: Reset counter for front end calibration inject commands received by the Front bus slave logic.			
Bits[25]: Reset counter for package error on serdes RX module			
Bits[31: 25]: TBD		Value	

Description	**Address**	**Access**	**Width**
DeltaT155_DeltaT255	**0x508**	**RW**	**32**
Limits of the time windows for the comparison type x55.		**Bit Value**	
		1	**0**
Bits[0: 12]: Time to be added		Value	

Bits[16: 28]: Time to be subtracted	Value

Description	**Address**	**Access**	**Width**
DeltaT15A_DeltaT25A	**0x50C**	**RW**	**32**
Limits of the time windows for the comparison type x5A.		**Bit Value**	
		1	**0**
Bits[0: 12]: Time to be added		Value	
Bits[16: 28]: Time to be subtracted		Value	

Description	**Address**	**Access**	**Width**
DeltaT1A5_DeltaT2A5	**0x510**	**RW**	**32**
Limits of the time windows for the comparison type xA5.		**Bit Value**	
		1	**0**
Bits[0: 12]: Time to be added		Value	
Bits[16: 28]: Time to be subtracted		Value	

Description	**Address**	**Access**	**Width**
SnapShot	**0x514**	**RW**	**32**
This register is used to read all the data captured by the snap shot memory. This memory stores the data received by the SerDes.		**Bit Value**	
		1	**0**
Bits[0: 9]:		Value	
Bits[10: 19]:		Value	
Bits[20: 29]:		Value	
Bits[31: 30]: TBD		Value	

Description	**Address**	**Access**	**Width**
XTAL ID	**0x518**	**RW**	**32**
XTAL ID used on the slow data that is sent to the TTCL.		**Bit Value**	
		1	**0**
Bits[7: 0]: XTAL ID		Value	
Bits[31: 8]: TBD		Value	

Description	**Address**	**Access**	**Width**
Length of Time to Get Hit Pattern	**0x51C**	**RW**	**32**
If necessary to add a delay on the CC_LED signal that goes to the slave boards this register shows how big it is. Default zero		**Bit Value**	
		1	**0**
Bits[15: 0]: Time		Value	
Bits[31: 16]: TBD		Value	

Description	**Address**	**Access**	**Width**
Front Bus Debug Register	**0x520**	**RW**	**32**
Data that tell what information is to be read/written from the FB.		**Bit Value**	
		1	**0**
Bits[9: 0]: Write Data			
Bits[11: 10]: 00			
Bits[19: 12]: Address		Value	
Bits[27: 20]: TBD		Value	
Bit [28]: RNW		Value	
Bits[31: 29]: TBD		Value	

For example:
If the boards are located as follow: Master board on slot 6, slave boards on slots 5, 4 and 3.

Write command through the front bus
Write on 520h 30AAh
Write on 500h 41h
Write on 500h 01h
This will write on address 4D8h on the slave board on slot 4 through the front bus.

Read command through the front bus
Write on 520h 10003000h

Write on 500h 41h
Write on 500h 01h
This will transfer the data from address 4D8h on the slave board on slot 4 through the front bus to address 558h in the master board.

Description	Address	Access	Width
Test digitizer Tx TTCL (not implemented)	**0x524**	**RW**	**32**
In order to test some of the functionalities of the master logic for debug purpose there is a module that generates the TTCL data package inside the SD block. This register configures that module as described below.		**Bit Value**	
		1	**0**
Bits[31]: Test Enable		Value	
Bits[30:18]: TBD			
Bits[17]: Send data package (one time only, rising edge)			
Bits[16]: Write enable (store user def data on the rising edge)			
Bits[15:11]: Address to write the package for the slow data			
Bits[10]: Fast Data – Central contact LED (Discriminator)			
Bits[9]: Fast Data – Error flag			
Bits[8]: Fast Data – Central contact pileup			
Bits[7: 0]: Slow Data			

Description	Address	Access	Width
Test digitizer Rx TTCL	**0x528**	**RW**	**32**
In order to test some of the functionalities of the master logic for debug purpose there is a module that generates the TTCL data package inside the SD block. This register configures that module as described below.		**Bit Value**	
		1	**0**
Bits[31]: Rx Test Enable		Value	
Bits[30:29]: Data package selection			
Bits[28]: Send alternate data package (one time only, rising edge)			
Bits[27:26]: Alternate data package selection			
Bits[25]: Enable DC balance.			
Bits[24]: TBD			
Bits[23]: Write data command (store user def data on the rising edge)			
Bits[22:16]: Address to write the package for the slow data. Addresses valid range from zero to 99.			
Bits[15: 0]: Data			

Description	Address	Access	Width
Slave Front Bus Send Box 10	**0x52C**	**RW**	**32**
Default value xDEADF00D.		**Bit Value**	
		1	**0**
Bits[31: 0]: TBD		Value	

Description	Address	Access	Width
Slave Front Bus Send Box 9	**0x530**	**RW**	**32**
Default value xDEADF00D.		**Bit Value**	
		1	**0**
Bits[31: 0]: TBD		Value	

Description	Address	Access	Width
Slave Front Bus Send Box 8	**0x534**	**RW**	**32**
Default value xDEADF00D.		**Bit Value**	
		1	**0**
Bits[31: 0]: TBD		Value	

Description	Address	Access	Width
Slave Front Bus Send Box 7	**0x538**	**RW**	**32**
Default value xDEADF00D.		**Bit Value**	
		1	**0**

Bits[31: 0]: TBD	Value

Description	Address	Access	Width
Slave Front Bus Send Box 6	**0x53C**	**RW**	**32**
Default value xDEADF00D.		**Bit Value**	
		1	**0**
Bits[31: 0]: TBD		Value	

Description	Address	Access	Width
Slave Front Bus Send Box 5	**0x540**	**RW**	**32**
Default value xDEADF00D.		**Bit Value**	
		1	**0**
Bits[31: 0]: TBD		Value	

Description	Address	Access	Width
Slave Front Bus Send Box 4	**0x544**	**RW**	**32**
Default value xDEADF00D.		**Bit Value**	
		1	**0**
Bits[31: 0]: TBD		Value	

Description	Address	Access	Width
Slave Front Bus Send Box 3	**0x548**	**RW**	**32**
Default value xDEADF00D.		**Bit Value**	
		1	**0**
Bits[31: 0]: TBD		Value	

Description	Address	Access	Width
Slave Front Bus Send Box 2	**0x54C**	**RW**	**32**
Default value xDEADF00D.		**Bit Value**	
		1	**0**
Bits[31: 0]: TBD		Value	

Description	Address	Access	Width
Slave Front Bus Send Box 1	**0x550**	**RW**	**32**
Default value xDEADF00D.		**Bit Value**	
		1	**0**
Bits[31: 0]: TBD		Value	

Description	Address	Access	Width
FrontBus Registers 0	**0x554**	**RW**	**32**
		Bit Value	
		1	**0**
Bits[0]: Sync command		Value	
Bits[1]: Front End Calibration Inject		Value	
Bits[2]: Latch Status		Value	
Bits[3]: Front End reset		Value	
Bits[4]: Imperative sync		Value	
Bits[5]: HM validate (Write Data package into external FIFO)		Value	
Bits[6]: Read Slow data info (hit pattern to send it to trigger system)			
Bits[9: 6]: TBD		Value	
Bits[10: 19]: TBD		Value	
Bits[20: 29]: TBD		Value	
Bits[31: 30]: TBD		Value	

Description	Address	Access	Width
FrontBus Registers 1	**0x558**	**RW**	**32**
		Bit Value	
		1	**0**

Bits[7: 0]: Serdes slow data HP_A	Value
Bits[15: 8]: Serdes slow data HP_B	Value
Bits[23: 16]: Serdes slow data HP_C	Value
Bits[29: 24]: Serdes slow data HP_D(5:0)	
Bits[31: 30]: TBD	Value

Description	Address	Access	Width
FrontBus Registers 2	**0x55C**	**RW**	**32**
		Bit Value	
		1	**0**
Bits[1: 0]: Serdes slow data HP_D(7:6)		Value	
Bits[9: 2]: Serdes slow data HP_E		Value	
Bits[29: 10]: TBD		Value	
Bits[31: 30]: TBD		Value	

Description	Address	Access	Width
FrontBus Registers 3	**0x560**	**RW**	**32**
		Bit Value	
		1	**0**
Bits[0: 9]:		Value	
Bits[10: 19]:		Value	
Bits[20: 29]:		Value	
Bits[31: 30]: TBD		Value	

Description	Address	Access	Width
FrontBus Registers 4	**0x564**	**RW**	**32**
		Bit Value	
		1	**0**
Bits[0: 9]:		Value	
Bits[10: 19]:		Value	
Bits[20: 29]:		Value	
Bits[31: 30]: TBD		Value	

Description	Address	Access	Width
FrontBus Registers 5	**0x568**	**RW**	**32**
		Bit Value	
		1	**0**
Bits[0: 9]:		Value	
Bits[10: 19]:		Value	
Bits[20: 29]:		Value	
Bits[31: 30]: TBD		Value	

Description	Address	Access	Width
FrontBus Registers 6	**0x56C**	**RW**	**32**
		Bit Value	
		1	**0**
Bits[0: 9]:		Value	
Bits[10: 19]:		Value	
Bits[20: 29]:		Value	
Bits[31: 30]: TBD		Value	

Description	Address	Access	Width
FrontBus Registers 7	**0x570**	**RW**	**32**
		Bit Value	
		1	**0**
Bits[0: 9]:		Value	
Bits[10: 19]:		Value	
Bits[20: 29]:		Value	
Bits[31: 30]: TBD		Value	

Description	Address	Access	Width
FrontBus Registers 8	**0x574**	**RW**	**32**
		Bit Value	
		1	**0**
Bits[0: 9]:		Value	
Bits[10: 19]:		Value	
Bits[20: 29]:		Value	
Bits[31: 30]: TBD		Value	

Description	Address	Access	Width
FrontBus Registers 9	**0x578**	**RW**	**32**
		Bit Value	
		1	**0**
Bits[0: 9]:		Value	
Bits[10: 19]:		Value	
Bits[20: 29]:		Value	
Bits[31: 30]: TBD		Value	

Description	Address	Access	Width
FrontBus Registers 10	**0x57C**	**RW**	**32**
		Bit Value	
		1	**0**
Bits[0: 9]:		Value	
Bits[10: 19]:		Value	
Bits[20: 29]:		Value	
Bits[31: 30]: TBD		Value	

Description	Address	Access	Width
Sync counter	**0x580**	**RW**	**32**
This counter is increased by one every time the master logic received a sync command.		**Bit Value** **1**	 **0**
Bits[31: 0]: Counter		Value	

Description	Address	Access	Width
Imperative Sync counter	**0x584**	**RW**	**32**
This counter is increased by one every time the master logic received an imperative sync command.			**0**
Bits[31: 0]: Counter			

Description	Address	Access	Width
Latch status counter	**0x588**	**RW**	**32**
This counter is increased by one every time the master logic received a latch command.			**0**
Bits[31: 0]: Counter			

Description	Address	Access	Width
Header memory validate counter	**0x58C**	**RW**	**32**
This counter is increased by one every time the master logic received a trigger decision command. One data package should be written to the digitizer FIFO.		**Bit Value** **1**	 **0**
Bits[31: 0]: Counter		Value	

Description	Address	Access	Width
Header memory read slow data counter	**0x590**	**RW**	**32**
This counter is increased by one every time the master logic received a demand slow data (hit pattern) command.			**0**
Bits[31: 0]: Counter			

Description	Address	Access	Width

Front end reset and calibration inject counters	**0x594**	**RW**	**32**
This counter is increased by one every time the master logic received a front end reset (MSB) or calibration inject (LSB) command.			**0**
Bits[15: 0]: calibration inject counter			
Bits[31:16]: front end reset counter			

Description	Address	Access	Width
Sync counter	**0x598**	**RW**	**32**
This counter is increased by one every time the master front bus issues a sync command.		**Bit Value**	
		1	**0**
Bits[31: 0]: Counter		Value	

Description	Address	Access	Width
Imperative Sync counter	**0x59C**	**RW**	**32**
This counter is increased by one every time the master front bus issues an imperative sync command.			**0**
Bits[31: 0]: Counter			

Description	Address	Access	Width
Latch status counter	**0x5A0**	**RW**	**32**
This counter is increased by one every time the master front bus issues a latch command.			**0**
Bits[31: 0]: Counter			

Description	Address	Access	Width
Header memory validate counter	**0x5A4**	**RW**	**32**
This counter is increased by one every time the master front bus issues a header memory validate command. One data package should be written to the digitizer FIFO.		**Bit Value**	
		1	**0**
Bits[31: 0]: Counter		Value	

Description	Address	Access	Width
Header memory read slow data counter	**0x5A8**	**RW**	**32**
This counter is increased by one every time the master front bus issues a read slow data (hit pattern) command.			**0**
Bits[31: 0]: Counter			

Description	Address	Access	Width
Front end reset and calibration inject counters	**0x5AC**	**RW**	**32**
This counter is increased by one every time the master front bus issues a front end reset (MSB) or calibration inject (LSB) command.			**0**
Bits[15: 0]: calibration inject counter			
Bits[31:16]: front end reset counter			

Description	Address	Access	Width
Serdes data package error	**0x5B0**	**RW**	**32**
Every time the data package coming from the trigger gets out of sync a Data package error command happens. This counter shows how many time this event happened.			**0**
Bits[15: 0]: counter			

Description	Address	Access	Width
CC_LED enable	**0x5B4**	**RW**	**32**
This register enable which signal(s) that will drive the CC_LED signal that goes into the Master logic. This allows any channel to be cc_led and/or an external signal from the AUX_IO to be it.		**Bit Value**	
		1	**0**
Bits[0]: LED channel 0 enabled		Default 0	
Bits[1]: LED channel 1 enabled		Default 0	
Bits[2]: LED channel 2 enabled		Default 0	

Bits[3]: LED channel 3 enabled	Default 0
Bits[4]: LED channel 4 enabled	Default 0
Bits[5]: LED channel 5 enabled	Default 0
Bits[6]: LED channel 6 enabled	Default 0
Bits[7]: LED channel 7 enabled	Default 0
Bits[8]: LED channel 8 enabled	Default 0
Bits[9]: LED channel 9 enabled	Default 1
Bits[10]: AUX_IO input enabled	Default 0
Bits[31:11]:	

Description	**Address**	**Access**	**Width**
Debug data buffer address	**0x780**	**RW**	**32**
Used by debug module (see section XIII of the GRETINA VHDL modules description document).		**Bit Value**	
		1	**0**
Bits[31: 0]:		Value	

Description	**Address**	**Access**	**Width**
Debug data buffer data	**0x784**	**RW**	**32**
Used by debug module (see section XIII of the GRETINA VHDL modules description document).		**Bit Value**	
		1	**0**
Bits[31: 0]:		Value	

Description	**Address**	**Access**	**Width**
LED Flag window	**0x788**	**RW**	**32**
Number of clock cycles the LED flag is kept high on the serdes fast data as well as on the RJ45 Auxiliary signals. Range varies from 1 (10ns) to 1024 (~1us).		**Bit Value**	
		1	**0**
Bits[31: 0]:		Value	

Description	**Address**	**Access**	**Width**
AUX_IO_READ	**0x800**	**RW**	**32**
Data read from the Auxiliary IO		**Bit Value**	
		1	**0**
Bits[11: 0]: Auxiliary IO input value		Value	
Bits[31:12]: TBD		Value	

Description	**Address**	**Access**	**Width**
AUX_IO_WRITE	**0x804**	**RW**	**32**
Data written into the auxiliary IO		**Bit Value**	
		1	**0**
Bits[11: 0]: Auxiliary IO output value		Value	
Bits[31:12]: TBD		Value	

Description	**Address**	**Access**	**Width**
AUX_IO_CONFIG	**0x808**	**RW**	**32**
Enable signals for the Auxiliary IO's drivers.		**Bit Value**	
		1	**0**
Bits[0]: EN_AUX2A_TX			
Bits[1]: EN_AUX2A_RX			
Bits[2]: EN_AUX2B_TX			
Bits[3]: EN_AUX2B_RX			
Bits[4]: EN_AUX2C_TX			
Bits[5]: EN_AUX2C_RX			
Bits[6]: EN_AUX2D_TX			
Bits[7]: EN_AUX2D_RX			
Bits[8]: EN_AUX2E_TX			
Bits[9]: EN_AUX2E_RX			
Bits[10]: EN_AUX2F_TX			
Bits[11]: EN_AUX2F_RX			

Bits[15: 12]: MUX control*	0000 -> Debug mode 0001 -> Mode 1 of operation 0010 -> Mode 2 of operation others-> Mode 3 of operation
Bits[31:12]:	
Bits[31:24]: Test mux – just a test it should go to 874h	Value

* Mode 1 of operation:

Auxiliary IO signal	**Signal type (input/output)**	**Routed signal to the Aux_IO**
Aux IO (0)	Output	Global Trigger signal
Aux IO (1)	Output	100MHz clock
Aux IO (2)	Input	Global Validate/External trigger
Aux IO (3)	Input	Sync signal NOT (‘1’ or 5V resets time stamp)
Aux IO (4)	Input	Fake external CC_LED*- This signal will be feed to an OR gate with the CC_LED to allow an external signal to trigger all digitizers and trigger system.
Aux IO (5)	Input	TBD
Aux IO (6)	Input	TBD
Aux IO (7)	Input	TBD
Aux IO (8)	Input	TBD
Aux IO (9)	Input	TBD
Aux IO (10)	Input	TBD

* Mode 2 of operation:

Auxiliary IO signal	**Signal type (input/output)**	**Routed signal to the Aux_IO**
Aux IO (0)	Output	Trigger channel (0)
Aux IO (1)	Output	Trigger channel (1)
Aux IO (2)	Output	Trigger channel (2)
Aux IO (3)	Output	Trigger channel (3)
Aux IO (4)	Output	Trigger channel (4)
Aux IO (5)	Output	Trigger channel (5)
Aux IO (6)	Output	Trigger channel (6)
Aux IO (7)	Output	Trigger channel (7)
Aux IO (8)	Output	Trigger channel (8)
Aux IO (9)	Output	Trigger channel (9)
Aux IO (10)	Input	TBD

* Mode 3 of operation:

Auxiliary IO signal	**Signal type (input/output)**	**Routed signal to the Aux_IO**
Aux IO (0)	Input	TBD
Aux IO (1)	Input	TBD
Aux IO (2)	Input	TBD
Aux IO (3)	Input	TBD
Aux IO (4)	Input	TBD
Aux IO (5)	Input	TBD
Aux IO (6)	Input	TBD
Aux IO (7)	Input	TBD
Aux IO (8)	Input	TBD
Aux IO (9)	Input	TBD
Aux IO (10)	Input	TBD

Description	**Address**	**Access**	**Width**
FB_READ	**0x820**	**RW**	**32**
	Bit Value		
	1	**0**	
Bits[11: 0]: Auxiliary IO input value	Value		
Bits[31:12]: TBD	Value		

Description	Address	Access	Width
FB_WRITE	**0x824**	**RW**	**32**
		Bit Value	
		1	**0**
Bits[11: 0]: Auxiliary IO output value		Value	
Bits[31:12]: TBD		Value	

Description	Address	Access	Width
FB_CONFIG	**0x828**	**RW**	**32**
Not implemented in the present VHDL code.		**Bit Value**	
		1	**0**
Bits[0]: FB_CLK_OUT_EN		**Enable**	**Disable**
Bits[1]: FB_SD_DP_EN		**Enable**	**Disable**
Bits[2]: FB_DATA_DIR		**Transmit**	**Receive**
Bits[3]: FB_DATA_EN		**Enable**	**Disable**
Bits[4]: FB_CTRLO_DIR		**Transmit**	**Receive**
Bits[5]: FB_CTRL0_EN		**Enable**	**Disable**
Bits[6]: FB_CTRL1_DIR		**Transmit**	**Receive**
Bits[7]: FB_CTRL1_EN		**Enable**	**Disable**
Bits[8]: FB_DATA_TS – tri state date		**Tri state**	**Enabled**
Bits[9]: FB_ADDR_TS – tri state address		**Tri state**	**Enabled**
Bits[10]: FB_RNW_TS – tri state read not write signal		**Tri state**	**Enabled**
Bits[11]: FB_STRB_TS – tri state strobe		**Tri state**	**Enabled**
Bits[12]: FB_SPARE_TS – tri state spare signal		**Tri state**	**Enabled**
Bits[13]: FB_LED_TS – tri state leading edge signal		**Tri state**	**Enabled**
Bits[14]: FB_WORB_TS – tri state wired or signal		**Tri state**	**Enabled**
Bits[16: 15]: MUX control		00 -> Debug mode 01 -> Normal operation	
Bits[31:12]: TBD		Value	

Description	Address	Access	Width
SD_READ	**0x840**	**RW**	**32**
Data read from the Serdes.		**Bit Value**	
		1	**0**
Bits[17: 0]: SD input value		Value	
Bits[30:18]: TBD			
Bits[31]: Lock signal		Value	

Description	Address	Access	Width
SD_WRITE	**0x844**	**RW**	**32**
Data written into the Serdes.		**Bit Value**	
		1	**0**
Bits[18: 0]: SD output value		Value	
Bits[31:12]: TBD		Value	

Description	Address	Access	Width
SD_CONFIG	**0x848**	**RW**	**32**
Configuration signals for the SD x31 serdes disabled x22 Local loop and sync mode x02 Loop back mode Before using a loop back mode it is necessary to put the serdes in local loop and sync so the SD can lock and then move to the Loop back mode. If the DCM doesn’t lock it is necessary to reset it. For reliable use it is recommended to reset the ten channel module. The reset bit are set on reset and they should be cleared only after the source clock for the main FPGA is valid. Default value x1E31		**Bit Value** **1**	**0**
Bits[0]: SD_RPDWN			
Bits[1]: SD_LOCAL_LE			
Bits[2]: SD_PEM_0			

Bits[3]: SD_PEM_1		
Bits[4]: SD_TPDWN		
Bits[5]: SD_SYNC		
Bits[6]: SD_LINE_LE		
Bits[8: 7]: MUX control	00 -> Debug mode 01 -> Normal operation	
Bits[9]: DCMResetCommand : resets both at the same time using a pre-defined time between the two.		
Bits[10]: DCMReset0 : when asserted keeps the DCM for the 50->100MHZ clock under reset.		
Bits[11]: DCMReset1: when asserted keeps the DCM for the 100MHZ clock under reset.		
Bits[12]: TenChannelResetCommand : Reset the ten channel module.		
Bits[31:11]: TBD	Value	

Description	Address	Access	Width
ADC_CONFIG	**0x84C**	**RW**	**32**
		Bit Value	
		1	**0**
Bits[1:0]: Clock mux 00 => CLK 01 => CLK90 10 => CLK180 11 => CLK270			
Bits[2]: TBD			
Bits[11: 3]: TBD			
Bits[31:12]: TBD		Value	

Description	Address	Access	Width
Self Trigger Enable	**0x860**	**RW**	**32**
Set bit one to start the self trigger process. This bit resets itself when the number of trigger events is completed. This self trigger is valid on the external trigger mode only.		**Bit Value**	
		1	**0**
Bit 1: Self trigger start		Value	
Bits[31:2]: TBD		Value	

Description	Address	Access	Width
Self Trigger Period	**0x864**	**RW**	**32**
To get the period in time multiply this register by 10ns.		**Bit Value**	
		1	**0**
Bits[31:0]: Period between trigger events.		Value	

Description	Address	Access	Width
Self Trigger Count	**0x868**	**RW**	**32**
		Bit Value	
		1	**0**
Bits[31:0]: Number of trigger events.		Value	

Description	Address	Access	Width
Self Trigger Count	**0x868**	**RW**	**32**
		Bit Value	
		1	**0**
Bits[31:0]: Number of trigger events.		Value	

Description	Address	Access	Width
FIFOInterfaceSMReg	**0x870**	**RW**	**32**

FIFOInterfaceSMReg(31 downto 28) <= "0000";
FIFOInterfaceSMReg(27 downto 19) <= ADDRESS;
FIFOInterfaceSMReg(18) <= READDone;
FIFOInterfaceSMReg(17) <= BEINGRead;
FIFOInterfaceSMReg(16 downto 4) <= PREBUFFER_READY;
FIFOInterfaceSMReg(3 downto 0) <= STATE;
	Bit Value	

	1	**0**
Bits[31:0]: Number of trigger events.	Value	

Description	Address	Access	Width
Test signals register	**0x874**	**RW**	**32**
The digitizer board has 16 test signal plus a test clock (see connector P11). This register sets which signals are going to be output into those signals.	**Bit Value**		
	1	**0**	
Bits[7:0]: Selects signal to be output into the test pins.			
Bits[31:12]: TBD	Value		

Selection 00

```
Testsig00(15)                   <= FIFO_PAF_N;
Testsig00(14 downto 0)          <= TenChannelDebugsig(14 downto 0);
```

Where

```
        DEBUG_CHANNEL_SELECT => TenChannelDEBUG (3 DOWNTO 0), --open
        DEBUG_SIZE => TenChannelDEBUG (12 DOWNTO 4) --open
        TenChannelDEBUG(13) <= ENABLEout(0);
        TenChannelDEBUG(14) <= PREBUFFER_ACK(0);
```

Selection 01

```
Testsig01(15)       <= LED_TRIGGERsig(9);
Testsig01(14)       <= CC_LEDsig;
Testsig01(13)       <= FB_LED_OUTsig;
Testsig01(12)       <= HM_Validatesig
Testsig01(11)       <= FIFO_OE_N;
Testsig01(10)       <= FIFO_WEN_N;
Testsig01(9)        <= StatusReg9sig(15);-- <= PREBUFFER_READY;
Testsig01(8 downto 7)<= StatusReg9sig(11 downto 10);-- <= TRIG_POLARITY;
Testsig01(6 downto 5)<= StatusReg9sig(4 downto 3);-- <= TRIG_MODE;
Testsig01(4)        <= StatusReg9sig(2);-- <= PILEUP;
Testsig01(3)        <= StatusReg9sig(0);-- <= ENABLE;
Testsig01(2)        <= PREBUFFER_ACKMSDebugsig(9);
Testsig01(1)        <= PREBUFFER_READYDebugsig(9);
Testsig01(0)        <= FB_SYNCsig;
```

Selection 02

```
Testsig02(15)       <= LED_TRIGGERsig(9);
Testsig02(14)       <= CC_LEDsig;
Testsig02(13)       <= FB_LED_OUTsig;
Testsig02(12)       <= HM_Validatesig
Testsig02(11)       <= FIFO_OE_N;
Testsig02(10)       <= FIFO_WEN_N;
Testsig02(9 downto 5)<= PREBUFFER_ACKMSDebugsig(9 downto 5);
Testsig02(4 downto 0)<= PREBUFFER_READYDebugsig(9 downto 5);
```

Selection 03

```
Testsig03(15)       <= LED_TRIGGERsig(9);
Testsig03(14)       <= CC_LEDsig;
Testsig03(13)       <= FB_LED_OUTsig;
Testsig03(12)       <= HM_Validatesig
Testsig03(11)       <= FIFO_OE_N;
Testsig03(10)       <= FIFO_WEN_N;
Testsig03(9 downto 5)<= PREBUFFER_ACKMSDebugsig(4 downto 0);
Testsig03(4 downto 0)<= PREBUFFER_READYDebugsig(4 downto 0);
```

Epics commands

Esc k brings

l RIGHT

h left

x delete
i insert esc
return
r single character replacement