MCOR Hardware Design Notes

Version 7.0

Rev 2 Hardware

November 27, 2023

Jeff Olsen

Table of Contents

Revision History

Document

Version	Date	
1.0	7/20/2012	USB Waveform, no interrupts
2.0	7/20/2012	Added Interrupt section
		Updated Wave Form Acquisition section
	7/31/2012	Added EVR Control Registers
4.0	02/06/13	~~Change averaging~~
		Add Registers for Oversampling and Reference Control
6.0		Rev 2 Hardware
		Added separate Fullscale for Monitor and Feedback ADC’s
7.0	07/29/15	Updated FPGA documentation

FPGA

Version		Date		
MCOR	EVR			
2.00	01	7/12/2012	USB Waveform, no interrupts	
			Increased PCIe Address space to accommodate Wave Form Memory	
		7/31/2012	Added EVR Control Registers	
			Added EVR Functionality	
4.00		9/13/12	Worked on EVR Functionality	
5.1		02/06/13	~~Change averaging. Added 64, 128 samples~~	
			Add ADC Oversampling control which was left off the design	
			Add ADC External/Internal Reference Control	
			Changed UCF file to include Oversampling bits	
			Changed USB State Machine to accommodate plugging in the cable This generates a bunch of long transitions on the RXF# and TXE# signals	
			Changed Start Convert. After Power-up, the ADC starts, when done it starts the Averaging. When the averaging is done, it starts the ADC again. The conversion rate is now dependent on the oversampling	
5.2		03/11/13	Fixed EVR issues	
6.0		02/06/14	Rev 2 Hardware	
		03/06/14	Conditioned USB Data with Power Enable from USB Chip	
7.0		07/29/14	Fixed a bug in the PCIe writing of the ADC Full Scale Changed wr_strb to Fullscalewr in ADCChannelSeq	
8.0		08/01/14	Added separate eGains for Monitor and Feedback ADC's	
8.1		08/14/14	Changed Ripple Init from 1Khz to 10hz since at max oversampling the ADC run at ~2-3Khz	
			Changed Default oversampling to 6 instead of 0	
8.2		09/15/14	Saturate DAC output so it won't wrap from + full-scale to – Full-Scale Fix problem with writing Setpoint to Faulted channel	
8.5		10/15/14	Set DAC gain to 10.2564 Something happened to the project and I had to restore it from 10/15 am	
8.7		2/12/15	Fixed MCOR Inhibit bus. Or with Water fault in xilinx	
			Add 62.5Mhz clock divider from PCIe bus instead of 100Mhz from external clock.	
8.71		2/18/15	Water Sum Fault should be low true.	
9.01		4/7/15	Fixed EVR version register, 02C.	
9.02	02	6/9/15	Removed Event Interrupt Enable bit from the FIFO Status. (Interupt_Register.vhd)	jjo
			Changed MCOR Version to 9.02, EVR Version to 2	jjo
9.03	03	6/26/15	Changed BAR2, EVR to 32k from 16k	
			Temporarily commented out MCOR memory to make space for testing interrupts. Wfmcapture.vhd Used LED to output interrupt signals as test points Mcor_x.vhd Output interrupttst.mcs for now	

Version		Date		
MCOR	EVR			
9.10	03	7/2/15	Changed Timing IRQ from using latched versions to rising edges. Timing – Interrupt_Register.vhd	
10.00	03	7/21/15	Found Interrupt problem. It was in Xilinx’s Tx/RX application interface. Rx done is a pulse to the Tx engine. If the Tx engine is not ready because the interface is busy servicing the interrupt, the Tx engine misses the Rx Done and the cycle hangs. Added a state in Xilinx’s Tx engine to wait for the Rx Done.	
		7/29/15	Fixed error reading the Event FIFO	
10.10	03	7/30/15	Changed Event FIFO to First word Fall-thru mode	

BAR 0 Memory Map – MCOR Registers

BAR 0 Address	
0x00000 – 0x003C0	Channel Control Registers
0x00400 – 0x0043C	Bulk Supply Registers
0x00440 – 0x0047C	MCOR ADC Control Registers
0x00480 – 0x004BC	Fault Registers
0x004C0 – 0x004FC	Waveform Capture Registers
0x00500 – 0x0053C	Interlocks and Magnet Faults
0x00540 – 0x0057C	MCOR Voltage Monitor
0x00580 – 0x005BC	Xilinx System Monitor
0x005C0 – 0x005FC	MCOR System Information
0x00600 – 0x0067C	512 Bytes Transceiver data
0x00680 – 0x006BC	Interrupt Registers
0x006C0 – 0x006FC	MCOR EVR Control Registers
0x01000– 0x017FE	EVR Register Interface from USB (No BAR)
0x01800 – 0x01FFF	EVR Data Buffer Memory from USB (No BAR)
0x40000 – 0x7FFFC	Waveform Memory

BAR 2 Memory Map – EVR Registers

0x0000 – 0xFFFF	ERV Interface Base

Configuration/Status Set/Reset Registers

Many of the MCOR functions are controlled by three registers within the block of registers.

Configuration/Status Register

This Register will contain the current value of the settings for the particular bits.

Set Register

Writing a ‘1’ to a bit in this register will SET the corresponding bit in the Configuration/Status register without affecting any other bits.

Reset Register

Writing a ‘1’ to a bit in the register will CLEAR the corresponding bit in the Configuration/Status register without affecting any other bits.

Front Panel LED’s

Red and Green Blinking fast – Xilinx Boot done, no PCIe activity

Red blinking fast – PCIe link is up

Green Blinking Slow – A PCIe read has been executed

Green on steady – At least one channel on the MCOR has been configured.

MCOR Channel Registers

Each channel has a set of Configuration and Data Registers.

Channel Base	
0x0000	Channel 0
0x0040	Channel 1
0x0080	Channel 2
0x00C0	Channel 3
0x0100	Channel 4
0x0140	Channel 5
0x0180	Channel 6
0x01C0	Channel 7
0x0200	Channel 8
0x0240	Channel 9
0x0280	Channel 10
0x02C0	Channel 11
0x0300	Channel 12
0x0340	Channel 13
0x0380	Channel 14
0x03C0	Channel 15

Channel Registers

Offset	Reg		
0x00	0	Set Point Requested. The Channel will not respond to setpoint commands unless the configure bit is set.	Int32 in uA
0x04	1	Current Set Point	
0x08	2	Monitor ADC Reading	
0x0C	3	Monitor Average ADC Reading	
0x10	4	Monitor Ripple Measurement	
0x14	5	FeedBack ADC Reading	
0x18	6	FeedBack Average ADC Reading	
0x1C	7	FeedBack Ripple Measurement	
0x20	8	Fullscale DAC SetPoint Current	
0x24	9	Fullscale Monitor ADC ReadBack Current	
0x28	A	Ramp Rate	Int32 uA/sec
0x2C	B	Samples per Average	UInt32
0x30	C	Configuration/Status Register	
0x34	D	Set Configuration Register	
0x38	E	Reset Configuration Register	
0x3C	F	Fullscale FeedBack ADC ReadBack Current	

Channel Configuration/Status Register

Bit			
6	Fault Status	‘1’ → MCOR Power Module Faulted ‘0’ → MCOR Power Module OK	
5	Ramping	‘1’ → Ramping in progress ‘0’ → Ramping done	
4	Standardized Direction	‘1’ → Falling ‘0’ → Rising	Not implemented
3	Ramp Mode	‘1’ → Immediate, no ramping ‘0’ → Normal Ramp Mode Selected	
2	Closed Loop (Auto Trim)	‘1’ → Closed Loop ‘0’ → Open Loop	Not implemented
1	Fast FeedBack	‘1’ → Channel being used for Fast Feedback ‘0’ → Normal MCOR functionality	Not implemented
0	Configured	Cleared by power on, set to indicate the channel has been configured. The Channel will not respond to setpoint commands unless the configure bit is set.	

Configuration Set/Reset Register

Bit		
4	Standardized Direction	Not implemented
3	Ramp/Immediate	
2	Closed Loop	Not implemented
1	Fast FeedBack	Not implemented
0	Configured	

Ramp Rate

Bit		
[31:00]	Ramp Rate	Int32 in µA/sec

Full Scale Current

The Fullscale value for the SetPoint DAC and the ReadBack ADC can be different depending on the type of MCOR installed. MCOR 6, 9 and 12 have a ReadBack of 10V is 12A. MCOR 1, 1.5 and 2 have a ReadBack of 10V is 2A. The DAC full-scale voltage is 10.2564, so the actual DAC full-scale is 2.564% over the nominal value below.

	Nominal Fullscale Current			
MCOR	DAC	Monitor ADC	FeedBack ADC	DAC Full Scale Setting
30	30A	30A	30A	30.7692A
20	20A	20A	20A	20.5128A
12	12A	12A	12A	12.30768A
9	9A	12A	9A	9.23076A
6	6A	12A	6A	6.15384A
2	2A	2A	2A	2.05128A
1.5	1.5A	2A	1.5A	1.53846A
1	1A	2A	1A	1.02564A

Full Scale DAC SetPoint Current

Bit		
[31:00]	Full Scale Current	Int32 in µA

Full Scale ADC ReadBack Current

Bit		
[31:00]	Full Scale Current	Int32 in µA

Samples/Average

Applies to both the Monitor and the FeedBack Channel

Bit		
[31:03]	unused	
[02:00]	Number of samples to average over	0x5 → 32 0x4 → 16 0x3 → 8 0x2 → 4 0x1 → 2 0x0 → 1

Bulk Supply

Bulk Supply Registers

~~Offset~~	~~Reg~~	~~Base 0x0400~~	
~~0x00~~	~~0~~	~~Bulk Voltage Request~~	~~Int32 in µV~~
0x04	1	Bulk Supply Voltage 0-5V => 0-30V	Int32 in µV
~~0x08~~	~~2~~	~~Bulk Supply Current~~	~~Int32 in µA~~
0x0C	3	Ground Fault Current	Int32 in µA
~~0x10~~	~~4~~	~~Bulk Current Limit Request~~	~~Int32 in µA~~
~~0x14~~	~~5~~	~~Bulk Supply DAC Full Scale Voltage~~	~~Int32 in µV~~
~~0x18~~	~~6~~	~~Bulk Supply DAC Full Scale Current~~	~~Int32 in µA~~
0x1C	7	Bulk Supply ADC Full Scale Voltage	Int32 in µV
~~0x20~~	~~8~~	~~Bulk Supply ADC Full Scale Current~~	~~Int32 in µA~~
0x24	9	Bulk Supply ADC Full Scale Ground Current	Int32 in µA
0x28	A	Ground Fault Current Threshold	Int32 in µV
~~0x2C~~	~~B~~	~~Ramp Rate~~	~~Int32 in µV/Sec~~
0x30	C	Configuration/Status Register	UInt32
0x34	D	Set Configuration Register	UInt32
0x38	E	Reset Configuration Register	UInt32
0x3C	F		

Bulk Supply Status/Configuration Register

Bit		
7	Configured	‘1’ → Bulk Supply Configured ‘0’ → Bulk Supply not Configured
6	Ground Fault	‘1’ → Bulk Supply Ground Fault ‘0’ → No Bulk Supply Ground Fault
~~5~~	~~Fault~~	~~‘1’ → Bulk Supply Fault~~ ~~‘0’ → No Bulk Supply Fault~~
~~4~~	~~Ramping~~	~~‘1’ → Ramping in progress~~ ~~‘0’ → Ramping done~~
~~3~~	~~Bulk On Status~~ ~~(Bulk PS_OK)~~	~~‘1’ → Bulk Supply is ON~~ ~~‘0’ → Bulk Supply is OFF~~
~~2~~	~~Control Type~~	~~‘1’ → Slave~~ ~~‘0’ → Master~~
~~1~~	~~PS Reset~~ ~~(Bulk SO)~~	~~‘1’ → Reset Asserted~~ ~~‘0’ → Reset Not Asserted~~
0	PS On/Off Request (Bulk ENA_In/Out)	‘1’ → Bulk Supply ON ‘0’ → Bulk Supply OFF

Bulk Supply Set/Reset Configuration Register

Bit		
~~4~~	~~Control Type~~	~~‘1’ →Slave~~ ~~‘0’ → Master~~
~~1~~	~~PS Reset~~	~~‘1’ → Assert Reset~~ ~~‘0’ → De-assert Reset~~
0	PS On/Off Request	‘1’ → Turn Bulk Supply ON ‘0’ → Turn Bulk Supply OFF

MCOR ADC Control

The ADC’s used for reading the MCOR has the option of running with oversampling. The following registers control the functionality of the ADC’s.

ADC Control Registers

Offset	Base = 0x0440	
0x00	Control	Uint32
0x04	Set Control	
0x08	Reset Control	
0x0C	MCOR ADC Oversampling	
0x10	MCOR ADC External Ref.	
0x14 0x3C		

Control Register

Bit		
[3]	Bulk ADC Timeout	‘1’ → Bulk ADC timed out ‘0’ → Bulk ADC OK
[2]	ADC Timeout	‘1’ → One of the ADC’s timed out ‘0’ → ADC’s OK
[1]	Bulk ADC Reset	‘1’ → Bulk ADC Reset Asserted ‘0’ → Bulk ADC Reset Not Asserted
[0]	MCOR ADC’s Reset	‘1’ → MCOR ADC Reset Asserted ‘0’ → MCOR ADC Reset Not Asserted

Control Set/Reset Register

Bit		
[3]	Reset Bulk ADC Timeout	
[2]	Reset ADC Timeout	
[1]	Bulk ADC Reset	
[0]	MCOR ADC’s Reset	

Oversampling Control

Bit		
[14:12]	Bulk Supply	See Analog Devices AD7609 for more details
[11:09]	Feedback ADC Channels 15 – 8	
[08:06]	Feedback ADC Channels 7 – 0	
[05:03]	Monitor ADC Channels 15 – 8	
[02:00]	Monitor ADC Channels 7 – 0	

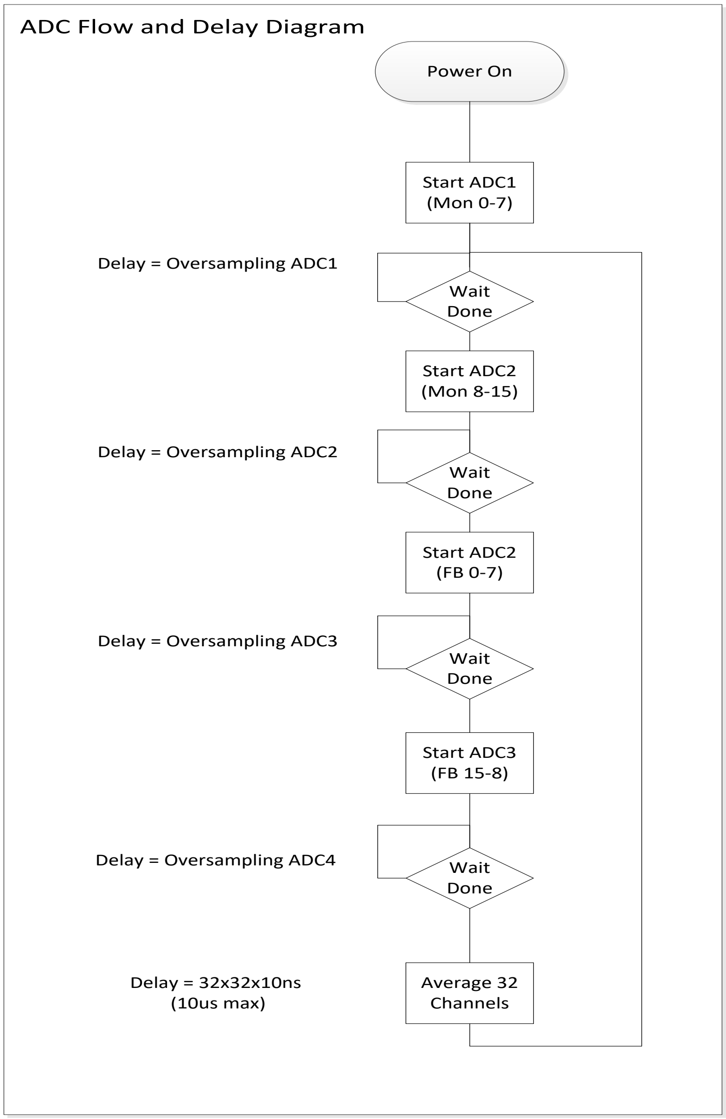
Increasing the oversampling rate will slow down the conversion rate of the ADC, this will affect the 100Khz normal operation of the ADC.

From Analog Devices AD7609 datasheet

OS [2:0]	OS Ratio	SNR ±5V Range (dB)	SNR ±10V Range (dB)	−3 dB BW 5V Range (kHz)	−3 dB BW 10V Range (kHz)
000	No OS	90.8	91.5	22	33
001	2	93.3	93.9	22	28.9
010	4	95.5	96.4	18.5	21.5
011	8	98	98.9	11.9	12
100	16	100.6	101	6	6
101	32	101.8	102	3	3
110	64	102.7	102.9	1.5	1.5
111	Invalid				

Internal Reference Control

Bit		
~~[04]~~	~~Bulk Supply~~	‘1’ → Internal Reference ‘0’ → External Reference See Analog Devices AD7609 for more details
[03]	Feedback ADC Channels 15 – 8	
[02]	Feedback ADC Channels 7 – 0	
[01]	Monitor ADC Channels 15 – 8	
[00]	Monitor ADC Channels 7 – 0	


ADC Flow and Delay Diagram
Power On
Start ADC1 (Mon 0-7)
Delay = Oversampling ADC1
Wait Done
Start ADC2 (Mon 8-15)
Delay = Oversampling ADC2
Wait Done
Start ADC2 (FB 0-7)
Delay = Oversampling ADC3
Wait Done
Start ADC3 (FB 15-8)
Delay = Oversampling ADC4
Wait Done
Delay = 32x32x10ns (10us max)
Average 32 Channels

MCOR Faults

Fault Registers

Offset	Base = 0x0480	
0x00	MCOR Power Module Fault Status	Uint32
0x04	MCOR Power Module Latched Fault Status	
0x08	Reset Latched Fault Status	
0x0C	Control	
0x10	Set Control	
0x14	Reset Control	
0x18	Fault ByPass	
0x1C	Set Fault Bypass	
0x20	Reset Fault Bypass	

The MCOR Faults are conditioned with a 2.55us filter.

Fault Status

Bit		
[15:00]	MCOR Power Module Fault Status	

Latched Fault Status

Bit		
[15:00]	MCOR Power Module Latched Fault Status	

Reset Latched Fault Status

Bit		
[15:00]	Reset MCOR Power Module Latched Fault Status	

Control Register

Bit		
[1]	MCOR Inhibit	‘1’ → MCOR Inhibit Asserted ‘0’ → MCOR Inhibit Not Asserted
[0]	MCOR Reset	‘1’ → MCOR Reset Asserted ‘0’ → MCOR Reset Not Asserted

Control Set/Reset Register

Bit		
[1]	MCOR Inhibit	
[0]	MCOR Reset	

Interlocks

For future use, 4 Interlock outputs and 8 Magnet Fault inputs and 1 Water Sum Fault are provided. The Magnet Faults are conditioned with a 2.55us filter.

Interlock and Magnet Fault Registers

Offset	Base = 0x0500	
0x00	External Interlocks Status	
0x04	Set External Interlocks	UInt32
0x08	Reset External Interlocks	
0x0C	Magnet Fault Status	
0x10	Magnet Latched Fault Status	
0x14	Reset Magnet Latched Fault Status	
0x10 0x0E		

External Interlock Status (0x0000)

Bit		
[31:04]	Unused	
[03:00]	Output	

External Interlock Set/Reset Output (0x0004,0x0008)

Bit		
[31:04]	Unused	
[03:00]	Output	

Input Status (0x000C)

Bit		
[31:09]		
8	Water Fault (Turn off Bulk)	
[07:00]	Inputs	

Latched Input Status (0x0010)

Bit		
[31:09]		
8	Water Fault (Turn off Bulk)	
[07:00]	Inputs	

Clear Latched Input Status (0x0010)

Bit		
[31:09]		
8	Water Fault (Turn off Bulk)	
[07:00]	Inputs	

MCOR Voltage Monitor Interface

Several Board voltages are monitored. They can be read back in the following registers.

Voltage Monitor Registers

Offset	Base = 0x0540	Int32
0x00	+15.0V(In)	$V = N * 6.0 * 305.18E^{-6}$
0x04	+12.0V(In)	$V = N * 4.83 * 305.18E^{-6}$
0x08	+5.0V(In)	$V = N * 2.0 * 305.18E^{-6}$
0x0C	+3.3V	$V = N * 305.18E^{-6}$
0x10	+3.3VCC IO	$V = N * 305.18E^{-6}$
0x14	-15.0V(In)	$V = N * -16.0 * 305.18E^{-6}$
0x18	+15.0V(In) Current	$I = N * 19.0735E^{-6}/60E^{-3}$
0x1C	+12.0V(In) Current	$I = N * 19.0735E^{-6}/150E^{-3}$
0x20	+5.0V(In) Current	$I = N * 19.0735E^{-6}/30E^{-3}$
0x24	+3.3V Current	$I = N * 19.0735E^{-6}/150E^{-3}$
0x28	+3.3VCCIO Current	$I = N * 19.0735E^{-6}/150E^{-3}$
0x2C	+2.5V Current	$I = N * 19.0735E^{-6}/150E^{-3}$
0x30	+1.0V Current	$I = N * 19.0735E^{-6}/150E^{-3}$
0x34	-15.0V(In) Current	$I = N * 19.0735E^{-6}/60E^{-3}$
0x38	Board Temperature	N* 0.0625
0x3C		

Xilinx System Monitor

Base Address for the Xilinx System Monitor is 0x00000580

The Xilinx contains an on chip monitor that, among other things, allows access to the chip temperature and V(Int) and V(Aux). The current value as well as the Min and Max are stored in the Xilinx and can be readout via the System Monitor Interface.

Read Xilinx Monitor

Offset	Base = 0x0580	
0x00	Current Temp	Int32 in ADC Counts
0x04	Current V(Int)	
0x08	Current V(Aux)	
0x0C	Max Temp	
0x10	Max V(Int)	
0x14	Max V(Aux)	
0x18	Min Temp	
0x1C	Min V(Int)	
0x20	Min V(Aux)	
0x24 0x3C		

$$Temperature\ (°\mathrm{C}) = (\frac{Counts}{64} * 503.975/1024) - 273.15$$

$$\mathrm{Volts}\ = \frac{counts}{64} / \ 1024 * 3$$

System Information

Offset	Base = 0x05C0	
0x00	Firmware Version	8 Bytes, ASCII = " 00000001"
0x08	System ID	4 Bytes, ASCII = "MCOR"
0x0C	Sub Type	10 Bytes, ASCII = " "
0x16	Firmware Date	10 Bytes, ASCII = " dd/mm/yyyy"
0x1F		

Beam Synchronous Acquisition (BSA) – NOT Implemented

Address		
	Time Stamp	
	⋮	
	Time Stamp	
	MCOR 0 Monitor Average ADC	
	⋮	
	MCOR 15 Monitor Average ADC	
	MCOR 0 FeedBack Average ADC	
	⋮	
	MCOR 15 FeedBack Average ADC	

Interrupts

The MCOR Controller could generate several interrupts.

1. Fault Detected
2. Waveform Acquisition complete
3. BSA Message available
4. EVR Interrupts
5. Ramping Done

Interrupt Registers

Offset	Base = 0x0680	
0x00	Interrupt Source	RW
0x04	Interrupt Source Enables	RO
0x08	Interrupt Source Set Enable	WO
0x0C	Interrupt Source Reset Enable	WO
0x10	Set Software Interrupt	WO

Interrupt Source

Bit			
[31:09]	Unused		
[08]	Software Interrupt		
[07]			
[06]	Ramp Done		Not Implemented
[05]	COMx GPI	From COMx GPIO	
[04]	BSA Message Available		Not Implemented
[03]	EVR Interrupt		
[02]	OR of the Magnet Faults	External Interlocks	Not Implemented
[01]	OR of the MCOR Channel Faults		Not Implemented
[00]	Waveform Acquisition complete		Not Implemented

Writing a ‘1’ to a bit in the Interrupt Source will reset the corresponding bit if the cause of the interrupt has been cleared. If the cause of the interrupt has not been cleared, the corresponding bit will not be reset and another interrupt will be generated.

Interrupt Source Enable, Set/Reset

Bit		
[31:01]	Unused	
[08]	Software Interrupt	
[07:01]	Not implemented yet	
[00]	End of Waveform Capture	

Software Interrupt register

Bit			
[31:09]	Unused		
[08]	Software Interrupt	Self Clearing	WO
[07:00]	unused		

Fiber Optic Transceiver data

The Transceivers have diagnostics built in to facilitate component monitoring, fault isolation and predictive failure (See AVAGO app note 5016). All 512 bytes of diagnostic data is available in the Xilinx memory starting at location 0x1000. The byte data from the transceiver is read as 32 bit word.

Offset	Base = 0x0600				
	SFP 0xA0	MSB			LSB
0x00		Byte 3	Byte 2	Byte 1	Byte 0
0x04		Byte 7	Byte 6	Byte 5	Byte 4
0x40	SFP 0XA2 Extended Data	Byte 3	Byte 2	Byte 1	Byte 0
0x44		Byte 7	Byte 6	Byte 5	Byte 4

EEPROM Serial ID Memory Contents –

The following tables are from the AVAGO datasheet. For detailed information see AVAGO Datasheet.

Serial ID data fields – Address 0xA0 (from SFF-8472 MSA).

BASE ID FIELDS			
Byte #	Size	Name Field	Description Field
0	1	Identifier	Type of serial transceiver
1	1	Ext. Identifier	Extended identifier of type of serial transceiver
2	1	Connector	Code for connector type
3-10	8	Transceiver	Code for electronic compatibility or optical compatibility
11	1	Encoding	Code for serial encoding algorithm
12	1	BR, Nominal	Nominal bit rate, units of 100 Mbits/sec.
13	1	Reserved	
14	1	Length (9µm) km	Link length supported for 9/125 Ïm fiber, units of km
15	1	Length (9µm)	Link length supported for 9/125 Ïm fiber, units of 100m
16	1	Length (50µm)	Link length supported for 50/125 Ïm fiber, units of 10m
17	1	Length (62.5µm)	Link length supported for 62.5/125 Ïm fiber, units of 10m
18	1	Length (Copper)	Link length supported for copper, units of meters
19	1	Reserved	
20-35	16	Vendor name	SFP vendor name (ASCII)
36	1	Reserved	
37-39	3	Vendor OUI	SRP vendor IEEE company ID
40-55	16	Vendor PN	Part number provided by SFP vendor (ASCII)
58-59	4	Vendor rev	Revision level for part number provided by vendor (ASCII)
60-61	2	Wavelength	Laser wavelength
62	1	Reserved	
63	1	CC_BASE	Check code for Base ID Fields (addresses 0 to 62)
EXTENDED ID FIELDS			
64-65	2	Options	Indicates which optional transceiver signals are implemented
66	1	BR, max	Upper bit rate margin, units of %
67	1	BR, min	Lower bit rate margin, units of %
68-83	16	Vendor SN	Serial number provided by vendor (ASCII)
84-91	8	Date Code	Vendor's manufacturing date code
92	1	Diagnostic Monitoring Type	Indicates which type of diagnostic monitoring is implemented (if any) in the transceiver
93	1	Enhanced Options	Indicates which optional enhanced features are implemented (if any) in the transceiver
94	1	SFF-8472 Compliance	Indicates which revision of SFF8472 the transceiver complies with
95	1	CC_EXT	Check code for the Extended ID Fields addresses 64 to94
VENDOR SPECIFIC ID FIELDS			
96-127	32	Vendor Specific	Vendor Specific EEPROM
128-255	128	Reserved	Reserved for future use

Enhanced Feature Set Memory (Address A2h)

Byte #		Byte #	
1	Temp H Alarm LSB[1]	34	Rx Pwr L Alarm MSB[5]
2	Temp L Alarm MSB[1]	35	Rx Pwr L Alarm LSB[5]
3	Temp L Alarm LSB[1]	36	Rx Pwr H Warning MSB[5]
4	Temp H Warning MSB[1]	37	Rx Pwr H Warning LSB[5]
5	Temp H Warning LSB[1]	38	Rx Pwr L Warning MSB[5]
6	Temp L Warning MSB[1]	39	Rx Pwr L Warning LSB[5]
7	Temp L Warning LSB[1]	40-55	Reserved
8	Vcc H Alarm MSB[2]	56-94	External Calibration Constants[6]
9	Vcc H Alarm LSB[2]	95	Checksum for Bytes 0-94[7]
10	Vcc L Alarm MSB[2]	96	Real Time Temperature MSB[1]
11	Vcc L Alarm LSB[2]	97	Real Time Temperature LSB[1]
12	Vcc H Warning MSB[2]	98	Real Time Vcc MSB[2]
13	Vcc H Warning LSB[2]	99	Real Time Vcc LSB[2]
14	Vcc L Warning MSB[2]	100	Real Time Tx Bias MSB[3]
15	Vcc L Warning LSB[2]	101	Real Time Tx Bias LSB[3]
16	Tx Bias H Alarm MSB[3]	102	Real Time Tx Power MSB[4]
17	Tx Bias H Alarm LSB[3]	103	Real Time Tx Power LSB[4]
18	Tx Bias L Alarm MSB[3]	104	Real Time Rx average MSB[5]
19	Tx Bias L Alarm LSB[3]	105	Real Time Rx average LSB[5]
20	Tx Bias H Warning MSB[3]	106	Reserved
21	Tx Bias H Warning LSB[3]	107	Reserved
22	Tx Bias L Warning MSB[3]	110	Status/Control
23	Tx Bias L Warning LSB[3]	111	Reserved
24	Tx Pwr H Alarm MSB[4]	112	Flag Bits
25	Tx Pwr H Alarm LSB[4]	113	Flag Bits
26	Tx Pwr L Alarm MSB[4]	114	Reserved
27	Tx Pwr L Alarm LSB[4]	115	Reserved
28	Tx Pwr H Warning MSB[4]	116	Flag Bits
29	Tx Pwr H Warning LSB[4]	117	Flag Bits
30	Tx Pwr L Warning MSB[4]	118-127	Reserved
31	Tx Pwr L Warning LSB[4]	128-247	Customer Writeable
32	Rx Pwr H Alarm MSB[5]	248-255	Vendor Specific
33	Rx Pwr H Alarm LSB[5]		

Notes:

1. Temperature (Temp) is decoded as a 16 bit signed two's compliment integer in increments of 1/256°C.
2. Supply Voltage (Vcc) is decoded as a 16 bit unsigned integer in increments of 100 µV.
3. Laser bias current (Tx Bias) is decoded as a 16 bit unsigned integer in increments of 2 µA.
4. Transmitted average optical power (Tx Pwr) is decoded as a 16 bit unsigned integer in increments of 0.1 µW.
5. Received average optical power (Rx Pwr) is decoded as a 16 bit unsigned integer in increments of 0.1 µW.
6. Bytes 55-94 are not intended for use with AFCT-57R5APZ, but have been set to default values per SFF-8472.
7. Byte 95 is a checksum calculated (per SFF-8472) and stored prior to product shipment.

MCOR Timing Receiver

Timing Control Registers

Address	Register	Description
0x0000	Status	Status Register
0x0004	Control	Control Register
0x0008	IrqFlag	Interrupt Flag Register
0x000C	IrqEnable	Interrupt Enable Register
0x0010	PulseIrqMap	Pulse Interrupt Map Register
0x0014 – 0x001C	Unused	
0x0020	DataBufCtrl	Rx Data Buffer Control and Status
0x0024[1]	Reserved	(Tx Data Buffer Control and Status)
0x0028	Unused	
0x002C[2]	FW Version	Firmware Version
0x0030 – 0x003C	Unused	
0x0040	EvntCntPresc	Event Counter Prescaler
0x0044 – 0x0048	Unused	
0x004C	µsec Divider	Divider from Event Clock to 1Mhz
0x0050[1]	Reserved	[Clock Control Register]
0x0054 – 0x0058	Unused	
0x005C	SecSReg	Seconds Register
0x0060	SecCntr	TimeStamp Seconds Counter
0x0064	EvntCntr	TimeStamp Event Counter
0x0068	SecLatch	TimeStamp Seconds Latch
0x006C	EvntCntrLatch	TimeStamp Event Counter
0x0070	EvntFIFOSec	Event FIFO Seconds Register
0x0074	EvntFIFOEvntCntr	Event FIFO Event Counter Register
0x0078	EvntFIFOEvntCode	Event FIFO Event Code Register
0x007C[1]	Reserved	(Event Log Status Register)
0x0080[1]	Reserved	(FracDiv)
0x0084	Unused	
0x0088[1]	Reserved	(RxInitPs)
0x008C	Unused	
0x0090[1]	Reserved	(GPIO DIR)
0x0094[1]	Reserved	(GPIO IN)
0x0098[1]	Reserved	(GPIO OUT)

[1] Not Used for MCOR

[2] SLAC MCOR Specific Definition

Prescaler Control Registers

0x0100	Presc0	Prescaler 0 Divider
0x0104	Presc1	Prescaler 1 Divider
0x0108	Presc2	Prescaler 2 Divider

Pulse Generator Control Registers

0x0200	PulseGen0	Pulse Generator 0 Control Registers
0x0204		Pulse Generator 0 Prescaler Registers
0x0208		Pulse Generator 0 Delay Registers
0x020C		Pulse Generator 0 Width Registers
0x0210 – 0x021C	PulseGen1	Pulse Generator 1 Registers
0x0220 – 0x022C	PulseGen2	Pulse Generator 2 Registers
0x0230 – 0x023C	PulseGen3	Pulse Generator 3 Registers
0x0240 – 0x024C	PulseGen4	Pulse Generator 4 Registers
0x0250 – 0x025C	PulseGen5	Pulse Generator 5 Registers
0x0260 – 0x026C	PulseGen6	Pulse Generator 6 Registers
0x0270 – 0x027C	PulseGen7	Pulse Generator 7 Registers
0x0280 – 0x028C	PulseGen8	Pulse Generator 8 Registers
0x0290 – 0x029C	PulseGen9	Pulse Generator 9 Registers

Front Panel Mapping Registers

0x0400	FrntPnlMap	Front Panel Output 0 and 1 Map Registers

Data Buffer

0x0800 – 0x0FFF	DataBufMem	Data Buffer Memory

Event Log

0x2000 – 0x2FFF	EvntLog	Event Log

Map Ram

0x4000 – 0x5FFF	MapRam1	Event Mapping RAM 1
0x6000 – 0x7FFF	MapRam2	Event Mapping RAM 2

Register Definitions

Status Register

Bit	Address 0x0000	
[31:24]	DBUS[07:00]	DBUS
23	Event FIFO Full	
22	Event FIFO Empty	
[21:17]	Unused	
16	LEGVIO	Legacy VIO
[15:06]	Unused	
5	FIFOStp	Event FIFO Stopped Flag
[04:00]	Unused	

Control Register

Bit	Address 0x0004		
31	EVREn	Event Receiver Enable	
30[1]	Reserved	(EvntFWD)	
29[1]	Reserved	(TXLP)	
28[1]	Reserved	(RXLP)	
[27:15]	Unused		
14	TSDBUS	Use DBU4 for TimeStamp Counter	
13	RSTS	Reset TimeStamp	'1' → Reset TimeStamp Counter Self clearing
[12:11]	Unused		
10	LTS	Latch TimeStamp	'1' → Latch TimeStamp Counter Self clearing
9	MAPEn	Event Mapping RAM Enable	
8	MAPRS	Mapping RAM Select	'1' → Select Mapping RAM2 '0' → Select Mapping RAM1
7[1]	Reserved	(LOGRS)	Reset Event Log
6[1]	Reserved	(LOGEN)	Enable Event Log
5[1]	Reserved	(LOGDIS)	Disable Event Log
4[1]	Reserved	LOGSE	Log Stop Event
3	RSFIFO	Reset Event FIFO	'1' → Reset Event FIFO Self clearing
[02:00]	Unused		

[1] Not Used for SLAC MCOR

Interrupt Flag Register

Bit	Address 0x0008		
[31:6]	Unused		
5	IFDBUF	Data Buffer Flag	
4	IFHW	Hardware Flag	See Hardware Interrupt Mapping Register
3	IFEV	Event Flag	Event FIFO not Empty
2	IFHB	HeartBeat FLAG	
1	IFFF	Event FIFO Full Flag	
0	IFVIO	Receiver Violation Flag	

The interrupt flag is normally cleared by writing a ‘1’ to the corresponding bit. However, the Event Flag is the Event FIFO Not Empty Flag. It is cleared when the FIFO has been read out.

Interrupt Enable Register

Bit	Address 0x000C		
31	IRQEN	Master Interrupt Enable	
[30:6]	Unused		
5	IEDBUF	Data Buffer Interrupt Enable	
4	IEHW	Hardware Interrupt Enable	See Hardware Interrupt Mapping Register
3	IEEV	Event Interrupt Enable	
2	IEHB	HeartBeat Interrupt Enable	
1	IEFF	Event FIFO Full Interrupt Enable	
0	IEVIO	Receiver Violation Interrupt Enable	

Hardware Interrupt Mapping Register

Bit	Address 0x0010		
[31:6]	Unused		
[05:00]	See Selection Control Word		

Receive Data Buffer Control and Status Register

Bit	Address 0x0020		
[31:16]	Unused		
15	DBRX/DBENA		Writing a ‘1’ Enables Data Buffer Recorder for a Single Event. Clears CheckSum Error. Reads ‘0’ when Done.
14	DBRDY/DBDIS		
13	DBCS	Data Buffer Checksum Error	Cleared by writing ‘1’ to DBENA or DBDIS or disabling the Data Buffer
12	DBEN	Data Buffer Enable	‘1’ → Distributed BUS Multiplexed with Data Buffer ‘0’ → Distributed BUS NOT Multiplexed with Data Buffer
11			‘0’
[10:00]	RX Size		

FPGA Firmware Version Register

Bit	Address 0x002C		
[31:28]	EVR		0x01
[27:24][1]	Form Factor		0x0F → SLAC MCOR …. 0x02 → VME64x 0x01 → PMC 0x00 → Compact PCI
[23:08]	Reserved		
[07:00]	Version ID	Version	

[1] SLAC Added, NOT standard

Event Counter Clock Prescaler Register

Bit	Address 0x0040		
[31:16]	Unused		
[15:00]	TSEvntCntrPresc	TimeStamp Event Counter Prescaler	

Microsecond Divider Register

Bit	Address 0x004C		
[31:16]	Unused		
[15:00]	µSDivider	Value to get 1µs timer from System Clock	For 119Mhz clock should be 119

Seconds Shift Register

Bit	Address 0x005C

[31:00]	SecSR	Seconds Shift Register	

Seconds Counter Register

Bit	Address 0x0060		
[31:00]	SecCntr	Seconds Counter Register	

TimeStamp Event Counter Register

Bit	Address 0x0064		
[31:00]	TsEvntCntr	TimeStamp Event Counter Register	

Seconds Latch Register

Bit	Address 0x0068		
[31:00]	SecLatch	Seconds Latch Register	

TimeStamp Event Latch Register

Bit	Address 0x006C		
[31:00]	TsEvntLatch	TimeStamp Event Latch Register	

Event FIFO Seconds Register

Bit	Address 0x0070		
[31:00]	FIFOSec	FIFO Seconds	Only valid after reading the corresponding Event Code

Event FIFO TimeStamp Register

Bit	Address 0x0074		
[31:00]	FIFOTs	FIFO TimeStamp	Only valid after reading the corresponding Event Code

Event FIFO Event Code Register

Bit	Address 0x0078		
[31:08]	Unused		
[07:00]	FIFOEvntCode	FIFO Event Code	Reading the Event Code also moves the Seconds and TimeStamp to the corresponding register to be read out

Prescaler Registers

Offset	Base Address 0x0100
0x00	Prescale 0
0x04	Prescale 1
0x08	Prescale 2

Prescaler Register

Bit			
[31:00]	PrescVal		

Pulse Generator Registers

Offset	Base Address 0x0200
0x00	Pulse Generator 0 Control Register
0x04	Pulse Generator 0 Prescaler Register Divides the Delay and Width clock by N+1
0x08	Pulse Generator 0 Delay Register
0x0C	Pulse Generator 0 Width Register
0x10 – 0x1C	Pulse Generator 1 Registers
0x20 – 0x2C	Pulse Generator 2 Registers
0x30 – 0x3C	Pulse Generator 3 Registers
0x40 – 0x4C	Pulse Generator 4 Registers
0x50 – 0x5C	Pulse Generator 5 Registers
0x60 – 0x6C	Pulse Generator 6 Registers
0x70 – 0x7C	Pulse Generator 7 Registers
0x80 – 0x8C	Pulse Generator 8 Registers
0x90 – 0x9C	Pulse Generator 9 Registers

Pulse Generator (x) Control Register

Bit			
[31:08]	Unused		
08[1]	P(x)_n	Re-Triggerable	'1' → Generator is NOT Retriggerable '0' → Generator is Retriggerable
07	P(x)Out	Pulse Generator Output	
06	P(x)SWS	Software Set	Terminates any previous Pulse Request
05	P(x)SWC	Software Reset	Terminates any previous Pulse Request
04	P(x)POL	Polarity Control	'1' → Invert Polarity (Low True) '0' → Normal Polarity (High True)
03	P(x)MRE	Event Map Reset Enable	
02	P(x)MSE	Event Map Set Enable	
01	P(x)MTE	Event Map Trigger Enable	
00	P(x)ENA	Pulse Enable	

[1] SLAC Added Control BIT, NOT standard

Pulse Generator Retriggerability

If P(x)RTEN is a '0' the Pulse Generator is Retriggerable. If a new Trigger Event is received while the Pulse is TRUE, the width will be stretched by programmed width. If a new Trigger Event is received while in the Delay period, it will be ignored.

Pulse Generator Prescaler Register

The value in the Prescaler Register is used to divide the clock going to the Pulse Generator Delay and Width by N+1;

Timing Output Mapping

Offset	Base Address 0x0400	
0x00	FPOutMap	Front Panel Mapping control See Selection Control Word
0x04 – 0x0F	Reserved	
0x10	Timing WFM Trigger	

Front Panel Output Mapping

Bit			
[31:22]	Unused		
[21:16]	FPMAP1	Front Panel Output 1 Mapping	See Selection Control Word
[15:06]	Unused		
[05:00]	FPMAP0	Front Panel Output 0 Mapping	See Selection Control Word

Timing WFM Trigger

Bit			
[31:06]	Unused		
[05:00]	WFMTrig	WaveForm Trigger	See Selection Control Word

Selection Control word

Value	EVR Trigger
63	Force Output High
62	Force Output Low
[61:43]	Reserved
42	Prescaler 2
41	Prescaler 1
40	Prescaler 0
[39:32]	Distributed Bus Bit [07:00]
[31:10]	Reserved
[09:00]	Pulse Generator Output [09:00]

Data Buffer (SLAC Specific)

Offset	Base Address 0x0800
0x0000	Type and Version
0x0004 – 0x0010	PNET Modifier
0x0014 – 0x0018	EVR Extension
0x001C – 0x0020	EPICS TimeStamp
0x0024 – 0x0030	BSA Information

Type and Version

Offset	Bits	
0x000	[31:16]	Type
	[15:00]	Version

EPICS TimeStamp

Offset	Bits	
0x000	[31:00]	Seconds since 1990
0x004	[31:17]	*131072 nSeconds
	[16:00]	Pulse ID

Event RAM

Offset	Event Code	Base Address 0x4000
0x0000	0x00	Event Map Reset
0x0004		Event Map Set
0x0008		Event Map Trigger
0x000C		Functions
0x0010	0x01	Event Map Reset
0x0014		Event Map Set
0x0018		Event Map Trigger
0x001C		Functions
-----	-----	-----
0x0FF0	0xFF	Event Map Reset
0x0FF4		Event Map Set
0x0FF8		Event Map Trigger
0x0FFC		Functions

Event MAP Reset

Bit		
[31:10]	Unused	
[09:00]	Reset Pulse Generator	If Enabled, Terminates any previous Pulse Request

Event MAP Set

Bit		
[31:10]	Unused	
[09:00]	Set Pulse Generator	If Enabled, Terminates any previous Pulse Request

Event MAP Trigger

Bit		
[31:10]	Unused	
[09:00]	Trigger Pulse Generator	

Event MAP Functions

Bit			Event Code
31	Save Event in FIFO		
30	Latch TimeStamp		
29	Reserved	(LED Event)	
28	Reserved	(Forward Event from RX to TX)	
27	Reserved	(Stop Event Log)	
26	Reserved	(Log Event)	
[25:06]	Reserved		
5	HeartBeat		0x7A
4	Reset Prescalers		0x7B
3	TimeStamp Reset Event		0x7D
2	TimeStamp Clock Event		0x7C
1	Seconds Shift Register ‘1’		0x71
0	Seconds Shift Register ‘0’		0x70

MCOR Voltage and Current Monitoring ADC

Three Linear Technology LTC2991, Octal I^{2}C Voltage, Current and Temperature Monitor chips are used to read the MCOR Monitor voltage and Currents. They are configured as follows:

Address “000”

Register		Value
0x00	Status Low	
0x01	Channel Enable/Trigger	0xF8
0x06	V1-4 Control	0xAA
0x07	V5-8 Control	0xAA
0x08	Acquisition Mode	0x18
0x0D-0x0C	V1-V2	+3.3V Current
0x11-0x10	V3-V4	+2.5V Current
0x15-0x14	V5-V6	+1.0V Current
0x19-0x18	V7-V8	+5.0V(In) Current

Register 1

Bit			Value
7	Enable V7/V8	+5.0V(In) Current	1
6	Enable V5/V6	+1.0V Current	1
5	Enable V3/V4	+2.5V Current	1
4	Enable V1/V2	+3.3V Current	1
3	Enable Temp and VCC		1
2	Busy	Read Only	0
1	Tempature(Internal) Valid	Read Only	0
0	VCC Valid	Read Only	0
			0xF8

Register 6

Bit			Value
7	Enable Filter		1
6	Kelvin/Celsius		0
5	Temperature/Volts		0
4	V3-V4	+2.5V Current	1
3	Enable Filter		1
2	Kelvin/Celsius		0
1	Temperature/Volts		0
0	V1-V2	+3.3V Current	1
			0x99

Register 7

Bit			Value
7	Enable Filter		1
6	Kelvin/Celsius		0
5	Temperature/Volts		0
4	V7-V8	+5.0V(In) Current	1
3	Enable Filter		1
2	Kelvin/Celsius		0
1	Temperature/Volts		0
0	V5-V6	+1.0V Current	1
			0x99

Register 8

Bit		Value
7	PWM[0]	0
6	PWM Invert	0
5	PWM Enable	0
4	Repeated Acquisition Mode	1
3	Temperature(Internal) Filter	1
2	Kelvin/Celsius	0
1	Reserved	0
0	Reserved	0
		0x18

Address “001”

Register		Value	
0x00	Status Low		
0x01	Channel Enable/Trigger	0xF8	
0x06	V1-4 Control	0xAA	
0x07	V5-8 Control	0xAA	
0x08	Acquisition Mode	0x18	
0x0D-0xC	V1-V2	+15.0V(In) Current	
0x11-0x10	V3-V4	-15.0V(In) Current	
0x15-0x14	V5-V6	+3.3VCCIO Current	
0x19-0x18	V7-V8	+12.0V(In) Current	

Register 1 – Channel Enable

Bit			Value
7	Enable V7/V8	+12.0V(In) Current	1
6	Enable V5/V6	+3.3VCCIO Current	1
5	Enable V3/V4	-15.0V(In) Current	1
4	Enable V1/V2	+15.0V(In) Current	1
3	Enable Temp and VCC		1
2	Busy	Read Only	0
1	Temperature(Internal) Valid	Read Only	0
0	VCC Valid	Read Only	0
			0xF8

Register 6 – V1-V4 Control

Bit			Value
7	Enable Filter		1
6	Kelvin/Celsius		0
5	Temperature/Volts		0
4	V3-V4	-15.0V(In) Current	1
3	Enable Filter		1
2	Kelvin/Celsius		0
1	Temperature/Volts		0
0	V1-V2	+15.0V(In) Current	1
			0xAA

Register 7 – V5-V7 Control

Bit			Value
7	Enable Filter		1
6	Kelvin/Celsius		0
5	Temperature/Volts		0
4	V7-V8	+12.0V(In) Current	1
3	Enable Filter		1
2	Kelvin/Celsius		0
1	Temperature/Volts		0
0	V5-V6	+3.3VCCIO Current	1
			0xAA

Register 8 – Temperature Control

Bit		Value
7	PWM[0]	0
6	PWM Invert	0
5	PWM Enable	0
4	Repeated Acquisition Mode	1
3	Temperature(Internal) Filter	1
2	Kelvin/Celsius	0
1	Reserved	0
0	Reserved	0
		0x18

Address “010”

Register		Value	
0x00	Status Low		
0x01	Channel Enable/Trigger	0x78	
0x06	V1-4 Control	0x88	
0x07	V5-8 Control	0x08	
0x08	Acquisition Mode	0x18	
0x0B-0x0A	V1	+3.3V	
0x0D-0x0C	V2	+3.3VCCIO	
0x0F-0x0E	V3	+5.0V(In)	
0x11-0x10	V4	+15.0V(In)	
0x13-0x12	V5	-15.0V(In)	
0x15-0x14	V6	+12.0V(In)	

Register 1

Bit			Value
7	Enable V7/V8	Unused	0
6	Enable V5/V6	-15.0V(In) , +12.0V(In)	1
5	Enable V3/V4	+5.0V(In) , +15.0V(In)	1
4	Enable V1/V2	+3.3V, +3.3VCCIO	1
3	Enable Temp and VCC		1
2	Busy	Read Only	0
1	Temperature(Internal) Valid	Read Only	0
0	VCC Valid	Read Only	0
			0x78

Register 6

Bit			Value
7	Enable Filter		1
6	Kelvin/Celsius		0
5	Temperature/Volts		0
4	V3 and V4	+5.0V(In) , +15.0V(In)	0
3	Enable Filter		1
2	Kelvin/Celsius		0
1	Temperature/Volts		0
0	V1 and V2	+3.3V, +3.3VCCIO	0
			0x88

Register 7

Bit			Value
7	Enable Filter		0
6	Kelvin/Celsius		0
5	Temperature/Volts		0
4	V7 and V8	Unused	0
3	Enable Filter		1
2	Kelvin/Celsius		0
1	Temperature/Volts		0
0	V5 and V6	-15.0V(In) , +12.0V(In)	0
			0x08

Register 8

Bit		Value
7	PWM[0]	0
6	PWM Invert	0
5	PWM Enable	0
4	Repeated Acquisition Mode	1
3	Temperature(Internal) Filter	1
2	Kelvin/Celsius	0
1	Reserved	0
0	Reserved	0
		0x18

USB Interface Protocol

The USB protocol consists of sending a header to define a read or write operation and 16/32 bit data. The Header is 8 Bytes with the following:

USB Request Header

Byte			
[00]	Control Byte		
[03:01]	24 Bit Address		
[05:04]	16 Bit Byte Count		
[07:06]	Dummy (Word Alignment)	"0000"	

Control Byte

Bit		
[07]	Response Flag	'1' → Response from Board '0' → Request from USB
[06]	Direction	'1' → Read from Board '0' → Write to Board
[05]	Size	'1' → 32 Bit Data
[04]	"0"	
[03:00]	TAG	User definable. Returned in the response message

Every USB transfer has a response message. For a write the response is the control byte followed by a zero. For a read, the response is the control byte, and a zero, followed by N-bytes of data.

USB Response Header

Byte			
[00]	Control Byte		
[01]	"0"		

USB Write Request

Byte			
[00]	Control Byte		
[03:01]	24 Bit Address		
[05:04]	16 Bit Byte Count		
[07:06]	Dummy (Word Alignment)	"0000"	
	N-Words of data		

PCIe Testing

BARA

```
/afs/slac/u/qb/strauman/bara -h
```

gives you usage info. The basic operation is

```
bara -s <signature> -r <resource>  -w <width>  offset [value]
```

```
signature: PCI signature  domain:bus:device.function, e.g., 0000:04:01.0
           (note that 'domain' must be 4 digits, 'bus' and 'dev' two
           and 'function' a single digit)

width:     access width (in bytes: 1,2 or 4)
resource:  bar number e.g., 2 (can be negative in which case the PCI config
space is accessed)
offset:    byte offset into BAR
value:     if given, then a 'write' (and readback) is performed, otherwise a
'read'
```

ssh laci@eioc-b34-mg05

cd /afs/slac/g/lcls/epics/modules/R3-14-12/drvPciMcor/MAIN_TRUNK/
cd /afs/slac1/g/lcls/epics/modules/R3-14-12/drvPciMcor/MAIN_TRUNK/

bin/linuxRT-x86/pciMcorDemo   bin/linuxRT-x86/demo.cexp

starts cexp. The script defines the base address of the MCOR as

'theMcor'

int md(unsigned address, int count, int size)

dumps 'count' items of 'size' starting at 'address'. Size maybe 1, 2 or 4 (bytes).
If not specified, 4 is assumed. To dump 52 (32-bit) words at offset xyz you say

md(theMcor + xyz, 52, 4)

Dump MCOR ID

```
md (theMcor + 0x5c0, 32, 1)
```

Dump Timging ID

```
md (theMcor + 0x102C, 4, 4)
```

Write Data Buffer Conrol, enable and trigger

```
*(long *)(theMcor + 0x1020) = 0x9000
```

Dump Data Buffer

```
md (theMcor + 0x1800, 52,1)
```