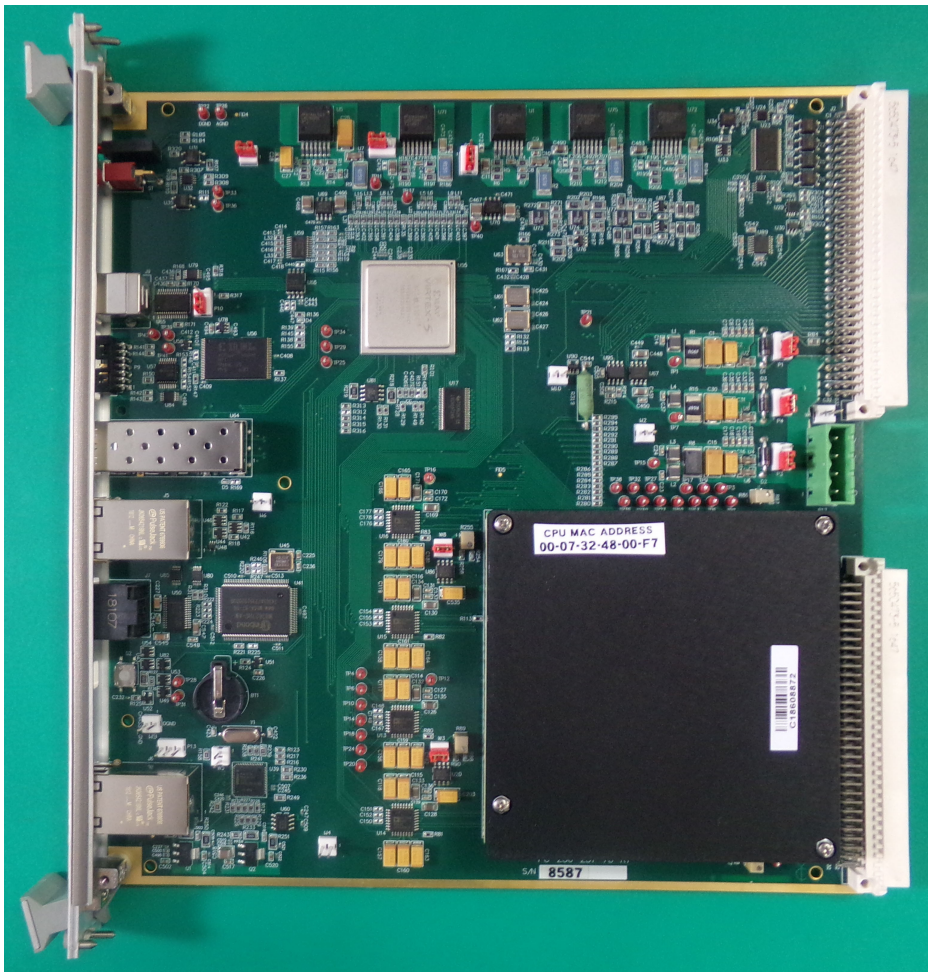
User's Manual

Ethernet MCOR (EMCOR) Controller Module



Ethernet MCOR Controller Module User Instructions

This document contains instructions on the use of the EMCOR (Ethernet MCOR Controller Module).

Table of Contents **Page**

Setup Guide

The EMCOR module is properly configured for use before shipment. The following jumper configuration guide is for reference only and shows how the module should be configured for normal use. All testing is performed with the jumpers in the following configuration. Table 1 shows the signal names and jumper configurations. Figure 1 shows the actual location of the jumpers on the module. The table below does not reference the jumper on P9, which is installed to ensure a predictable CPU boot-up process.

The EMCOR module comes equipped with a CPU module attached (non SLAC customers only). The CPU must go through a boot up process each time power is applied to the crate before the EMCOR module can be used to control MCOR modules, unless the USB Diagnostic Application is being used. Instructions for the CPU boot up process, which loads the OS into the CPU module, can be found later in this document.
Note that due to the amount of heat generated by the CPU module, it is necessary to always make certain that a blower assembly is installed to the crate and that air flow is constantly available across the EMCOR and CPU module as a means to remove heat being generated.

Table 1- *Table of jumper connections on the EMCOR module*

Header/Jumper	**Connection Pins**	**Signal 1**	**Signal 2**	**Signal 3**	**Use**
P1	1 to 2	+15V_VIN_Crate	+15V_V_IN		+15V Power
P2	2 to 3		+3.3V	+5V_IN	+3.3V Power
P3	1 to 2	+5V_IN_CRATE	+5_VIN		+5V Power
P4	1 to 2	-15V_IN_CRATE	-15V_IN		-15V Power
P10	2 to 3		VCCIO	+3.3VCCIO	USB Output VCC
P11	2 to 3		+3.3VCCIO	+5V_IN	+3.3VCCIO Power
W3	In	+5V_REF1	+5V_REF1		DAC Ref.
W7	In	DGND	AGND		DGND to AGND
W8	In	+5V_REF2	+5V_REF2		DAC Ref.

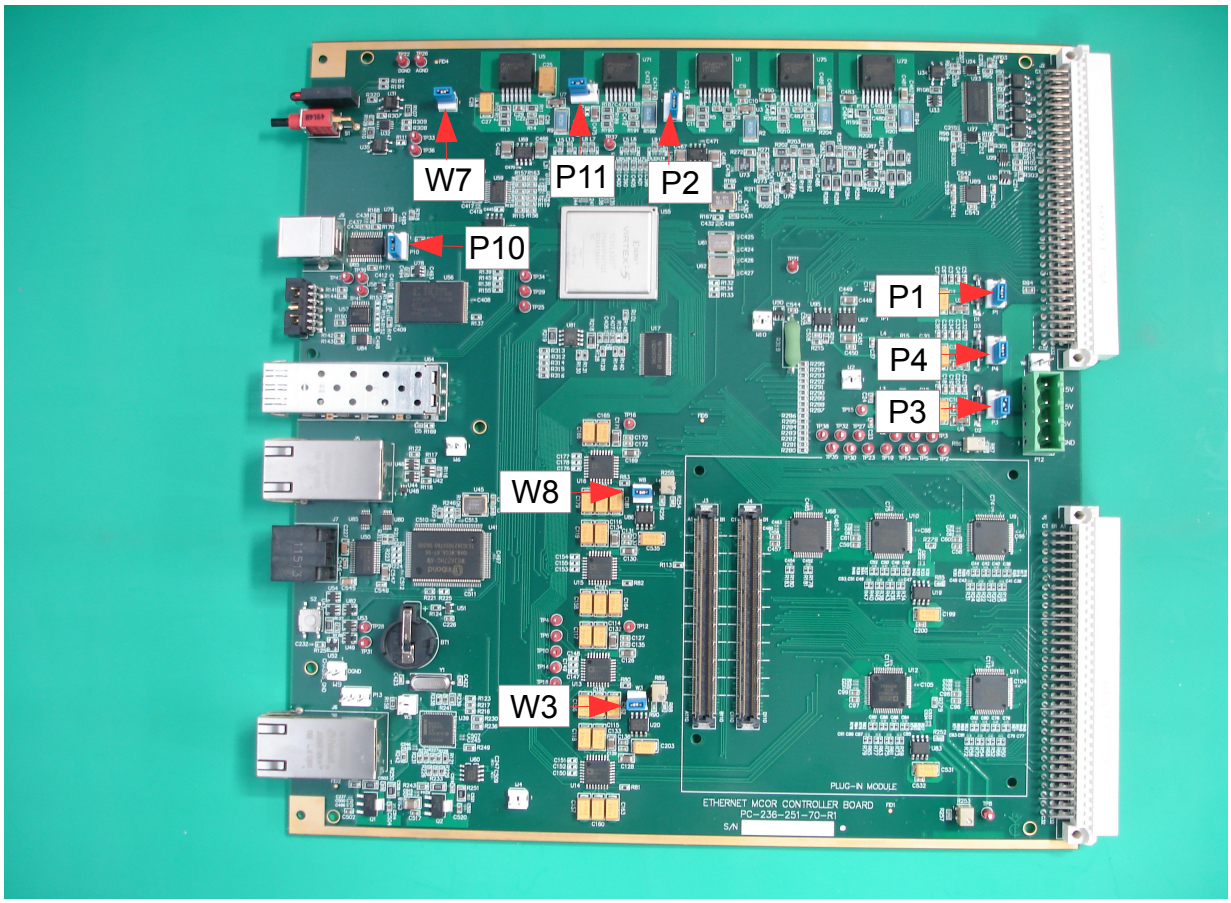


Figure 1--Jumper locations

EMCOR Controller Module Description

The EMCOR module was designed to replace the Interface Controller module that resides in slot 00 of the crate power supply. The EMCOR module will only fit in slot 00 since that is the only location in the crate with the 2 connectors (P1 and P2) necessary to mate with the EMCOR module. The EMCOR module will receive the required power needed from the crate power supply via crate connection P1 and EMCOR module connection J2. The EMCOR module will control up to 16 MCOR modules using a GUI based software such as EPICS and the EMCOR front panel EPICS CA port. It is also possible to communicate with the EMCOR module using an Ethernet connection and software such as UDPwin

(again using the front panel EPICS CA port) or via a serial connection using the EMCOR COM front panel port, as well as a USB connection (J9). Note that all of the signal connections for the EMCOR module J1 are identical to the Interface Controller module J2 connector, but many of the EMCOR J2 connections are not the same as the Interface Control module J1 connections. The signal names for the EMCOR connections (J1 and J2) are shown in table 2 below.

Caution: Crate connections J3 and J4 on the back of the crate were designed for use with the C4 Interface Controller, and **NOT** the EMCOR Controller. Do not make connections to either J3 or J4 if you are going to be using the EMCOR controller, as damage to the controller or external connections could occur. Table 3 has been created as a means to map the EMCOR J2 connections to the crate J3 and J4 connections.

Table 2 *EMCOR J1 and J2 Signal Connections*

J2	Signal	J2	Signal	J2	Signal	J1	Signal	J1	Signal	J1	Signal
A1	+15V_IN_AUX	B1	DGND	C1	INTLK1+	**A1**	GND_0	**B1**	GND_4	**C1**	GND_8
A2	+15V_IN_AUX	B2	DGND	C2	INTLK1-	**A2**	+REF_0	**B2**	+REF_4	**C2**	+REF_8
A3	+5V_IN_AUX	B3	-15V_IN_AUX	C3	INTLK2+	**A3**	AGND	**B3**	AGND	**C3**	AGND
A4	+5V_IN_AUX	B4	MAG_FAULT0	C4	INTLK2-	**A4**	MON_0	**B4**	MON_4	**C4**	MON_8
A5	DGND	B5	MAG_FAULT1	C5	INTLK3+	**A5**	MCOR_FAULT0	**B5**	MCOR_FAULT4	**C5**	MCOR_FAULT8
A6	DGND	B6	MAG_FAULT2	C6	INTLK3-	**A6**	FDBK_0	**B6**	FDBK_4	**C6**	FDBK_8
A7	-15V_IN_AUX	B7	MAG_FAULT3	C7	INTLK4+	**A7**	GND_1	**B7**	GND_5	**C7**	GND_9
A8	-15V_IN_AUX	B8	MAG_FAULT4	C8	INTLK4-	**A8**	+REF_1	**B8**	+REF_5	**C8**	+REF_9
A9	+15V_IN_AUX	B9	+5V_D	C9	MAG_FAULT5	**A9**	AGND	**B9**	AGND	**C9**	AGND
A10	+5V_IN_AUX	B10	AD7_CS	C10	MAG_FAULT6	**A10**	MON_1	**B10**	MON_5	**C10**	MON_9
A11	+5V_IN_AUX	B11	AD7_DOUTA	C11	MAG_FAULT7	**A11**	MCOR_FAULT1	**B11**	MCOR_FAULT5	**C11**	MCOR_FAULT9
A12	+5V_IN_AUX	B12	AD7_DOUTB	C12	DGND	**A12**	FDBK_1	**B12**	FDBK_5	**C12**	FDBK_9
A13	AD6_RESET	B13	AD7_BUSY	C13	RS232_DATA_OUT	**A13**	GND_2	**B13**	GND_6	**C13**	GND_10
A14	AD6_FDATA	B14	AD7_SCLK	C14	RS232_DATA_IN	**A14**	+REF_2	**B14**	+REF_6	**C14**	+REF_10
A15	AD6_CONVST	B15	AD7_REFSEL	C15	N/C	**A15**	AGND	**B15**	AGND	**C15**	AGND
A16	AD6_SCLK	B16	4KHZ_TRIG_POS	C16	N/C	**A16**	MON_2	**B16**	MON_6	**C16**	MON_10
A17	AD6_CS	B17	AGND	C17	DA5_SD0	**A17**	MCOR_FAULT2	**B17**	MCOR_FAULT6	**C17**	MCOR_FAULT10
A18	AD6_DOUTA	B18	AGND	C18	DA5_RESET	**A18**	FDBK_2	**B18**	FDBK_6	**C18**	FDBK_10
A19	AD6_DOUTB	B19	DGND	C19	DA5_LDAC	**A19**	GND_3	**B19**	GND_7	**C19**	GND_11
A20	AD6_BUSY	B20	DGND	C20	DA5_CLR	**A20**	+REF_3	**B20**	+REF_7	**C20**	+REF_11
A21	AD6_REFSEL	B21	+5V_A	C21	DA5_SDIN	**A21**	AGND	**B21**	AGND	**C21**	AGND
A22	AD7 RESET	**B22**	+5V_A	**C22**	DA5_SCLK	**A22**	MON_3	**B22**	MON_7	**C22**	MON_11
A23	AD7 FDATA	**B23**	+5V_D	**C23**	DA5_SYNC	**A23**	MCOR_FAULT3	**B23**	MCOR_FAULT7	**C23**	MCOR_FAULT11
A24	AD7 CONVST	**B24**	+5V_D	**C24**	N/C	**A24**	FDBK_3	**B24**	FDBK_7	**C24**	FDBK_11
A25	HVDC(F)	**B25**	GND RES+	**C25**	GND_RES-	**A25**	GND_12	**B25**	MON_12	**C25**	AGND

A26	+5V_D (thru R84)	**B26**	BULK ENA IN	**C26**	N/C	**A26**	+REF_12	**B26**	FDBK_12	**C26**	MCOR_FAULT12
A27	Water Sum Fault	**B27**	BULK ENA OUT	**C27**	N/C	**A27**	FDBK_15	**B27**	+REF_13	**C27**	GND_13
A28	+5V_IN_CRATE	**B28**	+5V_IN_CRATE	**C28**	+5V_IN_CRATE	**A28**	MCOR_FAULT15	**B28**	MON_13	**C28**	AGND
A29	DGND	**B29**	DGND	**C29**	DGND	**A29**	MON_15	**B29**	FDBK_13	**C29**	MCOR_FAULT13
A30	+15V_IN_CRATE	**B30**	+15V_IN_CRATE	**C30**	+15V_IN_CRATE	**A30**	AGND	**B30**	+REF_14	**C30**	GND_14
A31	-15V_IN_CRATE	**B31**	-15V_IN_CRATE	**C31**	-15V_IN_CRATE	**A31**	+REF_15	**B31**	MON_14	**C31**	AGND
A32	SYNC BUS	**B32**	INH_BUS	**C32**	RESET_BUS	**A32**	GND_15	**B32**	FDBK_14	**C32**	MCOR_FAULT14

Table 3 *Crate J3 and J4 connections to EMCOR J1 connections*

EMCOR J2 Pin	Signal Name	Corresponding Crate J3 Pin	EMCOR J2 Pin	Signal Name	Corresponding Crate J4 Pin
A1	+15V_IN_AUX	A1	A13	AD6 RESET	A1
A2	+15V_IN_AUX	A2	A14	AD6 FDATA	A2
A3	+5V_IN_AUX	A3	A15	AD6 CONVST	A3
A4	+5V_IN_AUX	A4	A16	AD6 SCLK	A4
A5	DGND	A5	A17	AD6 CS	A5
A6	DGND	A6	A18	AD6 DOUTA	A6
A7	-15V_IN_AUX	A7	A19	AD6 DOUTB	A7
A8	-15V_IN_AUX	A8	A20	AD6 BUSY	A8
A9	+15V_IN_AUX	A9	A21	AD6 REFSEL	A9
A10	+5V_IN_AUX	A10	A22	AD7 RESET	A10
A11	+5V_IN_AUX	A11	A23	AD7 FDATA	A11
A12	+5V_IN_AUX	A12	A24	AD7 CONVST	A12
B1	DGND	B1	B13	AD7 BUSY	B1
B2	DGND	B2	B14	AD7 SCLK	B2
B3	-15V_IN_AUX	B3	B15	AD7 REFSEL	B3
B4	MAG FAULT0	B4	B16	4KHZ TRIG POS	B4
B5	MAG FAULT1	B5	B17	AGND	B5
B6	MAG FAULT2	B6	B18	AGND	B6
B7	MAG FAULT3	B7	B19	DGND	B7
B8	MAG FAULT4	B8	B20	DGND	B8
B9	+5V_D	B9	B21	+5V_A	B9
B10	AD7 CS	B10	B22	+5V_A	B10
B11	AD7 DOUTA	B11	B23	+5V_D	B11
B12	AD7DOUTB	B12	B24	+5V_D	B12
C1	INTLK1+	C1	C13	RS232 DATA OUT	C1
C2	INTLK1-	C2	C14	RS232 DATA IN	C2
C3	INTLK2+	C3	C15	N/C	C3

C4	INTLK2-	C4	C16	N/C	C4
C5	INTLK3+	C5	C17	DA5 SD0	C5
C6	INTLK3-	C6	C18	DA5 RESET	C6
C7	INTLK4+	C7	C19	DA5 LDAC	C7
C8	INTLK4-	C8	C20	DA5 CLR	C8
C9	MAGFAULT5	C9	C21	DA5 SDIN	C9
C10	MAGFAULT6	C10	C22	DA5 SCLK	C10
C11	MAGFAULT7	C11	C23	DA5 SYNC	C11
C12	DGND	C12	C24	N/C	C12

EXT INTLK connection J1 Description

Crate connection J1, which is a BNC connector on the back of the crate labeled "**EXT INTLK",** must be configured correctly for normal operation of the crate. As is currently the case with the Interface Controller module, a closed switch must be present across J1 or an MCOR fault will be generated. The signal name for this connection is 'Water_Sum_Fault'. An open switch will cause +5V on the water_sum_fault circuits which will generate a fault condition and disable all MCOR modules.

Crate OK connection J2 Description

When properly configured this BNC connection may be used as a dry contact output to enable/disable a properly configured Bulk Power Supply output.
A properly equipped power supply used as the bulk supply must have a connection available that will enable the output of the supply with a short across the applicable pins. Consult the operator's manual of the power supply in use for more specific information.

Event Receiver (EVR)

Currently these functions have not yet been implemented.

Fast Feedback

Currently the Fast Feedback function has not yet been developed.

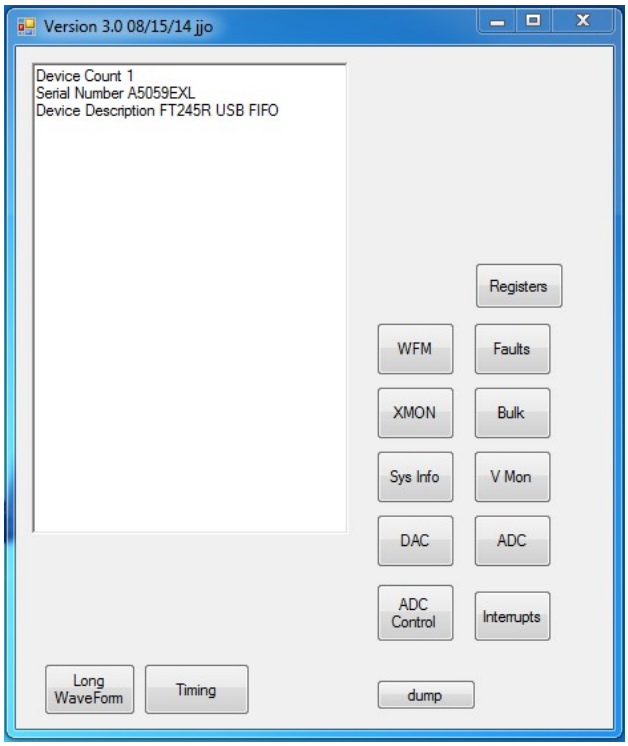
EMCOR Crate Diagnostic Application

This is a Windows based GUI application primarily used as a diagnostic application to monitor and reset certain faults, set and monitor MCOR channels, and turn on/off a properly configured bulk power supply. A USB male Type A connection attached to a PC with a USB male Type B connection attached to EMCOR connector J9 is required for use with this application.

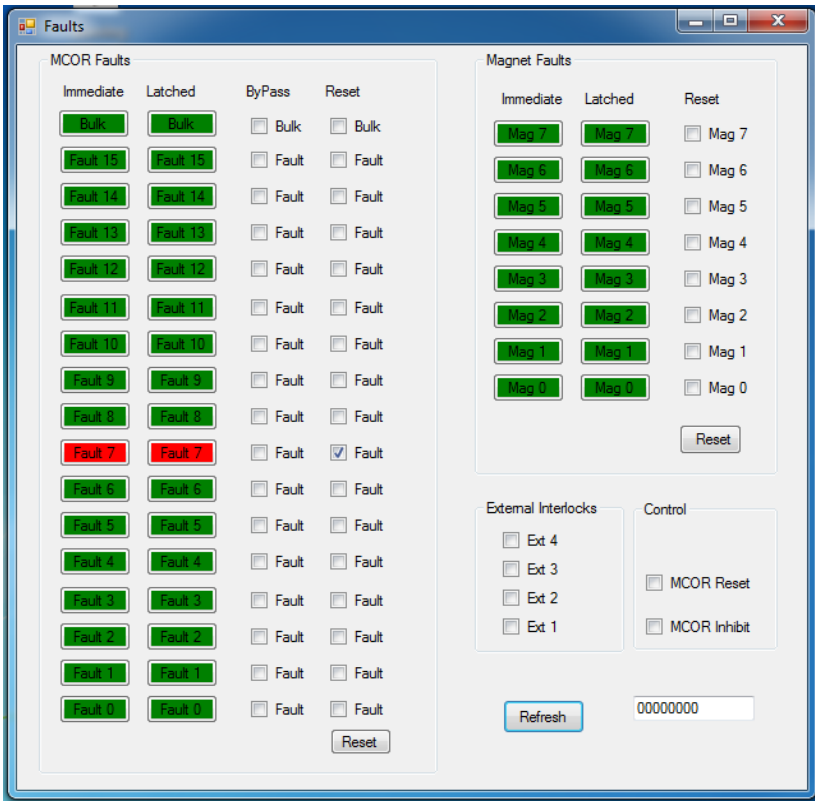
The application may be used as soon as the crate has been powered up and can be used whether the CPU boot process has been completed or not. If the CPU boot up process is not going to be performed, both the serial and Ethernet cables (J7 and J5) do not need to be connected. Once the crate is powered up and the power-up sequence of the EMCOR module has completed, a green (OK) LED on the EMCOR module should be slowly flashing. A red LED on the MCOR module(s) may also be flashing (assumes that the bulk supply is connected and enabled, and a closed switch across the crate J1

connector) which would indicate a fault on the MCOR module(s).
The application may now be opened and the main panel shown in screen shot 1 will appear.
To remove the fault(s) associated with any MCOR modules during the power up sequence, (assuming there are any) click on the 'Faults' tab on the main panel. The 'Faults' panel will open up (screen shot 2). After the 'Refresh' tab of the 'Faults' panel is clicked, all faults associated with the MCOR modules will show up in red (a fault indication on this screen will correspond to a flashing red LED on the indicated channel MCOR module). In screen shot 2 only one module is showing in red since the crate is only populated with one MCOR module (channel 7). The fault can be cleared by checking the 'MCOR Reset' box and then clicking on the 'Refresh' tab. Note that the 'Faults' display shows two types of faults, Latched and Immediate. The only way to clear a 'Latched' fault is with the software 'MCOR Reset' selection. An 'Immediate' fault can be cleared by using the EMCOR front panel Reset button. The 'Faults' display should now appear as shown in screen shot 3, with all MCOR faults appearing in green as shown on the panel, and no flashing red LEDs as shown on the MCOR module(s). It should be noted at this time that even though the 'Faults' panel displays 'Magnet' and 'External Interlock' faults, they are not active on the EMCOR module at this time (have not been implemented) and will not display faults. Once the fault(s) of the MCOR modules has been cleared, the EMCOR module and MCOR modules are ready to be programmed per the instructions below.

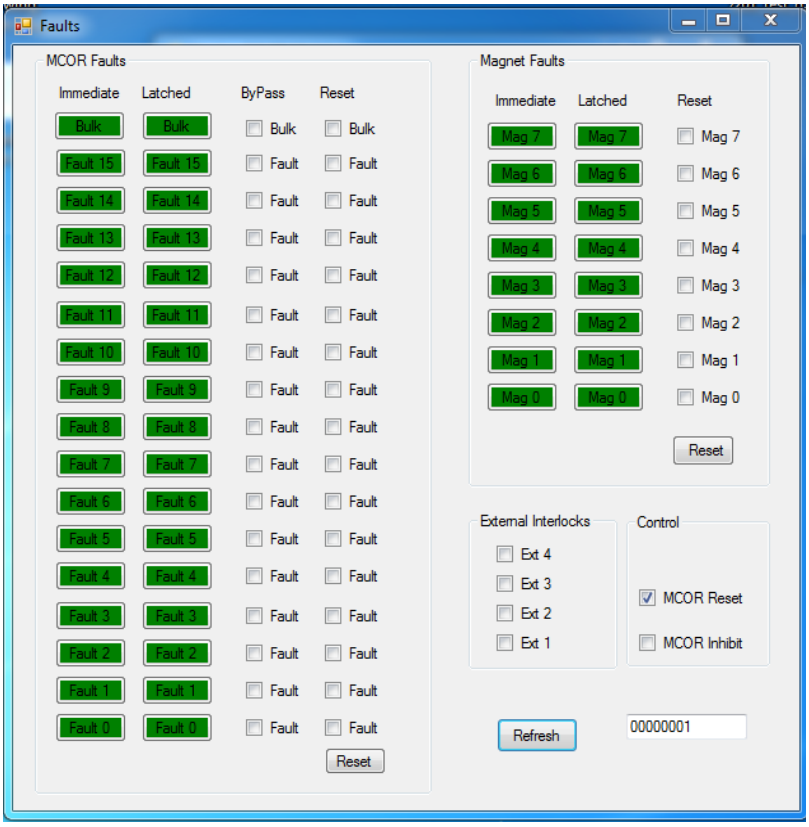
The application is ideally suited for configuring the current setting on any MCOR module and then determining if the output setting matches the desired setting. The following instructions can be used as an example to set an MCOR module's current setting and then reading the value back to confirm that the setting was properly performed. The screen shots below will aid in the use of the application.
From the main panel of the application click on the 'DAC' button.



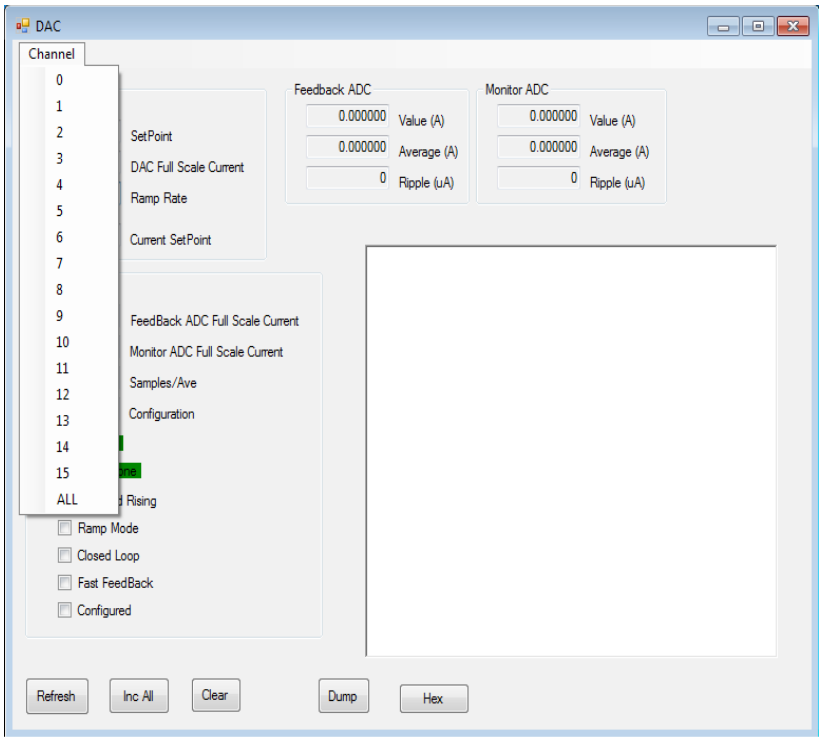
Screen shot 1- Main Panel



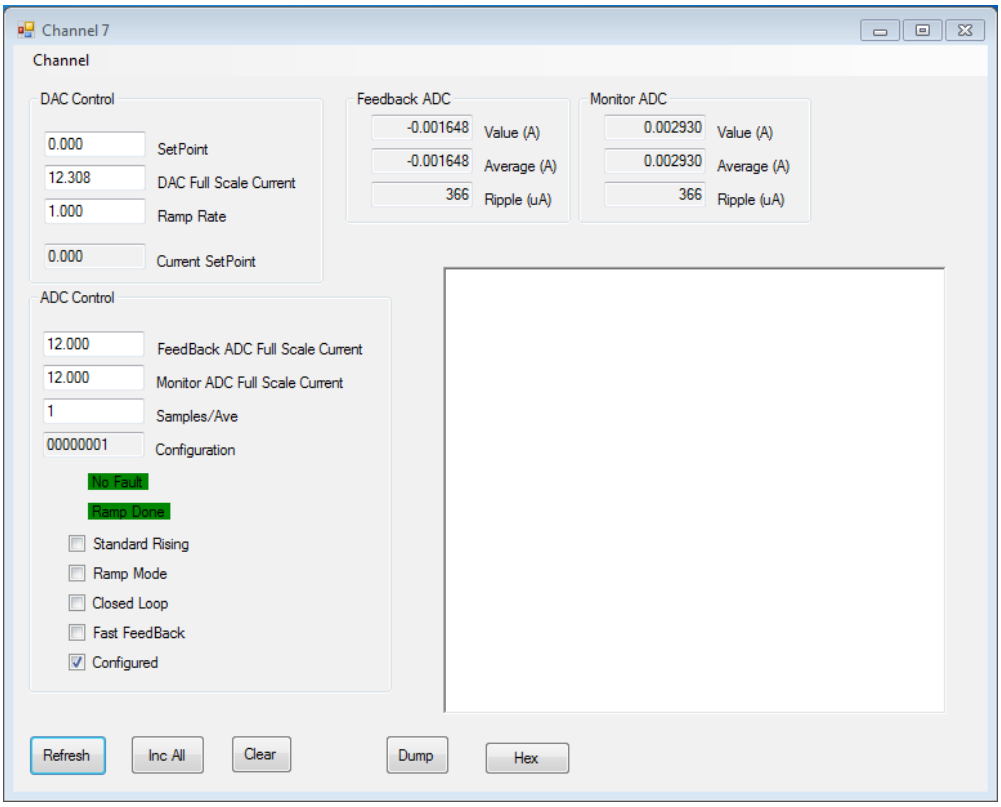
Screen shot 2- Faults Panel with fault



Screen Shot 3- Faults Panel with no faults



Screen Shot 4-Channel Pulldown



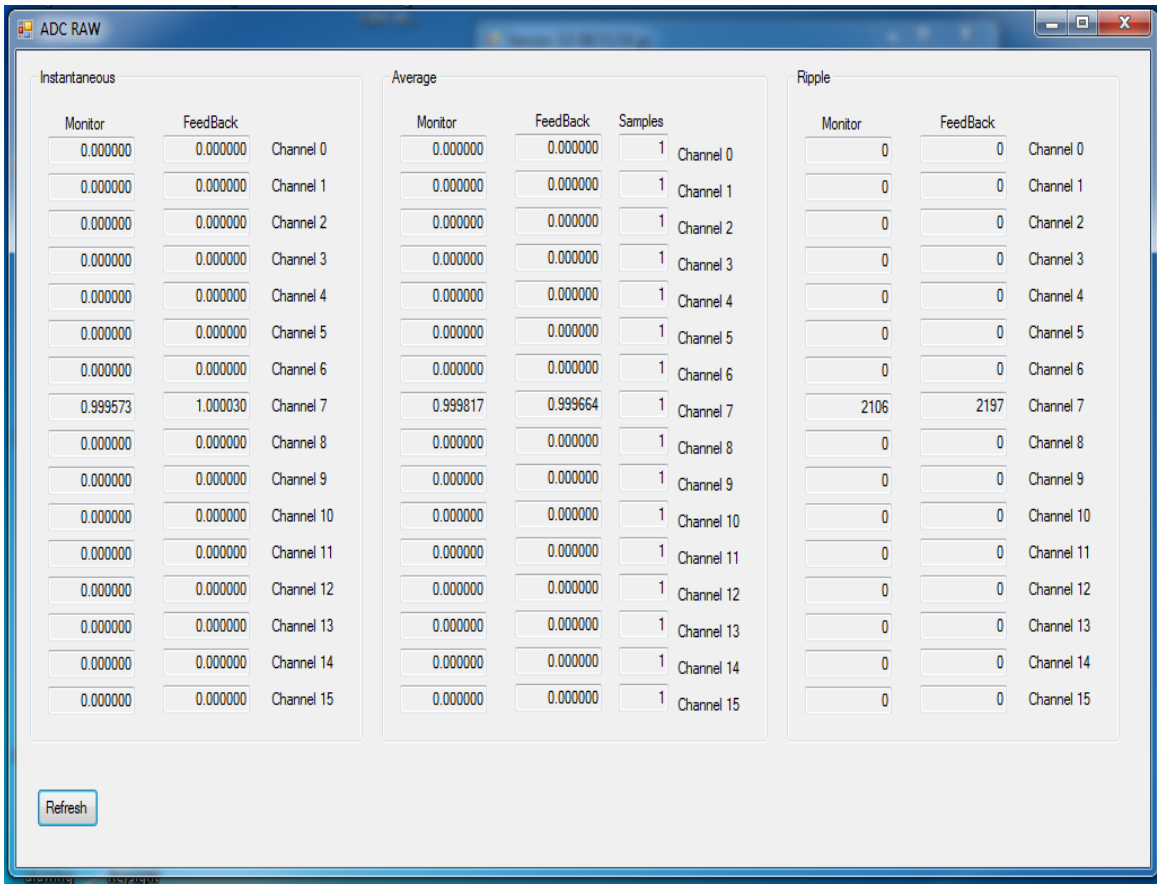
Screen Shot 5-DAC Panel

Select the channel to be set from the 'Channel' pull-down (Screen shot 4) at the top left of the DAC panel. Next, set the 'DAC Full Scale Current' setting from the **'Full Scale Current**' table below. This setting will be based on the type of MCOR module being used (for an MCOR 12 module as an example, '12.30768' would need to be entered in the 'DAC Full Scale Current' window). Once this value has been set, you must then hit the 'Enter' key. Use the same table to determine the Monitor ADC and Feedback ADC settings and enter those values using the 'Enter' button as well. Once those values have all been entered, click the 'Configured' box at the bottom of the panel. Then set the 'Set Point' to the desired value of current and hit the enter or return key (note that there must be a load connected to the output on the back of the crate before any current will flow). The 'Set Point' window cannot be set to a value unless the 'Configured' box has been checked. Click on the 'Refresh' button for the changes to take effect on the panel display. Select the 'ADC' panel (from the main panel). This panel (screen shot 5) will enable the user to monitor any settings made and determine that the output is the same as what was requested (after clicking the 'Refresh' button). The 'DAC' panel also displays the Feedback and Monitor current values for the channel selected.

Note that a ramp rate for the current to reach full level can be set by entering the desired value in the 'Ramp Rate' window (DAC Panel). The 'Ramp Mode' box must be displayed for the setting to take effect after the enter button is pressed. The value entered in the ramp rate window is in amps/second. The alternative to 'Ramp Mode' is the 'Immediate' Mode.

Full Scale Current Table

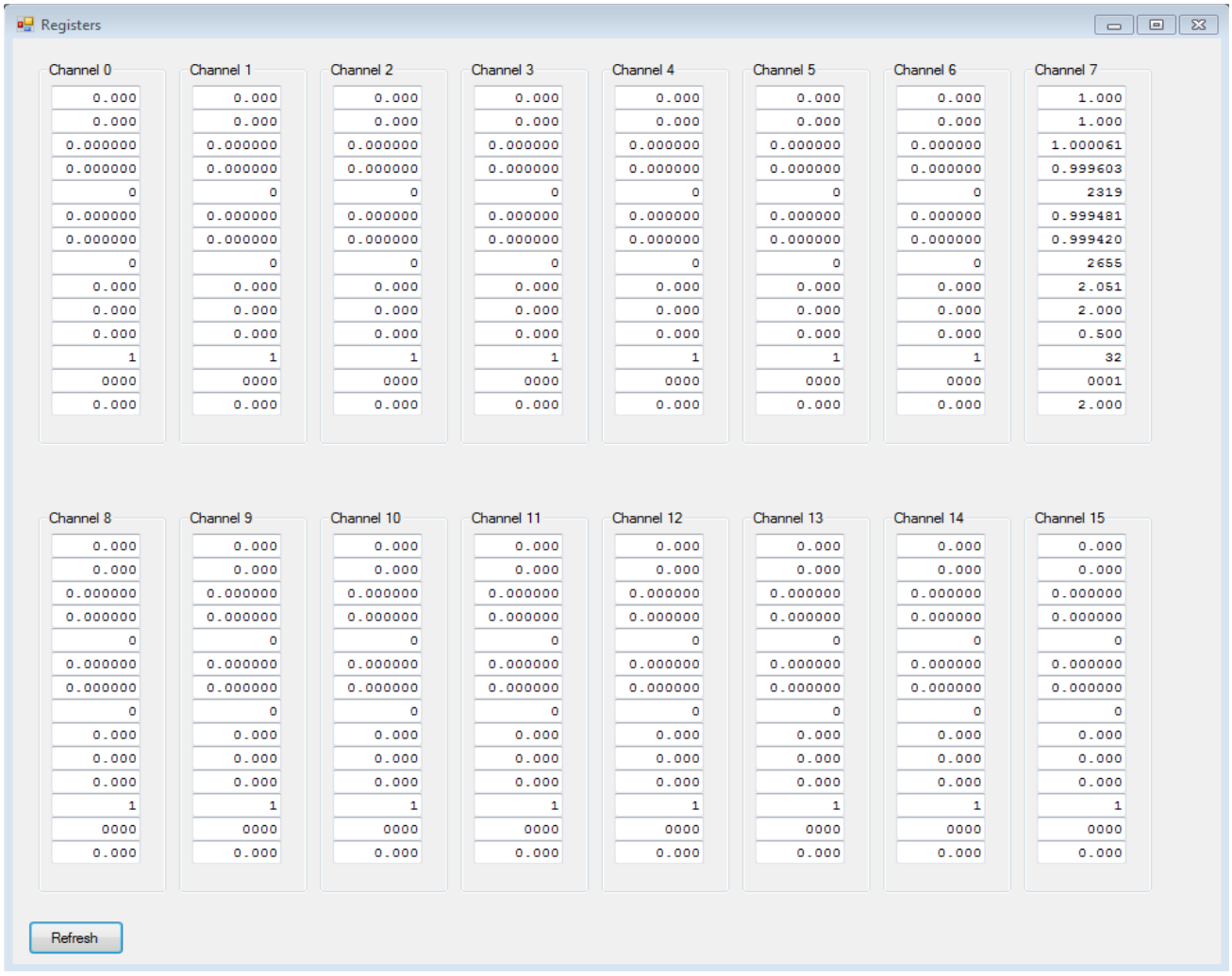
MCOR	DAC	Monitor ADC	Feed-Back ADC	DAC Full Scale Setting
30	30A	30A	30A	30.7692A
20	20A	20A	20A	20.5128A
12	12A	12A	12A	12.30768A
9	9A	9A	9A	9.23076A
7.5	7.5A	7.5A	7.5A	7.6923A
6	6A	6A	6A	6.15384A
2	2A	2A	2A	2.05128A
1.5	1.5	1.5A	1.5A	1.53846A
1	1	1A	1A	1.02564A



Screen Shot 6- ADC Raw Panel

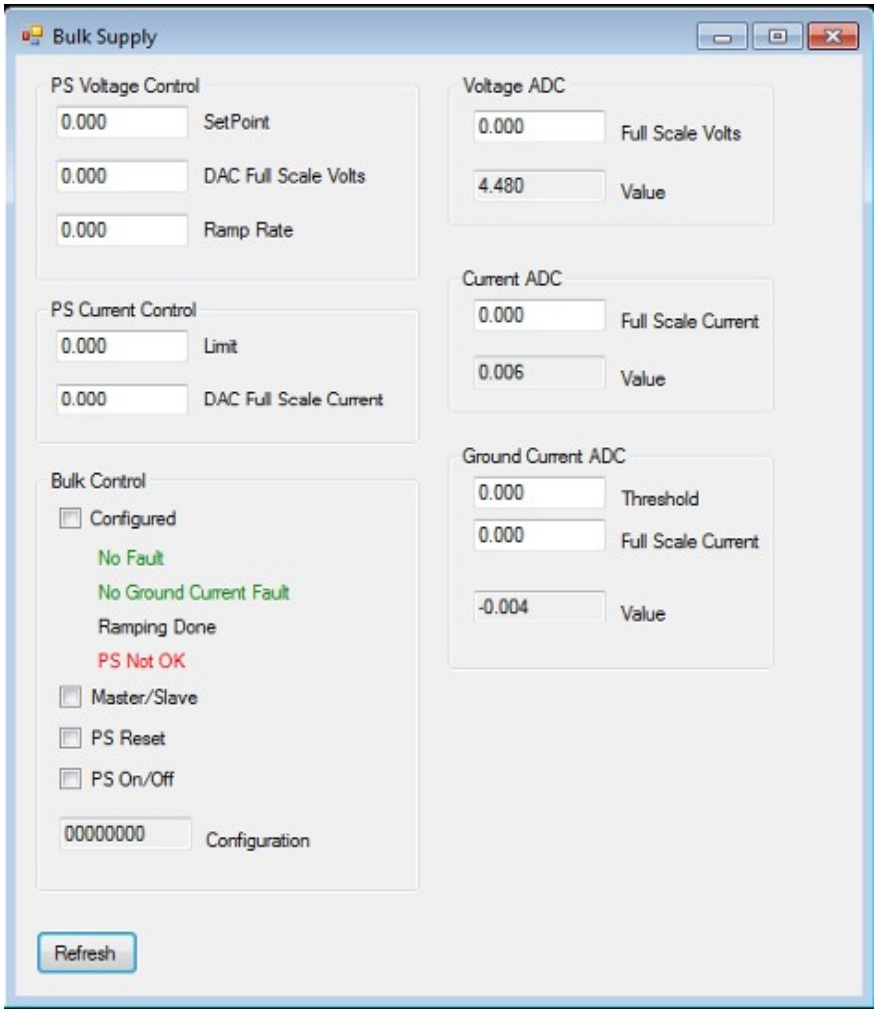
The 'Register' Panel (screen shot 7) allows all 16 crate channels to be monitored for the following (top to bottom) parameters:Note that readings do not automatically update. 'Refresh' must be utilized for updated results.

- Set Point
- Current Set Point
- Monitor ADC Reading
- Monitor Average ADC Reading
- Monitor Ripple Measurement (in uA)
- Feedback ADC Reading
- Feedback Average ADC Reading
- Feedback Ripple Measurement (in uA)
- Full scale DAC Set Point Current
- Full scale Monitor ADC Read Back Current
- Ramp Rate (amps/sec)
- Samples per Average
- Configuration/Status Register
- Full scale Feedback ADC Read back Current



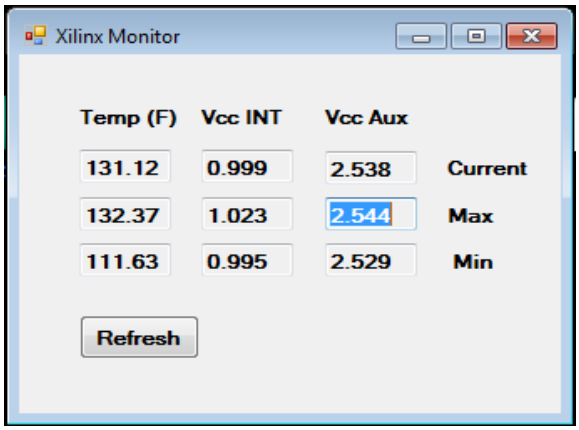
Screen Shot 7- Registers Panel (Channel Parameters)

The 'Bulk Supply' Panel is shown in screen shot 8 below. This panel is used for enabling/disabling an external bulk supply only. The bulk supply must support being remotely enabled/disabled, which is accomplished by the crate via the 'Crate OK' connection on the back of the crate. This connection is a BNC connector (J2) which is a dry contact (relay) output. When enabled ('PS On/Off' box checked) a short is placed across the enable connection of the power supply pins to turn the output of the bulk supply on. The user must consult the manual of the specific power supply in use to determine the connections necessary for this function. BiRa Systems can help with this if necessary.
It is necessary to configure the bulk supply properly to ensure that the supply comes up in the correct state so that the output of the supply does not come on until commanded. The bulk supply voltage, Over Current, Over Voltage protection must be set up locally and cannot be controlled by the EMCOR module. The power supply should be set up so that when powered on the output is enabled but the output is not on. Note that it is not necessary to remotely access the bulk supply output. The bulk supply may be used manually by the user if desired.



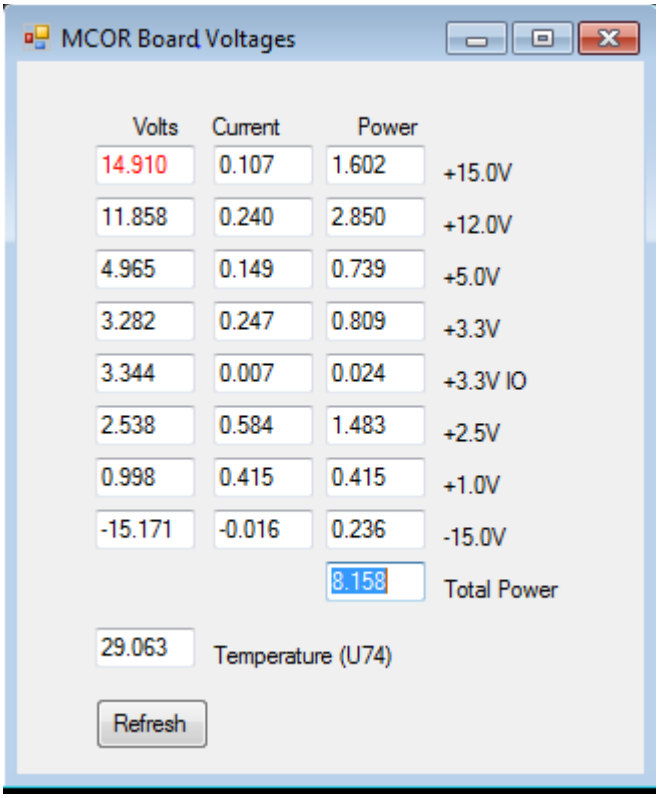
Screen Shot 8- Bulk Supply Panel

The USB application can also be used for monitoring the Xilinx (FPGA) temperature and voltage. From the main panel select 'XMON' and the panel shown in screen shot 8 will appear. The current values of the 1.0V and 2.5V supply voltages and the temperature of the part will be shown upon clicking the 'Refresh' button.



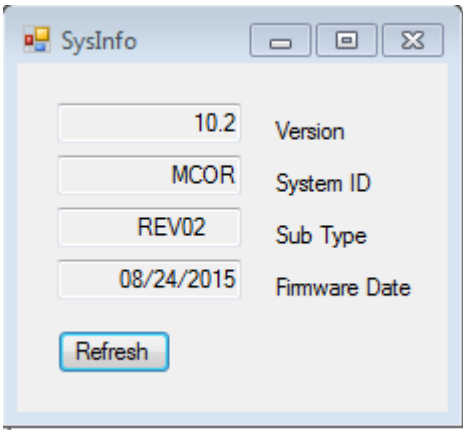
Screen Shot 9- Xilinx Monitor

To monitor the EMCOR board voltages select the 'V MON' tab from the main panel. Screen shot 10 is representative of this panel after clicking the 'Refresh' button.



Screen Shot 10 EMCOR Board Voltages Panel

For System information the 'Sys Info' tab can be used as shown in screen shot 11.



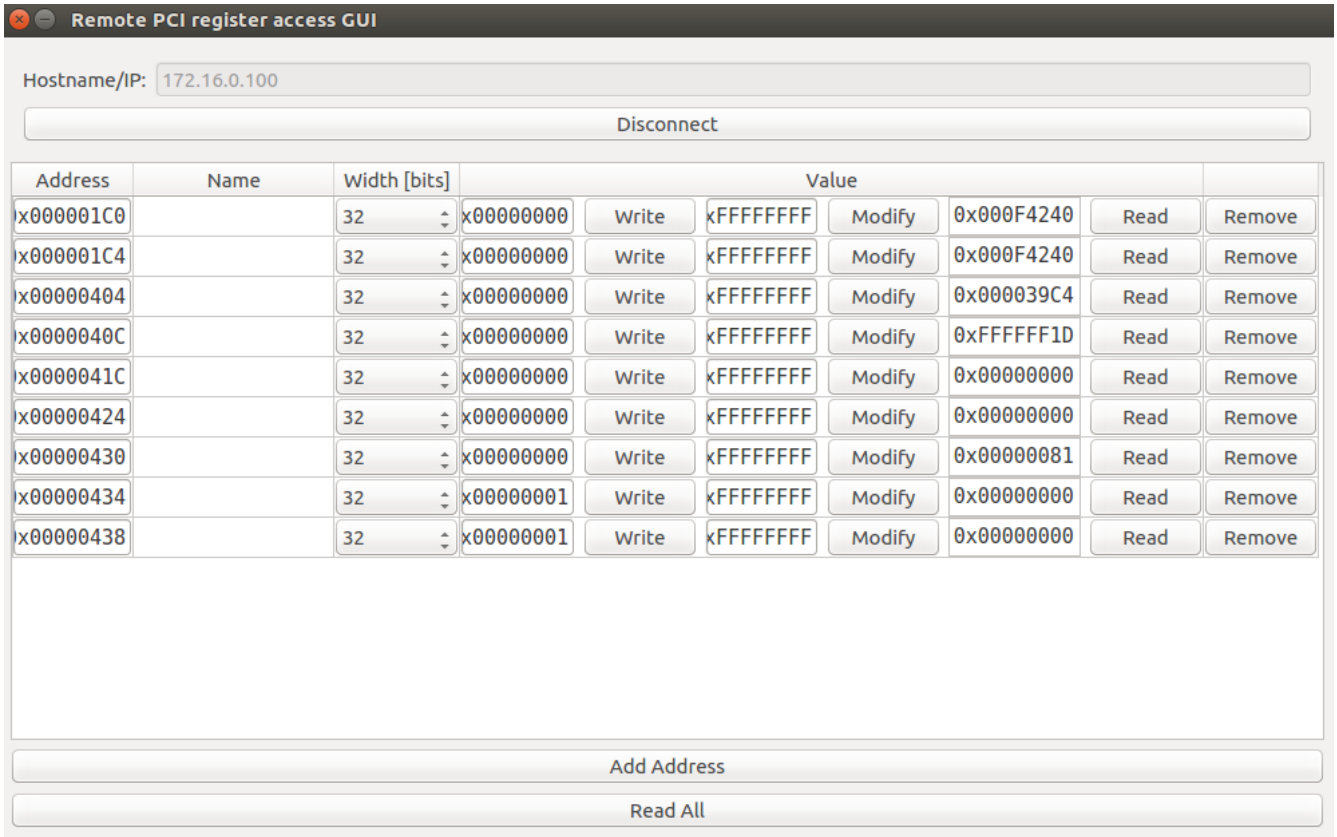
Screen Shot 11- Sys Info Panel

The front panel of the EMCOR module is shown below. The J9 connector is used for USB communication. The P9 connector (JTAG) is used for programming of the FPGA, which is done by BiRa Systems and should never need to be accessed by the user. The jumper that can be seen in the P9 connector is for stability of the boot-up process and should not be removed. The EVR connector is not active at this time. The EPICS CA (J5) connector is for Ethernet communication. The COM (J7) connection port is used for serial communication. CPU RST (S2) is a switch that when activated will reset the CPU, which will cause a re-boot of the module to occur. The Fast Feedback function is not active at this time.

EMCOR Front Panel

Remote PCI Register Information

The tables below contain detailed information on the EMCOR Registers. Refer to the following tables for use in accessing data from the EMCOR module.
Some of the registers may not be active since some of the features of the EMCOR module have not yet been incorporated.
The registers may be accessed by a GUI that resides on the PC that contain the CPU boot-up software, and is a LINUX based OS.
Should it be necessary or desirable to access the PCI Registers, the following instructions should be used. The following instructions assume that the CPU boot-up process has completed normally (see boot-up instructions below).
Open up a terminal on the LINUX based machine used for the boot up process, and enter the following, being careful to make the entries exactly as shown.
"**cd emcor-python-package/Python-RemotePCI-GUI**". Once this information is entered you are in the correct directory to access the function. Next, enter "**python pyrpci_gui.py**", which is the actual python based GUI. In order to access a desired register it is first necessary to enter the IP address (172.16.0.100) at the top of the GUI, and then hit the connect button. A screen shot of the register is shown here.



Once the GUI has been accessed, the specific address of the register with the desired data must be entered, and a read of the register executed. The data is in a 32 bit hex format which must then be converted to a decimal equivalent. Information in the tables below will indicate what format the data is in (generally signed or unsigned integer) and the units of the data.
As an example of how to convert the data to usable information, suppose that one wanted to know what the Set point requested value was for channel 7. The BAR (Base Address Register) tells us that the Channel Control Registers reside between address 0x00000000 to 0x000003C0. The EMCOR channel registers table tells us that the base address for channel 7 is 0x000001c0. Since the channel settings registers table shows us that the information we're looking for (Set Point Requested) has an offset of 0x00, we would enter the address 0x000001c0 into the GUI and request a read of the information in this register. In this case the data in the register reads 0x000f4240. The channel settings register tells us this data is a signed integer format with units of uA. When translated to decimal, the result is 1000000, or the equivalent of 1amp, which is the current set point for channel 7.

The BAR 0 (**Base Address Register**) Memory Map of the EMCOR Registers is shown below.

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

The Table below shows the offset of each of the 16 channels.

MCOR Channel 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

The Tables below shows how each channel is configured.

Channel Settings 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 Registers

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

Bulk Supply Register Information

Offset	Reg	Base 0x0400	
0x04	1	Bulk Supply Voltage 0-5V => 0-30V	Int32 in µV
0x0C	3	Ground Fault Current	Int32 in µA
0x1C	7	Bulk Supply ADC Full Scale Voltage	Int32 in µV
0x24	9	Bulk Supply ADC Full Scale Ground Current	Int32 in µA
0x28	A	Ground Fault Current Threshold	Int32 in µV
0x30	C	Configuration/Status Register	UInt32
0x34	D	Set Configuration Register	UInt32
0x38	E	Reset Configuration Register	UInt32

Bulk Supply Status/Configuration Register Information

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
0	PS On/Off Request (Bulk ENA_In/Out)	'1' → Bulk Supply ON '0' → Bulk Supply OFF

Bulk Supply Set/Reset Configuration Register

Bit		
0	PS On/Off Request	'1' → Turn Bulk Supply ON '0' → Turn Bulk Supply OFF

MCOR ADC Control Registers Information

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

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

Set/Reset Control Register

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

ADC 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	

ADC External Reference Control

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

MCOR Faults 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	

MCOR Fault Status

Bit		
[15:00]	MCOR Power Module Fault Status	

MCOR Latched Fault Status

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

MCOR Reset Latched Fault Status

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

MCOR 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

MCOR Set Control/Reset Control Registers

Bit		
[1]	MCOR Inhibit	
[0]	MCOR Reset	

Interlock and Magnet Fault Registers (note that 4 Interlock outputs and 8 Magnet Fault inputs are not currently implemented)

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	

External Interlock Status

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

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

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

Input Status

Bit		
[31:09]		
8	Water Fault	
[07:00]	Inputs	

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 (0x0014)

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

Voltage Monitor Registers

Offset	**Base = 0x0540**	Int32
0x00	+15.0V(In)	
0x04	+12.0V(In)	
0x08	+5.0V(In)	
0x0C	+3.3V	
0x10	+3.3VCC IO	
0x14	-15.0V(In)	
0x18	+15.0V(In) Current	
0x1C	+12.0V(In) Current	
0x20	+5.0V(In) Current	
0x24	+3.3V Current	
0x28	+3.3VCCIO Current	
0x2C	+2.5V Current	
0x30	+1.0V Current	
0x34	-15.0V(In) Current	
0x38	Board Temperature	N* 0.0625

Xilinx System Monitor Registers

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)	

System Information Registers

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”

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

Interrupt Source Enables, Set/Reset Registers

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

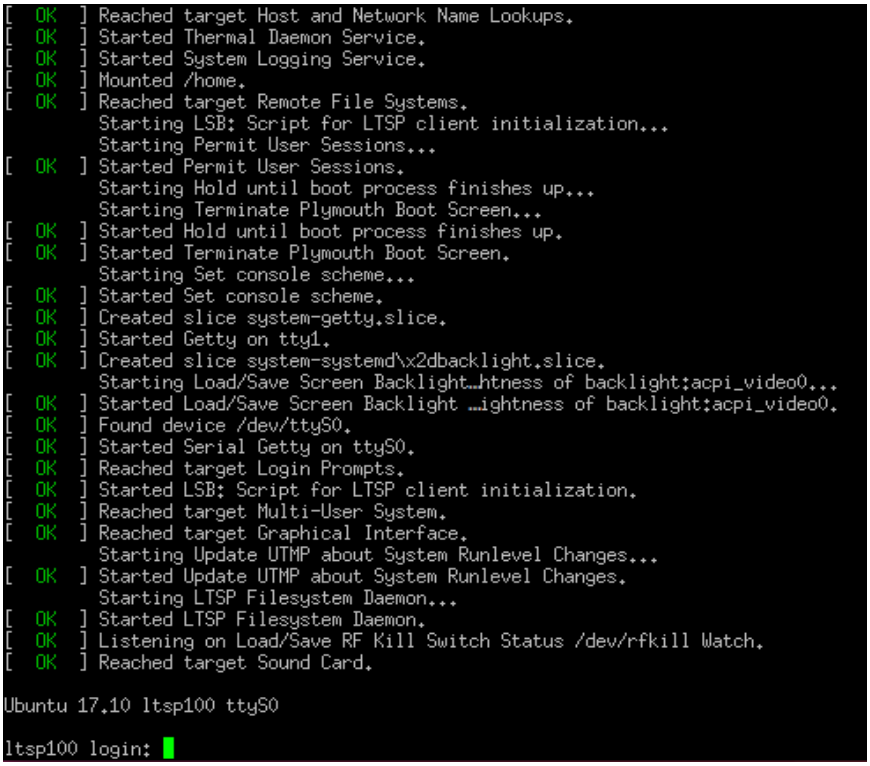
Software Interrupt Register

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

EMCOR Module CPU Boot-up process

Before the EMCOR module can be used for controlling MCOR modules (except if using the USB application to control MCOR modules) when installed in a crate power supply, the EMCOR module must go through a boot-up process each time the crate power is cycled.
The software used for the boot-up process uses a Linux based OS. Use the following instructions to perform the CPU boot-up process. Note that if the CPU MAC address had not previously been entered and saved, that process must be done before the boot-up process will complete. The CPU MAC address only needs to be entered and saved once, but the CPU boot-up process must be completed every time power to the crate is cycled. Use the instructions in the 'Entering and Saving the CPU MAC address' below if necessary, otherwise use the instructions to boot up the EMCOR module immediately below.

The proper connections from the PC to the EMCOR module necessary are a serial connection and an Ethernet connection. The serial connection from the PC to the EMCOR module may use a USB output and serial adapter if a serial port is not available on the PC. The connector on the EMCOR module (J7) is an Ethernet connection, which will require an adapter from the serial connection of the PC. The Ethernet connection from the PC will connect directly to the J5 connector on the EMCOR module. Once the connections from the boot-up PC to the EMCOR module in the crate have been made, power up the crate and observe that the red (Fault) LED on the EMCOR front panel begins to flash for a period of time, go off for a short period of time, and then begin to flash again for a shorter period of time, then go off. Once the red LED goes off the second time, the green LED (OK) will begin to slowly flash . This set of events indicates a successful power up sequence for the module. Once the green LED begins flashing, the CPU boot up process on the PC will begin. If this process is successful, the screen shown below should appear with a log in prompt. Once the login user name (steve) and password (abcd1234) have been entered, the boot-up process is complete.



Screen shot showing the display after a successful boot up

Entering and Saving the CPU MAC address

If it is necessary to enter the CPU MAC address, follow the instructions in this paragraph. If the dhcpd.leases file is not loaded with the correct MAC address of the CPU, the boot up process will not complete.
Open a new terminal on the workstation computer by right clicking on the Terminal icon on the left side of the monitor. Once opened, the default prompt should be: "steve@steve-dell: ~$".

At the prompt type in "cd /", which will place you in the root directory. Type the command "**cd var/lib/dhcp**" which will place you in the directory where the file "**dhcpd.leases**" resides. To edit this file and set the correct MAC address, type the command "**sudo gedit dhcpd.leases**" enter the password when prompted and then enter the correct 12 character MAC address into the file that was recorded during the CPU installation described above (the MAC address should also be on a label attached to the CPU heat spreader). Save the file and enter the password when prompted. At this point the dhcp server must be restarted for the new MAC address to be recognized. This is done by entering the command "**sudo service isc-dhcp-server restart**".
The screen shot below shows the dhcpd.leases file with the CPU MAC address highlighted.

```
# The format of this file is documented in the dhcpd.leases(5) manual page.
# This lease file was written by isc-dhcp-4.3.3

lease 172.16.0.100 {
  starts 1 2018/06/11 16:43:24;
  ends 2 2018/06/12 04:43:24;
  cltt 1 2018/06/11 16:43:24;
  binding state active;
  next binding state free;
  rewind binding state free;
  hardware ethernet 00:07:32:48:01:5e;
  set vendor-class-identifier = "PXEClient:Arch:00000:UNDI:002001";
}
```

Screen shot of DHCPD.leases file

Once these instructions have been completed go back to the previous paragraph and follow the instructions for the boot up process.

After a successful log in

Once the log in has been completed successfully it may be necessary to log into the rios server. This is primarily dependent upon the software used to control the EMCOR module. If the ethernet connection on the EMCOR module will be used to communicate with the EMCOR module (such as EPICS), it will be necessary to follow the instructions below and log on to the rios server.

Log On to the Rios

To log on to the rios server, use these instructions after a successful log in and at the prompt type: "**sudo ./rios 0000:01:00.0@0**". If successful the response will be "Bound socket to port 4444".

If the user is using the USB port to communicate with the EMCOR module and control MCOR modules (using the Windows based USB application for example) it is not necessary to log in to the rios server.