

SPEAR 3

MCOR30 System

TABLE OF CONTENTS

1.	System Description	5
1.1.	MCOR30 System Block Diagram	5
1.2.	MCOR Crate	5
1.3.	MCOR30 – Corrector Power Supply	6
1.4.	CANDI (Controller ANalog/Digital Interface).....	6
1.5.	FrankenBoard.....	6
1.6.	AUX Board (Bride of FrankenBoard)	6
1.7.	VME Processor	6
2.	MCOR30 Feedback Operation	6
3.	CANDI Board	7
3.1.	Front Panel	7
3.2.	RS232 connector.....	8
3.3.	Back Plane Signals.....	8
3.4.	MCOR30 Interface Signals.....	9
3.5.	Calibration DAC	11
3.6.	Set Point DAC.....	11
3.7.	CANDI ADC's.....	11
3.8.	I1MON ADC	11
3.9.	I2MON ADC	12
3.10.	VMON ADC.....	12
3.11.	MCOR30 Faults.....	12
3.12.	CANDI CSR	13
3.13.	CANDI Commands.....	13
3.13.1.	Serial Interface Operation	13
3.13.2.	RS232 Interface	14
3.13.3.	CANDI Commands.....	14
3.13.4.	Write Set Point DAC Command (9)	15
3.13.5.	SYNC Command (15).....	15
4.	AUX Board	15
4.1.	Power Connection.....	15
4.2.	AUX Board Configuration Switch.....	16
4.2.1.	Crate ID.....	16
4.2.2.	DAC/SAM Mode	16
4.2.3.	Reset Switch.....	17
4.3.	RS232 Connector	17
5.	FrankenBoard.....	17
5.1.	VME Address and Data Interface	17
5.2.	FrankenBoard Interrupts	17
5.3.	FrankenBoard to VME Memory Map.....	17
5.4.	Read CANDI Board data	18
5.5.	Write CANDI CSR	18
5.6.	Refresh Command	19
5.7.	Write FrankenBoard CSR	19
5.8.	CSR0 -- Franken Board Link Status	19
5.9.	CSR1 -- Internal Sync Timer	19

5.10.	CSR2 and CSR3 System Sync Word.....	20
5.11.	CSR4 -- Aux Card Interface and Mode Bits.....	20
5.11.1.	AUX board settings.....	20
5.11.2.	Ground Fault	20
5.11.3.	SYNC Mode.....	21
5.11.4.	Data Mode.....	21
5.12.	CSR5 – MCOR Back-plane Interface.....	21
5.13.	FrankenBoard Voltage Monitor	23
5.13.1.	Ground sense voltage	23
5.13.2.	MCOR Bulk Supply.....	23
6.	Firmware Modifications and Versions.....	24
6.1.	Franken Board Version History	24
6.2.	CANDI Board Version History	25
7.	MCOR30 DAC/SAM Connector Assignment.....	26
8.	MCOR30 Slot Zero Back-plane Signal Assignment (P1)	27
9.	MCOR30 Slot Zero Back-plane Signal Usage (P2)	28

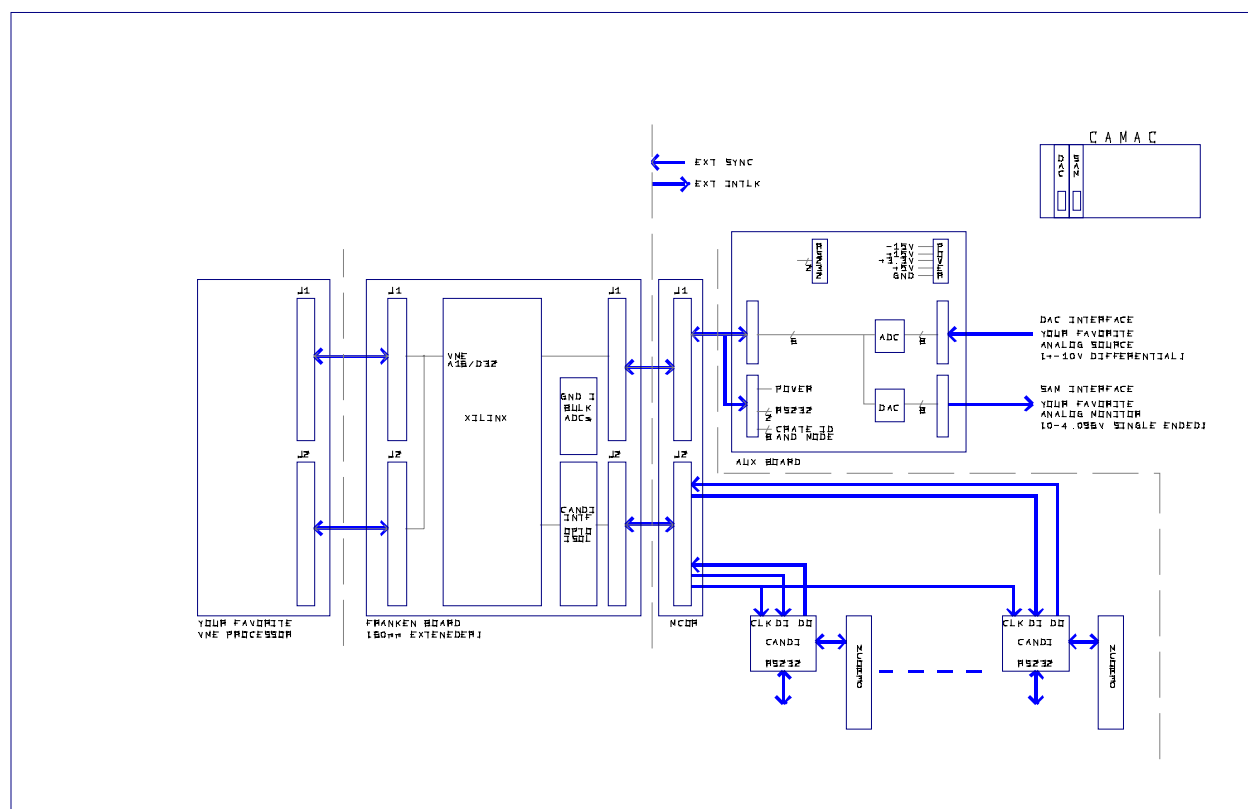
LIST OF TABLES

Table 1 : Crate J1 and J2 Assignment.....	5
Table 2 : Front Panel Fault Display	7
Table 3 : RS232 Pinout.....	8
Table 4 : JTAG Pinout.....	8
Table 5 : I1MON ADC Multiplexer	11
Table 6 : I2MON ADC Multiplexer	12
Table 7 : VMON ADC Multiplexer.....	12
Table 8 : CANDI Control and Status Register (CSR)	13
Table 9 : Serial Command and Response Format.....	14
Table 10 : RS232 Command Format	14
Table 11 : CANDI Commands	14
Table 12 : AUX Board Power Connector Assignment.....	15
Table 13 : AUX Board Configuration Switch	16
Table 14 : FrankenBoard Commands	18
Table 15 : CSR0 -- Franken Board Link Status (Write = 160, Read = 180)	19
Table 16 : CSR1 -- Internal Sync Timer (Write = 164, Read = 184)	19
Table 17 : CSR2 -- System Sync word Low (Read = 188).....	20
Table 18 : CSR3 -- System Sync word High (Read = 18C)	20
Table 19 : CSR4 -- Aux Card Interface and Mode Bits (Write = 170, Read = 190)	21
Table 20 : CSR5 -- Daughter Card Interface (Write = 174, Read = 194).....	22
Table 21 : CSR6 -- Internal Sync Counter ReadBack (Read = 198)	22
Table 22 : FBMON -- Franken Board Voltage Monitor (Read = 1A0 - 1CC)	23
Table 23 : Franken Board Version History.....	24
Table 24 : CANDI Board Version History	25
Table 25 : DAC/SAM Connector Assignment	26
Table 26 : MCOR30 Slot Zero Back-plane Signal Assignment.....	27
Table 27 : MCOR30 Slot Zero Back-plane Signal Assignment.....	28

The MCOR30 system consists of 6 main components:

1.1. MCOR30 System Block Diagram

Figure 1 : MCOR30 System Block Diagram



1.2. MCOR Crate

This crate is the same crate as used by the MCOR12 system. However, instead of using the back-plane for analog signals, all data to and from the slot zero controller are digital in the MCOR30 system (see Table 25, Page 26 for description of the new back-plane signal assignment). The two AMP36 connectors on the crate are used to bring power into the system for the FrankenBoard and the VME Processor as well as digital controls and Crate ID. The two isolated BNC are now used for SYNC input and InterLock out.

Table 1 : Crate J1 and J2 Assignment

Connector	Original Use	Dir.	MCOR30 Use	Dir.
J1	EXT INTLK	IN	Interlock	OUT

J2	CRATE OK	OUT	SYNC	IN
----	----------	-----	------	----

1.3. MCOR30 – Corrector Power Supply

The MCOR30 power supply is a two wide module. It is capable of either analog or digital control. A daughter board is used in either case to connect the power supply to the control interface. This document is only concerned with the digital interface (CANDI board). See separate document for more information on the Power Supply.

1.4. CANDI (Controller ANalog/Digital Interface)

This daughter board plugs onto the MCOR30. It receives digital signals from the MCOR back-plane and converts them to analog signals to control the MCOR30. It also converts analog signals from the MCOR30 to digital signals and transmits them on the back-plane. The CANDI also monitors the status of the MCOR30, controls various multiplexers and faults conditions. There is an RS232 interface to control and monitor the CANDI and MCOR30 operation. This document is only concerned with the digital aspects of the CANDI board. For analog performance information see separate CANDI documentation.

1.5. FrankenBoard

This board plugs into slot zero of the MCOR crate. It is the interface between a standard VME processor and the MCOR back-plane. It accepts VME commands from the processor, and converts them to the serial protocol used on the MCOR back-plane. It also receives data and status from MCOR30 back-plane and makes it available through VME commands.

1.6. AUX Board (Bride of FrankenBoard)

This board plugs onto the DAC/SAM connectors on the rear of the MCOR30 crate. The DAC/SAM connectors on the crate are used to supply power and digital control signals to the FrankenBoard (see for description of the new DAC/SAM signal usage). A reduced resolution DAC/SAM interface is provided by two connectors on the AUX Board. The AUX board supplies a 6 bit crate ID number via a dip switch. An RS232 interface allows control and monitoring of the FrankenBoard and all of the CANDI.

1.7. VME Processor

This can be any standard VME processor. This allows the MCOR30 System to be controlled by any commercial interface standard that the system requires.

2. MCOR30 Feedback Operation

The MCOR30 system is driven by the SYNC pulse. First, the VME Processor updates the Set Point register in all of the CANDI boards. This value is not loaded into the CANDI boards Set Point DAC until the SYNC pulse is received on the back-plane. When the SYNC pulse is

received, all of the CANDI boards load their Set Point DAC's. The CANDI boards also return the current values of I1MON, I2MON, VMON, CSR, Set Point DAC register, and Calibration DAC register. These values are stored in the FrankenBoards data buffer. The SYNC pulse causes the read and write buffer pointers in the FrankenBoard to be swapped. The new data from the CANDI boards is written into the write buffer, and the previous data is made available to VME. The SYNC pulse generates a VME interrupt, which signals the VME processor to read the previous data from the read data buffer, and load new values into the CANDI Set Point registers.

3. CANDI Board

The CANDI (Controller ANalog/Digital Interface) daughter Board is the interface between the MCOR back-plane and the MCOR3. It can be controlled from either a 2MBit serial interface from MCOR back-plane or the 115.2Kbd RS232 interface.

3.1. Front Panel

The CANDI board has several LED's and connectors on its front edge. Due to airflow requirements, a labeled front panel was not designed.

1. Bulk supply ON – Top LED

Yellow LED indicating that the bulk supply voltage is over 30 volts.

2. Bulk supply over voltage – Middle LED

RED LED indicating that the bulk supply is over 60 volts.

3. Fault – Bottom LED

RED LED indicating that there is a fault condition.

4. HEX display

If there is a fault condition, the value of the fault register is displayed. If there is not a fault and IMON1 MUX is set to I1MON the MCOR30 output current times 10 is displayed.

Table 2 : Front Panel Fault Display

BIT	Name	Description
8	FAULT	1 => MCOR Protection Fault
7	BULKON	1 => BULK Supply ON (Over 10V)
6	INHIB	1 => Inhibit asserted on the Backplane
5	PGMBRD	1 => MCOR Programming Module is missing
4	OCFILT	1 => MCOR Filter Current excessive
3	OTSINK	1 => MCOR Heat Sink temperature is excessive
2	OCOUT	1 => MCOR Output Current is excessive

1	BALOUT	1 => MCOR Output Connections do not share similar current levels
0	OVSUPP	1 => MCOR Supply Bus is excessive

3.2. RS232 connector

The RS232 connector is a 9 pin circular connector that provides complete control and monitoring of the CANDI and MCOR.

Table 3 : RS232 Pinout

Pin	Name
1	
2	RX_DATA
3	TX_DATA
4	
5	
6	
7	
8	
9	
CASE	GND VME

5. JTAG connector

The JTAG connector allows for re-programming of the XILINX boot PROM. This connector has power and ground on it.

Table 4 : JTAG Pinout

Pin	Name
1	TDO
2	KEY
3	TDI
4	TCK
5	TMS
6	+5V VME
7	GND VME

3.3. Back Plane Signals

The CANDI board connects to the MCOR back-plane and receives the following signals:

1. SYNC*

The SYNC signal is a bussed signal to all CANDI boards on the back-plane. The leading edge of SYNC causes the CANDI to update the Set Point DAC and transmit the content of all of its registers back to the FrankenBoard.

2. BK_RESET*

The BK_RESET signal is a bussed signal to all CANDI boards on the back-plane. It resets the fault register and the MCOR fault latch only if no faults are currently active. This signal does not reset any other registers in the CANDI and therefore has no effect on a functioning CANDI. This signal is filtered by the 20 MHz system clock and must be at least 200ns wide.

3. INHB*

The INHB signal is a bussed signal to all CANDI boards on the back-plane. It inhibits the MCOR. This signal is filtered by the 20 MHz system clock and must be at least 200ns wide.

4. X_BOOT*

The X_BOOT signal is a bussed signal to all CANDI boards on the back-plane. However, there is a jumper that must be installed on the CANDI for this feature to work. When asserted on the back-plane and the jumper is installed, the XILINX on the CANDI will be re-booted.

5. DATAOUT*

The DATAOUT signal drives data from the CANDI board to the FrankenBoard. Data is clocked out with the CLKIN signal. There is a 2 bit digital filter on this signal.

6. DATAIN*

The DATAIN signal received data from the FrankenBoard. Data is clocked in with the CLKIN signal. There is a 2 bit digital filter on this signal.

7. CLKIN

The CLKIN signal is driven by the FrankenBoard. The CANDI uses this signal to clock data in and out. There is a 2 bit digital filter on this signal. The CLKIN signal is not used directly, but is synchronized to the 20MHZ CANDI system clock.

All of the above signals are driven by the FrankenBoard and are optically isolated from the MCOR back-plane. The FrankenBoard uses the VME +5 and +/-12 volts supplied by the AUX board and is completely isolated from the MCOR +5 and +/- 15 volt supply and the bulk supply.

3.4. MCOR30 Interface Signals

The CANDI is a daughter board that connects to the MCOR30. It uses the following signals:

1. IMON1

IMON1 is an analog input from the MCOR30. This measures the current across a shunt and gives a value of 10V for a full-scale MCOR30 output of 30A. This voltage can be measured by the I1MON ADC.

2. IMON2

IMON2 is an analog input from the MCOR30. This measures the current across a shunt and gives a value of 10V for a full-scale MCOR30 output of 30A. This voltage can be measured by the I2MON ADC.

3. POUT

POUT is an analog input from the MCOR30. This is the value of the Positive rail of the MCOR30 output. This voltage can be measured by the VMON ADC as VMON1. There is a voltage divider of 11/111 on this signal.

4. MOUT

MOUT is an analog input from the MCOR30. This is the value of the Negative rail of the MCOR30 output. This voltage can be measured by the VMON ADC as VMON2. There is a voltage divider of 11/111 on this signal.

5. PGMMOD*

PGMMOD is a digital signal indicating that whether the programming card is installed on the MCOR30. If the programming card is not installed, a fault will be generated and the MCOR30 will be shutdown. There is a 300ms digital filter on PGMMOD.

6. OCFILT*

OCFILT is a digital signal indicating that the excessive current is flowing in the MCOR30 output filter. If asserted, a fault will be generated and the MCOR30 will be shutdown. There is a 300ms digital filter on OCFILT.

7. OTSINK*

OTSINK is a digital signal indicating that the MCOR30 is over temperature. If asserted, a fault will be generated and the MCOR30 will be shutdown. There is a 300ms digital filter on OTSINK.

8. OCOUT*

OCOUT is a digital signal indicating that the MCOR30 output current is too high. If asserted, a fault will be generated and the MCOR30 will be shutdown. There is a 300ms digital filter on OCOUT.

9. BALOUT*

BALOUT is a digital signal indicating that the MCOR30 output current is out of balance. If asserted, a fault will be generated and the MCOR30 will be shutdown. There is a 300ms digital filter on BALOUT.

3.5. Calibration DAC

The Calibration DAC is a Maxim MAX542; a 16 bit high resolution, low drift DAC used to calibrate the CANDI. The DAC output can be connected to any of the three ADC inputs via input multiplexers. The DAC output is scaled to produce a +/-12.5 Volt output. The CANDI accepts a 32 bit signed integer to program the output of the Calibration DAC. The data word for the DAC is calculated as follows:

$$\text{Value} = -1.0 * 65535.0 * (\text{Voltage} + 12.5) / 25.0$$

3.6. Set Point DAC

The Set Point DAC is a Burr-Brown/TI PCM1704U-K; a 24 bit DAC used to set the output current of the MCOR30. The DAC output is scaled to produce a +/-10 Volt output. The CANDI accepts a 32 bit signed integer to program the output of the Set Point DAC. The data word for the DAC is calculated as follows:

$$\text{Value} = 8388607 * \text{Current} / 30$$

3.7. CANDI ADC's

There are three ADC's on the CANDI board. There is a separate oscillator on the CANDI board to control the clock frequency of the ADC. Version 1 of the CANDI firmware is configured to run the ADC's at a 4 KHz conversion rate. This conversion rate was chosen to take advantage of the inherent filter properties of the ADC. This document is concerned with the digital control aspects of the CANDI board, for a more detailed description of the ADC performance, see separate documentation.

3.8. I1MON ADC

The I1MON ADC is a Burr-Brown/TI ADS1251; a 24 bit ADC. The ADC input is scaled to accept +/-12.5V full-scale signal. The I1MON ADC's input is selected by a 4 to 1 Multiplexer controlled by 2 bits in the CANDI CSR.

Table 5 : I1MON ADC Multiplexer

Channel 0	IMON1	Voltage proportional to Current in Shunt 1. 10V/30A
Channel 1	CAL DAC	Output voltage from Calibration DAC
Channel 2	GND	GND voltage
Channel 3	TEMP	$T(^{\circ}\text{C}) = (\text{V}(\text{out}) - 500\text{mV}) / 10\text{mV}$

$$\text{Voltage} = 12.5 * (\text{Value} / 8388607.0)$$

3.9. I2MON ADC

The I2MON ADC is a Burr-Brown/TI ADS1251; a 24 bit ADC. The ADC input is scaled to accept +/-12.5V full-scale signal. The I2MON ADC's input is selected by a 4 to 1 Multiplexer controlled by 2 bits in the CANDI CSR.

Table 6 : I2MON ADC Multiplexer

Channel 0	IMON2	Voltage proportional to Current in Shunt 2. 10V/30A
Channel 1	CAL DAC	Output voltage from the Calibration DAC
Channel 2	GND	GND voltage
Channel 3	SP DAC	Output voltage from the Set Point DAC

Voltage = 12.5 * (Value / 8388607.0)

3.10. VMON ADC

The VMON ADC is a Burr-Brown/TI ADS1251; a 24 bit ADC. The ADC input is scaled to accept +/-12.5V full-scale signal. The VMON ADC's input is selected by a 4 to 1 Multiplexer controlled by 2 bits in the CANDI CSR.

Table 7 : VMON ADC Multiplexer

Channel 0	VMON1	Positive Output voltage from the MCOR30
Channel 1	VMON2	Negative Output voltage from the MCOR30
Channel 2	CAL DAC	Output voltage of Calibration DAC
Channel 3	VMON1-VMON2	Difference between the two MCOR30 outputs

Voltage = 12.5 * (Value / 8388607.0)

The VMON1, VMON2 and VMON1-VMON2 signals have a 11/111 voltage divider on the input since the actual voltage on these signals can be over 70V. Therefore, the voltage read for these signals must be multiplied by 111/11 to get the correct value.

3.11. MCOR30 Faults

The MCOR30 can produce 5 fault conditions that are latched in the CANDI CSR. The fault outputs from the MCOR30 are digitally filtered on the CANDI board. The CANDI board also detects if the bulk supply voltage is too high and will cause a fault. When any fault occurs, the CANDI will latch the fault condition and shutdown the MCOR30. If subsequent faults occur, they will also be latched in the CANDI CSR. The fault latch can only be cleared if there are no faults and a reset is issued either from the back-plane or the RS232 command. If a reset is received and there is no fault, nothing happens on the CANDI board. The following signals can generate a fault:

1. PGMMOD

2. OCFILT
3. OTSINK
4. OCOUT
5. BALOUT
6. OVSUPP
7. INHB

3.12. CANDI CSR

Table 8 : CANDI Control and Status Register (CSR)

Data	Name	Function	Default
[31..29]	Zero	Returns "0000" from CANDI	"0000"
[28..21]	Version	8 Bit CANDI version number	
[20]	RESET	1 => Reset asserted on the Backplane	
[19]	FAULT	1 => MCOR Protection Fault	
[18]	BULKON	1 => BULK Supply ON (Over 10V)	
[17]	INHB	1 => Inhibit asserted on the Backplane	
[16]	PGMBRD	1 => MCOR Programming Module is missing	
[15]	OCFILT	1 => MCOR Filter Current excessive	
[14]	OTSINK	1 => MCOR Heat Sink temperature is excessive	
[13]	OCOUT	1 => MCOR Output Current is excessive	
[12]	BALOUT	1 => MCOR Output Connections do not share similar current levels	
[11]	OVSUPP	1 => MCOR Supply Bus is excessive (over 65V)	
[10..06]	Unused	R/W	
[05..04]	VMONMUX	00 => VMON1 01 => VMON2 10 => CAL DAC 11 => VMON1 - VMON2	"00"
[03..02]	I2MUX	00 => IMON2 01 => CAL DAC 10 => GND 11 => SET POINT DAC	"00"
[01..00]	I1MUX	00 => IMON1 01 => CAL DAC 10 => GND 11 => TEMP	"00"

3.13. CANDI Commands

3.13.1. Serial Interface Operation

The interface between the Franken Board and the CANDI board is a 2 MHz serial interface. 36 bit messages are transferred across the MCOR30 back-plane. The bit serial stream for

commands and responses consists of a START bit, a 4 bit command field, a 32 bit data field, and Odd parity. For read commands, the data field is ignored.

Table 9 : Serial Command and Response Format

Bit	Name	Description
	Start	Always a 1
[35..32]	CMD[03..00]	CANDI Command
[31..00]	DATA[31..00]	DATA
	Parity	Odd parity of Bits[35..00]

The transmit time on the back-plane is:

$$(32 \text{ Data} + 4 \text{ Command} + 1 \text{ Start} + 1 \text{ Parity} + 2 \text{ inter-command}) * 500\text{ns} = 20\mu\text{s/Command}$$

3.13.2. RS232 Interface

The CANDI can be controlled and monitored from the RS232 interface. Data is transferred as a HEX ASCII character string. The RS223 port runs at a fixed rate of 115.2Kbd, Even Parity, 8 Data bits, 1 Stop bit (115.2K, E, 8, 1). The command consists of the header 'FB' followed by a 9 byte hex value for writes or a 1 byte hex value for reads. The string is terminated by a <CR>. The value is preceded by '0X' to signify a hex value.

FB0XCHHHHHHHH<CR>

where:

Table 10 : RS232 Command Format

RS232 Command Format	
FB0X	Header FB for the CANDI Board
C	8 bit command in HEX ASCII
HHHHHHHHH	32 bit data in HEX ASCII (for load commands only)
CR	Carriage return and optionally a Linefeed

3.13.3. CANDI Commands

Table 11 : CANDI Commands

CMD[03..00]	Name	Direction	Data[31..00]
0			Reserved
1	IMUX1	Read	IMUX1 ADC Data as selected in CSR I1MUX
2	IMUX2	Read	IMUX2 ADC Data as selected in CSR I2MUX
3	VMONMUX	Read	VMONMUX Data as selected in CSR MONMUX
4	RCSR	Read	CSR

5	RSP	Read	Read Set Point DAC Register
6	RCAL	Read	Read Calibration DAC Register
7	RESET	CMD	None - Resets Status Latch if no faults exist
8	WCSR	Write	CSR
9	WSP	Write	Write to Set Point DAC Register
10	WCAL	Write	Write to Calibration DAC
[14..11]			Reserved
15	SYNC	CMD	None – Generates a SYNC Command

3.13.4. Write Set Point DAC Command (9)

The write Set Point DAC command does not load the Set Point DAC immediately. Instead, the value is written to the Set Point DAC register. The Set Point DAC register is transferred to the Set Point DAC when the CANDI receives a SYNC pulse from either the back-plane serial interface or the RS232 interface.

3.13.5. SYNC Command (15)

The SYNC command causes the Set Point DAC register value to be transferred to the Set Point DAC. The value of all of the registers is transmitted on the back-plane serial interface. The data is not transmitted on the RS232 interface. The format of this data is as described in Table 9 on page 14. The command field defines which register is being transmitted so the FrankenBoard knows where to store this data.

Note: The SYNC command can also be generated from the MCOR back-plane.

4. AUX Board

The AUX board connects to the DAC and SAM connectors on the back of the MCOR30 crate. These connectors are mapped to the MCOR30 back-plane and on to the P1 connector in slot zero. In the MCOR12, these are analog inputs and outputs to the MCOR30 back-plane. In the MCOR30, these are digital connections to the FrankenBoard.

4.1. Power Connection

The P1 connector on the AUX board brings power to the AUX board, the FrankenBoard and the VME processor. This power is earth grounded and isolated from the MCOR crate power. The voltages are:

Table 12 : AUX Board Power Connector Assignment

Pin	Voltage	Current(max)
1	+5V	3A
2		
3		
4	-12V	500ma
5	GND	

6		
7		
8	+12V	500ma

4.2. AUX Board Configuration Switch

There is an 8 position configuration switch on the AUX board.

Table 13 : AUX Board Configuration Switch

Position	Definition	Description
8	Reset	Unused
7	Mode	‘ON’ => Stand-alone DAC/SAM ‘OFF’ => Normal MCOR30
[06..00]	Crate ID	‘ON’ => ‘0’ ‘OFF’ => ‘1’

4.2.1. Crate ID

A 6 bit crate ID is settable by SW1. This ID number is readable from the FrankenBoard’s CSR 4 (see Table 19 on Page 21). The ‘ON’ position sets the corresponding bit to a ‘ZERO’.

4.2.2. DAC/SAM Mode

The Mode switch on the AUX board allows the AUX board together with the FrankenBoard to be used in a stand-alone ‘DAC/SAM’ mode. The 7th switch position controls this feature. When in the ‘ON’ position, the stand-alone ‘DAC/SAM’ mode is chosen. In the ‘OFF’, normal MCOR30 feedback mode is selected. In this mode, the ADC on the AUX board digitizes the eight analog values on the DAC connector. These values are read-out by the FrankenBoard and transferred to the corresponding CANDI board in the Crate. The I1MON value returned by the CANDI is written to the DAC on the AUX board. This analog voltage is output on the SAM connector on the AUX board. These digitization’s and data transfers are initiated by the SYNC pulse.

The ADC is a 16 bit, +/-12volt full-scale ADC. This means that there is a gain of 10/12 in the data transferred to the CANDI. In order to drive the MCOR30 to a full 30A output, a 12V analog signal is required at the DAC connector.

The DAC is a 10 bit, 0-4volt full-scale DAC. The value written to the DAC is the value digitized by the CANDI board I1MON ADC. The CANDI ADC is +/-12.5 Volts full-scale, but the MCOR30 output for I1MON is only 10V. This means that a full-scale output current of 30A will only produce an output of 3.2 volts. The DAC output therefore becomes 0.8-3.2V full-scale relative to the MCOR30 Current of +/- 30A

4.2.3. Reset Switch

This is currently an unused input to the FrankenBoard XILINX.

4.3. RS232 Connector

The RS232 connector on the AUX board is connected through the back-plane to an RS232 transceiver on the FrankenBoard. This interface provides control and monitoring of the MCOR30 system through RS232. This interface is described in under the FrankenBoard section of this document (see xxx).

5. FrankenBoard

5.1. VME Address and Data Interface

The FrankenBoard is an A24/D32 module. All registers are 32 bits wide only. The FrankenBoard will not respond to D16 or D8 cycles. The FrankenBoard VME interface only responds to the following A24 VME address cycles:

Address Mod	Address Bits	Description
39	24	A24 non-privileged data access
3B	24	A24 non-privileged block transfer (BLT)
3D	24	A24 supervisory data access
3F	24	A24 supervisory block transfer (BLT)

Note: The original design used A16 address cycles, however, A16 does not support block transfers. Although the FrankenBoard responds to A24 address cycles, it only decodes the lower 16 bits. This means that the FrankenBoard appears 256 times in A24 space and uses all of A24 space. Since the FrankenBoard is the only device allowed on this single slot VME interface, this should not be a problem.

5.2. FrankenBoard Interrupts

The FrankenBoard is a ROAK (Release-on-AcKnowledge) interrupter. The FrankenBoard generates an interrupt when it generates a system SYNC pulse from any of the three SYNC sources. The FrankenBoard uses interrupt level 2, IRQ2. In response to the interrupt acknowledge cycle, the FrankenBoard returns a vector of 0x00000001 and clears the interrupt.

5.3. FrankenBoard to VME Memory Map

All of the data in the FrankenBoard is memory mapped to VME and available to the RS232 interface. All of VME memory addresses are either read or write, but not read and write as might be expected. (This may get fixed in a future version of the FrankenBoard code.) The read data is double buffered in the FrankenBoard. The buffers are swapped upon receipt of a SYNC. Each CANDI board has a 16 deep FIFO in its output channel on the FrankenBoard.

Data written to a CANDI board is put into this FIFO and sent out serially to the CANDI board.

Table 14 : FrankenBoard Commands

RS232		VME			
CMD	ADDR	A[9:0]	Name	DIR	Data[31..00]
0	0-F				Reserved
1	0-7	000-01C	I1MUX	Read	CANDI IMUX1[7:0] From Local RAM
2	0-7	020-03C	I2MUX	Read	CANDI IMUX2[7:0] From Local RAM
3	0-7	040-05C	VMONMUX	Read	CANDI VMONMUX[7:0] From Local RAM
4	0-7	060-07C	RCSR	Read	CANDI CSR[7:0] From Local RAM
5	0-7	080-09C	RSP	Read	CANDI Set Point DAC[7:0] From Local RAM
6	0-7	0A0-0BC	RCAL	Read	CANDI Cal DAC[7:0] From Local RAM
7	0-7	0C0-0DC	RESET	CMD	CANDI Reset Status Latch[7:0]
8	0-7	0E0-0FC	WCSR	Write	CANDI CSR[7:0]
9	0-7	100-11C	WSP	Write	CANDI Set Point DAC[7:0]
A	0-7	120-13C	WCAL	Write	CANDI Calibration DAC[7:0]
B	0-7	140-15C	REFRESH	Write	Send Read Command to CANDI to refresh Local RAM CANDI command is 4 LSB's
C	0-7	160-17C	WFBCSR	Write	FrankenBoard CSR[7:0] (Currently VME only write)
D	0-7	180-19C	RFBCSR	Read	FrankenBoard CSR[7:0]
E	0-F	1A0-1DC	FBMON	Read	FrankenBoard Voltages[15:0]
F	0	1E0	VMESYNC	Write	

5.4. Read CANDI Board data

Memory locations 0x000-0x0BC contain the values that were transmitted from the CANDI to the FrankenBoard. This memory is double buffered. The new data transmitted by the CANDI due to the SYNC is written into the write buffer and the data previously stored into the write buffer is made available to the VME read buffer. The pointer is swapped upon receipt of a SYNC.

5.5. Write CANDI CSR

Memory locations 0x0E0-0x13C allows VME to write to each of the eight CANDI boards. This data is put into a 16 deep FIFO before being sent serially out to each CANDI. The FIFO

controls signals, full and empty, are not readable anywhere. It is assumed that all data for a CANDI will be transmitted between SYNC's.

5.6. Refresh Command

Memory locations 0x140-0x15C allows VME to issue a refresh command to the CANDI. This will cause the CANDI to return the current value in the requested register. The new data will be written into the FrankenBoard's write memory. In order to make this new value available to the VME read buffer, a SYNC must be issued. This SYNC will in turn cause all of the CANDI data to be refreshed. The refresh command was implemented before the functionality of SYNC was fully defined.

5.7. Write FrankenBoard CSR

Memory locations 0x160-17C write to the FrankenBoard's internal registers. These registers control the functionality of the system.

5.8. CSR0 -- Franken Board Link Status

CSR0 contains the time out and parity error flags for each of the CANDI links. If a link generates a time out or a parity error, the contents of the FrankenBoard write memory is not updated. The values for a CANDI board will appear to be constant if there are link errors. The link error flags should be checked to be sure that the data in the FrankenBoard memory is valid. The flag bits are set by an error on the link, and cleared when the register is read.

Table 15 : CSR0 -- Franken Board Link Status (Write = 160, Read = 180)

Bits	Name	Dir	Definition
[32..24]	Version	Read	FirmWare Version
[23..16]	Unused	---	Unused
[15..08]	Perror	Read	Link Parity Error
[07..00]	TimeOut	Read	Link Timeout Bits

5.9. CSR1 -- Internal Sync Timer

This register controls the period of the internally generated SYNC Pulse. In the internal SYNC mode, selected by the SYNC_MODE field in CSR4 (see Table 19 on page 21) the value of the SYNC register is loaded into the SYNC counter, which is then decremented at the system clock of 20Mhz. When the counter reaches zero, the counter is reloaded, and a SYNC pulse is generated.

In the external and VME SYNC modes, the counter is reloaded when ever the external or VME SYNC is generated. The value of the counter can be read in CSR 6. This way, the SYNC counter can be used to time the period from a SYNC pulse to some VME event.

Table 16 : CSR1 -- Internal Sync Timer (Write = 164, Read = 184)

Bits	Name	Dir	Definition
[31..24]	Unused	R	
[23..00]	INT_SYNC	R/W	Internal Sync Pulse delay

Period = Value * 50ns

5.10. CSR2 and CSR3 System Sync Word

The system Sync word is a value distributed over the external sync input. The external SYNC input can have the format of an RS232 command. This includes a start bit which is used as the actual sync. The rest of the command is decoded to 7 controls bits readable in CSR6 (see Table 21 on page 22). Bit 8 of the command is shifted into a 64 bit shift-register. Logic in the FrankenBoard determines when to interpret these bits as the 64 bit sync word and stores the value into CSR2 and CSR3. This allows for a system wide distribution of a control word and a sync word.

Table 17 : CSR2 -- System Sync word Low (Read = 188)

Bits	Name	Dir	Definition
[31..00]	SyncLow	R	Low 32 bits of the System Sync word

Table 18 : CSR3 -- System Sync word High (Read = 18C)

Bits	Name	Dir	Definition
[31..00]	SyncHigh	R	High 32 bits of the System Sync word

5.11. CSR4 -- Aux Card Interface and Mode Bits

5.11.1. AUX board settings

CSR 4 contains the setting of the 8 switches on the AUX board (see AUX Board Configuration Switch on page 5).

5.11.2. Ground Fault

CSR4 also contains the ground fault status flag. This bit is the output of a comparator that checks the voltage across the ground fault resistor in the MCOR30 crate. The value is undefined at this time.

5.11.3. SYNC Mode

The SYNC mode bit determines which source of SYNC will be used. In the Internal SYNC mode, the SYNC counter in CSR1 is used to control the period of SYNC. In the External mode, the SYNC from the MCOR30 crate is used. In CPU SYNC, the SYNC generated by the VME cycle or RS232 cycle is used.

5.11.4. Data Mode

The Data mode bit determines what data is sent to the CANDI on a VME or RS232 command to write to the Set Point DAC. When in the DAC/SAM mode, only data from the AUX board DAC/SAM interface is written to the CANDI. In the VME mode, only the VME or RS232 data is written. In the SUM mode, the DAC/SAM data is added to the VME or RS232 data before being sent to the CANDI. The MODE switch on the AUX board over-rides this setting and forces the DAC/SAM mode.

Data from the DAC/SAM interface is double buffered in the FrankenBoard. This buffer is controlled by the SYNC. The SYNC pulse starts a new conversion cycle from the DAC/SAM interface, putting the new data into the write buffer and making the previous data available for use in the FrankenBoard.

Table 19 : CSR4 -- Aux Card Interface and Mode Bits (Write = 170, Read = 190)

Bits	Name	Dir	Definition
[26..11]	DATA_MODE	R/W	"11" => Unused "10" => Sum Mode "01" => VME Mode "00" => DACSAM Mode
[10..09]	SYNC_MODE	R/W	"11" => Unused "10" => Internal SYNC "01" => External SYNC "00" => CPU SYNC
08	EXT_RESET	R	Reset Bit from AUX Board Switch
07	GFAULT	R	Ground Fault Status
06	EXT_MODE	R	Mode Bit from AUX Board Switch "0" => VME CPU Mode "1" => DAC/SAM Mode
[05..00]	ID	R	ID from AUX Board Switch

5.12. CSR5 – MCOR Back-plane Interface

Table 20 : CSR5 -- Daughter Card Interface (Write = 174, Read = 194)

Bits	Name	Dir	Definition
[31..20]	Unused		
19	DBRESET_RESET	W	Reset RESET on CANDI
18	INH_RESET	W	Reset INH on CANDI
17	XBOOT_RESET	W	Reset XBoot on CANDI
16	INTLK_RESET	W	Reset INTLK Crate output
[15..07]	Unused		
08	AUX_ADC_TO	R	Auxiliary Board ADC failed to Calibrate
07	FB_ADC_TO	R	Franken Board ADC failed to Calibrate
06	ADCCAL	R W	Calibration in progress Start ADC Calibration (4ms)
05	CHIPRESET	W	Reset Internal registers
04	SYSRESET	R/W	Set VME SYSReset
03	DBRESET_SET	R/W	Set RESET on CANDI
02	INH_SET	R/W	Set INH on CANDI
01	XBOOT_SET	R/W	Set XBoot on CANDI
00	INTLK_SET	R/W	Set INTLK Crate output

Table 21 : CSR6 -- Internal Sync Counter ReadBack (Read = 198)

Bits	Name	Dir	Definition
[31]	Unused	R	
[30..24]	CTRL_WRD	R	System Control word *(not implemented in Version 1.0)
[23..00]	INT_COUNT	R	Internal Sync Count

5.13. FrankenBoard Voltage Monitor

The FrankenBoard as a MAX1132 16 bit bi-polar ADC with a full-scale input voltage of 12V. This ADC digitizes 10 on board voltages which can then be readout.

5.13.1. Ground sense voltage

The ground sense voltage is the voltage across a 15ohm resistor connected between the chassis ground and the MCOR30 bulk supply ground. This signal is buffered and multiplied by 2 before the ADC.

5.13.2. MCOR Bulk Supply

The MCOR bulk supply voltage is connected through a 1.5k resistor from the MCOR back-plane. On the FrankenBoard, this is further divided by a 61.9K over 10K which gives a final 1/7.34 voltage gain.

Table 22 : FBMON -- Franken Board Voltage Monitor (Read = 1A0 - 1CC)

Address	Name	Definition	Conversion scale factor
1A0	GSENSE	Ground Sense Voltage	½
1A4	+15VMCOR	MCOR Crate +15V	2
1A8	+12V	VME Crate +12V	2
1AC	PHVDC	MCOR Bulk Supply	7.34
1B0	-15VMCOR	MCOR Crate -15V	2
1B4	-12V	VME Crate -12V	2
1B8	VDDMCOR	MCOR Crate +5V	1
1BC	VDD	VME Crate +5V	1
1C0	+3.3V	Xilinx +3.3V	1
1C4	+2.5V	Xilinx +2.5V	1
1CC	Unused		
1C8	Unused		
1D0	Unused		
1D4	Unused		
1D8	Unused		
1DC	Unused		

6. Firmware Modifications and Versions

6.1. Franken Board Version History

Table 23 : Franken Board Version History

Version	Date	Name	Description	Files
fb_xV02	09/13/02	jjo	Fixed Bugs in Block transfer	sch/top sch/vme_intf vhdl/fbseq
			Fixed Bugs in interrupt	sch/vmeirq
			Add SLINK Register	vhdl/serial_tx
			VME Interrupt Cycle respond to D08, D16, D32	sch/vmeirq
	09/25/02	jjo	Remove BUSY	frankenboard.doc
			Remove SLINK Register, Add FIFO	vhdl/serial_tx
			Changed from CLB RAM to BLOCK RAM	
fb_xV02a	10/10/02	jjo	Added DAC/SAM Not Tested yet	vhdl/dacsamseq vhdl/fbcsr
			Fixed Interrupt Problem	sch/vme_intf sch/vmeirq
fb_xV02b	10/11/02	jjo	Fixed RS232	vhdl/fbseq vhdl/receive
fb_xV03	10/16/02	jjo	Fixed Address error in VME Sync - Moved from 1D0 to 1E0	vhdl/fbseq
			Changed to VME Interrupt level IRQ2 since NI board uses IRQ1	sch/vmeirq
Fb_xV1.0	01/13/03	jjo	Reset Internal Sync count on External sync if not in Internal Mode	sync.vhd
	02/03/03	jjo	Changed Scale for DAC SAM interface	slinkio.l
	02/03/03	jjo	Added timeout for ADC Cal	dacsamseq.vhd dacsamseq.l fbcsr.vhd fbcsr.l top.l
	02/12/03	jjo	Fixed Default DAC/SAM Mode (No AUX) to CPU Added Pullup and INV to mode	fb_x.l

6.2. CANDI Board Version History

Table 24 : CANDI Board Version History

Version	Date	Name	Description	Files
V0.3	08/22/02	jjjo	Swapped SLINKIN and SLINKOUT	control_x.ucf
V0.4	09/27/02	jjjo	Fix SYNC so that ALL data is returned on SYNC	
			Add FIFO to SLINKOUT	
			Fix SP_DAC interface to obey Sync	
			Clear SP_DAC on powerup	
			Renamed Xilinx Project to DB_X	
V0.5	12/03/02	jjjo	Added RS232 SYNC command	seq.vhd
V0.5a	12/04/02	jjjo	Fix CAL_DAC to powerup at 0V	max542drv.vhd
	01/28/03	jjjo	Fixed SYNC bug	pcm1704drv.vhd
	01/82/03	jjjo	Un Inverted PGM BRD signal	db_x.1
	01/28/03	jjjo	Divide ADC clk by 4	db_x.1
V0.6	01/28/03	jjjo	Remove 16 average boxcar filter from ADC	adc.1
	01/28/03	jjjo	Changed 15HZ to an enable	protect.vhd dfilter4.vhd dstretch.vhd pkg.vhd
V1.0	02/05/03	jjjo	Fixed Display Scale	

7. MCOR30 DAC/SAM Connector Assignment

Table 25 : DAC/SAM Connector Assignment

Pin	Name	Function	Pin	Name	Function	Pin	Name	Function
A1	DAC-A1	DB15-1	B1	DAC-B1	DB15-13	C1	DAC-C1	TDO
A2	DAC-A2	DB15-2	B2	DAC-B2	DB15-14	C2	DAC-C2	
A3	DAC-A3	DB15-3	B3	DAC-B3	DB15-15	C3	DAC-C3	TDI
A4	DAC-A4	DB15-4	B4	DAC-B4	DIN (Xilinx Out)	C4	DAC-C4	TCK
A5	DAC-A5	DB15-5	B5	DAC-B5	SSTRB	C5	DAC-C5	TMS
A6	DAC-A6	DB15-6	B6	DAC-B6	DOUT (Xilinx In)	C6	DAC-C6	
A7	DAC-A7	DB15-7	B7	DAC-B7	SCLK	C7	DAC-C7	
A8	DAC-A8	DB15-8	B8	DAC-B8	CS*	C8	DAC-C8	
A9	DAC-A9	DB15-9	B9	DAC-B9	FS	C9	DAC-C9	
A10	DAC-A10	DB15-10	B10	DAC-B10		C10	DAC-C10	
A11	DAC-A11	DB15-11	B11	DAC-B11		C11	DAC-C11	
A12	DAC-A12	DB15-12	B12	DAC-B12		C12	DAC-C12	
A13	SAM-A1	TX	B13	SAM-B1	VDDVME	C13	SAM-C1	ID0
A14	SAM-A2	RX	B14	SAM-B2	VDDVME	C14	SAM-C2	ID1
A15	SAM-A3	GNDVME	B15	SAM-B3	VDDVME	C15	SAM-C3	ID2
A16	SAM-A4	GNDVME	B16	SAM-B4	VDDVME	C16	SAM-C4	ID3
A17	SAM-A5	GNDVME	B17	SAM-B5	VDDVME	C17	SAM-C5	ID4
A18	SAM-A6	GNDVME	B18	SAM-B6	VDDVME	C18	SAM-C6	ID5
A19	SAM-A7	GNDVME	B19	SAM-B7	VDDVME	C19	SAM-C7	MODE
A20	SAM-A8	GNDVME	B20	SAM-B8	VDDVME	C20	SAM-C8	RESET
A21	SAM-A9	GNDVME	B21	SAM-B9	P12VVME	C21	SAM-C9	
A22	SAM-A10	GNDVME	B22	SAM-B10	P12VVME	C22	SAM-C10	
A23	SAM-A11		B23	SAM-B11	M12VVME	C23	SAM-C11	
A24	SAM-A12		B24	SAM-B12	M12VVME	C24	SAM-C12	
A25	PHVDC	PHVDC	B25	GNDMCOR	GNDMCOR	C25	CHASSIS	GNDVME
A26	INTLK-C	INTLK-C	B26	OK-C	SYNC-C	C26	ACSYNC	
A27	INTLK-R	INTLK-R	B27	OK-R	SYNC-R	C27	ACSYNC	
A28	VDDMCOR	VDDMCOR	B28	VDDMCOR	VDDMCOR	C28	VDDMCOR	
A29	GNDMCOR	GNDMCOR	B29	GNDMCOR	GNDMCOR	C29	GNDMCOR	
A30	P15VMCOR	P15VMCOR	B30	P15VMCOR	P15VMCOR	C30	P15VMCOR	
A31	M15VMCOR	M15VMCOR	B31	M15VMCOR	M15VMCOR	C31	M15VMCOR	
A32	SYNC	SYNC*	B32	INH	INH*	C32	B_RESET	B_RESET*

Note: Xilinx JTAG interface (TDO, TDI, TCK, TMS) are misconnected between the FrankenBoard and the AUX Board! Only the FrankenBoard connector is correct.

8. MCOR30 Slot Zero Back-plane Signal Assignment (P1)

Table 26 : MCOR30 Slot Zero Back-plane Signal Assignment

Pin	Name	Pin	Name	Pin	Name
A1		B1		C1	
A2		B2		C2	
A3		B3		C3	
A4		B4	DIN	C4	
A5		B5	SSTRB	C5	
A6		B6	DOUT	C6	
A7		B7	SCLK	C7	
A8		B8	CS*	C8	
A9		B9	FS	C9	
A10		B10		C10	
A11		B11		C11	
A12		B12		C12	
A13	RS232_TX*	B13	VDDVME	C13	ID0
A14	RS232_RX*	B14	VDDVME	C14	ID1
A15	GNDVME	B15	VDDVME	C15	ID2
A16	GNDVME	B16	VDDVME	C16	ID3
A17	GNDVME	B17	VDDVME	C17	ID4
A18	GNDVME	B18	VDDVME	C18	ID5
A19	GNDVME	B19	VDDVME	C19	MODE
A20	GNDVME	B20	VDDVME	C20	EX_RESET
A21	GNDVME	B21	P12VVME	C21	TDO
A22	GNDVME	B22	P12VVME	C22	TDI
A23		B23	M12VVME	C23	TCK
A24		B24	M12VVME	C24	TMS
A25	PHVDC	B25	MGNDSSENSE	C25	PGNDSSENSE
A26	INTLK_C	B26	SYNC_C	C26	ACSYNC1
A27	INTLK_R	B27	SYNC_R	C27	ACSYNC2
A28	VDDMCOR	B28	VDDMCOR	C28	VDDMCOR
A29	GNDMCOR	B29	GNDMCOR	C29	GNDMCOR
A30	P15VMCOR	B30	P15VMCOR	C30	P15VMCOR
A31	M15VMCOR	B31	M15VMCOR	C31	M15VMCOR
A32	SYNC*	B32	INH*	C32	RESET*

9. MCOR30 Slot Zero Back-plane Signal Usage (P2)

Table 27 : MCOR30 Slot Zero Back-plane Signal Assignment

Pin	Name	Pin	Name	Pin	Name
A1		B1		C1	
A2		B2		C2	
A3	CLK0	B3	CLK2	C3	CLK4
A4	DATAIN0*	B4	DATAIN2*	C4	DATAIN4*
A5	XBOOT*	B5	XBOOT*	C5	XBOOT*
A6	DATAOUT0*	B6	DATAOUT2*	C6	DATAOUT4*
A7		B7		C7	
A8		B8		C8	
A9		B9		C9	
A10		B10		C10	
A11		B11		C11	
A12		B12		C12	
A13		B13		C13	
A14		B14		C14	
A15	CLK1	B15	CLK3	C15	CLK5
A16	DATAIN1*	B16	DATAIN3*	C16	DATAIN5*
A17	XBOOT*	B17	XBOOT*	C17	XBOOT*
A18	DATAOUT1*	B18	DATAOUT3*	C18	DATAOUT5*
A19		B19		C19	
A20		B20		C20	
A21		B21		C21	
A22		B22		C22	
A23		B23		C23	
A24		B24		C24	
A25		B25	DATAIN6*	C25	CLK6
A26		B26	DATAOUT6*	C26	XBOOT*
A27		B27		C27	
A28		B28		C28	
A29		B29		C29	
A30		B30		C30	
A31		B31	DATAIN7*	C31	CLK7
A32		B32	DATAOUT7*	C32	XBOOT*

