

# **PMC-AD8**

**Intelligent DSP Based  
8 Channel Analog Input Card  
for *PMC* systems**

## **REFERENCE MANUAL**

771-13-000-4000

Version 1.3

June 2003

**ALPHI TECHNOLOGY CORPORATION**

**6202 S. Maple Avenue #120**

**Tempe, AZ 85283 USA**

**Tel: (480) 838-2428**

**Fax: (480) 838-4477**

**NOTICE**

The information in this document has been carefully checked and is believed to be entirely reliable. While all reasonable efforts to ensure accuracy have been taken in the preparation of this manual, ALPHI TECHNOLOGY assumes no responsibility resulting from omissions or errors in this manual, or from the use of information contain herein.

ALPHI TECHNOLOGY reserves the right to make any changes, without notice, to this or any of ALPHI TECHNOLOGY's products to improve reliability, performance, function or design.

ALPHI TECHNOLOGY does not assume any liability arising out of the application or use of any product or circuit described herein; nor does ALPHI TECHNOLOGY convey any license under its patent rights or the rights of others.

**ALPHI TECHNOLOGY CORPORATION**

**All Rights Reserved**

This document shall not be duplicated, nor its contents used for any purpose, unless express permission has been granted in advance.

**TABLE OF CONTENTS**

---

<b><u>1.</u></b>	<b><u>GENERAL DESCRIPTION</u></b>	<b><u>1</u></b>
<b><u>1.1</u></b>	<b><u>INTRODUCTION</u></b>	<b><u>1</u></b>
<b><u>1.2</u></b>	<b><u>FUNCTIONAL DESCRIPTION</u></b>	<b><u>1</u></b>
<b><u>1.2.1</u></b>	<b><u>CLOCK / TRIGGER SOURCES</u></b>	<b><u>3</u></b>
<b><u>1.2.2</u></b>	<b><u>CLOCK / TRIGGER TARGETS</u></b>	<b><u>4</u></b>
<b><u>1.3</u></b>	<b><u>SOFTWARE SUPPORT</u></b>	<b><u>4</u></b>
<b><u>1.4</u></b>	<b><u>REFERENCE MATERIALS LIST</u></b>	<b><u>4</u></b>
<b><u>2.</u></b>	<b><u>PMC BRIDGE TO LOCAL DSP BUS</u></b>	<b><u>6</u></b>
<b><u>2.1</u></b>	<b><u>INTERFACE TO HOST (PMC)</u></b>	<b><u>6</u></b>
<b><u>2.2</u></b>	<b><u>PMC CONFIGURATION SPACE</u></b>	<b><u>6</u></b>
<b><u>2.3</u></b>	<b><u>PMC BASE ADDRESS REGIONS</u></b>	<b><u>8</u></b>
<b><u>3.</u></b>	<b><u>LOCAL DSP BUS</u></b>	<b><u>9</u></b>
<b><u>3.1</u></b>	<b><u>LOCAL SRAM</u></b>	<b><u>10</u></b>
<b><u>3.2</u></b>	<b><u>CARD CONTROL / STATUS REGISTERS</u></b>	<b><u>10</u></b>
<b><u>3.2.1</u></b>	<b><u>9080 CONTROL / STATUS REGISTER</u></b>	<b><u>11</u></b>
<b><u>3.2.2</u></b>	<b><u>HOST INTERRUPT ACKNOWLEDGE</u></b>	<b><u>12</u></b>
<b><u>3.2.3</u></b>	<b><u>BOARD CONTROL / STATUS REGISTER</u></b>	<b><u>12</u></b>
<b><u>3.2.4</u></b>	<b><u>RESET THE HARDWARE FIFO</u></b>	<b><u>13</u></b>
<b><u>3.2.5</u></b>	<b><u>READ THE HARDWARE FIFO</u></b>	<b><u>13</u></b>
<b><u>3.2.6</u></b>	<b><u>WRITE THE HARDWARE FIFO FOR TESTING AND CONFIGURATION</u></b>	<b><u>13</u></b>
<b><u>3.2.7</u></b>	<b><u>PGA GAIN REGISTERS</u></b>	<b><u>13</u></b>
<b><u>3.2.8</u></b>	<b><u>SOFTWARE STROBE FOR MANUAL CLOCKING BY DSP</u></b>	<b><u>14</u></b>
<b><u>3.2.9</u></b>	<b><u>TRIGGER / GATE CONFIGURATION REGISTERS</u></b>	<b><u>14</u></b>
<b><u>3.2.10</u></b>	<b><u>PROGRAMMING PMC-AD8 (72285) FIFO</u></b>	<b><u>15</u></b>
<b><u>3.2.11</u></b>	<b><u>INTERNAL SERIAL PORT</u></b>	<b><u>17</u></b>
<b><u>3.2.12</u></b>	<b><u>SOFTWARE STROBE FOR MANUAL CLOCKING BY HOST</u></b>	<b><u>18</u></b>
<b><u>3.3</u></b>	<b><u>LOCAL FLASH</u></b>	<b><u>18</u></b>
<b><u>3.4</u></b>	<b><u>PLX OPERATION REGISTERS</u></b>	<b><u>19</u></b>
<b><u>3.4.1</u></b>	<b><u>LOCAL CONFIGURATION REGISTERS</u></b>	<b><u>19</u></b>
<b><u>3.4.2</u></b>	<b><u>RUNTIME REGISTERS</u></b>	<b><u>20</u></b>
<b><u>3.4.3</u></b>	<b><u>DMA REGISTERS</u></b>	<b><u>20</u></b>
<b><u>3.5</u></b>	<b><u>PMC PASS-THROUGH REGION</u></b>	<b><u>21</u></b>
<b><u>4.</u></b>	<b><u>HARDWARE DETAILS</u></b>	<b><u>22</u></b>
<b><u>4.1</u></b>	<b><u>RESET SIGNALS</u></b>	<b><u>22</u></b>
<b><u>4.2</u></b>	<b><u>LOCAL DSP INTERRUPT SOURCES</u></b>	<b><u>22</u></b>
<b><u>4.3</u></b>	<b><u>HOST INTERRUPT SOURCE</u></b>	<b><u>22</u></b>
<b><u>4.4</u></b>	<b><u>CONNECTORS, JUMPERS, AND LEDS</u></b>	<b><u>23</u></b>
<b><u>4.4.1</u></b>	<b><u>JUMPER DESCRIPTIONS</u></b>	<b><u>23</u></b>
<b><u>4.4.2</u></b>	<b><u>CONNECTOR DESCRIPTIONS</u></b>	<b><u>23</u></b>
<b><u>4.4.3</u></b>	<b><u>LEDS</u></b>	<b><u>26</u></b>
<b><u>5.</u></b>	<b><u>SPECIFICATIONS</u></b>	<b><u>27</u></b>

## **CPCI-AD8 HARDWARE REFERENCE MANUAL**

---

<b><u>5.1</u></b>	<b><u>INPUTS</u></b>	<b><u>27</u></b>
<b><u>5.2</u></b>	<b><u>TRIGGERS / CLOCKS</u></b>	<b><u>27</u></b>
<b><u>5.3</u></b>	<b><u>DSP</u></b>	<b><u>29</u></b>
<b><u>5.4</u></b>	<b><u>HOST INTERFACE</u></b>	<b><u>29</u></b>
<b><u>5.5</u></b>	<b><u>CONNECTIONS</u></b>	<b><u>29</u></b>
<b><u>5.6</u></b>	<b><u>POWER</u></b>	<b><u>29</u></b>

## 1. GENERAL DESCRIPTION

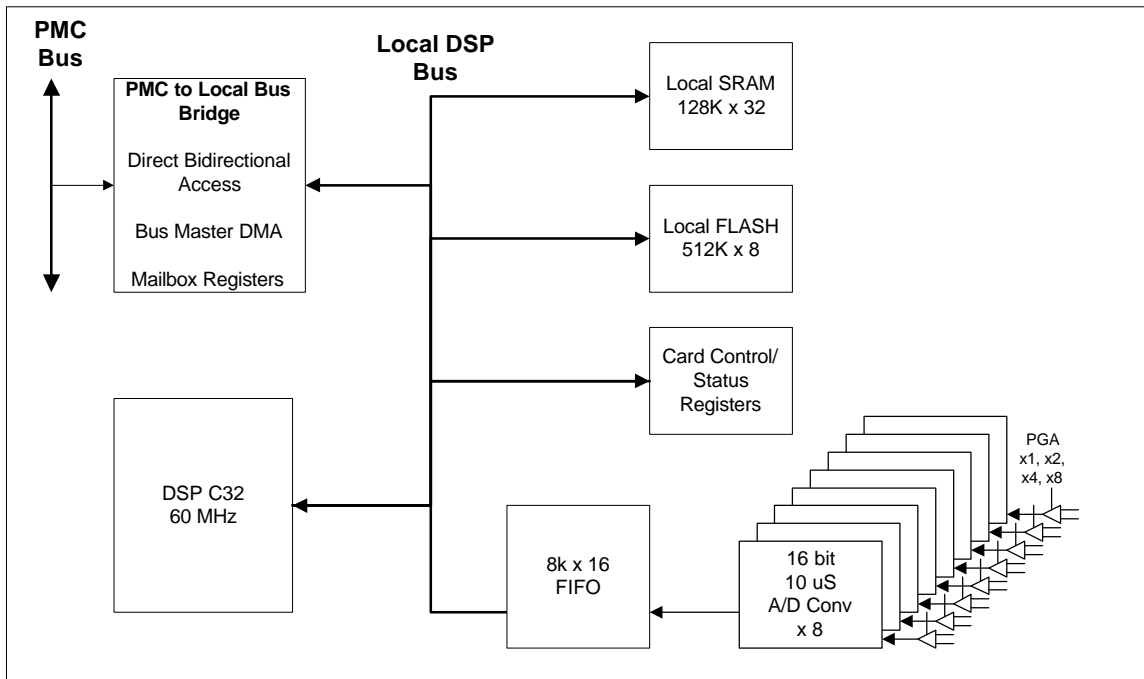
### 1.1 INTRODUCTION

The **PMC-AD8** is an intelligent DSP based 8 channel analog input card in a PMC form factor. The **PMC-AD8** provides 8 differential inputs to individual 16 bit 2us converters. The DSP allows for the offloading from the HOST CPU any necessary I/O processing to monitor and service these channels. The primary features of the **PMC-AD8** are as follows:

- 8 Channels of simultaneous 16 bit A/D acquisition.
- Integrated DSP (TI TMS320C32) at 60 MHz to offload I/O operation from host.
- 128K x 32 bit zero wait state SRAM for the DSP.
- 512K by 8 bit FLASH device for bootloader and customer applications.
- Programmable capture rates up to 100 KHz. Optionally up to 200 kHz.
- Programmable PGAs allow for gains of 1, 2, 4, or 8 on a per channel basis. Gains of 1, 10, 100, and 1000 are optional, with a drop in input bandwidth.
- Extremely flexible triggering, including internal, external.
- Bus Master DMA to offload HOST processor.
- Fully shared HOST access to SRAM, and DSP access to HOST memory.
- Full software support including DSP code and Drivers/DLL for WinNT.

### 1.2 FUNCTIONAL DESCRIPTION

A data flow block diagram of the **PMC-AD8** is presented in Figure 1-1.



**Figure 1.1: Data Flow Block Diagram**

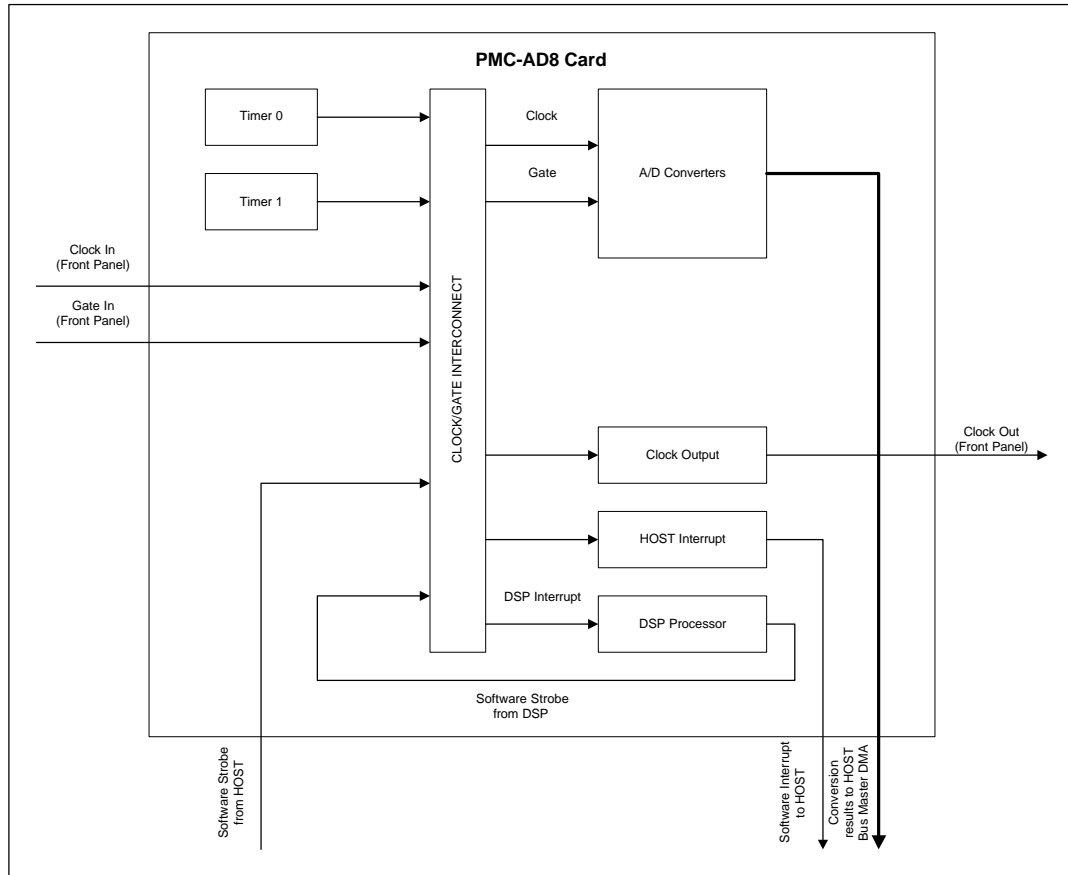
Each of the input channels has a Programmable Gain Amplifier (PGA). Inputs are differential mode, and the gains can be selected under DSP control. Depending on factory input option, the gains are x1, x2, x4, and x8, or optionally x1, x10, x100, and x1000. Individual channels can have different input options. Note that the X10, X100, and X1000 option will greatly reduce the input bandwidth, depending upon the gain selected. Contact the factory for more details.

Upon receipt of a trigger signal, the A/D converters sample the PGA output voltages and start a conversion. When the conversions are complete, a hardware state machine copies the conversion results to a hardware FIFO.

When a programmable number of samples are available in the FIFO, a DSP interrupt is generated. The DSP then has the samples available for further processing and transmission to the HOST via Bus Master DMA. The DSP code and NT driver, provided in the Board Support Package, supports full scatter / gather type DMA to the HOST.

There is full bi-directional access through the PMC to Local Bus bridge. The DSP can directly address and access memory on the HOST processor in addition to DMA. The HOST can directly access resources on the Local DSP bus.

The **PMC-AD8** has extremely flexible triggering. The triggering clock / gate block diagram is presented in Figure 1-2.



**Figure 1.2: Trigger Clock / Gate Signal Block Diagram**

It is easier to describe the sources and targets separately. Any source can be connected to any target.

**1.2.1 CLOCK / TRIGGER SOURCES**

**DSP Timers**

There are two timers internal to the DSP which are capable of generating clock outputs. The outputs can be selected between positive or negative pulses, and square wave. Periods are multiples of 66.7 nS.

**External TTL inputs**

There are two inputs on the front panel to the 50-pin connector. The inputs are TTL compatible.

**Trigger by DSP**

The DSP can trigger a pulse by writing to a location in the registers.

**Trigger by HOST**

The HOST can trigger a pulse by writing to a location in the registers.

**Discrete Logic Level**

A forced 0 or 1 can be used as a source. This may be useful for the A/D Gate if the Gate functionality is not desired.

### **1.2.2 CLOCK / TRIGGER TARGETS**

These targets can be connected to any of the above sources.

#### **A/D Converters**

The A/D converters are controlled by two signals: Clock and Gate. A rising edge on Clock will trigger the start of a conversion. Gate serves as a Clock qualifier, and must be high for Clock to be recognized.

#### **External Clock Output**

There is an output on the front panel to the 50-pin connector. The output is TTL compatible.

#### **HOST Interrupt**

The HOST can be interrupted on a rising edge of the source.

#### **DSP Interrupt**

The DSP can be interrupted on a rising edge of the source.

## **1.3 SOFTWARE SUPPORT**

The **PMC-AD8** is supported under *Windows NT* by a **Board Support Package** which is supplied with the card.

The card is also supported by a **DSP Support Library** providing the following features:

- Common access routines for communications with the HOST. These routines also allow for Direct and DMA access to HOST memory.
- Identify the applicable card resources and parameters.
- Allow for serial communication through the provided serial port for debugging DSP code and configuring the card.

A bootloader provided on the card allows for control by the HOST, including downloading custom DSP code. User code can also be downloaded to FLASH memory and booted automatically on reset.

All these are provided in a manner consistent across ALPHI Technology platforms. Other documentation supplied with the card will describe the support in full detail.

## **1.4 REFERENCE MATERIALS LIST**

### **PMC Local Bus Specification:**



PMC Special Interest Group

Tel: (800) 433-5177

Tel: (503) 797-4207

Fax: (503) 234-6762

**PCI 9080 PCI Controller data book:**

PLX Technology Corporation

Tel: (408) 744-9060

<http://www.plxtech.com>

[apps@plxtech.com](mailto:apps@plxtech.com)

**DSP (TMS320C32) C Compiler, Assembler, Linker:**

*Part Number TMDU3243855-02*

Texas Instruments

Tel: (410) 312-7900

<http://www.ti.com>

**DSP Debugger and Integrated Development Environment:**

*Code Composer*

GO-DSP Corporation

Tel: (416) 599-6868

<http://www.go-dsp.com>

**DSP Emulators suitable for use with *Code Composer* or TI debugger:**

White Mountain DSP

(603) 833-2430

<http://www.wmdsp.com>

**WindowsNT and Windows95 Programming Tools:**

BlueWater Systems

Tel: (206) 771-3610

Fax: (206) 771-2742

[info@bluewatersystems.com](mailto:info@bluewatersystems.com)

<http://www.bluewatersystems.com>

## **2. PMC BRIDGE TO LOCAL DSP BUS**

A PLX PCI9080 bridge chip is used to interface the Local DSP Bus to the PMC Bus of the HOST. This interface provides the card with the following features.

- Completely transparent access by HOST to the Local DSP Bus.
- Completely transparent access by DSP to the PMC bus.
- Eight mailbox registers and two doorbell registers for communication between the HOST and the DSP.
- Two Bus Master DMA channels with scatter / gather support.
- Full interrupt support to HOST and DSP.

### **2.1 INTERFACE TO HOST (PMC)**

All PMC devices contain a set of registers in Configuration Space which allow for determining the manufacturer and model of the device, determining the resources necessary for the correct operation of the device, and other configuration information. Configuration space is decoded on a per slot / device basis via a mechanism described in the PMC specification.

All PMC devices can be relocated in physical memory by means of several Base Address Registers. These registers contain the high address bits, which must be matched for any access to the card to be successful. Part of the protocol of programming the Base Address Registers allows for the transfer of information regarding the size of the regions. The actual Base Address Registers are located in Configuration Space.

Normally, configuring the card is performed by the system controller and some form of firmware or BIOS. Unfortunately, determining Base Addresses and other configuration information in Configuration Space is operating system dependent. One of the primary functions of the device driver under Windows NT is to map these resources to the user application.

The card is actually accessed through the decoded base address registers.

### **2.2 PMC CONFIGURATION SPACE**

The card has the following registers available to PMC Configuration Space. They are implemented in the PLX chip. The registers are also accessible from the DSP Local bus.

Offset Into PMC CFG	DSP Address	31 – 24	23 – 16	15 – 8	7 – 0
0x00	0x880000	Device ID		Vendor ID	
0x04	0x880001	Status		Command	
0x08	0x880002	Class Code			Revision ID
0x0C	0x880003	BIST	Header Type	PMC Latency Timer	Cache Line Size
0x10	0x880004	PMC Base Address 0 (Memory Access to PLX Registers)			
0x14	0x880005	PMC Base Address 1 (I/O Access to PLX Registers)			
0x18	0x880006	PMC Base Address 2 (Memory Access to DSP SRAM and card registers)			
0x1C	0x880007	PMC Base Address 3 (Not Used for this card)			
0x20	0x880008	Unused PMC Base Address 4			
0x24	0x880009	Unused PMC Base Address 5			
0x28	0x88000A	Cardbus CIS Pointer (Not Supported)			
0x2C	0x88000B	Subsystem ID		Subsystem Vendor ID	
0x30	0x88000C	PMC Base Address for Expansion ROM			
0x34	0x88000D	Reserved			
0x38	0x88000E	Reserved			
0x3C	0x88000F	Max Latency	Min Grant	Interrupt Pin	Interrupt Line

**Table 2.1: PMC Configuration Space**

The card presents the following initial configuration values to the PMC system, based on the values stored in the NVRAM device read by the PLX PCI9080 interface chip.

<b>Register</b>	<b>Value (Meaning)</b>
Vendor ID	0x13C5 (ALPHI Technology)
Device ID	0x010B
Revision ID	0x00
Class Code	0xff0000 (Device does not fit into defined class codes)
Interrupt Line	0xff
Interrupt Pin	A
Multifunction Device	No
Build In Self Test	No
Latency Timer	0x00
Minimum Grant	0x00
Maximum Latency	0x00
Base Address 0 Size	0x100 Bytes (Memory Access to PLX Registers)
Base Address 1 Size	0x100 Bytes (I/O Access to PLX Registers)
Base Address 2 Size	0x200000 Bytes (Memory Access to DSP SRAM and card registers)
Base Address 3 Size	None (Not applicable to this card)
Expansion ROM Size	None

**Table 2.2: PMC Configuration Register Default Values**

### **2.3 PMC BASE ADDRESS REGIONS**

There are 4 base address regions available on the PLX PCI9080. The **PMC-AD8** uses 3 of these regions.

- BAR0: Memory access the PLX operation registers.
- BAR1: I/O access the PLX operation registers.
- BAR2: Passthru to the DSP bus and is used to access the DSP SRAM and the card control / status registers. The passthru is established at power on via the NVRAM.
- BAR3: Not used on this design.

**NOTE: Regions, which are accessed by memory cycles, will reside at physical addresses above 1MB. Therefore, non-protected (real) mode operating systems running on a x86 class processor, such as DOS, will not be able to access these regions. Any applications, which must run under DOS, must be compiled with a DOS Protected Mode Extender, such as PharLap or will require making DPMI calls to switch to protected mode. ALPHI Technology does not provide any DOS software support, but we will be happy to help with hardware technical issues.**

### **3. LOCAL DSP BUS**

The following devices are present on the Local DSP Bus at the addresses specified. Many of these devices are also accessible by the HOST directly through the Base Address Regions. The PLX PCI9080 accesses the Local DSP Bus by placing the DSP in a HOLD state, performing any cycles necessary, and releasing the DSP from HOLD.

<b>HOST Address</b>	<b>DSP Address</b>	<b>Data</b>	<b>R/W</b>	<b>Description</b>
BAR2: 0x000000 – 0x07FFFF	0x000000 – 0x01FFFF	D31-D00	R/W	Local SRAM
BAR2: 0x080000 – 0x0FFFFFFF	0x020000 – 0x03FFFF	D31-D00	R/W	Local SRAM repeated
BAR2: 0x100000 – 0x17FFFF	0x040000 – 0x05FFFF	D15-D00	R/W	Card Control / Status Registers
BAR0: 0x00 – 0xFF	0x880000 – 0x88003F	D31-D00	R/W	PLX Operation Registers
Not Accessible	0x900000 – 0x97FFFF	D07-D00	RO	Local FLASH
Not Accessible	0xC00000 – 0xFFFFFFFF	D31-D00	R/W	Pass through to PMC bus

**Table 3.1: Local DSP Bus Overview**

In certain applications, it may be necessary to calculate the impact of accessing the Local DSP Bus from the HOST. The following table of events should help to explain the timing of a SRAM read. A single 32 bit read will stall the DSP for 10 clocks, or 333 nS. Burst access will add a single clock for each additional word. Of course, the DSP can continue running from the DSP cache and using internal RAM inside the DSP. Additionally, access to the Local Bus by the PLX can be turned off by the DSP for short periods of time during critical code sections.

<b>Cycle</b>	<b>Action by DSP</b>	<b>Action by PLX</b>
0	Normal Access	Signals request for bus
1	Idles bus prior to relinquish	Waits for acknowledgement
2	Relinquishes bus and acknowledges	Seizes bus, driving idle
3	Waits for release	Drives Address Phase for Read
3+1	Waits for release	Performs Data Phase for Read
3+2	Waits for release	Optionally continue burst Read
...	...	...
3+n	Waits for release	Last burst read
4+n	Waits for release	Internal Synchronization
5+n	Waits for release	Internal Synchronization
6+n	Waits for release	Releases bus. Driven idle by card.
7+n	One cycle lost to synchronization	Idle
8+n	DSP recognizes release	Idle
9+n	DSP performs idle cycle	Idle

**Table 3.2: HOST Read of Local DSP SRAM**

### **3.1 LOCAL SRAM**

There is 128k x 32 of Local SRAM for storing DSP programs and buffering data. The SRAM is accessible to both the DSP and the HOST.

The SRAM operates at zero wait states to the DSP, and one wait state to the PLX PCI9080 due to the separate address and data phase. Additional access in burst mode by the PLX will occur at zero wait state.

The decoding of the SRAM occurs twice, at 0x000000 and 0x020000 in order to make use of the 4k section at 0x000000 which is normally lost to internal DSP access. This memory can be accessed at 0x020000 – 0x020FFF.

### **3.2 CARD CONTROL / STATUS REGISTERS**

The following locations are the Control / Status registers for the card. These registers are accessible to both the DSP and the HOST.

These registers operate with three wait states to the DSP.

HOST Address BAR2:	DSP Address	Data	R/W	Description
0x100000	0x040000	D03-D00	R/W	9080 Control / Status Register
0x100020	0x040008	N/A	WS	HOST Interrupt Acknowledge
0x100040	0x040010	D07-D00	R/W	Board Control / Status Register
0x100060	0x040018	N/A	WS	Reset the hardware FIFO
0x100080	0x040020	D15-D00	R	Read the hardware FIFO
0x100080	0x040020	D15-D00	W	Write the hardware FIFO for testing and configuration
0x1000A0	0x040028	N/A	WS	Not applicable to this design
0x1000C0	0x040030	D31-D00	R/W	PGA Gain Registers
0x1000E0	0x040038	N/A	WS	Software Strobe for manual clocking by DSP
0x100100	0x040040	D07-D00	R/W	Trigger / Gate Configuration Registers
0x100120	0x040048	D07-D00	R/W	Internal Serial Port
0x100140	0x040050	N/A	WS	Software Strobe for manual clocking by HOST

**Table 3.3: Control / Status Registers**

### 3.2.1 9080 CONTROL / STATUS REGISTER

This register allows for simplified determination of the cause of a DSP interrupt from the PLX, as well as providing a way to temporarily lockout access by the HOST.

BIT 03	BIT 02	BIT 01	BIT 00
LSERR	LINTO	DMPAF	LOCKOUT9080

#### **LSERR**

*Read Only*

This bit tracks the current state of the LSERR line from the PLX PCI9080. When this is low, (0), it implies that the DSP interrupt was caused by this hardware signal. See the PLX documentation for more details and for a list of causes.

#### **LINTO**

*Read Only*

This bit tracks the current state of the LINTO line from the PLX PCI9080. This line is the normal way in which the PLX interrupts the DSP. When this is low, (0), it implies that the DSP interrupt was caused by this hardware signal. See the PLX documentation for more details and for the enable bits.

#### **DMPAF**

*Read Only*

This bit tracks the current state of the DMPAF line from the PLX PCI9080. This line can be used to throttle direct writes by the DSP to the HOST PCI bus. When this is low, (0), it implies that the FIFO inside the PLX has a programmable number of accesses queued, but not yet completed for the HOST PMC bus. See the PLX documentation for more details and for the enable bits.

**LOCKOUT9080**

*Read/Write*

This bit, when set (1) will prevent the PLX PCI9080 from seizing the Local Bus. This can allow for time critical DSP code to complete in a timely manner. This bit should only be set for short intervals of time, since it may potentially lock up the HOST PMC bus until it has been cleared (when an access to the card is performed).

**3.2.2 HOST INTERRUPT ACKNOWLEDGE**

*Write Strobe*

The HOST can write to this register to turn off the HOST interrupt generated by LINTI. It is intended for use by the HOST device driver.

**3.2.3 BOARD CONTROL / STATUS REGISTER**

This register allows for querying the current state of the hardware FIFO flags and a means to program the FIFO.

BIT 07	BIT 06	BIT 05	BIT 04	BIT 03	BIT 02	BIT 01	BIT 00
FIFO FF	FIFO PAF	FIFO PAE	FIFO EF	N/A	N/A	N/A	FIFO REG

**FIFO FF**

*Read Only*

This bit, when low (0), indicates that there are 8192 samples stored in the hardware FIFO.

**FIFO PAF**

*Read Only*

This bit, when low (0), indicates that there is a programmable number of samples stored in the hardware FIFO.

**FIFO PAE**

*Read Only*

This bit, when low (0), indicates that there is less than a programmable number of samples stored in the hardware FIFO.

**FIFO EF**

*Read Only*



This bit, when low (0), indicates that there are no samples stored in the hardware FIFO.

**FIFO REG**

*Read / Write*

This bit, when cleared to low (0), causes writes to the FIFO to be placed into the FIFO for testing purposes. When the bit is set high (1), writes and reads to the FIFO will access the registers.

**3.2.4 RESET THE HARDWARE FIFO**

*Write Strobe*

A write to this location will reset the hardware FIFO to empty. It is required to initialize the FIFO before any access is made.

**3.2.5 READ THE HARDWARE FIFO**

*Read Access*

If the **FIFO REG** bit is clear (0), then a read of this location will respond with the oldest sample in the FIFO. If the **FIFO REG** bit is set (1), then a read of this location will read the internal configuration registers of the FIFO.

When the state machine writes to the FIFOs, it writes all eight conversion results in channel order (0 – 7).

**3.2.6 WRITE THE HARDWARE FIFO FOR TESTING AND CONFIGURATION**

*Write Access*

If the **FIFO REG** bit is clear (0), then a write to this location will directly add the value to the FIFO for testing purposes. If the **FIFO REG** bit is set (1), then a write to this location will program the internal configuration registers of the FIFO.

The FIFO is actually one IDT – 72285 ( 64k) FIFO.

**3.2.7 PGA GAIN REGISTERS**

*Write Only*

There are two PGA gain registers as follows

HOST Address BAR2:	DSP Address	BITS 07-06	BITS 05-04	BITS 03-02	BITS 01-00
0x1000C0	0x040030	Chan 4	Chan 3	Chan 2	Chan 1
0x1000C4	0x040031	Chan 8	Chan 7	Chan 6	Chan 5

If the optional additional 8 channels are present, the following registers are also present.

HOST Address BAR2:	DSP Address	BITS 07- 06	BITS 05-04	BITS 03-02	BITS 01-00
0x1000C8	0x040032	Chan 12	Chan 11	Chan 10	Chan 9
0x1000CC	0x040033	Chan 16	Chan 15	Chan 14	Chan 13

The gains can be selected from the following:

BITS 01-00	Factory Default	Special Order
00	x1	x1
01	x2	x10
10	x4	x100
11	x8	X1000

**Table 3.4: Programmable Gain Selections**

### 3.2.8 SOFTWARE STROBE FOR MANUAL CLOCKING BY DSP

*Write Strobe*

A write to this location can potentially trigger a conversion or other event if selected in the Trigger / Gate Configuration Registers.

### 3.2.9 TRIGGER / GATE CONFIGURATION REGISTERS

These registers control the trigger and gate signal routing inside the card.

HOST Address BAR2:	DSP Address	BITS 07-04	BITS 03-00
0x100100	0x040040	A/D Clock Source	A/D Gate Source
0x100104	0x040041	HOST Interrupt Source	Front Panel Clock Output Source
0x100108	0x040042	N/A	DSP Interrupt Source

**Trigger / Gate Sources**

*Read / Write*

The following signals can be selected from several sources.

- A/D Clock
- A/D Gate
- Front Panel Clock Output
- HOST Interrupt
- DSP Interrupt

<b>BITS 03-00</b>	<b>Source</b>
<b>0x0</b>	Low (0)
<b>0x1</b>	High (1)
<b>0x2</b>	DSP Timer 0
<b>0x3</b>	DSP Timer 1
<b>0x4</b>	Trigger Input (Front Panel)
<b>0x5</b>	Gate Input (Front Panel)
<b>0x6</b>	Strobe Register from HOST
<b>0x7</b>	Strobe Register from DSP

**Table 3.5: Trigger / Gate Sources**

**3.2.10 PROGRAMMING PMC-AD8 (72285) FIFO**

This procedure explains how set the FIFO pointer then write values into the FIFO and readback.

Follow the procedure in the order it's written do not skip any steps.

***Program the A/D sample clock***

To receive samples from the A/D 7805 the FIFO needs to be programmed with a sample clock source and a gate source. For our example the trigger/gate configuration register for the A/D clock will be the DSP Timer 0 (DSP-TCLK0) and for the gate source we must set a state condition of 1 = High this will start the acquisition of data to the FIFO and a 0 = low will stop the acquisition of data in the FIFO. The gate condition will not be programmed until you are ready to start acquisition.

Program A/D sample clock.

Trigger/gate configuration register \$40040, write \$20 (DSP Timer 0).

HOST Address BAR2:	DSP Address	BITS 07-04	BITS 03-00
0x100100	0x040040	A/D Clock Source	A/D Gate Source
0x100104	0x040041	HOST Interrupt Source	Front Panel Clock Output Source
0x100108	0x040042	N/A	DSP Interrupt Source

BITS 03-00 BITS 07-04	Source
0x0	Low (0)
0x1	High (1)
0x2	DSP Timer 0
0x3	DSP Timer 1
0x4	Trigger Input (Front Panel)
0x5	Gate Input (Front Panel)
0x6	Strobe Register from HOST
0x7	Strobe Register from DSP

#### **Program DSP Timer 0 (DSP- TCLK0) Frequency**

- \$808028 – write \$47 This sets the frequency of the TCLK0 (\$47 = 100Khz) See TI (TMSC3X) user guide for explanation on how to change TCLK0 to a different frequency.

#### **Start DSP Timer 0**

\$808020 – write \$3C3 , this will start TCLK0 counting.

#### **Program PAE pointer.**

- \$40010 - Board control/status register, write \$1 this sets FIFO\_REG bit to "1". This enables and allows access of the FIFO programming registers.
- \$40018 - Hardware FIFO reset, write \$0000, a write to this location will reset the hardware FIFO.
- Program FIFO register. ( example 64 locations)
- \$40020 - Hardware FIFO, should read \$7F ( PAE = 128 locations) Note: first read of the FIFO is PAE pointer and next read of same address is PAF and then revolves between the two.
- \$40020 - Hardware FIFO, write \$3F ( PAE = 64 locations) then exit. If you want more or less samples the amount of channel samples are equal in Hex. So write to the PAE pointer the value you wish, here are some examples 0= 0ch, 8= 8ch, 10= 16ch ect.

**Close FIFO register**

- \$40010 - Board control/status register, write \$0 FIFO\_Reg bit = "0". Now we are in FIFO data normal mode. Data are written inside the FIFO and data can be read from the FIFO.

**Start FIFO acquisition.**

- \$40040 - Now program the gate source which will start the acquisition of the FIFO. For the gate source you must set a state condition of 1 = High this will start the acquisition of data to the FIFO.
  1. Write \$21 , This will start the acquisition of data from the I/O channels.

**End acquisition**

- \$40040 – Now you will stop the acquisition of data to the FIFO.
  1. Writing \$20 = low will stop the acquisition of data in the FIFO.

**Verify Access and Data in FIFO**

\$40010 - Board control/status, read of FIFO status register should be \$32 the 2 means nothing.

BIT 07	BIT 06	BIT 05	BIT 04	BIT 03	BIT 02	BIT 01	BIT 00
FIFO FF	FIFO PAF	FIFO PAE	FIFO EF	N/A	N/A	N/A	FIFO REG

- \$40020 - Hardware FIFO, read data in the FIFO = 1 time. This will clear the FIFO-FF.
- \$40010 - Board control/status, FIFO-FF should be back to a one now you should read \$B2.
- If you read more than the number samples programmed to PAE pointer then the PAF will be cleared and should read at \$40010 Board control/status \$F2.
- If all is correct then you have completed a successful acquisition.

**3.2.11 INTERNAL SERIAL PORT**

The **PMC-AD8** includes a simple UART for board configuration and for debug output of DSP code. Although the UART is a custom design, it is fairly simple, and there is already DSP support in the DSP library. It supports asynchronous input and output at the most common rates, 8 bits per character, no parity, and one stop bit. See I/O pinout for configuration.

The registers are implemented as follows.

HOST Address BAR2:	DSP Address	BITS 07-00
0x100120	0x040048	Serial Data Register
0x100124	0x040049	Serial Control / Status Register
0x100128	0x04004A	Serial Baud Rate Register

**Table 3.6: Internal Serial Port**

### **Serial Data Register**

*Read / Write*

When read, this register contains the last character received. When written, this register contains the next character for output.

### **Serial Control / Status Register**

*Read / Write and Read Only*

BIT 07	BIT 06	BIT 05	BIT 04	BIT 03	BIT 02	BIT 01	BIT 00
En Tx Int	En Rx Int	N/A	N/A	N/A	N/A	Rx Ready	Tx Busy

Rx Ready is set when a character has been received. Tx Busy is set when a character is being output. The interrupt enable lines are not used in this design.

### **Serial Baud Rate Register**

*Read / Write*

This register contains the divisor to generate the baud rate clock. The input frequency is 2 MHz.

$$register = \frac{2,000,000}{DesiredBaudRate}$$

### **3.2.12 SOFTWARE STROBE FOR MANUAL CLOCKING BY HOST**

*Write Strobe*

A write to this location can potentially trigger a conversion or other event if selected in the Trigger / Gate Configuration Registers.

## **3.3 LOCAL FLASH**

There is 512k x 8 of FLASH memory available to the DSP on the Local Bus. Part of this is reserved for the use of the DSP Bootloader with the remainder available for customer applications. Access occurs with about 10 wait states due to the slow speed of the device.

### 3.4 PLX OPERATION REGISTERS

The HOST processor can access the internal PLX registers through BAR0 and BAR1. BAR0 is accessed via normal memory operations and BAR1 is accessed via I/O operations.

The DSP can access these registers through normal memory cycles at the addresses specified.

The registers can be divided into several categories.

- Local Configuration Registers
- Runtime Registers
- DMA Registers

There are insufficient resources on the card to support I<sub>2</sub>O message queues, so these registers are not described.

#### 3.4.1 LOCAL CONFIGURATION REGISTERS

See the PLX PCI9080 Documentation for more details about these registers.

HOST Address BAR0/1:	DSP Address	31 – 24	23 – 16	15 – 8	7 – 0
0x00	0x880020	Range for PMC to Local Address Space 0			
0x04	0x880021	Local Base Address for PMC to Local Address Space 0			
0x08	0x880022	Mode / Arbitration Register			
0x0C	0x880023	Big / Little Endian Register			
0x10	0x880024	Range for PMC to Local Expansion ROM			
0x14	0x880025	Local Base Address for PMC to Local Expansion ROM			
0x18	0x880026	Local Bus Region Descriptors			
0x1C	0x880027	Range for Direct Master to PMC			
0x20	0x880028	Local Base Address for Direct Master to PMC Memory			
0x24	0x880029	Local Base Address for Direct Master to PMC IO/CFG			
0x28	0x88002A	PMC Base Address for Direct Master to PMC			
0x2C	0x88002B	PMC Configuration Address for Direct Master to PMC IO/CFG			
0xF0	0x88005C	Range for PMC to Local Address Space 1			
0xF4	0x88005D	Local Base Address for PMC to Local Address Space 1			
0xF8	0x88005E	Local Bus Region Descriptor for PMC to Local			

**Table 3.7: PLX Local Configuration Registers**

### 3.4.2 RUNTIME REGISTERS

See the PLX PCI9080 Documentation for more details about these registers.

HOST Address BAR0/1:	DSP Address	31 – 24	23 – 16	15 – 8	7 – 0
0x40	0x880030	Mailbox Register 0			
0x44	0x880031	Mailbox Register 1			
0x48	0x880032	Mailbox Register 2			
0x4C	0x880033	Mailbox Register 3			
0x50	0x880034	Mailbox Register 4			
0x54	0x880035	Mailbox Register 5			
0x58	0x880036	Mailbox Register 6			
0x5C	0x880037	Mailbox Register 7			
0x60	0x880038	PMC to Local Doorbell Register			
0x64	0x880039	Local to PMC Doorbell Register			
0x68	0x88003A	Interrupt Control / Status			
0x6C	0x88003B	Serial EEPROM, PMC Command Codes, User I/O, Init			
0x70	0x88003C	Device ID		Vendor ID	
0x74	0x88003D	Unused		Revision ID	
0x78	0x88003E	Mailbox Register 0			
0x7C	0x88003F	Mailbox Register 1			

**Table 3.8: PLX Runtime Registers**

### 3.4.3 DMA REGISTERS

See the PLX PCI9080 Documentation for more details about these registers.



HOST Address BAR0/1:	DSP Address	31 – 24	23 – 16	15 – 8	7 – 0
0x80	0x880040	DMA Ch0 Mode			
0x84	0x880041	DMA Ch0 PMC Address			
0x88	0x880042	DMA Ch0 Local Address			
0x8C	0x880043	DMA Ch0 Transfer Byte Count			
0x90	0x880044	DMA Ch0 Descriptor Pointer			
0x94	0x880045	DMA Ch1 Mode			
0x98	0x880046	DMA Ch1 PMC Address			
0x9C	0x880047	DMA Ch1 Local Address			
0xA0	0x880048	DMA Ch1 Transfer Byte Count			
0xA4	0x880049	DMA Ch1 Descriptor Pointer			
0xA8	0x88004A	Reserved		DMA Ch1 CSR	DMA Ch0 CSR
0xAC	0x88004B	Mode / Arbitration Register			
0xB0	0x88004C	DMA Threshold Register			

**Table 3.9: PLX DMA Registers**

### 3.5 PMC PASS-THROUGH REGION

The **PMC-AD8** can directly access HOST memory on the PMC system. HOST memory can be mapped into this region in up to 16 Mbytes at a time.

**NOTE: Windows NT and Windows 95/98 supports virtual memory and separate flat address spaces for each task. Physical memory pages are not contiguous.**

**There is support in the Board Support Package and *alphi\_io* DSP library to allow the DSP to directly access HOST memory on a per task basis.**

## **4. HARDWARE DETAILS**

### **4.1 RESET SIGNALS**

The **PMC-AD8** can be reset by several different means.

- When PMC Reset is asserted, the PLX PCI9080 will reset itself and the rest of the card. The NVRAM is reloaded by the PLX.
- At power on, or when the reset jumper is shorted, the watchdog timer will hold the DSP RESET line low for 200 mS. The card registers will also be reset. The PLX will NOT be reset. When the DSP starts to run, certain registers are reloaded by the DSP to enable the passthru regions to the card registers.
- The PLX PCI9080 has a bit called USER0 which should not be used to reset the card. (This is a change from earlier versions of this manual.)
- The PLX PCI9080 has a bit called PMC Adapter Software Reset which the HOST can toggle to reset the DSP and the card registers. When the DSP starts to run, certain registers are reloaded by the DSP to enable the passthru regions to the card registers.

### **4.2 LOCAL DSP INTERRUPT SOURCES**

The local DSP has four interrupt lines. The source of each interrupt is listed in the table below. Additionally, at reset, the INT2 line is pulsed to tell the DSP to find the FLASH image at 0x900000 (if boot mode is MC).

<b>SOURCE</b>	<b>ENABLE SIGNAL</b>	<b>INTERRUPT LEVEL</b>
PLX LINTO and LSERR	Inside PLX	INT0
None	None	INT1
DMPAF	None	INT2
FIFO PAE	None	INT3

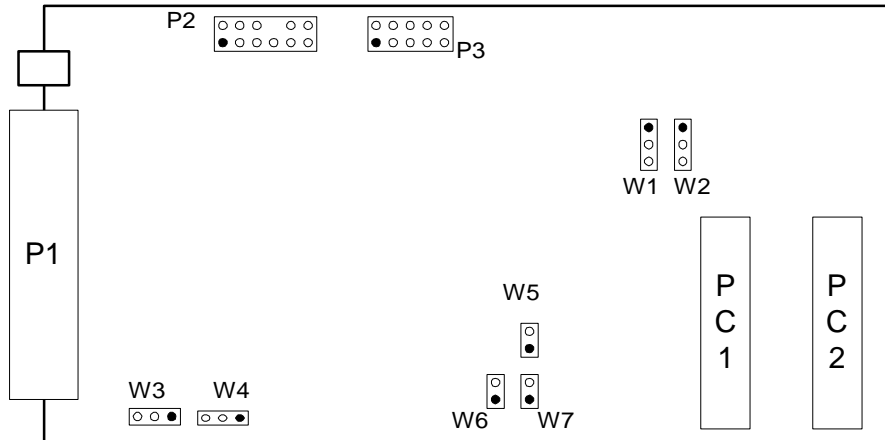
**Table 4.1: Interrupt Sources**

### **4.3 HOST INTERRUPT SOURCE**

The HOST can receive an interrupt from any of the other Trigger / Gate sources on the card. The interrupt can be cleared by writing to a particular location.

#### 4.4 CONNECTORS, JUMPERS, AND LEDS

The jumper and connector placement is depicted below.



**Figure 4.1: Jumper and Connector Locations**

##### 4.4.1 JUMPER DESCRIPTIONS

JUMPER	FACTORY SETTING	DESCRIPTION
W1	2-3	Factory use. Always connect 2-3.
W2	2-3	Factory use. Always connect 2-3.
W3	1-2 2-3	PGA Internal Power: -12V PGA External Power: -15V
W4	1-2 2-3	PGA Internal Power: +12V PGA External Power: +15V
W5	None	Factory use. Never connect.
W6	None	Factory use. Never connect.
W7	None	When shorted, provides DSP and board reset.

**Table 4.2 Jumper Descriptions**

##### 4.4.2 CONNECTOR DESCRIPTIONS

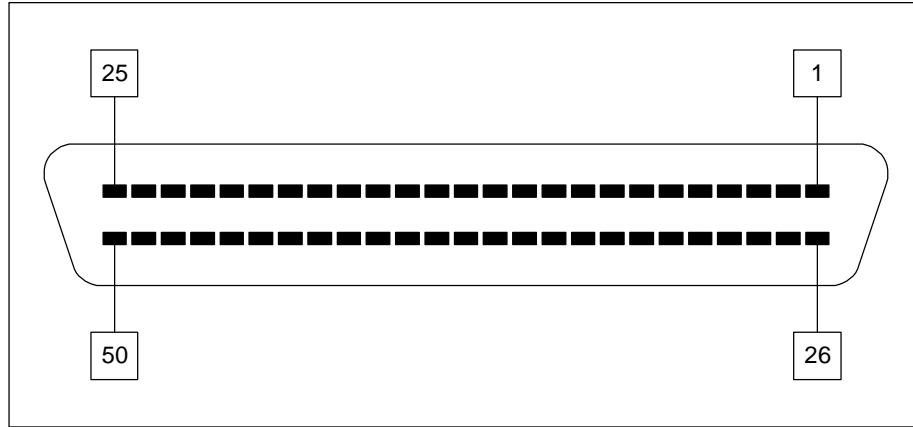
###### External I/O Connector (P1)

A 50 pin subminiature D shelled connector is used to route the analog signals to the card. ALPHI Technology can supply transition modules and cable assemblies to meet any special requirements. Contact the factory for additional information.

Connectors are manufactured by AMP.

Use	Model
On PC Board	787190-5
Suggested Plug	749111-4

**Table 4.3: I/O Connector Model Numbers**



**Figure 4.2 External I/O Connector**

Pin	Connection	Pin	Connection
1	GND	26	GND
2	Output Clock	27	Input Gate
3	GND	28	
4	Input Clock	29	GND
5	TXD: SERIAL	30	+5V
6	RXD: SERIAL	31	
7		32	
8	Ext +15v	33	Ext -15V
9	A_GND	34	A_GND
10		35	
11		36	
12	IN08+	37	IN08-
13	A_GND	38	A_GND
14	IN07+	39	IN07-
15	IN06+	40	IN06-
16	A_GND	41	A_GND
17	IN05+	42	IN05-
18	A_GND	43	A_GND
19	IN04+	44	IN04-
20	A_GND	45	A_GND
21	IN03+	46	IN03-
22	IN02+	47	IN02-
23	A_GND	48	A_GND
24	IN01+	49	IN01-
25	A_GND	50	A_GND

**Table 4.4: External I/O Connector**

**Description:**

**Input Clock (P1 PIN 4)**

Input pin to routes an external clock to the board. The input signal is TTL compatible. We recommend using pin 1 for GND with the clock.

**Input Gate (P1 pin 27)**

Input pin to routes an external gate to the board. The input signal is TTL compatible. We recommend using pin 26 for GND with the Gate pin.

**Output Clock (P1 pin 2)**

Input pin to routes a programmable clock off the board. The output signal is TTL compatible. We recommend using pin 3 for GND with the Output pin.

**Serial Port (P1 PIN 5,6,29,30)**

These pins provides TTL level signals to provide a serial port for debug / configuration. An external cable is available which converts the signals to RS232 levels.

**External power +/-15V (P1 PIN 8,33,9,34)**

These pins are used to power the PGA (Programmable gain Amplifier) with 15 Volt external power supply to have a full scale of +/-10V input range. When using external power supply, W3 and W4 must be jumpered correctly.

**Emulator Connector (P2)**

This connector is used to connect the emulator to the C32 DSP.

**Factory Use (P3)**

This connector is used at the factory for programming the FPGA.

**32 Bit PMC Bus (PC1, PC2)**

These connectors plug into the PMC slot and provides the standard PMC signals.

**4.4.3 LEDs**

**A/D Data Acquire (L1)**

When this LED is glowing, the A/D converters are being triggered. This is the top LED on the board.

## **5. SPECIFICATIONS**

### **5.1 INPUTS**

Number of Inputs	8.
Number Bits	16
Resolution	
Input Voltage Range	+/- 10 V.
Conversion Rate	100 kHz. Optionally up to 200 kHz.
Signal Conditioning	All channels are sampled and converted simultaneously. Input PGA allows for gains of x1, x2, x4 and x8 under software control. Optional x1, x10, x100, x1000, with a drop in input bandwidth. Inputs are differential mode.
Hardware FIFO	Samples are stored in an 8k x 16 FIFO. DSP can be interrupted at a programmable number of samples.
Integral Linearity Error	+/- 1.5 LSB.
Differential Linearity Error	+1.5 -1 LSB
Missing Codes	None to 16 bits
Full Scale Error	+/- 0.25%
Offset Error	+/- 10 mV
SFDR @ 45 kHz	96 dB
THD @ 45 kHz	-96 dB
SINAD @ 45 kHz	86 dB

### **5.2 TRIGGERS / CLOCKS**

Internal Sampling Clock	Two internal DSP timers.
Internal Clock Range	200 kHz to less than 1 Hz in steps of 33 nS.
External Clock / Gate Inputs	DC to 200 kHz TTL levels.
A/D Converter Clock / Gate	Selectable from Internal DSP timer, external trigger/gate, PXI trigger, DSP strobe, HOST strobe, or defined TTL level.
HOST and DSP Interrupt	Individually selectable from Internal DSP timer, external trigger/gate, DSP strobe, HOST strobe, or defined TTL level.
Clock Output	Selectable from Internal DSP timer, external trigger/gate, DSP strobe, HOST strobe, or defined TTL level.





### **5.3 DSP**

Type	TMS320C32 at 60 MHz.
RAM	128k x 32 running at zero wait states.
FLASH	512k x 8. Allows for storing bootloader and customer applications.
Interrupts	Generated by HOST (via PLX), FIFO state, and programmable trigger.
Application	Software provided to allow full access and control of the card.

### **5.4 HOST INTERFACE**

Bus Type	PMC Bus
Bridge Type	PLX 9080
Access To Host	Bus Master DMA and direct access to HOST memory.
Access To Card	Direct access by HOST via pass-through region and to DSP via mailboxes/doorbells.
Interrupts to HOST	Fully Programmable. Interrupt generated by DSP or by programmable trigger.

### **5.5 CONNECTIONS**

A/D Inputs	50 pin SCSI-type locking connector.
------------	-------------------------------------

### **5.6 POWER**

ADC Power	+/- 12 Volt From PMC connector
ADC Power	+/- 15 Volt supplied externally from P1