PCI-4IPM

Revision C

Second Generation Intelligent IP Carrier for PCI Systems
Up to Four IndustryPack® Modules
Dual Ported SRAM, Bus Master DMA

REFERENCE MANUAL

781-21-000-4000
Version 2.1
April 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

TABLE OF CONTENTS

1. GENERAL DESCRIPTION ____ 1

1.1 INTRODUCTION ____ 1

1.2 FUNCTIONAL DESCRIPTION ____ 2

1.3 TRIGGERING DESCRIPTION ____ 3

1.3.1 CLOCK / TRIGGER SOURCES ____ 3

1.3.2 INTERRUPT / TRIGGER TARGETS ____ 4

1.3.3 EXTERNAL TRIGGER OUTPUTS ____ 4

1.4 SOFTWARE SUPPORT ____ 6

1.5 REFERENCE MATERIALS LIST ____ 7

2. PCI BRIDGE TO LOCAL DSP BUS ____ 8

2.1 INTERFACE TO HOST (PCI) ____ 8

2.2 PCI CONFIGURATION SPACE ____ 9

2.3 PCI BASE ADDRESS REGIONS ____ 10

3. LOCAL DSP BUS ____ 11

3.1 LOCAL SRAM ____ 11

3.2 OPTIONAL ADDITIONAL LOCAL SRAM ____ 11

3.3 CARD CONTROL / STATUS REGISTERS ____ 11

3.3.1 9080 CONTROL / STATUS REGISTER ____ 13

3.3.2 HOST INTERRUPT ACKNOWLEDGE ____ 14

3.3.3 IP CONTROL REGISTER ____ 14

3.3.4 IP STATUS REGISTER ____ 16

3.3.5 NEXT DMA END A/B ____ 17

3.3.6 NEXT DMA END C/D ____ 17

3.3.7 CLEAR BERR ____ 17

3.3.8 SOFTWARE STROBE FOR MANUAL CLOCKING BY DSP ____ 18

3.3.9 CLOCK / INT CONFIGURATION REGISTERS ____ 18

3.3.10 8530 SERIAL PORTS ____ 20

3.3.11 SOFTWARE STROBE FOR MANUAL CLOCKING BY HOST ____ 21

3.3.12 DSP INTERRUPT STATUS REGISTER ____ 21

3.3.13 DSP INTERRUPT ENABLE REGISTER ____ 21

3.3.14 HOST/EXT INTERRUPT STATUS REGISTER ____ 22

3.3.15 HOST/EXT INTERRUPT ENABLE REGISTER ____ 22

3.3.16 IP DMA STATUS REGISTER ____ 22

3.3.17 IP DMA SELECTION REGISTER ____ 23

3.3.18 IP A INTERRUPT ACKNOWLEDGE ____ 23

3.3.19 IP A INTERRUPT ACKNOWLEDGE ____ 24

3.3.20 IP C INTERRUPT ACKNOWLEDGE ____ 24

3.3.21 IP D INTERRUPT ACKNOWLEDGE ____ 24

3.3.22 IP A 16 BIT IO ACCESS ____ 24

3.3.23 IP A 16 BIT IO ACCESS WITH DMA ACKNOWLEDGE ____ 25

3.3.24 IP B 16 BIT IO ACCESS ____ 25

3.3.25 IP B 16 BIT IO ACCESS WITH DMA ACKNOWLEDGE ____ 25

3.3.26 IP A/B 32 BIT IO ACCESS ____ 25

3.3.27 IP A/B 32 BIT IO ACCESS WITH DMA ACKNOWLEDGE ____ 25

3.3.28 IP A 16 BIT ID ACCESS ____ 26

3.3.29 IP B 16 BIT ID ACCESS 26
3.3.30 IP C IO ACCESS 26
3.3.31 IP C IO ACCESS WITH DMA ACKNOWLEDGE 26
3.3.32 IP D IO ACCESS 26
3.3.33 IP D IO ACCESS WITH DMA ACKNOWLEDGE 28
3.3.34 IP C/D 32 BIT IO ACCESS 28
3.3.35 IP C/D 32 BIT IO ACCESS WITH DMA ACKNOWLEDGE 28
3.3.36 IP C 16 BIT ID ACCESS 28
3.3.37 IP D 16 BIT ID ACCESS 28
3.4 LOCAL FLASH 29
3.5 PLX OPERATION REGISTERS 29
3.5.1 LOCAL CONFIGURATION REGISTERS 30
3.5.2 RUNTIME REGISTERS 31
3.5.3 DMA REGISTERS 32
3.6 PCI PASS-THROUGH REGION 33

4. HARDWARE DETAILS 34
4.1 DMA 34
4.1.1 DSP DMA 34
4.1.2 DSP TO PCI BRIDGE DMA 34
4.1.3 DEMAND MODE DMA 34
4.2 EXTERNAL TRIGGERING 34
4.3 INTERRUPTS 35
4.3.1 DSP INTERRUPTS 35
4.3.2 HOST INTERRUPTS 36
4.4 RESET 36
4.4.1 PCI HARDWARE RESET 36
4.4.2 SOFTWARE RESET FROM A PCI DEVICE 37
4.4.3 WATCHDOG RESET 37
4.5 CONNECTORS, JUMPERS, AND LEDS 38
4.5.1 JUMPER DESCRIPTIONS 39
4.5.2 FRONT PANEL DESCRIPTION 40
4.5.3 LED INDICATORS 40
4.5.4 IP I/O CONNECTORS 41
4.6 SERIAL RS232 PORT (P9) 41
4.7 SERIAL RS422 / RS485 PORT (P12) 41
4.8 DSP SERIAL PORT (P11) 42
4.9 EXT CLOCK / INT CONNECTOR (P10) 42
4.10 EMULATOR CONNECTION (P8) 42
4.11 POWER FOR STAND ALONE OPERATION (PW1) 43
4.12 FACTORY USE (P13) 43

1. GENERAL DESCRIPTION

1.1 INTRODUCTION

The **PCI-4IPM** is a PCI expansion card in a standard full length form factor. The **PCI-4IPM** provides mechanical support and the electrical interfaces for four single width IP modules, or two double width IP modules. Multiple **PCI-4IPM** boards may be installed in a single system. The primary features of the **PCI-4IPM** are as follows:

- Support for up to four IP modules
- 8 MHz or 32 MHz IP operation via software selection
- Integrated DSP (TI TMS320C32) at 60 MHz available to process the data, if desired, or to offload I/O operation from HOST processor.
- 128K x 32 bit zero wait state SRAM for the DSP.
- 128K x 32 bit Dual Port Ram for the HOST or DSP.
- 512K by 8 bit FLASH device for bootloader and customer applications.
- DSP is fully capable of accessing HOST RAM, and any other devices present on the PCI bus
- Full interrupt support of host and DSP C32
- Two channels of Bus Master DMA with Scatter/Gather support to offload the DSP and the HOST processor.
- Extremely flexible triggering/interrupting, including internal, external, and IPSTROBE sources.
- Supports double-wide form factor IndustryPack®
- Front panel I/O connectors for 2 IPs. Other 2 IPs require additional panel.
- One RS232 port and one RS422 port.
- Full software support including DSP code and Drivers/DLL for WinNT.

The **PCI-4IPM** is ALPHI Technology's second generation of IP carrier for the PCI bus. It improves upon the **PCI-4Pack** by adding the following:

- DSP is fully capable of accessing HOST RAM, and any other devices present on the PCI bus
- Zero wait state access to shared memory by DSP.
- Two channels of Bus Master DMA with Scatter/Gather support to offload the DSP and the HOST processor.
- Extremely flexible triggering/interrupting, including internal, external, and IPSTROBE sources.

1.2 FUNCTIONAL DESCRIPTION

An overview block diagram of the **PCI-4IPM** is presented in Figure 1-1.

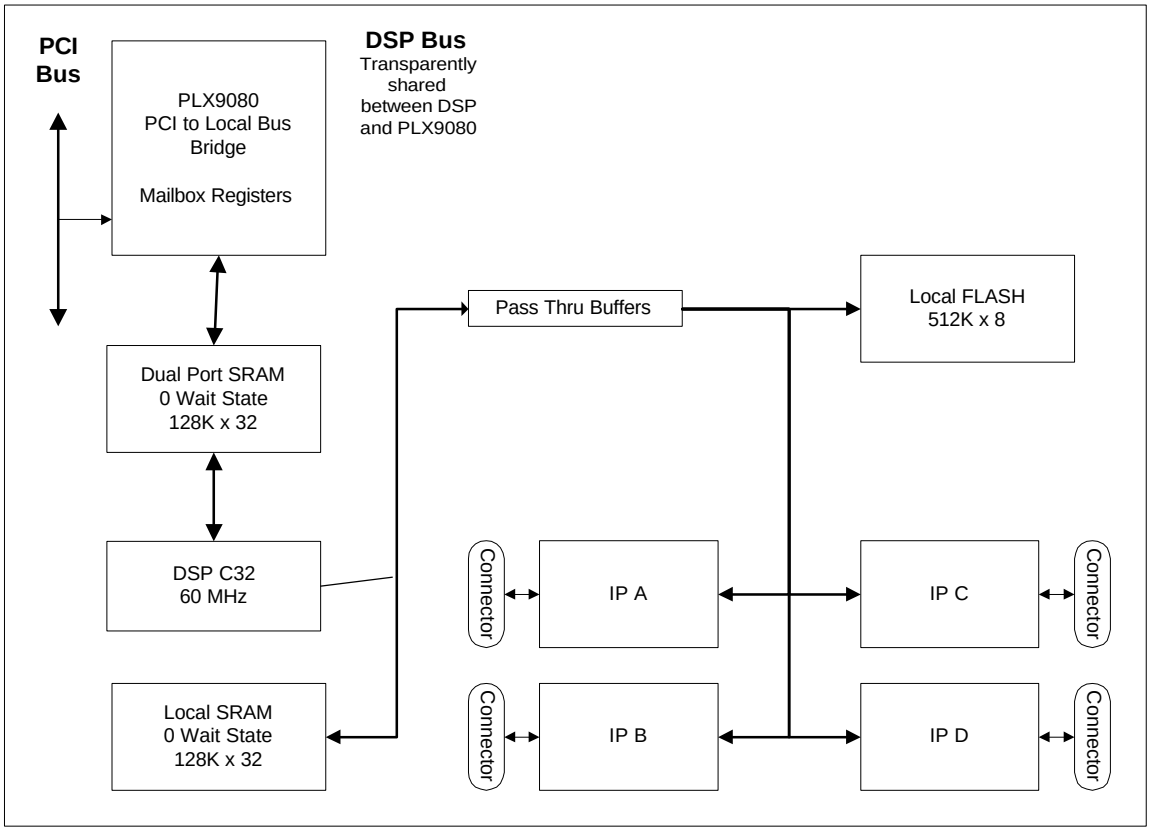

Figure 1.1: Overview Block Diagram

1.3 TRIGGERING DESCRIPTION

The **PCI-4IPM** has extremely flexible triggering of IPSTROBE, and the DSP and HOST interrupts. It allows for efficient synchronization between carriers and IPs. The triggering block diagram is presented in Figure 1-2.

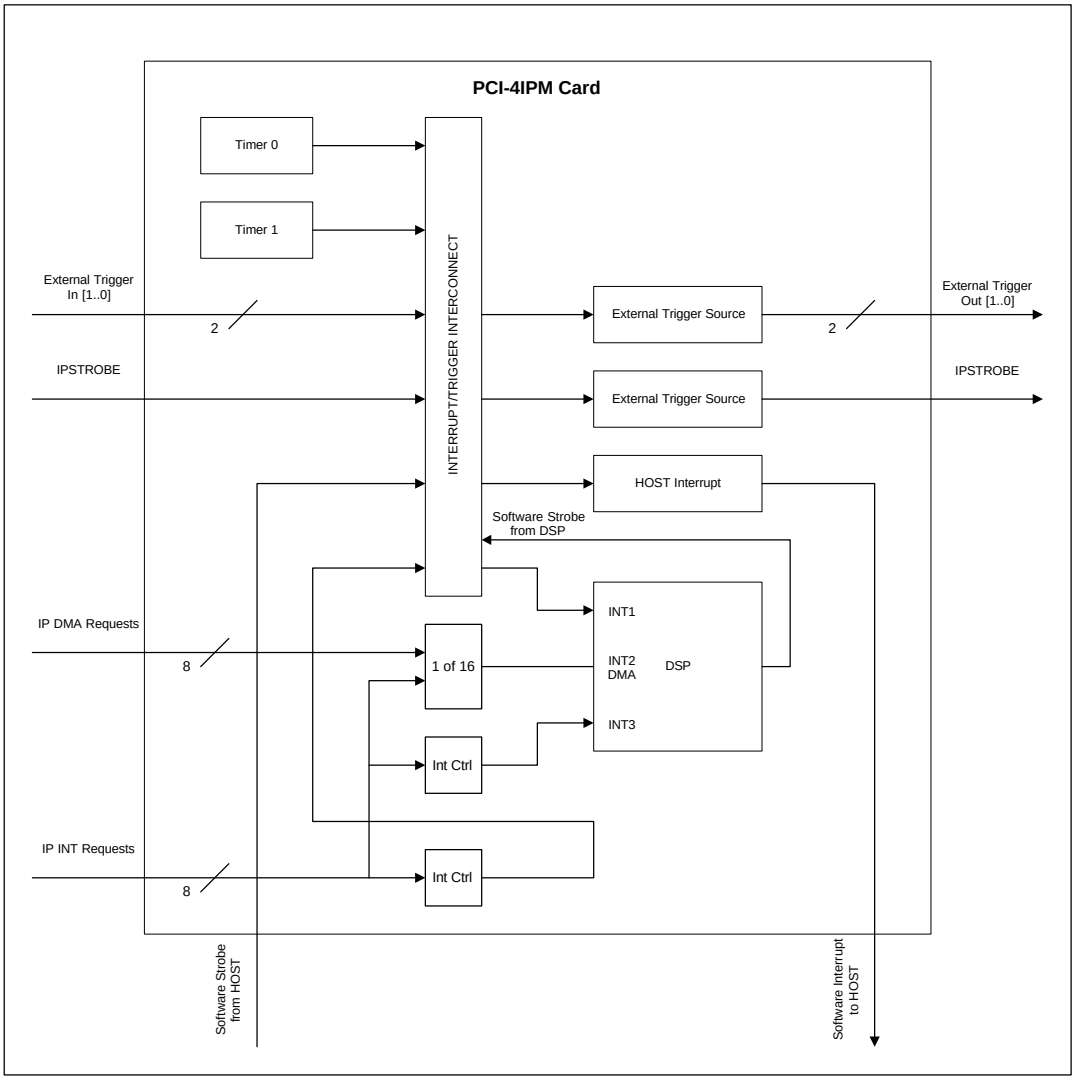

Figure 1.2: Trigger Clock / Gate Signal Block Diagram

It is easier to describe the sources and targets separately. Any source can be connected to any target, except for the External Trigger Outputs which are described as a special case.

1.3.1 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. (The DSP timers can also interrupt the DSP internal to the DSP.)

IP Interrupt
Any or all IP interrupts can be enabled to generate a trigger/interrupt source. Note that there is a preferred route to the DSP INT3 with independent interrupt enables. Additionally, one of the 8 interrupt lines or 8 DMA request lines can be selected to DSP INT2 for use by the DSP DMA engine.

IPSTROBE
An IP generated IPSTROBE can be buffered and routed to any target.

External Trigger Inputs
There are 2 external trigger lines on a header on the board, which may be used to synchronize multiple boards.

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 inter-board communication.

1.3.2 INTERRUPT / TRIGGER TARGETS

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

IPSTROBE
The IPSTROBE line to the IP modules can be triggered, allowing for synchronizing multiple IP modules.

HOST Interrupt
The HOST can be interrupted on a rising edge of the source. Additionally, the HOST can be interrupted directly by selected IP interrupts (separate from DSP IP interrupts).

DSP Interrupt
The DSP can be interrupted on a rising edge of the source. DSP INT1 is used for this.

1.3.3 EXTERNAL TRIGGER OUTPUTS

The card can source signals on the two external trigger lines from the following sources.

- DSP Timers
- IPSTROBE input
- IP Interrupt
- Trigger by DSP

- Trigger by HOST
- Discrete Logic Level

1.4 SOFTWARE SUPPORT

The **PCI-4IPM** 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.

1.5 REFERENCE MATERIALS LIST

PCI Local Bus Specification:

PCI Special Interest Group
Tel: (800) 433-5177
Tel: (503) 797-4207
Fax: (503) 234-6762

PXI Specification: PCI Extensions for Instrumentation:

PXI System Alliance
http://www.pxisa.org

PCI 9080 PCI Controller data book:

PLX Technology Corporation
Tel: (408) 744-9060
http://www.plxtech.com
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
http://www.bluewatersystems.com

2. PCI BRIDGE TO LOCAL DSP BUS

A PLX PCI-9080 bridge chip is used to interface the PCI Bus on the **PCI-4IPM** to the dual ported ram. This interface provides the card with the following features.

- HOST and other PCI peripherals have access to dual ported ram.
- Eight mailbox registers and two doorbell registers for communication between the HOST and the DSP.
- Two Bus Master DMA channels with scatter / gather support for transfers between the PCI bus and the DSP bus.
- Full interrupt support to HOST and DSP.

2.1 INTERFACE TO HOST (PCI)

All PCI 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 PCI specification.

All PCI 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 PCI CONFIGURATION SPACE

DSP Address: *0x880000 – 0x88000F*
PCI Address: *CONFIG:0x00 – 0x3C*
Mode of Access: *Read/Write*
Reset By *PCI Hardware Reset*

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

Offset Into PCI 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	PCI Latency Timer	Cache Line Size
0x10	0x880004	PCI Base Address 0 (Memory Access to PLX Registers)			
0x14	0x880005	PCI Base Address 1 (I/O Access to PLX Registers)			
0x18	0x880006	PCI Base Address 2 (Memory Access to DSP SRAM, IP IOSPACE, and other card registers)			
0x1C	0x880007	PCI Base Address 3 (Memory Access to IP MEMSPACE)			
0x20	0x880008	Unused PCI Base Address 4			
0x24	0x880009	Unused PCI Base Address 5			
0x28	0x88000A	Cardbus CIS Pointer (Not Supported)			
0x2C	0x88000B	Subsystem ID		Subsystem Vendor ID	
0x30	0x88000C	PCI Base Address for Expansion ROM			
0x34	0x88000D	Reserved			
0x38	0x88000E	Reserved			
0x3C	0x88000F	Max Latency	Min Grant	Interrupt Pin	Interrupt Line

Table 2.1: PCI Configuration Space

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

Register	Value (Meaning)
Vendor ID	0x13C5 (ALPHI Technology)
Device ID	0x0407
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	0x20000 - 40000(DPR)
Base Address 3 Size	N/A
Expansion ROM Size	None

Table 2.2: PCI Configuration Register Default Values

2.3 PCI BASE ADDRESS REGIONS

There are 4 base address regions available on the PLX PCI9080. The **PCI-4IPM** uses all 4 of these regions.

- BAR0: Memory access the PLX operation registers.
- BAR1: I/O access the PLX operation registers.
- BAR2: Passthru for the dual ported ram. The passthru is established at PCI reset when the DSP boots and enables it.
- BAR3 : N/A (but it is still Mapped in the Serial Prom)

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 PCI-9080 accesses the Local DSP Bus by placing the DSP in a HOLD state, performing any cycles necessary, and releasing the DSP from HOLD.

HOST Address	DSP Address	Data	R/W	Description
BAR2: 0x000000 – 0x07FFFF	0x000000 – 0x01FFFF	D31-D00	R/W	Local SRAM
BAR2: 0x080000 – 0x0FFFFF	0x020000 – 0x03FFFF	D31-D00	R/W	DPR
BAR0: 0x00 – 0xFF	0x880000 – 0x88003F	D31-D00	R/W	PLX Operation Registers
Not Accessible	0x900000 – 0x97FFFF	D07-D00	RO	Local FLASH
Not Accessible	0xC00000 – 0xFFFFFF	D31-D00	R/W	Pass through to PCI bus

Table 3.1: Local DSP Bus Overview

3.1 LOCAL SRAM

DSP Address: *0x000000 – 0x01FFFF, Bits 31 - 00*
PCI Address: *Not Accessible*
Mode of Access: *Read/Write*

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

The SRAM operates at zero wait states to the DSP.

3.2 DUAL PORT RAM AND ADDITIONAL LOCAL SRAM

DSP Address: *0x020000 - 0x03FFFF, Bits 31-00*
PCI Address: *BAR2:0x080000 - 0x0FFFFF, Bits 31-00*
Mode of Access: *Read/Write*

The dual ported ram is accessible to both the DSP and the HOST. The dual ported ram operates at one wait state from the DSP and one wait state from the host.

3.3 CARD CONTROL / STATUS REGISTERS

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

Most of these registers operate with two wait states to the DSP. Exceptions are access to the 8530 at 14 wait states, and any IP access which depends upon the IP clock, IP speed, and synchronization delays.

DSP Address	Data	R/W	Description
0x040000	D03-D00	R/W	9080 Control / Status Register
0x040008	N/A	WS	HOST Interrupt Acknowledge
0x040010	D07-D00	R/W	IP Control Register
0x040018	D01-D00	R	IP Status Register
0x040020	N/A	WS	Next DMA End A/B
0x040028	N/A	WS	Next DMA End C/D
0x040030	N/A	WS	Clear BERR
0x040038	N/A	WS	Software Strobe for manual clocking by DSP
0x040040	D07-D00	R/W	CLK / INT Configuration Regs
0x040048	D07-D00	R/W	8530 Serial Ports
0x040050	N/A	WS	Software Strobe for manual clocking by HOST
0x040070	D07-D00	R	DSP INT3 Pending Register
0x040071	D07-D00	R/W	DSP INT3 Enable Register
0x040072	D07-D00	R	HOST INT Pending Register
0x040073	D07-D00	R/W	HOST INT Enable Register
0x040074	D07-D00	R	IP DMA Pending Register
0x040075	D03-D00	R/W	DSP INT2/DMA Selection
0x040080	D07-D00	R	IP A INT0/1 Acknowledge
0x040084	D07-D00	R	IP B INT0/1 Acknowledge
0x040088	D07-D00	R	IP C INT0/1 Acknowledge
0x04008C	D07-D00	R	IP D INT0/1 Acknowledge
0x040200	D07-D00	R/W	IP A IOSPACE 16 bit
0x040240	D07-D00	R/W	IP A IOSPACE 16 bit DMA
0x040280	D07-D00	R/W	IP B IOSPACE 16 bit
0x0402C0	D07-D00	R/W	IP B IOSPACE 16 bit DMA
0x040300	D07-D00	R/W	IP A/B IOSPACE 32 bit
0x040340	D07-D00	R/W	IP A/B IOSPACE 32 bit DMA
0x040380	D07-D00	R/W	IP A IDSPACE 16 bit
0x0403C0	D07-D00	R/W	IP B IDSPACE 16 bit
0x040400	D07-D00	R/W	IP C IOSPACE 16 bit
0x040440	D07-D00	R/W	IP C IOSPACE 16 bit DMA
0x040480	D07-D00	R/W	IP D IOSPACE 16 bit
0x0404C0	D07-D00	R/W	IP D IOSPACE 16 bit DMA
0x040500	D07-D00	R/W	IP C/D IOSPACE 32 bit
0x040540	D07-D00	R/W	IP C/D IOSPACE 32 bit DMA
0x040580	D07-D00	R/W	IP C IDSPACE 16 bit
0x0405C0	D07-D00	R/W	IP D IDSPACE 16 bit

Table 3.2: Control / Status Registers

3.3.1 9080 CONTROL / STATUS REGISTER

DSP Address: *0x040000, Bits 03-00*
PCI Address: *Not Accessible*
Mode of Access: *Mixed*

This register allows for simplified determination of the cause of a DSP INT0 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

DSP Address: *0x040000, Bit 03*
PCI Address: *Not Accessible*
Mode of Access: *Read Only*
Reset By *See PLX Documentation*

This bit tracks the current state of the LSERR line from the PLX PCI-9080. 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

DSP Address: *0x040000, Bit 02*
PCI Address: *Not Accessible*
Mode of Access: *Read Only*
Reset By *See PLX Documentation*

This bit tracks the current state of the LINTO line from the PLX PCI-9080. 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

DSP Address: *0x040000, Bit 01*
PCI Address: *Not Accessible*
Mode of Access: *Read Only*
Reset By *See PLX Documentation*

This bit tracks the current state of the DMPAF line from the PLX PCI-9080. 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 PCI bus. See the PLX documentation for more details and for the enable bits.

LOCKOUT9080

DSP Address: *0x040000, Bit 00*
PCI Address: *Not Accessible*
Mode of Access: *Read/Write*
Reset By *PCI Hardware Reset, Software Reset, Watchdog Reset*

This bit, when set (1) will prevent the PLX PCI-9080 from seizing the dual ported ram. 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 PCI bus until it has been cleared (when an access to the card is performed).

3.3.2 HOST INTERRUPT ACKNOWLEDGE

DSP Address: *0x040008*
PCI Address: *Not Accessible*
Mode of Access: *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.

3.3.3 IP CONTROL REGISTER

DSP Address: *0x040010, Bits 07-00*
PCI Address: *Not Accessible*
Mode of Access: *Read/Write*

This register allows for setting the modes of access to the IPs.

BIT 07	BIT 06 – 04	BIT 03	BIT 02	BIT 01	BIT 00
IP MEM DMA	IP MEM CONTROL	IP BYTE LANE 1	IP BYTE LANE 0	IP C/D CLOCK	IP A/B CLOCK

IP MEM DMA

DSP Address: *0x040010, Bit 07*
PCI Address: *Not Accessible*
Mode of Access: *Read/Write*
Reset By *PCI Hardware Reset, Software Reset, Watchdog Reset*

This bit, when high (1), will cause an IP DMA cycle to be performed for IP memory cycles. When this bit is low (0), normal, non-DMA memory cycles will be performed.

IP MEM CONTROL

DSP Address: *0x040010, Bit 06 - 04*
PCI Address: *Not Accessible*
Mode of Access: *Read/Write*
Reset By *PCI Hardware Reset, Software Reset, Watchdog Reset*

These bits allow for selecting which IP memory space is to be accessible in the range of addresses 0x400000-0x7FFFFF. Choose from the following table:

BIT 06 – 04	Selected IP Memory Space
0	IP A, 16 bit, all 8 MB accessible.
1	IP B, 16 bit, all 8 MB accessible.
2	IP C, 16 bit, all 8 MB accessible.
3	IP D, 16 bit, all 8 MB accessible.
4	IP A/B, 32 bit, all 8 MB accessible.
5	IP C/D, 32 bit, all 8 MB accessible.
6	All 4 IPs, 16 bit, first 2 MB accessible from each.
7	All 4 IPs, 32 bit, first 4 MB accessible from each.

Options 4, 5, and 7 require that pairs of IPs be installed. They each will be accessed simultaneously, and the 16 bit words will be combined into one 32 bit DSP access.

Option 6 allows for accessing the first 2 MB of each IP, and maps it into the DSP / HOST memory space as follows:

HOST Address BAR3:	DSP Address	Data	R/W	Description
0x00000000	0x400000	D15-D00	R/W	IP A Memory Space
0x00400000	0x500000	D15-D00	R/W	IP B Memory Space
0x00800000	0x600000	D15-D00	R/W	IP C Memory Space
0x00C00000	0x700000	D15-D00	R/W	IP D Memory Space

Option 7 allows for accessing the first 4 MB of each IP, and maps it into the DSP / HOST memory space as follows:

HOST Address BAR3:	DSP Address	Data	R/W	Description
0x00000000	0x400000	D31-D00	R/W	IP A/B Memory Space
0x00800000	0x600000	D31-D00	R/W	IP C/D Memory Space

IP BYTE LANE 1

DSP Address: *0x040010, Bit 03*
PCI Address: *Not Accessible*
Mode of Access: *Read/Write*
Reset By *PCI Hardware Reset, Software Reset, Watchdog Reset*

This bit, when low (0), indicates that the high byte lane (D15 – D08) is to be included on any IP access. When high (1), the IP will ignore access to the high byte lane. NOTE: Normally this bit should never need to be set. However, there are a few IP modules which do not handle word write cycles correctly, and certain configuration registers must be written one byte at a time. By toggling this bit, BYTE accesses to the IPs can be performed.

IP BYTE LANE 0

DSP Address: *0x040010, Bit 02*
PCI Address: *Not Accessible*
Mode of Access: *Read/Write*
Reset By *PCI Hardware Reset, Software Reset, Watchdog Reset*

This bit, when low (0), indicates that the low byte lane (D07 – D00) is to be included on any IP access. When high (1), the IP will ignore access to the low byte lane. NOTE: Normally this bit should never need to be set. However, there are a few IP modules which do not handle word write cycles correctly, and certain configuration registers must be written one byte at a time. By toggling this bit, BYTE accesses to the IPs can be performed.

IP C/D CLOCK

DSP Address: *0x040010, Bit 01*
PCI Address: *Not Accessible*
Mode of Access: *Read/Write*
Reset By *PCI Hardware Reset, Software Reset, Watchdog Reset*

This bit, when low (0), will clock IP modules C and D at 8 MHz. When high (1), IP modules C and D will be clocked at 32 MHz.

IP A/B CLOCK

DSP Address: *0x040010, Bit 00*
PCI Address: *Not Accessible*
Mode of Access: *Read/Write*
Reset By *PCI Hardware Reset, Software Reset, Watchdog Reset*

This bit, when low (0), will clock IP modules A and B at 8 MHz. When high (1), IP modules A and B will be clocked at 32 MHz.

3.3.4 IP STATUS REGISTER

DSP Address: *0x040018*
PCI Address: *Not Accessible*
Mode of Access: *Read Only*

This register allows for checking the status of the IPs.

BIT 01	BIT 00
IP C/D BUS ERROR	IP A/B BUS ERROR

IP C/D BUS ERROR

DSP Address: *0x040018, Bit 01*
PCI Address: *Not Accessible*
Mode of Access: *Read/Write*
Reset By *PCI Hardware Reset, Software Reset, Watchdog Reset, Write to Clear BERR*

When this bit is set, an access to IP C or D resulted in a timeout. This would indicate a bad access on the IP. This bit is cleared by writing to Clear BERR.

IP A/B BUS ERROR

DSP Address: *0x040018, Bit 00*
PCI Address: *Not Accessible*
Mode of Access: *Read/Write*
Reset By *PCI Hardware Reset, Software Reset, Watchdog Reset, Write to Clear BERR*

When this bit is set, an access to IP A or B resulted in a timeout. This would indicate a bad access on the IP. This bit is cleared by writing to Clear BERR.

3.3.5 NEXT DMA END A/B

DSP Address: *0x040020*
PCI Address: *Not Accessible*
Mode of Access: *Write Strobe*

According to the IP specification, the final DMA access of a transfer should occur with /IPDMAEND active low.

By writing to this location prior to the final IP DMA transfer, the IP DMA access will occur with /IPDMAEND active low. It is restored inactive high after the access.

If no significance to /IPDMAEND is required by the IP, then this protocol can be ignored.

3.3.6 NEXT DMA END C/D

DSP Address: *0x040028*
PCI Address: *Not Accessible*
Mode of Access: *Write Strobe*

According to the IP specification, the final DMA access of a transfer should occur with /IPDMAEND active low.

By writing to this location prior to the final IP DMA transfer, the IP DMA access will occur with /IPDMAEND active low. It is restored inactive high after the access.

If no significance to /IPDMAEND is required by the IP, then this protocol can be ignored.

3.3.7 CLEAR BERR

DSP Address: *0x040030*
PCI Address: *Not Accessible*
Mode of Access: *Write Strobe*

If an IP access times out due to an invalid access or some other problem, a status bit in IP Status Register is set. Writing to this location will clear those bits.

3.3.8 SOFTWARE STROBE FOR MANUAL CLOCKING BY DSP

DSP Address: *0x040038*
PCI Address: *Not Accessible*
Mode of Access: *Write Strobe*

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

3.3.9 CLOCK / INT CONFIGURATION REGISTERS

DSP Address: *0x040040 – 0x40047, Bits 07 – 00*
PCI Address: *Not Accessible*
Mode of Access: *Mixed*
Reset By *PCI Hardware Reset, Software Reset, Watchdog Reset*

These registers control the trigger and gate signal routing inside the card, as well as allow for the control and monitoring of the EXT trigger lines. All of the registers are read/write, except for the EXT monitoring bits, which are read only.

If IPSTROBE is created by the carrier to the IPs, then Bit 04 IPSTROBE EN must be high (1). If the IP generates IPSTROBE, then Bit 04 should be low (0).

DSP Address	BITS 07-04	BITS 03-00
0x040040	Bit 04: IPSTROBE EN	IPSTROBE Source
0x040041	HOST Interrupt Source	N/A
0x040042	N/A	DSP Interrupt Source

Table 3.3: Clock / INT Source Registers

DSP Address	BITS 07-00
0x040043	External Trigger Direct Output

Table 3.4: Programmable Output Register

DSP Address	BIT 07	BITS 06-04	BIT 03	BITS 02-00
0x040044	EXT1 Mon	EXT1 Source	EXT0 Mon	EXT0 Source

Table 3.5: External Trigger Source Registers

Clock / INT Sources

DSP Address: *0x040040 – 0x40042, Bits 07 – 00*
PCI Address: *Not Accessible*
Mode of Access: *Read/Write*
Reset By *PCI Hardware Reset, Software Reset, Watchdog Reset*

The following signals can be selected from several sources.

- IPSTROBE Source
- HOST Interrupt
- DSP Interrupt

If IPSTROBE is created by the carrier to the IPs, then Bit 04 IPSTROBE EN must be high (1). If the IP generates IPSTROBE, then Bit 04 should be low (0).

BITS 03-00 BITS 07-04	Source
0x0	Low (0)
0x1	High (1)
0x2	DSP Timer 0
0x3	DSP Timer 1
0x4	IPSTROBE Input
0x5	HOST/EXT Interrupt logic
0x6	Strobe Register from HOST
0x7	Strobe Register from DSP
0x8	EXT Trigger 0
0x9	EXT Trigger 1
0xA	N/A
0xB	N/A
0xC	N/A
0xD	N/A
0xE	N/A
0xF	N/A

Table 3.6: Trigger / Gate Sources

EXT Trigger Direct Output

DSP Address: *0x040043, Bits 01 – 00*
PCI Address: *Not Accessible*
Mode of Access: *Read/Write*
Reset By *PCI Hardware Reset, Software Reset, Watchdog Reset*

The EXT Trigger can be driven to a specific logic state for gating or other applications.

If the **EXT Trigger Source** is set to EXT Trigger Direct Output, then the associated bit is output to the PXI Trigger line.

EXT Trigger Monitor

DSP Address: *0x040044, Bits 07and 03*
PCI Address: *Not Accessible*
Mode of Access: *Read Only*

These bits reflect the current state of the EXT Trigger lines.

EXT Trigger Source

DSP Address: *0x040044, Bits 06-04 and 02-00*
PCI Address: *Not Accessible*
Mode of Access: *Read/Write*
Reset By *PCI Hardware Reset, Software Reset, Watchdog Reset*

The PXI Trigger lines can be sourced from several sources, if desired.

BITS 02-00 BITS 06-04	Source
0x0	Line used as input or unused
0x1	EXT Trigger Direct Output
0x2	DSP Timer 0
0x3	DSP Timer 1
0x4	IPSTROBE Input
0x5	HOST/EXT Interrupt logic
0x6	Strobe Register from HOST
0x7	Strobe Register from DSP

Table 3.7: PXI Trigger Sources

3.3.10 8530 SERIAL PORTS

The **PCI-4IPM** includes an Z85C30 serial communications controller chip from ZILOG. There are two communications ports on this chip. Port 1 is connected to RS-232 drivers and are available on the board at P9. Port 2 is connected to RS-422 drivers, and are also available on the board at P12.

The Z85C30 is clocked at 7.3728 MHz, and allows for a multitude of common frequencies, including 115,200, 57,600, 38,400, and 19,200 baud.

The bootloader and hardware support libraries supplied with the **PCI-4IPM** utilizes the RS-232 port for a console for standard input and output by the DSP. The customer may alternatively wish to write his own software to use this port. Examples are provided in the **Board Support Package**.

There is hardware support for many RS422 and RS485 applications including HDLC, SDLC, and multidrop configurations. Clocking can be provided externally or internally.

The 85C30 can interrupt the DSP for servicing. The 85C30 is routed to INT1 via the interrupt control logic.

See documentation from ZILOG for more details about programming this device.

HOST Address BAR2:	DSP Address	BITS 07-00
0x100120	0x040048	Control Port for RS-422
0x100124	0x040049	Data Port for RS-422
0x100128	0x04004A	Control Port for RS-232
0x10012C	0x04004B	Data Port for RS-232

Table 3.8: Z85C30 Serial Port

3.3.11 SOFTWARE STROBE FOR MANUAL CLOCKING BY HOST

DSP Address: *0x040050*
PCI Address: *Not Accessible*
Mode of Access: *Write Strobe*

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

3.3.12 DSP INTERRUPT STATUS REGISTER

DSP Address: *0x040070, Bits 07-00*
PCI Address: *Not Accessible*
Mode of Access: *Read Only*

This register allows the DSP interrupt routine for INT3 to determine which of the IPs are causing an interrupt.

BIT 07	BIT 06	BIT 05	BIT 04	BIT 03	BIT 02	BIT 01	BIT 00
IP D, INT1	IP D, INT0	IP C, INT1	IP C, INT0	IP B, INT1	IP B, INT0	IP A, INT1	IP A, INT0

When the applicable bit is low (0), the IP is asserting the /INT line.

The interrupt routine must ensure that all interrupts have been cleared prior to returning. Since the DSP interrupt must be pulsed, the logic only generates a pulse when the transition from no interrupts are asserted, to at least one interrupt is asserted.

3.3.13 DSP INTERRUPT ENABLE REGISTER

DSP Address: *0x040071, Bits 07-00*
PCI Address: *Not Accessible*
Mode of Access: *Read/Write*
Reset By *PCI Hardware Reset, Software Reset, Watchdog Reset*

This register allows the DSP to select which interrupts should be allowed to generate INT3 on the DSP.

BIT 07	BIT 06	BIT 05	BIT 04	BIT 03	BIT 02	BIT 01	BIT 00
IP D, INT1 EN	IP D, INT0 EN	IP C, INT1 EN	IP C, INT0 EN	IP B, INT1 EN	IP B, INT0 EN	IP A, INT1 EN	IP A, INT0 EN

The appropriate bit should be set (1) to allow that IP interrupt to trigger the DSP.

3.3.14 HOST/EXT INTERRUPT STATUS REGISTER

DSP Address: *0x040072, Bits 07-00*
PCI Address: *Not Accessible*
Mode of Access: *Read Only*

This register allows the HOST/EXT interrupt routine to determine which of the IPs are causing an interrupt.

BIT 07	BIT 06	BIT 05	BIT 04	BIT 03	BIT 02	BIT 01	BIT 00
IP D, INT1	IP D, INT0	IP C, INT1	IP C, INT0	IP B, INT1	IP B, INT0	IP A, INT1	IP A, INT0

When the applicable bit is low (0), the IP is asserting the /INT line.

3.3.15 HOST/EXT INTERRUPT ENABLE REGISTER

DSP Address: *0x040073, Bits 07-00*
PCI Address: *Not Accessible*
Mode of Access: *Read/Write*
Reset By *PCI Hardware Reset, Software Reset, Watchdog Reset*

This register allows the HOST/EXT to select which interrupts should be allowed to generate an interrupt.

BIT 07	BIT 06	BIT 05	BIT 04	BIT 03	BIT 02	BIT 01	BIT 00
IP D, INT1 EN	IP D, INT0 EN	IP C, INT1 EN	IP C, INT0 EN	IP B, INT1 EN	IP B, INT0 EN	IP A, INT1 EN	IP A, INT0 EN

The appropriate bit should be set (1) to allow that IP interrupt to trigger the interrupt.

3.3.16 IP DMA STATUS REGISTER

DSP Address: *0x040074, Bits 07-00*
PCI Address: *Not Accessible*
Mode of Access: *Read Only*

This register allows reading which IP DMA requests are pending. It is provided so that the current DMA status is accessible for debugging, as well as if a DSP interrupt routine is required to perform IP DMA.

BIT 07	BIT 06	BIT 05	BIT 04	BIT 03	BIT 02	BIT 01	BIT 00
IP D, DMA1	IP D, DMA0	IP C, DMA1	IP C, DMA0	IP B, DMA1	IP B, DMA0	IP A, DMA1	IP A, DMA0

When the applicable bit is low (0), the IP is asserting the /DMA request line.

3.3.17 IP DMA SELECTION REGISTER

DSP Address: *0x040075, Bits 03-00*
PCI Address: *Not Accessible*
Mode of Access: *Read / Write*
Reset By *PCI Hardware Reset, Software Reset, Watchdog Reset*

This register allows the selection of which DMA request line or which INT request line will generate DSP INT2. If the use of the DSP DMA engine is to be used in a demand mode, this register determines which source causes the DMA request.

Select one of the following sources:

BIT 03 – 00	INT2 Selection
0	IP A, DMA0
1	IP A, DMA1
2	IP B, DMA0
3	IP B, DMA1
4	IP C, DMA0
5	IP C, DMA1
6	IP D, DMA0
7	IP D, DMA1
8	IP A, INT 0
9	IP A, INT 1
A	IP B, INT 0
B	IP B, INT 1
C	IP C, INT 0
D	IP C, INT 1
E	IP D, INT 0
F	IP D, INT1

3.3.18 IP A INTERRUPT ACKNOWLEDGE

DSP Address: *0x040080 or 0x040081, Bits 07-00*
PCI Address: *Not Accessible*
Mode of Access: *Read / Write*

Reading from this location generates a valid IP Interrupt Acknowledge cycle to IP A. 0x40080 issues an acknowledge to IP INT0, and 0x40081 issues an acknowledge to IP INT1.

Note that this is not required by the carrier, and is only supported for those IP modules requiring it.

3.3.19 IP A INTERRUPT ACKNOWLEDGE

DSP Address: *0x040084 or 0x040085, Bits 07-00*
PCI Address: *Not Accessible*
Mode of Access: *Read / Write*

Reading from this location generates a valid IP Interrupt Acknowledge cycle to IP B. 0x40084 issues an acknowledge to IP INT0, and 0x40085 issues an acknowledge to IP INT1.

Note that this is not required by the carrier, and is only supported for those IP modules requiring it.

3.3.20 IP C INTERRUPT ACKNOWLEDGE

DSP Address: *0x040088 or 0x040089, Bits 07-00*
PCI Address: *Not Accessible*
Mode of Access: *Read / Write*

Reading from this location generates a valid IP Interrupt Acknowledge cycle to IP C. 0x40088 issues an acknowledge to IP INT0, and 0x40089 issues an acknowledge to IP INT1.

Note that this is not required by the carrier, and is only supported for those IP modules requiring it.

3.3.21 IP D INTERRUPT ACKNOWLEDGE

DSP Address: *0x04008C or 0x04008D, Bits 07-00*
PCI Address: *Not Accessible*
Mode of Access: *Read / Write*

Reading from this location generates a valid IP Interrupt Acknowledge cycle to IP D. 0x4008C issues an acknowledge to IP INT0, and 0x4008D issues an acknowledge to IP INT1.

Note that this is not required by the carrier, and is only supported for those IP modules requiring it.

3.3.22 IP A 16 BIT IO ACCESS

DSP Address: *0x040200 – 0x4023F, Bits 15-00*
PCI Address: *Not Accessible*
Mode of Access: *Read / Write*

Access to these locations generates a valid IP IO SPACE cycles to IP A. Since this access is only 16 bits wide, the upper bits should be masked off in the DSP. Also, WORD access by the HOST will show that only addresses evenly divisible by 4 will return the correct information. In other words, every other WORD accesses consecutive IP addresses.

3.3.23 IP A 16 BIT IO ACCESS WITH DMA ACKNOWLEDGE

DSP Address: *0x040240 – 0x4027F, Bits 15-00*
PCI Address: *Not Accessible*
Mode of Access: *Read / Write*

Access to these locations generates a valid IP IO SPACE cycles with DMA Acknowledge to IP A. Since this access is only 16 bits wide, the upper bits should be masked off in the DSP. Also, WORD access by the HOST will show that only addresses evenly divisible by 4 will return the correct information. In other words, every other WORD accesses consecutive IP addresses.

3.3.24 IP B 16 BIT IO ACCESS

DSP Address: *0x040280 – 0x402BF, Bits 15-00*
PCI Address: *Not Accessible*
Mode of Access: *Read / Write*

Access to these locations generates a valid IP IO SPACE cycles to IP B. Since this access is only 16 bits wide, the upper bits should be masked off in the DSP. Also, WORD access by the HOST will show that only addresses evenly divisible by 4 will return the correct information. In other words, every other WORD accesses consecutive IP addresses.

3.3.25 IP B 16 BIT IO ACCESS WITH DMA ACKNOWLEDGE

DSP Address: *0x0402C0 – 0x402FF, Bits 15-00*
PCI Address: *Not Accessible*
Mode of Access: *Read / Write*

Access to these locations generates a valid IP IO SPACE cycles with DMA Acknowledge to IP B. Since this access is only 16 bits wide, the upper bits should be masked off in the DSP. Also, WORD access by the HOST will show that only addresses evenly divisible by 4 will return the correct information. In other words, every other WORD accesses consecutive IP addresses.

3.3.26 IP A/B 32 BIT IO ACCESS

DSP Address: *0x040300 – 0x4033F, Bits 31-00*
PCI Address: *Not Accessible*
Mode of Access: *Read / Write*

Access to these locations generates a valid IP IO SPACE cycles to both IPs A and B. In this way, two identical IP modules could be populated in A and B, and throughput would be improved by writing or reading 32 bits at a time.

3.3.27 IP A/B 32 BIT IO ACCESS WITH DMA ACKNOWLEDGE

DSP Address: *0x040340 – 0x4037F, Bits 31-00*
PCI Address: *Not Accessible*
Mode of Access: *Read / Write*

Access to these locations generates a valid IP IO SPACE cycles with DMA Acknowledge to both IPs A and B. In this way, two identical IP modules could be populated in A and B, and throughput would be improved by writing or reading 32 bits at a time.

3.3.28 IP A 16 BIT ID ACCESS

DSP Address: *0x040380 – 0x403BF, Bits 15-00*
PCI Address: *Not Accessible*
Mode of Access: *Read / Write*

Access to these locations generates a valid IP ID SPACE cycles to IP A. Since this access is only 16 bits wide, the upper bits should be masked off in the DSP. Also, WORD access by the HOST will show that only addresses evenly divisible by 4 will return the correct information. In other words, every other WORD accesses consecutive IP addresses.

3.3.29 IP B 16 BIT ID ACCESS

DSP Address: *0x0403C0 – 0x403FF, Bits 15-00*
PCI Address: *Not Accessible*
Mode of Access: *Read / Write*

Access to these locations generates a valid IP ID SPACE cycles to IP B. Since this access is only 16 bits wide, the upper bits should be masked off in the DSP. Also, WORD access by the HOST will show that only addresses evenly divisible by 4 will return the correct information. In other words, every other WORD accesses consecutive IP addresses.

3.3.30 IP C IO ACCESS

DSP Address: *0x040400 – 0x4043F, Bits 15-00*
PCI Address: *Not Accessible*
Mode of Access: *Read / Write*

Access to these locations generates a valid IP IO SPACE cycles to IP C. Since this access is only 16 bits wide, the upper bits should be masked off in the DSP. Also, WORD access by the HOST will show that only addresses evenly divisible by 4 will return the correct information. In other words, every other WORD accesses consecutive IP addresses.

3.3.31 IP C IO ACCESS WITH DMA ACKNOWLEDGE

DSP Address: *0x040440 – 0x4047F, Bits 15-00*
PCI Address: *Not Accessible*
Mode of Access: *Read / Write*

Access to these locations generates a valid IP IO SPACE cycles with DMA Acknowledge to IP C. Since this access is only 16 bits wide, the upper bits should be masked off in the DSP. Also, WORD access by the HOST will show that only addresses evenly divisible by 4 will return the correct information. In other words, every other WORD accesses consecutive IP addresses.

3.3.32 IP D IO ACCESS

DSP Address: *0x040480 – 0x404BF, Bits 15-00*
PCI Address: *Not Accessible*
Mode of Access: *Read / Write*

Access to these locations generates a valid IP IO SPACE cycles to IP D. Since this access is only 16 bits wide, the upper bits should be masked off in the DSP.

Also, WORD access by the HOST will show that only addresses evenly divisible by 4 will return the correct information. In other words, every other WORD accesses consecutive IP addresses.

3.3.33 IP D IO ACCESS WITH DMA ACKNOWLEDGE

DSP Address: *0x0404C0 – 0x404FF, Bits 15-00*
PCI Address: *Not Accessible*
Mode of Access: *Read / Write*

Access to these locations generates a valid IP IO SPACE cycles with DMA Acknowledge to IP D. Since this access is only 16 bits wide, the upper bits should be masked off in the DSP. Also, WORD access by the HOST will show that only addresses evenly divisible by 4 will return the correct information. In other words, every other WORD accesses consecutive IP addresses.

3.3.34 IP C/D 32 BIT IO ACCESS

DSP Address: *0x040500 – 0x4053F, Bits 31-00*
PCI Address: *Not Accessible*
Mode of Access: *Read / Write*

Access to these locations generates a valid IP IO SPACE cycles to both IPs C and D. In this way, two identical IP modules could be populated in C and D, and throughput would be improved by writing or reading 32 bits at a time.

3.3.35 IP C/D 32 BIT IO ACCESS WITH DMA ACKNOWLEDGE

DSP Address: *0x040540 – 0x4057F, Bits 31-00*
PCI Address: *Not Accessible*
Mode of Access: *Read / Write*

Access to these locations generates a valid IP IO SPACE cycles with DMA Acknowledge to both IPs C and D. In this way, two identical IP modules could be populated in C and D, and throughput would be improved by writing or reading 32 bits at a time.

3.3.36 IP C 16 BIT ID ACCESS

DSP Address: *0x040580 – 0x405BF, Bits 15-00*
PCI Address: *Not Accessible*
Mode of Access: *Read / Write*

Access to these locations generates a valid IP ID SPACE cycles to IP C. Since this access is only 16 bits wide, the upper bits should be masked off in the DSP. Also, WORD access by the HOST will show that only addresses evenly divisible by 4 will return the correct information. In other words, every other WORD accesses consecutive IP addresses.

3.3.37 IP D 16 BIT ID ACCESS

DSP Address: *0x0405C0 – 0x405FF, Bits 15-00*
PCI Address: *Not Accessible*
Mode of Access: *Read / Write*

Access to these locations generates a valid IP ID SPACE cycles to IP D. Since this access is only 16 bits wide, the upper bits should be masked off in the DSP. Also, WORD access by the HOST will show that only addresses evenly divisible by 4 will return the correct information. In other words, every other WORD accesses consecutive IP addresses.

3.4 LOCAL FLASH

DSP Address: *0x900000 - 0x97FFFF, Bits 07-00*
PCI Address: *Not Accessible*
Mode of Access: *Read/Write in page mode*

There is 512k x 8 of FLASH memory available to the DSP on the Local Bus. The first 64k 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.

Code examples of writing to this device is located in the Board Support Package.

3.5 PLX OPERATION REGISTERS

DSP Address: *0x880000 – 0x88005E*
PCI Address: *BAR0/1:0x0 - 0xF8*
Mode of Access: *Read/Write*
Reset By *Depends upon register*

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_2O message queues, so these registers are not described.

3.5.1 LOCAL CONFIGURATION REGISTERS

DSP Address: *0x880020 – 0x88005E*
PCI Address: *BAR0/1:0x0 – 0xF8*
Mode of Access: *Read/Write*
Reset By *PCI Hardware Reset, Software Reset, Watchdog Reset*

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 PCI to Local Address Space 0			
0x04	0x880021	Local Base Address for PCI to Local Address Space 0			
0x08	0x880022	Mode / Arbitration Register			
0x0C	0x880023	Big / Little Endian Register			
0x10	0x880024	Range for PCI to Local Expansion ROM			
0x14	0x880025	Local Base Address for PCI to Local Expansion ROM			
0x18	0x880026	Local Bus Region Descriptors			
0x1C	0x880027	Range for Direct Master to PCI			
0x20	0x880028	Local Base Address for Direct Master to PCI Memory			
0x24	0x880029	Local Base Address for Direct Master to PCI IO/CFG			
0x28	0x88002A	PCI Base Address for Direct Master to PCI			
0x2C	0x88002B	PCI Configuration Address for Direct Master to PCI IO/CFG			
0xF0	0x88005C	Range for PCI to Local Address Space 1			
0xF4	0x88005D	Local Base Address for PCI to Local Address Space 1			
0xF8	0x88005E	Local Bus Region Descriptor for PCI to Local			

Table 3.9: PLX Local Configuration Registers

3.5.2 RUNTIME REGISTERS

DSP Address: *0x880030 – 0x88003F*
PCI Address: *BAR0/1:0x40 – 0x7C*
Mode of Access: *Read/Write*
Reset By *PCI Hardware Reset*

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	PCI to Local Doorbell Register			
0x64	0x880039	Local to PCI Doorbell Register			
0x68	0x88003A	Interrupt Control / Status			
0x6C	0x88003B	Serial EEPROM, PCI 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 3.10: PLX Runtime Registers

3.5.3 DMA REGISTERS

DSP Address: *0x880040 – 0x88004C*
PCI Address: *BAR0/1:0x80 – 0xB0*
Mode of Access: *Read/Write*
Reset By *PCI Hardware Reset, Software Reset, Watchdog Reset*

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 PCI 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 PCI 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 3.11: PLX DMA Registers

3.6 PCI PASS-THROUGH REGION

DSP Address: *0xC00000 – 0xFFFFFF*
Mode of Access: *Read/Write*

The **PCI-4IPM** can directly access HOST memory and I/O ports on the PCI system and the PMC Module through correctly programming the PLX interface. Up to 16 Mbytes of PCI address space can be mapped at a time.

NOTE: Windows NT/2000 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 DMA

There are a total of four DMA channels available on the **PCI-4IPM**. Two are located in the TMS320C32 DSP and two are located in the PLX-9080 PCI to DSP bridge.

4.1.1 DSP DMA

The DMA channels in the DSP can be used for such purposes as:

- Transferring data between IP peripherals and SRAM memory.

These DMA engines can be free-running, or they can be triggered from several DSP interrupt sources, such as:

- Selected IP DMA or INT request. (DSP INT2)
- External trigger event, such as EXT trigger, external PCI device writing to a strobe register. (DSP INT1, source selected in the **Clock / INT Configuration Registers**)
- DSP timer causing a trigger. (Internal to DSP)

See the DSP Processor documentation for more details.

4.1.2 DSP TO PCI BRIDGE DMA

The PLX 9080 DSP to PCI bridge contains two DMA engines for transferring between the dual ported ram and the PCI bus. These DMA engines are the most efficient means of transferring data between the PCI host and dual ported ram.

The DMA engines are capable of processing a chained list of transfer requests, and can generate interrupts to the DSP or to the HOST processor with each transfer. Scatter/Gather is supported with these chains of requests. The list of requests can reside in PCI memory or in dual ported ram on the card.

4.1.3 DEMAND MODE DMA

The PLX-9080 is not configured to support Demand DMA on this product.

4.2 EXTERNAL TRIGGERING

The **PCI-4IPM** is designed to make use of two EXT trigger lines for communication and synchronization between other **PCI-4IPM** boards and other PCI compatible boards.

The **PCI-4IPM** is capable of driving the following signals to an EXT line:

- A constant 0 or 1, or a signal created by the DSP or PCI device toggling the bit which is output to the EXT trigger line.
- A timing source generated by either of the DSP timers.
- IPSTROBE generated by an IP module.

- One or more selected IP interrupts.
- A pulse generated by a write to a register by either the DSP or another PCI device.

Of course, the trigger output can be tri-stated when it is not used or is for input.

The **PCI-4IPM** is capable of performing the following actions based on a PXI trigger:

- Generating IPSTROBE to the IP modules.
- Provide an interrupt to the HOST.
- Provide an interrupt to the DSP.
- Provide a trigger of the DSP DMA engines.
- Can be read by the DSP or PCI processor to determine its current state.

4.3 INTERRUPTS

4.3.1 DSP INTERRUPTS

The TMS320C32 DSP processor has four external interrupt sources. They are called INT0 – INT3. The sources of these interrupts are described below.

INT0
INT0 is dedicated to handling PLX9080 issues. This assertion of either LINTO# or LSERR# by the PLX will generate INT0. The DSP can query the **9080 Control / Status Register** bits **LINTO** and **LSERR** to determine the cause of the interrupt, and determine that both signals are HIGH before returning from the interrupt handler.

Individual bits in the PLX will enable the following sources:

- DMA Completion. (**LINTO**)
- DMA Terminal Count Reached. (**LINTO**)
- One or more bits in Doorbell register are set. (**LINTO**)
- PCI write to a Mailbox. (**LINTO**)
- Master Abort on PCI bus. (**LSERR**)
- Target Abort on PCI bus. (**LSERR**)
- Retry count exceeded on PCI bus. (**LSERR**)
- Parity Error on PCI bus. (**LSERR**)

INT1
INT1 has two purposes. One is to handle any Z85C30 serial events requiring DSP interrupt support. The other is dedicated to handling an interrupt from the source selected in the **Clock / INT Configuration Registers**. The following sources can be connected to this interrupt:

- Rising edge on an EXT trigger line.
- External PCI device writing to one of the strobe registers.
- IPSTROBE from any IP.
- Enabled IP interrupt.

This interrupt line can be used to trigger a DSP DMA engine, or be handled as an interrupt, or both.

INT2

INT2 is designed to be utilized for DSP DMA on demand. It can route any IP DMA line or any IP Interrupt line to INT2. This interrupt line can be used to trigger a DSP DMA engine, or be handled as an interrupt, or both.

INT3

INT3 is dedicated to handling IP Interrupts. Any or all IP interrupts can be enabled to generate this interrupt. A status register can be read to determine which interrupts are pending.

This interrupt line can be used to trigger a DSP DMA engine, or be handled as an interrupt, or both.

4.3.2 HOST INTERRUPTS

The PLX can interrupt the PCI HOST from any of the following causes.

- DMA Completion.
- DMA Terminal Count Reached.
- One or more bits in Doorbell register are set.
- Master Abort on PCI bus.
- Target Abort on PCI bus.
- Retry count exceeded on PCI bus.
- Parity Error on PCI bus.

Individual bits in the PLX will enable the sources. Additionally, the following hardware source can generate a PCI HOST interrupt.

- HOST interrupt source selected in the **Clock / INT Configuration Registers**.
- One or more selected IP interrupts. A status register can be read to determine the pending interrupt cause.

4.4 RESET

The **PCI-4IPM** is reset by several different means.

- PCI Hardware RESET
- Software RESET from a PCI device
- Watchdog RESET

4.4.1 PCI HARDWARE RESET

The entire card is reset when the PCI reset line is held low by the PCI bus. All of the card's registers are reset as described in this manual, and the IPs will also be reset. The PLX is fully reset, and the NVRAM image is automatically read by the PLX.

All access by the PCI bus is locked out until either the DSP sets the **Local Init Status** bit in the **CNTRL** register of the PLX.

4.4.2 SOFTWARE RESET FROM A PCI DEVICE

A PCI device can reset the DSP, the card registers, and partially reset the PLX. The reset is accomplished by setting the **PCI Adapter Software Reset** bit in the PLX **CNTRL** register to a 1. After 200 mS, this bit needs to be set to a 0.

The DSP will be forced into reset, the card registers will reset, the IPs will be reset, and the PLX Local Configuration and PLX DMA registers will be reset. This will stop any active DMA, and will eliminate the passthru regions BAR2 and BAR3. Any access to BAR2 or BAR3 by the PCI bus will be issued a PCI Disconnect with Retry. Behavior of accessing a region which issues Disconnect with Retry is machine dependent, but most PC chipsets will retry indefinitely, appearing as though the system locks up.

After the reset is complete, the DSP is free to run, which will reload the PLX registers from the NVRAM, thus re-enabling the passthru regions.

4.4.3 WATCHDOG RESET

The **PCI-4IPM** incorporates a watchdog timer, which ensures proper RESET timing on power-up and when jumper W6 is shorted.

The DSP will be forced into reset, the card registers will reset, the IP modules will be reset, and the PLX Local Configuration and PLX DMA registers will be reset. This will stop any active DMA, and will eliminate the passthru region BAR2 and BAR3. Any access to BAR2 or BAR3 by the PCI bus will be issued a PCI Disconnect with Retry. Behavior of accessing a region which issues Disconnect with Retry is machine dependent, but most PC chipsets will retry indefinitely, appearing as though the system locks up.

After the reset is complete, the DSP is free to run, which will reload the PLX registers from the NVRAM. This will re-enable the passthru regions.

4.5 CONNECTORS, JUMPERS, AND LEDS

The jumper and connector placement is depicted below.

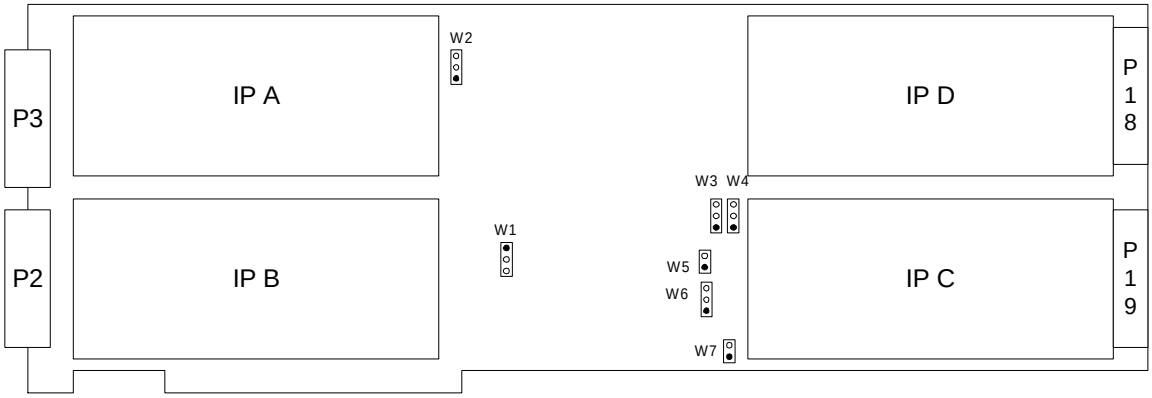

Figure 4.1: Jumper Locations

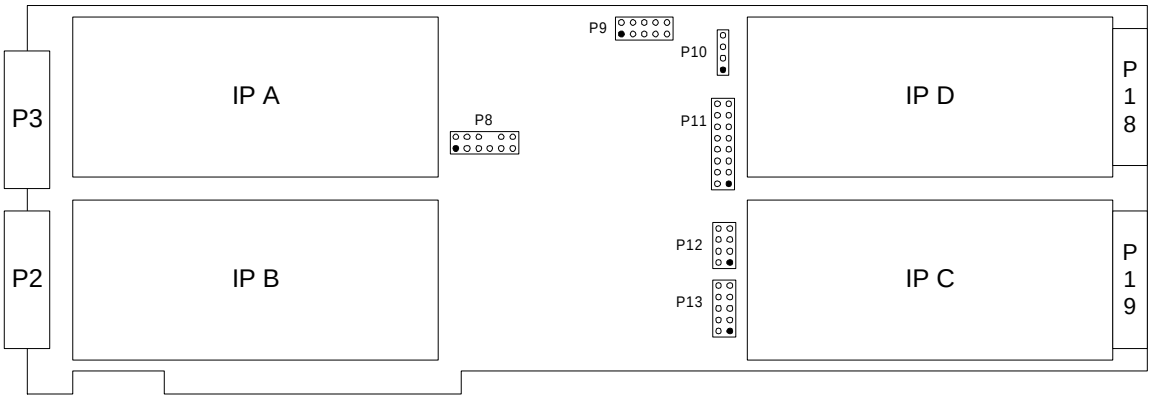

Figure 4.2: Connector Locations

4.5.1 JUMPER DESCRIPTIONS

JUMPER	FACTORY SETTING	DESCRIPTION
W1	1-2	Provides clock for onboard Altera 10K20
W2	1-2	Select Eprom size. Default 29C010 and larger.
W3	2-3	Select output to RS422 pair 4 and 5. 1-2 to output TXCLK generated by 8530. 2-3 to output RTS from the 8530. RTS is default.
W4	2-3	Select input from RS422 pair 7 and 8. 1-2 to connect to receive clock of 8530; 2-3 to connect to CTS of 8530. CTS is default.
W5	1-2	RS422 Driver Mode. If present, drive both outputs on RTS; otherwise drive always. Drive always is default.
W6	2-3	Provides clock for watchdog timer
W7	None	Provides an identical RESET to that of the watchdog timer.

Table 4.1 Jumper Descriptions

4.5.2 HEADER DESCRIPTIONS

JUMPER	DESCRIPTION
J1	For programming onboard Altera device
P8	C32 DSP emulator connection port.
P9	RS232 connection port.
P10	1-2 for enabling TTL trigger 0 3-4 for enabling TTL trigger 1
P11	DSP serial comm. port
P12	RS422 connection port
P13	For programming onboard Altera device.

4.5.3 FRONT PANEL DESCRIPTION

The following picture shows the front panel.

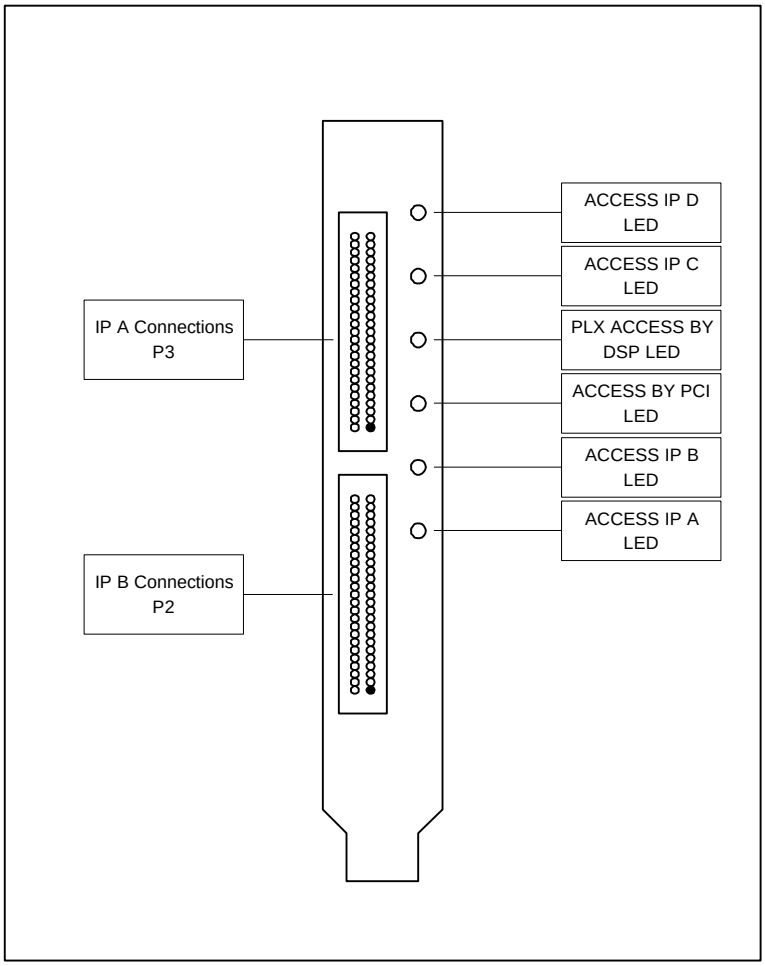

Figure 4.3: Front Panel

4.5.4 LED INDICATORS

There are six LED indicators visible at the PCI card bracket. They are not marked with a legend on the bracket, but they are labeled on the PCB as L1 – L6 where L1 is at the top of the card.

The LEDs have the following meanings:

LED	Meaning
L1	DSP is accessing IP D.
L2	DSP is accessing IP C.
L3	PLX access by DSP.
L4	Access by PCI to DSP Bus.
L5	DSP is accessing IP B.
L6	DSP is accessing IP A.

Table 4.2 LED Descriptions

4.5.5 IP I/O CONNECTORS

50 pin flat cable connectors are used to route all the IP I/O signals off the card. The mating connector reference is **HIROSE H1F6-50D-1.27R.** ALPHI Technology can supply transition modules and cable assemblies to meet any special requirements. Contact the factory for additional information.

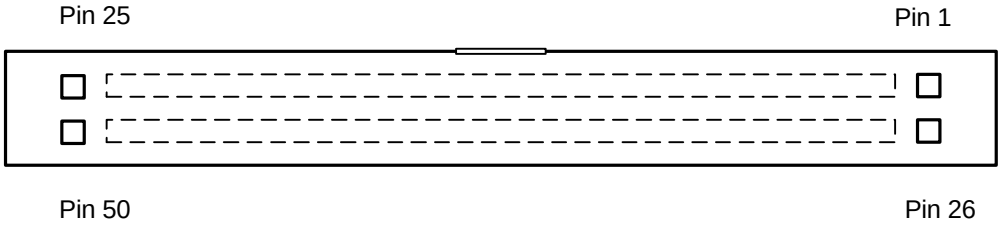

Figure 4.1: IP I/O CONNECTORS

IP A (P3) and B (P2)

The I/O signals for IPs A and B are directly routed off the card through the front panel.

IP C (P19) and D (P18)

The I/O signals for IPs C and D are placed such that they can only be accessed from inside the HOST chassis. ALPHI Technology has an optional panel and ribbon cable assembly which can be placed adjacent to the card to route these signals out of the HOST chassis.

4.6 SERIAL RS232 PORT (P9)

Port A of the 8530 is configured as RS232 port, and it serves as the console output for the bootloader and any DSP programs linked with the ALPHI_IO.LIB library discussed in the **DSP Support Library**. The pinout is described in the table below. An optional serial cable is available from the factory which connects to CON11 and terminates in a 9 PIN D, suitable for connection to a computer or terminal.

Pin	Description	Pin	Description
1	No Connection	2	No Connection
3	Receive Data	4	Ground
5	Transmit Data	6	No Connection
7	Clear To Send	8	No Connection
9	Request To Send	10	No Connection

Table 4.1: Serial RS232 Port (P9)

4.7 SERIAL RS422 / RS485 PORT (P12)

Port B of the 8530 is configured as a combination RS422 / RS485 port. The port can be configured to communicate with many other devices in a dedicated direct

link as well as a multidrop link, depending on jumper settings. The pinout is described in the table below.

Pin	Direction	Description	Pin	Direction	Description
1	Output	Transmit Data +	2	Output	Transmit Data -
3	Input	Receive Data +	4	Output	RTS + or TXCLK +
5	Output	RTS – or TXCLK -	6	Input	Receive Data -
7	Input	CTS + or RXCLK +	8	Input	CTS - or RXCLK -

Table 4.1: Serial RS422 / RS485 Port (P12)

4.8 DSP SERIAL PORT (P11)

The serial shift register of the DSP is also available for use as desired by the customer. The input and outputs are connected to 55194 and 55195 bipolar line drivers and receivers as shown in the following table. See the DSP Processor manual for details on how to use the serial port.

Pin	Description	Pin	Description
1	FSR0 -	2	FSX0 +
3	FSR0 +	4	FSX0 -
5	Ground	6	Ground
7	DR0 -	8	DX0 +
9	DR0 +	10	DX0 -
11	Ground	12	Ground
13	CLKR0 -	14	CLKX0 +
15	CLKR0 +	16	CLKX0 -

Table 4.1: DSP Serial Port (P11)

4.9 EXT CLOCK / INT CONNECTOR (P10)

This connector is used to connect between boards in order to share interrupts, clocks, or IPSTROBE among several boards. There are two external trigger signals, which may be configured individually as input or output.

Pin	Description	Pin	Description
1	EXT0	2	GND
3	EXT1	4	GND

Table 4.1: EXT Clock / INT (P10)

4.10 EMULATOR CONNECTION (P8)

This connector is used to connect the emulator to the C32 DSP. It follows the standard form as described by TI in their processor manual.

4.11 POWER FOR STAND ALONE OPERATION (PW1)

When the board is operating in stand alone mode, this connector can be used in place of the PCI connector to supply power to the card and IPs. The card requires +5 volts; the +12 and –12 volts are only necessary if the installed IPs require it. Connector available upon request.

Pin	Use
1	+5 Volts
2	Ground
3	+12 Volts
4	Ground
5	–12 Volts

Table 4. 1: Power for Stand Alone Operation (PW1)

4.12 FACTORY USE (P13)

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

