PCI-HPDI32-ASYNC

PMC-HPDI32-ASYNC

cPCI-HPDI32-ASYNC

PC104P-HPDI32-ASYNC

User Manual

Preliminary

General Standards Corporation
8302A Whitesburg Drive
Huntsville, AL 35802
Phone: (256) 880-8787
Fax: (256) 880-8788
URL: www.generalstandards.com
E-mail: support@generalstandards.com

PREFACE

General Standards Corporation
Preliminary, Revised: July 20, 2000

Additional copies of this manual or other literature may be obtained from:

General Standards Corporation
8302A Whitesburg Dr.
Huntsville, Alabama 35802
Tele: (256) 880-8787
FAX: (256) 880-8788
E-mail: support@generalstandards.com

The information in this document is subject to change without notice.

General Standards Corp. makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Although extensive editing and reviews are performed before release to ECO control, **General Standards Corp.** assumes no responsibility for any errors that may exist in this document. No commitment is made to update or keep current the information contained in this document.

General Standards Corp. does not assume any liability arising out of the application or use of any product or circuit described herein, nor is any license conveyed under any patent rights or any rights of others.

General Standards Corp. assumes no responsibility for any consequences resulting from omissions or errors in this manual, or from the use of information contained herein.

General Standards Corp. reserves the right to make any changes, without notice, to this product to improve reliability, performance, function, or design.

This user's manual provides information on the specifications, theory of operation, register level programming, installation of the board and information required for customized hardware/software development.

RELATED PUBLICATIONS

The following manuals and specifications provide the necessary information for in depth understanding of the specialized parts used on this board.

EIA Standard for the RS-422A Interface (EIA order number EIA-RS-422A)

PCI Local Bus Specification Revision 2.1 June 1, 1995. Questions regarding the PCI specification be forwarded to:

PCI Special Interest Group
P.O. Box 14070
Portland, OR 97214
(800) 433- 5177 (U.S.)
(503) 797-4207 (International)
(503) 234-6762 (FAX)

PCI-HPDI32A-ASYNC Documentation History

1) October 16, 2001 – Initial Release.

Table of Contents

Table of Figures

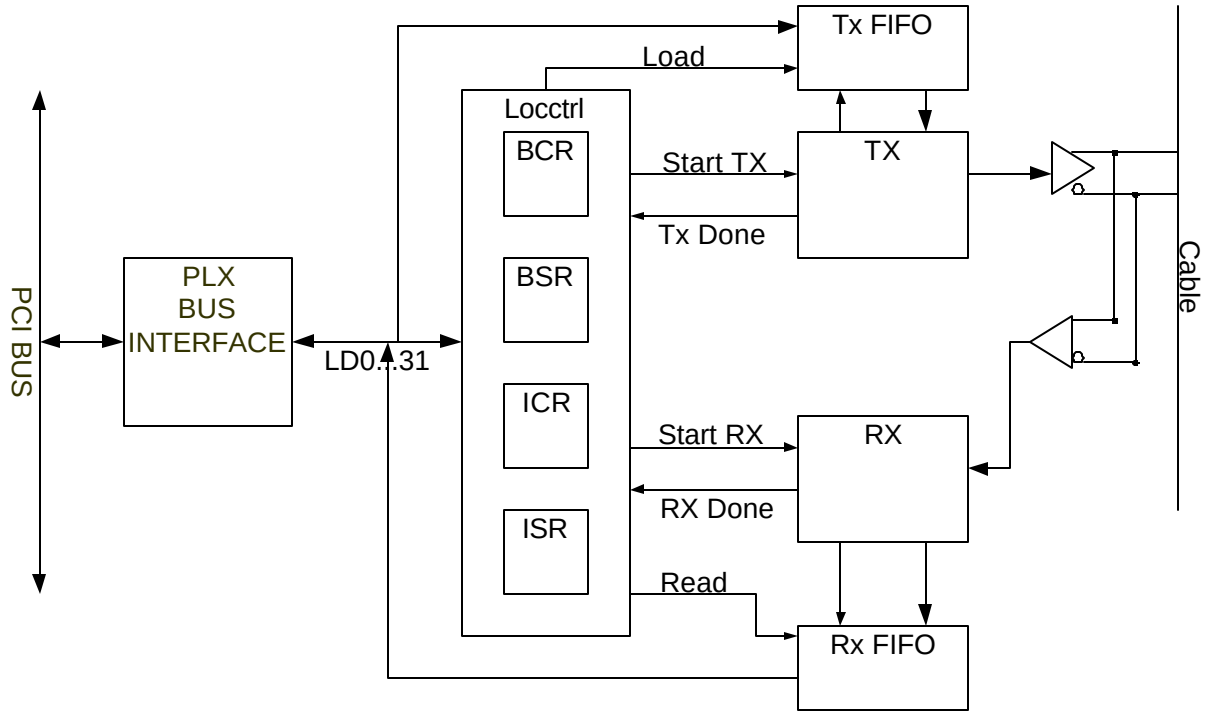
Table of Tables

1 INTRODUCTION

1.1 FUNCTIONAL DESCRIPTION

The PCI-HPDI32A-ASYNC Board includes a DMA Controller, 64/128/256/512 Kbytes of FIFO buffering, a cable Transmit controller, a cable Receive controller, and cable transceivers (RS-422/485 or differential pseudo ECL). The DMA on this board is intended for reading and writing the FIFOs. After the DMA is initialized and started, the host CPU will be free to proceed with other duties and need to respond only to interrupts. The DMA controller is capable of transferring data to host memory using D32 transfers; whereas the FIFO memory provides a means for continuous transmission of data without interrupting the DMA or requiring intervention from the host CPU. The board also provides for DMA chaining, interrupt generation for various states of the board, including End-Of-Transfer, TX FIFO Almost Empty, RX FIFO Almost Full, and more. The Transmitter and Receiver are in FPGAs that provide a configurable interface that is highly flexible in data width and transfer protocol.

Figure 1.1-1 Block Diagram



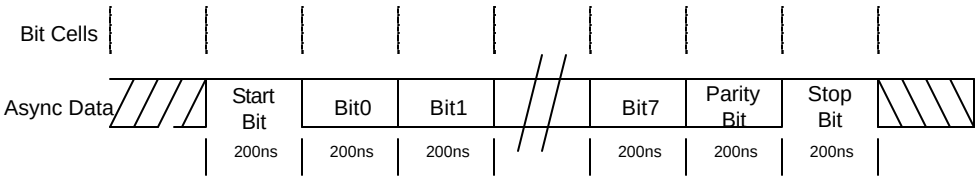
1.2 Theory of Operation

The transmitter and receiver of the standard HPDI32A board have been replaced by a ASYNC serial Data Transmitter and ASYNC serial data receiver with the following characteristics:

- Data rate of 5.0 megabits per second
- 8 Bits transmitter. LSB First.
- Software Selectable Even / Odd Parity.
- Software Selectable No Parity Bit Option.
- Software Selectable 1 or 2 stop bits.
- Software Selectable Big / Little Endian.
- Software Selectable 8 bit Tx / Rx FIFO size.
- Software Programmable gap between bytes transmitted.
- Software Programmable Message Length counters.
- Separate Transmit Data and Receive Data Lines.
- Software Selectable ability to Swap Transmit Data and Receive Data.
- Software Selectable ability to connect Transmit Data to Receive Data, for self-test loop back without requiring an external loop back cable.
- Typical 128K byte Transmit FIFO.
- Typical 128K byte Receive FIFO.
- 32 General Purpose I/O Lines

A Basic Serial Transfer is shown in Figure 1.2-1 below.

Figure 1.2-1 Basic Serial Transfer



The PCI-HPDI32A-ASYNC board also contains 32 programmable I/O Lines that are under software control. The lines are driven onto the cable on the Cable Data D0 thru Cable Data D31 Lines. Tri-state control on a byte wide basis is available thru the Board Control Register.

The PCI-HPDI32A-ASYNC contains a ASYNC data Transmitter, ASYNC data Receiver, and 32 programmable I/O lines. When reset, the PCI-HPDI32A-ASYNC will tri-state, 'hi-z', the 32 programmable I/O Lines and the Serial Data line out of the board. The board will tri-state 'on' and drive the 2 lines that indicate if the Transmitter and Receiver are running. They control signals are as follows:

- Transmit Data Line – Reset to tri-State 'hi-z'.
- Receive Data Line – Reset to tri-State 'hi-z'.
- Transmitter Running – Reset to tri-state 'on' and driven to logic low.
- Receiver Running – Reset to tri-state 'on' and driven to logic low.
- 32 programmable I/O Lines - Reset to tri-State 'hi-z'.

1.2.1 Transmitter Operation:

The transmitter is very flexible with mode bits and software controlled counters to control the following Software Selectable options:

- Data rate of 5.0 megabits per second.
- 8 Bit Transmitter. LSB First.
- Software Selectable Big / Little Endian.
- Software Selectable Number of D8 bytes per Message.
- Software Selectable 1 or 2 Stop Bits.
- Software Selectable Even or Odd Parity.
- Software Selectable No Parity Bit Option.
- Software Selectable Swap Tx to Rx Data Line.
- Software Selectable Clock Divider, for slower Data Rates.
- Software Selectable Cable Polarity, Positive or Negative.
- Software Selectable Programmable Gap between Bytes.
- Software Selectable No Gap between Bytes.
- Software Selectable 8 bit FIFO.

The Transmitter will also generate the following Status that may be read thru the Board Status Register and may generate Interrupts thru the Interrupt Control and Interrupt Status Registers.

- Tx Transmit Busy – The Transmitter is sending the Sync, or data bits.
- Tx Transmit Done – The Transmitter is sending a gap, or has no data to send.

The Transmitter operates from an on board clock oscillator, for 5 Mega bits per second operation the Transmitter will use a 40 Mhz clock. Each bit transmitted will be 8 clock cycles long and with a 40 Mhz clock each bit transmitted will be 200 nsec long. High end speeds will always be set by adjusting the clock frequency for the desired bit rate.

When a transmit is started by the Host processor the Transmitter will begin waiting for data to become available from the Transmit FIFO. Once Data is available from the FIFO, the transmitter will read the data and begin to send the Start Bit. After the Start Bit, the Transmitter will shift out the 8 Data Bits. If the Parity bit is to be transmitted, then the Parity bit will be shifted out after all 8 data bits have been shifted. The Transmitter will then always include 1 stop bit following the other data. There is a software selectable option to insert 2 stop bits after the final data bit. There is also a counter that will insert additional clock cycles after the stop bit(s) are complete. This counter will insert a gap that is ‘X’ clock cycles wide, using the Main Transmitter Clock. Typically, the Clock would be 40 Mhz, 25 nSec, So a count of 4 would yield an additional gap if 100 nSec, ½ bit cell time. Which would be 1 ½ or 2 ½ stop bits. If data is available from the FIFO, then the Transmitter will begin sending the Start bit of the next byte to Transmit. The Transmitter is capable of sending bytes back to back with no gap from the Stop bit of one byte to the Start bit of the next byte.

1.2.1.1 TX Big Endian Mode – BCR D16

Normally, the PCI-HPDI32A-ASYNC will transmit the D32 FIFO word as 4 D8 cable bytes, with D7..D0 being transmitted first, D15..D8 being transmitted second, D23..D16 being transmitted third, and D31..D24 being transmitted fourth. When this bit is set then the data will be taken out of the FIFO in Big Endian (Motorola) style. Using Big Endian, D31..D24 will be transmitted first, D23..D16will be transmitted second, D15..D8 will be transmitted third, and D7..D0 will be transmitted fourth.

1.2.1.2 Tx Even Parity - BCR D17

This bit will control whether the Transmitter will generate Even or Odd Parity on the Transmitted byte. When this bit is a ‘1’ the Transmitter will generate Even Parity. When this bit is a ‘0’ the Transmitter will generate Odd Parity.

1.2.1.3 Tx 2 Stop Bits - BCR D18

When this bit is a '1', the Transmitter will send 2 stop at the end of the Byte being transmitted. When this bit is a '0', the Transmitter will send only 1 Stop Bit. Default to '0'.

1.2.1.4 Tx Swap Tx Data to CC2 -BCR D19

Normally, the PCI-HPDI32A-ASYNC will transmit data on Cable Command 1 lines. When this bit is set to a '1' the Transmitter will send its data on Cable Command 2 lines. Default to '0'.

1.2.1.5 Tx No Parity -BCR D20

Normally, the PCI-HPDI32A-ASYNC will transmit data and a parity bit of selectable polarity. When this bit is set to a '1' the Transmitter will not send a parity bit. When this bit is a '0' the Transmitter will send a parity bit as part of each byte send. Default to '0'.

1.2.1.6 Tx 8 Bit FIFO -BCR D21

Normally, the PCI-HPDI32A-ASYNC will transmit all 32 bits of the FIFO data word. When this bit is set to a '1' the transmitter will only send 1 byte from each 32 bit FIFO word, the other 24 bits will be discarded. (For little Endian operation, D7..D0 is the Byte that is send, For Big Endian operation D31..D24 is the byte that is send). When this bit is set to a '0', the Transmitter will send the entire 32 bit FIFO word. Default to '0'.

1.2.1.7 Tx Invert Cable Polarity -BCR D22

Normally, the PCI-HPDI32A-ASYNC will transmit data in positive polarity, a '1' will generate a 'HI' on the '+' side of the differential pair and a 'LO' on the '-' side of the differential pair. This bit will cause the Transmitter to invert the serial data before it is sent to the cable drivers so that a '1' will generate a 'LO' on the '+' side of the differential pair and a 'HI' on the '-' side of the differential pair. Default to '0'.

1.2.1.8 Tx Data Port Output Enable D7-D0. BCR D12

This bit controls the output enable for the Data Port Bits D7-D0. A '1' will enable the board to Drive Cable Data D7-D0. A '0' will prevent the board from driving Cable Data D7-D0. Default to '0'.

1.2.1.9 Tx Data Port Output Enable D15-D8. BCR D13

This bit controls the output enable for the Data Port Bits D15-D8. A '1' will enable the board to Drive Cable Data D15-D8. A '0' will prevent the board from driving Cable Data D15-D8. Default to '0'.

1.2.1.10 Tx Data Port Output Enable D23-D16. BCR D14

This bit controls the output enable for the Data Port Bits D23-D16. A '1' will enable the board to Drive Cable Data D23-D16. A '0' will prevent the board from driving Cable Data D23-D16. Default to '0'.

1.2.1.11 Tx Data Port Output Enable D31-D24. BCR D15

This bit controls the output enable for the Data Port Bits D31-D24. A '1' will enable the board to Drive Cable Data D31-D24. A '0' will prevent the board from driving Cable Data D31-D24. Default to '0'.

1.2.1.12 Tx Row Valid Length Counter

The Transmit Row Valid Length Counter is the Number of D8 bytes that will be transmitted in a single message. If this counter not used, then the Transmitter will transmit data until the FIFO becomes empty. This Counter MUST be used for any message that is not an even multiple of D32 FIFO word size bytes long. If there is more data in the FIFO, the transmitter will begin transmit of the new FIFO data.

For example, with a Row Length of 5, Little Endian, D32 FIFO, and the following data in the FIFO:

First Load into FIFO	0x44332211
Second Load into FIFO	0x88776655
Third Load into FIFO	0xccbbaa99
Fourth Load into FIFO	0x00ffeedd

Will result in the transmission of data in the following order.

0x11
0x22
0x33
0x44
0x55
0x99
0xaa
0xbb
0xcc
0xdd

The Upper 3 bytes of the Second and Fourth words will be read from the FIFO and Discarded.

If the FIFO becomes empty, then the transmitter will end the message before the counter has decremented to zero.

1.2.1.13 Tx Row Invalid Counter.

The lower 16 bits, D15..D0, of the Row Invalid Counter / Register will be used to extend the gap between the stop bit(s) of one byte and the Start Bit of the next byte in a transmission. It can be set to zero for no additional gap between the stop bit(s) and the Start bit of the next byte.

The upper 16 bits, D31..D16, of the Row Invalid Counter / Register is used as the Transmit Clock Divider. When these bits are all zero the transmitter will operate at full speed, 1/8 of the main transmit clock frequency. The transmit bit time is calculated as follows,

(1 + count) * 8 * (clock period)

Typically, the Transmit clock will be 40 Mhz, 25 nsec period. The Bit times and Bit Rates for several division count values are given in the following table:

Figure 1.2-2 Count Vs Bits Per Second, Transmit

Count	Bit Time	Bits Per Second
0	200	5 Mbps
1	400	2.5Mbps
2	600	1.666Mbps
4	1000	1 Mbps
8	1800	555.555 Kbps
16	3825	261.438 Kbps
32	6600	151.515
128	25800	38.760 Kbps

1.2.1.14 Tx Status Block Length Counter

The Transmit Status Block Length Counter has been replaced by a counter that will count the actual number of bytes transmitted. If the Transmit FIFO becomes empty during a Transmit operation, the

Transmitter will STOP. This counter can be used to detect if the transmit DMA operation was interrupted by other PCI bus traffic.

1.2.1.15 Tx Clock Division Counter

This Counter has been removed from the PCI-HPDI32A-ASYNC. The Device Driver will not allow access to this register as of this time. The clock division Register / counter is located in the upper 16 bits of the Tx Row Invalid Counter.

1.2.1.16 Tx Output Data Port Register

The Firmware Features Register has been removed from the PCI-HPDI32A-ASYNC. It has been replaced by a 32 bit register that can is under software control. When Enabled by the proper output enable bits in the Board Control Register, the contents of this register will be driven out onto the cable on the Cable Data Bits on a 1 to 1 basis, D0 to D0 thru D31 to D31. This register can be read or written. When you read back this register, you get the contents of the holding register, NOT the contents of the Data Cable.

In order to access this register using the standard device driver you must include a define similar to what is shown below and access the register using this define in the routines as shown.

```
#define SPL_TX_OUTPUT_REG          5

// Write the Test Data to the Output Port Register
WriteLocal(TxBoard, SPL_TX_OUTPUT_REG, TestData)  ;

// Read Back the port register and verify it's contents
ReadData = ReadLocal(TxBoard, SPL_TX_OUTPUT_REG ) ;
```

1.2.2 Receiver Operation:

The Receiver is very flexible with mode bits to control the following Software Selectable options:

- Data rate of 5.0 megabits per second.
- 8 Bits Receiver. LSB First.
- Software Selectable Big / Little Endian.
- Software Selectable Even or Odd Parity.
- Software Selectable No Parity Option.
- Software Selectable Cable Polarity.
- Software Selectable 8 bit FIFO.
- Software Selectable Swap Rx to Tx Data line.
- Number of D8 Bytes Received Counter.
- Software Programmable clock divider, for slower data rates.
- Register for the Cable D31-D0 Lines.

When the Receiver is started it will begin searching for an Idle Cable condition, defined as no activity on the cable for 88 divided clock cycles. Only after locating a period when the bus is completely idle can the Receiver be sure that the next Transition detected on the cable is the Start Bit of a new Byte. After locating the Start Bit, The Receiver will begin shifting in the 8 Data Bits and the optional Parity Bit. During the Stop Bit, the Receiver is checking Parity and loading the received byte into the FIFO. The receiver does not check the Stop Bit, but it does require that a stop bit be present.

The Receiver will flag Receive Parity Errors, but will load the byte containing the error into the FIFO. The Receiver will also check for data errors during the receive operation. A Data error is defined as finding a transition during a bit cell time. Transitions are only allowed at the boundaries between bit cells. If a Data error occurs during a byte of data, that byte is lost and the Receiver will stop collecting data until the cable is detected as being Idle again.

The Receiver will generate the following status in the Board Status Register that can be read by the Host Processor and can be used to generate Interrupts thru the Interrupt control register.

- Rx Running – The Receiver is enabled and is ready.
- Rx Busy – The Receiver is in the process of receiving a message.
- Rx Error – The Receiver is in Error recovery, searching for an Idle Cable.

The Receiver will also generate the following status that can be used to generate interrupts.

- Rx Data Error – The Receiver has detected a Data Error.
- Rx Parity Error – The Receiver has detected a Parity Error.

1.2.2.1 Rx Fifo Big Endian Mode – BCR D24

Normally, the PCI-HPDI32A-ASYNC will receive cable bytes and assemble them into a D32 FIFO word, with D7..D0 being received first, D15..D8 being received second, D23..D16 being received third and D31..D24 being received fourth. When this bit is set then data will be written into the FIFO in Big Endian (Motorola) style. Using Big Endian, D31..D24 being received first, D23..D16 being received second, D15..D8 being received third and D7..D0 being received fourth. Default is ‘0’.

1.2.2.2 Rx Even Parity Select - BCR D25

When this bit is set to a ‘1’, the Receiver will expect the received byte of data to carry Even Parity. When this bit is set to a ‘0’ the Receiver will expect the received byte of data to carry Odd Parity. Default is ‘0’.

1.2.2.3 Rx Alternate Register Select - BCR D26

When this bit is set to a ‘1’, the host can access the Receiver Clock Division Register thru the address assigned to the Rx Status Length Counter. When this bit is set to a ‘0’, reading the RX Status Length Register will allow the Host processor to read the current state of the 32 Cable Data lines. Default is ‘0’.

1.2.2.4 Rx Swap Receiver to Cable Command 1 - BCR D27

When this bit is set to a '1', the Receiver will receive data on Cable Command 1. When this bit is set to a '0' , the Receiver will receive on Cable Command 2. Setting either BCR-D27 or BCR-D19 will allow the board to send self test loop back messages without requiring an external loop back cable. Setting both BCR-D27 and BCR-D19 will allow 2 boards to communicate back to back using a 1 to 1 cable without requiring a crossover cable. Default is '0'.

1.2.2.5 Rx 8 bit FIFO - BCR D28

When this bit is set to a '1', the Receiver will only load a single byte of data into each D32 FIFO word. When this bit is set to a '0', the Receiver will assemble 4 bytes into a D32 word and then write the D32 word into the Receive FIFO. Default is '0'. In Little Endian Mode, the received byte will be in D7..D0, D15..D8 will contain the lower 8 bits of the 16 bit time tag, D23..D16 will contain the upper 8 bits of the 16 bit time tag, and D31..D24 will contain the lower 8 bits of the Rx Row Length Counter. In Big Endian Mode, the received byte will be in D31..D24, D23..D16 will contain the lower 8 bits of the 16 bit time tag, D15..D8 will contain the upper 8 bits of the 16 bit time tag, and D7..D0 will contain the lower 8 bits of the Rx Row Length Counter

1.2.2.6 Rx Invert Cable Polarity - BCR D29

When this bit is set to a '1', the Receiver will invert the incoming data from the cable. When this bit is set to a '0' the Receiver will not invert the receive data. Default is '0'.

1.2.2.7 Rx No Parity Bit - BCR D30

This command bit will tell the Receiver that there is no parity bit in the incoming data. When this bit is set to a '1', the Receiver will expect to receive only 1 start bit, 8 data bits and 1 stop bit. When this bit is set to a '0' the Receiver will expect to receive 1 start bit, 8 data bits, 1 parity bit, and 1 stop bit. Default is '0'. NOTE: Do not use this command bit to try to ignore parity errors. Incorrect receiver operation will result.

1.2.2.8 Rx Row Length Counter

This counter will count the number of D8 bytes received in the current message. It will be reset to zero on a board reset, or every time the Receiver Starts up. When in the middle of a message, the Row Length Counter will have the Current count of the number of D8 bytes received in this message.

1.2.2.9 Rx Status Block Length Counter

The Status Block Length counter has been removed from the ASYNC board. When the Alternate Register Select bit, BCR-D26, is a '0' then this has been replaced by a 32 bit register that will allow the Host processor to read the current state of the 32 Cable Data lines. The RS-485 Receivers used on the HPDI32A-ASYNC project are guaranteed that a Floating Cable will be received as a logic '1'.

When the Alternate Register Select bit, BCR D26, is a '1' then reading and writing this register will read and write the Receiver Clock Division Register. The Receive Clock Division Register uses the lower 16 bits, D15..D0. The upper 16 bits, D31..D16 are not used. When these bits are all zero the receiver will operate at full speed, 1/8 of the main transmit clock frequency. The receive bit time is calculated the same as the transmit bit time, as follows,

(1 + count) * 8 * (clock period)

The Receiver runs off of the same clock as the Transmitter. Typically, the Transmit clock will be 40 Mhz, 25 nsec. The Bit times and Bit Rates for several division count values are given in the following table:

Figure 1.2-3 Count Vs Bits Per Second, Receive

Count	Bit Time	Bits Per Second
0	200	5 Mbps
1	400	2.5Mbps
2	600	1.666Mbps
4	1000	1 Mbps
8	1800	555.555 Kbps
16	3825	261.438 Kbps
32	6600	151.515
128	25800	38.760 Kbps

1.2.2.10 Rx Disable Output Status Lines - BCR D6

In normal operation, the board will drive Cable Command D5 and Cable Command D6 all of the time. Cable Command D5 will be driven to a logic 'high' when the transmitter on this board is running. Cable Command D6 will be driven to a logic 'high' when the receiver on this board is running. When this bit is set to a '1', this board will be prevented from driving Cable Command D5 and Cable Command D6. When this bit is set to a '0', then this board will drive Cable Command D5 and D6. This bit is included for testing 2 boards cabled back to back. Default is '0'.

1.3 CABLE INTERFACE

The cable interface consists of 32 bi-directional Cable Data lines, the ASYNC Transmit Data Line, the ASYNC Receive Data line, and 2 control / Status output lines, Transmitter Running and Receiver Running, which can employ either differential I/O RS485/422 compatibility or differential pseudo ECL. Refer to cable pin-out in Figure 4.3-1.

The Cable Command Lines are assigned as follows.

- ASYNC Transmit Data Line – Connected to Cable Command 1.
- ASYNC Receive Data Line - Connected to Cable Command 2.
- Transmitter Running - Connected to Cable Command 5.
- Receiver Running – Connected to Cable Command 6.
- 32 I/O Lines - Connected to Cable Data 0 thru Cable Data 31.

1.4 FIFOs

The FIFOs on the PCI-HPDI32A-ASYNC are used for buffering the transmit and receive data. There are a total of eight FIFOs on the board; 1 set of 4 FIFOs for transmit, and 1 set of 4 FIFOs for receive. Each set consists of 32 bits of data and 4 status flags. The typical configuration is the –256K board ordering option, which is 128Kbytes of Transmit FIFO and 128Kbytes of Receive FIFO. Both FIFOs are organized as 32K deep by 32 bits wide. The receive FIFOs are loaded by the cable receive control logic and read by either the CPU or the DMA. The transmit FIFOs are loaded by either the CPU or the DMA and read by the cable transmit control logic. The 4 status flags that accompany the FIFOs are all active low ('0' being TRUE) and are as follows: Empty, Almost Empty, Almost Full, Full. The Almost Empty and the Almost Full status flags can be programmed by the software to become true at most desired levels.

In addition there are 3 FIFO Event Flags that are part of the FIFO system.
Rx FIFO Underflow – Read of the Rx FIFO while the FIFO is Empty.
Rx FIFO Overflow – Write to the Rx FIFO while the Rx FIFO is Full.
Tx FIFO Overflow – Write to the Tx FIFO while the Tx FIFO is Full.

The Rx FIFO events are cleared by a Rx FIFO Reset. The Tx FIFO Event flag is cleared by a Tx FIFO reset.

2 REGISTERS

2.1 PCI-HPDI32A- ASYNC REGISTER INFORMATION

The PCI-HPDI32A-ASYNC Card complies with the **plug-n-play** concept. That is, at the time of power-up, an attempt will be made by the CPU to set up the board to meet the configuration requirements of the system. In doing this, the CPU will map the amount of I/O space requested by the PCI-HPDI32A-ASYNC Card and return the configuration base address into the PCI Configuration Register (Offset 0x18) of this Board. All PCI bus interface functions are handled by the PLX9080-3 PCI bus interface. For more information regarding the PCI bus interface, please refer to **Appendix A, PLX Register Programming.**

2.2 PCI-HPDI32A-ASYNC LOCAL REGISTERS, BIT DESCRIPTIONS

The PCI-HPDI32A-ASYNC board contains the following registers.

Figure 2.2-1 Register Map

Offset Address	Size	Access*	Register Name	Value after Reset
0x00	D32	RO	Firmware Revision	0x000F0200
0x04	D32	RW	Board Control	0x00000000
0x08	D32	RO	Board Status	0x0000CCXX
0x0C	D32	RW	Tx Programmable Almost	0x01000008
0x10	D32	RW	Rx Programmable Almost	0x01000008
0x14	N/A	N/A	Firmware Features	Removed
0x14	D32	RW	Tx Output Data Port Register	0x00000000
0x18	D32	RW	Tx/Rx FIFOs	EMPTY
0x1C	D32	R0	Tx Status Block Length Counter	0x00000000
0x20	D32	RW	Tx Row Valid Length Counter	0x00000000
0x24	D16	RW	Tx Line Invalid Length	0x00000008
0x28	D32	RO	Rx Status Block Length	Undefined
0x2C	D32	RO	Rx Row Length Counter	0x00000000
0x30	D32	RW	Interrupt Control	0x00000000
0x34	D32	RC	Interrupt Status	0x0000031X
0x38	D32	N/A	Tx Clock Divider - Removed	Removed
0x2C	N/A	N/A	Reserved	N/A
0x40	D32	RO	Tx FIFO Size	0x00000000
0x44	D32	RO	Rx FIFO Size - Removed	Removed
0x48	D32	RO	Tx FIFO Word Count	0x00000000
0x4c	D32	RO	Rx FIFO Word Count - Removed	Removed
0x50	D32	R/W	Interrupt Edge Select	0x00000000
0x54	D32	R/W	Interrupt Hi/Lo Level Select	0xFFFFFFFF

* RO - read only
RW - read/write capability
RC - read clear (a write clears the specified bits)

2.2.1 FIRMWARE REVISION: (Offset 0x00 - RO)

This Register is used to determine the version of firmware that is programmed into the board. If the logic is changed to accommodate a modification for any reason then the value in this register is incremented.

Revision 0x000F0200: Original version after debug and final release.

D7..D0	Firmware Revision
D15..D8	Board Revision Currently HPDI32A Revision A
D23..D16	Board Identifier 0x000F Identifies the HPDI32A-ASYNC
D30..D24	Reserved
D31	Firmware Features Identifier. Indicates that this board does not contains the Firmware Features Register.

2.2.2 BOARD CONTROL: (Offset 0x04 - RW)

The Board Control Register is strictly under software control and provides the following functionality's:

- Resets the board;
- Resets the transmit FIFOs;
- Resets the receive FIFOs;
- Enables the receive logic;
- Enables the transmit logic;
- Starts the transmission of data;
- Mode control for the transmitter.
- Mode control for the receiver.

D0	Board Reset Writing a *1* to this bit will generate a self-timed pulse that is used to reset the on-board logic and the FIFOs. There is no need for the software to clear this bit, the bit will clear itself.
D1	Tx FIFO Reset Writing a *1* to this bit will generate a self-timed pulse that will be used to reset the Tx FIFOs. After setting this bit, there should be a minimum of 25 Rx clk periods, or 1 millisecond if the receivers are turned off, before any local accesses are performed. There is no need for the software to clear this bit, the bit will clear itself.
D2	Rx FIFO Reset Writing a *1* to this bit will generate a self-timed pulse that will be used to reset the Rx FIFOs. After setting this bit, there should be a minimum of 25 Rx clk periods, or 1 millisecond if the receivers are turned off, before any local accesses are performed. There is no need for the software to clear this bit, the bit will clear itself.
D3	Reserved
D4	Enable Tx to Drive the cable. Writing a *1* to this bit will enable the transmit logic to drive the cable. Writing a 0 to this bit will disable the transmit logic from driving the cable.
D5	Start Rx Writing a *1* to this bit will start the receiver. Writing a 0 to this bit will stop the receiver.
D6	Rx Disable Output Status Lines When this bit is set to a '0', then this board will drive Cable Command D5 and D6 When this bit is set to a '1', this board will be prevented from driving Cable Command D5 and D6 Default is '0'.
D7	Reserved.

D8	Start Tx Writing a *1* to this bit will start the transmitter. Writing a 0 to this bit will stop the transmitter.
D9	Reserved
D10	Dual DMA Enable. – PCI ONLY. On boards with 2 DMA channel capability, this bit will enable use of both DMA channels. Not supported at this time. – This bit must ALWAYS be set to '0' for proper board operation.
D11	Reserved
D12	Tx Data Port Output Enable D7-D0 A '1' will enable the board to Drive Cable Data D7-D0. A '0' will prevent the board from driving Cable Data D7-D0. Default to '0'.
D13	Tx Data Port Output Enable D15-D8 A '1' will enable the board to Drive Cable Data D15-D8. A '0' will prevent the board from driving Cable Data D15-D8. Default to '0'.
D14	Tx Data Port Output Enable D23-D16 A '1' will enable the board to Drive Cable Data D23-D16. A '0' will prevent the board from driving Cable Data D23-D16. Default to '0'.
D15	Tx Data Port Output Enable D31-D24 A '1' will enable the board to Drive Cable Data D31-D24. A '0' will prevent the board from driving Cable Data D31-D24. Default to '0'.
D16	Tx Big Endian Mode. Writing a 1 to this bit will enable transmitting in Big Endian Mode Writing a 0 to this bit will enable transmitting in Little Endian Mode. Default is '0', Little Endian Mode.
D17	Tx Even Parity. Writing a '1' to this bit will enable the transmitter to generate and send Even parity with each byte. Writing a '0' to this bit will enable the transmitter to generate and send Odd parity with each byte. Default is '0', Odd Parity.
D18	Tx 2 Stop Bits. Writing a '1' to this bit Place 2 stop bits on the end of each byte sent. Writing a '0' to this bit Place 1 stop bits on the end of each byte sent. Default is '0', 1 Stop bit.
D19	Tx Transmit on Cable Command 2 Writing a '1' cause this board to transmit on Cable Command 2. Writing a '0' cause this board to transmit on Cable Command 1. Default is '0', transmit on Cable Command 1.
D20	Tx No Parity Bit. Writing a '1' cause this board to not transmit a parity bit. Writing a '0' cause this board to transmit a parity bit. Default is '0', transmit a parity bit. .
D21	Tx 8 bit FIFO Writing a '1' cause this board to transmit 1 byte from each D32 FIFO Word. The other 3 bytes are discarded. . Writing a '0' causes this board to transmit all 4 bytes from each D32 FIFO word. Default is '0', transmit all 4 bytes from each D32 FIFO word.
D22	Tx Invert Cable Polarity. Writing a '1' cause this board to Invert its serial data before it is sent to the Cable. Writing a '0' cause this board to transmit its serial data without inverting the data . Default is '0', do not invert the transmit data. .
D23	Reserved
D24	Rx FIFO Big Endian Mode Writing a '1' to this bit will enable receiving data in Big Endian Mode Writing a '0' to this bit will enable receiving data in Little Endian Mode. Default is '0', Little Endian Mode.

D25	Rx Even Parity Select. Writing a '1' to this bit will command the receiver to check the received byte for even parity. Writing a '0' to this bit will command the receiver to check the received byte for odd parity. Default is '0', odd parity
D26	Rx Alternate Register Select. Writing a '1' to this bit will enable reading and writing the Receive Clock Divider Register and will disable reading and writing the Receive Input Port Register. Writing a '0' to this bit will disable reading and writing the Receive Clock Divider Register and will enable reading the Receive Input Port Register. Default is '0', Read the Receive Input Port Register.
D27	Rx Swap Receiver to Cable Command 1. Writing a '1' causes this board to receive from Cable Command 1. Writing a '0' causes this board to receive from Cable Command 2. Default is '0', receive from Cable Command 2.
D28	Rx 8 bit FIFO Writing a '1' causes this board to receive 1 byte for each D32 FIFO Word. The other 3 bytes are the word count and a Time Tag. Writing a '0' causes this board to receive 4 bytes to form each D32 FIFO word. Default is '0', receive 4 bytes to form each D32 FIFO word.
D29	Rx Invert Cable Polarity. Writing a '1' causes this board to invert the data from the cable before it is sent to the receiver. Writing a '0' causes this board to receive data with no inversion. Default is '0', do not Invert the data from the cable before it is sent to the receiver
D30	Rx No Parity bit. Writing a '1' causes this board to receive data with no parity bit. Writing a '0' causes this board to receive data with the parity bit. Default is '0', Receive data with a parity bit.
D31	Reserved

2.2.3 BOARD STATUS: (Offset 0x08 - RO)

The Board Status Register is used to return information to the software about the most current status of the board, at the time of the reading. Listed below is the information that this register contains:

D0	Rx Cable Command D0
D1	Rx Cable Command D1.
D2	Rx Cable Command D2
D3	Rx Cable Command D3.
D4	Rx Cable Command D4.
D5	Rx Cable Command D5.
D6	Rx Cable Command D6.
D7	Tx Transmit Done. A 1 will indicate that a transmit is complete and the transmitter is waiting for the host processor to remove 'Start Tx', or the transmitter is waiting for more data in the Transmit FIFO A 0 will indicate that the transmitter is Stopped, or a transmit is in progress.

D8	Tx FIFO Empty L *1* will indicate the Tx FIFO is not empty. *0* will indicate the Tx FIFO is empty
D9	Tx FIFO Almost Empty L *1* will indicate the Tx FIFO is not almost empty. *0* will indicate the Tx FIFO is empty
D10	Tx FIFO Almost Full L *1* will indicate the Tx FIFO is not almost full. *0* will indicate the Tx FIFO is almost full
D11	Tx FIFO Full L *1* will indicate the Tx FIFO is not full. *0* will indicate the Tx FIFO is full.
D12	Rx FIFO Empty L *1* will indicate the Rx FIFO is not empty. *0* will indicate the Rx FIFO is empty.
D13	Rx FIFO Almost Empty L *1* will indicate the Rx FIFO is not almost empty. *0* will indicate the Rx FIFO is empty.
D14	Rx FIFO Almost Full L *1* will indicate the Rx FIFO is not almost full. *0* will indicate the Rx FIFO is almost full.
D15	Rx FIFO Full L *1* will indicate the Rx FIFO is not full. *0* will indicate the Rx FIFO is full.
D16	Board Jumper 0 1 will indicate that Board Jumper 0 is present 0 will indicate that board jumper 0 is absent
D17	Board Jumper 1 1 will indicate that Board Jumper 1 is present 0 will indicate that board jumper 1 is absent
D18	Reserved
D19	Rx FIFO Underflow. *1* will indicate that a Receive FIFO underflow has occurred *0* will indicate that no Receive FIFO underflow has occurred
D20	Rx FIFO Overflow. *1* will indicate that a Receive FIFO overflow has occurred *0* will indicate that no Receive FIFO overflow has occurred
D21	Tx FIFO Overflow. *1* will indicate that a Transmit FIFO overflow has occurred *0* will indicate that no Transmit FIFO overflow has occurred
D22	Rx Receive Parity Error. *1* will indicate that the Receiver has detected a Parity Error in the Receive Data. *0* will indicate that the Receiver has not detected a Parity Error. Reset to '0' when the Receiver is started.
D23	Rx Data Error. *1* will indicate that the Receiver has detected a Data Error in the receive data. *0* will indicate that the Receiver has not detected a Data Error in the receive data. A Data Error is a Transition during a receive Bit Cell Time. Transitions are only allowed at the boundaries between bit cells. Reset to '0' when the Receiver is started.
D24	Rx Busy 1 will indicate that the Receiver busy receiving a byte of data.. 0 will indicate that the Receiver is Idle, in Error Recovery, or not running.
D25	Tx Transmit Busy 1 will indicate the Transmitter is busy Transmitting data. 0 will indicate the Transmitter is either IDLE, sending a Gap or Waiting for FIFO data.
D31..D26	Reserved

2.2.4 TX ALMOST: (Offset 0x0C - RW)

This register is contains the values that are used to program the Almost Flags of the transmit FIFOs.

Default is 0x01000008.
Almost Empty = 0x08
Almost Full = 0x0100 from full.

D7..D0	Low byte, Almost Empty
D15..D8	High byte, Almost Empty
D23..D16	Low byte, Almost Full
D31..D24	High byte, Almost Full

2.2.5 RX ALMOST: (Offset 0x10 - RW)

This register is contains the values that are used to program the Almost Flags of the receive FIFOs.

Default is 0x01000008.
Almost Empty = 0x08
Almost Full = 0x0100 from full.

D7..D0	Low byte, Almost Empty
D15..D8	High byte, Almost Empty
D23..D16	Low byte, Almost Full
D31..D24	High byte, Almost Full

2.2.6 FIRMWARE FEATURES: (Offset 0x14 - RW)

Register to note new features present in FW. This Register has been removed from the ASYNC. It has been replaced by the Tx Output Data Port Register.

2.2.7 TX OUTPUT DATA PORT REGISTER: (Offset 0x14 - RW)

This is the holding register for data to be output on the 32 programmable I/O Lines. The outputs are enabled to drive the cable thru the Board Control Register Bits 15 thru 12. Reading this address will give the contents of the holding register, NOT the contents of the Cable Data Lines. To read the current state of the Cable Data Lines read the Rx Status Block Length register.

D31..D0	Tx Output Data Port Register.

2.2.8 FIFO I/O: (Offset 0x18 - RW)

The 64/128/256/512 Kbytes FIFOs are used for buffering data. This gives the software a means of buffering the data before it is transmitted to the cable or retrieved from the FIFOs. The FIFOs are also used by the DMA for the same purpose. This eliminates unnecessary PCI bus arbitration which provides for faster and more efficient bus cycles for transfers; hence, a faster and more efficient system. Typical configuration is 256Kbytes which is 128Kbytes Transmit FIFO (32K by 32 bits) and 128Kbytes Receive FIFO (32 K by 32 bits) .

D31..D0	FIFO Data 0..31 This is the buffer that contains both the transmit and receive data. A write to this offset will load into the Transmit FIFO. A read from this offset will read from the Receive FIFO.

2.2.9 TX STATUS LENGTH REGISTER: (Offset 0x1C – RO)

The Transmit Status Length Register has been replaced by a counter that will count the number of bytes that were transmitted in the current message. Reset to Zero when the transmitter starts. Useful for checking if the DMA was interrupted by bus loading.

D31..D0	Data 0..31 The number of Bytes transmitted in the current message. Reset to Zero when the Transmitter Starts. .

2.2.10 TX ROW VALID LENGTH REGISTER: (Offset 0x20 - RW)

This register / counter controls the number of D8 Bytes transmitted. For each byte that is transmitted, the counter will be decremented. When the Counter reaches Zero, the transmitter will halt the current message and any partial D32 Words will be discarded. If new data becomes available in the Transmit FIFO, the counter will be reset and the transmitter will begin sending the new message. This counter can be used to generate message sizes that are not multiples of D32 Words. In normal operation, this counter would not be used.

D31..D0	Data 0..31 The number of D8 Bytes to transmit per message.

2.2.11 TX ROW INVALID LENGTH REGISTER: (Offset 0x24 - RW)

This register / counter is used to control 2 functions. The lower 16 bits, B15..D0, is used to control the Gap between the Stop Bit of one byte and the Start bit of the next byte transmitted. In units of Main Clock Ticks, Typically 40 Mhz. Reset to 8, which will give an additional time of 200 nSec, a single bit time. Can be set to zero to give no gap between the Stop bit of one byte and the Start bit of the next byte. 16 Bit counter.

The Upper 16 bits, D31..D16, of the Row Invalid Register / Counter is used to control the Transmit Clock Divider. When these bits are all zero the transmitter will operate at full speed, 1/8 of the main transmit clock frequency. See Section 1.2.1.13 for a description of bit times VS Bits Per Second.

D15..D0	Data 0..15 Sets the size of the Gap between bytes in a message.
D31..D16	Transmit Clock Divider. Sets the Bit cell time of transmit data. See Section 1.2.1.13.

2.2.12 RX STATUS BLOCK LENGTH COUNTER: (Offset 0x28 - RO)

The Rx Status Block Length counter has been removed from the ASYNC board. When the Alternate Register Select bit, BCR-D26, is a '0' then this Counter has been replaced by a 32 bit register that will allow the Host processor to read the current state of the 32 Cable Data lines. The RS-485 Receivers used on the HPDI32A-ASYNC project are guaranteed that a Floating Cable will be received as a logic '1'.

When the Alternate Register Select bit, BCR D26, is a '1' then reading and writing this register will read and write the Receiver Clock Division Register. The Receive Clock Division Register uses the lower 16 bits, D15..D0. The upper 16 bits, D31..D16 are not used. When these bits are all zero the receiver will operate at full speed, 1/8 of the main transmit clock frequency. The receive bit time is calculated the same as the transmit bit time, as follows,

(1 + count) * 8 * (clock period)

See Section 1.2.2.9 for a description of Bit Times VS Bits Per Second

D15..D0	Cable Data D15 thru Cable Data D0 or Rx Clock Division Register.
D31..D16	Cable Data D31 thru Cable Data D16 or not used

2.2.13 RX ROW LENGTH COUNTER: (Offset 0x2C - RO)

This counter contains the number of D8 bytes received since the receiver was started. It is reset to 0 when the receiver starts running. The Host Processor can read it at any time.

D31..D0	Counter 0..31

	The number of D8 bytes received since the receiver was started.

2.2.14 INTERRUPT CONTROL: (Offset 0x30 - RW)

The Interrupt Control Register provides the software with a means of selecting what conditions will be allowed to generate an interrupt.

D0	Enable Interrupt on Cable Command 1 Data Rising Edge. *1* will allow an interrupt on the Cable ASYNC Data Lines rising edge. *0* will disallow an interrupt on the Cable ASYNC Data Lines rising edge.
D1	Enable Interrupt on Cable Command 1 Data Falling Edge. *1* will allow an interrupt on Cable ASYNC Data Falling Edge. *0* will disallow an interrupt on Cable ASYNC Data Falling Edge.
D2	Enable Interrupt on Cable Command 1 Data a logic '1'. *1* will allow an interrupt on Cable ASYNC Data a logic '1'. *0* will disallow an interrupt Cable ASYNC Data a logic '1'.
D3	Enable Interrupt on Cable Command 2 a logic '1'. *1* will allow an interrupt on Cable Command 2 a logic '1' *0* will disallow an interrupt Cable Command 2 a logic '1'
D4	Enable Interrupt on Cable Command 4 a logic '1'. *1* will allow an interrupt on Cable Command 4 a logic '1' *0* will disallow an interrupt Cable Command 4 a logic '1'
D5	Enable Interrupt on Cable Command 5 a logic '1'. – Cable Transmitter Driving the Cable *1* will allow an interrupt on Cable Command 5 a logic '1' *0* will disallow an interrupt Cable Command 5 a logic '1'
D6	Enable Interrupt on Cable Command 6 a logic '1'. – Cable Receiver Ready to Receive *1* will allow an interrupt on Cable Command 6 a logic '1' *0* will disallow an interrupt Cable Command 6 a logic '1'
D7	Enable Interrupt on Tx Transmit Done. *1* will allow an interrupt on Tx Transmit Done. *0* will disallow an interrupt on Tx Transmit Done.
D8	Enable Interrupt on Tx FIFO Empty *1* will allow an interrupt on Tx FIFO empty. *0* will disallow an interrupt on Tx FIFO empty.
D9	Enable Interrupt on Tx FIFO Almost Empty *1* will allow an interrupt on Tx FIFO almost empty. *0* will disallow an interrupt on Tx FIFO almost empty.
D10	Enable Interrupt on Tx FIFO Almost Full *1* will allow an interrupt on Tx FIFO almost full. *0* will disallow an interrupt on Tx FIFO almost full.
D11	Enable Interrupt on Tx FIFO Full *1* will allow an interrupt on Tx FIFO full. *0* will disallow an interrupt on Tx FIFO full.
D12	Enable Interrupt on Rx FIFO NOT Empty *1* will allow an interrupt on Rx FIFO NOT empty. *0* will disallow an interrupt on Rx FIFO NOT empty.
D13	Enable Interrupt on Rx FIFO NOT Almost Empty *1* will allow an interrupt on Rx FIFO NOT Almost empty. *0* will disallow an interrupt on Rx FIFO NOT Almost empty.
D14	Enable Interrupt on Rx FIFO Almost Full *1* will allow an interrupt on Rx FIFO almost full. *0* will disallow an interrupt on Rx FIFO almost full.
D15	Enable Interrupt on Rx FIFO Full *1* will allow an interrupt on Rx FIFO full. *0* will disallow an interrupt on Rx FIFO full.
D16	Enable Interrupt on Rx FIFO Underflow. *1* will allow an interrupt on Rx FIFO Underflow. *0* will disallow an interrupt on Rx FIFO Underflow.

D17	Enable Interrupt on Rx FIFO Overflow. *1* will allow an interrupt on Rx FIFO Overflow. *0* will disallow an interrupt on Rx FIFO Overflow.
D18	Enable Interrupt on Tx FIFO Overflow. *1* will allow an interrupt on Tx FIFO Overflow. *0* will disallow an interrupt on Tx FIFO Overflow.
D19	Enable Interrupt on Rx Parity Error *1* will allow an interrupt on Rx Parity Error. *0* will disallow an interrupt on Rx Parity Error.
D20	Enable Interrupt on Rx Data Error *1* will allow an interrupt on Rx Data Error. *0* will disallow an interrupt on Rx Data Error.
D21	Enable Interrupt on Rx Receive Busy *1* will allow an interrupt on Rx Receive Busy *0* will disallow an interrupt on Rx Receive Busy
D22	Enable Interrupt on Tx Transmitter Busy. *1* will allow an interrupt on Tx Transmitter Busy. *0* will disallow an interrupt on Tx Transmitter Busy.
D31..23	Reserved

2.2.15 INTERRUPT STATUS: (Offset 0x34 - RC)

The Interrupt Status Register serves as a dual-purpose register. Each bit in this register operates independently of each other. If an interrupt condition is enabled, in the Interrupt Control Register, the appropriate bit, in the Interrupt Status Register, will indicate if an interrupt has occurred or not, and it will continue to indicate this until the software resets that bit. If an interrupt bit is not enabled, in the Interrupt Control Register, then the appropriate bit, in the Interrupt Status Register, will indicate whether or not the condition exists for an interrupt request.

D0	Cable Command 1 Data Rising Edge If this bit is enabled as an interrupt a *1* will indicate that an interrupt on the Cable Command 1 Data Rising Edge has occurred. a *0* will indicate that an interrupt on the Cable Command 1 Data Rising Edge has not occurred. If this bit is not enabled as an interrupt a *1* will indicate that the Cable Command 1 Data is a Rising Edge now. a *0* will indicate that the Cable Command 1 Data is not a rising edge.
D1	Cable Command 1 Data Falling Edge If this bit is enabled as an interrupt a *1* will indicate that an interrupt on the Cable Command 1 Data Falling Edge has occurred. a *0* will indicate that an interrupt on the Cable Command 1 Data Falling Edge has not occurred. If this bit is not enabled as an interrupt a *1* will indicate that the Cable Command 1 Data is a Falling Edge now. a *0* will indicate that the Cable Command 1 Data is not a Falling edge.
D2	Cable Command 1 Data If this bit is enabled as an interrupt a *1* will indicate that an interrupt on Command 1 Data has occurred. a *0* will indicate that an interrupt on Command 1 Data has not occurred. If this bit is not enabled as an interrupt a *1* will indicate that Cable Command 1 Data is currently a '1'. a *0* will indicate that Cable Command 1 Data is not currently a '1'.
D3	Cable Command 2 If this bit is enabled as an interrupt a *1* will indicate that an interrupt on Command 2 has occurred. a *0* will indicate that an interrupt on Command 2 has not occurred. If this bit is not enabled as an interrupt a *1* will indicate that Cable Command 2 is currently a '1'. a *0* will indicate that Cable Command 2 is not currently a '1'.

D4	Cable Command 4 If this bit is enabled as an interrupt a *1* will indicate that an interrupt on Cable Command 4 has occurred. a *0* will indicate that an interrupt on Cable Co mmand 4 has not occurred. If this bit is not enabled as an interrupt a *1* will indicate that Cable Command 4 is currently a '1' a *0* will indicate that Cable Command 4 is not currently a '1'.
D5	Cable Command 5 - Cable Transmitter Driving the Cable If this bit is enabled as an interrupt a *1* will indicate that an interrupt on Cable Command 5 has occurred. a *0* will indicate that an interrupt on Cable Command 5 has not occurred. If this bit is not enabled as an interrupt a *1* will indicate that Cable Command 5 is currently a '1' a *0* will indicate that Cable Command 5 is not currently a '1'.
D6	Cable Command 6 – Cable Receiver Ready to Receive If this bit is enabled as an interrupt a *1* will indicate that an interrupt on Cable Command 6 has occurred. a *0* will indicate that an interrupt on Cable Command 6 has not occurred. If this bit is not enabled as an interrupt a *1* will indicate that Cable Command 6 is currently a '1' a *0* will indicate that Cable Command 6 is not currently a '1'.
D7	Tx Transmit Done If this bit is enabled as an interrupt a *1* will indicate that an interrupt on Tx Transmit Done has occurred. a *0* will indicate that an interrupt on Tx Transmit Done has not occurred. If this bit is not enabled as an interrupt a *1* will indicate that Tx Transmit Done has occurred. a *0* will indicate that Tx Transmit Done has not occurred.
D8	Tx FIFO Empty If this bit is enabled as an interrupt a *1* will indicate that an interrupt on the Tx FIFO empty has occurred. a *0* will indicate that an interrupt on the Tx FIFO empty has not occurred. If this bit is not enabled as an interrupt a *1* will indicate that the Tx FIFO is currently empty a *0* will indicate that the Tx FIFO is not currently empty.
D9	Tx FIFO Almost Empty If this bit is enabled as an interrupt a *1* will indicate that an interrupt on the Tx FIFO almost empty has occurred. a *0* will indicate that an interrupt on the Tx FIFO almost empty has not occurred. If this bit is not enabled as an interrupt a *1* will indicate that the Tx FIFO is currently almost empty. a *0* will indicate that the Tx FIFO is not currently almost empty.
D10	Tx FIFO Almost Full If this bit is enabled as an interrupt a *1* will indicate that an interrupt on the Tx FIFO almost full has occurred. a *0* will indicate that an interrupt on the Tx FIFO almost full has not occurred. If this bit is not enabled as an interrupt a *1* will indicate that the Tx FIFO is currently almost full. a *0* will indicate that the Tx FIFO is not currently almost full.
D11	Tx FIFO Full If this bit is enabled as an interrupt a *1* will indicate that an interrupt on the Tx FIFO full has occurred. a *0* will indicate that an interrupt on the Tx FIFO full has not occurred. If this bit is not enabled as an interrupt a *1* will indicate that the Tx FIFO is currently full. a *0* will indicate that the Tx FIFO is not currently full.

D12	Rx FIFO Empty If this bit is enabled as an interrupt a *1* will indicate that an interrupt on the Rx FIFO empty has occurred. a *0* will indicate that an interrupt on the Rx FIFO empty has not occurred. If this bit is not enabled as an interrupt a *1* will indicate that the Rx FIFO is currently empty a *0* will indicate that the Rx FIFO is not currently empty.
D13	Rx FIFO Almost Empty If this bit is enabled as an interrupt a *1* will indicate that an interrupt on the Rx FIFO almost empty has occurred. a *0* will indicate that an interrupt on the Rx FIFO almost empty has not occurred. If this bit is not enabled as an interrupt a *1* will indicate that the Rx FIFO is currently almost empty. a *0* will indicate that the Rx FIFO is not currently almost empty.
D14	Rx FIFO Almost Full If this bit is enabled as an interrupt a *1* will indicate that an interrupt on the Rx FIFO almost full has occurred. a *0* will indicate that an interrupt on the Rx FIFO almost full has not occurred. If this bit is not enabled as an interrupt a *1* will indicate that the Rx FIFO is currently almost full. a *0* will indicate that the Rx FIFO is not currently almost full.
D15	Rx FIFO Full If this bit is enabled as an interrupt a *1* will indicate that an interrupt on the Rx FIFO full has occurred. a *0* will indicate that an interrupt on the Rx FIFO full has not occurred. If this bit is not enabled as an interrupt a *1* will indicate that the Rx FIFO is currently full. a *0* will indicate that the Rx FIFO is not currently full.
D16	Enable Interrupt on Rx FIFO Underflow. If this bit is enabled as an interrupt a *1* will indicate that an interrupt on Rx FIFO Underflow has occurred. a *0* will indicate that an interrupt on Rx FIFO Underflow has not occurred. If this bit is not enabled as an interrupt a *1* will indicate that Rx FIFO Underflow has occurred. a *0* will indicate that Rx FIFO Underflow is not occurred.
D17	Enable Interrupt on Rx FIFO Overflow. If this bit is enabled as an interrupt a *1* will indicate that an interrupt on Rx FIFO Overflow has occurred. a *0* will indicate that an interrupt on Rx FIFO Overflow has not occurred. If this bit is not enabled as an interrupt a *1* will indicate that Rx FIFO Overflow has occurred. a *0* will indicate that Rx FIFO Overflow has not occurred.
D18	Enable Interrupt on Tx FIFO Overflow. If this bit is enabled as an interrupt a *1* will indicate that an interrupt on Tx FIFO Overflow has occurred. a *0* will indicate that an interrupt on Tx FIFO Overflow has not occurred. If this bit is not enabled as an interrupt a *1* will indicate that a Tx FIFO Overflow has occurred. a *0* will indicate that a Tx FIFO Overflow has not occurred.
D19	Enable Interrupt on Rx Parity Error. If this bit is enabled as an interrupt a *1* will indicate that an interrupt Rx Parity Error has occurred. a *0* will indicate that an interrupt Rx Parity Error has not occurred. If this bit is not enabled as an interrupt a *1* will indicate that a Rx Parity Error has occurred. a *0* will indicate that a Rx Parity Error has not occurred.

D20	Enable Interrupt on Rx Data Error. If this bit is enabled as an interrupt a *1* will indicate that an interrupt on Rx Data Error has occurred. a *0* will indicate that an interrupt on Rx Data Error has not occurred. If this bit is not enabled as an interrupt a *1* will indicate that Rx Data Error has occurred. a *0* will indicate that Rx Data Error has not occurred
D21	Enable Interrupt on Rx Receive Busy. If this bit is enabled as an interrupt a *1* will indicate that an interrupt on Rx Receive Busy has occurred. a *0* will indicate that an interrupt on Rx Receive Busy has not occurred. If this bit is not enabled as an interrupt a *1* will indicate that Rx Receive Busy has occurred. a *0* will indicate that Rx Receive Busy has not occurred
D22	Enable Interrupt on Tx Transmitter Busy. If this bit is enabled as an interrupt a *1* will indicate that an interrupt on Tx Transmitter Busy has occurred. a *0* will indicate that an interrupt on Tx Transmitter Busy has not occurred. If this bit is not enabled as an interrupt a *1* will indicate that Tx Transmitter Busy has occurred. a *0* will indicate that Tx Transmitter Busy has not occurred
D31..23	Reserved

2.2.16 TX CLOCK DIVIDER: (Offset 0x38 - RW)

The Transmit Clock Divider Register has been removed from the PCI-HPDI32A-ASYNC. This function can be accessed thru the upper 16 bits of the Tx Status Block Length Register.
NOTE: This register cannot be accessed using the currently released Windows NT Device Driver.

2.2.17 RESERVED: (Offset 0x3C - RW)

This register is reserved for future use.
NOTE: This register cannot be accessed using the currently released Windows NT Device Driver.

2.2.18 TX FIFO SIZE: (Offset 0x40 – RO)

Upon Power up (or any time FPGA code is reloaded), the logic will fill Tx FIFO and count the number of writes until the FIFO is full. This value is saved as the FIFO size, and the FIFOs are reset. The Counter is only 20 bits wide, the upper 12 bits are undefined.
NOTE: This register cannot be accessed using the currently released Windows NT Device Driver.

D19..D0	Tx FIFO Size. Number of D32 Words that the Tx FIFO Holds.
D31..D20	Undefined

2.2.19 RX FIFO SIZE: (Offset 0x44 – RO)

Upon Power up (or any time FPGA code is reloaded), the logic will fill Rx FIFO and count the number of writes until the FIFO is full. This value is saved as the FIFO size, and the FIFOs are reset. The Counter is only 20 bits wide, the upper 12 bits are undefined.
NOTE: This register cannot be accessed using the currently released Windows NT Device Driver.

D19..D0	Rx FIFO Size. Number of D32 Words that the Rx FIFO Holds.
D31..D20	Undefined

2.2.20 TX FIFO WORD COUNT: (Offset 0x48 – RO)

This counter tracks the number of D32 Words in the Tx FIFO. Every time a word is written to the FIFO, the count is incremented. When a D32 Word is read, the count is decremented. The Counter is only 20 bits wide, the upper 12 bits are undefined.
NOTE: This register cannot be accessed using the currently released Windows NT Device Driver.

D19..D0	Tx FIFO Word Count. Number of D32 Words currently in the Tx FIFO.
D31..D20	Undefined

2.2.21 RX FIFO WORD COUNT: (Offset 0x4C – RO)

This counter tracks the number of D32 Words in the Rx FIFO. Every time a word is written to the FIFO, the count is incremented. When a D32 Word is read, the count is decremented. The Counter is only 20 bits wide, the upper 12 bits are undefined.
NOTE: This register cannot be accessed using the currently released Windows NT Device Driver.

D19..D0	Rx FIFO Word Count. Number of D32 Words currently in the Rx FIFO.
D31..D20	Undefined

2.2.22 INTERRUPT EDGE REGISTER: (Offset 0x50 – RW)

Thirty-two bit register to define interrupt source to individually define as edge or level sensitive. Bits are the same order as the interrupt source. Default is 0, Level Triggered Interrupts to maintain compatibility with existing devices.
NOTE: This register cannot be accessed using the currently released Windows NT Device Driver.

D31..D0	Interrupt Edge Select Individual Interrupt level or edge trigger select. 0 = Level Triggered Interrupts. 1 = Edge Triggered Interrupts.

2.2.23 INTERRUPT HI/LO REGISTER: (Offset 0x54 – RW)

Thirty-two bit register to define interrupt source as active hi or active lo. Bits are in same order as interrupt source. Default is 1, Active High Interrupts to maintain compatibility with existing devices.
NOTE: This register cannot be accessed using the currently released Windows NT Device Driver.

.

D31..D0	Interrupt Hi/Lo Select. Individual Interrupt Hi or Lo active level selection. 0 = Active Low Interrupt. 1 = Active Hi Interrupt

3 PROGRAMMING

3.1 INITIALIZATION

Initializing the PCI-HPDI32A-ASYNC Card will generally need to be done only once by the software, unless the mode needs to change. The software is responsible for tracking any changes; for making all changes necessary to meet the needs of the application; and for making all the adjustments when requirements change. Most of the configuration status can be determined by reading the Board Control Register. Upon system reset, and also after a board reset is performed, the board will be in a state where the following initialization will apply:

1. all cable data transceivers will be in their default state, Tri-state off
2. all interrupts will be disabled;
3. the FIFOs will be empty;
4. the receive logic will be disabled.
5. the transmit logic will be disabled.
6. the board will be driving Cable Command 5 and Cable Command 6.
7. the board will be driving Cable Command 1, default transmit data line.
8. all RW registers will be set to 0 (Board Control Register bits are set to 0 on system reset)

3.2 RESETS

There are three (3) bits on this board that are used as resets to the local side. All three bits are located in the Board Control Register. These bits perform a reset when the software writes a 1 to them. After writing a 1, the software does not need to return to clear the bits, the bits operate as a self-timed pulse that will return to 0 after they have been asserted long enough to perform the reset(s). For further details on the resets refer to the PCI-HPDI32A-ASYNC Register Map, See Table 2.1-1.

D0	**Board Reset** will reset the local logic, clear the FIFOs, and place the appropriate registers into a known state.
D1	**Tx FIFO Reset** will reset the Tx FIFOs.
D2	**Rx FIFO Reset** will reset the Rx FIFOs.

The FIFOs are reset by either a hardware system reset of the board, a software FIFO reset (Board Control Register bit 1 or bit 2), or a software board reset (Board Control Register bit 0).

3.3 FIFOs

The FIFOs flags are used to indicate the current fill level of the FIFO. There are four flags for Tx and four flags for Rx. These flags are labeled and defined as follows:

Empty:
Almost Empty:
Almost Full:
Full:
There are 0 true signals. The almost registers are used for programming the FIFOs, states are:

Empty – 0
Almost Empty – programmable level
Almost Full – programmable level.
Full – depth

Use bits 1 and 2 of the BCR a board reset will not program the FIFOs.

The FIFO Almost Registers are used to program the Almost Empty and Almost Full flag levels.
The default is 0x01000008 which will give you levels of
Almost Empty – 0x0008 – 8 D32 words above empty.
Almost Full – 0x0100 – Almost Full is 256 D32 words below Full.

In addition there are 3 FIFO Event Flags that are part of the FIFO system.
Rx FIFO Underflow – Read of the Rx FIFO while the Rx FIFO is Empty.
Rx FIFO Overflow – Write to the Rx FIFO while the Rx FIFO is Full.
Tx FIFO Overflow – Write to the Tx FIFO while the Tx FIFO is Full.

The Rx FIFO events are cleared by a Rx FIFO Reset. The Tx FIFO Event flag is cleared by a Tx FIFO reset.

3.4 INTERRUPTS

In order for this board to generate interrupts to the PCI bus, Bits 8, 11, and 16 of the PLX Interrupt Control/Status Register must be set to a 1. These bits must be set to a 1 in order for the interrupts to occur.

The next step in initializing the interrupt, is to specify which interrupts are to be allowed. The board allows the software to enable some interrupts and leave others disabled. This is accomplished by writing a 1 to the appropriate bits in the Interrupt Control Register (ICR). For example, to enable the interrupt for FIFO Almost Empty, the software will need to write a 1 to bit 13 of the ICR. This bit will not need to be changed again until the need to disable this specific interrupt. This enable can be a one time process which will allow many interrupts to occur.

Multiple interrupts from the same cause are prevented via the Interrupt Status Register (ISR). Writing to the ISR is the method by which the software acknowledges to the board that it has received the previous interrupt request and signals to the board that it may now generate any other interrupts that may occur. This register will need some attention from the software after each interrupt has occurred. Following the previous example, when this interrupt has occurred, the software will find that bit 13 of the ISR is now a 1, indicating that a FIFO Almost Empty interrupt has occurred. This bit will remain a 1 and will not allow any additional interrupts to be generated until the software performs a write to this register. To re-enable the FIFO Almost Empty interrupt, the software must write a 1 to bit 13. This will clear the occurrence of the interrupt.

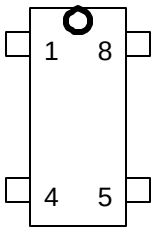
The enabling, latching, and clearing of the FIFO Almost Empty status bit will not effect the other bits of the register. This means that if the software receives the interrupt for FIFO Almost Empty (the only interrupt currently enabled) the software may very well find that the FIFO is now Empty, (indicated by bit 12 being a 1). Since this bit is not enabled as an interrupt, it is acting as a status bit. If it is enabled now, it will immediately generate an interrupt.

4 HARDWARE CONFIGURATION

4.1 THE ON-BOARD TRANSMIT CLOCK

The on-board oscillator, U11, is used as the transmit clock while in test mode. The oscillator is factory installed at 40MHz. This oscillator can be changed in the field by the end user. The maximum frequency supported by this clock is 40Mhz.

Figure 4.1-1 Oscillator Pinout

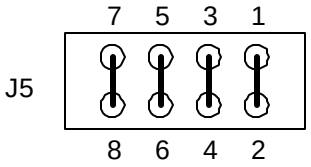


PIN	NAME
1	NC
4	GND
5	OUT
8	VCC

4.2 JUMPERS (J5)

J5 is a header consisting of four individual jumpers.

Figure 4.2-1 Jumper J5



J5: 1 to J5: 2 EEPROM Configuration
This jumper connects the PNP EEPROM to the PCI chipset for power-up configuration. This jumper is intended for factory use only and should always be installed.

J5: 3 to J5: 4 FPGA Reload
This jumper connects the FPGA Reload to the local PCI Reset. When this jumper is installed, the FPGA will reload when the PCI is reset. If the jumper is off, the FPGA will reload only at power-up. This jumper is intended for factory use only and should always be installed.

J5: 5 to J5: 6 Board Jumper 0
This jumper is Readable thru the Board Status Register and can be used to identify a particular board if multiple boards are installed in a system.

J5: 7 to J5: 8 Board Jumper 1
This jumper is Readable thru the Board Status Register and can be used to identify a particular board if multiple boards are installed in a system.

4.3 CABLE CONNECTOR

The 80-pin user connector (reference designator: P1) is manufactured by Robinson Nugent, the part number is P50E-080-P1-SR1-TG. The part number for the mate is P50E-080-S-TG, (50 mil. cabling is suggested for twisted pair), or P25E-080S-TG (25 mil. cabling may be used for multi-drop capability, but with loss of twisted pair).

Figure 4.3-1 Cable Pinout

Pin No.	Cable Signal Name
1	CABLE CLOCK + / Not Used
2	CABLE CLOCK -/ Not Used
3	Cable Command 0 + / Not Used
4	Cable Command 0 - / Not Used
5	Cable ASYNC Tx Data +
6	Cable ASYNC Tx Data -
7	Cable ASYNC Rx Data +
8	Cable ASYNC Rx Data -
9	Cable Command 3 + / Not Used
10	Cable Command 3 - / Not Used
11	Cable Command 4 + / Not Used
12	Cable Command 4 - / Not Used
13	Tx Transmitter Running +
14	Tx Transmitter Running -
15	Rx Receiver Running +
16	Rx Receiver Running -
17	CABLE D0 +
18	CABLE D0 -
19	CABLE D1 +
20	CABLE D1 -
21	CABLE D2 +
22	CABLE D2 -
23	CABLE D3 +
24	CABLE D3 -
25	CABLE D4 +
26	CABLE D4 -
27	CABLE D5 +
28	CABLE D5 -
29	CABLE D6 +
30	CABLE D6 -
31	CABLE D7 +
32	CABLE D7 -
33	CABLE D8 +
34	CABLE D8 -
35	CABLE D9 +
36	CABLE D9 -
37	CABLE D10 +
38	CABLE D10 -
39	CABLE D11 +
40	CABLE D11 -

Pin No.	Cable Signal Name
41	CABLE D12 +
42	CABLE D12 -
43	CABLE D13 +
44	CABLE D13 -
45	CABLE D14 +
46	CABLE D14 -
47	CABLE D15 +
48	CABLE D15 -
49	CABLE D16 +
50	CABLE D16 -
51	CABLE D17 +
52	CABLE D17 -
53	CABLE D18 +
54	CABLE D18 -
55	CABLE D19 +
56	CABLE D19 -
57	CABLE D20 +
58	CABLE D20 -
59	CABLE D21 +
60	CABLE D21 -
61	CABLE D22 +
62	CABLE D22 -
63	CABLE D23 +
64	CABLE D23 -
65	CABLE D24 +
66	CABLE D24 -
67	CABLE D25 +
68	CABLE D25 -
69	CABLE D26 +
70	CABLE D26 -
71	CABLE D27 +
72	CABLE D27 -
73	CABLE D28 +
74	CABLE D28 -
75	CABLE D29 +
76	CABLE D29 -
77	CABLE D30 +
78	CABLE D30 -
79	CABLE D31 +
80	CABLE D31 -

Appendix A PLX REGISTER PROGRAMMING

The following Appendix describes the registers of the PLX PCI Bus Interface that is used on the PMC-HPDI32A-CDC.

A.1 Configuration EEPROM

During configuration, the PLX will initialize itself from a EEPROM that is programmed at the factory. The following values are loaded into the PLX and are used to configure the board for operation.

Table A Configuration EEPROM Contents.

PCI Offset from base Addr	Size	Register Name	Value after Reset	PCI CFG Register Address
0x70	D32	Device ID / Vendor ID	0x9080/ 0x10B5	0x00
-	D32	Class Code / Revision ID	0x07800003	0x08
-	D32	Max / Min Latency / Int. Pin / Int. Line Routing Value	0x00010000	0x3C
0x78	D32	Mailbox 0 User defined	0x00000000	-
0x7c	D32	Mailbox 1 User defined	0x00000000	-
0x00	D32	Range PCI to Local Address Space 0	0xFFFFFE00	0x18
0x04	D32	Local Base Address (remap) Local Address Space 0	0x00000001	-
0x08	D32	Local Mode / Arbitration register	0x0000, Not Used	-
0x0C	D32	Big / little Endian descriptor	0x0000	-
0x10	D32	Range for PCI to local expansion ROM	0x0000, Not Used	-
0x14	D32	Local Base Address Remap PCI to Local Expansion ROM	0x0000, Not Used	-
0x18	D32	Bus region descriptors for PCI to Local Address Space 0	0x42000143	-
0x1C	D32	Range for direct Master to PCI	0x0000, Not Used	-
0x20	D32	Local Base Address for direct Master to PCI Memory	0x0000, Not Used	-
0x24	D32	Local Base Address for direct Master to PCI IO/CFG	0x0000, Not Used	-
0x28	D32	Base Address (remap) for direct Master to PCI	0x0000, Not Used	-
0x2C	D32	PCI Configuration Address Register for direct Master to PCI IO/CFG	0x0000, Not Used	-
-	D32	Subsystem ID / Subsystem Vendor ID	0x2400/ 0x10B5	0x2C
0xF0	D32	Range PCI to Local Address Space 1	0x0000, Not Used	-
0xF4	D32	Local Base Address Remap PCI to Local Address Space 1	0x0000, Not Used	-
0xF8	D32	Bus region descriptors for PCI to Local Address Space 1	0x0000, Not Used	-
-	D32	PCI Base Address for Local Expansion ROM	0x0000, Not Used	0x30

A.2 PCI CONFIGURATION REGISTER BIT MAPS

The PCI-HPDI32-CDC Card complies with the **plug-n-play** concept. That is, at the time of power-up, an attempt will be made by the CPU to set up the board to meet the configuration requirements of the system. In doing this, the CPU will map the amount of I/O space requested by the PCI-HPDI32-CDC Card and return the configuration base address into the PCI Configuration Register (Offset 0x18) of this Board.

The configuration registers are usually programmed by the system BIOS during system startup.

Table B Pci Configuration Register Mapping

Size	Access R	W	Register Name	Value after Reset	PCI CFG register address
D32	yes	no	Device ID/Vendor ID	0x9080/ 0x10B5	0x00
D32	yes	yes	Status/Command	0x00010000	0x04
D32	yes	no	Class Code/Revision ID	0x07800003	0x08
D32	yes	Yes 15..0	BIST/Header Type/Latency Timer/Cache Line Size	0x00000000	0x0C
D32	yes	Yes	PCI Base Address for Memory Mapped Runtime Registers	0x00000000	0x10
D32	yes	Yes	PCI Base Address for 1 for I/O Mapped Configuration Registers	0x00000001	0x14
D32	yes	Yes	PCI Base Address 2 for Local Address Space 0	0x00000000	0x18
D32	yes	Yes	PCI Base Address 3 for Local Address Space 1	0x00000000	0x1C
D32	yes	No	Unused Base Address	0x00000000	0x20
D32	yes	No	Unused Base Address	0x00000000	0x24
D32	yes	No	Cardbus CIS Pointer (Not Supported)	0x00000000	0x28
D32	yes	No	Subsystem ID/Subsystem Vendor ID	0x240001B5	0x2C
D32	yes	Yes	PCI Base Address to Local Expansion ROM	0x00000001	0x30
D32	yes	No	Reserved	0x00000000	0x34
D32	yes	No	Reserved	0x00000000	0x38
D32	yes	Yes 7..0	Max_lat/Min_Gnt/Interrupt Pin/Interrupt Line	0x00010000	0x3C

All registers may be written to or read from in byte, word, or Lword accesses.

A.2.1 PCI Configuration ID Register (PCI Configuration Offset 0x00)

D0..15	Vendor ID- Identifies the manufacturer of the device. Defaults to the PCI SIG issued vendor ID of PLX (0x10B5) if no serial EEPROM is present and pin NB# (no local bus initialization) is asserted low.
D16..31	Device ID - Identifies the particular device. Defaults to the PLX part number for PCI interface chip (0x9080) if no serial EEPROM is present and pin NB#(no local bus initialization) is asserted low.

A.2.2 PCI Command Register (PCI Configuration Offset 0x04)

D0	I/O Space A value of 1 allows the device to respond to I/O space accesses. A value of 0 disables the device from responding to I/O space accesses.
D1	Memory Space A value of 1 allows the device to respond to memory space accesses. A value of 0 disables the device from responding to memory space accesses.
D2	Master Enable. Controls a device's ability to act as a master on the PCI bus. A value of 1 allows the device to behave as a bus master. A value of 0 disables the device from generating bus master accesses. This bit must be set for the PCI 9080 to perform Direct Master or DMA cycles.
D3	Special Cycle. (This bit is not supported.)
D4	Memory Write/Invalidate. A value of 1 enables memory write/invalidate. A value of 0 disables memory write/invalidate.
D5	VGA Palette Snoop. (This bit is not supported.)
D6	Parity Error Response A value of 0 indicates that a parity error is ignored and operation continues. A value of 1 indicates that parity checking is enabled.
D7	Wait Cycle Control. Controls whether the device does address/data stepping. A 0 value indicates the device never does stepping. A value of 1 indicates that the device always does stepping. **Note:** *Hardcoded to 0.*
D8	SERR# Enable A value of 1 enables the SERR# driver. A value of 0 disables the driver.
D9	Fast Back-to-Back Enable. Indicates what type of fast back-to-back transfers a Master can perform on the bus. A value of 1 indicates that fast back-to-back transfers can occur to any agent on the bus. A value of 0 indicates fast back-to-back transfers can only occur to the same agent as the previous cycle.
D10..15	Reserved

A.2.3 PCI Status Register (PCI Configuration Offset 0x06)

D0..5	Reserved
D6	If high, supports User definable features. This bit can only be written from the local side. It is read-only from the PCI side.
D7	Fast Back-to-Back Capable. When this bit is set to a 1, it indicates the adapter can accept fast back-to-back transactions. A 0 indicates the adapter cannot.
D8	Master Data Parity Error Detected This bit is set to a 1 when three conditions are met: 1. the PCI9080 asserted PERR# itself or observed PERR# asserted. 2. the PCI9080 was the bus master for the operation in which the error occurred. 3. the Parity Error Response bit in the Command Register is set. Writing a 1 to this bit clears the bit to a 0.
D9..10	DEVSEL Timing. Indicates timing for DEVSEL# assertion. A value of 01 indicates a medium decode. **Note:** *Hardcode to 01.*
D11	Target Abort When this bit is set to a 1, this bit indicates the PCI9080 has signaled a target abort. Writing a 1 to this bit clears the bit (0).
D12	Received Target Abort When set to a 1, this bit indicates the PCI9080 has signaled a target abort. Writing a 1 to this bit clears the bit (0).
D13	Master Abort When set to a 1, this bit indicates the PCI9080 has generated a master abort signal. Writing a 1 to this bit clears the bit (0).
D14	Signal System Error When set to a 1, this bit indicates the PCI9080 has reported a system error on the SERR# signal. Writing a 1 to this bit clears the bit (0).
D15	Detected Parity Error When set to a 1, this bit indicates the PCI9080 has detected a PCI bus parity error, even if parity error handling is disabled (the Parity Error Response bit in the Command Register is clear). One of three conditions can cause this bit to be set: 1. the PCI9080 detected a parity error during a PCI address phase. 2. the PCI9080 detected a data parity error when it was the target of a write. 3. the PCI9080 detected a data parity error when performing a master read operation. Writing a 1 to this bit clears the bit (0).

A.2.4 PCI Revision ID Register (PCI Configuration Offset 0x08)

D0..7	Revision ID. The silicon revision of the PCI9080.

A.2.5 PCI Class Code Register (PCI Configuration Offset 0x09..0B)

D0..7	Register level programming interface 0x00 = Queue Ports at 0x40 and 0x44. 0x01 = Queue Ports at 0x40 and 0x44, and Int Status and Int Mask at 0x30 and 0x34, respectively.
D8..15	Sub-class Code 0x80 = Other Communications device.
D16..D23	Base Class Code. 0x07 = Communications Device.

A.2.6 PCI Cache Line Size Register (PCI Configuration Offset 0x0C)

D0..7	System cache line size in units of 32-bit words.

A.2.7 PCI Latency Timer Register (PCI Configuration Offset 0x0D)

D0..7	PCI Latency Timer. Units of PCI bus clocks, the amount of time the PCI9080, as a bus master, can burst data on the PCI bus.

A.2.8 PCI Header Type Register (PCI Configuration Offset 0x0E)

D0..6	Configuration Layout Type. Specifies the layout of bits 0x10 through 0x3F in configuration space. Only one encoding 0 is defined. All other encodings are reserved.
D7	Header Type. A 1 indicates multiple functions. A 0 indicates a single function.

A.2.9 PCI Built-In Self Test (BIST) Register (PCI Configuration Offset 0x0F)

D0..3	A value of 0 means the device has passed its test. Non-zero values mean the device failed. Device specific failure codes can be encoded in the non-zero value.
D4..5	Reserved, Device returns 0.
D6	PCI writes a 1 to invoke BIST. Generates an interrupt to local bus. Local bus resets the bit when BIST is complete. Software should fail device if BIST is not complete after 2 seconds. Refer to Runtime registers for interrupt control/status.
D7	Return 1 if device supports BIST. Return 0 if the device is not BIST compatible.

A.2.10 PCI Base Address Register for Memory Access to Runtime Registers (PCI Configuration Offset 0x010)

D0	Memory space indicator A value of 0 indicates register maps into Memory space. A value of 1 indicates the register maps into I/O space **Note:** *Hardcoded to 0.*
D1..2	Location of register: 00 - Locate anywhere in 32 bit memory address space 01 - Locate below 1 Mbyte memory address space 10 - Locate anywhere in 64 bit memory address space 11 – Reserved **Note:** *Hardcoded to 0.*
D3	Prefectchable. A value of 1 indicates there are no side effects on reads. This bit has no effect on the operation of the PCI 9080. **Note:** *Hardcoded to 0.*
D4..7	Memory Base Address. Memory base address for access to Local, Runtime and DMA registers. (default is 256 bytes.) **Note:** *Hardcoded to 0.*
D8..31	Memory Base Address. Memory base address for access to Local, Runtime, and DMA registers.

A.2.11 PCI Base Address Register for I/O Access to Runtime Registers (PCI Configuration Offset 0x14)

D0	Memory space indicator A value of 0 indicates register maps into Memory space. A value of 1 indicates the register maps into I/O space. **Note:** *Hardcoded to 1*
D1	Reserved
D2..7	I/O Base Address. Base Address for I/O access to runtime registers. (Minimum Block Size = 128 bytes.) **Note:** *Hardcoded to 0*
D8..31	I/O Base Address. Base Address for I/O access to Local, Runtime, and DMA Registers.

A.2.12 PCI Base Address Register for Memory Access to Local Address Space 0 (PCI Configuration Offset 0x18)

D0	Memory space indicator A value of 0 indicates register maps into Memory space. A value of 1 indicates the register maps into I/O space. (Specified in Local Address Space 0 Range Register)
D1..2	Location of register (if memory space). Location values: 00 - Locate anywhere in 32 bit memory address space 01 - Locate below 1 Mbyte memory address space 10 - Locate anywhere in 64 bit memory address space 11 - Reserved (Specified in Local Address Space 0 Range Register,.)
D3	Prefetchable (if memory space) A value of 1 indicates there are no side effects on reads. This bit reflects the value of bit 3 in the LASORR register and provides only status to the system. This bit has no effect on the operation of the PCI 9080. Prefetching features of this address space are controlled by the associated Bus Region Descriptor Register. (Specified in LASORR register.) If I/O Space, bit 3 is included in the base address.
D4..31	Memory Base Address Memory base address for access to Local Address Space 0.

A.2.13 PCI Base Address Register for Memory Access to Local Address Space 1 (PCI Configuration Offset 0x1C)

D0	Memory space indicator A value of 0 indicates register maps into Memory space. A value of 1 indicates the register maps into I/O space. (Specified in Local Address Space 1 Range Register.)
D1..2	Location of register (if memory space). Location values: 00 - Locate anywhere in 32 bit memory address space 01 - Locate below 1 Mbyte memory address space 10 - Locate anywhere in 64 bit memory address space 11 - Reserved (Specified in Local Address Space 1 Range Register.)
D3	Prefetchable (if memory space) A value of 1 indicates there are no side effects on reads. This bit reflects the value of bit 3 in the LAS1RR register and provides only status to the system. This bit has no effect on the operation of the PCI 9080. Prefetching features of this address space are controlled by the associated Bus Region Descriptor Register. (Specified in LAS1RR register.) If I/O Space, bit 3 is included in the base address.
D4..31	Memory Base Address Memory base address for access to Local Address Space 1.

A.2.14 PCI Base Address Register (PCI Configuration Offset 0x20)

D0..31	Reserved

A.2.15 PCI Base Address Register (PCI Configuration Offset 0x24)

D0..31	Reserved

A.2.16 PCI Cardbus CIS Pointer Register (PCI Configuration Offset 0x28)

D0..31	Cardbus Information Structure Pointer for PCMCIA. (Not supported.)

A.2.17 PCI Subsystem Vendor ID Register (PCI Configuration Offset 0x2C)

D0..15	Subsystem Vendor ID (unique add-in board Vendor ID).

A.2.18 PCI Subsystem ID Register (PCI Configuration Offset 0x2E)

D0..15	Subsystem ID (unique add-in board Device ID).

A.2.19 PCI Expansion ROM Base Register (PCI Configuration Offset 0x30)

D0	Address Decode Enable A value of 1 indicates the device accepts accesses to the expansion ROM address. A value of 0 indicates the device does not accept accesses to expansion ROM space. Should be set to 1 by PCI host if expansion ROM is present.
D1..10	Reserved
D11..31	Expansion ROM Base Address (upper 21bits)

A.2.20 PCI Interrupt Line Register (PCI Configuration Offset 0x3C)

D0..7	Interrupt Line Routing Value. Indicates which input of the system interrupt controller(s) to which the interrupt line of the device is connected.

A.2.21 PCI Interrupt Pin Register (PCI Configuration Offset 0x3D)

D0..7	Interrupt Pin register. Indicates which interrupt pin the device uses. The following values are decoded: 0=No Interrupt Pin 1=INTA# 2=INTB# 3=INTC# 4=INTD# **Note:** *PCI 9080 supports only one PCI interrupt pin (INTA#).*

A.2.22 PCI Min_Gnt Register (PCI Configuration Offset 0x3E)

D0..7	Min_Gnt. Used to specify how long a burst period the device needs assuming a clock rate of 33 MHz. Value is multiple of ¼ use increments.

A.2.23 PCI Max_Lat Register (PCI Configuration Offset 0x3F)

D0..7	Max_Lat. Specifies how often the device mu st gain access to the PCI bus. Value is multiple of ¼ user increments.

A.3 LOCAL CONFIGURATION REGISTERS

Table C LOCAL CONFIGURATION REGISTERS

PCI Offset from base Addr	Size	Access R	Access W	Register Name	Value after Reset
0x00	D32	yes	yes	Range for PCI to Local Address Space 0	0xFFFFE00
0x04	D32	yes	yes	Local Base Address (Re-map) for PCI to Local Address Space 0	0x00000001
0x08	D32	yes	no	Mode Arbitration	
0x0C	D32	yes	no	Big/Little Endian Descriptor	
0x10	D32	yes	yes	Range for PCI to Local Expansion ROM	0xFFF00000
0x14	D32	yes	Bit Dep.	Local Base Address (Re-map) for PCI to Local Expansion ROM and BREQo control	0x00000000
0x18	D32	yes	yes	Bus Region Descriptions for PCI Local Accesses	0x42000143 (Cx Mode)
0x1C	D32	yes	Bit Dep.	Range for Direct Master to PCI	0x00000000
0x20	D32	yes	Bit Dep.	Local Base Address for Direct Master to PCI Memory	0x00000000
0x24	D32	yes	Bit Dep.	Local Base Address for Direct Master to PCI Memory IO/CFG	0x00000000
0x28	D32	yes	Bit Dep.	PCI Base Address (Re-map) for Direct Master to PCI	0x00000000
0x2C	D32	yes	yes	PCI Configuration Address Register for Direct Master to PCI IO/CFG	0x00000000
0xF0	D32	yes	yes	Range for PCI to Local Address Space 1	0x00000000
0xF4	D32	yes	yes	Local Base Address (Remap) for PCI to Local Address Space 1	0x00000000
0xF8	D32	yes	yes	Local Bus Region Descriptor (Space 1) for PCI to Local Accesses	0x00000000

A.3.1 Local Address Space 0 Range Register for PCI to Local bus (PCI Offset 0x00)

D0	Memory space indicator A value of 0 indicates Local address space 0 maps into PCI memory space. A value of 1 indicates address space 0 maps into PCI I/O space.
D1..2	If mapped into memory space, encoded as follows (D1 being the LSB): Meaning 00-locate anywhere in 32 bit PCI address space 01-locate below 1 Meg in PCI address space 10-locate anywhere in 64 bit PCI address space 11-reserved If mapped into I/O space, bit 1 must be a 0. Bit 2 is included with bits 3 through 31 to indicate decoding range.
D3	If mapped into memory space, a value of 1 indicates that reads are pre-fetchable (bit has no effect on the PCI9080, but it is used for system status). If mapped into I/O space, bit is included with bits [31:2] to indicate decoding range. If mapped into I/O space, bit is included with bits 2 through 31 to indicate decoding range
D4..31	Specifies which PCI address bits to use for decoding a PCI access to local bus space 0. Each bit corresponds to a PCI address bit. Bit 31 corresponds to Address bit 31. Write a value of 1 to all others (used in conjunction with PCI Configuration register 0x18). Default is 1 Meg.

A.3.2 Local Address Space 0 Local Base Address (Re-map) Register for PCI to Local Bus (PCI Offset 0x04)

D0	Space 0 Enable A 1 value enables decoding of PCI addresses for Direct Slave access to local space 0. A value of 0 disables Decode. If this bit is set to 0, the PCI BIOS may not allocate (assign) the base address for Space 0. **Note:** *Must be set to 1 for any Direct Slave access to Space 0.*
D1	Reserved
D2..3	If local space 0 is mapped into memory space, bits are not used. If mapped into I/O space, bit is included with bits 4 through 31 for re-mapping.
D4..31	Re-map of PCI Address to Local Address Space 0 into a Local Address Space. The bits in this register re-map (replace) the PCI Address bits used in decode as the Local Address bits. **Note:** *Remap Address value must be multiple of Range (not the Range register)*

A.3.3 Mode/Arbitration Register (PCI Offset 0x08)

D0..7	Local bus Latency Timer Number of local bus clock cycles before negating HOLD and releasing the local bus. This timer is also used with bit 27 to delay BREQ input to give up the local bus only when this timer expires.
D8..15	Local bus Pause Timer Number of local bus clock cycles before reasserting HOLD after releasing the local bus. **Note:** *Applicable only to DMA operation.*
D16	Local bus Latency Timer Enable A value of 1 enables latency timer.
D17	Local bus Pause Timer Enable A value of 1 enables pause timer.
D18	Local bus BREQ Enable A value of 1 enables local bus BREQ input. When the BREQ input is active, PCI 9080 negates HOLD and releases the local bus.
D19..20	DMA Channel Priority A value of 00 indicates a rotational priority scheme. A value of 01 indicates Channel 0 has priority. A value of 10 indicates Channel 1 has priority. A value of 11 is reserved.
D21	Local bus direct slave give up bus mode When set to 1, PCI 9080 negates HOLD and releases the local bus when the Direct Slave write FIFO becomes empty during a Direct Slave write or when the Direct Slave read FIFO becomes full during a Direct Slave read.
D22	Direct slave LLOCKo# Enable A value of 1 enables PCI Direct Slave locked sequences. A value of 0 disables Direct Slave locked sequences.
D23	PCI Request Mode A value of 1 causes PCI9080 to negate REQ when it asserts FRAME during a master cycle. A value of 0 causes PCI 9080 to leave REQ asserted for the entire bus master cycle.
D24	PCI Rev 2.1 Mode When set to 1, PCI 9080 operates in Delayed Transaction mode for Direct Slave Reads. PCI 9080 issues a RETRY and prefetches the read data.
D25	PCI Read No Write Mode A value of 1 forces a retry on writes if read is pending. A value of 0 allows writes to occur while read is pending.
D26	PCI Read with Write Flush Mode A value of 1 submits a request to flush a pending read cycle if a write cycle is detected. A value of 0 submits a request to not effect pending reads when a write cycle occurs. (PCI v2.1 compatible)
D27	Gate the Local Bus Latency Timer with BREQ If this bit is set to 0, PCI 9080 gives up the local bus during Direct Slave or DMA transfer after the current cycle (if enabled and BREQ is sampled). If this bit is set to 1, PCI 9080 gives up the local bus only (if BREQ is sampled) and the Local Bus Latency Timer is enabled and expires during Direct Slave or DMA transfer.
D28	PCI Read No Flush Mode A value of 1 submits request to not flush the read FIFO if PCI read cycle completes (Read Ahead mode). A value of 0 submits request to flush read FIFO if PCI read cycle completes.

D29	Reads Device or Vendor ID If set to 0, reads from the PCI Configuration Register address 0x00 and returns the Device ID and Vendor ID. If set to 1, reads from the PCI Configuration Register address 0x00 and returns the Subsystem and Subsystem Vendor ID.
D30..31	Reserved

A.3.4 Big/Little Endian Descriptor Register (PCI Offset 0x0C)

D0	Configuration Register Big Endian Mode A value of 1 specifies use of Big Endian data ordering for local accesses to the configuration registers. A value of 0 specifies Little Endian ordering. Big Endian mode can be specified for configuration register accesses by asserting the BIGEND# pin during the address phase of the access.
D1	Direct Master Big Endian Mode A value of 1 specifies use of Big Endian data ordering for Direct Master accesses. A value of 0 specifies Little Endian ordering. Big BIGEND# input pin during the address phase of the access.
D2	Direct Slave Address Space 0 Big Endian Mode A value of 1 specifies use of Big Endian data ordering for Direct Slave accesses to Local Address space 0. A value of 0 specifies Little Endian ordering.
D3	Direct Slave Address Expansion ROM 0 Big Endian Mode A value of 1 specifies use of Big Endian data ordering for Direct Slave accesses to Expansion ROM. A value of 0 specifies Little Endian ordering.
D4	Big Endian Byte Lane Mode A vlaue of 1 specifies that in Big Endian mode, use byte lanes 31:16 for a bit local bus and byte lanes 31:24 for an 8 bit local bus. A value of 0 specifies that in Big Endian mode, byte lanes 15:0 be used for a 16 bit local bus byte lanes 7:0 for an 8 bit local bus.
D5	Direct Slave Address Space 1 Big Endian Mode A value of 1 specifies use of Big Endian data ordering for Direct Slave accesses to loal Address Space 1. A value of 0 specifies Little Endian ordering.
D6	DMA Channel 1 Big Endian Mode A value of 1 specifies use of Big Endian data ordering for DMA Channel 1 accesses to the local Address Space. A value of 0 specifies Little Endian ordering.
D7	DMA Channel 0 Big Endian Mode A value of 1 specifies use of Big Endian data ordering for DMA Channel 0 accesses to the local Address Space. A value of 0 specifies Little Endian ordering.
D8..31	Reserved

A.3.5 Local Expansion ROM Range Register for PCI to Local Bus (PCI Offset 0x10)

D0..10	Reserved
D11..31	Specifies which PCI address bits will be used for decoding a PCI to local bus expansion ROM. Each of the bits corresponds to an Address bit. Bit 31 corresponds to Address bit 31. Write a value of 1 to all bits to be included in decode and a 0 to all others (Used in conjunction with PCI Configuration register 0x30). Default is 64 Kbytes.

A.3.6 Local Expansion ROM Local Base Address (Re-map) register for PCI to Local Bus and BREQo Control (PCI Offset 0x14)

D0..3	Direct Slave BREQo (Backoff Requests Out) Delay Clocks. Number of local bus clocks in which a Direct Slave HOLD request is pending and a Local Direct Master access is in progress and not being granted the bus (HOLDA) before asserting BREQo. Once asserted, BREQo remains asserted until the PCI900 receives HOLDA (LSB = 8 or 64 clocks).
D4	Local Bus BREQo Enable A 1 value enables the PCI9080 to assert the BREQo output.
D6..10	Reserved
D11..31	Re-map of PCI Expansion ROM space into a Local address space. The bits in this register re-map (replace) the PCI address bits used in decode as the Local address bits.

A.3.7 Local Address Space 0/Expansion ROM Bus Region Descriptor Register (PCI Offset 0x18)

D0..1	Memory Space 0 Local Bus Width A value of 00 indicates a bus width of 8 bits A value of 01 indicates a bus width of 16 bits A value of 10 or 11 indicates a bus width of 32 bits The bus width is forced to 16 bits for the Sx mode.
D2..5	Memory Space 0 Internal Wait States (data to data; 0-15 wait states).
D6	Memory Space 0 Ready Input Enable A 1 value enables Ready input. A value of 0 disables the Ready input.
D7	Memory Space 0 BTERM# Input Enable A 1 value enables BTERM# input. A value of 0 disables the BTERM# input.
D8	Memory Space 0 Prefetch Disable If mapped into memory space, A 0 enables read pre-fetching. A value of 1 disables prefetching. If prefetching is disabled, the PCI9080 will disconnect after each memory read.
D9	Expansion ROM Space Prefetch Disable. A 0 enables read prefetching. A 1 disables prefetching If prefetching is disabled, the PCI9080 will disconnect after each memory read.
D10	Read Prefetch Count Enable. When set to a 1 and memory prefetching is enabled, PCI 9080 prefetches up to the number of Lwords specified in the prefetch count. When set to 0, PCI 9080 ignores the count and continues prefetching until terminated by the PCI bus.
D11..14	Prefetch Counter Number of Lwords to prefetch during memory read cycles (0-15). A count of zero selects a prefetch of 16 Lwords.

D15	Reserved
D16..17	Expansion ROM Space Local Bus Width. A value of 00 indicates a bus width of 8 bits A value of 01 indicates a bus width of 16 bits A value of 10 or 11 indicates a bus width of 32 bits.
D21..18	Expansion ROM Space Internal Wait States (data to data; 0-15 wait states).
D22	Expansion ROM Space Ready Input Enable A 1 value enables Ready input. A value of 0 disables the Ready input.
D23	Expansion ROM Space BTERM# Input Enable A 1 value enables BTERM# input. A value of 0 disables the BTERM# input. If this bit is set to 1, PCI 9080 bursts four Lword maximum at a time.
D24	Memory Space 0 Burst Enable A 1 value enables bursting. A value of 0 disables bursting. If burst is disabled, the local bus performs continuous single cycles for burst PCI read/write cycles.
D25	Extra Long serial EEPROM. A value of 1 loads the Subsystem ID and Local Address Space 1 registers. A value of 0 indicates not to load them.
D26	Expansion ROM Space Burst Enable A 1 value enables bursting. A value of 0 disables bursting. If burst is disabled, the local bus performs continuous single cycles for burst PCI read/write cycles.
D27	Direct Slave PCI write mode A 0 indicates that the PCI9080 should disconnect when the Direct Slave write FIFO is full. A 1 indicates that the PCI9080 should de-assert TRDY when the write FIFO is full.
D28..31	PCI Target Retry Delay Clocks. Contains the value (multiplied by 8) of the # of PCI bus clocks after receiving a PCI-Local read or write access and not successfully completing a transfer. Only pertains to Direct Slave writes when bit 27 is set to 1.

A.3.8 Local Range register for Direct Master to PCI (PCI Offset 0x1C)

D0..15	Reserved (64 KB increments)
D16..31	Specifies which local address bits to use for decoding a Local to PCI bus access. Each of the bits corresponds to an address bit. Bit 31 corresponds to Address bit 31. A value of 1 should be written to all bits that should be included in decode and a 0 to all others. This range is used for Direct Master memory, I/O, or configuration accesses.

A.3.9 Local Bus Base Address register for Direct Master to PCI Memory (PCI Offset 0x20)

D0..15	Reserved
D16..31	Assigns a value to the bits which will be used to decode a Local to PCI memory access.

A.3.10 Local Base Address for Direct Master to PCI IO/CFG Register (PCI Offset 0x24)

D0..15	Reserved
D16..31	Assigns a value to the bits to be used for decoding a Local to PCI I/O or configuration access. This base address is used for Direct Master I/O and configuration accesses.

A.3.11 PCI Base Address (Re-map) register for Direct Master to PCI Memory (PCI Offset 0x28)

D0	Direct Memory Access Enable A value of 1 enables decode of Direct Master Memory accesses. A value of 0 disables decode of Direct Master Memory accesses.
D1	Direct Master I/O Access Enable A value of 1enables decode of Direct Master I/O accesses. A value of 0 disables decode of Direct Master I/O accesses.
D2	LOCK# Input Enable A 1 value enables LOCK# input. A value of 0 disables the LOCK# input.
D3	Direct Master Red Prefetch Size control 00=PCI 9080 continues to prefetch read data from the PCI bus until Direct Master access is finished. This may result in an additional four unneeded Lwords being prefetched from the PCI bus. 01= Prefetch up to four Lwords from the PCI bus 10= Prefetch up to eight Lwords from the PCI bus 11= Prefetch up to 16 Lwords from the PCI bus. If PCI memory prefetch is not wanted, performs a Direct Master single cycle. The direct master burst reads must not exceed the programmed limit.
D4	Direct Master PCI read mode A value of 0 indicates that the PCI9080 should release the PCI bus when the read FIFO becomes full. A value of 1 indicates that the PCI9080 should keep the PCI bus and de-assert IRDY when the read FIFO becomes full.
D5..8,10	Programmable Almost Full flag. When the number of entries in the 32 word direct master write FIFO exceed this value, the output pin DMPAF# is asserted low
D9	Write and Invalidate Mode. When set to 1, PCI 9080 waits for 8 or 16 Lwords to be written from the local bus before starting PCI accesses. When set , all local Direct Master to PCI write accesses must be 8 or 16 Lwords bursts. Use in conjunction with (PCI 0x04)
D10	Direct Master Prefetch Limit. If set to 1, don't prefetch past 4K (4098 bytes) boundaries.
D13	I/O Remap Select. When set to1, forces PCI address bits [31:16] to all zeros. When set to 0, uses bits [31:16] of this register as PCI address bits [31:16].
D14..15	Direct Master Write Delay. This register is used to delay the PCI bus request after direct master burst write cycle has started. Values: 00=No delay; start the cycle immediately 01=Delay 4 PCI clocks 10=Delay 8 PCI clocks 11=Delay 16 PCI clocks

D16..31	Re-map of Local to PCI space into a PCI address space. The bits in this register re-map (replace) the Local address bits used in decode as the PCI address bits. This PCI Remap address is used for Direct Master memory and I/O accesses.

A.3.12 PCI Configuration Address Register for Direct Master to PCI IO/CFG (PCI Offset 0x2C)

D0..1	Configuration Type (00=Type 0, 01=Type 1)
D2..7	Register Number. If different register read/write is needed, this register value must be programmed and a new PCI configuration cycle must be generated.
D8..10	Function Number
D11..15	Device Number
D16..23	Bus Number
D24..30	Reserved
D31	Configuration Enable. A value of 1 allows Local to PCI I/O accesses to be converted to a PCI configuration cycle. The parameters in this table are used to generate the PCI configuration address.

A.3.13 Local Address Space 1 Range Register for PCI to Local Bus (PCI Offset 0xF0)

D0	Memory Space Indicator A value of 0 indicates Local Address Space 1 maps into PCI memory space. A value of 1 indicates Local Address Space 1 maps into PCI I/O space.
D1..2	Encoded for Memory Space If mapped into memory space, encoding is as follows: 00-Locate anywhere in 32 bit PCI address space 01-Locate below 1 MB in PCI address space 10-Locate anywhere in 64 bit PCI address space 11-Reserved If mapped into I/O space, bit 1 must be set to 0. Bit 2 is included with bits [31:3] to indicate decoding range.
D3	If mapped into memory space, a value of 1 indicates reads are prefetchable (bit has no effect on the operation of the PCI 9080, but is for system status). If mapped into I/O space, bit is included with bits [31:2] to indicate decoding range.
D4..31	Specifies which PCI address bits to use for decoding a PCI access to local bus space 1. Each of the bits corresponds to a PCI address bit. Bit 31 corresponds to Address bit 31. Write a value of 1 to all bits that must be included in decode and a 0 to all others (Used in conjunction with PCI Configuration Register). Default is 1 MB.

A.3.14 Local Address Space 1 Local Base Address (Remap) Register (PCI Offset 0xF4)

D0	Space 1 Enable A value of 1 enables decoding of PCI addresses for Direct Slave access to local space 1. A value of 0 disables decoding. If this bit is set to 0, the PCI BIOS may not allocate (assign) the base address for Space 1. **Note:** *Must be set to 1 for any Direct Slave access to Space 1.*
D1	Reserved
D2..3	If local space 1 is mapped into memory space, bits are not used. If mapped I/O space, bit is included with bits [31:4] for remapping.
D4..31	Remap of PCI Address to Local Address space 1 into a Local Address Space. The bits in this register remap (replace) the PCI Address bits used in decode as the Local Address bits.

A.3.15 Local Address Space 1 Bus Region Descriptor Register (PCI Offset 0xF8)

D0..1	Memory Space 1 Local Bus Width A value of 00 indicates bus width of 8 bits A value of 01 indicates bus width of 16 bits A value of 10 indicates bus width of 32 bits
D2..5	Memory space 1 Internal Wait States (data to data; 0-15 wait states).
D6	Memory space 1 Ready Input Enable A value of 1 enables BTERM# input. A value of 0 disables Ready input.
D7	Memory space 1 BTERM# Input Enable A value of 1 enables BTERM# input. A value of 0 disables BTERM# input. If this bit is set to 0, PCI 9080 bursts four Lword maximum at a time.
D8	Memory space 1 Burst Enable A value of 1 enables bursting. A value of 0 disables bursting. If burst is disabled, the local bus performs continuous single cycle for burst PCI read/write cycles.
D9	Memory space 1 Prefetch Disable If mapped into memory space A value of 0 enables read prefetching. A value of 1 disables read prefetching. If prefetching is disabled, PCI 9080 disconnects after each memory read.
D10	Read Prefetch Count Enable When set to 1 and memory prefetching is enabled, PCI 9080 prefetches up to the number of Lwords specified in the prefetch count. When set to 0, PCI 9080 ignores the count and continues prefetching until terminated by the PCI bus.
D11..14	Prefetch Counter Number of Lwords to prefetch during memory read cycles (0-15).
D15..31	Reserved

A.4 RUNTIME REGISTERS

Table D RUN TIME REGISTERS

PCI Offset from base Addr	Size	Access R	Access W	Register Name	Value after reset
0x40	D32	yes	yes	Mailbox Register 0	0x00000000
0x44	D32	yes	yes	Mailbox Register 1	0x00000000
0x48	D32	yes	yes	Mailbox Register 2	0x00000000
0x4C	D32	yes	yes	Mailbox Register 3	0x00000000
0x50	D32	yes	yes	Mailbox Register 4	0x00000000
0x54	D32	yes	yes	Mailbox Register 5	0x00000000
0x58	D32	yes	yes	Mailbox Register 6	0x00000000
0x5C	D32	yes	yes	Mailbox Register 7	0x00000000
0x60	D32	yes	yes	PCI to Local Doorbell Register	0x00000000
0x64	D32	yes	yes	Local to PCI Doorbell Register	0x00000000
0x68	D32	yes	Bit Dep.	Interrupt Control/Status	0x00000000
0x6C	D32	yes	Bit Dep.	EEPROM Control, PCI Command Codes, User I/O Control, Init Control	0x001767E
0x70	D32	yes	Bit Dep.	Device ID/Vendor ID	0x00
0x74	D32	yes	Bit Dep.	Unused/Revision ID	0x00
0x70	D32	yes	Bit Dep.	Mailbox Register 0 (1)	0x00
0x74	D32	yes	Bit Dep.	Mailbox Register 1 (1)	0x00

A.4.1 Mailbox Register 0 (PCI Offset 0x40)

D0..31	32 bit mailbox register

A.4.2 Mailbox Register 1 (PCI Offset 0x44)

D0..31	32 bit mailbox register

A.4.3 Mailbox Register 2 (PCI Offset 0x48)

D0..31	32 bit mailbox register

A.4.4 Mailbox Register 3 (PCI Offset 0x4C)

D0..31	32 bit mailbox register

A.4.5 Mailbox Register 4 (PCI Offset 0x50)

D0..31	32 bit mailbox register

A.4.6 Mailbox Register 5 (PCI Offset 0x54)

D0..31	32 bit mailbox register

A.4.7 Mailbox Register 6 (PCI Offset 0x58)

D0..31	32 bit mailbox register

A.4.8 Mailbox Register 7 (PCI Offset 0x5C)

D0..31	32 bit mailbox register

A.4.9 PCI to Local Doorbell Register Description (PCI Offset 0x60)

D0..31	Doorbell register. A PCI master can write to this register and it will generate a local interrupt to the local processor. The local processor can then read this register to determine which doorbell bit was asserted. The PCI master sets a doorbell by writing a 1 to a particular bit. The local processor can clear a doorbell bit by writing a 1 to that bit position.

A.4.10 Local to PCI Doorbell Register Description (PCI Offset 0x64)

D0..31	Doorbell register. The local processor can write to this register and it will generate a PCI interrupt. A PCI master can then read this register to determine which doorbell bit was asserted. The local processor sets a doorbell by writing a 1 to a particular bit. The PCI master can clear a doorbell bit by writing a 1 to that bit position.

A.4.11 Interrupt Control /Status (PCI Offset 0x68)

D0	Enable Local Bus LSERR#. A value of 1 enables PCI 9080 to assert LSERR# interrupt output when PCI bus Target Abort or Master Abort status bit is set in the PCI Status configuration register.
D1	Enable Local Bus SERR# when PCI parity error occurs during PCI 9080 Master Transfer or PCI 9080 Slave access or Outbound Free List FIFO Overflow Init.
D2	Generate PCI Bus SERR#. When this bit is set to 0, writing a 1 generates a PCI bus SERR#.
D3	Mailbox Interrupt Enable. A value of 1 enables a Local Interrupt to be generated when the PCI bus writes to Mailbox register 0-3. To clear the Local Interrupt, the Local master must read the Mailbox. Used in conjunction with Local interrupt enable.
D4..7	Reserved
D8	PCI Interrupt Enable. A value of 1 enables PCI interrupts.
D9	PCI Doorbell Interrupt Enable. A value of 1 enables doorbell interrupts. Used in conjunction with PCI interrupt enable. Clearing the doorbell interrupt bits that caused the interrupt also clears the interrupt.
D10	PCI Abort Interrupt Enable A value of 1 enables a master abort or master detect of a target abort to generate a PCI interrupt. Used in conjunction with PCI interrupt enable. Clearing the abort status bits also clears the PCI interrupt.
D11	PCI Local Interrupt Enable. A value of 1 enables a local interrupt input to generate a PCI interrupt. Use in conjunction with PCI Interrupt enable. Clearing the local bus cause of the interrupt also clears the interrupt.
D12	Retry Abort Enable. A value of 1 enables PCI 9080 to treat 256 Master consecutive retries to a Target as a target abort. A value of 0 enables PCI 9080 to attempt Master Retries indefinitely. **Note:** *for diagnostic purposes only*.
D13	Value of 1 indicates PCI doorbell interrupt is active.
D14	Value of 1 indicates PCI abort interrupt is active.
D15	Value of 1 indicates local interrupt is active (LINTi#).
D16	Local Interrupt Output Enable. A value of 1 enables local interrupt output.
D17	Local Doorbell Interrupt Enable. A value of 1 enables doorbell interrupts. Used in conjunction with Local interrupt enable. Clearing the local doorbell interrupt bits that caused the interrupt also clears the interrupt.
D18	Local DMA Channel 0 Interrupt Enable. A value of 1 enables DMA Channel 0 interrupts. Used in conjunction with Local interrupt enable. Clearing the DMA status bits also clears the interrupt.
D19	Local DMA Channel 1 Interrupt Enable. A value of 1 enables DMA Channel 1 interrupts. Used in conjunction with Local interrupt enable. Clearing the DMA status bits also clears the interrupt.
D20	Value of 1 indicates local doorbell interrupt is active.
D21	Value of 1 indicates DMA Ch 0 interrupt is active.
D22	Value of 1 indicates DMA Ch 1 interrupt is active.

D23	Value of 1 indicates BIST interrupt is active. BIST (Built-In Self Test) interrupt is generated by writing 1 to bit 6 of the PCI Configuration BIST register. Clearing bit 6 clears the interrupt. Refer to the BIST Register for a description of self test.
D24	Value of 0 indicates a Direct master was the bus master during a master or Target abort. (Not valid until abort occurs.)
D25	Value of 0 indicates a DMA CH 0 was the bus master during a master or Target abort. (Not valid until abort occurs.)
D26	Value of 0 indicates a DMA CH 1 was the bus master during a master or Target abort. (Not valid until abort occurs.)
D27	Value of 0 indicates a Target Abort was generated by the PCI 9080 after 256 consecutive Master retries to a Target. (Not valid until abort occurs.)
D28	**V**alue of 1 indicates PCI wrote data to the Mailbox #0. Enabled only if MBOXINTENB is enabled (bit 3 high).
D29	Value of 1 indicates PCI wrote data to the Mailbox #1. Enabled only if MBOXINTENB is enabled (bit 3 high).
D30	Value of 1 indicates PCI wrote data to the Mailbox #2. Enabled only if MBOXINTENB is enabled (bit 3 high).
D31	Value of 1 indicates PCI wrote data to the Mailbox #3. Enabled only if MBOXINTENB is enabled (bit 3 high).

A.4.12 Serial EEPROM Control, PCI Command Codes, User I/O Control, Init Control Register (PCI Offset 0x6C)

D0..3	PCI Read Command Code for DMA This PCI command is sent out during DMA read cycles.
D4..7	PCI Write Command Code for DMA This PCI command is sent out during DMA write cycles.
D8..11	PCI Memory Read Command Code for Direct Master This PCI command is sent out during Direct Master read cycles.
D12..15	PCI Memory Write Command Code for Direct Master This PCI command is sent out during Direct Master write cycles.
D16	General Purpose Output A value of 1 will cause the USERO output to go high. A value of 0 will cause the output to go low.
D17	General Purpose Input A value of 1 indicates that USERI input pin is high. A value of 0 indicates that USERI pin is low.
D18..23	Reserved
D24	Serial EEPROM clock for Local or PCI bus reads or writes to serial EEPROM. Toggling this bit generates a serial EEPROM clock. (Refer to the manufacturer's data sheet for the particular EEPROM being used.)
D25	Serial EEPROM chip select. For Local or PCI bus reads or writes to serial EEPROM Setting this bit to a 1 provides the EEPROM chip select.
D26	Write bit to serial EEPROM. For writes, this output bit is the input to the serial EEPROM. It is clocked into the serial EEPROM by the serial EEPROM clock.
D27	Read serial EEPROM data bit. For reads, this input bit is the output of the serial EEPROM. It is clocked out of the serial EEPROM by the serial EEPROM clock.
D28	Serial EEPROM present A 1 in this bit indicates that an EEPROM is present.
D29	Reload Configuration Registers When this bit is 0, writing a 1 causes the PCI9080 to reload the local configuration registers from the serial EEPROM.
D30	PCI Adapter Software Reset A value of 1 written to this bit will hold the local bus logic in the PCI9080 reset and LRESETO# asserted. The contents of the PCI configuration registers and Shared Run Time registers will not be reset. Software Reset can only be cleared from the PCI bus. (Local bus remains reset until this bit is cleared.)
D31	Local Init Status. Value of 1indicates local init done. Responses to PCI accesses will be RETRYs until this bit is set. While Input NB# is asserted low this bit will be forced to 1.

A.4.13 PCI Permanent Configuration ID Register (PCI Offset 0x70)

D0..15	Permanent Vendor ID. Identifies device manufacturer. **Note:** *Hardcoded to the PCI SIG issued vendor ID of PLX (10B5).*
D16..31	Permanent Device ID. Identifies the particular device. **Note:** *Hardcoded to the PLX part number for PCI interface chip PCI 9080.*

A.4.14 PCI Permanent Revision ID Register (PCI Offset 0x74)

D0..15	Permanent Revision ID. **Note**: *Hardcoded to the silicon revision of the PCI 9080.*

A.5 LOCAL DMA REGISTERS

Table E DMA REGISTERS

PCI Offset from base Addr	Size	Access R	Access W	Register Name	Value after Reset
0x80	D32	yes	yes	DMA Ch 0 Mode	0x00000003 (Cx and Jx modes)
0x84	D32	yes	yes	DMA Ch 0 PCI Address	0x00000000
0x88	D32	yes	yes	DMA Ch 0 Local Address	0x00000000
0x8C	D32	yes	yes	DMA Ch 0 Transfer Byte Count	0x00000000
0x90	D32	yes	yes	DMA Ch 0 Descriptor Pointer	0x00000000
0x94	D32	yes	yes	DMA Ch 1 Mode	0x00000003 (Cx and Jx modes)
0x98	D32	yes	yes	DMA Ch 1 PCI Address	0x00000000
0x9C	D32	yes	yes	DMA Ch 1 Local Address	0x00000000
0xA0	D32	yes	yes	DMA Ch 1 Transfer Byte Count	0x00000000
0xA4	D32	yes	yes	DMA Ch 1 Descriptor Pointer	0x00000000
0xA8	D32		Bit Dep.	Reserved/DMA Ch 1Command/Status Register/DMA Ch 0 Command/Status	0x00000010
0xAC	D32	yes	Bit Dep.	Mode/ Arbitration Register	0x00000000
0xB0	D32	yes	Bit Dep.	DMA Threshold Register	0x00000000

A.5.1 DMA Channel 0 Mode Register (PCI Offset 0x80)

D0..1	Local DMA Bus Width A value of 00 indicates a DMA bus width of 8 bits. A value of 01 indicates DMA bus width of 16 bits. A value of 10 or 11 indicates a DMA bus width of 32 bits.
D2..5	Internal Wait States (data to data)
D6	Ready Input Enable A value of 1 enables Ready input. A value of 0 disables the Ready input.
D7	Bterm Input Enable A value of 1 enables Bterm input. A value of 0 disables Bterm input.
D8	Local Burst Enable A value of 1 enables bursting. A value of 0 disables bursting.

D9	Chaining A 1 value indicates non-chaining mode is enabled. For chaining mode, the DMA source address, destination address and byte count are loaded from memory in PCI or Local Address Spaces. A 0 value indicates non-chaining mode is enabled.
D10	Done Interrupt Enable A 1 value enables interrupt when done. A 0 value disables interrupt when done. If DMA clear count mode is enabled, the interrupt won't occur until the byte count is cleared.
D11	Local Addressing Mode A 1 value indicates local addresses LA [31:2] to be held constant. A 0 value indicates local address is incremented.
D12	Demand Mode A value of 1 causes the DMA controller to operate in demand mode. In demand mode the DMA controller transfers data when it's DREQ# input is asserted. It asserts DACK# to indicate that the current local bus transfer is in response to the DREQ# input. The DMA controller transfers Lwords (32bits) of data. This may result in multiple transfers for an 8 or 16 bit bus.
D13	Write and Invalidate mode for DMA transfers. When set to 1, PCI 9080 performs Write and Invalidate cycles to the PCI bus. PCI 9080 supports Write and Invalidate sizes of 8 or 16 Lwords. The size is specified in the PCI Cache Line Size Register. If a size other than 8 or 16 is specified, PCI 9080 performs write transfers rather than Write and Invalidate transfers. Transfers must start and end at the Cache Line Boundaries.
D14	DMA EOT (End Of Transfer) Enable. A Value of 1 enables EOT# input pin. A Value of 0 disables EOT# input pin.
D15	DMA Stop Data Transfer Mode A Value of 0 sends a BLAST to terminate DMA transfer. A Value of 1 indicates an EOT asserted or DREQ# negated during demand mode DMA terminates the DMA transfer.
D16	DMA Clear Count Mode. When set to 1, the byte count in each chaining descriptor, if it is in local memory, is cleared when the corresponding DMA transfer is complete. **Note:** *If chaining descriptor is in PCI memory, the count is not cleared.*
D17	DMA Channel 0 Interrupt Select A Value of 1 routes the DMA Channel 0 interrupt to the PCI interrupt. A Value of 0 routes the DMA Channel 0 interrupt to the local bus interrupt.
D18..31	Reserved

A.5.2 DMA Channel 0 PCI Address Register (PCI Offset 0x84)

D0..31	PCI Address Register. This indicates where in the PCI memory space the DMA transfers (reads or writes) will start from.

A.5.3 DMA Channel 0 Local Address Register (PCI Offset 0x88)

D0..31	Local Address Register. This indicates where in the local memory space the DMA transfers (reads or writes) will start from.

A.5.4 DMA Channel 0 Transfer Size (Bytes) Register (PCI Offset 0x8C)

D0..22	DMA Transfer Size (Bytes). Indicates number of bytes to be transferred during DMA operation.
D23..31	Reserved

A.5.5 DMA Channel 0 Descriptor Pointer Register (PCI Offset 0x90)

D0	Descriptor Location. A Value of 1 indicates PCI address space. A Value of 0 indicates Local Address Space.
D1	End of Chain A 1 value indicates end of chain. A 0 value indicates not end of chain descriptor. (Same as Nonchaining Mode)
D2	Interrupt after Terminal Count A 1 value causes an interrupt to be generated after the terminal count for this descriptor is reached. A 0 value disables interrupts from being generated.
D3	Direction of transfer A 1 value indicates transfers from local bus to PCI bus. A 0 value indicates transfers from PCI to local bus.
D4..31	Next Descriptor Address. Quad word aligned (Bit[3:0] = 0000).

A.5.6 DMA Channel 1 Mode Register (PCI Offset 0x94)

D0..1	Local Bus Width A value of 00 indicates a bus width of 8 bits. A value of 01 indicates bus width of 16 bits. A value of 10 or 11 indicates a DMA bus width of 32 bits. The bus width is forced to 16 bits for the Sx mode
D2..5	Internal Wait States (data to data).
D6	Ready Input Enable A 1 value enables Ready input. A 0 value disables the Ready input.
D7	Bterm Input Enable A 1 value enables Bterm input. A value of 0 disables the Bterm input. If this bit is set to 0, PCI 9080 bursts four Lword maximum at a time.
D8	Local Burst Enable A 1 value enables Local bursting input. A value of 0 disables Local bursting. If burst is disabled, the local bus performs continuous single cycles for burst PCI read/write cycles.
D9	Chaining A 1 value indicates chaining mode enabled. For chaining mode, the DMA source address, destination address and byte count are loaded from memory in PCI or Local address spaces. A 0 value indicates non-chaining mode.
D10	Done Interrupt Enable A 1 value enables interrupt when done. A 0 value disables the interrupt when done. If DMA Clear Count Mode is enabled, the interrupt won't occur until the byte count is cleared.
D11	Local Addressing Mode A 1 value indicates local addresses LA[31:2] to be held constant. A 0 value indicates local addresses is incremented.

D12	Demand Mode A value of 1 causes the DMA controller to operate in demand mode. In demand mode the DMA controller transfers data when its DREQ# input is asserted. It asserts DACK# to indicate that the current local bus transfer is in response to the DREQ# input. The DMA controller transfers Lwords (32bits) of data. This may result in multiple transfers for an 8 or 16 bit bus.
D13..31	Reserved

A.5.7 DMA Channel 1 PCI Data Address Register (PCI Offset 0x98)

D0..31	PCI Data Address Register. This indicates where in the PCI memory space the DMA transfers (reads or writes) will start from.

A.5.8 DMA Channel 1 Local Data Address Register (PCI Offset 0x9C)

D0..31	Local data Address Register. This indicates where in the local memory space the DMA transfers (reads or writes) will start from.

A.5.9 DMA Channel 1 Transfer Size (bytes) register (PCI Offset 0xA0)

D0..22	DMA Transfer Size (Bytes). Indicates number of bytes to be transferred during DMA operation.
D23..31	Reserved

A.5.10 DMA Channel 1 Descriptor Pointer Register (PCI Offset 0xA4)

D0	Descriptor Location. A 1 value indicates PCI address space. A 0 value indicates Local address space.
D1	End of Chain A 1 value indicates end of chain. A 0 value indicates not end of chain descriptor. (Same as Nonchaining Mode)
D2	Interrupt after Terminal Count A 1 value causes an interrupt to be generated after the terminal count for this descriptor is reached. A 0 value disables interrupts from being generated.
D3	Direction of transfer A 1 value indicates transfers from local bus to PCI bus. A 0 value indicates transfers from PCI bus to local bus.
D4..31	Next Descriptor Address. Quad word aligned (bits [3:0] =0000).

A.5.11 DMA Command/Status Register (PCI Offset 0xA8)

D0	Channel 0 Enable A 1 value enables the channel to transfer data. A 0 value disables the channel from starting a DMA transfer and if in the process of transferring data suspend transfer (Pause).
D1	Channel 0 Start Writing a 1 to this bit causes the channel to start transferring data if the channel is enabled.
D2	Channel 0 Abort Writing a 1 to this bit causes the channel to abort the current transfer. The channel enable bit must be cleared. The channel complete bit is set when the abort has completed.
D3	Clear Interrupt. Writing a 1 to this bit clears channel 0 interrupts.
D4	Channel 0 Done A 1 value indicates this channels transfer is complete. A 0 value indicates the channel transfer is not complete.
D5..7	User Defined

A.5.12 DMA Channel 1 Command/Status Register 0 (PCI Offset 0xA8)

D0	Channel 1 Enable A 1 value enables the channel to transfer data. A 0 value disables the channel from starting a DMA transfer and if in the process of transferring data suspend transfer (Pause).
D1	Channel 1 Start Writing a 1 to this bit causes the channel to start transferring data if the channel is enabled.
D2	Channel 1 Abort Writing a 1 to this bit causes the channel to abort the current transfer. The channel enable bit must be cleared. The channel complete bit is set when the abort has completed.
D3	Clear Interrupt. Writing a 1 to this bit clears channel 1 interrupts.
D4	Channel 1 Done A 1 value indicates this channel's transfer is complete. A 0 value indicates the channel transfer is not complete.
D5..7	Reserved

A.5.13 DMA Arbitration Register 1 (PCI Offset 0xAC)

D0..7	Local bus Latency Timer Number of local bus clock cycles before negating HOLD and releasing the local bus. This timer is also used with bit 27 to delay BREQ input to give up the local bus only when this timer expires.
D8..15	Local bus Pause Timer Number of local bus clock cycles before reasserting HOLD after releasing the local bus. **Note:** *Applicable only to DMA operation.*
D16	Local bus Latency Timer Enable A value of 1 enables latency timer.
D17	Local bus Pause Timer Enable A value of 1 enables pause timer.
D18	Local bus BREQ Enable A value of 1 enables local bus BREQ input. When the BREQ input is active, PCI 9080 negates HOLD and releases the local bus.
D19..20	DMA Channel Priority A value of 00 indicates a rotational priority scheme. A value of 01 indicates Channel 0 has priority. A value of 10 indicates Channel 1 has priority. A value of 11 is reserved.
D21	Local bus direct slave give up bus mode When set to 1, PCI 9080 negates HOLD and releases the local bus when the Direct Slave write FIFO becomes empty during a Direct Slave write or when the Direct Slave read FIFO becomes full during a Direct Slave read.
D22	Direct slave LLOCKo# Enable A value of 1 enables PCI Direct Slave locked sequences. A value of 0 disables Direct Slave locked sequences.
D23	PCI Request Mode A value of 1 causes PCI9080 to negate REQ when it asserts FRAME during a master cycle. A value of 0 causes PCI 9080 to leave REQ asserted for the entire bus master cycle.
D24	PCI Rev 2.1 Mode When set to 1, PCI 9080 operates in Delayed Transaction mode for Direct Slave Reads. PCI 9080 issues a RETRY and prefetches the read data.
D25	PCI Read No Write Mode A value of 1 forces a retry on writes if read is pending. A value of 0 allows writes to occur while read is pending.
D26	PCI Read with Write Flush Mode A value of 1 submits a request to flush a pending read cycle if a write cycle is detected. A value of 0 submits a request to not effect pending reads when a write cycle occurs. (PCI v2.1 compatible)
D27	Gate the Local Bus Latency Timer with BREQ If this bit is set to 0, PCI 9080 gives up the local bus during Direct Slave or DMA transfer after the current cycle (if enabled and BREQ is sampled). If this bit is set to 1, PCI 9080 gives up the local bus only (if BREQ is sampled) and the Local Bus Latency Timer is enabled and expires during Direct Slave or DMA transfer.
D28	PCI Read No Flush Mode A value of 1 submits request to not flush the read FIFO if PCI read cycle completes (Read Ahead mode). A value of 0 submits request to flush read FIFO if PCI read cycle completes.

D29	Reads Device or Vendor ID If set to 0, reads from the PCI Configuration Register address 0x00 and returns the Device ID and Vendor ID. If set to 1, reads from the PCI Configuration Register address 0x00 and returns the Subsystem and Subsystem Vendor ID.
D30..31	Reserved

A.5.14 DMA Threshold Register 1 (PCI Offset 0xB0)

D0..3	DMA Channel 0 PCI to Local Almost Full (C0PLAF): # of Full Entries (minus 1) in FIFO before Requesting Local Bus for Writes. (C0PLAF+1) + (C0PLAE+1) should be <= FIFO Depth of 16
D4..7	DMA Channel 0 Local to PCI Almost Empty (C0LPAE): # of Empty Entries (minus 1) in FIFO before Requesting Local Bus for Reads. (C0LPAF+1) + (C0LPAE+1) should be <= FIFO Depth of 16
D8..11	DMA Channel 0 Local to PCI Almost Full (C0LPAF): # of Full Entries (minus 1) in FIFO before requesting PCI bus for Writes.
D12..15	DMA Channel 0 PCI to Local Almost Empty (C0PLAE): # of Empty Entries (minus 1) in FIFO before Requesting PCI Bus for Reads.
D16..19	DMA Channel 1 PCI to Local Almost Full (C0PLAF): # of Full Entries (minus 1) in FIFO before Requesting Local Bus for Writes. (C0PLAF+1) + (C0PLAE+1) should be <= FIFO Depth of 16
D20..23	DMA Channel 1 Local to PCI Almost Empty (C0LPAE): # of Empty Entries (minus 1) in FIFO before Requesting Local Bus for Reads. (C0LPAF+1) + (C0LPAE+1) should be <= FIFO Depth of 16
D24..27	DMA Channel 1 Local to PCI Almost Full (C0LPAF): # of Full Entries (minus 1) in FIFO before requesting PCI bus for Writes.
D28..31	DMA Channel 1 PCI to Local Almost Empty (C0PLAE): # of Empty Entries (minus 1) in FIFO before Requesting PCI Bus for Reads.

A.6 MESSAGING QUEUE REGISTERS

Table F MESSAGING QUEUE REGISTERS

PCI Offset from base Addr	Size	Access R	Access W	Register Name	Value after reset
0x30	D32	yes	yes	Outbound Post Queue Interrupt Status	0x00000000
0x34	D32	yes	yes	Outbound Post Queue Interrupt Mask	0x00000000
0x40	D32	yes	yes	Inbound Queue Port	0x00000000
0x44	D32	yes	yes	Outbound Queue Port	0x00000000
0xC0	D32	yes	yes	Messaging Unit Configuration	0x00000000
0xC4	D32	yes	yes	Queue Base Address	0x00000000
0xC8	D32	yes	yes	Inbound Free Head Pointer	0x00000000
0xCC	D32	yes	yes	Inbound Free Trail Pointer	0x00000000
0xD0	D32	yes	yes	Inbound Post Head Pointer	0x00000000
0xD4	D32	yes	yes	Inbound Post Tail Pointer	0x00000000
0xD8	D32	yes	yes	Outbound Free Head Pointer	0x00000000
0xDC	D32	yes	yes	Outbound Free Tail Pointer	0x00000000
0xE0	D32	yes	yes	Outbound Post Head Pointer	0x00000000
0xE4	D32	yes	yes	Outbound Post Tail Pointer	0x00000000
0xE8	D32	yes	yes	Queue Status/Control Register	0x00000000

A.6.1 Outbound Post List FIFO Interrupt Status Register (PCI Offset 0x30)

D0..2	Reserved
D3	Outbound Post List FIFO Interrupt. This bit is set when the Outbound Post List FIFO is not empty. This bit is not affected by the interrupt mask bit.
D4..31	Reserved

A.6.2 Outbound Post List FIFO Interrupt Status Register (PCI Offset 0x34)

D0..2	Reserved
D3	Outbound Post List FIFO Interrupt Mask. Interrupt is masked when this bit is set.
D4..31	Reserved

A.6.3 Inbound Queue Port Register (PCI Offset 0x40)

D0..31	Value written by PCI master is stored into the Inbound Post List FIFO, which is located in local memory at the address pointed to by the Queue Base Address + FIFO Size + Inbound Post Head Pointer. From the time of the PCI write until the local memory write and update of the Inbound Post Queue Head Pointer, further accesses to this register result in a retry. A local interrupt is generated when the Inbound Post List FIFO is not empty. When the port is read by the PCI master, the value is read from the Inbound Free List FIFO, which is located in local memory at the address pointed the by The Queue Base Address + Inbound Free Tail Pointer. If FIFO is empty, a value of FFFFFFFh is returned..

A.6.4 Outbound Queue Port Register (PCI Offset 0x44)

D0..31	Value written by PCI master is stored into the Outbound Free List FIFO, which is located in local memory at the address pointed to by the Queue Base Address + (3*FIFO Size) + Outbound Free Head Pointer. From the time of the PCI write until the local memory write and update of the Outbound Free Head Pointer, further accesses to this register result in a retry. If FIFO fills up, a local LSERR interrupt is generated. When the port Is read by the PCI master, the value is read from the Outbound Post List FIFO, which is located in local memory at the address pointed to by the Queue Base address + (2*FIFO Size) + Outbound Trail Pointer. If FIFO is empty, a value of FFFFFFFh is returned. A PCI interrupt is generated if Outbound Post List FIFO is not empty.

A.6.5 Messaging Queue Configuration Register (PCI Offset 0xC0)

D0	Queue Enable Value of 1 allows accesses to the Inbound and Outbound Queue ports. If cleared to 0, writes are accepted but ignored and reads return FFFFFFFF. All pointer initialization and frame allocation should be completed before enabling this bit.
D1..5	Circular FIFO Size Defines the size of one of the circular FIFOs. Each of the four FIFOs are the same size. Each FIFO entry is one 32 bit word.

FIFO Size Encoding

5:1	Max entries per FIFO	FIFO Size	Total FIFO Memory
00001	4K entries	16 KB	64 KB
00010	8K entries	32 KB	128 KB
00100	16K entries	64 KB	256 KB
01000	32K entries	128 KB	512 KB
10000	64K entries	256 KB	1 MB

D6..31	Reserved

A.6.6 Queue Base Address Register (PCI Offset 0xC4)

D0..19	Reserved
D20..31	Queue Base Address Local Memory base address of the Inbound and Outbound Queues (four contiguous and equal size FIFOs). Queue base address must be aligned on a 1 MB boundary.

A.6.7 Inbound Free Head Pointer Register (PCI Offset 0xC8)

D0..1	Reserved

D2..19	Inbound Free Head Pointer Local memory Offset for Inbound Free List FIFO. This register is initialized as (0*FIFO Size) and maintained by the local CPU software.
D20..31	Queue Base Address

A.6.8 Inbound Free Head Tail Register (PCI Offset 0xCC)

D0..1	Reserved
D2..19	Inbound Free Tail Pointer Local Memory Offset for Inbound Free List FIFO. This register is initialized as (0*FIFO Size) by the local CPU software. It is maintained by the MU hardware and is incremented modulo the FIFO size.
D20..31	Queue Base Address

A.6.9 Inbound Post Head Pointer Register (PCI Offset 0xD0)

D0..1	Reserved
D2..19	Inbound Post Head Pointer Local Memory Offset for Inbound Post List FIFO. This register is initialized as (1*FIFO Size) by the local CPU software. It is maintained by the MU hardware and is incremented modulo the FIFO size.
D20..31	Queue Base Address

A.6.10 Inbound Post Tail Pointer Register (PCI Offset 0xD4)

D0..1	Reserved
D2..19	Inbound Post Tail Pointer Local Memory Offset for Inbound Post List FIFO. This register is initialized as (1*FIFO Size) by the local CPU software.
D20..31	Queue Base Address

A.6.11 Outbound Free Head Pointer Register (PCI Offset 0xD8)

D0..1	Reserved
D2..19	Outbound Free Head Pointer Local Memory Offset for Outbound Free List FIFO. This register is initialized as (3*FIFO Size) by the local CPU software. It is maintained by the MU hardware and is incremented modulo the FIFO size.
D20..31	Queue Base Address

A.6.12 Outbound Free Tail Pointer Register (PCI Offset 0xDC)

D0..1	Reserved
D2..19	Outbound Free Tail Pointer Local Memory Offset for Outbound Free List FIFO. This register is initialized as (3*FIFO Size) by the local CPU software.
D20..31	Queue Base Address

A.6.13 Outbound Post Head Pointer Register (PCI Offset 0xE0)

D0..1	Reserved
D2..19	Outbound Post Head Pointer Local Memory Offset for Outbound Post List FIFO. This register is initialized as (2*FIFO Size) by the local CPU software.
D20..31	Queue Base Address

A.6.14 Outbound Post Tail Pointer Register (PCI Offset 0xE4)

D0..1	Reserved
D2..19	Outbound Post Tail Pointer Local Memory Offset for Outbound Post List FIFO. This register is initialized as (2*FIFO Size) and maintained by the MU hardware and is incremented modulo the FIFO size.
D20..31	Queue Base Address

A.6.15 Queue Status/Control Register (PCI Offset 0xE8)

D0	I(2)O Decode Enable When this bit is set, Mailbox registers 0 and 1 are replaced by the Inbound and Outbound Queue Port Registers and redefines Space 1 as PCI Base Address 0 to be accessed by PCIBAR0. Former Space 1 registers F0, F4, and F8 should be programmed to configure their shared I(2)O memory space, defined as PCI Base Address 0.
D1	Queue Local Space Select When this bit is set to 0, use Local Address Space 0 bus region descriptor for queue accesses. When this bit is set to 1, use Local Address Space 1 bus region descriptor for queue accesses.
D2	Outbound Post List FIFO Prefetch Enable When this bit is set, prefetching occurs from the Outbound Post List FIFO if not empty.
D3	Inbound Free List FIFO Prefetch Enable When this bit is set, prefetching occurs from the Inbound Free List FIFO if not empty.
D4	Inbound Post List FIFO Interrupt Mask When this bit is set, interrupt is masked.
D5	Inbound Post List FIFO Interrupt This bit is set when the Inbound Post List FIFO if not empty. This bit is not affected by the Interrupt Mask bit.
D6	Outbound Free List FIFO Overflow Interrupt Mask When this bit is set, interrupt is masked
D7	Outbound Free List FIFO Overflow Interrupt This bit is set when the Outbound Free List FIFO becomes full. A local SERR (NMI) interrupt is generated if enabled in the Interrupt Control/Status Register. Writing 1 clears the interrupt.
D8..31	Unused