User Manual

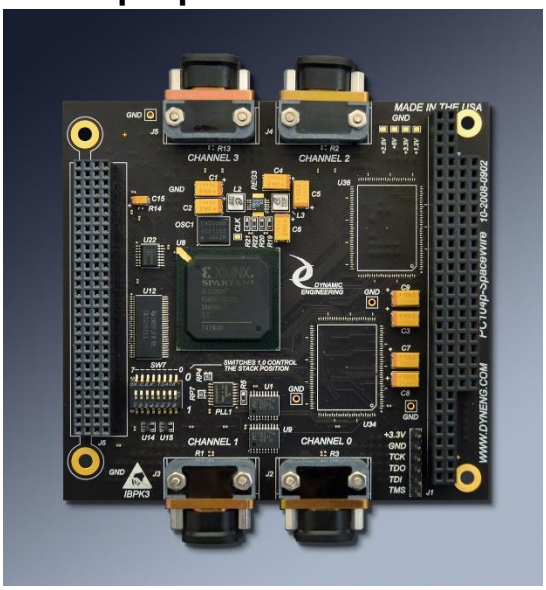
Hardware

SpaceWire Monitor

Manual Revision 2p0
Revision Date 09/13/2024

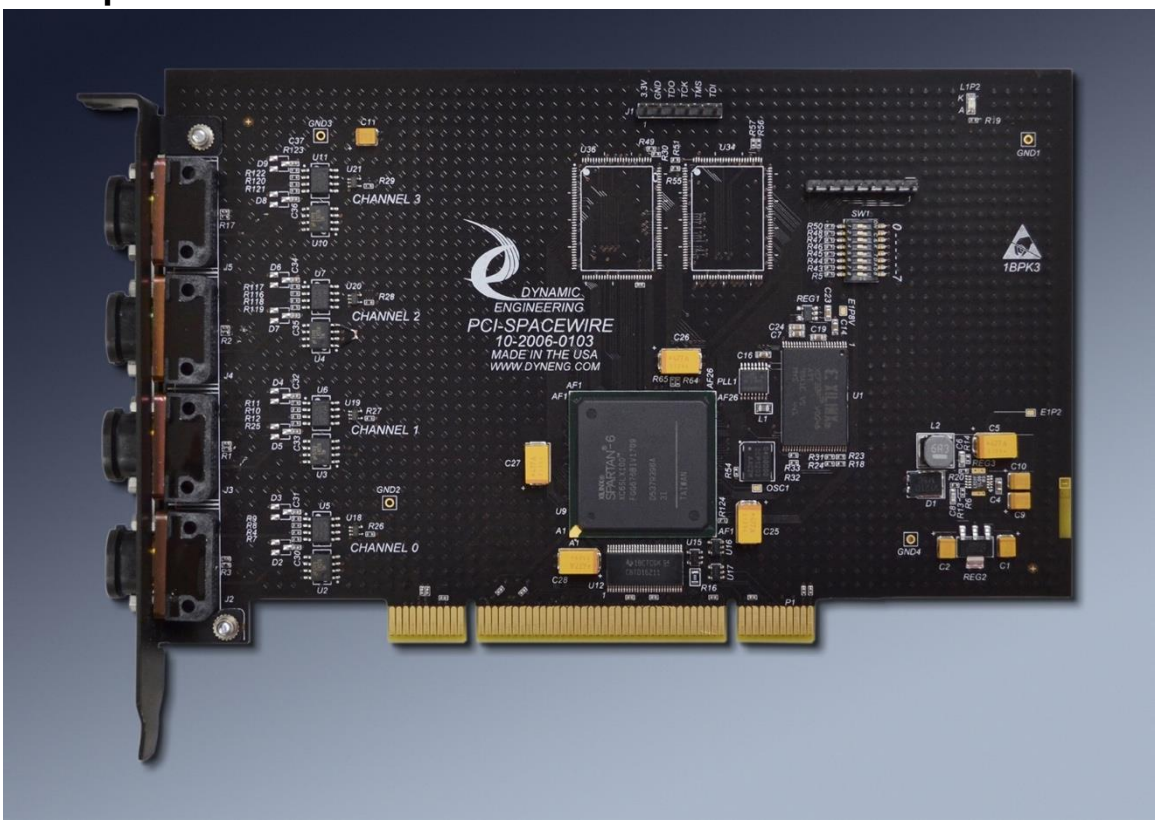
Dynamic Engineering
150 DuBois St. Suite B&C
Santa Cruz, CA 95060
(831) 457-8891
www.dyneng.com
sales@dyneng.com
Est. 1988

PC104p-SpaceWire



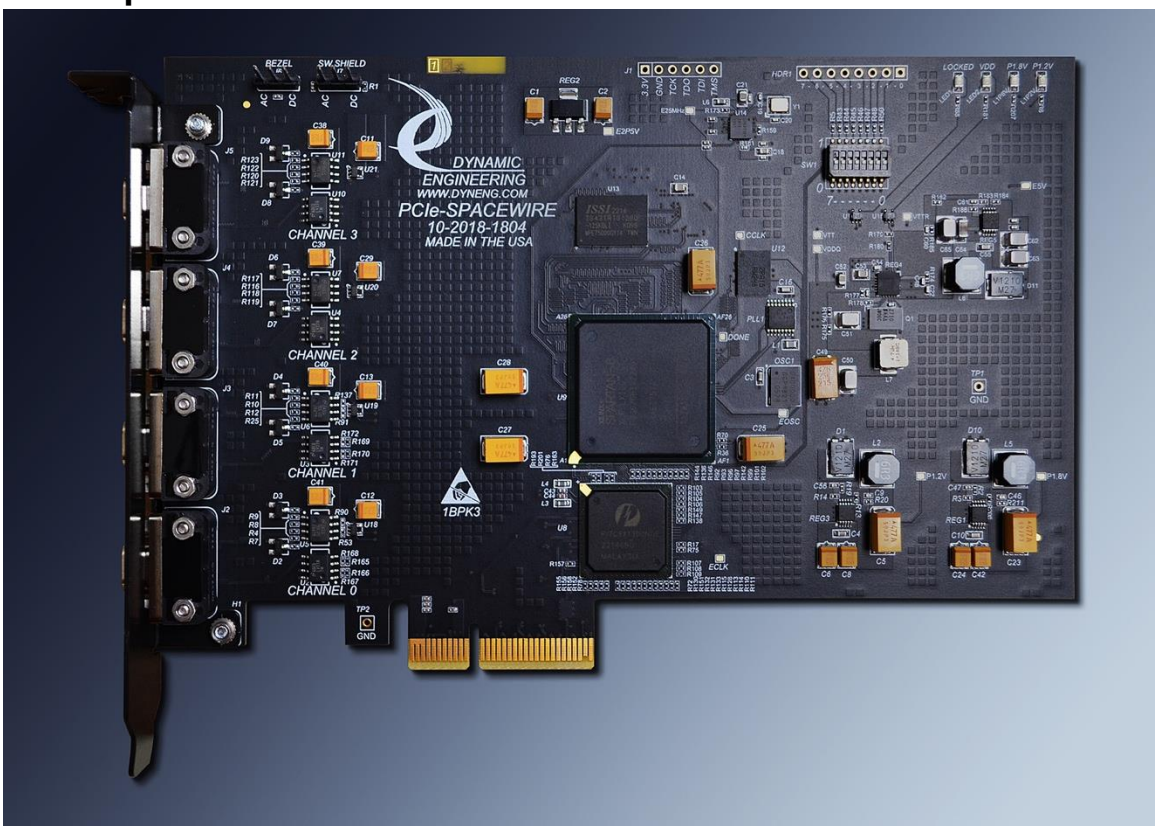
Corresponding Hardware: 10-2008-0904

PCI-SpaceWire



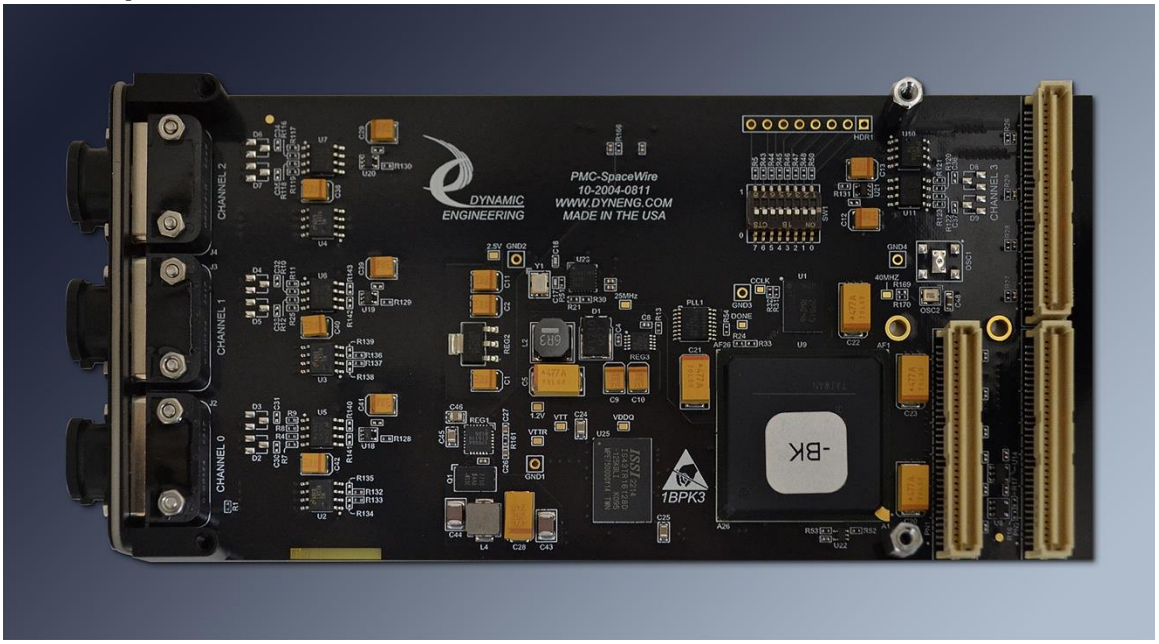
Corresponding Hardware: 10-2006-0106

PCIe-SpaceWire



Corresponding Hardware: 10-2018-1804

PMC-SpaceWire



Corresponding Hardware: 10-2004-0811

NOTE: This manual includes:

1. PC104p-SpaceWire-Monitor
2. PCI-SpaceWire-Monitor
3. PCIe-SpaceWire-Monitor
4. PMC-SpaceWire-Monitor

which are collectively referred to as SpaceWire Monitor henceforth.

NOTE: This manual provides design and usage details of the SpaceWire Monitor. The Monitor function occupies 2 ports. The remaining two ports are standard BK implementations.

Cautions and Warnings

The electronic equipment described herein generates, uses, and can radiate radio frequency energy. Operation of this equipment in a residential area is likely to cause radio interference, in which case the user, at their own expense, will be required to take whatever measures may be required to correct the interference.

Dynamic Engineering's products are not authorized for use as critical components in life support devices or systems without express written approval from the president of Dynamic Engineering.

Connection of incompatible hardware is likely to cause serious damage.

Table of Contents

Tables

Figures

Design Revision History

Table 1: Flash and Software Revision History

Revision	Date	Description
Flash		
2.1	7/2/2021	Initial release
3.1	9/13/2024	Updated for DDR and additional 2 BK type ports
Software		
1	7/2/2021	Initial release Linux only
	9/13/2024	Updated to support added ports, expanded memory, Linux and Windows

Manual Revision History

Table 2: Manual Revision History

Revision	Date	Description
01p1	7/27/21	Initial release of design and manual
01p4	8/23/21	Minor clean-up revs through p4
01p5	9/24/21	Updated due to support of packet mode disabled
1p6	01/02/23	Updated to new manual format, misc. clean-up. Removal of some items [included in SW manuals]
1p7	6/17/24	Update to include PCIe, PMC, PCI, PC104p DDR models
2p0	9/9/24	Update to include added BK ports.

NOTE: Dynamic Engineering has made every effort to ensure that this manual is accurate and complete; that being said, the company reserves the right to make improvements or changes to the product described in this document at any time and without notice. Furthermore, Dynamic Engineering assumes no liability arising out of the application or use of the device described herein.

Key Product Features

Seamlessly monitor SpaceWire traffic between any two devices using two standard SpaceWire cables. SpaceWire Monitor captures and stores full duplex traffic at link speeds up to 200 MHz. The large storage memory accommodates a wide range of packet sizes.

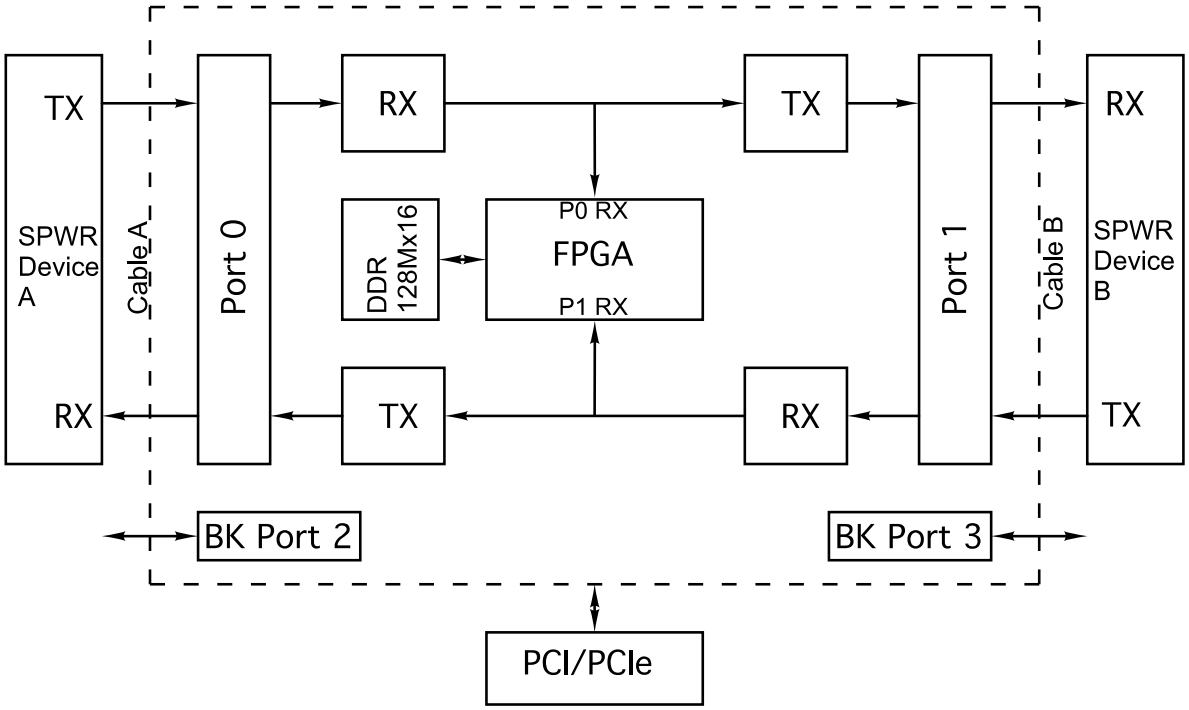


Figure 1: SpaceWire Monitor Connectivity (block diagram)

Product Description

Frequently, it is essential to be able to look at the history of a communication link in order to determine where some unexpected action was triggered. SpaceWire Monitor allows the user to capture the communication between two SpaceWire nodes. The timing between the nodes being captured is not affected. The decoded data can be stored to system memory, disk, or another device.

SpaceWire-Monitor leverages Dynamic Engineering's previous design experience with the SpaceWire interface series of modular IO. For the current revision the design has been updated to incorporate the upgraded SpaceWire-BK version with 32 MB per port. In addition, ports 2 and 3 are now live with standard SpaceWire-BK port operation.

For the Monitor ports, SpaceWire SIN/DIN signals are received from each port (0,1) and retransmitted through the opposite port to complete the path for the nodes being captured. In addition, the received data from both ports is decoded and stored to local memory. Local TX functions are disabled – the cross coupled nodes provide the flow control in this configuration.

Each port has 32 Mbytes of FIFO memory available [per Tx and Rx node]. Unlike a standard SpaceWire node SpaceWire Monitor does not have flow control on the connected SpaceWire ports. All data decoded must be moved to host memory before the local memory fills to prevent overflow.

Each SpaceWire Monitor port has an independent DMA engine to support the application/driver moving data from the on-card storage to the destination of choice. *It is recommended to use a hard drive with cache memory to allow for the burst nature of DMA transfers, for high bandwidth applications an NVMe drive is recommended, please refer to the SW section of this manual for further usage details.*

SpaceWire Monitor incorporates logic to allow starting the monitoring process with the link up and already transferring as well as link down situations. The hardware will find the end of the current transfer and start capture with the new data.

Data is time tagged to allow the two port streams to be compared and manipulated after capture.

Product Specifications

Table 3: Product Specifications

Specification	Description
Memory	32 MB data storage/port
Ports	Two ports to capture data stream from both devices
DMA	Separate DMA engines to move data to host memory
Frequency	Up to 200 MHz. auto-frequency Rx
FLASH	Upgradable as new features are released
Temperature	Industrial temperature components
Connectors	Standard MDM connectors and pinout.
IO	SpaceWire specification ECSS-E-ST-50-12C: compliant

Construction and Reliability

Dynamic Engineering Modules are conceived and engineered for rugged industrial environments. The SpaceWire family is constructed out of 0.062-inch thick High-Temp RoHS-compliant FR4 material.

RoHS and standard processing are available options.

Through-hole and surface-mount components are used. PMC connectors are rated at 1 Amp per pin, 100 insertion cycles minimum. These connectors make consistent, correct insertion easy and reliable. The PCI and PCIe gold fingers are gold over nickel for high reliability and long-lasting connections. PC104p stacking connectors are mounted in accordance with the manufacturers specifications and using gold plated mounting holes for reliable connections.

PMCs are secured against the carrier with four screws attached to the two standoffs and two locations on the front panel. The four screws provide significant protection against shock, vibration, and incomplete insertion for PMC. ccPMC has additional thermal rail mounting points, which also enhance operation in high-vibration environments. PC104p are stacked and retained with inter-module standoffs and mounting hardware. PCI/PCIe cards are retained with bezel mounting screws.

The PCB provides a (typical based on PMC) low temperature coefficient of 2.17 W/°C for uniform heat. This is based upon the temperature coefficient of the base FR4 material of 0.31 W/m-°C, and taking into account the thickness and area of the board. The coefficient means that if 2.17 Watts are

applied uniformly on the component side, the temperature difference between the component side and solder side is one degree Celsius.

The PC104p version of the design has the ground plane tied in with the mounting hardware to allow for inter-stack cooling in conduction cooled environments. Air cooling is a viable method also.

Installation and Interfacing Guidelines

Some general interfacing guidelines are presented below. If you need more assistance, contact Dynamic Engineering. Also refer to the base HW manuals for each SpaceWire board type.

Installation

Warning: Connection of incompatible hardware is likely to cause serious damage.

ESD

Proper ESD handling procedures must be followed when handling the SpaceWire boards. The cards are shipped in anti-static shielded bags. The cards should remain in their bags until ready to use. When installing the card, the installer must be properly grounded and the hardware should be on an anti-static workstation.

Start-Up

Guidelines

Grounds

All electrically grounded equipment should have a fail-safe common ground that is large enough to handle all current loads without affecting noise immunity. Power supplies and power consuming loads should all have their own ground wires back to a common point.

Power Supply

Inputs can be damaged by static discharge or by applying voltage outside of the device-rated voltages.

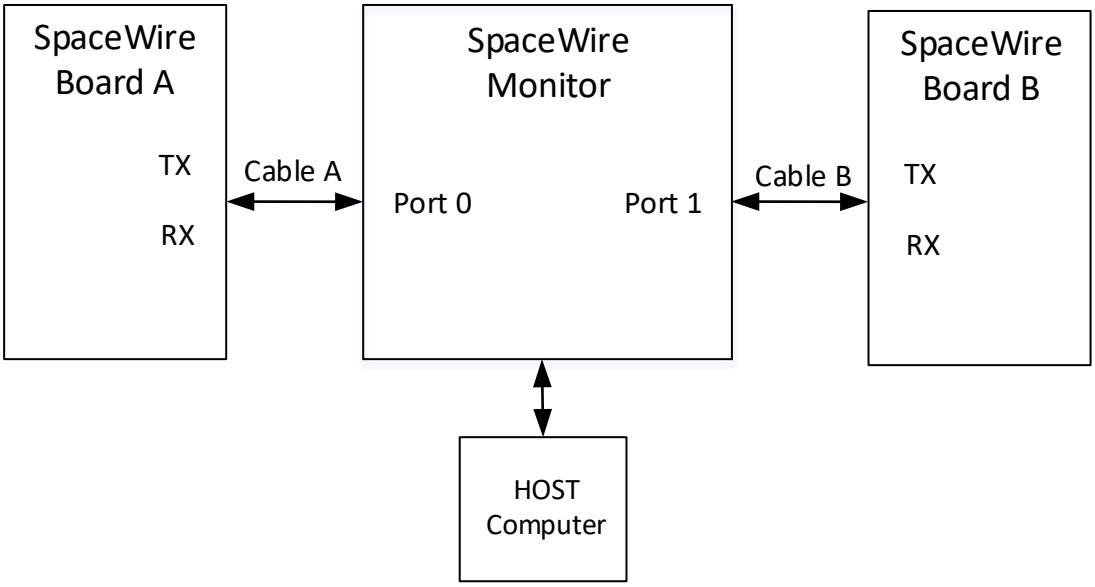
Thermal Considerations

The SpaceWire design consists of CMOS circuits. The power dissipation due to internal circuitry is very low. It is possible to create higher power dissipation with the externally connected logic. If more than one Watt is required to be dissipated due to external loading; forced-air cooling is recommended. With the one-degree differential temperature to the solder side of the board, external cooling is easily accomplished.

Theory of Operation

To use the SpaceWire Monitor, replace the SpaceWire cable between two devices with two cables. Install the Monitor into the appropriate location (PMC, PCI, PCIe, or PCI-104), install the driver, and run the application. Please see the SW section for more information on installing the driver and running the Monitor application.

Figure 2: SpaceWire Monitor Usage Diagram



When the Monitor application & Hardware are in an operational state, the data transferred over the link is captured by the SpaceWire Monitor hardware. (FPGA) In parallel, the SpaceWire driver waits for the FPGA to assert an interrupt indicating data has been captured. The driver initiates a DMA transfer to move the captured data to disc storage. The captured data is logged into output files and is referred to as A-side and B-side data, which is data received (seen) by Port 0 and Port 1 respectively. See the Software Description section later in this manual.

Two capture modes have been designed into the hardware and can be initiated when invoking the application: Link-Down-Start and Link-Up-Start. When Link-Down-Start is used, the Monitor detects when the link comes up and is able to capture the data immediately. When Link-Up-Start is used, the Monitor detects when an EOP occurs, syncs up to the SpaceWire link protocol, and is then able to capture data as it passes through the link. This allows faster synchronization when the link is in a known non-operational state or to filter the message currently active allowing synchronization at the end point.

Address Maps and Register Definitions

This section documents the register addresses, and register bit maps for the SpaceWire Monitor ports. The BK ports are documented later in this document. SpaceWire Monitor register bits are all initialized to '0' on reset.

Because the SpaceWire Monitor channel is a receive (RX) only channel, the majority of transmit (TX) logic that existed in the SpaceWire channel has been disabled. Some logic and state machines in the SpaceWire design contain both RX and TX logic, for that logic, the TX-related signals have been hardwired such that only the RX logic is operational. Some Tx logic is retained to allow for BIT – memory loop-back etc.

Top Level Memory Map

Base	x00 - x4C	Common functions – PLL etc.
Port 0	x50 - x9C	Monitor Port 0 [A]
Port 1	xA0 - xEC	Monitor Port 1 [B]
Port 2	xF0 - x13C	BK Port 0
Port 3	x140 - x18C	BK Port 1

Table 4:SpaceWire Address Map

Register Name	Offset	Description
SPWR_BASE_CNTRL	0x0000	0 Base control register
SPWR_USER_SWITCH	0x0004	1 User switch & status read port
SPWR_TIME_CNTRL	0x0008	2 Time Control Register
SPWR_TIME_COUNT	0x000C	3 Time Code Update Rate
SPWR_PLL_FIFO	0x0010	4 Write to PLL programming FIFO, Read PLL read-back FIFO
SPWR_PLL_STATUS	0x0014	5 Status associated with PLL programming
Monitor Port Definition		
SPWR_CHAN_CNTRL	0x0000	0 Mon Base Control register
SPWR_CHAN_STATUS	0x0004	1 Mon Status register
SPWR_CHAN_FIFO	0x0008	2 Mon FIFOs single word access
SPWR_CHAN_WR_DMA_PNTR	0x000C	3 Mon Write Physical Address for WriteFile
SPWR_CHAN_TX_FIFO_COUNT	0x000C	3 Mon Transmit FIFO Count Partial
SPWR_CHAN_RD_DMA_PNTR	0x0010	4 Mon Write Physical Address for ReadFile
SPWR_CHAN_RX_FIFO_COUNT	0x0010	4 Mon receive FIFO Count Partial
SPWR_CHAN_RX_PKT_LEN	0x0014	5 Mon Read RX packet-length
SPWR_CHAN_TX_AMT	0x0018	6 Mon Tx Almost Empty programmable level
SPWR_CHAN_RX_AFL	0x001C	7 Mon RX almost full level
SPWR_CHAN_STATUS_II	0x0020	8 Mon Extended Status
SPWR_CHAN_RX_PKT_GT_CNTL	0x0024	9 Mon RX Packet FIFO Greater Than Level control
SPWR_CHAN_MONITOR_RD_SIZE	0x0028	10 Mon Packet Descriptor Read Port
SPWR_CHAN_RX_FIFO_CNT_COM	0x002C	11 Mon RX FIFO Count Complete
SPWR_CHAN_TX_FIFO_CNT_COM	0x0030	12 Mon TX FIFO Count Complete
SPWR_CHAN_BRST_OUT_PTR	0x0034	13 Mon Current Physical address for Readfile DMA function
SPWR_CHAN_TX_STRT_ADD	0x0038	14 Mon Tx Starting Address for DDR
SPWR_CHAN_TX_END_ADD	0x003C	15 Mon Tx End Address for DDR
SPWR_CHAN_RX_STRT_ADD	0x0040	16 Mon Rx Starting Address for DDR
SPWR_CHAN_RX_END_ADD	0x0044	17 Mon Rx End Address for DDR
SPWR_CHAN_MONITOR_CNTL	0x0048	18 Monitor Function Control Register
SPWR_CHAN_MONITOR_STATUS	0x004C	19 Monitor Function Status Register

BK Port Definition		
SPWR_CHAN_CNTRL	0x0000	0 BK Base Control register
SPWR_CHAN_STATUS	0x0004	1 BK Status register
SPWR_CHAN_FIFO	0x0008	2 BK FIFOs single word access
SPWR_CHAN_WR_DMA_PNTR	0x000C	3 BK Write Physical Address for Writefile
SPWR_CHAN_TX_FIFO_COUNT	0x000C	3 BK Transmit FIFO Count Partial
SPWR_CHAN_RD_DMA_PNTR	0x0010	4 BK Write Physical Address for Readfile
SPWR_CHAN_RX_FIFO_COUNT	0x0010	4 BK receive FIFO data count
SPWR_CHAN_PKT_LEN	0x0014	5 BK Write/Read Tx/Rx packet-length
SPWR_CHAN_TX_AMT	0x0018	6 BK Tx Almost Empty programmable level
SPWR_CHAN_RX_AFL	0x001C	7 BK RX almost full level
SPWR_CHAN_STATUS_II	0x0020	8 BK Extended Status
SPWR_CHAN_RX_PKT_FF_FULL_CNTL	0x0024	9 BK RX Packet FIFO Almost Full Control register
SPWR_CHAN_TX_FIFO_CNT_COM	0x0028	10 BK TX FIFO Count Complete
SPWR_CHAN_RX_FIFO_CNT_COM	0x002C	11 BK RX FIFO Count Complete
SPWR_CHAN_BRST_IN_PTR	0x0030	12 BK Current Physical address for Writefile DMA function
SPWR_CHAN_BRST_OUT_PTR	0x0034	13 BK Current Physical address for Readfile DMA function
SPWR_CHAN_TX_STRT_ADD	0x0038	14 BK Tx Starting Address for DDR
SPWR_CHAN_TX_END_ADD	0x003C	15 BK Tx End Address for DDR
SPWR_CHAN_RX_STRT_ADD	0x0040	16 BK Rx Starting Address for DDR
SPWR_CHAN_RX_END_ADD	0x0044	17 BK Rx End Address for DDR

Add the Port offset to each port. For example, Port 0 => x50 : SPWR_CHAN_STATUS = x50 + 4 for port 0. xA4 for Port1, xF4 for port 2 and x144 for port 3.

Base Register Definitions

SpaceWire Base Control Register

Table 5: SpaceWire Base Control Register

SPWR_BASE_CNTL	
[0x0000] Base Control Register (read/write)	
Data Bit	**Description**
31	BigEndianDma
30	IO-RST
29-28	Time-Code Control Flags
27-25	Spare
24	PLL USE ALT
23	PLL CHK
22	PLL RD
21	PLL RST
20	PLL Enable
19-15	IO Clock D Initial Divisor [Port3 BK Port 1]
14-10	IO Clock C Initial Divisor [Port2 BK Port 0]
9-5	IO Clock B Initial Divisor [Port1 Mon Port 1]
4-0	IO Clock A Initial Divisor [Port0 Mon Port 0]

31-24, 23-16, 15-8, 7-0 ⇔ 7-0, 15-8, 23-16, 31-24-byte swapping pattern implemented.

All bits are active high and are reset on system power-up or reset; except PLL enable, which defaults to enabled (high) on power-up or reset.

BigEndianDma: (Bit 31) ‘0’ disables this option. ‘1’ enables this option. When operating with a BigEndian platform and using PCI accesses, DMA can have challenges. The register accesses directly over the PCI bus are usually automatically taken care of with byte swapping within the CPU or PCI interface on the CPU. DMA data is written-to or read-from the local memory and is not swapped. The direct read/write from memory ends up with scrambled data [relative to SpaceWire little endian definitions]. Setting this bit will byte reverse the data for the DMA path into the Tx and out of the Rx FIFOs only. Register accesses are not affected.

IoRst (Bit 30) when ‘1’ causes a reset to the DDR controller for each of the 4 ports. Separate port only resets are available within the port. ‘0’ for normal operation.

Time-Code Control Flags: (29-28) These two bits are added to the six-bit time count in bit positions 7 and 6. Their purpose is currently not defined in the SpaceWire specification.

PLL USE ALT: (Bit 24) When set, selects the Alternate PLL address. 0 → x69, 1 → x6A

PLL CHK: (Bit 23) Set to check PLL address.

PLL RD: (Bit 22) when set, selects reading the PLL. When cleared, selects writing to the PLL registers.

PLL RST: (Bit 21) When set, '1' causes a reset to the PLL programming hardware.

PLL Enable: (Bit 20) When this bit is set to a one, the signals used to program and read the PLL are enabled.

By default the driver programs the external PLL to provide IO Clock(s) A-D to the SpaceWire Monitor design. IO Clocks provided by the PLL are required for the design to function properly.

IO Clock A-D Initial Divisor: These four fields determine the divisor used to generate the 10 MHz connection clock rate. IO clock A is used for channel 0, IO clock B is used for channel 1, IO clock C is used for channel 2 and IO clock D is used for channel 3. Depending on the frequencies programmed into the PLL, different divisors are required to achieve the 10 MHz bit-rate required by the SpaceWire specification for establishing the link connection. These fields specify those divisors. The frequency divisor is actually one more than the value entered. A value of zero corresponds to a divisor of one. A value of one corresponds to a divisor of two etc. If the Dynamic Engineering device driver is used, these values are written automatically when the PLL is programmed. Please note: Ports 0,1 correspond to the Monitor ports and are not used. The BK ports require the proper value to be written to ensure proper link establishment.

SpaceWire User Switch Port

Table 6: SpaceWire User Switch Port

SPWR_USER_SWITCH	
[0x0004] User Switch Port (read only)	
Data Bit	**Description**
31	Calibration Done
30-28	spare
27	Channel 3 Interrupt Active
26	Channel 2 Interrupt Active
25	Channel 1 Interrupt Active
24	Channel 0 Interrupt Active
23-20	Xilinx Design Revision Minor
19-16	Xilinx Design Configuration Type
15-8	Xilinx Design Revision Number Major
7-0	Switch Setting

Channel 0-3 Interrupt Active: When a one is read, it indicates that the corresponding channel's interrupt is active. When a zero is read, that interrupt is inactive.

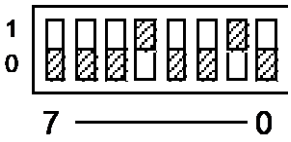
Calibration Done: This bit indicates the DDR initialization process has completed successfully. Also note the blinking LED indicating the DDR clock controller is locked onto the reference and supplying the required high speed clock outputs.

Xilinx Design Configuration Type Major and Minor and Xilinx Design Revision Number: These values of the describe the channel configuration and revision of the Xilinx design.

SpaceWire-Monitor is a type 4 design => S6 with DDR plus 2, 1K FIFOs per path. Default of 32Mbytes per Rx or Tx port

Switch 7-0: The user switch is read through this port. The bits are read as the lowest byte. Access the read-only port as a long word and mask off the undefined bits. The DipSwitch positions are defined in the silkscreen. For example, the switch figure below indicates a 0x12.

Figure 3: DipSwitch Silkscreen Position Definition Example



SpaceWire Time Control Register

Table 7:: SpaceWire Time Control Register

SPWR_TIME_CNTL	
[0x0008] (read/write)	
Data Bit	**Description**
31-15	Spare
14-12	Channel 3 [BK port 1] Time-Code Source Cntl
11	Spare
10-8	Channel 2 [BK port 0] Time-Code Source Cntl
7	Spare
6-4	Channel 1 [Mon port 1] Time-Code Source Cntl
3	Spare
2-0	Channel 0 [Mon port 0] Time-Code Source Cntl

All bits are active high and are reset to zero on system power-up or reset.

Channel 0-3 Time-Code Source Control: These four fields control the source of the time-code signals for the four channels as follows:

"000"-> Disabled: Time-code characters are not sent by this channel.

"001"-> Master: Time-code characters for this channel come from the master timer and time-code counter.

"010"-> Channel 0: Time-code characters received by channel 0 are subsequently sent by this channel.

"011"-> Channel 1: Time-code characters received by channel 1 are subsequently sent by this channel.

"100"-> Channel 2: Time-code characters received by channel 2 are subsequently sent by this channel.

"101"-> Channel 3: Time-code characters received by channel 3 are subsequently sent by this channel.

Write 0-5 to each of the fields to select the source for that channel. For example, writing the "101" pattern to channel 2 will have the time code received by channel 3 passed to channel 2 for retransmission. To use the local master use code "001" and so forth. Ports 0, 1 should be set to "000" since the monitor function does not support this feature.

SpaceWire Time Count Register

Table 8: SpaceWire Time Count Register

SPWR_TIME_COUNT	
[0x000C] (read/write)	
Data Bit	**Description**
31-0	Master Timer Divider Count

Master Timer Divider Count: This count is used to generate the TICK_IN signal when a channel is used as the source for time-codes. The counter is clocked by the 80 MHz link clock and this count represents the count at which the counter resets to zero, increments the time-code and issues a TICK_IN signal. If this field is set to zero, the six-bit time-code counter and the 32-bit tick timer are reset to zero; when the count is set to a non-zero value, counting proceeds. The 32-bit counter allows a maximum time between ticks of approximately 53.68 seconds.

Normally the timer is set to a much lower value, 1 mS for example. Program for N-1 to be exact since the count runs from 0-> programmed value and then back to 0.

SpaceWire PLL Data FIFO

Table 9: SpaceWire PLL Data FIFO

SPWR_PLL_FIFO	
[0x0010] PLL Data FIFO (read/write)	
Data Bit	**Description**
31-0	Data to PLL or Data from PLL

SpaceWire has an improved I2C interface for programming the PLL. Dynamic Engineering driver support packages include utilities to take the .jed file from the Cypress CyberClocks program, parse and load into the FIFO with the proper sequence of controls via Base Control Register. Please see the reference code for the sequence. Linux, VxWorks, Win7 packages.

The data to program the PLL is written to this address. The hardware has a state-machine to read the data from the FIFO and load into the PLL. Similarly, the state-machine can read the data from the PLL and write it to the read side FIFO.

The IO clock is used in the design and the PLLA, PLLB frequencies must be set for proper operation. Operational SW automatically programs the PLL.

SpaceWire PLL Status Register

Table 10: SpaceWire PLL Status Register

SPWR_PLL_STATUS	
[0x0014] PLL Status (read/write)	
Data Bit	**Description**
31-11	Spare
10	PLL Error Latched
9	PLL Done Latched
8	PLL Ready
7	Spare
6	PLL FIFO RX Data Valid
5	PLL FIFO RX Full
4	PLL FIFO RX Empty
3	Spare
2	PLL FIFO TX Data Valid
1	PLL FIFO TX Full
0	PLL FIFO TX MT

The PLL Status bits are used as feed-back to control the transfer of data to and from the PLL FIFO. TX refers to programming the PLL and RX refers to reading back from the PLL.

The Latched Bits {10,9} are held until cleared by writing back with the bit position(s) set. Usually these bits are cleared before starting an operation.

PLL Error Latched: (Bit 10) is set when an error is detected in the I2C transfer. The main purpose for this bit is in discovery for the address of the PLL. The Address can be x6A or x69. Once the correct address is known, this bit should be checked but not set. Sticky bit, write with bit position set to clear.

PLL Done Latched: (Bit 9) is set when the transfer is completed. This bit can be polled to know when the PLL has been programmed or when the PLL has been read.

NOTE: The PLL settling time is in addition to the transfer time. Several mS should be delayed after programming the PLL to make sure the specified frequencies are within range. 10 mS is recommended.

PLL FIFO RX Data Valid: (Bit 6) is set when data is Valid in the output port for the PLL read path. Data is pre-read from the FIFO and held in the FIFO holding register. The FIFO can be Empty and still have 1 word left in the holding register if Valid is still set.

PLL FIFO RX FULL: (Bit 5) is set when the read-back FIFO for the PLL is full.

PLL FIFO RX Empty: (Bit 4) is set when the read-back FIFO for the PLL is empty.

PLL FIFO TX Data Valid: (Bit 2) is set when data is valid in the pipeline between the FIFO and the State –Machine. The bit is cleared each time the data is read. During operation, this bit will toggle to provide some indication that the transfer is occurring.

PLL FIFO TX FULL: (Bit 1) is set when the programming FIFO for the PLL is full.

PLL FIFO TX MT (Bit 0) is set when the programming FIFO for the PLL is empty.

Monitor Register Definitions

SpaceWire Monitor Control Register

Table 11: SpaceWire Monitor Control Register

SPWR_CHAN_CNTRL_0-1	
[0x0000] Channel Control Register (read/write)	
Data Bit	**Description**
31	Read DMA Ready (read only)
29-28	Time-Code Flags (read only)
27	Spare
26	PktCntGtIen
25	Return Valid Packet-Lengths Only Enable
23	Receive FIFO Programmable Level Load
21	Read DMA Arbitration Priority Enable
19	Read DMA Interrupt Enable
17	Force Interrupt
16	Master Interrupt Enable
14	Packet Received Interrupt Enable
13	**RX Error Interrupt Enable**
12	RX Almost Full Interrupt Enable
10	Packet Disable
9	Link Auto-Start
8	Link Start
7	**Link Enable**
6	FIFO Write Control
5	**Receive FIFO Reset**
4	**Transmit FIFO Reset**

All bits are active high and are reset on system power-up or reset.

Read DMA Ready: (Bit 31) These two read-only bits report the DMA state-machine status. If they are read as a one, the corresponding DMA state-machine is idle and available to start a transfer. If the bits are read as a zero, the corresponding DMA state-machine is processing a data transfer.

Time-Code Flags: (Bit 29-28) The time-code flags have been moved to the control register to make room for the latched almost empty/full status bits that were added to the status register. These two read-only bits are currently undefined in the SpaceWire specification and will most likely always be seen as zeros.

PktCntGtIen when ‘1’ enables the Packet Count Greater Than interrupt. If the status bit is set the interrupt is generated [assuming Master is enabled]. The programmed level is in the Packet FIFO Greater Than register. When ‘0’ the status is still available but the interrupt is not created. Level based interrupt – stays active until the count is reduced via reading the packet descriptors or the interrupt enable is disabled.

Return Valid Packet-Lengths Only Enable: (Bit 25) When this bit is set to a one, only valid packet-lengths will be returned. If no new packet has been received since the packet-length FIFO was read,

the packet-length will be returned as zero. When a zero is written to this bit and no new packet has been received since the packet-length FIFO was read, the packet-length from the last packet received will be returned. When enabled, this control allows packet-lengths to be confidently read without first checking the Receive Packet Length Valid status bit in the channel status register.

Receive FIFO Programmable Level Load: (Bit 23) These bits are only valid for channels with external data FIFOs. The load bits must be active during FIFO reset to select the programmable level feature. Once selected, these bits must be set to zero for normal FIFO operation. When set to one, data accesses are instead directed to the almost empty and almost full level registers (See FIFO data sheet for details).

Read DMA Arbitration Priority Enable: (Bit 21) These two bits, when set to one, enable the DMA arbiter to use the TX almost empty and/or RX almost full status to give priority to a channel that is approaching the limits of its FIFOs. The levels written to the TX almost empty and RX almost full registers are used to determine these status values. When these bits are zero, normal round-robin arbitration is used to determine access to the PCI bus for DMA transfers.

Read DMA Interrupt Enable: (Bit 19) These two bits, when set to a one, enable the interrupts for DMA write and read completion for the referenced channel. These two interrupts cannot be disabled by the master interrupt enable.

Force Interrupt: (Bit 17) When this bit is set to one, a system interrupt will occur provided the channel master interrupt enable is set. This is useful to test interrupt service routines.

Master Interrupt Enable: (Bit 16) When this bit is set to a one, all enabled interrupts for the referenced channel (except the DMA interrupts) will be gated through to the PCI host; when this bit is a zero, the interrupts can be used for status without interrupting the host.

Packet Received Interrupt Enable: (Bit 14) When this bit is set to a one, an interrupt will be generated when a complete packet is received, provided the channel master interrupt enable is asserted. When a zero is written to this bit, an interrupt will not be generated, but the latched status can still be read from the channel status register.

RX Error Interrupt Enable: Channel Control Register (Bit 13): This bit is still read/writeable but inside the SpaceWire Monitor design, RX Errors are blocked from generating interrupts. Sources of RX interrupts are bits [12:8] in the SPWR_CHAN_STATUS register and include bits such as Receive FIFO Overflow, which is now handled by the Monitor Status bits, and Credit Error Detected, which would be detected by either of the devices being monitored.

RX Almost Full Interrupt Enable: (Bit 12) When this bit is set to a one, an interrupt will be generated when the receive FIFO level becomes equal or greater to the value specified in the SPWR_CHAN_RX_AFL register, provided the channel master interrupt enable is asserted. When this bit is zero, an interrupt will not be generated, but the status can still be read from the channel status register.

Packet Disable: (Bit 10) When this bit is set to a one, data is transferred without being separated into packets. No end-of-packet characters are generated or received and the packet-length FIFOs are not used. As soon as data is written to the transmit FIFO it will be sent out, provided all other conditions

allow this. When this bit is zero, the data will be sent in packets. Data must be written to the transmit data FIFO and packet lengths must be written to the TX packet-length FIFO, before data can be transferred.

Link Auto-Start: (Bit 9) The behavior of this bit is similar to Link start, however, when this bit is set and Link start is not set, the state-machine will not proceed to the Started state unless a Null character has been seen, which indicates that the other node is attempting to establish a connection. This bit allows the connection process to be cleanly initiated from one side of the link only.

Link Start: (Bit 8) When this bit is set to a one, the link state-machine will move from the Ready state to the Started state and will attempt to establish a connection with another node. When this bit is zero, the state-machine will remain in the Ready state, provided it has already achieved this state. Once the state-machine has left the Ready state, this bit has no effect.

Link Enable: Channel Control Register (Bit 7): This bit is still read/writable but inside the SpaceWire Monitor, it has been replaced by the decoded monitor mode bits [1:0] in the SPWR_CHAN_MONITOR_CONTROL register.

FIFO Write Control: (Bit 6) When this bit is set to a one, any data written to the Tx FIFO will be written to the receive FIFO. This allows for fully testing the data FIFO path without connecting to another SpaceWire node. When this bit is zero, normal operation is enabled.

Transmit/Receive Reset (Bit 4 and 5) These bits when set cause a reset of the related logic. Some functions are protected to only reset when the monitor is not enabled to prevent losing track of link status etc. For a complete reset clear the monitor enable bits. The resets are also used to clear the DDR counts.

SpaceWire Monitor Status Register

Table 12: SpaceWire Channel Status Register

SPWR_CHAN_STATUS_0-1	
[0x0004] (status read/latch clear write)	
Data Bit	**Description**
31	Latched Rx FIFO Almost Full
30	Latched Tx FIFO Almost Empty
29-24	Time-Code Data
23	Interrupt Active
22	Receive Packet Length Valid
20	**SpaceWire Link Established**
19	Read DMA Error
18	Write DMA Error
17	Read DMA List Complete
16	Write DMA List Complete
15	PktCntGt
14	Packet Received
13	**Receive Error**
12	**Receive FIFO Overflow**
11	**Credit Error Detected**
10	**Escape Error Detected**
9	**Disconnect Error Detected**
8	**Parity Error Detected**
7	Receive Data Valid
6	Rx FIFO Full
5	Rx FIFO Almost Full
4	Rx FIFO Empty
3	Tx FIFO Empty
2	Tx FIFO Full
1	Tx FIFO Almost Empty
0	Tx FIFO Empty

Latched Rx FIFO Almost Full: (Bit 31) When a one is read, it indicates that the receive FIFO data count has become greater than the value in the SPWR_CHAN_RX_AFL register. A zero indicates that the FIFO has not become almost full. This bit is latched and can be cleared by writing back to the Status register with a one in this bit position.

Latched Tx FIFO Almost Empty: (Bit 30) When a one is read, it indicates that the transmit FIFO data count has become less than the value in the SPWR_CHAN_TX_AMT register. A zero indicates that the FIFO has not become almost empty. This bit is latched and can be cleared by writing back to the Status register with a one in this bit position.

Time-Code Data: The last time-code value received can be read from this six-bit data-field. The TICK_OUT received status bit will indicate if the data is a new valid time-code value. A time-code is considered valid if it is one more than the previous stored value. If the time-code is the same as the

stored value, it is assumed to be a duplicate resulting from a cycle in the SpaceWire network and is therefore ignored. If the time-code meets neither of these conditions, it is stored, but the TICK_OUT signal is not asserted until the next time-code is received and is one more than that last stored value. At this point, the time-code is deemed to be re-synchronized.

Interrupt Active: When a one is read, it indicates that an enabled interrupt condition (other than the DMA interrupts) is active for the referenced channel. A zero indicates that no enabled interrupt condition is active.

Receive Packet Length Valid: When a one is read, there is at least one valid receive packet-length value available. When this bit is a zero, it indicates that there are no valid receive packet-length values.

SpaceWire Link Established: Channel Status Register bit [20] – In a SpaceWire design, this bit indicates a link has been established. In the SpaceWire Monitor design, it indicates the link-up, active, and being monitored.

Read/Write DMA Error: When a one is read, it indicates that an error has occurred while the corresponding DMA was in progress. This could be a target or master abort or an incorrect direction bit in one of the DMA descriptors. These bits are latched and must be cleared by writing the same bit back to the channel status port. A zero indicates that no DMA error has occurred.

Read/Write DMA List Complete: When a one is read, it indicates that the corresponding DMA has completed. These bits are latched and must be cleared by writing the same bit back to the channel status port. A zero indicates that the corresponding DMA has not completed.

PktCntGt (15) when set ‘1’ means the Rx Packet FIFO has more than the programmed threshold number of entries. This bit can cause an interrupt if the PktCntGtIen bit is also set in the control register. When ‘0’ the level is less than or equal to the programmed level.

Packet Received: When a one is read, it indicates that a packet has been received since this bit was last cleared. This bit is latched and must be cleared by writing the same bit back to the channel status port. A zero indicates that a packet has not been received.

Receive Error: Channel Status Register bit [13] – This bit is now hard-wired to ‘0’ as the five error sources for it (Channel Status Register bits [12:8]) are all hard-wired to ‘0’.

Receive FIFO Overflow: Channel Status Register bit [12] – This bit is now hard-wired to ‘0’ as Receive FIFO Overflows are handled using the SPWR_CHAN_MONITOR_STATUS register.

Credit Error Detected: Channel Status Register bit [11] – this bit is now hard-wired to ‘0’ as Credit Error detection is handled by the devices being monitored.

Escape Error Detected: Channel Status Register bit [10] – This bit is now hard-wired to ‘0’. Disabling this Escape Error Detection allows the monitor driver to continue running and capturing data when/if the monitored link goes down and comes back up.

Disconnect Error Detected: Channel Status Register bit [9] – This bit is now hard-wired to ‘0’. Disabling this Disconnect Error Detection allows the monitor driver to continue running and capturing data when/if the monitored link goes down and comes back up.

Parity Error Detected: Channel Status Register bit [8] – This bit is now hard-wired to '0' as the monitor logic/driver/application is not currently configured to test/log/report parity errors and it is expected the detection of parity errors will be handled by the devices being monitored.

NOTE: The memory path for the Rx function is IO FIFO => DDR => DMA FIFO. The Tx function is DMA FIFO => DDR => TX IO FIFO. Loop-back reads from the Tx IO FIFO and writes to the Rx IO FIFO.

Receive Data Valid: When a one is read, there is at least one word of valid receive data. When data is written to the receive FIFO, the first four words are read and held in batches to be ready for a PCI read DMA or single-word read. Therefore, although the FIFO is empty if this bit is set, there are as many as four additional long-words of receive data. A zero indicates that there is no valid receive data.

Receive FIFO Full: When a one is read, the receive data FIFO for the corresponding channel is full; when a zero is read, there is room for at least one more data-word in the FIFO. This bit is a combination of IO and DMA FIFOs full.

Receive FIFO Almost Full: When a one is read, the number of data-words in the receive data FIFO for the corresponding channel is greater or equal to the value written to the SPWR_CHAN_RX_AFL register for that channel; when a zero is read, the level is less than that value.

Receive FIFO Empty: When a one is read, the receive data FIFO for the corresponding channel contains no data; when a zero is read, there is at least one data-word in the FIFO. This bit is the DMA FIFO MT. If making sure all data is read it is recommended to use the Rx Count Complete as this includes all three memory elements.

The Tx FIFO status is included for BIT / loop-back purposes.
The empty flags for both FIFOs are shown. If '1' the corresponding FIFO is empty.
The Almost Empty flag is set when the total count of the Tx chain is below the programmed value.
The Full flag indicates the DMA and IO FIFOs are full.

SpaceWire Monitor FIFO Port

Table 13: SpaceWire Monitor FIFO Port

SPWR_CHAN_WR_DMA_PNTR_0-1	
[0x0008] (R/W)	
Data Bit	**Description**
31-0	Data

These ports are used for single-word accesses into the channel TX FIFO and out of the channel RX FIFO. Reading an Empty FIFO will return the last valid data. Writing to a Full FIFO will go in the bit bucket.

SpaceWire Monitor Write DMA Pointer Port

Table 14: SpaceWire Monitor Read DMA Pointer Port

SPWR_CHAN_WR_DMA_PNTR_0-1	
[0x000C] (write only)	
Data Bit	**Description**
31-0	First Chaining Descriptor Physical Address

This write-only port is used to initiate a scatter-gather write [TX] DMA. When the address of the first chaining descriptor is written to this port, the DMA engine reads three successive long words beginning at that address. Essentially, this data acts like a chaining descriptor value pointing to the next value in the chain.

The 1st read retrieve the address of the first memory block (DMA buffer) to write to, the 2nd is the length in bytes of that block, and 3rd is the address of the next chaining descriptor in the linked list (buffer memory blocks). This process is continued until the end-of-chain bit of the next pointer value read indicates it is the last chaining descriptor in the list.

All three values are on LW boundaries and LW in size. Addresses for successive parameters are incremented. The addresses are physical addresses the HW will use on the PCI bus to access the Host memory. In most operating systems you will need to convert from virtual to physical. The length parameter is a number of bytes, and must be on a LW divisible number of bytes.

Status for the DMA activity can be found in the channel control register and channel status register.

NOTES:

1. Writing a zero to this port will abort a write DMA in progress.
2. End of chain should not be set for the address written to the DMA Pointer Address Register. End of chain should be set when the descriptor follows the last length parameter.
3. The Direction should be set to ‘0’ for Burst In DMA in all chaining descriptor locations.
4. The segment size is 31-0 however the byte count is shifted down two positions to create a LW count in memory. This means the actual count is 1G-1 [4Gbytes minus 1 LW] or xFFFFFFFC for the max byte count.

NOTE: The direction bit (bit 1) must be ‘0’ when the physical address of the first chaining descriptor is written to this register or write DMA error will result.

SpaceWire Monitor Read DMA Pointer Port

Table 15: SpaceWire Monitor Read DMA Pointer Port

SPWR_CHAN_RD_DMA_PNTR_0-1	
[0x0010] (write only)	
Data Bit	**Description**
31-0	First Chaining Descriptor Physical Address

This write-only port is used to initiate a scatter-gather read [RX] DMA. When the address of the first chaining descriptor is written to this port, the DMA engine reads three successive long words beginning at that address. Essentially, this data acts like a chaining descriptor value pointing to the next value in the chain.

The 1st read retrieve the address of the first memory block (DMA buffer) to write to, the 2nd is the length in bytes of that block, and 3rd is the address of the next chaining descriptor in the linked list (buffer memory blocks). This process is continued until the end-of-chain bit of the next pointer value read indicates it is the last chaining descriptor in the list.

All three values are on LW boundaries and LW in size. Addresses for successive parameters are incremented. The addresses are physical addresses the HW will use on the PCI bus to access the Host memory. In most operating systems you will need to convert from virtual to physical. The length parameter is a number of bytes, and must be on a LW divisible number of bytes.

Status for the DMA activity can be found in the channel control register and channel status register.

NOTES:

5. Writing a zero to this port will abort a write DMA in progress.
6. End of chain should not be set for the address written to the DMA Pointer Address Register. End of chain should be set when the descriptor follows the last length parameter.
7. The Direction should be set to ‘1’ for Burst Out DMA in all chaining descriptor locations.
8. The segment size is 31-0 however the byte count is shifted down two positions to create a LW count in memory. This means the actual count is 1G-1 [4Gbytes minus 1 LW] or xFFFFFFFC for the max byte count.

NOTE: The direction bit (bit 1) must be set when the physical address of the first chaining descriptor is written to this register or read DMA error will result.

SpaceWire Monitor TX FIFO Partial Count Port

Table 16: SpaceWire Monitor RX FIFO Data Count Port

SPWR_CHAN_RX_FIFO_COUNT_0-1	
[0x0008] TX FIFO Partial Data Count (read only)	
Data Bit	**Description**
31-16	Number of TX Packet-length Values Stored
15-0	Number of TX Data-Words Stored in IO FIFO

These read-only register ports report the number of 32-bit data words in the Tx FIFOs. The count for the Data corresponds to the data in the IO FIFO – ready to transmit. See the alternate “complete” count port for full Tx chain count.
NOTE: The transmit features are included to support BIT. Actual transmission is not supported for Monitor ports.

The transmit packet-length count field is in this register. This allows the user to know how many packet-length values can be safely written to the packet-length FIFO without causing an overflow. It also aids in error recovery if there was a connection error during a packet transmission and the transmitter is unable to automatically purge the remaining data from the impacted packet. In this case the user must intervene to reestablish a consistent packet/data state by resetting the transmit FIFO and rewriting packet data and packet-length values. Knowing how many packets were pending when the error occurred will greatly aid in this procedure.

SpaceWire Monitor RX FIFO Partial Count Port

Table 17: SpaceWire Monitor RX FIFO Data Count Port

SPWR_CHAN_RX_FIFO_COUNT_0-1	
[0x0010] RX FIFO Partial Data Count (read only)	
Data Bit	**Description**
31-16	Number of RX Packet-length Values Stored
15-0	Number of RX Data-Words Stored in DMA FIFO

These read-only register ports report the number of 32-bit data words in the corresponding receive DMA FIFO. There are four additional latches in the read DMA pipeline that may contain data, which allows this value to be a maximum of 0x4003 Refer to the “complete Count” port for the full chain count.

The receive packet-length count field is also in this register. This allows the user to know how many packet-length values are stored in the packet-length FIFO. The first receive packet-length written to the packet-length FIFO is read as soon as it is written to be immediately available. This causes the receive packet-length valid status bit to be set and the packet-length count to become one. As subsequent packets are received, the count will increment and as lengths are read, the count will decrement.

SpaceWire Monitor RX Packet-Length FIFO Ports

Table 18: SpaceWire Monitor RX Packet-Length FIFO Ports

SPWR_CHAN_RX_PKT_LEN_0-1	
[0x0014] RX Packet-Length FIFO Ports (write only)	
Channels with 2 Gbyte Maximum Packet-Lengths	
Data Bit	**Description**
31	Terminate Packet with an Error-End-of-Packet
30-0	Packet Length (31 bits)

These ports access the write transmit packet-length FIFO and the read receive packet-length FIFO ports for the respective channels. These FIFOs are used to store packet lengths for sending transmit packets and reading received packets.

The bit above the packet-length field is an error flag. If this bit is read as a one, it indicates that either the packet was terminated with an EEP (Error-End-of-Packet) or an error condition occurred while the packet was being received. If this bit is written as a one, it indicates that the transmit packet should be terminated with an EEP rather than an EOP.

SpaceWire Monitor TX Almost Empty Register

Table 19: SpaceWire Monitor TX Almost Empty Register

SPWR_CHAN_TX_AMT_0-1	
[0x0018] RX Almost Full Level (read/write)	
Data Bit	**Description**
31-0	TX FIFO Almost Empty Level

These read/write ports access the transmitter almost-empty level registers for the respective channels. When the number of data words in the transmit data FIFO is less than this value, the almost empty status bit is set and an interrupt may be generated if it is enabled. While the port is defined larger than the FIFO size in LW, the comparison should be set to a value within the range of the FIFO for proper operation. The comparator uses the full Tx memory chain count for comparison.

SpaceWire Monitor RX Almost Full Level Register

Table 20: SpaceWire Monitor RX Almost Full Level Register

SPWR_CHAN_RX_AFL_0-1	
[0x001C] RX Almost Full Level (read/write)	
Data Bit	**Description**
31-0	RX FIFO Almost Full Level

These read/write ports access the receiver almost-full level registers for the respective channels. When the number of data words in the receive data FIFO is greater than this value, the almost full status bit is set and an interrupt may be generated if it is enabled. While the port is defined larger than the FIFO size in LW, the comparison should be set to a value within the range of the FIFO for proper operation. The comparator uses the full Tx memory chain count for comparison.

SpaceWire Monitor Extended Status

Table 21: SpaceWire Monitor Status II

SPWR_CHAN_STATUS_II_0-1	
[0x0020] RX Almost Full Level (read only)	
Data Bit	**Description**
31-20	Rx Packet Count
19-18	Spare
17-12	Time Code In
11-10	Port Number
9-4	Tx Credit
3	Spare
2-0	FCT Count

This port contains “extra” status used in debugging and not normally used in operation. Brief descriptions follow:
FCT or Flow Control Token Count : current number of outstanding FCTs.
Transmit Credit: Number of characters port is authorized to send
Port Number: 0,1,2,3 [binary] number of physical port. Monitor ports are 0,1 this design.
Time Code data into port.
Receive Packet Count from Packet FIFO.

SpaceWire Monitor RX Packet FIFO Greater Than Level

Table 22: SpaceWire Monitor RX Packet FIFO Greater Than Level Register

SPWR_CHAN_RX_PKT_GT_CNTL_0-1	
[0x0024] RX Packet FIFO Full Control Register (read/write)	
Data Bit	**Description**
31-12	Spare
11-0	RX Packet FIFO Greater Than Level

These bits are set to 0x3DF on system power-up or reset.

RX Packet FIFO Greater Than Level: These bits set the RX Packet FIFO Greater Than Threshold level. By default, these bits are set to 0x3DF. The RX Packet FIFO has 4095 locations (0xFFF) therefore, by default, the logic will see the FIFO as full once 991 or more locations are occupied. Once the threshold is reached passed the corresponding status bit PktCntGt will be ‘1’. If the interrupt enable is set the status can cause a level type interrupt – cleared by reading the Packet FIFO until the level is less than or equal to the programmed threshold or disabling the interrupt enable. When operating with high density small packets it is possible to take longer to process the per packet interrupt than the gap between packets. The level-based interrupt can be used to make sure software keeps up since overlapped per packet interrupts can lead to missed packet reads. The threshold may need to be reduced to insure closer to real time processing.

SpaceWire Monitor Read Length Register

Table 23: SpaceWire Monitor Read Length Size Register

SPWR_CHAN_MONITOR_RD_SIZE	
[0x0028] Channel Monitor Read Size Register (read/write)	
SPWR Monitor Read Size Register	
Data Bit	**Description**
31	Spare
30-0	Packet Read Length Size

Packet Read Length Size: Read from the Received Packet FIFO to obtain the received packet read length size. The Packet FIFO is 4095 deep. The packets are stored in the same order as the data is stored in the Data FIFO path. Larger data transfers can be done and recomposed after the fact using the descriptors. Data is packed on a 32 bit basis – each new descriptor always references data on a LW boundary.

SpaceWire Monitor RX FIFO Count Complete

Table 24: SpaceWire Monitor Rx FIFO Count Complete

SPWR_CHAN_RX_FIFO_CNT_COM	
[0x002C] Channel Monitor Rx FIFO Count (read only)	
Data Bit	**Description**
31-0	Rx Full Chain Data Stored Count

The Rx Data path includes the IO FIFO, DMA FIFO, DDR allocated, and a 4 deep pipeline to support the DMA function. The count in this register includes the counts from all of these sources. Some counts are based on non-PCI based clocks and need to be converted. The count will be correct when static and may jump slightly when data is actively flowing. Each count represents 1 32 bit word.

SpaceWire Monitor TX FIFO Count Complete

Table 25: SpaceWire Monitor Tx FIFO Count Complete

SPWR_CHAN_TX_FIFO_CNT_COM	
[0x0030] Channel Monitor Tx FIFO Count (read only)	
Data Bit	**Description**
31-0	Tx Full Chain Data Stored Count

The Tx Data path includes the DMA FIFO, IO FIFO, and DDR allocated. The count in this register includes the counts from all of these sources. Some counts are based on non-PCI based clocks and need to be converted. The count will be correct when static and may jump slightly when data is actively flowing. Each count represents 1 32 bit word.

SpaceWire Monitor RX DMA Pointer

Table 26: SpaceWire Monitor Rx DMA PTR

SPWR_CHAN_BURST_OUT_PTR	
[0x0034] Channel Monitor ReadFile Current Location (read only)	
Data Bit	**Description**
31-0	Rx Physical Address Pointer

Normally not required, however when debugging DMA operations, it is sometimes helpful to see where the HW is pointing. The DMA operation keeps track of the address being accessed or about to be accessed. Reading this port returns the current address in the Burst Out DMA controller used for the Readfile operation.

SpaceWire Monitor TX Start Address DDR

Table 27: SpaceWire Monitor Tx Start Address DDR

SPWR_CHAN_TX_STRT_ADD	
[0x0038] Channel Monitor DDR Transmit Starting Address (read/write)	
Data Bit	**Description**
31-0	Tx DDR Start Address

HW defaults to 32 MegaByte allocations. Read this port to see the starting address for this ports Tx DDR allocation. Write to this port to change the starting address. If changing make sure the DDR is not operating with the DDR disable function, reset and restart. Make sure the boundaries are on page boundaries for efficient operation. Start and End addresses are included in the range

SpaceWire Monitor TX End Address DDR

Table 28: SpaceWire Monitor Tx End Address DDR

SPWR_CHAN_TX_END_ADD	
[0x003C] Channel Monitor DDR Transmit Ending Address (read/write)	
Data Bit	**Description**
31-0	Tx DDR End Address

See notes above

SpaceWire Monitor RX Start Address DDR

Table 29: SpaceWire Monitor Rx Start Address DDR

SPWR_CHAN_RX_STRT_ADD	
[0x0040] Channel Monitor DDR Receive Starting Address (read/write)	
Data Bit	**Description**
31-0	Rx DDR Start Address

HW defaults to 32 MegaByte allocations. Read this port to see the starting address for this ports Rx DDR allocation. Write to this port to change the starting address. If changing make sure the DDR is not operating with the DDR disable function, reset and restart. Make sure the boundaries are on page boundaries for efficient operation. Start and End addresses are included in the range

SpaceWire Monitor RX End Address DDR

Table 30: SpaceWire Monitor Rx End Address DDR

SPWR_CHAN_RX_END_ADD	
[0x0044] Channel Monitor DDR Receive Ending Address (read/write)	
Data Bit	**Description**
31-0	Rx DDR End Address

See notes above

SpaceWire Monitor Mode Control Register

Table 31: SpaceWire Monitor Control Register

SPWR_CHAN_MONITOR_CONTROL	
[0x0048] Channel Monitor Control Register (read/write)	
SPWR Monitor Control Register	
Data Bit	**Description**
31-2	Spare
1-0	Monitor Mode

Monitor Mode: these two bits control the Monitor Mode of operation for the channel.

Monitor Mode	Name	Definition
“00”	Disabled:	Monitor is disabled
“01”	Link-Down-Start:	Configures Monitor to capture data after link comes up
“10”	Link-Up-Start:	Configures Monitor to sync to an active/running link then capture data
“11”	Reserved:	Same as “01” Link-Down-Start

SpaceWire Monitor Status Register

Table 32: SpaceWire Monitor Status Register

SPWR_CHAN_MONITOR_STATUS	
[0x004C] Channel Monitor Status Register (read/write)	
SPWR Monitor Status Register	
Data Bit	**Description**
31-16	Packet Overflow Count
15-1	Spare
0	Monitor Pause Activated

Monitor Pause Activated: This write ‘1’ to clear status bit is set to ‘1’ when the number of packets stored in the Packet FIFO reaches or exceeds 4,063 (0xFDF).

Packet Overflow Count: These bits indicate the number of packets that were not captured and stored in the Packet FIFO.

Under normal operation conditions, the Monitor Pause Activated should not be set and the Packet Overflow Count should remain 0x0000. The Monitor Pause Activated status bit is set when the internal pause monitor (pause_mon) signal is asserted and the hardware attempts to write the packet FIFO. The pause monitor signal is asserted when the Packet FIFO reaches or exceeds 4,063 (0xFDF). Once the Monitor Pause Activated status bit is set, it stays set until a logic reset of the hardware occurs or a ‘1’ is written to bit [0] of the SPWR Monitor Status register.

The Packet Overflow Count increments each time the hardware attempts to write the packet FIFO while the pause monitor signal is asserted. The Packet Overflow Counter is cleared by disabling the Monitor (Monitor Mode = 00) or by a logic reset.

BK Register Definitions

SpaceWire BK Control Register

Table 33: SpaceWire BK Control Register

SPWR_CHAN_CNTRL_0-1	
[0x0000] Channel Control Register (read/write)	
Data Bit	**Description**
31	Read DMA Ready (read only)
30	Write DMA Ready (read only)
29-28	Time-Code Flags (read only)
27	DDR Disable
26	PktCntGtIen
25	Return Valid Packet-Lengths Only Enable
24	Transmit Packet Length Repeat
23	Spare
22	Spare
21	Read DMA Arbitration Priority Enable
20	Write DMA Arbitration Priority Enable
19	Read DMA Interrupt Enable
18	Write DMA Interrupt Enable
17	Force Interrupt
16	Master Interrupt Enable
15	Tick Received Interrupt Enable
14	Packet Received Interrupt Enable
13	RX Error Interrupt Enable
12	RX Almost Full Interrupt Enable
11	Tx Almost Empty Interrupt Enable
10	Packet Disable
9	Link Auto-Start
8	Link Start
7	Link Enable
6	FIFO Loop-Back Enable
5	Receive FIFO Reset
4	Transmit FIFO Reset
3-0	IO Clock Divisor

All bits are active high and are reset on system power-up or reset.

IO Clock Divisor: This field determines the divisor used to generate the operational clock rate. The frequency divisor is actually one more than the value entered. A value of zero corresponds to a divisor of one, one corresponds to a divisor of two etc.

Transmit/Receive FIFO Reset: When one or both of these bits are set to a one, the corresponding data FIFO, packet-length FIFO and control and status circuitry will be reset. When these bits are zero, normal FIFO operation is enabled. FIFO resets are referenced to the PCI clock, two periods are required for proper reset.

FIFO Loop-Back Enable: When this bit is set to a one, any data written to the transmit FIFO will be immediately transferred to the receive FIFO. This allows for fully testing the data FIFOs without connecting to another SpaceWire node. When this bit is zero, normal operation is enabled.

Link Enable: When this bit is set to a one, the link connection process will be initiated. The connection state-machine will proceed to the Ready state until Start or Auto-Start is asserted. When this bit is zero, the link connection will be disabled.

Link Start: When this bit is set to a one, the link state-machine will move from the Ready state to the Started state and will attempt to establish a connection with another node. When this bit is zero, the state-machine will remain in the Ready state, provided it has already achieved this state. Once the state-machine has left the Ready state, this bit has no effect.

Link Auto-Start: The behavior of this bit is similar to Link start, however, when this bit is set and Link start is not set, the state-machine will not proceed to the Started state unless a Null character has been seen, which indicates that the other node is attempting to establish a connection. This bit allows the connection process to be cleanly initiated from one side of the link only.

Packet Disable: When this bit is set to a one, data is transferred without being separated into packets. No end-of-packet characters are generated or received and the packet-length FIFOs are not used. As soon as data is written to the transmit FIFO it will be sent out, provided all other conditions allow this. When this bit is zero, the data will be sent in packets. Data must be written to the transmit data FIFO and packet lengths must be written to the TX packet-length FIFO, before data can be transferred.

TX Almost Empty Interrupt Enable: When this bit is set to a one, an interrupt will be generated when the transmit FIFO level becomes equal or less than the value specified in the SPWR_CHAN_TX_AMT register, provided the channel master interrupt enable is asserted. When this bit is zero, an interrupt will not be generated, but the status can still be read from the channel status register.

RX Almost Full Interrupt Enable: When this bit is set to a one, an interrupt will be generated when the receive FIFO level becomes equal or greater to the value specified in the SPWR_CHAN_RX_AFL register, provided the channel master interrupt enable is asserted. When this bit is zero, an interrupt will not be generated, but the status can still be read from the channel status register.

RX Error Interrupt Enable: When this bit is set to a one, an interrupt will be generated when a receiver error condition is detected, provided the channel master interrupt enabled is asserted. When a zero is written to this bit, an interrupt will not be generated, but the latched status can still be read from the channel status register.

Packet Received Interrupt Enable: When this bit is set to a one, an interrupt will be generated when a complete packet is received, provided the channel master interrupt enable is asserted. When a zero

is written to this bit, an interrupt will not be generated, but the latched status can still be read from the channel status register.

Tick Received Interrupt Enable: When this bit is set to a one, an interrupt will be generated when a valid time-code is received, provided the channel master interrupt enable is asserted. When a zero is written to this bit, an interrupt will not be generated, but the latched status can still be read from the Interrupt Status register.

Master Interrupt Enable: When this bit is set to a one all enabled interrupts for the referenced channel (except the DMA interrupts) will be gated through to the PCI host; when this bit is a zero, the interrupts can be used for status without interrupting the host.

Force Interrupt: When this bit is set to a one a system interrupt will occur provided the channel master interrupt enable is set. This is useful to test interrupt service routines.

Write/Read DMA Interrupt Enable: These two bits, when set to one, enable the interrupts for DMA write and read completion for the referenced channel. These two interrupts cannot be disabled by the master interrupt enable.

Write/Read DMA Arbitration Priority Enable: These two bits, when set to one, enable the DMA arbiter to use the TX almost empty and/or RX almost full status to give priority to a channel that is approaching the limits of its FIFOs. The levels written to the TX almost empty and RX almost full registers are used to determine these status values. When these bits are zero normal round-robin arbitration is used to determine access to the PCI bus for DMA transfers.

Transmit Packet-Length Repeat: When this bit is set to a one, the current transmit packet-length will be used continually until there is no more transmit data or this control bit is cleared. When a zero is written to this bit, the packet-length FIFO must be read to obtain a new packet-length for each transmitted packet.

Return Valid Packet-Lengths Only Enable: When this bit is set to a one, only valid packet-lengths will be returned. If no new packet has been received since the packet-length FIFO was read, the packet-length will be returned as zero. When a zero is written to this bit and no new packet has been received since the packet-length FIFO was read, the packet-length from the last packet received will be returned. When enabled, this control allows packet-lengths to be confidently read without first checking the Receive Packet Length Valid status bit in the channel status register.

PktCntGtIen when ‘1’ enables the Packet Count Greater Than interrupt. If the status bit is set the interrupt is generated [assuming Master is enabled]. The programmed level is in the Packet FIFO Greater Than register. When ‘0’ the status is still available but the interrupt is not created. Level based interrupt – stays active until the count is reduced via reading the packet descriptors or the interrupt enable is disabled.

DDR Disable: This bit when set disables the port's DDR controller. The reset / default condition is enabled. To change values for segments the controller should be disabled. The new parameters entered, and then re-enabled to resume operation. It is recommended to reset the DDR after the parameters are changed to make sure the controller starts up within the defined space. See the Start/End registers for more information. *Note: the read-back value is inverted. '1' = enabled and '0' = disabled.*

Time-Code Flags: The time-code flags have been moved to the control register to make room for the latched almost empty/full status bits that were added to the status register. These two read-only bits are currently undefined in the SpaceWire specification and will most likely always be seen as zeros.

Write/Read DMA Ready: These two read-only bits report the DMA state-machine status. If they are read as a one, the corresponding DMA state-machine is idle and available to start a transfer. If the bits are read as a zero, the corresponding DMA state-machine is processing a data transfer.

SpaceWire BK Status Register

Table 34: SpaceWire BK Channel Status Register

SPWR_CHAN_STATUS_0-1	
[0x0004] (status read/latch clear write)	
Data Bit	**Description**
31	Latched Rx FIFO Almost Full
30	Latched Tx FIFO Almost Empty
29-24	Time-Code Data
23	Interrupt Active
22	Receive Packet Length Valid
21	Transmit Purge Error
20	SpaceWire Link Established
19	Read DMA Error
18	Write DMA Error
17	Read DMA List Complete
16	Write DMA List Complete
15	TICK_OUT received
14	Packet Received
13	Receive Error
12	Receive FIFO Overflow
11	Credit Error Detected
10	Escape Error Detected
9	Disconnect Error Detected
8	Parity Error Detected
7	Receive Data Valid
6	Rx FIFO Full
5	Rx FIFO Almost Full
4	Rx FIFO Empty
3	PktCntGt
2	Tx FIFO Full
1	Tx FIFO Almost Empty
0	Tx FIFO Empty

Latched Rx FIFO Almost Full(Bit 31): When a one is read, it indicates that the receive FIFO data count has become greater than the value in the SPWR_CHAN_RX_AFL register. A zero indicates that the FIFO has not become almost full. This bit is latched and can be cleared by writing back to the Status register with a one in this bit position.

Latched Tx FIFO Almost Empty(Bit 30): When a one is read, it indicates that the transmit FIFO data count has become less than the value in the SPWR_CHAN_TX_AMT register. A zero indicates that the FIFO has not become almost empty. This bit is latched and can be cleared by writing back to the Status register with a one in this bit position.

Time-Code Data(29-24): The last time-code value received can be read from this six-bit data-field. The TICK_OUT received status bit will indicate if the data is a new valid time-code value. A time-code

is considered valid if it is one more than the previous stored value. If the time-code is the same as the stored value, it is assumed to be a duplicate resulting from a cycle in the SpaceWire network and is therefore ignored. If the time-code meets neither of these conditions, it is stored, but the TICK_OUT signal is not asserted until the next time-code is received and is one more than that last stored value. At this point, the time-code is deemed to be re-synchronized.

Interrupt Active(23): When a one is read, it indicates that an enabled interrupt condition (other than the DMA interrupts) is active for the referenced channel. A zero indicates that no enabled interrupt condition is active.

Receive Packet Length Valid(22): When a one is read, there is at least one valid receive packet-length value available. When this bit is a zero, it indicates that there are no valid receive packet-length values.

Transmit Purge Error(21): When a one is read, it indicates that a connection error occurred while a transmit packet was in progress and the transmitter was unable to completely delete the remainder of the packet's data from the transmit data FIFO. This can occur if the packet-length the packet data had not been all written to the transmit data FIFO. The link connection state-machine allows a 30 microsecond delay for purging transmit data after a link error. At 132 Mbytes/second this allows almost 4 Kbytes of data to be discarded. This bit is latched and must be cleared by writing the same bit back to the channel status port. A zero indicates that no transmit purge error has occurred.

SpaceWire Link Established(20): A one indicates that the referenced link is in the run state, which allows all types of link and normal characters to be exchanged. A zero indicates that the link is not in the run state.

Read/Write DMA Error(19-18): When a one is read, it indicates that an error has occurred while the corresponding DMA was in progress. This could be a target or master abort or an incorrect direction bit in one of the DMA descriptors. These bits are latched and must be cleared by writing the same bit back to the channel status port. A zero indicates that no DMA error has occurred.

Read/Write DMA List Complete(17-16): When a one is read, it indicates that the corresponding DMA has completed. These bits are latched and must be cleared by writing the same bit back to the channel status port. A zero indicates that the corresponding DMA has not completed.

TICK_OUT Received(15): When a one is read, it indicates that a valid time code has been received by the referenced channel since this bit was last cleared. This bit is latched and must be cleared by writing the same bit back to the channel status port. A zero indicates that a valid time-code has not been received.

Packet Received(14): When a one is read, it indicates that a packet has been received since this bit was last cleared. This bit is latched and must be cleared by writing the same bit back to the channel status port. A zero indicates that a packet has not been received.

Receive Error(13): When a one is read, it indicates that one of the five preceding error conditions has been detected since this bit was last cleared. This bit is latched and must be cleared by writing the same bit back to the channel status port. A zero indicates that no receive error has occurred.

Receive FIFO Overflow(12): When a one is read, it indicates that either an attempt has been made to write data to a full receive data FIFO or a packet-length value to a full packet-length FIFO. Neither of these conditions should occur if the data flow-control protocol is functioning correctly. This bit is latched and must be cleared by writing the same bit back to the channel status port. A zero indicates that no overflow condition has occurred.

Credit Error Detected(11): When a one is read, it indicates that a credit error occurred since the status was last cleared. This bit is latched and must be cleared by writing the same bit back to the channel status port. A zero indicates that no credit error has occurred. A credit error is a violation of the SpaceWire data flow-control protocol.

Escape Error Detected(10): When a one is read, it indicates that an escape error has occurred since the status was last cleared. This bit is latched and must be cleared by writing the same bit back to the channel status port. A zero indicates that no escape error has occurred. An escape error occurs when an escape character is followed by anything other than an FCT or a data character.

Disconnect Error Detected(9): When a one is read, it indicates that a disconnect error has occurred since the status was last cleared. This bit is latched and must be cleared by writing the same bit back to the channel status port. A zero indicates that no disconnect error has occurred. A disconnect error occurs when there is no activity on the data or strobe line for 850 nanoseconds.

Parity Error Detected(8): When a one is read, it indicates that a parity error has occurred since the status was last cleared. This bit is latched and must be cleared by writing the same bit back to the channel status port. A zero indicates that no parity error has occurred.

NOTE: The memory path for the Rx function is IO FIFO => DDR => DMA FIFO. The Tx function is DMA FIFO => DDR => TX IO FIFO. Loop-back reads from the Tx IO FIFO and writes to the Rx IO FIFO.

Receive Data Valid(7): When a one is read, there is at least one word of valid receive data. When data is written to the receive FIFO, the first four words are read and held in batches to be ready for a PCI read DMA or single-word read. Therefore, although the FIFO is empty if this bit is set, there are as many as four additional long-words of receive data. A zero indicates that there is no valid receive data.

Receive FIFO Full(6): When a one is read, the receive data FIFO for the corresponding channel is full; when a zero is read, there is room for at least one more data-word in the FIFO. This bit is a combination of IO and DMA FIFOs full.

Receive FIFO Almost Full(5): When a one is read, the number of data-words in the receive data FIFO for the corresponding channel is greater or equal to the value written to the SPWR_CHAN_RX_AFL register for that channel; when a zero is read, the level is less than that value.

Receive FIFO Empty(4): When a one is read, the receive data FIFO for the corresponding channel contains no data; when a zero is read, there is at least one data-word in the FIFO. This bit is the

DMA FIFO MT. If making sure all data is read it is recommended to use the Rx Count Complete as this includes all three memory elements.

PktCntGt (3) when set '1' means the Rx Packet FIFO has more than the programmed threshold number of entries. This bit can cause an interrupt if the PktCntGtIen bit is also set in the control register. When '0' the level is less than or equal to the programmed level.

Transmit FIFO Full(2): When a one is read, the transmit data FIFO for the corresponding channel is full; when a zero is read, there is room for at least one more data-word in the FIFO.

Transmit FIFO Almost Empty(1): The Almost Empty flag is set when the total count of the Tx chain is below the programmed value.

Transmit FIFO Empty(0): When a one is read, the transmit data FIFO for the corresponding channel contains no data; when a zero is read, there is at least one data-word in the FIFO.

SpaceWire BK FIFO Port

Table 35: SpaceWire BK FIFO Port

SPWR_CHAN_FIFO_0-1	
[0x0008] (R/W)	
Data Bit	**Description**
31-0	Data

These ports are used for single-word accesses into the channel TX FIFO and out of the channel RX FIFO. Reading an Empty FIFO will return the last valid data. Writing to a Full FIFO will go in the bit bucket.

SpaceWire BK Write DMA Pointer Port

Table 36: SpaceWire BK Read DMA Pointer Port

SPWR_CHAN_WR_DMA_PNTR_0-1	
[0x000C] (write only)	
Data Bit	**Description**
31-0	First Chaining Descriptor Physical Address

This write-only port is used to initiate a scatter-gather write [TX] DMA. When the address of the first chaining descriptor is written to this port, the DMA engine reads three successive long words beginning at that address. Essentially, this data acts like a chaining descriptor value pointing to the next value in the chain.

The 1st read retrieve the address of the first memory block (DMA buffer) to write to, the 2nd is the length in bytes of that block, and 3rd is the address of the next chaining descriptor in the linked list (buffer memory blocks). This process is continued until the end-of-chain bit of the next pointer value read indicates it is the last chaining descriptor in the list.

All three values are on LW boundaries and LW in size. Addresses for successive parameters are incremented. The addresses are physical addresses the HW will use on the PCI bus to access the Host memory. In most operating systems you will need to convert from virtual to physical. The length parameter is a number of bytes, and must be on a LW divisible number of bytes.

Status for the DMA activity can be found in the channel control register and channel status register.

NOTES:

1. Writing a zero to this port will abort a write DMA in progress.
2. End of chain should not be set for the address written to the DMA Pointer Address Register. End of chain should be set when the descriptor follows the last length parameter.
3. The Direction should be set to ‘0’ for Burst In DMA in all chaining descriptor locations.
4. The segment size is 31-0 however the byte count is shifted down two positions to create a LW count in memory. This means the actual count is 1G-1 [4Gbytes minus 1 LW] or xFFFFFFFC for the max byte count.

NOTE: The direction bit (bit 1) must be ‘0’ when the physical address of the first chaining descriptor is written to this register or write DMA error will result.

SpaceWire BK Read DMA Pointer Port

Table 37: SpaceWire BK Read DMA Pointer Port

SPWR_CHAN_RD_DMA_PNTR_0-1	
[0x0010] (write only)	
Data Bit	**Description**
31-0	First Chaining Descriptor Physical Address

This write-only port is used to initiate a scatter-gather read [RX] DMA. When the address of the first chaining descriptor is written to this port, the DMA engine reads three successive long words beginning at that address. Essentially, this data acts like a chaining descriptor value pointing to the next value in the chain.

The 1st read retrieve the address of the first memory block (DMA buffer) to write to, the 2nd is the length in bytes of that block, and 3rd is the address of the next chaining descriptor in the linked list (buffer memory blocks). This process is continued until the end-of-chain bit of the next pointer value read indicates it is the last chaining descriptor in the list.

All three values are on LW boundaries and LW in size. Addresses for successive parameters are incremented. The addresses are physical addresses the HW will use on the PCI bus to access the Host memory. In most operating systems you will need to convert from virtual to physical. The length parameter is a number of bytes, and must be on a LW divisible number of bytes.

Status for the DMA activity can be found in the channel control register and channel status register.

NOTES:

5. Writing a zero to this port will abort a write DMA in progress.
6. End of chain should not be set for the address written to the DMA Pointer Address Register. End of chain should be set when the descriptor follows the last length parameter.
7. The Direction should be set to ‘1’ for Burst Out DMA in all chaining descriptor locations.
8. The segment size is 31-0 however the byte count is shifted down two positions to create a LW count in memory. This means the actual count is 1G-1 [4Gbytes minus 1 LW] or xFFFFFFFC for the max byte count.

NOTE: The direction bit (bit 1) must be set when the physical address of the first chaining descriptor is written to this register or read DMA error will result.

SpaceWire BK TX FIFO Partial Count Port

Table 38: SpaceWire BK TX FIFO Data Count Port

SPWR_CHAN_TX_FIFO_COUNT_0-1	
[0x0008] TX FIFO Partial Data Count (read only)	
Data Bit	**Description**
31-16	Number of TX Packet-length Values Stored
15-0	Number of TX Data-Words Stored in IO FIFO

These read-only register ports report the number of 32-bit data words in the Tx FIFOs. The count for the Data corresponds to the data in the IO FIFO – ready to transmit. See the alternate “complete” count port for full Tx chain count.
NOTE: The transmit features are included to support BIT. Actual transmission is not supported for Monitor ports.

The transmit packet-length count field is in this register. This allows the user to know how many packet-length values can be safely written to the packet-length FIFO without causing an overflow. It also aids in error recovery if there was a connection error during a packet transmission and the transmitter is unable to automatically purge the remaining data from the impacted packet. In this case the user must intervene to reestablish a consistent packet/data state by resetting the transmit FIFO and rewriting packet data and packet-length values. Knowing how many packets were pending when the error occurred will greatly aid in this procedure.

SpaceWire BK RX FIFO Partial Count Port

Table 39: SpaceWire BK RX FIFO Data Count Port

SPWR_CHAN_RX_FIFO_COUNT_0-1	
[0x0010] RX FIFO Partial Data Count (read only)	
Data Bit	**Description**
31-16	Number of RX Packet-length Values Stored
15-0	Number of RX Data-Words Stored in DMA FIFO

These read-only register ports report the number of 32-bit data words in the corresponding receive DMA FIFO. There are four additional latches in the read DMA pipeline that may contain data, which allows this value to be a maximum of 0x4003 Refer to the “complete Count” port for the full chain count.

The receive packet-length count field is also in this register. This allows the user to know how many packet-length values are stored in the packet-length FIFO. The first receive packet-length written to the packet-length FIFO is read as soon as it is written to be immediately available. This causes the receive packet-length valid status bit to be set and the packet-length count to become one. As subsequent packets are received, the count will increment and as lengths are read, the count will decrement.

SpaceWire BK RX Packet-Length FIFO Ports

Table 40: SpaceWire BK RX Packet-Length FIFO Ports

SPWR_CHAN_PKT_LEN_0-1	
[0x0014] Packet-Length FIFO Ports (write only)	
Channels with 2 Gbyte Maximum Packet-Lengths	
Data Bit	**Description**
31	Terminate Packet with an Error-End-of-Packet
30-0	Packet Length (31 bits)

These ports access the write transmit packet-length FIFO and the read receive packet-length FIFO ports for the respective channels. These FIFOs are used to store packet lengths for sending transmit packets and reading received packets.

The bit above the packet-length field is an error flag. If this bit is read as a one, it indicates that either the packet was terminated with an EEP (Error-End-of-Packet) or an error condition occurred while the packet was being received. If this bit is written as a one, it indicates that the transmit packet should be terminated with an EEP rather than an EOP.

SpaceWire BK TX Almost Empty Register

Table 41: SpaceWire BK TX Almost Empty Register

SPWR_CHAN_TX_AMT_0-1	
[0x0018] RX Almost Full Level (read/write)	
Data Bit	**Description**
31-0	TX FIFO Almost Empty Level

These read/write ports access the transmitter almost-empty level registers for the respective channels. When the number of data words in the transmit data FIFO is less than this value, the almost empty status bit is set and an interrupt may be generated if it is enabled. While the port is defined larger than the FIFO size in LW, the comparison should be set to a value within the range of the FIFO for proper operation. The comparator uses the full Tx memory chain count for comparison.

SpaceWire BK RX Almost Full Level Register

Table 42: SpaceWire BK RX Almost Full Level Register

SPWR_CHAN_RX_AFL_0-1	
[0x001C] RX Almost Full Level (read/write)	
Data Bit	**Description**
31-0	RX FIFO Almost Full Level

These read/write ports access the receiver almost-full level registers for the respective channels. When the number of data words in the receive data FIFO is greater than this value, the almost full status bit is set and an interrupt may be generated if it is enabled. While the port is defined larger than the FIFO size in LW, the comparison should be set to a value within the range of the FIFO for proper operation. The comparator uses the full Tx memory chain count for comparison.

SpaceWire BK Extended Status

Table 43: SpaceWire BK Status II

SPWR_CHAN_STATUS_II_0-1	
[0x0020] RX Almost Full Level (read only)	
Data Bit	**Description**
31	BurstOut Idle
30	BurstIn Idle
29	DDR Boundary Error Tx
28	DDR Boundary Error Rx
27-21	Spare
20	DDR SM Enabled
19-18	Spare
17-12	Time Code In
11-10	Port Number
9-4	Tx Credit
3	Spare
2-0	FCT Count

Important note: This register was implemented for test and debug purposes only. Brief descriptions of each bit field are provided for reference. Detailed functional behavior is not provided as this register is not expected to be used in normal operation.

Flow Control Token Count: Current number of outstanding FCTs.

Transmit Credit: Number of characters channel is authorized to send - sync'd to PCI clock.

Channel Time Code: Timecode data into Channel – sync'd to PCI clock.

BurstOut and BurstIn Idle = '1' when in the idle state. '0' when busy.

DDR Boundary Error Tx and Rx = '1' when an error has occurred. The HW checks if the Stop address is less than the Start address. '0' is normal status.

DDR SM Enabled is '1' when the controller for the DDR is enabled. Read-back to make sure DDR is operating.

SpaceWire BK RX Packet FIFO Greater Than Level

Table 44: SpaceWire BK RX Packet FIFO Greater Than Level Register

SPWR_CHAN_RX_PKT_GT_CNTL_0-1	
[0x0024] RX Packet FIFO Full Control Register (read/write)	
Data Bit	**Description**
31-10	Spare
9-0	RX Packet FIFO Greater Than Level

These bits are set to 0x3DF on system power-up or reset.

RX Packet FIFO Greater Than Level: These bits set the RX Packet FIFO Greater Than Threshold level. By default, these bits are set to 0x3DF. The RX Packet FIFO has 1023 locations (0x3FF) therefore, by default, the logic will see the FIFO as full once 991 or more locations are occupied. Once the threshold is reached passed the corresponding status bit PktCntGt will be '1'. If the interrupt enable is set the status can cause a level type interrupt – cleared by reading the Packet FIFO until the level is less than or equal to the programmed threshold or disabling the interrupt enable. When operating with high density small packets it is possible to take longer to process the per packet interrupt than the gap between packets. The level-based interrupt can be used to make sure software keeps up since overlapped per packet interrupts can lead to missed packet reads. The threshold may need to be reduced to insure closer to real time processing.

Per the SpaceWire specification the maximum outstanding FCTs is 56. If single byte transfers are being used and the maximum FCTs are outstanding when the FIFO goes full, 56 additional packets maybe received, 0x3DF was chosen to be the default value to cover this case. The hardware automatically asserts back pressure on the link by limiting the available FCTs to the upstream device based on this level.

Setting this register to either 0x3FF or 0x000 puts this logic in legacy mode. Other values can be used to retain protection and optimize for your system based on the minimum packet size [vs. FCTs].

SpaceWire BK TX FIFO Count Complete

Table 45: SpaceWire BK Tx FIFO Count Complete

SPWR_CHAN_TX_FIFO_CNT_COM	
[0x0028] Channel Tx FIFO Count (read only)	
Data Bit	**Description**
31-0	Tx Full Chain Data Stored Count

The Tx Data path includes the DMA FIFO, IO FIFO, and DDR allocated. The count in this register includes the counts from all of these sources. Some counts are based on non-PCI based clocks and need to be converted. The count will be correct when static and may jump slightly when data is actively flowing. Each count represents 1 32 bit word.

SpaceWire BK RX FIFO Count Complete

Table 46: SpaceWire BK Rx FIFO Count Complete

SPWR_CHAN_RX_FIFO_CNT_COM	
[0x002C] Channel Rx FIFO Count (read only)	
Data Bit	**Description**
31-0	Rx Full Chain Data Stored Count

The Rx Data path includes the IO FIFO, DMA FIFO, DDR allocated, and a 4 deep pipeline to support the DMA function. The count in this register includes the counts from all of these sources. Some counts are based on non-PCI based clocks and need to be converted. The count will be correct when static and may jump slightly when data is actively flowing. Each count represents 1 32 bit word.

SpaceWire BK TX DMA Pointer

Table 47: SpaceWire BK Tx DMA PTR

SPWR_CHAN_BURST_IN_PTR	
[0x0030] Channel Monitor ReadFile Current Location (read only)	
Data Bit	**Description**
31-0	Tx Physical Address Pointer

Normally not required, however when debugging DMA operations, it is sometimes helpful to see where the HW is pointing. The DMA operation keeps track of the address being accessed or about to be accessed. Reading this port returns the current address in the BurstIn DMA controller used for the Writefile operation.

SpaceWire BK RX DMA Pointer

Table 48: SpaceWire BK Rx DMA PTR

SPWR_CHAN_BURST_OUT_PTR	
[0x0034] Channel Monitor ReadFile Current Location (read only)	
Data Bit	**Description**
31-0	Rx Physical Address Pointer

Reading this port returns the current address in the Burst Out DMA controller used for the Readfile operation.

SpaceWire BK TX Start Address DDR

Table 49: SpaceWire BK Tx Start Address DDR

SPWR_CHAN_TX_STRT_ADD	
[0x0038] Channel Monitor DDR Transmit Starting Address (read/write)	
Data Bit	**Description**
31-0	Tx DDR Start Address

HW defaults to 32 MegaByte allocations. Read this port to see the starting address for this ports Tx DDR allocation. Write to this port to change the starting address. If changing make sure the DDR is not operating with the DDR disable function, reset and restart. Make sure the boundaries are on page boundaries for efficient operation. Start and End addresses are included in the range

SpaceWire BK TX End Address DDR

Table 50: SpaceWire BK Tx End Address DDR

SPWR_CHAN_TX_END_ADD	
[0x003C] Channel Monitor DDR Transmit Ending Address (read/write)	
Data Bit	**Description**
31-0	Tx DDR End Address

See notes above

SpaceWire BK RX Start Address DDR

Table 51: SpaceWire BK Rx Start Address DDR

SPWR_CHAN_RX_STRT_ADD	
[0x0040] Channel Monitor DDR Receive Starting Address (read/write)	
Data Bit	**Description**
31-0	Rx DDR Start Address

HW defaults to 32 MegaByte allocations. Read this port to see the starting address for this ports Rx DDR allocation. Write to this port to change the starting address. If changing make sure the DDR is not operating with the DDR disable function, reset and restart. Make sure the boundaries are on page boundaries for efficient operation. Start and End addresses are included in the range

SpaceWire BK RX End Address DDR

Table 52: SpaceWire BK Rx End Address DDR

SPWR_CHAN_RX_END_ADD	
[0x0044] Channel Monitor DDR Receive Ending Address (read/write)	
Data Bit	**Description**
31-0	Rx DDR End Address

See notes above

Warranty and Repair

Please refer to the warranty page on our website for the warranty and options that are currently offered.

www.dyneng.com/warranty

Service Policy

Before returning a product for repair, verify to the best of your ability, that the suspected unit is as fault. Then call the Dynamic Engineering Customer Service Department for a Return Material Authorization (RMA) number. Carefully package the product, in the original packaging if possible, and ship prepaid and insured with the RMA number clearly written on the outside of the package. Include a return address and the telephone number of a technical contact. For out-of-warranty repairs, a purchase order for repair charges must accompany the return. Dynamic Engineering will not be responsible for damages due to improper packaging of returned items. For service on Dynamic Engineering products not purchased directly from Dynamic Engineering, contact your reseller. Products returned to Dynamic Engineering for repair by anyone other than the original customer will be treated as out-of-warranty.

Out-of-Warranty Repairs

Out-of-warranty repairs will be billed on a material and labor basis. Customer approval will be obtained before repairing any item if the repair charges will exceed one half of the list price for one of that kind of unit. Return transportation and insurance will be billed as part of the repair in addition to the minimum RMA charge.

Contact:

Customer Service Department
Dynamic Engineering
150 DuBois St. Suite B&C
Santa Cruz, CA 95005
(831) 457-8891
support@dyneng.com

Ordering Information

Industrial Temperature Rated Components: -40 - 85°C

Table 53: Ordering Information

Product	Description	
SpaceWire Monitor:	combination of Interface Module and Software. Capture data from 2 links cross connected through Monitor. DMA transfer to host. Up to 200 MHz link rate. Continuous capture. Two additional SpaceWire ports suitable for IO are provided	
Models:	add as prefix PCI, PCIe, PMC, PCI-104 e.g. **"PCIe-SpaceWire-Monitor"**	
Options:	SpaceWire Cables type AL	50-2004-0801-XX.YY.ZZ XX.YY.ZZ Major.Minor.Units
	SpaceWire Cables type A	50-2020-0801-XX.YY.ZZ XX.YY.ZZ Major.Minor.Units
	-ROHS,	switch to ROHS soldering. Standard is Non-ROHS.
	-CC	Conformal Coating

Glossary

Baud	Used as the bit period when talking about UARTs; Not strictly correct, but is the common usage when talking about UARTs.
CardID	Unique number assigned to a design to distinguish between all designs of a particular vendor
CFM	Cubic feet per minute
FIFO	First In First Out memory
Flash	Non-volatile memory used on Dynamic Engineering boards to store FPGA configurations or BIOS
JTAG	Joint Test Action Group – a standard used to control serial data transfer for test and programming operations.
LFM	Linear feet per minute
LVDS	Low Voltage Differential Signaling
MUX	Multiplexor – multiple signals multiplexed to one with a selection mechanism to control which path is active.
Packed	When UART characters are always sent/received in groups of four, allowing full use of host bus/FIFO bandwidth.
Packet	Group of characters transferred. When the characteristics of the group of characters is known, the data can be stored in packets and transferred as such; the system is optimized as a result. Any number of characters can be transferred.
PCI	Peripheral Component Interconnect – parallel bus from host to this device
PIM	PMC Interface Module (PIM). Provides rear I/O in cPCI systems. Mounts to PIM Carrier
PIM Carrier	PIM Mounting Device. Mounts on rear of cPCI backplane.
PMC	PCI Mezzanine Card – establishes common connectors, connections, size and other mechanical features.
TAP	Test Access Port – basically a multi-state port that can be controlled with JTAG [TMS, TDI, TDO, TCK]. The TAP States are the states in the State Machine that are controlled by the commands received over the JTAG link.
TCK	Test Clock provides synchronization for the TDI, TDO, and TMS signals

TDI	Test Data in – this serial line provides the data input to the device controlled by the TMS commands. For example, the data to program the FLASH comes on the TDI line while the commands to the state machine to move through the necessary states comes over TMS. Rising edge of TCK valid.
TDO	Test Data Out is the shifted data out. Valid on the falling edge of the TCK. Not all states output data.
TMS	Test Mode State – this serial line provides the state switching controls. '1' indicates to move to the next state, '0' means stay put in cases where delays can happen; otherwise, 0,2 are used to choose which branch to take. Due to the complexity of state manipulation, the instructions are usually precompiled. Rising edge of TCK valid.
UART	Universal Asynchronous Receiver Transmitter. Common serialized data transfer with start bit, stop bit, optional parity, optional 7/8 bit data. Can be over any electrical interface. RS232 and RS422 are most common.
Unpacked	When UART characters are sent on an unknown basis requiring single character storage and transfer over the host bus
VendorID	Manufacturers number for PCI/PCIe boards. DCBA is Dynamic Engineering's VendorID
VME	Versa Module European
VPX	Family of standards based on the VITA 46.0
XMC	Switched mezzanine card (PMC with PCIe)