PCI RCI

PCI Bus Remote Camera Interface

Digital Remote Camera Interface Module

008-01091-02

The information in this document is subject to change without notice and does not represent a commitment on the part of Engineering Design Team, Inc. The software described in this document is furnished under a license agreement or nondisclosure agreement. The software may be used or copied only in accordance with the terms of the agreement.

Engineering Design Team, Inc. ("EDT"), makes no warranties, express or implied, including without limitation the implied warranties of merchantibility and fitness for a particular purpose, regarding the software described in this document ("the software"). EDT does not warrant, guarantee, or make any representations regarding the use or the results of the use of the software in terms of its correctness, accuracy, reliability, currentness, or otherwise. The entire risk as to the results and performance of the software is assumed by you. The exclusion of implied warranties is not permitted by some jurisdictions. The above exclusion may not apply to you.

In no event will EDT, its directors, officers, employees, or agents be liable to you for any consequential, incidental, or indirect damages (including damages for loss of business profits, business interruption, loss of business information, and the like) arising out of the use or inability to use the software even if EDT has been advised of the possibility of such damages. Because some jurisdictions do not allow the exclusion or limitation of liability for consequential or incidental damages, the above limitations may not apply to you. EDT's liability to you for actual damages for any cause whatsoever, and regardless of the form of the action (whether in contract, tort [including negligence], product liability or otherwise), will be limited to $50.

Sun, SunOS, SBus, SPARC, and SPARCstation are trademarks of Sun Microsystems, Incorporated.

Windows NT is a registered trademark of Microsoft Corporation.

Intel and Pentium are registered trademarks of Intel Corporation.

UNIX is a registered trademark of X/Open Company, Ltd.

OPEN LOOK is a registered trademark of UNIX System Laboratories, Inc.

Red Hat is a trademark of Red Hat Software, Inc.

IRIX is a trademark of Silicon Graphics, Inc.

AIX is a registered trademark of International Business Machines Corporation.

Xilinx is a registered trademark of Xilinx, Inc.

Kodak is a trademark of Eastman Kodak Company.

The software described in this manual is based in part on the work of the independent JPEG Group.

EDT and Engineering Design Team are trademarks of Engineering Design Team, Inc.

Contents

Tables

Overview

The PCI Bus Remote Camera Interface Module (PCI RCI) is a self-contained unit that connects a 1.25 Gigabit-per-second fiber optic cable to a digital camera, by means of a standard 68-pin high-density connector using RS-422 signal levels. Each PCI RCI module is powered by its own power supply.

The fiber optic interface of the PCI RCI is designed to work with EDT's PCI FOI board installed in the PCI bus of a host computer. The PCI RCI module and the PCI FOI board both have 4 KB FIFOs to facilitate data transfers from the camera to the host PCI bus, and to accommodate short busy periods of other activity on the PCI bus. However, neither has sufficient memory to buffer an entire frame of camera data. Instead, camera data must be passed through the PCI FOI board to main memory on the host computer as fast as the camera presents the data.

Each PCI RCI module can support a camera with pixel rates of up to 30 MHz, and of 8, 10, 12, 14 or 16 bits per pixel. The PCI RCI module does not normally pack bits from two different pixels into one data byte, so a 30 MHz 10-bit camera has a data rate of 60 MB per second.

The PCI RCI supports AIA monochrome, Category 1, extended single-channel digital cameras, as well as a variety of other models. Check the EDT web site at `www.edt.com` for a complete, up-to-date list. Below are listed the most common supported camera models:

- Kodak MEGAPLUS Models 1.4i, 1.6, 1.6i, 4.2i, 6.3i, 16.8i, ES 1.0, XHF, or other AIA monochrome, Category 1, extended single-channel digital cameras.
- Kodak MEGAPLUS Models 1.4 and 4.2
- Cincinnati Electronics IRC-160, IRC 160ST, and IRRIS-256
- Dalsa digital video cameras
- Hamamatsu C4742, C4880, and C4880-81/82
- Xillix Micro Imager 1400 and 1412

Some PCI RCI modules allow you to enable a *region of interest*, a rectangle you can define to crop an image horizontally and vertically, thus eliminating superfluous pixels. To determine whether your module supports this, follow the procedure on page 22.

A single PCI FOI board can support up to four active PCI RCI modules, by cabling them in a ring, as shown in Figure 3. (It is possible to include up to fifteen PCI RCI modules in the fiber optic ring, though DMA channels on the PCI FOI limit the number that can be simultaneously active to four.) The maximum combined data rate for all active PCI RCI modules through the PCI FOI board is typically limited by the host computer's PCI bus performance. On 80*x*86 machines with the LX chipset, and on Sun Ultra 30 workstations, we've seen sustained PCI FOI transfer rates of 66 MB per second. This is sufficient to support four 8-bit cameras with 15 MHz pixel rates simultaneously, or one 16-bit camera with a 30 MHz pixel rate. If your application requires higher data rates or more than four simultaneously active PCI RCI modules, contact Engineering Design Team.

The interface to the camera on the PCI RCI module is implemented with a field-programmable gate array (FPGA). This gate array can be reconfigured to accommodate different cameras. New cameras can be quickly supported. Customer-specific data manipulation is possible; contact EDT for details.

The PCI RCI is not compatible with the SCD FOI or the SDV RCI, comparable EDT SBus products that use Cypress Hotlink fiber optic transceivers instead.

For more information on the PCI FOI, see the *PCI FOI 1.25 GBaud Fiber Optic Interface for PCI Local Bus User's Guide*, EDT part number 008-01089.

The complete PCI RCI system, including the remote unit, the fiber optic interface, and the digital video camera software, is described in three manuals:

- This manual, the *PCI RCI Remote Camera Interface User's Guide*, EDT part number 008-01091, describes the remote camera interface module.
- The *PCI FOI 1.25 GBaud Fiber Optic Interface for PCI Local Bus User's Guide*, EDT part number 008-01089, describes the PCI RCI board in the host computer and the generic software that comes with it.
- The *PCI DV Digital Video Camera Interface User's Guide*, EDT part number 008-00966, describes the software that operates digital video cameras connected to the remote unit. The manual also describes the PCI DV hardware; this portion is irrelevant to the PCI RCI system.

Installation

To install the PCI RCI:

1. Cable the PCI RCI to the PCI FOI board in the host computer, to the camera, and to the power supply as shown in Figure 1.
2. Power up the PCI RCI module and camera.

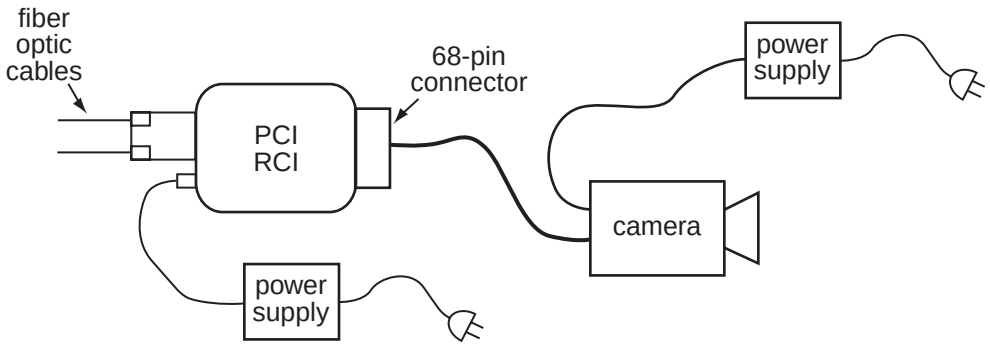

Figure 1. Cabling the PCI RCI

Architecture

Figure 2 shows the architecture of the PCI RCI:

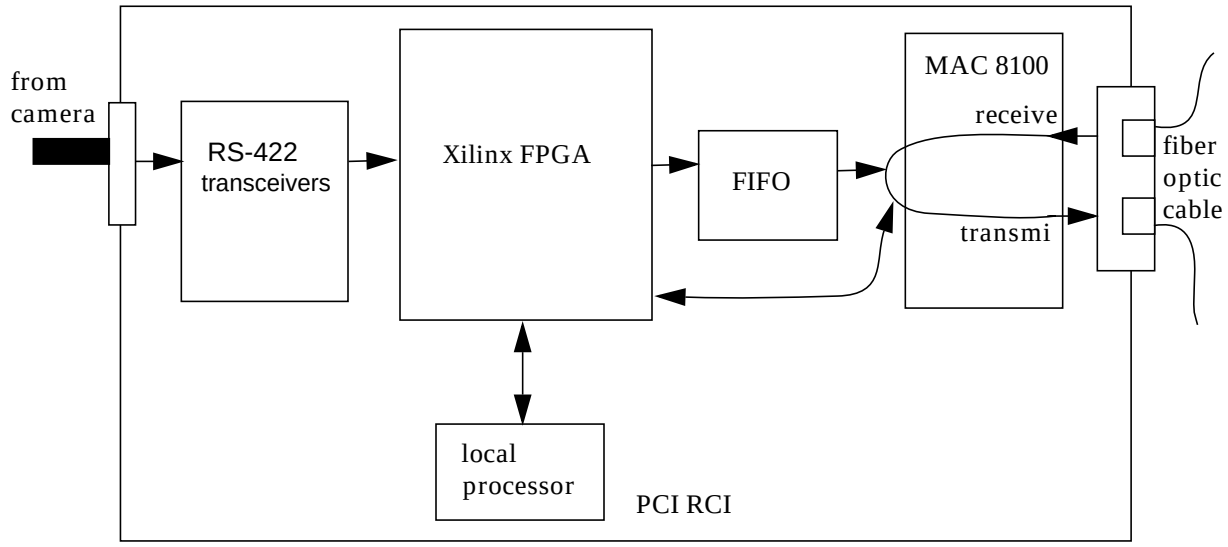

Figure 2. PCI RCI Architecture

Signals from the camera enter the PCI RCI through the 68-pin connector. They first reach the RS-422 transceivers, then travel a 32-bit wide path to the Xilinx field-programmable gate array (FPGA) and from there to the PCI RCI FIFO, which sends them on to the SEEQ 8100 Gigabit Ethernet Media Access Controller (MAC 8100) chip.

Command packets travel a separate 8-bit data path between the Xilinx and the MAC 8100. For more information about command packets, see the section entitled "Command Packet Opcodes" on page 4.

Signals from the PCI FOI or other upstream PCI RCI units enter the fiber optic receiver, loop through the MAC 8100 where the Xilinx FPGA inspects them, and exit the fiber optic transmitter to continue their journey around the ring.

A local processor in the PCI RCI handles the following tasks:

- configuring the Xilinx FPGA when reset,
- loading the MAC 8100 chip registers when reset,
- communicating with the host through the PCI FOI, and
- reading or writing the camera control registers in the Xilinx as instructed by the host.

The PCI RCI FIFO

The FIFO in the PCI RCI buffers the DMA packets for the image data stream between the camera and the MAC 8100 Gigabit Ethernet interface. It does not buffer command packets, register write packets, or packets from previous remote nodes to be passed through the ring to the PCI FOI in the host. When the

Xilinx firmware detects 256 bytes or more in this FIFO, it assembles a DMA packet and sends it to the PCI FOI in the host.

If you use a camera that does not send an even multiple of 256 bytes per frame, the PCI FOI driver must send a FIFO_FLUSH command packet at the end of the frame, in order to flush the remaining data from the PCI RCI FIFO. The packet of flushed data is always 256 bytes long; the packet is padded to that length with undefined data.

Command Packet Opcodes

The software driver on the host computer uses the following command packet opcodes to communicate with the PCI RCI.

NOTE Unless you're writing your own software driver for the PCI RCI, you need not concern yourself with these implementation details. For a complete discussion of command packets, see the the PCI FOI Digital Video Remote Camera Interface User's Guide, EDT part number 008-01089.

Each command packet starts with a routing byte to identify its source and destination; the examples below do not show the routing byte. Following the routing byte is an opcode byte, generally a single ASCII character. In the examples below, the fields labeled <addr>, <data>, and <unit> are integers in ASCII hexadecimal, delimited by spaces. The field labeled <string> is an ASCII string, terminated by the end of the packet.

For example, you can write a value of 0x53 to address 0xC024 with a packet composed of a routing byte followed by the ASCII characters:

```
w C024 53
```

This is called a WRITE command.

The response to most of these commands is a command packet with just a routing byte, referred to as a NULL command packet. In some cases, the response returns requested data. The READ command returns a command packet consisting of a routing byte followed by an integer value represented by four hexadecimal characters, abbreviated below as <data>.

If an error occurs while the PCI RCI attempts to execute the command packet, it sends a command packet in response consisting of a routing byte followed by a #, followed by the opcode byte of the command that failed, followed by a single byte error code. For example, sending a command packet of 9 results in the following response, indicating that the 9 opcode was not recognized:

```
#9?
```

w <addr> <data>

Op-code	Argu-ments	Name	Description	Response
w	<addr> <data>	WRITE	Write to PCI RCI internal address space.	NULL
r	<addr>	READ	Read from PCI RCI internal address space.	<data>
i		INIT	Reset the PCI RCI FIFO and initialize all PCI RCI registers.	NULL

Op-code	Argu-ments	Name	Description	Response
z		ZAP_FIFO	Reset the PCI RCI FIFO.	NULL
f		FIFO_FLUSH	Flush data from the PCI RCI FIFO with final 256-byte DMA packet	NULL
v		VERSION	Request revision level and checksum of PCI RCI code and field-programmable gate array (FPGA) configuration.	4 hexadecimal integers, representing respectively the code version, FPGA version, code_checksum, and FPGA_checksum
m		MORE	Send additional response data back to host, up to 256 bytes maximum.	Next 14 ASCII characters of response to previous command
A		AUTOCONFIG	Turns off automatic pass-through, sets up for UNITASSIGN. The PCI FOI sends this command to unit 15 (the broadcast address) with the DIRIN bit of the routing byte set.	The PCI FOI sees the AUTOCONFIG command looped back
U	<unit>	UNITASSIGN	The PCI FOI sends: U 0 to unit 15 (the broadcast address) with the DIRIN bit of the routing byte set. When the first PCI RCI sees `U 0`, it sets its own unit number, turns on automatic pass-through, then sends to the next PCI RCI: U <unit+1> To give the PCI RCI time to turn off automatic pass-through, the host must wait 10 ms after the AUTOCONFIG opcode returns before sending the UNITASSIGN command.	The PCI FOI receives: U <unit_count> where `unit_count` is the number of PCI RCI modules in the ring.
c	<string>	CAMERA_ COMMAND	Send command string to camera serial port	<camera_response _string>
b		BLINK	Turn on the LED for approximately one second; useful to verify unit number.	NULL

Troubleshooting

When a PCI RCI unit is first powered up, the LED lights for two seconds while the unit initializes itself. It then starts blinking. The blink rate conveys the following information (listed in order of increasing severity):

- One Hz single blink, 6% on:
  The system is operating correctly; the PCI RCI has received at least one command from the PCI FOI host, and is ready for use.
- One Hz double-blink (6% on, 12% off, 6% on, 76% off):
  The PCI RCI sees a link from previous node, but has not yet received an AUTOCONFIG command from the PCI FOI. The PCI FOI software driver on the host computer may not be properly configured. A PCI RCI also enters this state if reset (or powered up) with a fiber optic loopback cable in place.
- Ten Hz blink, on half the time:
  Can't lock to previous node. Check to make sure that the fiber optic cable is properly connected.
- One Hz blink, on half the time:
  The PCI RCI has entered a debug state. (Not normally encountered; call EDT.)
- One Hz blink, 90% on (the LED is lighted 90% of the time, and is dark 10%):
  Xilinx configuration has failed, indicating a PCI RCI hardware problem.

Because the PCI RCIs are connected in a ring, a failure at any node stops the operation of all modules.

If power is interrupted at any PCI RCI or if a fiber optic cable becomes unplugged, operation halts until the fault is corrected. Then the ring resumes operation.

If you add a PCI RCI to the ring or remove one, reinitialize the software driver and expect the unit numbers to change.

A software program is provided to test communication with all PCI RCI modules from the PCI FOI in the host. Invoke it by entering:

```
rcilooptest <pcifoi_unit_number>
```

The argument is the unit number of the PCI FOI, typically 0 if only one PCI FOI board is installed in the host. For example, to test the modules in the ring attached to a lone PCI FOI board, enter:

```
rcilooptest 0
```

Connectors and Cables

The PCI RCI connects to the PCI FOI board with a fiber optic connector and to the camera with a 68-pin high-density connector. These are described below.

The Fiber Optic Connector

Both the PCI FOI board and the PCI RCI module include a fiber optic transceiver using a standard SC Duplex connector. Two versions of the transceiver are available, differing in part by the wavelength of the light emitted and the power level used. These two different transceivers cannot be used together in one fiber optic ring.

The 850 ηm version is less expensive, more reliable, and has a range sufficient for most applications: 300 meters when used with 62.5 / 125 µm multimode fiber. As currently shipped, the 850 ηm version has a black plastic shell around the connector.

The 1300 ηm option provides greater range: when used with 8 / 125 µm single-mode fiber optic cable, it can support links of up to three kilometers. (With 62.5 / 125 µm multimode fiber, its range is 550 meters.) As currently shipped, the 1300 ηm version has a blue plastic shell around the connector.

For more information, see the section entitled "Specifications" on page 27.

When holding the PCI RCI so that the fiber optic transceiver is facing you with the notched part of the shell uppermost, the transmitter is on the left and the receiver is on the right.

CAUTION

To avoid damage to your eyesight, never look straight into the fiber optic transceiver..

If a single PCI RCI module is in the ring with the PCI FOI and the SC duplex fiber optic cable has been built correctly, correct cabling is enforced by the position of the notches. The transmitting side of the fiber optic transceiver in the PCI FOI board is coupled to the receiver on the PCI RCI, and vice versa. The single PCI RCI module is assigned a unit number of 0.

If you need to connect multiple PCI RCI modules to the PCI FOI, then you must use single strand fiber optic cables to create the ring. A fiber optic cable coming fromthe right side of one transceiver must enter the left side of the next transceiver. The PCI RCI that receives its signal from the PCI FOI's transmitter becomes unit 0. Successive PCI RCI modules are assigned successively greater unit numbers, and the final PCI RCI unit transmits directly to the PCI FOI's receiver. This is shown in Figure 3.

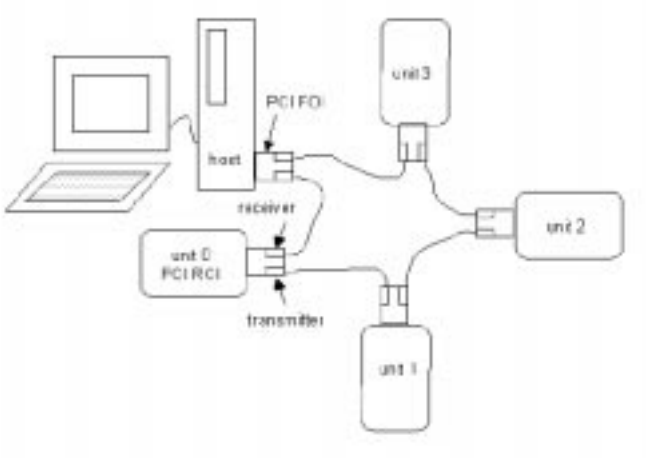

Figure 3. The PCI FOI and Four PCI RCI Modules in a Ring

The Camera Connector

The PCI RCI uses a high-density 68-pin connector to the camera. The digital cameras use a similar connector with the same pinout. The pinout is provided on page 9.

The four mode control outputs and the serial control output to the camera are driven with differential TTL RS-422 compatable drivers. These outputs can also drive single-ended TTL camera control lines. The four mode control pins can be configured as inputs to accommodate a dual-channel camera.

All inputs are received by LVDS-compatible receivers. The low voltage differential signal (LVDS) standard specifies a smaller voltage swing than RS-422 and is much faster, but the common mode range available from the LVDS receivers allows standard RS-422 signals to be received as well.

To connect the camera to the PCI RCI, you can:

- get a cable from the camera manufacturer,
- use an industry-standard SCSI cable, or
- contact EDT and we'll make you one.

Connector Pinout

The PCI RCI uses a high-density 68-pin male cable connector, AMP part number 749621-7, with a shielded backshell (AMP part number 750752-1).

The following pinout diagrams describe the connection from the cable to the camera.

AIA Pin	AIA Signal	PCI RCI Signal	AIA Pin	AIA Signal	PCI RCI Signal
1	Ground	Ground	35	Ground	Ground
2	MSB+	VD0 +	36	MSB –	VD0 –
3	MSB-1 +	VD1 +	37	MSB-1 –	VD1 –
4	MSB-2 +	VD2 +	38	MSB-2 –	VD2 –
5	MSB-3 +	VD3 +	39	MSB-3 –	VD3 –
6	MSB-4 +	VD4 +	40	MSB-4 –	VD4 –
7	MSB-5 +	VD5 +	41	MSB-5 –	VD5 –
8	MSB-6 +	VD6 +	42	MSB-6 –	VD6 –
9	MSB-7 +	VD7 +	43	MSB-7 –	VD7 –
10	MSB-8 +	VD8 +	44	MSB-8 –	VD8 –
11	MSB-9 +	VD9 +	45	MSB-9 –	VD9 –
12	Ground	Ground	46	Ground	Ground
13	MSB-10 +	VD10 +	47	MSB-10 –	VD10 –
14	MSB-11 +	VD11 +	48	MSB-11 –	VD11 –
15	MSB-12 +	VD12 +	49	MSB-12 –	VD12 –
16	MSB-13 +	VD13 +	50	MSB-13 –	VD13 –
17	do not use		51	do not use	
18	do not use		52	do not use	
19	MSB-14 +	VD14 +	53	MSB-14 –	VD14 –
20	MSB-15 +	VD15 +	54	MSB-15 –	VD15 –
21	reserved		55	reserved	
22	Serial Control Out +	SCNTLO +	56	Serial Control Out –	SCNTLO –
23	Serial Cont. In +	SCNTLI +	57	Serial Cont. In –	SCNTLI –
24	Field ID +	FLDID +	58	Field ID –	FLDID –
25	Frame Enable +	FRME +	59	Frame Enable –	FRME –
26	Line Enable +	LINE +	60	Line Enable –	LINE –
27	Channel ID 0 +	ID0 +	61	Channel ID 0 –	ID0 –
28	Channel ID 1 +	ID1 +	62	Channel ID 1 –	ID1 –
29	Pixel Data Strobe +	PSTRB +	63	Pixel Data Strobe –	PSTRB –
30	Mode Control 0 +	FRMRST / EXP +	64	Mode Control 0 –	FRMRST / EXP –
31	Mode Control 1 +	MC0 +	65	Mode Control 1 –	MC0 –
32	Mode Control 2 +	MC1 +	66	Mode Control 2 –	MC1 –
33	Mode Control 3 +	MC2 +	67	Mode Control 3 –	MC2 –
34	Ground	Ground	68	Ground	Ground

Table 1. PCI RCI for Single-channel Grayscale Cameras

AIA Pin	AIA Signal	PCI RCI Signal	AIA Pin	AIA Signal	PCI RCI Signal
1	ground	ground	35	ground	ground
2	AMSB+	VDA0+	36	AMSB–	VDA0–
3	AMSB-1 +	VDA1+	37	AMSB-1 –	VDA1–
4	AMSB-2 +	VDA2+	38	AMSB-2–	VDA2–
5	AMSB-3 +	VDA3+	39	AMSB-3–	VDA3–
6	AMSB-4 +	VDA4+	40	AMSB-4 –	VDA4–
7	AMSB-5 +	VDA5+	41	AMSB-5–	VDA5–
8	AMSB-6 +	VDA6+	42	AMSB-6–	VDA6–
9	AMSB-7 +	VDA7+	43	AMSB-7 –	VDA7–
10	BMSB +	VDB0+	44	BMSB–	VDB0–
11	BMSB-1 +	VDB1+	45	BMSB-1 –	VDB1–
12	ground	ground	46	ground	ground
13	BMSB-2 +	VDB2+	47	BMSB-2–	VDB2–
14	BMSB-3 +	VDB3+	48	BMSB-3 –	VDB3–
15	BMSB-4 +	VDB4+	49	BMSB-4–	VDB4–
16	BMSB-5 +	VDB5+	50	BMSB-5 –	VDB5–
17	do not use		51	do not use	
18	do not use		52	do not use	
19	BMSB-6 +	VDB6+	53	BMSB-6 –	VDB6–
20	BMSB-7 +	VDB7+	54	BMSB-7 –	VDB7–
21	AMSB-8+	VDA8+	55	AMSB-8–	VDA8–
22	Serial Control Out +	SCNTLO+	56	Serial Control Out –	SCNTLO–
23	Serial Cont. In +	SCNTLI+	57	Serial Cont. In –	SCNTLI–
24	Field ID +	FLDID+	58	Field ID –	FLDID–
25	Frame Enable +	FRME+	59	Frame Enable –	FRME–
26	Line Enable +	LINE+	60	Line Enable–	LINE–
27	Channel ID 0 +	ID0+	61	Channel ID 0 –	ID0–
28	Channel ID 1 +	ID1+	62	Channel ID 1 –	ID1–
29	Pixel Data Strobe +	PSTRB+	63	Pixel Data Strobe –	PSTRB–
30	EXPOSE +	EXPOSE+	64	EXPOSE –	EXPOSE–
31	AMSB-9 +	VDA9+	65	AMSB-9 –	VDA9–
32	BMSB-8+	VDB8+	66	BMSB-8 –	VDB8–
33	BMSB-9+	VDB9+	67	BMSB-9–	VDB9–
34	ground	ground	68	ground	ground

Table 2. PCI RCI for Dual-channel Grayscale Cameras

AIA Pin	AIA Signal	PCI RCI Signal	AIA Pin	AIA Signal	PCI RCI Signal
1	ground	ground	35	ground	ground
2	RedMSB-0+	VDR0+	36	RedMSB-0–	VDR0–
3	RedMSB-1 +	VDR1+	37	RedMSB-1 –	VDR1–
4	RedMSB-2 +	VDR2+	38	RedMSB-2–	VDR2–
5	RedMSB-3 +	VDR3+	39	RedMSB-3–	VDR3–
6	RedMSB-4 +	VDR4+	40	RedMSB-4 –	VDR4–
7	RedMSB-5 +	VDR5+	41	RedMSB-5–	VDR5–
8	RedMSB-6 +	VDR6+	42	RedMSB-6–	VDR6–
9	RedMSB-7 +	VDR7+	43	RedMSB-7 –	VDR7–
10	GrnMSB -0+	VDG0+	44	GrnMSB-0–	VDG0–
11	GrnMSB-1 +	VDG1+	45	GrnMSB-1 –	VDG1–
12	ground	ground	46	ground	ground
13	GrnMSB-2 +	VDG2+	47	GrnMSB-2–	VDG2–
14	GrnMSB-3 +	VDG3+	48	GrnMSB-3 –	VDG3–
15	GrnMSB-4 +	VDG4+	49	GrnMSB-4–	VDG4–
16	GrnMSB-5 +	VDG5+	50	GrnMSB-5 –	VDG5–
17	do not use		51	do not use	
18	BluMSB-4+	VDB4+	52	BluMSB-4–	VDB4–
19	GrnMSB-6 +	VDG6+	53	GrnMSB-6 –	VDG6–
20	GrnMSB-7 +	VDG7+	54	GrnMSB-7 –	VDG7–
21	BluMSB-0+	VDB0+	55	BluMSB-0–	VDB8–
22	Serial Control Out +	SCNTLO+	56	Serial Control Out –	SCNTLO–
23	Serial Cont. In +	SCNTLI+	57	Serial Cont. In –	SCNTLI–
24	BluMSB-5+	VDB5+	58	BluMSB-5 –	VDB5–
25	Frame Enable +	FRME+	59	Frame Enable –	FRME–
26	Line Enable +	LINE+	60	Line Enable–	LINE–
27	BluMSB-6+	VDB6+	61	BluMSB-6–	VDB6–
28	BluMSB-7+	VDB7+	62	BluMSB-7 –	VDB7–
29	Pixel Data Strobe +	PSTRB+	63	Pixel Data Strobe –	PSTRB–
30	EXPOSE +	EXPOSE+	64	EXPOSE –	EXPOSE–
31	BluMSB-1+	VDB1+	65	BluMSB-1 –	VDB1–
32	BluMSB-2+	VDB2+	66	BluMSB-2 –	VDB2–
33	BluMSB-3+	VDB3+	67	BluMSB-3–	VDB3–
34	ground	ground	68	ground	ground

Table 3. PCI RCI for Single-channel Color Cameras

Registers

The EDT software driver on the host computer uses the following registers (implemented in the Xilinx FPGA) to control the camera.

NOTE If you're not writing your own driver for the PCI RCI, you need not concern yourself with these implementation details.

As the camera data passes through the Xilinx, the firmware performs various operations on it, in the following order. Use the registers described below to affect the data pipeline as follows:

1. The DUAL_CHAN bit, if set in the ROI Control register, arranges the data bits coming in from the camera to support dual channel cameras such as the Kodak ES 4.0. If not set, data is treated as 16-bit single-channel.
2. If the SWAP_FOR_AIA bit is true in the Shift register, the firmware swaps the 16-bit camera data end for end, so that bit 0 becomes bit 15, bit 1 becomes bit 14, and so on.
3. Barrel-shift the 16-bit camera data 0 to 15 places, as determined by the four least significant bits of the Shift register.
4. If the INVERT_DATA bit is true in the Data Path register, invert the data.
5. Zero any bits for which the 16-bit Mask registers have a value of 0.
6. Apply the FILTER_00_ON or FILTER_F0_ON bits of the Configuration register to clip 8-bit data. If the data represents two 8-bit pixels, both are clipped simultaneously.
7. If the EXT_DEPTH bit of the Data Path register is 0, then only bits 0–7 are sent to the host; bits 8–15 are ignored.
8. If the BSWAP bit of the Utility register is true, swap the two bytes.

Camera Xilinx Registers

Command Register

Size 8-bit

I/O write-only

Address 0x0000.8080

Comments Bits written to this register are strobe bits and need not be cleared after setting. When read, this register is the Firmware ID register, described next.

Bit	Name	Description
7	STROBE_PIXEL	For debug, setting this bit strobes one pixel of camera data (perhaps from the video data hi/lo registers) out to the host.
6-4	not used	
3	CLEAR_CONT	Strobe this bit to clear the CONTINUOUS bit of the Data Path register the next time that the AQUIRE_IP bit in the Status register is true. If AQUIRE_IP is true when this bit is strobed, CONTINUOUS is cleared immediately.
2	AQ_CLR	Resets AQUIRE_INT bit of serial data status register.
1	ENABLE_GRAB	Enable acquisition of the next complete frame. If enabled in the configuration register, the shutter time is started. If the continuous acquisition bit is set in the data path register, then ENABLE_GRAB starts acquisition, which continues until the continuous bit is reset.
0	RESET_INTFC	Setting this bit resets the PCI RCI interface board.

Firmware ID Register

Size 8-bit

I/O read-only

Address 0x0000.8080

Comments When written, this register is the Command register, described above.

Bit	Name	Description
5–7	not used	
4	VDIO	When true, video data lines can be used for output as well as input. (Data comes from Video Data Hi and Lo registers.)
0–3	REV	The Xilinx firmware revision level.

Status Register

Size 8-bit

I/O read-only

Address 0x0000.8081

Comments The executable *watchstat* (included with the PCI RCI software) reads and displays this register symbolically.

Bit	Name	Description
7	AQUIRE_IP	When set, the camera interface has detected a valid beginning of frame and is presently acquiring data.
6	GRAB_ARMED	When set, the grab command has been issued, any specified hardware trigger has been detected, and the camera interface is waiting for a valid beginning of a frame.
5	CHAN_ID1	Reflects state of the AIA Channel Identification 1 signal.
4	CHAN_ID0	Reflects state of the AIA Channel Identification 0 signal.
3	TRIGGER_ARMED	When set, the grab command has been issued, and the board is ready for a hardware trigger. Cleared when AQUIRE_IP goes true unless the AQUIRE_MULT bit of the Utility2 register is set.
2	EXPOSURE	When set, the camera shutter is open.
1	FRAME_VALID	When set, the camera shutter has closed and valid data is being transmitted (though the PCI RCI is not necessarily acquiring data)..
0	OVERRUN	When set, indicates that data was lost during a frame transfer because the host was not ready to receive at the rate at which the camera was transmitting. Therefore the data has been corrupted.

Configuration Register

Size 8-bit

I/O write-only

Address 0x0000.8082

Bit	Name	Description
7	INT_ENAQ	Setting this bit enables the acquisition interrupt, which occurs with the rising edge of the status register's AQUIRE_IP bit. Clear the interrupt with the command register's AQ_CLR bit.
6	EN_DALSA	Setting this bit enables DALSA mode. Maps correct signals and polarities out to MC[0-3] for controlling exposure time with PRIN & external sync on Dalsa area scan cameras..
5	FILTER_00	Setting this bit turns on a filter that linearly maps 8 bit pixel values from 0x00–0x0F up into 0x08-0x0F such that 0x00–0x07 are not used. This saves the lowest eight color palette indexes for use by the window manager.

Bit	Name	Description
4	FILTER_FF	Setting this bit turns on a filter that linearly maps 8 bit pixel values from 0xF0–0xFF down into 0xF0-0xF7 such that 0xF8–0xFF are not used. This saves the highest eight color palette indexes for use by the window manager.
3	FIFO_RESET	Set and clear this bit to reset the PCI RCI input FIFO. (The **z** command packet opcode described starting on page 4 performs this function.)
2	INV_SHUTTER	When set, invert the polarity of the shutter signal. The shutter signal is assigned to a mode control signal in the mode register. When cleared, the mode signal is positive when the shutter is open.
1	TRIG	Obsolete; set to 0. Instead, set the shutter timer to the desired length of the trigger pulse, using the Shutter register and the DECADE bits of the Data Path register.
0	DIS_SHUTTER	Obsolete; set to 0. Disable the EXPOSE signal out to the camera by clearing the four most significant bits of the Mode Control register.

Shutter Register

Size 8-bit

I/O write-only

Address 0x0000.8083

Bit	Description
7–0	Specifies the exposure time –1 (time that the shutter must remain open) in increments of 1 ms, 10 ms or 100 ms: the time base is selected by the DECADE bits in the data path register. If the time base is 1 ms, write 2 for a 3 ms exposure, 4 for a 3 ms exposure, and so on. If the time base is 10 ms, write 2 for a 30 ms exposure.

Table 4. Xilinx Programmable Gate Array Shutter Register

Shutter Time Left Register

Size 8-bit

I/O read-only

Address 0x0000.8083

Comments When written, this is the Shutter register, described above.

Bit	Description
7–0	Specifies the amount of time left before the current exposure terminates.

Data Path Register

Size 8-bit

I/O read-write

Address 0x0000.8086

Bit	Name	Description
7-6	DECADE[1-0]	Selects the time base for the shutter counter DECADE1 DECADE0 Time base 0 0 1 millisecond 0 1 10 milliseconds 1 0 100 milliseconds 1 1 reserved
5	INTERLACED	Set if camera is interlaced. In this case, acquisition waits until the first frame valid when the Field ID signal is true.
4	CONTINUOUS	Set if PCI RCI is to acquire successive frames of data. ENABLE_GRAB in the command register must be set to start acquisition. After CONTINUOUS has been cleared, acquisition continues until the end of a complete frame. Clear in either of two ways: write 0 to this bit to clear it immediately, or strobe the CLEAR_CONT bit of the Command register to clear this bit after the next DMA transfer completes.
3	INVERT_DATA	Inverts the incoming data.
2-1	not used	
0	EXT_DEPTH	Set to send 16 bits to the host for each pixel clock from the camera. Clear to send only the 8 LSB's..

Mode Control Register

Size 8-bit

I/O read-write

Address 0x0000.8087

Bit	Name	Description
7-4	EN_SHUTTER[3-0]	Selects which mode code signal is driven by the internal signal EXPOSE from the shutter timer. If all of these bits are 0, no EXPOSE signal is sent to the camera. If one of these bits is asserted when in continuous mode, the shutter time asserts EXPOSE to the camera once per frame. This bit is ignored if bit 4 (ENMCOUTL) is set in the Utility register.
3-0	AIA_MC[3-0]	If the output is not enabled as a shutter with the EN_SHUTTER bits above, sets the state of the mode control outputs.

Video Data Lo Register

Size 8-bit

I/O read-write

Address 0x0000.8088

Bit	Name	Description
7-0	VD[7-0]	Data written to this register is driven out on the 8 data signal pairs to the camera if the VID_DIRECTION bit of the Utility register is 1. When read, it shows the current state of those 8 lines. Note that VD[0] is the LSB of this register, though the AIA cameras place their MSB on this line. (Not available in all Xilinx configuration files.)

Video Data Hi Register

Size 8-bit

I/O read-write

Address 0x0000.8089

Bit	Name	Description
7-0	VD[15-8]	Data written to this register is driven out on the 8 data signal pairs to the camera if the VID_DIRECTION bit of the Utility register is 1. When read, it shows the current state of those 8 lines. (Not available in all Xilinx configuration files.)

The following three registers implement an asynchronous UART in the Xilinx for communicating with the camera.

NOTE The transmit and receive data lines to the camera are RS-422 differential, not RS-232 levels. Not all cameras use this serial communications channel.

Serial Data Register

Size 8-bit

I/O read-write

Address 0x0000.808A

Bit	Name	Description
7-0	SERIAL_DATA[0-7]	Data written to this register is output on the Serial Control Out signal if the TRANSMIT_RDY bit is set. Data read from this register reflects the last character received when the RECEIVE_RDY bit is set. (See the serial data status register below for descriptions of these bits.)

Serial Data Status Register

Size 8-bit

I/O read-write

Address 0x0000.808B

Bit	Name	Description
7	INTFC_INT	INTFC_INT is set when the following boolean expression is true: ((TRANSMIT_RDY and EN_TX_INT) or (RECEIVE_RDY and EN_RX_INT) or AQUIRE_INT). This expression is then anded with the EN_GLOB_INT bit of the serial data control register and the result passed on to the PCI Xilinx. If the RMT_EN_INTR and PCI_EN_INTR bits of the PCI interrupt and remote Xilinx configuration register in the PCI Xilinx are set (described in the PCIDV user's guide), then an interrupt is asserted over the PCI bus to the host computer.
6	not used	
5	AQUIRE_INT	Set when an acquisition interrupt is enabled through the INT_ENAQ bit of the configuration register, and a rising edge of the AQUIRE_IP bit in the status register has been detected. Clear this interrupt using the AQ_CLR bit of the command register.
4-2	not used	
1	TRANSMIT_RDY	Set when the transmitter is enabled and the holding register is ready for the next character.
0	RECEIVE_RDY	Set when the receiver is enabled and a character is available for reading.

Serial Data Control Register

Size 8-bit

I/O read-write

Address 0x0000.808C

Bit	Name	Description
7-6	BAUD[1-0]	Select serial port baud rate for the serial data: **Bit 7 / Bit 6 / Baud rate** 0 / 0 / 9600 0 / 1 / 19200 1 / 0 / 38400 1 / 1 / 115200
5	CL_RECEIVE_RDY	Set this bit to clear the RECEIVE_RDY bit in the Serial data Status register.
4	EN_GLOB_INT	Global interrupt enable—enables all interrupts.
3	EN_TX_INT	Enables the TRANSMIT_RDY interrupt.
2	EN_RX_INT	Enables the RECEIVE_RDY interrupt.

1	EN_TX	Enables the serial data transmitter.
0	EN_RX	Enables the serial data receiver.

Utility Register

Size 8-bit

I/O read-write

Address 0x0000.808F

Bit	Name	Description
7	VID_DIRECTION	When 1, drive data from PCIDV to the camera over the video data signal pairs. Data value determined by contents of the registers video data hi and video data low. (Not available with some Xilinx configuration files.)
6	SKIP_X	When 1, skip every other pixel in X to scale image down by 2
5	not used	
4	ENMCOUTL	A value of 0 enables Mode Control output; a value of 1 disables it, allowing the four signal pairs labeled MC[0–3] to be used for incoming data. In this case, values set in the Mode Control register are ignored.
3	SSWAP	A value of 1 swaps the order of 16-bit shorts in a 32-bit word of data coming in from the camera, to accommodate host computer byte order.
2-1	PAD[1-0]	Append 0 to 3 extra pixels at the end of each raster. Use to achieve an even number of 32 bit words per raster to optimize processing of the camera data by the host. (For the special Cincinnati Electronics Xilinx configuration file, when PAD0 is set, every 161st pixel is discarded, yielding160 pixels per raster on the IRRIS160ST. PAD1 is ignored.)
0	BSWAP	A value of 1 swaps the order of bytes in a 16-bit word of data coming in from the camera, to accommodate host computer byte order.

Hardware Triggering

Hardware triggering provides a way to send a signal over a wire directly to the PCI RCI indicating when to acquire a frame. When requested by the application, the driver sets the appropriate hardware trigger bits (0–2) in the Utility2 register and is not further involved with hardware triggering.

The host initiates data acquisition from the camera by strobing the ENABLE_GRAB bit of the Command register. What happens next depends on how the registers have been set; Table 5, "Hardware Triggering"

below refers to bit 4 (CONTINUOUS) in the Data Path register (see page 16) and the hardware trigger bits (0–2) in the Utility2 register (starting on page 20).

CONTIN-UOUS	HWTRIG_ CONT	HWTRIGEN_ FLDID	HWTRIGEN_ OPTO	Description
0	don't care	0	0	Acquire the next single frame from the camera.
0	don't care	0	1	Wait for a trigger signal from the user, then acquire a single frame from the camera.
0	don't care	1	0	
1	0	0	0	Acquire all subsequent frames from the camera until CONTINUOUS is cleared.
1	0	0	1	Acquire a single frame from the camera for each trigger signal from the user until the CONTINUOUS bit is cleared.
1	0	1	0	
1	1	0	0	Undefined
1	1	0	1	Wait for a trigger signal from the user, then acquire all subsequent frames from the camera until the CONTINUOUS bit is cleared.
1	1	1	0	
don't care	don't care	1	1	Undefined: specify only one hardware trigger source.

Table 5. Hardware Triggering

To use HWTRIGEN_OPTO (the optical coupler) for the hardware trigger source, call EDT and ask for the optical coupler PCI panel. It uses a standard female DB9 connector. drive the signal into pins 2 and 3 at 5 V, 10 mA; either polarity is acceptable.

To use HWTRIGEN_FLDID (the field ID pins) for the hardware trigger source, drive an RS-422 differential signal into pins 70 (+) and 71 (–) of the camera connector for the PCI RCI (pins 24 (+) and 58 (–) for the PCI RCIK).

Utility2 Register

Size 8-bit

I/O read-write

Address 0x0000.8090

Comments The ENABLE_GRAB bit in the Command register arms the interface for a single hardware trigger. If no hardware trigger is enabled, then the Xilinx triggers immediately and grabs the next frame of data from the camera. If OPTO_TRIGGER is enabled, then the Xilinx waits until a trigger is received through the optical isolator instead. If FIELDID_TRIGGER is true, then the Xilinx waits until a trigger is received from the FIELDID differential pair. If you are using an interlaced camera, the FIELDID pair is not available for this use.

Once triggered, the Xilinx ordinarily acquires a single frame; if the CONTINUOUS bit is set in the Data Path register, then a single trigger grabs all subsequent frames from the camera.

Bit	Name	Description
7	SELECT_MC4	Setting SELECT_MC4 causes the differential pair normally assigned to SCNTLO (Serial Control Out) to be used instead as a fifth mode control bit.

6	MC4	A fifth mode control bit, used by some Photonics cameras as the Photonics SELECT line.
5	PULNIX	True for all Pulnix cameras. Allows EXPOSE to camera to be asserted even when the incoming FRAME VALID line is true; waits till the end of EXPOSE before GRAB_ARMED (bit 6 of the Status register) goes true.
4	DBL_TRIG	Enable double trigger mode on the EXPOSE signal out to the camera. Used for some Pulnix model cameras.
3	not used	
2	AQUIRE_MULT	The ENABLE_GRAB bit in the Command register ordinarily arms the interface for a single hardware trigger. Setting AQUIRE_MULT allows a single write to the ENABLE_GRAB bit of the Command register to enable the acquisition of multiple frames, one frame for each hardware trigger from the optical isolator or FIELDID differential pair.
1	FIELDID_TRIGGER	When true, you can use the FIELDID differential pair to trigger data acquisition from the camera, but only after the START bit of the Command register has been strobed.
0	OPTO_TRIGGER	When true, you can use the optical isolator on the PCIDV to trigger data acquisition from the camera, but only after the START bit of the Command register has been strobed.

Shift Register

Size 8-bit

I/O read-write

Address 0x0000.8091

Comments The default state of zero causes all 16 bits of camera data to pass through unchanged.

Bit	Name	Description
7–6	not used	
5	MARK09	When true, forces a one on bits 0 and 9 of the 16-bit data path after the mask has been applied. The driver can use this to determine how the host orders bytes during DMA. See the discussion below.
4	SWAP_FOR_AIA	When true, the incoming 16-bit camera data is swapped end for end (bit 0 becomes bit 15, bit 14 becomes bit 1, etc.).
3-0	SHIFT[3-0]	A shift value of 0–15, determining how many places to barrel-shift incoming data downward. For example, a value of 4 moves bits 4–15 down to bits 0–11 (and bits 0-3 around to bits 15-12 where they may subsequently be masked off), suitable for many 12-bit cameras.

The PCI Bus is little-endian. Big-endian hosts sometimes swap byte order during DMA to accommodate the PCI Bus, and sometimes do not. To determine host byte order:

1. Clear the BSWAP and SSWAP bits of the Utility register.
2. Clear the Mask Lo and Mask Hi registers.
3. Use the STROBE_PIXEL bit of the Command register to strobe any 16-bit word into the DMA pipeline. (It will be masked out by the zeroes in the Mask registers.)
4. Set the MARK09 bit of the Shift register (above)—the values of all other bits are irrelevant.
5. Use the STROBE_PIXEL bit of the Command register to strobe another 16-bit word into the DMA pipeline.
6. Have the host perform a DMA read operation of one 32-bit word from the PCI RCI.
7. The correct little-endian byte stream for this word is 0x00 0x00 0x01 0x02 (the bits forced to one by the MARK09 bit, above). Determine your result and, if necessary, adjust the BSWAP and SSWAP bits of the Utility register to compensate as required.

Mask Lo Register

Size 8-bit

I/O read-write

Address 0x0000.8092

Bit	Name	Description
7-0	MASK[7-0]	Bits that are zero force the corresponding camera data bit to zero.

Mask Hi Register

Size 8-bit

I/O read-write

Address 0x0000.8093

Bit	Name	Description
7-0	MASK[15-8]	Bits that are zero force the corresponding camera data bit to zero.

Region of Interest

Some PCI RCI boards support a *region of interest*, a rectangle you can define to crop an image horizontally and vertically, thus eliminating superfluous pixels. To determine whether your board supports this capability:

1. Write a 0 to the Region of Interest Control register.
2. Read the register back. The value returned specifies whether region of interest is supported:

0x00 region of interest, simulator, Dalsa line scan, and dual channel cameras supported

0x80 region of interest, simulator, Dalsa line scan, and dual channel cameras *not* supported

0xC4	region of interest, simulator, Dalsa line scan, and dual channel cameras *not* supported; special Xilinx gate array configuration for Irris 256ST and 160ST camera models
other values	undefined

NOTE If region of interest is not supported, then the only part of the ROI Control register available is the bottom three bits, which set the pixel clock rate. The Window registers are also unimplemented unless ROI is supported.

Region of Interest (ROI) Control Register

Size 8-bit

I/O read-write

Address 0x0000.8097

Comments Besides enabling or disabling a region of interest, use this register to enable or disable Dalsa line scan mode and simulator mode. These bits are not all independent: in particular, simulator mode requires compatible settings: you can run with simulated data (bit 4 true) using either internal or camera timing (bit 5 true or false), but if bit 5 is true, enabling internal timing, then you must also set bit 4 true to use simulated data; otherwise, you will see random pixel data from the camera, as the data and timing will be unsynchronized.

Simulated data is generated as follows:

Data bits 0–7 receive the eight least significant bits of the horizontal pixel count. Bit 7 is inverted once each frame. The data at the start of each raster begins with a value of –2 (0xFE) and counts up. Data bits 8–15 receive the eight least significant bits of the vertical line count, starting with 0x00, and count up.

Simulated data has the least significant bit in data bit 0; therefore, do not set the SWAP_FOR_AIA bit of the Shift register when using the simulator.

Bit	Name	Description
7	DALSA_LS	A value of 1 (true) indicates Dalsa line scan mode, in which the horizontal window registers are used to determine the timing of the EXTSYNC and PRIN lines out to the Dalsa camera. If region of interest is also enabled (see bit 6), the region of interest affects only the number of lines per frame as determined by the vertical window registers; the number of pixels per raster is always as specified by the Dalsa camera.
6	ROI_EN	A value of 1 (true) enables region of interest mode, allowing you to crop the image to a rectangular region of interest. The camera transfers to the host only those pixels in the region of interest, as specified by the window registers.

Bit	Name	Description
5	SIM_SYNC	A value of 1 (true) indicates simulator mode. A simulated image of the size specified by the window registers is generated automatically, as if from a camera. To run in simulator mode, also set the SIM_DAT bit to true, and the ROI and DALSA_LS bits to false (0). The pixel clock for the simulator can be either internal or external, as determined by the PCLKSEL bits 2–0.
4	SIM_DAT	A value of 1 (true) indicates that simulated data comes from the window register counters rather than from the camera. You can use simulated data with or without the SIM_SYNC (simulated timing) and PCLKSEL bits.
3	DUAL_CHAN	Set to 1 to enable a data path suitable for dual-channel cameras. Available only for boards that support region of interest capability.
0–2	PCLKSEL	Determines the speed of the clock for the pixel data path:.

Bit 2	Bit 1	Bit 0	Pixel clock rate
0	0	0	from camera
0	0	1	double rate from camera
0	1	0	undefined
0	1	1	undefined
1	0	0	20 MHz
1	0	1	10 MHz
1	1	0	5 MHz
1	1	1	undefined

Window Registers

The window registers are 8-bit write-only registers; each set of two stores a 16-bit integer. Assuming region of interest is enabled, together, the window registers specify the size of the region of interest, as described below. The examples that follow assume a 1024 x 1024 image with a 10-pixel border on all four sides that you wish to crop; thus, a region of interest that is 1004 x 1004.

If you set the SIM_SYNC bit in the ROI Control register, thus enabling simulator mode, and you then write a value to the window registers, the Horizontal and Vertical Skip registers determine the amount of time spent in horizontal and vertical blanking, respectively, and the Horizontal and Vertical Active registers cause the PCI RCI to acquire same number of pixels as would be the case if you'd set the ROI_EN bit, thus enabling region-of-interest mode.

If you've enabled Dalsa line scan mode by setting the DALSA_LS bit, the period of the EXTSYNC pulse generated is determined by sum of the Horizontal Skip and Horizontal Active pixel counts. The amount of time that PRIN is asserted is determined by the value in the Horizontal Skip registers. If region of interest is also enabled, then the vertical counters are used to determine how many rasters to skip and how many rasters to acquire.

For more information about EXTSYNC and PRIN, see the Dalsa documentation.

Horizontal Skip Lo and Horizontal Skip Hi Registers

Size 8-bit

I/O write-only

Address 0x08098 and 0x8099

Comments Load these registers with a 16-bit integer specifying the number of pixels to skip at the start of each line. For example, to skip the first 10 pixels of each line, load these registers with the 16-bit value 0x000A.

Bit	Name	Description
7-0	HSKIP[7-0]	Number of pixels to crop at start of each line.

Bit	Name	Description
7-0	HSKIP[15-8]	Number of pixels to crop at start of each line.

Horizontal Active Lo and Horizontal Active Hi Registers

Size 8-bit

I/O write-only

Address 0x0809A and 0x809B

Comments Load these registers with a 16-bit integer specifying the number of pixels to transfer minus 1, on each line, after the blank pixels have been skipped (as specified in Horizontal Skip registers). For example, to acquire 1004 pixels per line, load these registers with 0x03EB.

Bit	Name	Description
7-0	HACT[7-0]	(Number of pixels to transfer)–1 on each line, after blank pixels are skipped.

Bit	Name	Description
7-0	HACT[15-8]	(Number of pixels to transfer)–1 on each line, after blank pixels are skipped.

Vertical Skip Lo and Vertical Skip Hi Registers

Size 8-bit

I/O write-only

Address 0x0809C and 0x809D

Comments Load these registers with a 16-bit integer specifying the number of lines (rasters) to skip at the start of each frame. For example, to skip the first 10 lines of each frame, load these registers with the 16-bit value 0x000A.

Bit	Name	Description
7-0	VSKIP[7-0]	Number of lines to crop at start of each frame.

Bit	Name	Description
7-0	VSKIP[15-8]	Number of lines to crop at start of each frame.

Vertical Active Lo and Vertical Active Hi Registers

Size 8-bit

I/O write-only

Address 0x0809E and 0x809F

Comments Load these registers with a 16-bit integer specifying the number of lines (rasters) to transfer minus 1, on each line, after the blank lines have been skipped (as specified in Vertical Skip registers). For example, to acquire 1004 lines per frame, load these registers with 0x03EB.

Bit	Name	Description
7-0	VACT[7-0]	(Number of lines to transfer)–1 for each frame, after blank lines are skipped.

Bit	Name	Description
7-0	VACT[15-8]	(Number of lines to transfer)–1 for each frame, after blank lines are skipped.

Specifications

Transceiver options	1.25 gigabaud 1300 ηm laser transceiver for longer distances 1.25 gigabaud 850 ηm VCSEL transceiver for lower cost
Video Interface	68-pin, 16-bit camera interface A programmable gate array configured for a specific camera model

Fiber Optic Cable (not supplied)

1300 ηm laser transceiver (black shell)	*fiber type* 8/125 μm single mode fiber 62.5/125 μm multimode fiber	*distance* 3 km 550 m
850 ηm VCSEL transceiver (blue shell)	*fiber type* 62.5/125 μm multimode fiber	*distance* 300 m

Connectors	AMP 787169-7 high-density 68-pin SC duplex fiber optic connector between fiber optic cable and transceiver
Laser safety	Class 1, eye-safe under a single fault condition

Power supply

AC power supply included	Draws 1 A at 24 V, requires 100–240 V, 50–60 Hz Center conductor is positive, outer shell is negative.
Dimensions	6" x 3" x 1.8"
Weight	15 oz. (6 oz. additional with AC power cord)

Environmental

Temperature	Operating: 10 to 40° C Nonoperating: –20 to 60° C
Humidity	Operating: 20 to 80% noncondensing at 40° C Nonoperating: 95% noncondensing at 40° C

Physical

Dimensions	5.6" x 2.7" x 1.3" Camera cable radius of curvature: 6" Fiber optic cable radius of curvature: 3" If installing in closed area, minimum length is nearly 15" (5.6 + 6 + 3).
Weight	12 oz.

Table 6. PCI Bus Remote Camera Interface Module Specifications

Index

Z