FASTSERIES

FAST I/O HARDWARE
USER'S MANUAL

FAST CAPTURE
FAST PROCESSING
FAST RESULTS

FASTSERIES PCI BOARD

FastVision
FastImage 1300
FastFrame 1300

FAST SERIES PMC

FastMem
Fast4 1300
Fast I/O 1300

30002-00153

COPYRIGHT NOTICE

Copyright ã 2002 by Alacron Inc.

Document Name	Fast I/O Hardware User's Manual
Document Number:	30002-00153
Revision History:	1.5 July 2, 2002

Trademarks:

All trademarks are the property of their respective holders.

Alacron Inc.
71 Spit Brook Road, Suite 200
Nashua, NH 03060
USA

Telephone: 603-891-2750
Fax: 603-891-2745

Web Site: *http://www.alacron.com/*

Email: *sales@alacron.com* **or** *support@alacron.com*

TABLE OF CONTENTS

Copyright Notice ..... *ii*
Table of Contents ..... *iii*
Manual Figures & Tables ..... *v*
Other Alacron Manuals ..... *vi*

I. INTRODUCTION ***1***

A. FastIO **1**

B. Processors and Memory **2**

C. Analog and Digital Input **2**
1. Front End 2
2. Digital Input 2
3. Analog Input 2
4. Digital Control Output 2

II. FUNCTIONAL DESCRIPTION ***3***

A. Processors and Memory **3**
1. TriMedia Processors 3
2. Video Data Paths 4
3. Digital Crosspoint Switch 5
4. Interrupts 5
5. Front End Controls 5

B. Analog Video Input **5**
1. NTSC/PAL Composite Analog Video Input 6
2. Monochrome and RGB Analog Video Input 6

C. Digital Input **7**
1. Digital Data Inputs 7
2. Digital Control Inputs 7
3. Camera Controls 7
4. General-Purpose Outputs 9

D. FPGA-Based Front End **9**
1. Data Reordering 9
2. Data Valid Signal 9

E. CPLD **10**

F. UART **10**

G. Power **10**

III. PROGRAMMING, CONFIGURATION AND TEST ***11***

A. I2C Bus **11**

B. V.34 SSI Interface (TM1 to CPLD) **11**
1. Serial Communication Port (UART) 12
2. FPGA Serial Programming 12

C. JTAG Chains **13**
1. JTAG Chain #2 - TriMedia Processors 13

D. JTAG Chain #3 – CPLD, FPGA, and I-Cube IQ32B **13**

IV. CPLD Specification ***14***

A. V.34 SSI Interface **14**

B. JTAG Interface **15**
C. External Five-Wire Interface **16**
D. PCI Bus Secondary Clock Mask **16**
E. Frame Timing Functions **16**
F. BT261 Parallel Bus Interface **17**
G. CPLD Programming Requirements **17**
V. UART Specification ***20***
A. Purpose **20**
B. Features **20**
C. I^2C Address **20**
D. Interrupts **20**
E. Register descriptions **21**
1. Subaddress 00, Write – UART Control Register 21
2. Subaddress 00, Read - UART Status Register 22
3. Subaddress 01, Write – Transmit Data Register 22
4. Subaddress 01, Read – Receive Data Register 23
5. Subaddress 02, Write – Ports A / B latch 23
6. Subaddress 02, Read – Ports A / B Pins 23
7. Subaddresses 03-FF – PIC Register File 23

VI. CONNECTORS AND JUMPERS ***25***
A. Connectors **25**
1. Rear Bracket Connectors 25
2. Analog and Digital Input Connector (J1A/J1B) 25
3. RS232 I/O (J2) 26
4. PMC Slot (PN4) 27
5. Power 28
6. Test Connectors 28

B. JUMPERS **28**
1. P2 28
2. P1 29
3. P5 29

VII. FASTSERIES CABLES ***30***
A. Cable Summary **30**
B. Power Cable **30**
C. Digital Input Cable **30**
D. Analog Input Cable **32**
VIII. SPECIFICATIONS ***34***
A. Processors **34**
B. NTSC/PAL Composite Video Capture **34**
C. Analog Video Capture **35**
D. Digital Video Capture **35**
E. Camera Control **35**

F. PCI Interface **35**

G. PMC Interface **36**

IX. TROUBLESHOOTING ***37***

X. ALACRON TECHNICAL SUPPORT ***38***

A. Contacting Technical Support **38**

B. Returning Products for Repair or Replacements **39**

C. Reporting Bugs **40**

MANUAL FIGURES & TABLES

FIGURE	PAGE	SUBJECT	TABLE	PAGE	SUBJECT
1	1	Block Diagram of the Fast I/O Daughter Card	1	6	Composite Analog Video Inputs
2	3	Processor & Memory	2	6	8-Bits Analog Inputs
3	4	Message Passing Connection	3	18	CPLD Control Register Bits 1,0
4	5	TriMedia Interrupts	4	13	BT261 Control Register Bits 6,5
5	8	Camera Control Output Signals	5	26	Analog & Digital Input Connectors J1A/J1B
6	11	Configuration and Control	6	26	RS232 Input/ouput ConnectorJ2
7	15	V.34 Interface Timing	7	27	PMC-PN4 Connector
8	17	Frame Timing Signals	8	28	Test Connector P3
9	25	Fast I/O Rear Panel Bracket	9	28	Test Connector P4
			10	29	Jumper P2
			11	29	Jumper P1
			12	29	Jumper P5
			13	30	Digital Input Cable – P2 Connector
			14	31	Digital Input Cable – P3 Connector
			15	32	Analog Input Cable

OTHER ALACRON MANUALS

Alacron manuals cover all aspects of FastSeries hardware and software installation and operation. Call Alacron at 603-891-2750 and ask for the appropriate manuals from the list below if they did not come in your FastSeries shipment.

- 30002-00148 ALFAST Runtime Software Programmer's Guide & Reference
- 30002-00150 FastSeries Library User's Manual
- 30002-00155 FastMem Hardware User's Manual
- 30002-00162 FOIL – **F**astSeries **O**bject **I**maging **L**ibrary User's Manual
- 30002-00169 ALRT Runtime Software Programmer's Guide & Reference
- 30002-00170 ALRT, ALFAST & FASTLIB Software Installation Manual for Linux
- 30002-00171 ALRT, ALFAST, & FASTLIB Software Installation for Windows NT
- 30002-00173 FastMem Programmer's Guide & Reference
- 30002-00176 FastImage 1300 Hardware User's Manual
- 30002-00180 Fast4/1300 Hardware User's Manual
- 30002-00184 FastSeries Getting Started Manual
- 30002-00185 FastVision Hardware User's Manual
- 30002-00186 FastVision Software User's Manual
- 30002-00187 FastFrame Hardware User's Manual

I. INTRODUCTION

A. FastIO

The FastIO is an input and processing daughter card that can process continuous video streams with minimal impact on the main FastImage or FastFrame board, or that can share data and processing with the main board system. The FastIO system consists of a main board with one DSP processor and associated memory and analog or digital input.

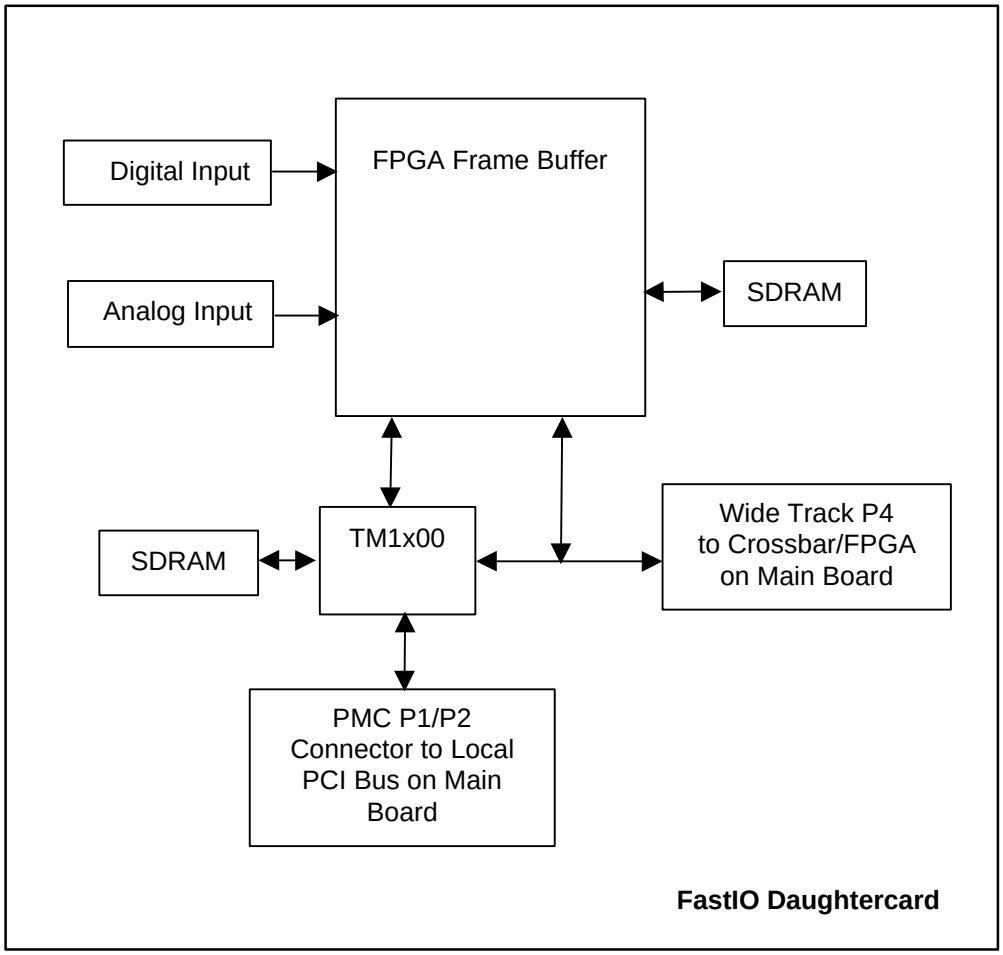

Figure 1. Block Diagram of the FastIO Daughter Card

B. Processors and Memory

The FastIO includes one TM1100 DSP processor with 8MB or 16MB of SDRAM.

C. Analog and Digital Input

The daughter card has four inputs, TAP1 through TAP4; each TAP (8 bits data, 2 bits control, pixel clock) can be configured for analog input or for digital input (but not both). TAP1 through TAP3 can be populated as either digital input or RS170 compatible analog input. TAP4 can be either digital input or composite analog input (NTSC/PAL/SECAM or S-Video). All four TAPs are configured for digital or for analog, rather than mixing inputs.

NOTE: Although audio data can be handled by the TriMedia processors, the FastIO board does not support audio input or output.

1. Front End

The Front End components for both analog and digital input options are:

- 68-pin double row input connector (J1A/J1B)
- Two input FPGAs (Xilinx XCS20/30/40PQ208), each with 8MB SDRAM frame buffer.

2. Digital Input

When installed, the Digital input option provides 32 bits of RS422 differential digital inputs, terminated into 100Ω.

3. Analog Input

When installed, the Analog input option includes these components:

- SA71111 Video Input Processor
- Three eight-bit analog input channels with A/D
- RS422 digital video control and clock inputs (for analog line scan or other non-composite sync cameras).
- RS422 digital camera/strobe control and clock outputs (for analog line scan or other non-composite sync cameras).

4. Digital Control Output

The daughter card provides RS422 digital video control and clock outputs

II. FUNCTIONAL DESCRIPTION

A. Processors and Memory

The processors and memory subsystem is diagrammed in Figure 2.

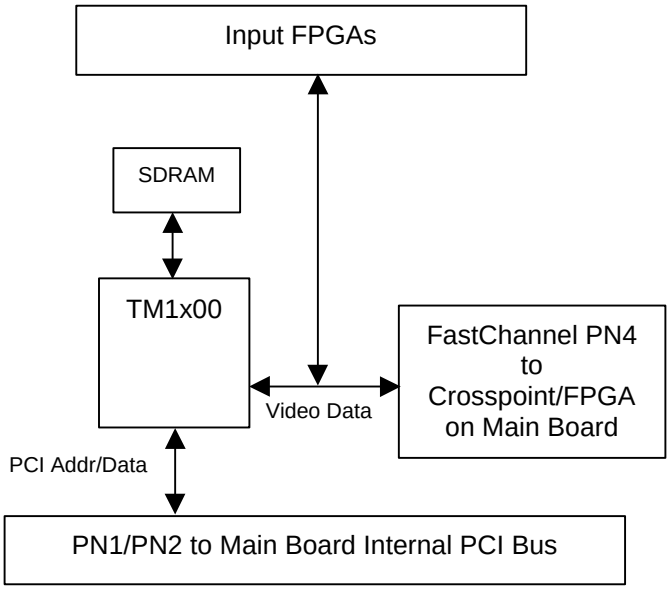

Figure 2. Processors and Memory

1. TriMedia Processors

One TriMedia TM1100 processor is installed on the FastIO card. It has built-in PCI bus interface and glueless interface to local Synchronous Dynamic Random-Access Memory (SDRAM) for program and data. The TM1100 processor can have 8 or 16MB SDRAM.

The heart of the TriMedia is a VLIW digital signal processor which can issue up to 5 instructions in a single clock cycle for 500 peak MIPS (@ 100 MHz). Special DSP instructions allow simultaneous operation on four 8-bit or two 16-bit operands in a single instruction. This pushes the peak rate to 2 billion operations per second for 8-bit data.

The TriMedia can issue up to four floating point operations per clock cycle for a peak rate of 400 MFLOPS. The TriMedia does not have floating point multiply/accumulate instructions, but it can issue up to two floating point adds and two floating point multiplies in a single clock cycle.

The TriMedia has 8 or 16 megabytes of local SDRAM running at 120 MHz. The SDRAM is accessible from the Host PCI bus (via the bridge) to allow the host to download programs, and (directly) from the secondary PCI bus to allow the TriMedia to transfer data to each other at full PCI bandwidth. All programs to be run in the TriMedia processor must reside in local SDRAM.

TriMedia processors have built-in Audio Input and Output interfaces. These synchronous serial ports were intended for stereo digital audio processing, but can be used as general purpose block DMA devices. No connection is made to these ports on the FastIO.

The TriMedia has a small serial EEPROM, which contains configuration data (not shown in diagram). Additional configuration data must be loaded by the system BIOS (memory base addresses) and by the run-time software. After a power-on or PCI bus reset, the TriMedia loads clock and ID parameters from the EEPROM. It then stays in a reset state waiting for the host to finish configuration. Since the DSP CPU will not run until the Host completes the configuration process, individual TriMedia may be run stand-alone for debug or test. The DSP CPU reads all instructions from the local SDRAM. After a reset, it begins operation starting at the first location in SDRAM. Thus the host is required to load code into each SDRAM before releasing the reset state of the DSP CPU. Standard PCI and “Plug and Play” requires the host CPU to assign address bases and other parameters at start-up.

2. Video Data Paths

Using the input FPGA, the TriMedia can receive video input from the SAA7111A video input processor or from an external analog or digital video source. Data can also be received directly from the PMC PN4 FastChannel™ connector to the main board.

Each TM1100 has byte-wide video input and output ports. TriMedia video input ports accept ITU-R BT.656 (ITU recommendation - broadcast television 656, formerly known as CCIR656) encoded 8-bit color data as well as 8- and 10-bit raw data (with sign or zero extend to 16-bits for 10-bit input). When receiving color data in ITU-R BT.656 mode, the incoming data is automatically broken into three components and stored as separate arrays for Y Cr and Cb. The maximum video input clock rate is 38 MHz. TriMedia video output ports can generate ITU-R BT.656 encoded data streams as well as 8-bit raw data. A special "message passing" mode allows video outputs to connect to video inputs using 8 data bits along with start and stop message bits. All video input and output data is passed between the port and the TriMedia local SDRAM.

The FastIO board can also use the Video In and Video Out units for message-passing mode. The Video Data bus and control signals are routed to the P4 PMC connector, which connects to the crossbar on the processor board (Figure 3). From that crossbar, the signals can be linked into the chain of TriMedia on the processor board if desired.

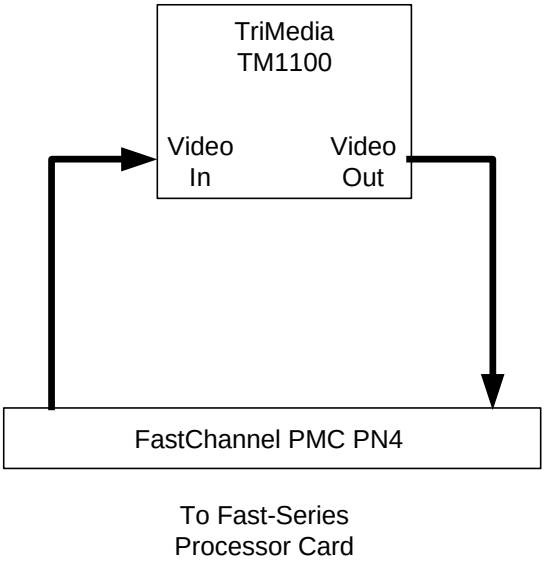

Figure 3. Message Passing Connections

3. Digital Crosspoint Switch

An IQ32B digital crosspoint switch is used for flexible routing of clock signals with low latency and skew. The IQX32B is a 32-port non-blocking switch matrix with flexible I/O buffers at each port. The single-bit ports allow the maximum flexibility of interconnection including one-to-many and many-to-one modes. The I/O buffers resemble the I/O macro cells of a gate array or CPLD. The switch matrix is capable of connecting any I/O port to any other I/O port on the device.

Configuration of the switch matrix and I/O buffers is SRAM-based, allowing reconfiguration in system. Configuration data is downloaded to the part via a standard JTAG port. Because of the relatively slow rate of configuration, the crosspoint switch is used as a static interconnect for most applications. The JTAG bit stream data can be generated by tools available from I-Cube from a simple configuration file.

4. Interrupts

The TriMedia processor receives the Host interrupt via a line from the PCI interface.

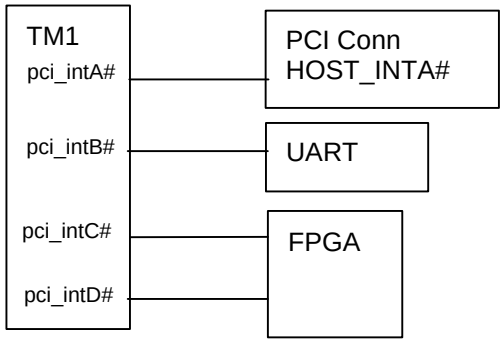

Figure 4. TriMedia Interrupts

On the PMC slot, HOST_INTA# connects to PCI_INTA#, B to B, C to C, and D to D. Thus, when the daughter card processor interrupts on PMC interrupt line A, it gets back to the Host on the Host's interrupt line A, and so forth.

5. Front End Controls

The TriMedia processor can communicate with the input FPGA via the internal I^2C bus. The TriMedia can send start, stop, or other messages to the FPGA via this bus. The I^2C bus is discussed further in Programming, Configuration and Test below.

B. Analog Video Input

Analog video capture signals come in through a dual 68-pin VHDCI connector. Four RS-170 level video inputs allow selection from four composite video sources. The same inputs can also be used in pairs to receive S-video signals.

Three additional RS-170 level video inputs can be used to capture images from three additional monochrome cameras or one RGB camera. Sync detection circuitry for these inputs is very flexible, allowing the use of asynchronous reset cameras as well as interlaced or progressive area scan cameras and line scan cameras.

1. NTSC/PAL Composite Analog Video Input

A Philips SAA7111A Enhanced Video Input Processor digitizes composite or S-video from any source adhering to the NTSC, PAL or SECAM standards for 525-line 59.94 Hz and 625-line 50 Hz video. The chip has four inputs (Table 1), and can select one of four composite video sources (one per input) or two S-video sources (Y on VID1, UV on VID3 or Y on VID2, UV on VID4). These inputs are multiplexed into a common converter (two converters in S-video mode). The TriMedia controls the input selection using the I^2C bus to access the internal registers of the SAA7111A.

The SAA7111A outputs 8-bit parallel digitized video encoded per ITU-R BT.656, which can be directly acquired by the TriMedia processors. The pixel rate is phase locked to the horizontal scan rate of the input image and is nominally 13.5 MHz. Since each pixel requires 16 bits of data, the data output clock rate is 27 MHz. Color video requires three values per pixel. NTSC and PAL use Y (luminance) Cr (red portion of chroma) and Cb (blue portion of chroma). For eight bit resolution of each component, 24 bits per pixel would be required. The pixel size of 16 bits (rather than 24 bits) is realized by sub-sampling the chroma portion of the input signal per the ITU-R BT.601, 13.5 MHz 4:2:2 encoding standard.

J1B Pins	J1B Signal	Analog In to EVIP	EVIP Signal
58, 59	TAP4_D0, GND	VID1	AI11
56, 57	TAP4_D1, GND	VID2	AI12
54, 55	TAP4_D2, GND	VID3	AI21
52, 53	TAP4_D3, GND	VID4	AI22

Table 1. Composite Analog Video Inputs

2. Monochrome and RGB Analog Video Input

Three 8-bit A/D converters are available for three channels of RS170 compatible monochrome video or one RGB video source in parallel with the EVIP. Each channel has associated sync detection and pixel clock generation to allow simultaneous acquisition from three independent (non-genlocked) sources. Sync and pixel clocks may also be driven from an external RS-422 source. Independent offset and gain controls are available for each of the three channels.

Pins	**Signal**	**Timing**	**ADC (U35)**
J1A-10, J1A-11	TAP1-D0, GND	BT1 (U38)	Channel A
J1A-58, J1A-59	TAP2-D0, GND	BT2 (U39)	Channel B
J1B-10, J1B-11	TAP3-DO, GND	BT3 (U37)	Channel C

Table 2. 8-Bit Analog Input Channels

Output from the A/D converters can be further processed in the FPGA-based digital front end.

C. Digital Input

Video capture signals come in through a dual 68-pin VHDCI connector (J1A/J1B). To save pins, analog and digital signals share pins allowing only one or the other for each of the four input taps. Digital video input lines allow direct connection of digital line-scan or area-scan cameras of up to 32 bits.

All digital video data and control signals are differential RS-422 level signals with 100-ohm termination for twisted pair cable. High speed line drivers and receivers are used on all digital signals, however the RS-422 standard was not designed for very high data rates. Thus the interface circuitry may limit the maximum practical data rate to somewhat less than the 38 MHz video input bandwidth of the TriMedia chips. Industry standard pin-out devices are used for all RS422 receivers and drivers (26LS31 and 26LS32 footprint) to allow upgrade to faster parts as required. LVDS interface drivers and receivers are available as selective stuffing options.

1. Digital Data Inputs

Thirty-two digital data inputs from connector J1A/J1B are received by high-speed RS422 differential receivers (terminated into 100Ω) whose outputs are run through the FPGA-based front end. The FPGA can then route these to the data inputs of the TriMedia processors.

2. Digital Control Inputs

Four clock inputs (TAP*n*_PIXCK) come in through the Digital input connector J1A/J1B. These inputs are received by high-speed RS422 differential receivers. The outputs of these receivers go to dedicated global clock inputs (PIX_CLK*n*) of the FPGAs. PIX_CLK1 through PIX_CLK3 go to the FPGA. In addition, all four signals go a crosspoint switch (IQ32B). Programmable clock polarity and input delay allow compensation for clock to data skew.

Eight additional control inputs (TAP*n*_LVAL and TAP*n*_FVAL) come in through the Digital input connector J1A/J1B. These inputs are received by high-speed RS422 differential receivers. The outputs of these receivers (LVAL*n*, FVAL*n*) go to the FPGAs (4 to each FPGA). These lines may be used for frame valid and line valid signals when attaching multiple cameras.

Six general purpose static inputs (GPIN*n*) are provided. These inputs are received by RS422 differential receivers. The outputs of these receivers go to the UART, where they can be read by the primary TriMedia via the I^2C bus.

3. Camera Controls

Camera Controls are outputs sent back through the digital input connector J1A/J1B. Three kinds of controls are available: start and exposure signals, master clocks, and general purpose control signals.

a) Frame/Line Start and Exposure

Four strobe lines, STROBE1-STROBE4, are output to connector J1A/J1B. These RS422 differential outputs are for line scans or area scans cameras, which require a scan, start pulse, readout, or exposure control signals (e.g., EXSYNC and PRIN to Dalsa cameras).

The CPLD has two counters for generating line/frame start and exposure timing signals (Figure 5). These 9-bit counters run at 10 KHz, allowing timing from 100 microseconds to 50 milliseconds. The counters are triggered by the rising edge of one of the two EXT_TRIG*n* inputs from J1A/J1B. Upon triggering, they count up from zero and stop when they reach maximum count (511). Two 9-bit compare registers, COMP1 and COMP2, are associated with each counter. These are programmed to create timing events from 1 to 512 clock cycles after the trigger. In normal usage the value of COMP1 is less than that of COMP2. Four output signals per counter, Cmp1, Cmp1Pls, Cmp2, and Cmp2Pls, are run to the IQX32B where they can be selected to run to the four strobe lines. Signals Cmp1Pls and Cmp2Pls are active high pulses, 100 microseconds wide, beginning at the times programmed in COMP1 and COMP2. Cmp1 is an active high pulse, which starts at the trigger and ends at the time programmed into COMP1. Cmp2 is an active high pulse starting at the time programmed into COMP1 and ending at the time programmed into COMP2 (note that if COMP2 is less than COMP1 this signal will begin during one trigger cycle and end during the next). The counters cannot be retriggered while running until the time programmed into COMP2 has elapsed.

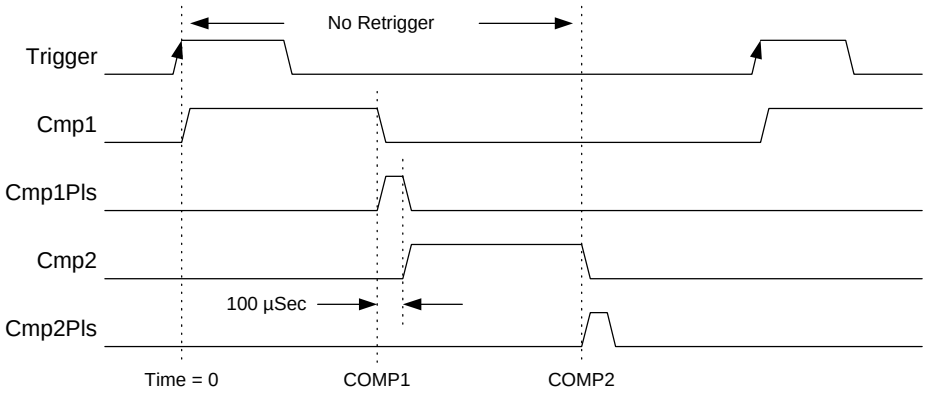

Figure 5. Camera Control Output Signals

The camera control signals provide a flexible control interface. In a typical application, a Cmp1 signal could be used to start camera exposure. The corresponding Cmp2 signal could initiate transmission of the camera data to the FastIO.

The four external strobe signals on connector J1A/J1B, STROBE1 through STROBE4, can output any four of the eight camera control signals, depending on the crosspoint switch programming.

b) Master Clocks

Four master clock outputs (MASTER_CK*n*) to connector J1A/J1B are provided to generate a time base for cameras. These RS422 differential outputs are intended for use with line scan or area scan cameras, which require an external time base. They are generated by the primary and secondary TriMedia using the AI_OSCLK outputs. Only one of these signals is available on the single processor model. These pins can generate any frequency from 1 Hz to 40 MHz in .07 Hz steps using the direct digital synthesizer of the TriMedia. Nominal jitter on these outputs due to digital synthesis is 3.3 nanoseconds. This will be reduced to less than 1 nanosecond in the TM1100 when the improved mode is used. In TM1100 improved mode, the frequency resolution is 0.3 Hz.

4. General-Purpose Outputs

Four general purpose static outputs (GPOUT0-3) are provided to connector J1A/J1B. These RS422 differential outputs can be written out through the UART by the primary TriMedia (via the I^2C bus).

D. FPGA-Based Front End

Digital data inputs pass through an FPGA, Xilinx XCS20PQ208, XCS30PQ208, or XCS40PQ208 with external SDRAM, which can perform multiplexing and re-order incoming data for Odd/Even and Left/Right tapped cameras. Data is divided into two 16-bit halves, one to each FPGA. When using three or four tap cameras, half the bits from each tap are routed to each FPGA, allowing the two parts to operate identically. When using multiple smaller cameras, each FPGA services a different camera or pair of cameras. These FPGAs are in-system reprogrammable, allowing application-specific operations to be performed. Standard download files will be generated for this part to cover the most popular cameras, including quad single tap and dual two tap camera systems. Additional configurations can be generated to customer requirements.

1. Data Reordering

Data from multitap cameras is re-ordered using the SDRAM. Incoming data is formed into groups of 2 32-bit words inside each FPGA and written to the SDRAM location corresponding to its desired position in the output image. When a complete image has been buffered in the SDRAM, it is read out to the TriMedia's while the next image is buffered in the other half of the SDRAM. Four pairs of words are read then four pairs written to reduce the buffering requirements of the FPGA while keeping the SDRAM running near its optimal burst rate.

2. Data Valid Signal

A data valid signal is generated by the FPGA for the TriMedia and also sent to the P4 connector for use by the processor board. The signal is synchronous to the video input clock of the TriMedia. The signal connects to the VI_DVALID pin of the TriMedia via the IQX32B to match the delay time of the data lines. The purpose of the data valid signals is to synchronize the data capture to the valid pixels provided by the camera when using raw captures modes.

Another purpose of the data valid is to pad (or truncate) the frame to a multiple of 64 pixels. This allows use of cameras with non-binary height and width. The 64 pixel limitation comes from the TriMedia, which must define buffers in 64 byte increments. Providing dummy (garbage) pixels at the end of the frame to fill out the buffer greatly simplifies programming in TriMedia raw capture modes. Padding is accomplished with a 6-bit (modulo 64) counter, which runs only when data valid is active. If the contents of the counter are not zero when the next frame sync arrives, data valid is asserted until the counter rolls over to zero. Note that this requires the frame synch signal to precede the first valid pixel by up to 63 clock cycles depending on the number of pixels in a frame. This is not necessary for cameras with a multiple of 64 pixels per frame (most line scan cameras and square format area scan cameras) since the counter will be at zero after the last pixel of the frame.

In addition to the input FPGA, a small crosspoint switch (IQ32B) is used for flexible routing of clock signals with low latency and skew.

E. CPLD

A Complex Programmable Logic Device (CPLD) provides some camera control and board interface logic. The primary TriMedia processor communicates with this device via its V.34 synchronous serial interface. The CPLD provides logic to convert the V.34 serial data to a JTAG stream for programming the IQ32B crosspoint switch and to a Xilinx standard serial stream for programming the FPGAs. This device also provides variable time delays for camera and strobe control.

The CPLD is downloaded from a serial EPROM each time the system is started. See Programming, Configuration, and Test for details on these CPLD functions.

F. UART

The UART provides transmit and receive data, as well as one handshake input and one handshake output. The handshake lines are under program control. They typically tie to Data Terminal Ready and Data Set Ready lines of the remote equipment.

G. Power

The FastIO is built primarily with 3.3 volt components. The user must connect the system power supply to the auxiliary input connector (P7) using the Alacron-supplied cable (or a standard disk-drive cable).

III. PROGRAMMING, CONFIGURATION AND TEST

This section, along with the CPLD and UART Specifications, review the components and connections used for configuration and internal control, diagrammed in Figure 6.

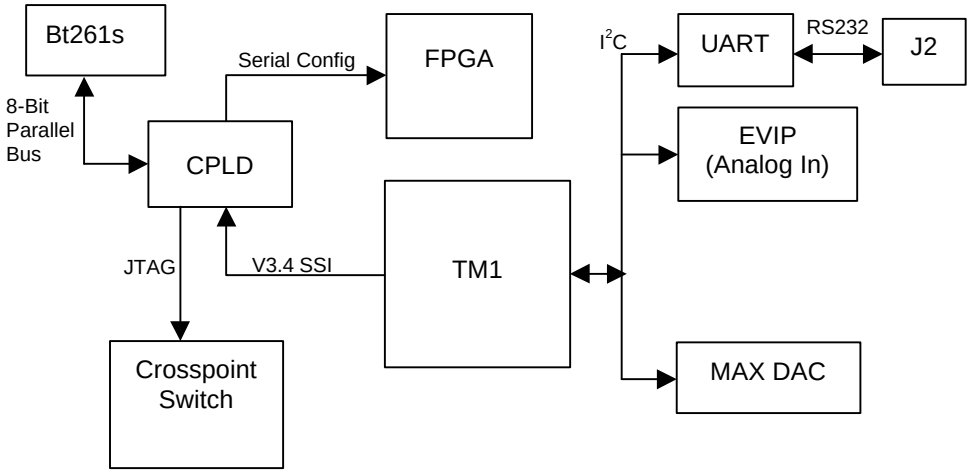

Figure 6. Configuration and Control

A. I2C Bus

The TriMedia processor configures several peripheral devices via an Inter-Integrated Circuit (I^2C) 2-wire serial bus. Devices on the I^2C bus include the SAA7111A Enhanced Video Input Processor, the MAX521 DAC (used for setting analog video gain and offset), and the UART. All I^2C peripherals on the FastIO use 7-bit addressing. The SAA7111A is accessed at 0x48 for writing and 0x49 for reading. The MAX521 is accessed at 0x50 for writing and 0x51 for reading. The UART is accessed at 0x54 for writing and 0x55 for reading.

The TriMedia processor also uses its I^2C bus lines to communicate to the front-end FPGAs.

B. V.34 SSI Interface (TM1 to CPLD)

The CPLD accepts transmitted data from the TriMedia TM1 via the V.34 SSI in 16-bit packets marked by the frame sync signal. Each packet contains 8 bits of data and 8 bits of address / control including a bit which must change state on each frame to indicate valid data. This bit is necessary since the V.34 interface will continue transmitting the last data word when its transmit FIFO is empty.

The CPLD provides independent frame sync signals for transmit and receive sections of the V.34 SSI. The transmit frame sync runs continuously, one sync for each 16 clocks. The receive frame sync only runs when there is data to be transferred from the CPLD to the TriMedia. The receive clock is also gated off when there is no data available. Note that with continuous receive channel transmission, a protocol such as that used for the transmit channel could be used, but it would burden the DSP CPU with regular interrupts whether or not data was being transferred.

TM1 can access CPLD input and output bits via the V.34 SSI interface. Writing output bits is similar to writing JTAG data. Reading input bits requires a write to a control location, which causes the data to be sent back on the receiver interface. Because the TriMedia's V.34 receiver packs 16-bit frames into 32-bit words, a dummy read is required to make the data available to the TriMedia processor when an odd number of reads are performed.

The 16-bit transmitted frame (transmit vs. receive is defined per the TriMedia data book from the perspective of the DSP CPU) is broken into three fields. The first, consisting of just the MSB, is the change detect bit. A frame is only considered valid if this bit differs from the change detecting bit of the previous frame. The next field, which is 7 or 6 bits for 8 and 9 bit data registers respectively, denotes the internal register address. The remaining (low order) 8 or 9 bits carry data.

Received frames are generated during JTAG download operations and in response to writes of the internal read request registers. The most significant 4 bits of these frames identifies the source of the data in the lower 12 bits. Note that when an odd number of frames are to be received, a dummy read request is necessary to allow the DSP CPU to see the last frame because of the 32-bit internal interface of the V.34 SSI.

The TriMedia's V.34 SSI needs to be programmed to use separate clock and frame sync signals for transmit and receive. This is accomplished by setting the V34_IO1 mode select bits to 10 and the V34_IO2 mode select bits to 11. The frame size and valid slot size should both be set to 1 slot. All clock polarity, shift direction, sync mode, and sync polarity bits should be set to 0. Endian mode select can be programmed as desired, but in most cases should match the endian mode of the DSP CPU.

Internally the V.34 SSI uses a 32-bit interface. This means that a minimum of two frames must be transmitted or received at a time. When transmitting a single frame of valid data, the other half of the 32-bit word can be filled with a copy of the valid word (the second frame transmitted will be ignored by the CPLD because the change detect bit matches the first frame). Alternately the unused half of the word can be all 1s or all 0s, which are not mapped to valid internal registers. When using the latter approach it is important to note the order of transmission, which is determined by the Endian Select bit in the SSI Control Register.

1. Serial Communication Port (UART)

A simple UART drives a four-wire RS-232 interface (J2) to allow asynchronous communication at 600 to 19,200 baud for camera setup and low speed control. Two wires implement standard transmit and receive data. The other two implement input and output handshakes. These signals are directly programmable by the primary TriMedia using the I^2C bus to access the internal registers of the UART. This port is provided primarily to allow camera configuration data to be downloaded without the use of a host system communication port. See the UART Specification later in the manual.

2. FPGA Serial Programming

The Xilinx serial data interface allows in-system programming of the FPGA from the CPLD using the slave serial mode of the FPGA (It is also possible to use the JTAG port for configuration). This interface sends 8 bits of data for each data word written at 10 MHz. Register bits are provided to drive the PROGRAM, INIT, and DONE lines and to read their current state. Xilinx tools provide the proper bitstream format for downloading configuration data to the FPGA. The same format applies whether JTAG or Serial Slave mode configuration is used. Unlike the bitstream files for the IQX crosspoint switch, these files do not contain JTAG control information. A description of the use of JTAG to configure Xilinx FPGA's is available as an application note (XAPP017) "Boundary Scan in XC4000 and XC5200 Series Devices."

C. JTAG Chains

1. JTAG Chain #2 - TriMedia Processors

A TriMedia JTAG chain with JTAG header (P4) is provided on the FastIO board.

The TriMedia processors do not support JTAG boundary scan, however they have a JTAG port, which will be used for program debugging by a remote host. The TriMedia JTAG port allows access to internal debug registers for communication between a JTAG host and a debugger running in the TriMedia. This is described in the data book in Chapter 17. Note however that the JTAG port is not capable of downloading code or booting the TriMedia, thus the basic SDRAM and PCI interfaces must be operational to use the debugger.

The TriMedia processor with its associated SDRAM and boot EEPROM makes up a complete subsystem which may be operated without bringing up other processors. In addition, the SDRAM may be written and read by the host while the TriMedia is held idle. This allows simple host-based memory testing before downloading TriMedia self-test software. Once the primary TriMedia has been tested, other peripheral tests may proceed. TriMedia peripheral operation can be tested using built-in diagnostic loopback modes for both the Video In/Out and Audio In/Out systems.

D. JTAG Chain #3 – CPLD, FPGA, and I-Cube IQ32B

JTAG test header P3 is provided to allow cable access to the CPLD, Front End FPGA, and the IQ32B JTAG ports. Once the CPLD is programmed, either by serial EEPROM or JTAG, it will allow JTAG access to program all the other devices in its JTAG chain.

For the IQ32B device, I-Cube software can be used to download and test the part using a cable connected to a PC's parallel (printer) port.

Under normal operation the cable is removed and the IQX32B is accessed by the primary TriMedia using the CPLD as a "JTAG UART." The file format for switch configuration download is the same whether using the cable or under normal operation. This allows the development of diagnostic (e.g. loopback) switch configurations using the I-Cube tools and cable, and later development of TriMedia test code for in-system use of these same configurations.

IV. CPLD SPECIFICATION

The primary TriMedia processor communicates with this device via its V.34 synchronous serial interface. The CPLD provides logic to convert the V.34 serial data to a JTAG stream for programming the IQ32B crosspoint switch, and the two XCS30 FPGA's. This device also provides variable time delays for camera and strobe control, a capture control signal to the front end FPGAs, and interface to the BT261 line lock controllers. A five-wire auxiliary interface allows an external connection to the JTAG interfaces for debug and test. Note that each JTAG interface is accessed independently; there are no chains. This is accomplished by providing independent clocks (TCK) to each device and multiplexing the data outputs (TDO) from each device.

A. V.34 SSI Interface

The CPLD accepts transmitted data from the V.34 SSI in 16-bit packets marked by the frame sync signal. Each packet contains 8-9 bits of data and 7-8 bits of address / control including a bit that must change state on each frame to indicate valid data, since the V.34 interface will continue transmitting the last data word when its transmit FIFO is empty.

The CPLD provides independent frame sync signals for transmit and receive sections of the V.34 SSI. The transmit frame sync runs continuously, one sync for each 16 clocks. The receive frame sync only runs when there is data to be transferred from the CPLD to the TriMedia. The receive clock is also gated off when there is no data available.

Data to and from the V.34 interface of the TriMedia is sent MSB first. Data to and from the JTAG port is sent LSB first. Data from the TriMedia to the CPLD is referred to as transmitted data. Signals relating to transmitted data are TxFSX (transmit frame sync), TxDATA, and the 20 MHz free-running clock. Data from the CPLD to the TriMedia is referred to as received data. Signals relating to received data are RxFSX, RxDATA, and RxCK.

Bit 15 (MSB) of the transmitted data is the change bit. No action is taken if a word's change bit matches the change bit of the previous word. This allows the TriMedia to let the transmitter underrun since re-transmitted words will have no effect. Bits 14 through 8 (14 through 9 for 9-bit registers) are the internal register select bits. Internal addresses are completely decoded. Addresses are chosen with a mix of 1's and 0's to prevent unintended action when the TriMedia transmitter is shut down. The low 8 or 9 bits of the transmitted word contain data. Internal registers and their function are described below.

Bits 15 through 10 of the received data indicate the data source as follows:

0101xx	JTAG TDO read back of device xx (xx is per Table 3)
1000	Internal Control register read back
1001	Tap 1 BT261 register read back
1010	Tap 2 BT261 register read back
1011	Tap 3 BT261 register read back

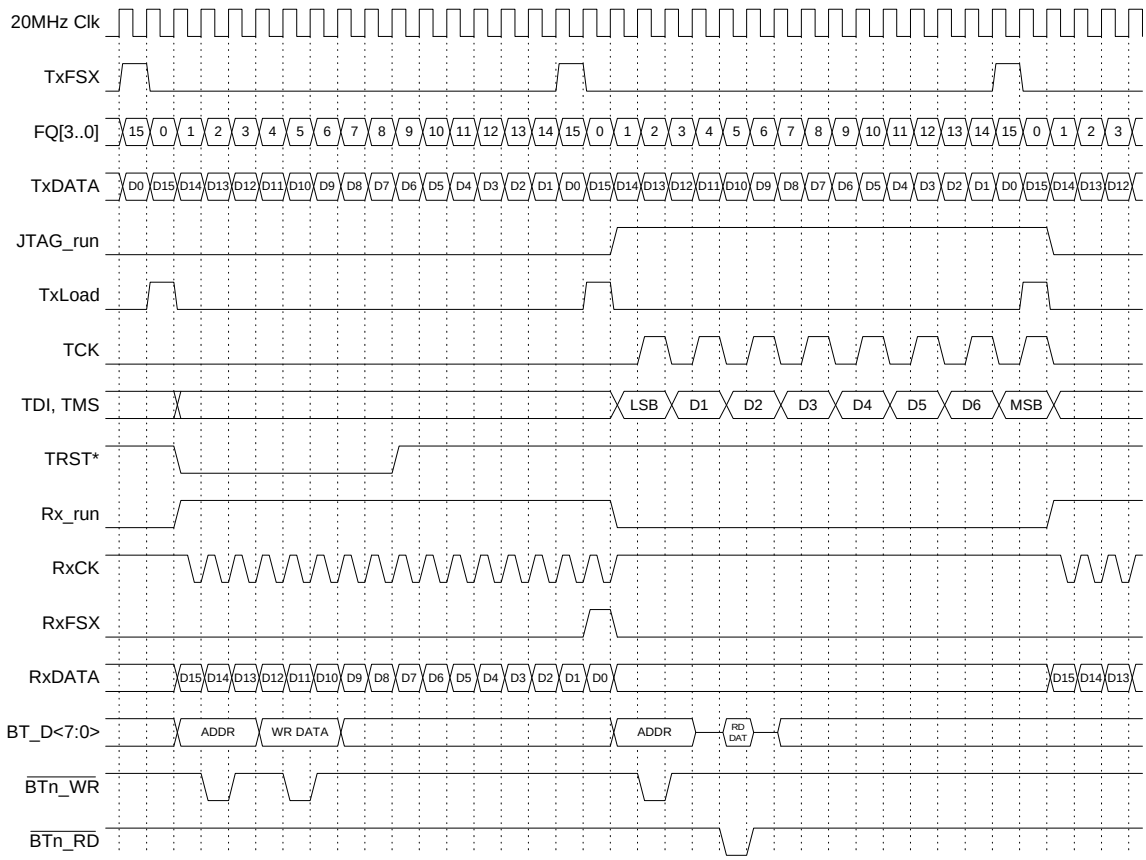

Figure 7. V.34 Interface Timing

B. JTAG Interface

JTAG data is provided as pairs of 8-bit words, one serialized onto the TMS line, and the other onto the TDI line. This is a standard format used by the I-Cube tools to store JTAG stream data. The clock line to be driven is selected by bits in the control register. Actual transmission of the JTAG stream starts after the TDI word is written. Thus a typical sequence would write a TMS byte followed by a TDI byte, however during long downloads the TMS byte can stay at the same value over a large number of cycles. The CPLD allows successive data words to be written to the TDI byte when the TMS byte is unchanged (i.e., the CPLD re-transmits the previous TMS data). This effectively doubles the throughput for long downloads.

The maximum clock rate of the V.34 SSI is 20 MHz. Since 16 bits are transmitted to write 8 bits of data, the effective JTAG download rate is only 10 MHz when writing successive TDI bytes, and 5 MHz when writing a TMS byte for each TDI byte. The JTAG clock generated by the CPLD is always 10 MHz, but stops between bytes when no new TDI data is available (see figure 8).

During JTAG operations, the TDO signal from the selected device is shifted into the CPLD and the data returned to the TriMedia via the V.34 receiver interface, again with 8 data bits per 16-bit frame. This data can be used to verify JTAG operation or to upload the current state of the device.

C. External Five-Wire Interface

Five lines from a test header allow access to the JTAG ports of the crosspoint switches and FPGA from an external system for debugging or test. These lines are defined as EXT_TCK, EXT_TDI, EXT_TMS, EXT_TRST*, and EXT_ENABLE*. All lines are internally pulled up.

While EXT_ENABLE* is high (inactive), EXT_TCK, EXT_TRST*, EXT_TMS and EXT_TDI are ignored.

When EXT_ENABLE* is brought low (active), the external interface is allowed to drive the JTAG lines and the internal V.34 to JTAG interface (from the TriMedia) is disabled. At this point the remaining lines have the usual JTAG functions when accessing the I-cube crosspoint switches. When either of the FPGAs is selected, the EXT_TRST* line acts as the PROGRAM* signal to the selected FPGA.

Once the CPLD has been configured, its JTAG lines are available as an I/O pins. These pins are available on the same header as the external five-wire interface. While EXT_ENABLE* is held low, the CPLD's TDO line passes through the TDO signal from the selected device. The TMS and TDI lines are used to select the device per Table 1 where TMS is "bit 1" and TDI is "bit 0". The CPLD's TCK line should remain low during this time to prevent the boundary scan circuitry from inadvertently reprogramming the part.

In order to keep the interface as close as possible to a direct JTAG connection to the selected device, all JTAG signals are routed combinatorially (there is no shift register stage passing through the CPLD). This added delay must be taken into consideration when selecting a test clock frequency for the external device.

D. PCI Bus Secondary Clock Mask

A parallel-in, serial-out shift register is implemented in the CPLD to load the secondary clock mask into the 21150 bridge chip after PCI reset. Sense signals from the PMC slots and TriMedia processors (except the first which must always be present) are used to determine which of the secondary clocks will be driven. This reduces system noise when clocks are not needed. The operation of this register is described in the 21150 data sheet in section10.2.

The output mask is active low, i.e. the clocks are enabled by a zero in the appropriate bit position(s) of the shifted data. A pull-down resistor on the printed circuit board will cause all clocks to go active if the release of PCI reset happens before the CPLD has been configured. The CPLD can take as long as 50 milliseconds to configure. The PCI specification only requires the reset line to remain low for 1 millisecond after power supplies are stable. Note however that the CPLD will only load after power-on and not after subsequent PCI resets (e.g. from front panel reset switch or BIOS firmware) while the bridge chip will re-load the clock mask at each reset.

E. Frame Timing Functions

The CPLD has two 9-bit counters for generating low-speed timing signals. These counters run at 10 KHz, allowing timing from 100 microseconds to 50 milliseconds. The two counters are triggered by a rising edge of the EXT_TRIG1 and EXT_TRIG2 signals respectively. Upon triggering, they count up from zero and stop when they reach maximum count (511). Two 9-bit compare registers, COMP1 and COMP2 are associated with each counter. These are programmed to create timing events from 1 to 512 clock cycles after the trigger. In normal usage the value of COMP1 is less than that of COMP2. Four output signals are associated with each counter. These are run to the IQX32B where they can be selected to run to the various control output drivers. Two of these signals are active high pulses, 100 microseconds wide, beginning at the times programmed in COMP1 and COMP2.

Another is an active high pulse which starts at the trigger and ends at the time programmed into COMP1. The last is an active high pulse starting at the time programmed into COMP1 and ending at the time programmed into COMP2 (note that if COMP2 is less than COMP1 this signal will begin during one trigger cycle and end during the next).

The counters cannot be retriggered while running until the time programmed into COMP2 has elapsed. The COMP2 pulse output can be externally routed to the trigger input to generate a free-running timebase.

The intent of the timing signals is to create readout and exposure control signals to line-scan or area-scan cameras (e.g. EXSYNC and PRIN to Dalsa cameras). Four crosspoint switch outputs and RS-422 drivers are designated for these signals.

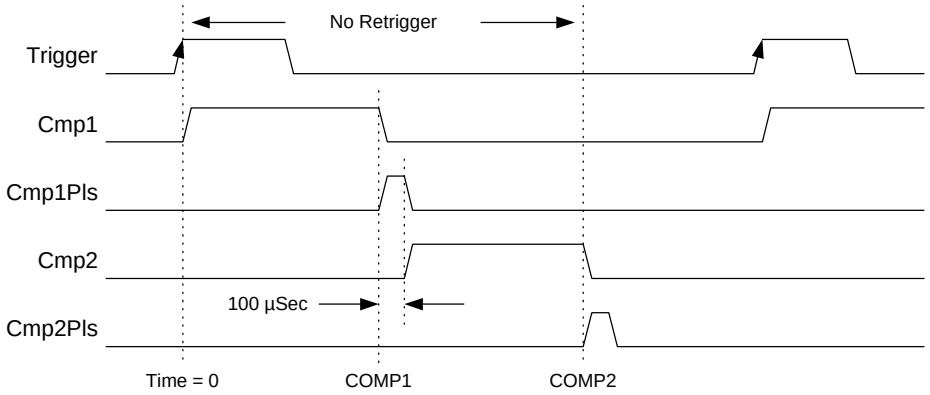

Figure 8. Frame Timing Signals

F. BT261 Parallel Bus Interface

A simple 8-bit interface is provided for programming the BT261 Line-Lock Controllers. The programming interface consists of three registers. The first of these is a control register which selects the BT261 and the address of the internal register to be accessed. No action is performed when writing this register. The next register is the write data register. Writing to this register causes a write to be performed to the BT261 and internal register selected by the control register. The last is the read register. Data written to this location is ignored. Writing to this register causes the selected BT261 and internal register to be read and the data to be sent to the TriMedia.

G. CPLD Programming Requirements

The CPLD has a fixed "program" which is downloaded from a serial EPROM each time the system is powered on. It contains registers to set pulse timing and delay parameters. It also provides the interface to the JTAG port of the IQX32B crosspoint switch. Register definitions for this part are as follows:

Address	Description
2B	TDI register. 8 data bits. Writing this register causes the JTAG state machine to shift out 8 bits of TDI and TMS data, LSB first. TDI data is supplied with this write; TMS data must be written to the TMS register first.

2D TMS register. 8 data bits. This register should be loaded before writing TDI. These bits are shifted out synchronously with the TDI data written on a subsequent cycle. Note that JTAG shift of TMS is non-destructive (shift register recirculates). Thus back-to-back writes of TDI register cause the same 8-bit sequence of TMS data to be output. This is useful to speed up long downloads when TMS remains at 0 for large numbers of bytes.

35 Control register. 8 data bits. Bits 0 and 1 select the destination device for JTAG transfers (see Table 3). Bits 2 and 3, when set to 1 cause the FPGA1 and FPGA2 DONE lines to be driven low respectively. When 0, the respective DONE line is tri-stated. Bit 4 selects the download mode when addressing either of the FPGA's (bits 1:0 = 1x). A 1 in this bit position selects serial download via the Xilinx-style programming interface pins. A 0 in this position selects download via the JTAG port. Note that the JTAG port may also be used for boundary scan or for communication with user logic after download is complete. Bits 5 and 6, when set to 1, cause the FPGA1 and FPGA2 INIT* lines to be driven low respectively. When 0, the respective INIT* line is tri-stated. Bit 7 resets the device selected by bits 0 and 1 when written as 1 (no action when written as 0). For the crosspoint switches this is accomplished via the TRST* line. For the FPGAs the PROGRAM* line is pulsed (allowing re-load of configuration data).

Bits 1,0	Selected Device:
0 0	
0 1	IQ32B Crosspoint Switch (U41)
1 0	FPGA 1 (U20)
1 1	FPGA 2 (U23)

Table 3. CPLD Control Register Bits 1,0

45 General purpose output register. Bit 1 when set to 1 enables the Digital output RS-422 drivers (if installed). Bit 0 when set to 1 turns on the test LED (#11). This can be used as a simple way to debug V.34 communications.

4A BT261 control register and capture bit. Writing this register sets up the address for the internal BT261 registers and selects a BT261, but does not initiate a data transfer to these parts. Bit 7 is the CAPTURE bit. The state of this bit is propagated to both front-end FPGAs and is intended to start or stop image capture, although its actual use depends on the FPGA program. Bits 6 and 5 select a BT261 for the next read or write transfer (see table 4). Bits 4 through 0 select an internal register of the BT261 (see BT261 datasheet Table 2). Note that the write of the address register described in the data sheet happens every time the part is read or written. This is handled in this CPLD. Unlike the part's internal address register, these bits are NOT incremented after each access. Thus successive reads or writes will address the same internal register.

Bits 6,5	Selected Device:
0 0	Tap 1 BT261 (U38)
0 1	Tap 1 BT261 (U38)
1 0	Tap 2 BT261 (U37)
1 1	Tap 3 BT261 (U36)

Table 4. BT261 Control Register Bits 6,5

4B BT261 read register. Data written to this "register" is ignored. Writes to this register cause a read of the selected BT261 register and a subsequent posting of this data to the received data port. Bits 15 through 12 of the received data are 1001 for tap 1, 1010 for tap2 or 1011 for tap3 based on the current device selection (bits 6,5 of BT261 control register). Bits 11 through 8 indicate the state of the external five-wire interface lines (except the clock). Bits 7 through 0 carry the data from the selected BT261 internal register.

4D BT261 write register. Data written to this register is written to the selected BT261 internal register.

54 Read control and status. Data bits are ignored. Writing this location causes the control and status to be sent to the received data port. Bits 15 through 12 of the received data are 1000 (binary) to indicate control and status read. Bits 11 through 8 indicate the state of the external five-wire interface lines (except the clock). Bit 7 returns the current state of the CAPTURE bit. Bits 6 and 5 indicate the current state of the FPGA2 and FPGA1 INIT* signals respectively. Bit 4 returns the current value of the Xilinx mode bit (see control register bit 4). Bits 3 and 2 indicate the current state of the FPGA2 and FPGA1 DONE signals respectively. Bits 1 and 0 return the currently selected device number (see control register bits 1,0).

58-59 ST1C1 - Slow timer 1 compare register 1. 9 data bits. Indicates delay to compare pulse 1 in units of 100 microseconds. Note for 9-bit data, the address is shortened, thus the LSB of the "address" (58 vs 59) is really the MSB of the data.

5A-5B ST1C2 - Slow timer 1 compare register 2. 9 data bits. Indicates delay to compare pulse 2 in units of 100 microseconds.

5C-5D ST2C1 - Slow timer 2 compare register 1. 9 data bits. Indicates delay to compare pulse 1 in units of 100 microseconds.

5E-5F ST2C1 - Slow timer 2 compare register 2. 9 data bits. Indicates delay to compare pulse 2 in units of 100 microseconds.

V. UART SPECIFICATION

A. Purpose

The FastIO board needs a simple asynchronous RS-232 port to set up certain intelligent cameras. The TriMedia has no simple asynchronous serial communication port. It also has no general purpose parallel data bus (other than PCI—the TM1100's 8-bit XIO bus shares pins with the PCI interface, making it unusable in hosted systems). This leaves one with the option of designing a PCI bus interface to add a standard UART chip, or designing a UART which attaches to I^2C (or one of the other complex interfaces such as audio or video I/O ports or the V.34 serial interface port). The PIC16C63 microcontroller, with its built in I^2C slave interface and USART was an ideal choice to implement the latter approach.

B. Features

- I^2C slave Universal Asynchronous Receiver Transmitter
- Standard Baud rates from 600 to 19,200 bits per second
- 7 or 8 data bits
- Selectable Odd, Even, Mark or Space parity
- RS-232 handshake lines: software controlled, one input, one output
- General purpose parallel I/O bits: four outputs, six inputs
- Simple programming interface
- Direct access to PIC registers for ease of debugging
- Powers on to 9600 Baud, 8 data bits, no parity, handshake output high

C. I^2C Address

The I^2C UART is addressed as a standard 7-bit peripheral at locations 0x54 (write) and 0x55 (read). Internal registers are accessed by subaddress using the same procedure as accessing a standard I^2C EEPROM. Multiple reads and writes are also possible. The internal subaddress is incremented after each access. This again works just like a standard I^2C EEPROM.

D. Interrupts

The I^2C UART provides a common interrupt output pin. This is an active low output (Port A bit 4) intended for use as a level-triggered interrupt source. It is high (inactive) after a reset. It is low (active) whenever received data is available. It will also go low when a transmitted byte is moved from the internal transmit data buffer register to the UART transmit data buffer. If no received data is present it will go high at the next access to subaddress 00. This pin is pulled up and wired to the TriMedia PCI_INTC# pin.

Note that the Transmit Buffer Empty condition will only create an interrupt when the state is entered. Thus a program wishing to transmit UART data should poll the status before sending the first byte under program control. Subsequent bytes my be transmitted by the interrupt handler. Also note that the interrupt may be cleared by reading incoming data during a transmission. The interrupt handler should be prepared to handle this condition by checking to see if outgoing data is present and whether the transmit data buffer is empty after handling a received byte.

Interrupts are not required to use the UART. The UART Status Register can be polled to determine if received data is available and if the transmitter is ready to accept transmitted data. Also the PCI INTC pin of the TriMedia, which connects to the UART interrupt output, can be programmed as a general purpose input pin and polled as well.

E. Register descriptions

1. Subaddress 00, Write – UART Control Register

W-0	W-1	W-0	W-0	W-1	W-1	W-0	W-0
Rsvd	HshkO	Parity		Size	Baud Rate		
Bit 7	6	5	4	3	2	1	Bit 0

Bit 7 Reserved. Should be set to **zero** for compatibility with future versions.

Bit 6 Handshake output. Resets to **one** on power on. The use of this bit is system dependant. Writing a zero to this bit activates the RS-232 handshake output. This may be wired to DCD and/or DSR of the external device as required.

Bits 5:4 Parity. These bits affect both transmit and receive parity. Note that only odd or even parity are checked. Mark or space parity are not checked. Parity is always stripped from incoming data. The bits encode as:

00 No parity (if 8-bit) or space parity (if 7-bit). (power-on default)

01 Mark parity

10 Even parity

11 Odd Parity

Note that parity other than 00 results in a 9-bit data word if 8-bit word length is selected. An 8-bit data word with mark parity resembles 8-bit data with no parity and 2 stop bits.

Bit 3 Word length. Selects number of bits exclusive of parity.

0 7 bits (plus parity)

1 8 bits (plus parity if not 00) (power-on default)

Bits 2:0 Baud rate. Currently available standard Baud rates are:

000	600
001	1,200
010	2,400
011	4,800
100	9,600 (power-on default)
101	19,200
11x	Reserved.

2. Subaddress 00, Read - UART Status Register

R-0	R-x	R-0	R-0	R-0	R-1	R-0	R-0
Rsvd	HshkI	TxOVR	RxOVR	RxPerr	TxMT	TxAvail	RxAvail
Bit 7	6	5	4	3	2	1	Bit 0

Bit 7 Reserved for future use. Currently reads zero but user code should not depend on this.

Bit 6 Handshake input. The use of this bit is system dependant. A zero in this bit indicates an active RS-232 handshake input. This may be wired to DTR of the external device as required.

Bit 5 Transmitter overrun. A one in this bit indicates that the host (I^2C master) has attempted to write data to the transmit buffer when it was still full. This is a "sticky" bit. Once set, it stays on until the UART Status Register is read.

Bit 4 Receiver overrun. A one in this bit indicates that the host (I^2C master) did not read the receive data buffer in time, and an incoming character has been dropped. This is a "sticky" bit. Once set, it stays on until the UART Status Register is read.

Bit 3 Receive parity error. A one in this bit indicates that the most recently read character was received with incorrect parity. This bit is not "sticky" - if parity checking is desired, the host must read status for each received byte. Note that Mark and Space parity are not checked, only Even or Odd parity.

Bit 2 Transmit Shift Register Empty. A one in this bit indicated that the UART transmit shift register is empty. All outgoing characters have been completely sent. At this point the host may write two more characters without overrun.

Bit 1 Transmit Buffer Empty. A one in this bit indicates that the host may write a new character to the transmit data buffer. Previous characters may be still in process of transmission (see Bit 2).

Bit 0 Receive Data Available. A one in this bit indicates that a new character is available for reading in the receive data buffer. Characters must be read by the host before the next character is fully received to avoid overrun errors.

N.B. In order to reduce latency when reading the UART Status Register, it is updated from internal flags when the I^2C subaddress becomes 00. This usually occurs as a result of writing 00 to I^2C address 0x54, but it can also occur due to wrap of internal subaddress after reading or writing subaddress 0xff. Updating the status information has the side effect of clearing the internal "sticky" bits (Rx and Tx overrun error bits).

3. Subaddress 01, Write – Transmit Data Register

Writing to the Transmit Data Register initiates a UART transmission. Either 7 or 8 bits of data written to this register are valid depending on the word size selected in the UART Control Register. The UART status register must be read to make sure the transmit buffer is empty before writing new data to the transmit data register. Attempting to write data to the Transmit Data Register while the buffer is full will result in the new data being dropped and the Transmitter Overrun bit being set in the UART status register.

4. Subaddress 01, Read – Receive Data Register

Data received by the UART can be read from the Receive Data Register. UART status register bit 0 indicates when data is available to be read. Reading this register when the Receive Data Available status bit is not set may result in loss of data.

5. Subaddress 02, Write – Ports A / B latch

Bits 7:6 Reserved. Set to 00 for compatibility with future revisions.

Bit 5 Multitap bit. A one in this bit position causes nibble swapping in the front end input logic so each FPGA gets half (4 bits) of each input tap. A zero in this bit position unswaps the nibbles so each FPGA gets two complete (8-bit) taps.

Bit 4 Power Reset bit (active low 0 = reset). This bit comes up 0 (active). It holds off the CPLD and clock chips from driving the V.34 IO pins. Do not program this bit high until the V.34 IO pins have been programmed as inputs. This fixes a bug in the TriMedia SSI which causes the V.34 IO pins to initialize as outputs. Caution: while this fixes the bug at power-on, subsequent resets without power cycling can cause the V.34 to reset to its improper condition while the PIC has already been programmed to release the Power Reset bit. It is probably a good idea to program this bit back to zero when the V.34 interface is not in use, at least when using TriMedia processors (the TM1100 has fixed this bug).

Bits 3:0 General purpose outputs. Writing these bits sets the output latches for the associated pins. Bit 3 is general purpose output 4 to differential outputs on pins J1B-33 and J1B-34. Bit 2 is general purpose output 3 to differential outputs on pins J1B-31 and J1B-32. Bit 1 is general purpose output 2 to differential outputs on pins J1A-33 and J1A-34. Bit 0 is general purpose output 1 to differential outputs on pins J1A-31 and J1A-32.

6. Subaddress 02, Read – Ports A / B Pins

Bits 7:4 General purpose inputs. Reading these bits always results in the current value on the associated input pins. Bit 7 is general purpose input 4 from differential inputs on pins J1B-61 and J1B-62. Bit 6 is general purpose input 3 from differential inputs on pins J1B-7 and J1B-8. Bit 5 is general purpose input 2 from differential inputs on pins J1A-61 and J1A-62. Bit 4 is general purpose input 1 from differential inputs on pins J1A-7 and J1A-8.

Bit 3 Multitap bit. (readback of current value)

Bits 2:1 General purpose inputs. Reading these bits always results in the current value on the associated pins. Bit 2 is general purpose input 6 from differential inputs on pins J1B-29 and J1B-30. Bit 1 is general purpose input 5 from differential inputs on J1A-29 and J1A-30.

Bit 0 Power Reset bit (active low 0 = reset). (readback of current value)

7. Subaddresses 03-FF – PIC Register File

Access to the entire register file of the PIC16C63 allows for debugging and some amount of control. Note that indiscriminate writes to this area can crash the PIC program and make the UART inaccessible. Among other things, note that the register file includes access to the PIC program counter and the synchronous serial port (I^2C port) control registers. It also includes all of the USART registers, allowing alternate uses of this port. Some useful cases are listed below:

PORTA (05) The low order four bits are best accessed via the Port A/B pins register at subaddress 02. Bits 4 and 5 are the UART interrupt (active low) output and the RS232 handshake (active low) output respectively.

PORTB (06) This port has four outputs and four inputs which can be accessed via the Port A/B pins register at subaddress 02. Reading PORTB directly allows you to read back the value of the four general purpose output bits.

flags (26) Read this location to check UART status without clearing the sticky bits. It contains a subset of the UART Status Register bits. See the program listing for a description of this register.

UARTctrl (2A) This is the UART Control Register accessed at subaddress 00, however writing to this location will not cause the hardware to be updated using the new values. Reading this location is useful to verify current operating parameters since reading at subaddress 00 returns the status register instead of reading back the value written.

t_ticks (2E) This indicates the time since reset in units of 26.67 milliseconds. It holds at maximum count after approximately 6.8 seconds. It is a legacy of earlier code versions which included power reset sequencing for the four TriMedia CPUs.

SPBRG (99) This is the baud rate divider register. Read it to check current operating value. Write it to set a non-standard baud rate. Actual baud rate is calculated by the formula: 153,600 / (SPBRG + 1).

Many other useful and or destructive operations are possible. Refer to the PIC16C6x data sheet for more information.

VI. CONNECTORS AND JUMPERS

A. Connectors

1. Rear Bracket Connectors

Digital/Analog Input Connectors (J1A/J1B on FastIO) and RS232 Connector (J2 on FastIO) are accessible from the PCI rear panel bracket.

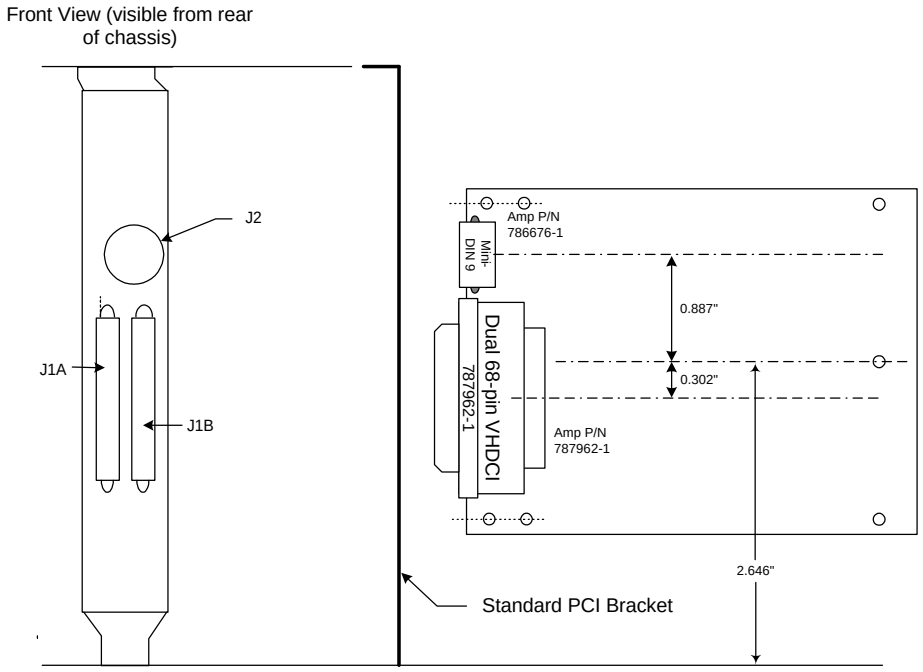

Figure 9. FastIO Rear Panel I/O Bracket

2. Analog and Digital Input Connector (J1A/J1B)

The Analog and Digital Input connector is on the Rear PCI Bracket: JA1/JB1 is a Dual 68-Pin Connector, with 32-bit Digital Inputs and Control or Analog Inputs (Table 4). All signals are differential, consisting of a positive and a negative signal pair; in the table, they are combined (e.g., differential signals TAP1_LVALP and TAP1_LVALN* are shown as just TAP1_LVAL). The positive signal is always the lower-numbered in each pair; the negative signal is the higher-numbered pin.

J1A/J1B Pins	J1A Signal	J1B Signal	J1A/J1 B Pins	J1A Signal	J1B Signal
1,2	TAP1_LVAL	TAP3_LVAL	35, 36	STROBE_1	STROBE_3
3, 4	TAP1_FVAL	TAP3_FVAL	37, 38	STROBE_2	STROBE_4
5, 6	TAP1_PXCK	TAP3_PXCK	39, 40	MASTER_CK 1	MASTER_CK 3
7, 8	GPIN1	GPIN3	41, 42	MASTER_CK 2	MASTER_CK 4
9	GND	GND	43	GND	GND
10, 11	TAP1_D0	TAP3_D0	44, 45	TAP2_D7	TAP4_D7
12, 13	TAP1_D1	TAP3_D1	46, 47	TAP2_D6	TAP4_D6
14, 15	TAP1_D2	TAP3_D2	48, 49	TAP2_D5	TAP4_D5
16, 17	TAP1_D3	TAP3_D3	50, 51	TAP2_D4	TAP4_D4
18, 19	TAP1_D4	TAP3_D4	52, 53	TAP2_D3	TAP4_D3
20, 21	TAP1_D5	TAP3_D5	54, 55	TAP2_D2	TAP4_D2
22, 23	TAP1_D6	TAP3_D6	56, 57	TAP2_D1	TAP4_D1
24, 25	TAP1_D7	TAP3_D7	58, 59	TAP2_D0	TAP4_D0
26	GND	GND	60	GND	GND
27, 28	EXT_TRIG1	EXT_TRIG2	61, 62	GPIN2	GPIN4
29, 30	GPIN5	GPIN6	63, 64	TAP2_LVAL	TAP4_LVAL
31, 32	GPOUT1	GPOUT3	65, 66	TAP2_FVAL	TAP4_FVAL
33, 34	GPOUT2	GPOUT4	67, 68	TAP2_PXCK	TAP4_PXCK

Table 5. Analog and Digital Input Connector J1A/J1B

3. RS232 I/O (J2)

Connector J2 is Mini 9-Pin DIN Connector on the rear PCI bracket. This connector is for RS232 I/O.

Pin	Signal	Pin	Signal
1		6	RS232-HSHKO
2		7	RS232-TxD
3		8	RS232-RxD
4		9	RS232-HSHKI
5	GND	10-12	Chassis

Table 6. RS232 Input/Output Connector J2

4. PMC Slot (PN4)

The FastIO Board has Standard PMC Connection PMC #1 (PN1,PN2, PN4) on the rear of the PCI Card. PN1 and PN2 are standard PCI connectors; PN4 has 32-bits of Digital input/output, 8-bits of control, 4 clocks, and I^2C. Table 7 shows the pinout.

Pin	PN4 Signal	Pin	PN4 Signal
1	DIG_DO0	33	DIG_DO26
2	DIG_DO1	34	DIG_DO27
3	DIG_DO2	35	GND
4	DIG_DO3	36	DIG_DO28
5	GND	37	DIG_DO29
6	DIG_DO4	39	DIG_DO30
7	DIG_DO5	39	DIG_DO31
8	DIG_DO6	40	GND
9	DIG_DO7	41	DIG_CO0
10	GND	42	DIG_CO1
11	DIG_DO8	43	DIG_CO2
12	DIG_DO9	44	DIG_CO3
13	DIG_DO10	45	+3V_PMC1
14	DIG_DO11	46	DIG_CO4
15	+3V_PMC1	47	DIG_CO5
16	DIG_DO12	48	DIG_CO6
17	DIG_DO13	49	DIG_CO7
18	DIG_DO14	50	GND
19	DIG_DO15	51	DO_PXCK1
20	GND	52	DO_PXCK2
21	DIG_DO16	53	DO_PXCK3
22	DIG_DO17	54	DO_PXCK4
23	DIG_DO18	55	GND
24	DIG_DO19	56	PMC1_SCL
25	GND	57	PMC1_SDA
26	DIG_DO20	58	PMC2_SCL
27	DIG_DO21	59	PMC2_SDA
28	DIG_DO22	60	+3V_PMC1
29	DIG_DO23	61	PMC3_SCL
30	+3V_PMC1	62	PMC3_SDA
31	DIG_DO24	63	PMC4_SCL
32	DIG_DO25	64	PMC4_SDA

Table 7. PMC-Pn4 Connector

5. Power

The FastIO Board has one Auxiliary Power Connector:

P7 - +3.3VDC. All systems require at least one power cable supplying +3.3V to P7.

6. Test Connectors

a) P3 (Test)

Pin	Signal	Pin	Signal
1	GND	10	CPLD_DIN
2	CPLD_TDI	11	CPLD_DOUT
3	CPLD_TDO	12	CPLD_CCLK
4	CPLD_TMS	13	CPLD_MODE
5	CPLD_TCK	14	TST_IN0
6	GND	15	TST_IN1
7	CPLD_PROG*	16	TST_IN2
8	CPLD_DONE	17	TST_IN3
9	CPLD_INIT*	18	TST_IN4

Table 8. Test Connector P3

b) P4 (Test)

Pin	Signal
1	DBUG_TRST* (Not used)
2	DBUG_TCK
3	DBUG_TMS
4	DBUG_TDI
5	DBUG_TDO
6	GND

Table 9. Test Connector P4

B. JUMPERS

The FastIO board has no user-configurable jumpers. The jumpers listed here are for Alacron use only.

1. P2

Jumper block P2 is set at the factory for the CPU and DRAM clock speeds. The default for 100 MHz and 133 MHz TriMedia is CKn_CPU = 50 MHz, SDRAM_CLK = 100 MHz. The default for 120 MHz TriMedia is CKn_CPU = 47.619, SDRAM_CLK = 142.85 (RAM @ 95.238 MHZ, CPU @ 119.0476 MHz).

1-2	3-4	5-6	CKn_CPU	SDRAM_CLK
OUT	OUT	OUT	66.666 MHz	133.33 MHz
IN	OUT	OUT	60	120
OUT	IN	OUT	55.555	166.666
IN	IN	OUT	50	100
OUT	OUT	IN	47.619	142.85
IN	OUT	IN	36.5	73
OUT	IN	IN	41.666	125
IN	IN	IN	40	80

Table 10. Jumper P2

2. P1

Jumper block P1 is set at the factory for other internal clock speeds. The default is BT_OSC1 = 80 MHz, TM_VIDCLK = 40 MHz.

1-2	3-4	5-6	BT_OSC1	TM_VIDCLK
OUT	OUT	OUT	73 MHz	36.5 MHz
IN	OUT	OUT	74	37
OUT	IN	OUT	75	37.5
IN	IN	OUT	76	38
OUT	OUT	IN	77	38.5
IN	OUT	IN	78	39
OUT	IN	IN	79	39.5
IN	IN	IN	80	40

Table 11. Jumper P1

3. P5

Jumper P5 enables programming of the EEPROM associated with the TriMedia processor.

Pin	Signal
1	E2PROM_WE*
2	GND

Table 12. Jumper P5

VII. FASTSERIES CABLES

A. Cable Summary

The FastIO board uses the following cables:

- 10024-00161 Cable, FastSeries Digital Input
- 10024-00162 Cable, FastSeries Analog Input

B. Power Cable

An auxiliary power cable connects to the system power supply in the PC to FastIO connector P7, providing +3.3V on pins 1 and 2, and GND on pins 3 and 4.

C. Digital Input Cable

The Digital Input cable 10024-00161 is a Y cable with one 68-pin connector (Cable-P1) and two 37-pin female DSUB37 connectors (Cable-P2 on the direct cable to Cable-P1 and Cable-P3 at the end of the Y). Cable-P1 mates with one of the 68-pin sides on the dual 68-pin connector (J1A/J1B on FastIO). Tables 13 and 14 show the signals at the pins of P2 and P3 (respectively) when connected to either J1A or J1B. Each digital input signal is a differential pair. The positive-true signal has suffix P in the tables; the negative-true signal has suffix N.

Conn-Pin	J1A Signal	J1B Signal
P2-1	EXT_TRIG1P	EXT_TRIG2P
P2-2	STROBE_1P	STROBE_3P
P2-3	GPOUT1P	GPOUT3P
P2-4	MASTER_CK1P	MASTER_CK3P
P2-5	GND	GND
P2-6	TAP1_D7P	TAP3_D7P
P2-7	TAP1_D6P	TAP3_D6P
P2-8	TAP1_D5P	TAP3_D5P
P2-9	TAP1_D4P	TAP3_D4P
P2-10	TAP1_D3P	TAP3_D3P
P2-11	TAP1_D2P	TAP3_D2P
P2-12	TAP1_D1P	TAP3_D1P
P2-13	TAP1_D0P	TAP3_D0P
P2-14	GND	GND
P2-15	TAP1_PXCKP	TAP3_PXCKP
P2-16	GPIN1P	GPIN3P
P2-17	TAP1_LVALP	TAP3_LVALP
P2-18	TAP1_FVAL	TAP3_FVALP
P2-19	N/C	N/C
P2-20	EXT_TRIG1N	EXT_TRIG2N
P2-21	STROBE_1N	STROBE_3N
P2-22	GPOUT1N	GPOUT3N

P2-23	MASTER_CK1N	MASTER_CK3N
P2-24	N/C	N/C
P2-25	TAP1_D7N	TAP3_D7N
P2-26	TAP1_D6N	TAP3_D6N
P2-27	TAP1_D5N	TAP3_D5N
P2-28	TAP1_D4N	TAP3_D4N
P2-29	TAP1_D3N	TAP3_D3N
P2-30	TAP1_D2N	TAP3_D2N
P2-31	TAP1_D1N	TAP3_D1N
P2-32	TAP1_D0N	TAP3_D0N
P2-33	N/C	N/C
P2-34	TAP1_PXCKN	TAP3_PXCKN
P2-35	GPIN1N	GPIN3N
P2-36	TAP1_LVALN	TAP3_LVALN
P2-37	TAP1_FVALN	TAP3_FVALN

Table 13. Digital Input Cable 10024-00161, DSUB37 Connector P2

Conn-Pin	**J1A Signal**	**J1B Signal**
P3-1	GPIN5P	GPIN6P
P3-2	STROBE_2P	STROBE_4P
P3-3	GPOUT2P	GPOUT4P
P3-4	MASTER_CK2P	MASTER_CK4P
P3-5	GND	GND
P3-6	TAP2_D7P	TAP4_D7P
P3-7	TAP2_D6P	TAP4_D6P
P3-8	TAP2_D5P	TAP4_D5P
P3-9	TAP2_D4P	TAP4_D4P
P3-10	TAP2_D3P	TAP4_D3P
P3-11	TAP2_D2P	TAP4_D2P
P3-12	TAP2_D1P	TAP4_D1P
P3-13	TAP2_D0P	TAP4_D0P
P3-14	GND	GND
P3-15	TAP2_PXCKP	TAP4_PXCKP
P3-16	GPIN2P	GPIN4P
P3-17	TAP2_LVALP	TAP4_LVALP
P3-18	TAP2_FVALP	TAP4_FVALP
P3-19	N/C	N/C
P3-20	GPIN5N	GPIN6N
P3-21	STROBE_2N	STROBE_4N
P3-22	GPOUT2N	GPOUT4N
P3-23	MASTER_CK2N	MASTER_CK4N
P3-24	N/C	N/C
P3-25	TAP2_D7N	TAP4_D7N

P3-26	TAP2_D6N	TAP4_D6N
P3-27	TAP2_D5N	TAP4_D5N
P3-28	TAP2_D4N	TAP4_D4N
P3-29	TAP2_D3N	TAP4_D3N
P3-30	TAP2_D2N	TAP4_D2N
P3-31	TAP2_D1N	TAP4_D1N
P3-32	TAP2_D0N	TAP4_D0N
P3-33	N/C	N/C
P3-34	TAP2_PXCKN	TAP4_PXCKN
P3-35	GPIN2N	GPIN4N
P3-36	TAP2_LVALN	TAP4_LVALN
P3-37	TAP2_FVALN	TAP4_FVALN

Table 14. Digital Input Cable 10024-00161, DSUB37 Connector P3

D. Analog Input Cable

The Analog Input cable 10024-00162 has a 68-pin connector (Cable-P1) on one end and four output connectors, three BNC (Cable-P2, -P3, and -P4) and one DB-9F (Cable-P5). The three BNC connectors are labeled P2, P3, and P4. Cable-P1 mates with one of the 68-pin sockets in the dual 68-pin input connector, J1A/J1B on FastIO, FastIO and FastIO. Cable-P2 is Table 15 shows the signals for the pins of P2, P3, P4, and P5 when connected to either J1A or J1B.

Conn-Pin	**J1A/J1B Pin**	**J1A Signal**	**J1B Signal**
P2-Center	10	Tap1 RS170 In	Tap3 RS170 In
P2-Shell	11	Tap1 RS170 Return	Tap3 RS170 Return
P3-Center	54	Not Used for Analog	Tap4 EVIP VID3 In (Comp/C)
P3-Shell	55	Not Used for Analog	Tap4 EVIP VID3 Return
P4-Center	58	Tap2 RS170 In	Tap4 EVIP VID1 In (Comp/Y)
P4-Shell	59	Tap2 RS170 Return	Tap4 EVIP VID1 Return
P5-1	1	Tap1 Line Valid Pos	Tap3 Line Valid Pos
P5-2	5	Tap1 Pixel Clock Pos	Tap3 Pixel Clock Pos
P5-3	26	GND	GND
P5-4	27	Ext Trigger 1 Pos	Ext Trigger 2 Pos
P5-5	37	Strobe 2 Pos	Strobe 4 Pos
P5-6	2	Tap1 Line Valid Neg	Tap3 Line Valid Neg
P5-7	6	Tap1 Pixel Clock Neg	Tap3 Pixel Clock Neg
P5-8	28	Ext Trigger 1 Neg	Ext Trigger 2 Neg
P5-9	38	Strobe 2 Neg	Strobe 4 Neg

Table 15. Analog Input Cable 10024-00161

The cable to J1A provides RS170 analog video inputs to Tap1 and Tap2. The cable to J1B provides RS170 analog input to Tap3 and composite or component video inputs to Tap4. A composite video source (NTSC/PAL/SECAM) can connect to either VID1 or VID3 (the input to the EVIP is software-downloadable). A component (S-video) source connects the Y (luma) component to VID1 and the C (chroma) component to VID3.

VIII. SPECIFICATIONS

A. Processors

- Full computational power of one or two TriMedia TM1000 or TM1100 VLIW media processors (~ 2GOPS) applied to the video streams
- Up to 64 MB of distributed SDRAM (8 or 16 MB per TM1000 processor, 8 to 32 MB per TM1100 processor)
- 0.8 GB/s (2 x 400 MB/s) local memory bandwidth
- Memory-mapped host access to distributed SDRAM for bootstrap and program load
- All on-board and host PCI resources can be directly addressed by each processor.

Clock rate	100 MHz (120 and 133 MHz processors when TM1100s are installed)
Memory	8, 16, or 32 MB SDRAM per processor at 100 MHz (120 MHz memory with faster processors), 400 MB/s peak access rate at 100 MHz (effective instruction access rate is higher due to compression)
Instruction Cache	32K on-chip cache per processor, 8-way set-associative, 2750 MB/s peak access rate
Data Cache	16K on-chip cache per processor, 8-way set-associative, 800 MB/s peak access rate

B. NTSC/PAL Composite Video Capture

- Continuous simultaneous image acquisition from one of four selectable composite NTSC/PAL video streams via the SAA7111 EVIP.

Input levels	1V peak to peak nominal 0.3 to 1.2Vp-p max.
Input impedance	75 Ohms
Channel crosstalk	-50 dB max
Resolution	8 bits
Bandwidth	6 MHz +/- 1dB (-36dB at conversion freq.)
Conversion rate	12.8 to 14.3 MHz (line locked)
Formats supported	PAL BGHI, PAL N, PAL M, NTSC M, NTSC N,NTSC 4.43, NTSC-Japan and SECAM
Frame rates	50 Hz 625 line and 59.94 Hz 525 line nominal
Horizontal line frequency (Hz)	15625 (50 Hz) or 15734 (59.94 Hz) +/- 5.7% max
Subcarrier frequency (Hz)	4433619 (PAL BGHI) 3579545 (NTSC) 3575612 (PAL M) 3582056 (PAL N)
Subcarrier lock range	+/- 400 Hz

C. ***Analog Video Capture***

- Continuous simultaneous image acquisition of three analog video streams generated by line or area scan cameras

Input levels	1V peak to peak nominal 2.0Vp-p max. 50mV minimum sync level when using composite sync.
Input impedance	75 Ohms
Resolution	8 bits x 3 channels
Conversion rate	0 to 40 MHz
Formats supported	Line scan and area scan up to 4K pixels per line.
Clock sources	Line locked (with composite sync) or external RS-422.

D. ***Digital Video Capture***

- Continuous simultaneous image acquisition from one of up to 4 digital video streams generated by line or area scan cameras
- 160 MB/s continuous digital capture bandwidth

Common mode input range	0 to +5V (0 to 2.4 with LVDS option)
Input sensitivity	250 mV differential (100 mV with LVDS option)
Input hysteresis	50 mV typ.
Maximum clock rate	45 MHz
Maximum input data width	32 bits
Formats supported	ITU-R BT.656 (4:2:2 interlaced color), 8/10 bit monochrome variable scan / line scan, 8/10 bit raw data, 8/10 bit RGB

E. ***Camera Control***

Serial port	Asynchronous RS232 600 to 19,200 Baud
Frame / line start outputs	Two RS422
Exposure control outputs	Two RS422
Master clock outputs	Four RS422 programmable in .07 Hz steps up to 40 MHz
General purpose outputs	Four RS422
Pixel clock inputs	Two RS422
Line / frame valid inputs	Four RS422
External trigger inputs	Two RS422
General purpose inputs	Six RS422
Power	No camera power provided.

F. ***PCI Interface***

- 132 MB/s peak local PCI bandwidth
- 33 MHz, 32-bit primary PCI bus interface. Bridge to 33 MHz, 32-bit local PCI bus

- Full compliance with PCI bus revision 2.1
- Memory-mapped host access to distributed SDRAM for bootstrap and program load.

Clock rate	33 MHz max
Data width	32 bits
Peak DMA rate	132 MB/s
Standards compliance	PCI Rev 2.1
Mechanical	Full length universal (5.0 / 3.3V) card. (Oversize: 12.783” x 4.775”)
Power	+/-12V at 100 mA max. +5V at 8 amps. (6.5 amps are drawn from a disk-drive supply cable).

G. PMC Interface

- On-board expansion slot for one standard PMC module with back panel access to module I/O
- Second PMC expansion slot for additional CPU's.

Clock rate	33 MHz max
Data width	32 bits
Standards compliance	PCI Rev 2.1, PMC 3.3V Signalling
Mechanical	Fits standard length single-width module, 10mm board stacking height.
Power	+/-12V, +5V supplied by PCI connector, +3.3V can be supplied via an auxiliary connector on FastIO board.

IX. TROUBLESHOOTING

There are several things you can try before you call Alacron Technical Support for help.

_____ Make sure the computer is plugged in. Make sure the power source is on.

_____ Go back over the hardware installation to make sure you didn't miss a page or a section.

_____ Go back over the software installation to make sure you have installed all necessary software.

_____ Run the Installation User Test to verify correct installation of both hardware and software.

_____ Run the user-diagnostics test for your main board to make sure it's working properly.

_____ Insert the Alacron CD-ROM and check the various Release Notes to see if there is any information relevant to the problem you are experiencing.

The release notes are available in the directory: **\usr\alacron\alinfo**

_____ Compile and run the example programs found in the directory: **\usr\alacron\src\examples**

_____ Find the appropriate section of the Programmer's Guide & Reference or the Library User's Manual for the particular library and problem you are experiencing. Go back over the steps in the guide.

_____ Check the programming examples supplied with the runtime software to see if you are using the software according to the examples.

_____ Review the return status from functions and any input arguments.

_____ Simplify the program as much as possible until you can isolate the problem. Turning off any operations not directly related may help isolate the problem.

_____ Finally, first **save your original work**. Then remove any extraneous code that doesn't directly contribute to the problem or failure.

X. ALACRON TECHNICAL SUPPORT

Alacron offers technical support to any licensed user during the normal business hours of 9 a.m. to 5 p.m. EST. We offer assistance on all aspects of processor board and PMC installation and operation.

A. Contacting Technical Support

To speak with a Technical Support Representative on the telephone, call the number below and ask for Technical Support:

Telephone: **603-891-2750**

If you would rather FAX a written description of the problem, make sure you address the FAX to Technical Support and send it to:

Fax: **603-891-2745**

You can email a description of the problem to support@alacron.com

Before you contact technical support have the following information ready:

_____ Serial numbers and hardware revision numbers of all of your boards. This information is written on the invoice that was shipped with your products.

_____ Also, each board has its serial number and revision number written on either in ink or in bar-code form.

_____ The version of the ALRT, ALFAST, or FASTLIB software that you are using.

_____ You can find this information in a file in the directory: **\usr\alfast\alinfo**

_____ The type and version of the host operating system, i.e., Windows 98.

_____ Note the types and numbers of all your software revisions, daughter card libraries, the application library and the compiler

_____ The piece of code that exhibits the problem, if applicable. If you email Alacron the piece of code, our Technical-Support team can try to reproduce the error. It is necessary, though, for all the information listed above to be included, so Technical Support can duplicate your hardware and system environment.

B. Returning Products for Repair or Replacements

Our first concern is that you be pleased with your Alacron products.

If, after trying everything you can do yourself, and after contacting Alacron Technical Support, you feel your hardware or software is not functioning properly, you can return the product to Alacron for service or replacement. Service or replacement may be covered by your warranty, depending upon your warranty.

The first step is to call Alacron and request a "Return Materials Authorization" (RMA) number.

This is the number assigned both to your returning product and to all records of your communications with Technical Support. When an Alacron technician receives your returned hardware or software he will match its RMA number to the on-file information you have given us, so he can solve the problem you've cited.

When calling for an RMA number, please have the following information ready:

_____ Serial numbers and descriptions of product(s) being shipped back

_____ A listing including revision numbers for all software, libraries, applications, daughter cards, etc.

_____ A clear and detailed description of the problem and when it occurs

_____ Exact code that will cause the failure

_____ A description of any environmental condition that can cause the problem

All of this information will be logged into the RMA report so it's there for the technician when your product arrives at Alacron.

Put boards inside their anti-static protective bags. Then pack the product(s) securely in the original shipping materials, if possible, and ship to:

Alacron Inc.
71 Spit Brook Road, Suite 200
Nashua, NH 03060
USA

Clearly mark ***the outside of your package:***

Attention **RMA #80XXX**

Remember to include your return address and the name and number of the person who should be contacted if we have questions.

C. Reporting Bugs

We at Alacron are continually improving our products to ensure the success of your projects. In addition to ongoing improvements, every Alacron product is put through extensive and varied testing. Even so, occasionally situations can come up in the fields that were not encountered during our testing at Alacron.

If you encounter a software or hardware problem or anomaly, please contact us immediately for assistance. If a fix is not available right away, often we can devise a work-around that allows you to move forward with your project while we continue to work on the problem you've encountered.

It is important that we are able to reproduce your error in an isolated test case. You can help if you create a stand-alone code module that is isolated from your application and yet clearly demonstrates the anomaly or flaw.

Describe the error that occurs with the particular code module and email the file to us at:

support@alacron.com

We will compile and run the module to track down the anomaly you've found.

If you do not have Internet access, or if it is inconvenient for you to get to access, copy the code to a disk, describe the error, and mail the disk to Technical Support at the Alacron address below.

If the code is small enough, you can also:

FAX the code module to us at **603-891-2745**

If you are faxing the code, write everything large and legibly and remember to include your description of the error.

When you are describing a software problem, include revision numbers of all associated software.

For documentation errors, photocopy the passages in question, mark on the page the number and title of the manual, and either FAX or mail the photocopy to Alacron.

Remember to include the name and telephone number of the person we should contact if we have questions.

Alacron Inc.
71 Spit Brook Road, Suite 200
Nashua, NH 03060
USA

Telephone: 603-891-2750
FAX: 603-891-2745

Web site:
http://www.alacron.com/

Electronic Mail:
sales@alacron.com
support@alacron.com