DYNAMIC ENGINEERING
435 Park Dr., Ben Lomond, Calif. 95005
831-336-8891 **Fax** 831-336-3840
sales@dyneng.com
www.dyneng.com
Est. 1988

User Manual

PMC-BiSerial-BA1

Bi-directional Serial Data Interface
PMC Module

Revision A
Corresponding Hardware: Revision A

PMC-BiSerial
Bi-directional Serial Data Interface
PMC Module

Dynamic Engineering
435 Park Drive
Ben Lomond, CA 95005
831- 336-8891
831-336-3840 FAX

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

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

This product has been designed to operate with PMC Module carriers and compatible user-provided equipment. Connection of incompatible hardware is likely to cause serious damage.

Manual Revision A. Revised March 9, 2002.

Table of Contents

List of Figures

Product Description

PMC-BiSerial is part of the PMC Module family of modular I/O components by Dynamic Engineering. The PMC-BiSerial is capable of providing multiple serial protocols. The PMC-BiSerial-BA1 implements two protocols. The first uses data and clock only, with start and stop bits delimiting eight data bits plus an odd parity bit. The second protocol uses a burst clock to shift in 32 bits of data. After the last clock a data sync pulse goes high for one bit period to indicate the end of the data word.

Custom interfaces are available. We will redesign the state machines and create a custom interface protocol. That protocol will then be offered as a "standard" special order product. Please see our web page for current protocols offered. Please contact Dynamic Engineering with your custom application.

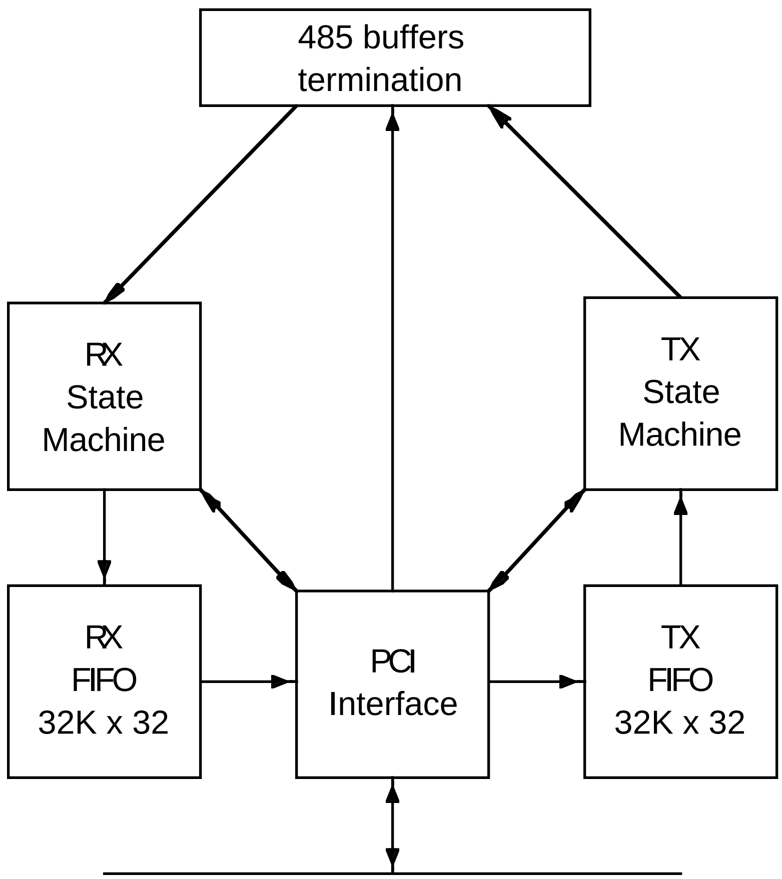

FIGURE 1 PMC-BISERIAL BLOCK DIAGRAM

Several clocking options are available with the standard –IO version. The PMC Clock, 10 MHz. oscillator and external reference input are the clock sources. The source can be selected with software as can the divisor. A 12-bit counter is provided with a programmable divisor to offer a multitude of frequency options based on the three standard references. If a specific frequency is required that is not attainable with the standard choices we can install a “user” oscillator. Please be sure to select the proper source and clock divisors after reset to insure proper operation. Please refer to the programming section for details.

Differential I/O is available on the serial signals. The drivers and receivers conform to the RS-485 specification (exceeds RS-422 specification). The RS-485 input signals are selectively terminated with 130Ω. The resistors are in discrete 1206 packages to allow individual termination options for custom formats and protocols. There are 20 transceivers for the IO. The transceivers are programmable to allow more outputs or more inputs as needed for a specific protocol implementation. The configuration implemented in the BA1 uses four outputs for data, sync, free-running clock, and burst-clock and six inputs: clock and data for each of the protocols, a sync for protocol two plus a selectable external reference clock. The terminations are programmable to be active or not.

All configuration registers support read and write operations for maximum software convenience. All addresses are long word aligned.

The PMC-BISERIAL conforms to the PMC and CMC draft standards. This guarantees compatibility with multiple PMC Carrier boards. Because the PMC may be mounted on different form factors, while maintaining plug and software compatibility, system prototyping may be done on one PMC Carrier board, with final system implementation on a different one.

PMC-BiSerial uses a 10 mm inter-board spacing for the front panel, standoffs, and PMC connectors. The 10 mm height is the "standard" height and will work in most systems with most carriers. If your carrier has non-standard connectors [height] to mate with the PMC-BiSerial, please let us know. We may be able to do a special build with a different height connector to compensate.

The serial channels are each supported by a 32K by 32 bit FIFO. The FIFO supports long word reads and writes. A path exists for loop-back testing of each FIFO. The PMC data path is 32 bits wide. The Data path from the FIFO to the Xilinx is 8 bits wide. The hardware automatically performs the 4 data accesses and byte lane manipulation to make the internal port appear as a 32 bit port to the PMC bus. The design is optimized for the system configuration with minimal delay on the PCI write to TX FIFO path and PCI read from the RX FIFO path. The added delay for reading and writing to the internal FIFO ports only affects the loop-back path.

As the serial receive channel receives data, LSB first, it is stored in 32-bit words. A 32-bit time stamp is stored at the beginning of each data block and a data count is stored at the end. For the first protocol, an 8-bit value in the receive control register determines the time that separates one data block from another. If this value is zero, all data is treated as a single block, if the value is one, every data sample is seen as a separate block with associated time stamp and count. Larger values determine the time between the stop bit of one word and the start bit of the next, which is presumed to separate data blocks. For the second protocol, all ones in the upper eight bits of the 32-bit data word indicate the first word of a message. The count for the previous message and the time stamp for the current message are determined and stored at this time. The host can poll the empty flag status or use the programmable FIFO Almost Full flag to determine when data is available. The message can be read directly from the input FIFO.

The Output channel has a separate 32K x 32 FIFO. The FIFO can be written as long words. Data is sent LSB first. Data is sent whenever the transmitter is enabled and data is stored within the FIFO. Transmission completes and the transmitter is disabled when the FIFO is detected to be empty.

Interrupts are supported by the PMC-BISERIAL. For the BA1 version interrupts can be generate by the following conditions: transmission complete, transmitter almost empty, receiver almost full, and an error condition for the second protocol where the sync goes high, but 32 clocks were not detected. The interrupts are individually maskable. The interrupt occurs on INTA. The FIFO status is available for making it possible to operate in a polled mode.

Theory of Operation

The PMC-BISERIAL is a part of the PMC Module family of modular I/O products. It meets the PMC and CMC draft Standards. Contact Dynamic Engineering for a copy of this specification. It is assumed that the reader is at least casually familiar with this document and logic design. In standard configuration, the PMC-BiSerial is a Type 1 mechanical with no components on the back of the board and one slot wide, with 10 mm inter-board height.

The PMC-BISERIAL-BA1 is designed for transferring data from one point to another with two distinct serial protocols. It features a Xilinx FPGA, which contains all of the registers and protocol controlling elements of the BISERIAL design. Only the transceivers, switches, and FIFOs are external to the Xilinx device.

A logic block within the Xilinx controls the PCI interface to the host CPU. The BISERIAL design requires one wait state for read or write cycles to any address other than the loop-back ports, which require eight. The wait states refer to the number of clocks after the PCI core decode and before the “terminate with data” state is reached. Two additional clock periods account for the 1 clock delay to decode the signals from the PCI bus and to convert the terminate with data state into the TRDY signal.

The PMC-BISERIAL-BA1 is designed to monitor traffic on two serial busses with different protocols. The first bus (see figure 2) has a free running clock with a constant high level on the data line indicating an idle state. When the data line goes low this indicates the start of an eight-bit data transfer. After the start bit and the eight data bits, transferred least significant bit first, an odd parity bit is inserted followed by a high stop bit. The data line remains high until the next transfer begins.

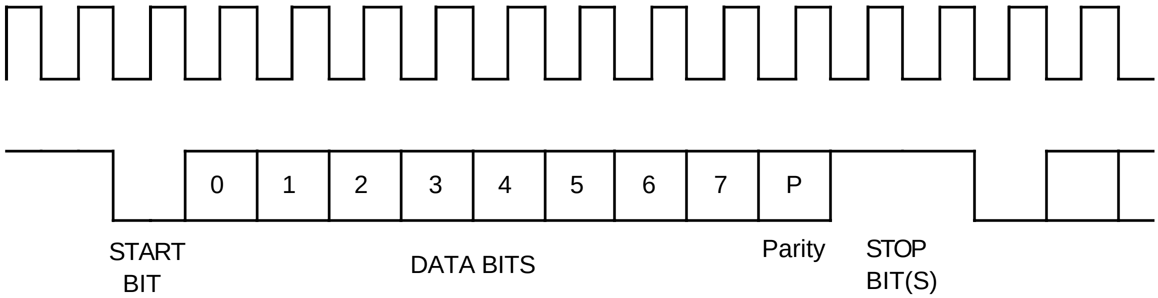

FIGURE 2 PMC-BISERIAL-BA1 PROTOCOL 1 TIMING

On the receive side, the data is latched on the rising edge of the clock, the parity is calculated and a bit is set in the stored data if an error is detected. An eight-bit field in the receive control register is used to determine the number of high bit-periods that must occur after the parity bit before it is assumed that the present data block i.e. message has ended. If this value is zero all data is treated as a single large block, if the value is one, a single stop bit is sufficient to indicate the end of a block, therefore every transfer is seen as a single byte block. If the value is between two and 255 that is the number of high bits that must be seen after the parity bit to cause the receiver to recognize the end of a block. When the end of a block is detected the byte count of that block is stored in the FIFO and when the next block begins a time stamp is stored. The first byte of a block is stored by itself in a 32-bit FIFO word. Subsequent data is stored up to three bytes per FIFO word.

The time stamp is generated by a 32- bit counter that counts microseconds derived from the on-board 10 MHz oscillator. This counter can be reset but otherwise counts continuously.

The second bus (see figure 3) transfers 32-bit values using a burst of 32 clock pulses. The data is also sent least significant bit first but is latched by the receiver on the falling edge of the clock. When the last clock pulse ends a one-bit wide sync is sent on a third serial line indicating the end of the word. The first word of a message has bits 31-24 high and this is used to determine the beginning and end of a message. When the first word is detected the count of the previous block is stored and the time stamp for the current block is latched and stored in the FIFO. The sync pulse causes the data word to be stored in the FIFO regardless of the number of clocks that occurred, however, if the clock count is not 32 an error bit is sent to the interrupt status register.

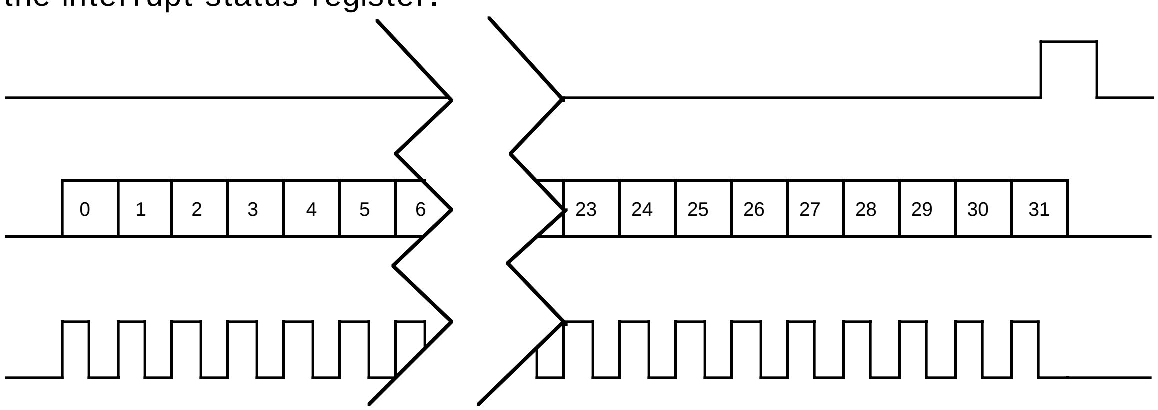

FIGURE 3 PMC-BISERIAL-BA1 PROTOCOL 2 TIMING

Since the PMC-BISERIAL-BA1 is a bus monitor the transmit section is only used to simulate the bus data for test purposes. To simulate both protocols with a single transmit section an extra (33^{rd}) bit has been added to the output data stream for each FIFO word. In the first protocol this bit provides the stop bit for the third data value in a 32-bit FIFO word. Each data word consists of a start bit eight data bits a parity bit and a stop bit for a total of 11 bits. The 33^{rd} bit is always high and allows three 11-bit values to be sent from one FIFO word. In the second protocol this bit occurs between groups of 32 clock pulses and is therefore not registered as data.

The free running clock for the first protocol and the 32-pulse burst clock and sync for the second protocol come out on three different lines for a total of four serial outputs.

Clock reference sources include an on-board 10 MHz oscillator, the PCI clock and a user input clock. The reference clock is available at the base rate or after a programmable 12-bit divider. Please refer to the memory map for more details.

An interrupt can be generated by four conditions: the transmit complete condition, determined by the transmit FIFO going empty; the transmit FIFO programmable almost empty; the receive FIFO almost full; and an error condition of the second protocol where the sync occurs when 32 clock pulses weren't registered. These conditions are latched until cleared by writing a one to the corresponding bit of the interrupt status register. Each interrupt is individually maskable by bits in the transmit and receive control registers and there is a master interrupt enable in the base control register.

The values used for the almost empty and almost full FIFO flags default to seven words from the empty and full conditions. However, resetting the FIFOs with the load bit held low allows these defaults to be changed. After the reset bit is set high again writing to the FIFO writes to four registers in a continuously rotating order: almost empty lower eight bits, almost empty upper seven bits, almost full lower eight bits, and almost full upper seven bits. In order to write to the receive FIFO the test mode bit must be set to select the PCI bus as the data source. The flags are read from the FIFO that stores bits 31-24, therefore the values written to the level registers must be placed on this byte lane e.g. 0x10000000 translates to a value of ten hex or sixteen decimal. After the four registers are written the load bit is set high and the FIFO is ready for normal operation.

Address Map

Function		Offset	function	Type
bis_base	EQU	$00	base control register	read/write
bis_tx	EQU	$04	transmit base control	read/write
bis_rx	EQU	$08	receive base control	read/write
bis_int_stat_clr	EQU	$0C	interrupt status/clear	read/write
bis_dir_term	EQU	$10	direction & termination	read/write
bis_stat0	EQU	$14	status register 0	read
bis_fifotx	EQU	$18	transmit FIFO access	read/write
bis_fiforx	EQU	$1C	receive FIFO access	read/write

FIGURE 4 PMC-BISERIAL-BA1 INTERNAL ADDRESS MAP

The address map provided is for the local decoding performed within the PMC-BiSerial. The addresses are all offsets from a base address. The carrier board that the PMC is installed into provides the base address.

The VendorId = 0x10EE. The CardId = 0x000B. Current revision = 0x00

Programming

Programming the PMC-BISERIAL requires only the ability to read and write data from the host. The PMC Carrier board determines the base address. This documentation refers to the first user address for the slot that the PMC is installed in as the base address.

Depending on the software environment it may be necessary to set-up the system software with the PMC-BiSerial "registration" data. For example in WindowsNT there is a system registry, which is used to identify the resident hardware.

In order to receive data the software is only required to enable the Rx state machine and select the Rx mode. If desired, the timer can be reset and interrupts can be enabled. Data will be loaded into the FIFOs as it is received.

If interrupts are to be used, the interrupt service routine should be loaded and the mask bits should be set. After the interrupt is received, appropriate action can be taken. New messages can be received even as the current one is read from the FIFO and data can be loaded into the Tx FIFO while data is being sent.

The end of transmission interrupt will indicate to the software that the Tx message has been started and that the message has terminated. If more than one interrupt has been enabled then the SW needs to read the interrupt status register to see which source caused the interrupt. After the condition that caused the interrupt has been removed, the interrupt status can be cleared by writing a one to the corresponding bit, then the interrupts can be re-enabled.

Before transmitting data, the FIFOs are enabled and the data loaded. If the clock rate desired is something other than the default rate then the rate should be selected. Be sure to set the clock source and rate bits appropriately. Once the complete message is loaded and the controls set properly the start bit can be set to cause the transfer to begin. If a slow clock rate is selected and a long message is sent then data can be loaded during transmission to save operational time. Care must be taken to insure that the FIFOs do not become empty or the transmission will terminate. When the Tx interrupt is received the transmission has been completed and another message can be loaded. All that needs to happen for a second message to be sent is to load the FIFO and set the start bit.

To poll read the Status 0 register during the transfer and take appropriate action when the full, empty or programmable flag shows that there is data to read or space to write. The PAE flag is implemented to provide an almost empty interrupt to allow the TX side to operate in an interrupt driven mode with longer messages. Similarly the PAF can be used to provide an almost full interrupt for the receive side to allow interrupt driven long message capability.

Refer to the Theory of Operation section above and the Interrupts section below for more information regarding the exact sequencing and interrupt definitions.

Register Definitions

bis_base

$00 BISERIAL Control Register Port read/write

CONTROL BASE	
DATA BIT	DESCRIPTION
31-23	spare
22	FRX_LD
21	FTX_LD
20	FIFO_EN
19-18	spare
17	INT_SET
16	INT_EN_MASTER
15	spare
14-13	clock pre-selector
12	clock post-selector
11-0	clock divisor

FIGURE 5 PMC-BISERIAL-BA1 BASE CONTROL REGISTER BIT MAP

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

FRX_LD is tied to the RX FIFO WE2 / _LD pin. FTX_LD is tied to the TX FIFO WE2 / _LD pin. When the FIFOs are taken out of reset it is possible to set-up the FIFO to accept commands to program the way the programmable almost empty and programmable almost full signals operate. ***In the standard transfer mode these pins are set hi before FIFO_EN is set to use as a second WE control pin***. If the PAE and PAF flags are used then the FIFOs may require programming. See the theory of operation section for information on performing this operation.

FIFO_EN when '1' takes the FIFO out of reset. To create a reset be sure to leave in the '0' state long enough for the reference clock to capture the reset. This can be an issue if a slow transmission rate is chosen. To guarantee reset the PCI clock can be used as a reference for both the Tx and Rx FIFOs temporarily then set back to the original settings.

INT_SET is used for test and software development purposes to create an interrupt request: 1 = assert interrupt request. 0 = normal operation. This is useful to stimulate interrupt acknowledge routines for development.

INT_EN_MASTER when '1' gates all interrupts through to the PCI host. When '0' the interrupts can be used for status without interrupting the host.

Clock Pre-Selector

00	oscillator
01	oscillator
10	external
11	PCI clock

The clock pre-selector is used to select which reference clock to use with the divisor hardware (the clock source). The external clock is on IO channel 0.

Divisor [11-0] are the clock divisor select bits. A counter divides the clock source. The reference clock for the counter is selected with the CLK Pre-Selector. The output frequency is [reference / [2(n+1)], n $\geq$ 1. The reference oscillator is 10 MHz in frequency. The counter divides by n+1 due to counting from 0 to n before rolling over. The output is then divided by 2 to produce a square wave output.

Post Selector when '1' sets clock out to clock divided, when '0' clock out is set to the pre-selector reference clock.

Please note that the 485 buffers are rated for 12 MHz. With most systems the larger divisors will be used. The smaller divisors are provided for use with external oscillators and the external clock line.

bis_tx

$04 BISERIAL TX Control Register Port read/write

CONTROL TX

DATA BIT	DESCRIPTION
31-4	spare
3	clock en
2	int_en_pae
1	int_en_tx
0	start_tx

FIGURE 6 PMC-BISERIAL TX CONTROL REGISTER BIT MAP

Clock_en when ‘1’ gates the transmit clock to be driven to TX_CLK. The gate is provided to allow FIFO loop back testing without driving the clock onto the transmission line. When ‘0’ the TX_CLK is held in at ‘0’. This enable should normally be set.

INT_EN_TX when ‘1’ enables the Tx interrupt. The default state is off. If enabled and the master interrupt enable is also enabled then an interrupt is requested when the transmission is complete.

INT_EN_PAE when ‘1’ enables the FIFO Programmable Almost Empty interrupt. When enabled an interrupt is generated when the data level falls to the programmed level.

Start_tx when ‘1’ will start a transmission assuming that there is data in the Tx FIFO. Start_tx is auto-cleared when the transmission is complete.

bis_rx

$08 BISERIAL RX Control Register Port read/write

CONTROL RX

DATA BIT	DESCRIPTION
31-16	spare
15-8	gap_cnt
7	spare
6	rst_sm
5	testmode
4	time_clr
3	int_en_err2
2	int_en_paf
1	rx_mode
0	start_rx

FIGURE 7 PMC-BISERIAL RX CONTROL REGISTER BIT MAP

Gap_cnt determines the number of high bits that must be seen between data words of protocol 1 before and end of block is registered. Zero causes all blocks to be seen as one.

Rst_sm when '1' resets the Rx state machine to the idle state. This bit needs to be '0' for normal operation.

Testmode when '1' selects the PCI bus as the data source for the Rx FIFOs. Normally set to '0'. Set to '1' for FIFO loop-back or programming the almost full flag level.

Time_clr when '1' resets the time stamp generator to zero, when '0' the timer counts freely.

Int_en_err2 when '1' enables the error2 interrupt for protocol 2. This occurs when the data sync pulse is detected, but the count of clock pulses is not 32.

Int_en_paf when '1' enables the RX FIFO Programmable Almost Full flag interrupt. The interrupt becomes active when the data in the RX FIFO reaches a user programmed point of almost full. Requires the master enable to create a system level interrupt.

Rx_mode when '0' enables the protocol 1 receive state machine, when '1' enables the protocol 2 receive state machine.

Start_rx is a master enable required to be '1' for either receive state machine to receive messages.

int_stat_clr

$0C BISERIAL Status Port [read only]

INTERRUPT STATUS/CLEAR	
DATA BIT	**DESCRIPTION**
31-4	spare
3	err2_int
2	rx_paf_int
1	tx_pae_int
0	tx_int

FIGURE 8 PMC-BISERIAL INT STATUS REG BIT MAP

Err2_int when '1' indicates an error condition has occurred in the protocol 2 state machine where an incorrect number of clocks was detected. This bit is latched until a '1' is written to bit 3.

Rx_paf_int when '1' indicates that the receive FIFO became almost full. This bit is latched until a '1' is written to bit 2.

Tx_pae_int when '1' indicates that the transmit FIFO became almost empty. This bit is latched until a '1' is written to bit 1.

Tx_ int when '1' indicates that the transmit state machine has completed sending a message. This bit is latched until a '1' is written to bit 0.

bis_dir_term

$10 BISERIAL direction and termination Port [read/write]

CONTROL REGISTER DIR_TERM	
DATA BIT	DESCRIPTION
8-0	DIRection 10-0 0 = read, 1 = drive
23-16	TERMination10-0 1 = terminated

FIGURE 9 PMC-BISERIAL DIRECTION TERMINATION CONTROL BIT MAP

The direction for each of the 20 differential pairs is controlled through this port. The port defaults to ‘0’ which corresponds to tri-stating the drivers. The output and input pins are separated and independently connected to the Xilinx to allow loop-back testing.

Pull-up and Pull-down resistors built into some '485 interface devices may make the signal appear to be driven [if open] when in the tri-stated mode. Enabling the termination on a tristated line will yield approximately 2.5V on each side of the tri-stated driver.

The PMC_BiSerial_IO_BA1 design sets the direction of the signals to be set to output except for the signal group 12-15 and IO 0,1, and 2, which are inputs. The table is shown for future use when we plan to add in a read / write parallel port on the unused transmission lines. Currently the forced bits are read-write but have no effect.

CONTROL	CORRESPONDING IO BIT(S)
DIR_0..3	IO_0..3.
DIR4	IO_4..7
DIR5	IO_8..11
DIR6	IO_12..15
DIR7	IO_16..19

Parallel termination resistors are supplied on each differential pair along with a switch to allow the user to select which lines are terminated and where. In some systems it will make sense to terminate the lines in the cable and in others it will make sense to use the onboard terminations.

The terminations for the receive groups should be set to terminate with the user software in most cases. [term0,1,2, and term6]

CONTROL	CORRESPONDING IO BIT(S)
TERM_0..3	IO_0..3.
TERM4	IO_4..7
TERM5	IO_8..11
TERM6	IO_12..15
TERM7	IO_16..19

bis_stat0

$14 BISERIAL Status Port [read only]

Data Bit	Status	
15	int_request	1 = interrupt request int_en after mask
14	error2	1 = error2 condition is present
13	frx_ffn	0 = rxfifo full, 1 = not full
12	frx_fafn	0 = almost full 1 = not almost full
11	frx_mtn	0 = rx fifo empty 1 = not empty
10	ftx_ffn	0 = tx fifo full 1 = not full
9	ftx_faen	0 = almost empty 1 = not almost empty
8	ftx_mtn	0 = tx fifo empty, 1 = not empty
7,6	spare	
5-0	sw_in	User dip switch setting

FIGURE 10 PMC-BISERIAL STATUS REG 0 BIT MAP

When int_request is '1' then at least one of the maskable interrupts is active and enabled.

When error2 is '1' the current word of the protocol 2 receiver has a clocking error. When the data sync pulse occurred the clock count was not 32.

The FIFO flags are active low. When the empty bit is low then the FIFO is empty. When the empty flag is high then the FIFO has at least one piece of data stored. When the Full Flag is set [low] the FIFO is full. When not set then the FIFO still has room.

Sw_in has the user bit values selected by switching bits 5-0 of the onboard dip-switch.

bis_fifotx

`$18 BISERIAL TX FIFO read/write port`

The BISERIAL supports 32 bit writes to the transmit data FIFOs. Data is aligned D31-0. Normally this port is only written to. For loop-back testing the contents of the FIFO can be read from the “Xilinx” side of the FIFO. The reference clock must be set to the PCI source for the loop-back to work. The engineering kit contains software which performs a TX FIFO loop-back. Once data is read from the FIFO it is no longer available for transmission.

bis_fiforx

`$1C BISERIAL RX FIFO read/write port`

The BISERIAL supports 32 bit reads from the receive data FIFOs. Data is aligned D31-0. Normally this port is only read from. For loop-back testing the contents of the FIFO can be written through the “Xilinx” side of the FIFO. The testmode bit must be set to cause the PCI bus to be selected as the data source. The engineering kit contains software, which performs an Rx FIFO loop-back. Once data is read from the FIFO it is no longer available.

Interrupts

PMC-BiSerial interrupts are treated as auto-vectored. When the software enters into an exception handler to deal with a PMC-BiSerial interrupt the software must read the interrupt status register to determine the cause(s) of the interrupt, clear the interrupt condition(s) and process accordingly.

The PMC-BiSerial Tx state machine generates an interrupt request when a transmission is complete and the INTEN bit in the control register is set. The transmission is considered complete when the Tx start bit is cleared. The interrupt is mapped to INTA on the PMC connector. INTA may be mapped to a different interrupt in your system. For example in our NT systems it is mapped to interrupt B. The source of the interrupt is obtained by reading the interrupt status register. The status remains valid until the status register written with the appropriate bit.

The interrupt level seen by the CPU is determined by the rest of the system. The master interrupt can be disabled or enabled through the bis_base register. The individual enables for the various interrupt conditions are controllable through bis_tx and bis_rx.

The individual enables operate after the interrupt holding latch, which stores the status for the CPU. Once the interrupt request is set, the way to clear the request is to write the proper bit to the interrupt status / clear register, assuming the original cause is no longer in effect, disable the individual interrupt mask, or disable the master interrupt enable. The master enable is a mask and can be used to disable the interrupt from reaching the CPU, but still leaves the internal interrupt request hardware active which is useful for polled operation.

If operating in a polled mode and making use of the interrupts for status then the master interrupt should be disabled and the Tx, and programmable level interrupts enabled as needed. When bis_stat0 shows an interrupt pending the appropriate FIFO action can take place and the appropriate action taken, if any, to remove the interrupt condition. Then the corresponding bit is written to the interrupt status / clear register to clear the request.

If interrupt driven the normal sequence to service an interrupt is to disable the master interrupt enable, then read the interrupt status / clear register to determine the cause of the interrupt. If an almost full or empty interrupt has occurred the appropriate FIFO is read or written to remove the interrupt condition. In any case the bit corresponding to the interrupt is then written to the interrupt status / clear register, which will clear only that bit and insure that interrupt conditions will not be missed. The master interrupt enable is once again asserted and the system is ready to capture the next interrupt event.

Power on initialization will provide a cleared interrupt request and interrupts disabled.

Loop-back

The Engineering kit has reference software, which includes an external loop-back test. The test requires and external cable with the following pins connected.

<u>Protocol 1</u>

Data+	7 - 4
Data-	41 - 38
Clk+	9 - 2
Clk-	43 - 36

<u>Protocol 2</u>

Data+	7 - 22
Data-	41 - 56
Clk+	6 - 24
Clk-	40 - 58
Sync+	11 - 21
Sync-	45 - 55

PMC PCI Pn1 Interface Pin Assignment

The figure below gives the pin assignments for the PMC Module PCI Pn1 Inte
BiSerial-IO.   See the User Manual for your carrier board for more informa
be assigned by the specification and not needed by this design.

	-12V[unused]	1	2
GND	INTA#	3	4
		5	6
BUSMODE1#	+5V	7	8
		9	10
GND -		11	12
CLK	GND	13	14
GND -		15	16
	+5V	17	18
	AD31	19	20
AD28-	AD27	21	22
AD25-	GND	23	24
GND -	C/BE3#	25	26
AD22-	AD21	27	28
AD19	+5V	29	30
	AD17	31	32
FRAME#-	GND	33	34
GND	IRDY#	35	36
DEVSEL#	+5V	37	38
GND	LOCK#	39	40
		41	42
PAR	GND	43	44
	AD15	45	46
AD12-	AD11	47	48
AD9-	+5V	49	50
GND -	C/BE0#	51	52
AD6-	AD5	53	54
AD4	GND	55	56
	AD3	57	58
AD2-	AD1	59	60
	+5V	61	62
GND		63	64

FIGURE 11 PMC-BISERIAL PN1 INTERFACE

PMC PCI Pn2 Interface Pin Assignment

The figure below gives the pin assignments for the PMC Module PCI Pn2 Inte
BiSerial-IO.   See the User Manual for your carrier board for more informa
be assigned by the specification and not needed by this design.

+12V[unused]		1	2
		3	4
	GND	5	6
GND		7	8
		9	10
		11	12
RST#	BUSMODE3#	13	14
	BUSMODE4#	15	16
	GND	17	18
AD30	AD29	19	20
GND	AD26	21	22
AD24		23	24
IDSEL	AD23	25	26
	AD20	27	28
AD18		29	30
AD16	C/BE2#	31	32
GND		33	34
TRDY#		35	36
GND	STOP#	37	38
PERR#	GND	39	40
	SERR#	41	42
C/BE1#	GND	43	44
AD14	AD13	45	46
GND	AD10	47	48
AD8		49	50
AD7		51	52
		53	54
	GND	55	56
		57	58
GND		59	60
		61	62
GND		63	64

FIGURE 12 PMC-BISERIAL PN2 INTERFACE

BiSerial-BA1 Front Panel IO Pin Assignment

The figure below gives the pin assignments for the PMC Module IO Interface
Also, see the User Manual for your carrier board for more information. GN
tied to GND through a 1206 0Ω resistor. AC or open are options – contact

IO_0p [ref clk+]	IO_0m [ref clk-]	1	35
IO_1p [RX_CLK1+]	IO_1m [RX_CLK1-]	2	36
GND*	GND*	3	37
IO_2p [RX_DATA1+]	IO_2m [RX_DATA1-]	4	38
GND*	GND*	5	39
IO_3p [TX_CLK2+]	IO_3m [TX_CLK2-]	6	40
IO_4p [TX_DATA+]	IO_4m [TX_DATA-]	7	41
GND*	GND*	8	42
IO_5p [TX_CLK1+]	IO_5m [TX_CLK1-]	9	43
GND*	GND*	10	44
IO_6p [TX_SYNC+]	IO_6m [TX_SYNC-]	11	45
IO_7p	IO_7m	12	46
GND*	GND*	13	47
IO_8p	IO_8m	14	48
GND*	GND*	15	49
IO_9p	IO_9m	16	50
IO_10p	IO_10m	17	51
GND*	GND*	18	52
IO_11p	IO_11m	19	53
GND*	GND*	20	54
IO_12p [RX_SYNC2+]	IO_12m [RX_SYNC2+]	21	55
IO_13p [RX_DATA2+]	IO_13m [RX_DATA2-]	22	56
GND*	GND*	23	57
IO_14p [RX_CLK2+]	IO_14m [RX_CLK2-]	24	58
GND*	GND*	25	59
IO_15p	IO_15m	26	60
GND*	GND*	27	61
IO_16p	IO_16m	28	62
GND*	GND*	29	63
IO_17p	IO_17m	30	64
GND*	GND*	31	65
IO_18p	IO_18m	32	66
GND*	GND*	33	67
IO_19p	IO_19m	34	68

FIGURE 13 PMC-BISERIAL FRONT PANEL INTERFACE

PMC Pn4 User Interface Pin Assignment

The figure provides the pin assignments for the PMC-BiSerial Module routed User Manual for your carrier board for more information.

IO_0p	REFCLK+	IO_0m	REFCLK-	1	2
IO_1p	RX_CLK1p	IO_1m	RX_DATA1m	3	4
IO_2p	RX_DATA1p	IO_2m	RX_CLK1m	5	6
IO_3p		IO_3m		8	9
IO_4p	TX_DATAp	IO_4m	TX_DATAm	9	10
IO_5p	TX_CLKp	IO_5m	TX_CLKm	11	12
IO_6p	TX_STBp	IO_6m	TX_STBm	13	14
IO_7p		IO_7m		15	16
IO_8p		IO_8m		17	18
IO_9p		IO_9m		19	20
IO_10p		IO_10m		21	22
IO_11p		IO_11m		23	24
IO_12p	RX_SYNC2p	IO_12m	RX_SYNC2m	25	26
IO_13p	RX_DATA2p	IO_13m	RX_DATA2m	27	28
IO_14p	RX_CLK2p	IO_14m	RX_CLK2m	29	30
IO_15p		IO_15m		31	32
IO_16p		IO_16m		33	34
IO_17p		IO_17m		35	36
IO_18p		IO_18m		37	38
IO_19p		IO_19m		39	40
				41	42
				43	44
				45	46
				47	48
				49	50
				51	52
				53	54
				55	56
				57	58
				59	60
				61	62
				63	64

FIGURE 14 PMC-BISERIAL PN4 INTERFACE

Applications Guide

Interfacing

The pinout tables are displayed with the pins in the same relative order as the actual connectors. The pin definitions are defined with noise immunity in mind. The pairs are chosen to match standard SCSI II/III cable pairing to allow a low cost commercial cable to be used for the interface.

Some general interfacing guidelines are presented below. Do not hesitate to contact the factory if you need more assistance.

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

Power all system power supplies from one switch. Connecting external voltage to the PMC-BiSerial when it is not powered can damage it, as well as the rest of the host system. This problem may be avoided by turning all power supplies on and off at the same time. Alternatively, the use of OPTO-22 isolation panels is recommended.

Keep cables short. Flat cables, even with alternate ground lines, are not suitable for long distances. PMC-BISERIAL does not contain special input protection. The connector is pinned out for a standard SCSI II/III cable to be used. The twisted pairs are defined to match up with the BiSerial pin definitions. It is suggested that this standard cable be used for most of the cable run.

Terminal Block. We offer a high quality 68 screw terminal block that directly connects to the SCSI II/III cable. The terminal block can mount on standard DIN rails. HDEterm68
[http://www.dyneng.com/HDEterm68.html]

We provide the components. You provide the system. Safety and reliability can be achieved only by careful planning and practice. Inputs can be damaged by static discharge, or by applying voltage outside of the RS-485 devices rated voltages.

Construction and Reliability

PMC Modules were conceived and engineered for rugged industrial environments. The PMC-BiSerial is constructed out of 0.062 inch thick FR4 material.

Through hole and surface mounting of components are used. IC sockets use gold plated screw machine pins. High insertion and removal forces are required, which assists in the retention of components. If the application requires unusually high reliability or is in an environment subject to high vibration, the user may solder the corner pins of each socketed IC into the socket, using a grounded soldering iron.

The PMC connectors are rated at 1 Amp per pin, 100 insertion cycles minimum. These connectors make consistent, correct insertion easy and reliable.

The PMC secured against the carrier with four screws attached to the 2 stand-offs and 2 locations on the front panel. The four screws provide significant protection against shock, vibration, and incomplete insertion.

The PMC Module provides a low temperature coefficient of 2.17 W / ^{o}C for uniform heat. This is based upon the temperature coefficient of the base FR4 material of 0.31 W / m-^{o}C, and taking into account the thickness and area of the PMC. The coefficient means that if 2.17 Watts are applied uniformly on the component side, then the temperature difference between the component side and solder side is one degree Celsius.

Thermal Considerations

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

Warranty and Repair

Dynamic Engineering warrants this product to be free from defects in workmanship and materials under normal use and service and in its original, unmodified condition, for a period of one year from the time of purchase. If the product is found to be defective within the terms of this warranty, Dynamic Engineering's sole responsibility shall be to repair, or at Dynamic Engineering's sole option to replace, the defective product. The product must be returned by the original customer, insured, and shipped prepaid to Dynamic Engineering. All replaced products become the sole property of Dynamic Engineering.

Dynamic Engineering's warranty of and liability for defective products is limited to that set forth herein. Dynamic Engineering disclaims and excludes all other product warranties and product liability, expressed or implied, including but not limited to any implied warranties of merchandisability or fitness for a particular purpose or use, liability for negligence in manufacture or shipment of product, liability for injury to persons or property, or for any incidental or consequential damages.

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

Service Policy

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

Out of Warranty Repairs

Out of warranty repairs will be billed on a material and labor basis. The current minimum repair charge is $100. Customer approval will be obtained before repairing any item if the repair charges will exceed one half of the quantity one list price for that unit. Return transportation and insurance will be billed as part of the repair and is in addition to the minimum charge.

For Service Contact:

Customer Service Department
Dynamic Engineering
435 Park Dr.
Ben Lomond, CA 95005
831-336-8891
831-336-3840 fax
support@dyneng.com

Specifications

Host Interface:	PCI Mezzanine Card
Serial Interface:	RS-485 Tx_Data, Tx_Clk1, Tx_Clk2, Tx_Sync, Rx_Data1, Rx_Clk1, Rx_Data2, Rx_Clk2, Rx_Sync, REFCLKI
TX CLK rates generated:	12 bit divisor with 10 MHz, 33 MHz. and REFCLK input rates. Other rates available with special oscillator installation
Software Interface:	Control Registers, Status Ports, FIFO
Initialization:	Hardware Reset forces all registers to 0.
Access Modes:	LW boundary Space (see memory map)
Wait States:	1 for all addresses except FIFO loop-back which requires 8.
Interrupt:	Tx interrupt at end of transmission Programmable Almost Empty Programmable Almost Full Error condition Rx protocol 2
DMA:	No DMA Support implemented at this time
Onboard Options:	All Options are Software Programmable
Interface Options:	68 pin twisted pair cable 68 screw terminal block interface
Dimensions:	Standard Single PMC Module.
Construction:	FR4 Multi-Layer Printed Circuit, Through Hole and Surface Mount Components. Programmable parts are socketed.
Temperature Coefficient:	0.89 W / ^{o}C for uniform heat across PMC
Power:	Max. **TBD** mA @ 5V

Order Information

PMC-BISERIAL-IO-BA1	PMC Module with 1 Tx and 2 Rx serial channels, Programmable data rates Dual special protocol support, RS-485 drivers and receivers 32 bit data interface
Eng Kit–PMC-BISERIAL-BA1	HDEterm68 - 68 position screw terminal adapter HDEcabl68 - 68 IO twisted pair cable Technical Documentation, 1. PMC-BISERIAL Schematic 2. PMC-BISERIAL-BA1 Reference test software Data sheet reprints are available from the manufacturer's web site reference software.

Note*: The Engineering Kit is strongly recommended for first time* PMC-BISERIAL *buys.*

Schematics

Schematics are provided as part of the engineering kit for customer *reference only*. This information was current at the time the printed circuit board was last revised. This revision letter is shown on the front of this manual as "Corresponding Hardware Revision." This information is not necessarily current or complete manufacturing data, nor is it part of the product specification.