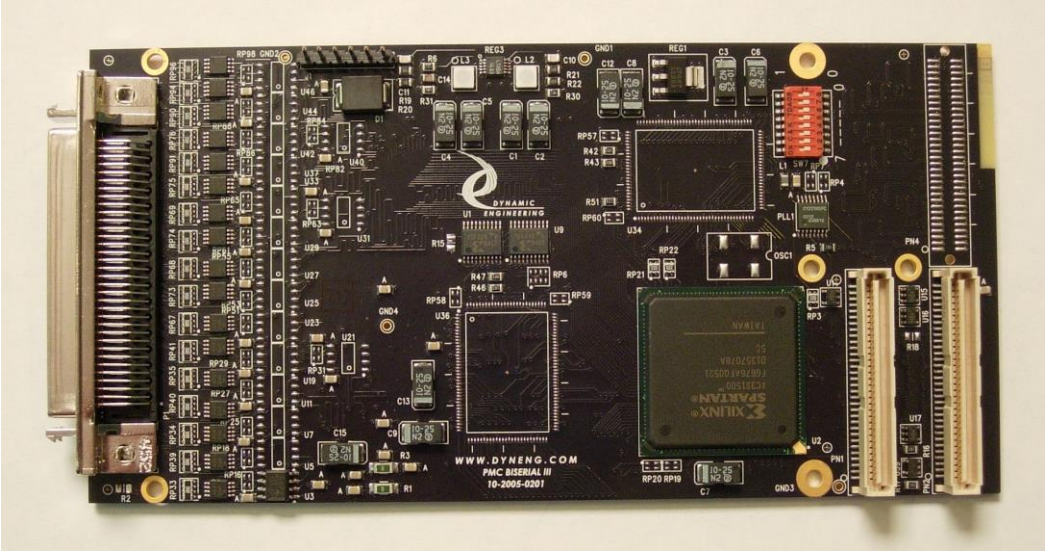
DYNAMIC ENGINEERING
150 DuBois, Suite C Santa Cruz, CA 95060
(831) 457-8891, Fax: (831) 457-4793
www.dyneng.com
sales@dyneng.com
Est. 1988

User Manual

PMC-BiSerial-III SDLC

8 channel
SDLC Interface
PMC Module



Revision A1
Corresponding Hardware: Revision E
10-2005-0205
Corresponding Firmware: Revision B

PMC-BiSerial-III SDLC
PMC Module

Dynamic Engineering
150 DuBois, Suite C
Santa Cruz, CA 95060
(831) 457-8891
FAX: (831) 457-4793

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

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.

Table of Contents

List of Figures

Product Description

The PMC BiSerial-III-SDLC is part of the PMC Module family of modular I/O components by Dynamic Engineering. The PMC BiSerial-III is capable of providing multiple serial protocols. The SDLC protocol implemented provides 8 full-duplex SDLC I/O channels.

Other 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.

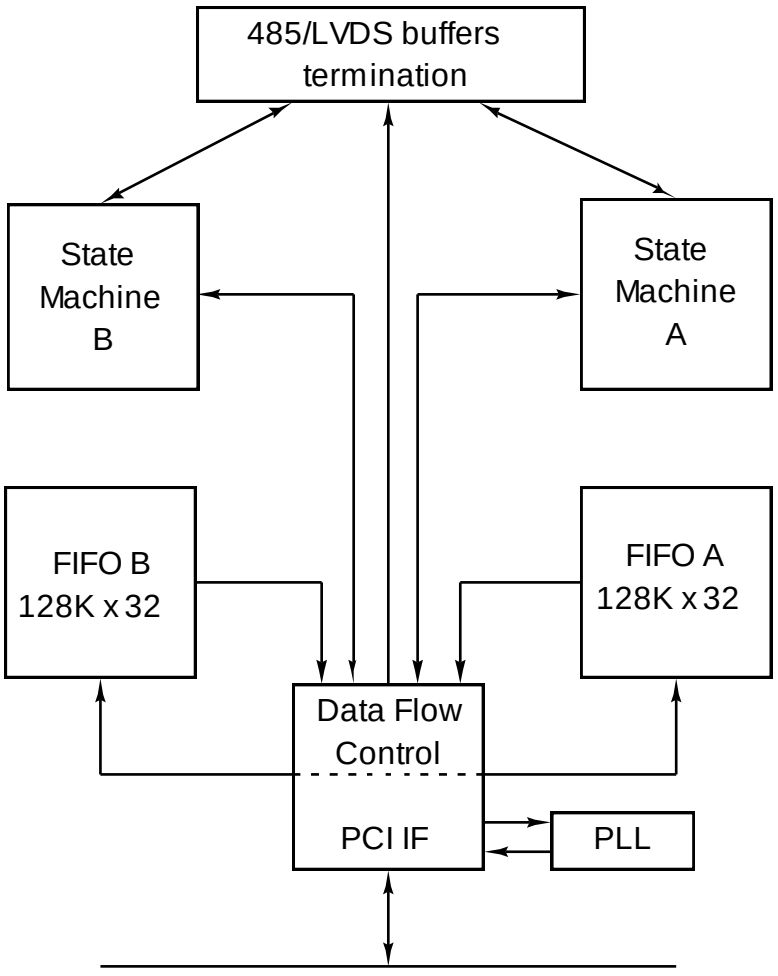


FIGURE 1 PMC BISERIAL-III BASE BLOCK DIAGRAM

The PMC BiSerial-III 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 uses a different one. Contact Dynamic Engineering for a copy of this specification. It is assumed that the reader is at least casually familiar with this document and basic logic design.

In standard configuration, the PMC BiSerial-III is a Type 1 mechanical with only low-profile components on the back of the board and one slot wide, with 10 mm inter-board height 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-III, please let us know. We may be able to do a special build with a different height connector to compensate.

The standard configuration shown in Figure 1 makes use of two external (to the Xilinx) FIFOs. The FIFOs can be as large as 128K deep x 32 bits wide. Some designs do not require so much memory, and are more efficiently implemented using the Xilinx internal memory.

The SDLC implementation has sixteen 4 Kbyte Dual Port RAM (DPR) blocks implemented using the Xilinx internal block RAM. Each channel has two associated DPRs. Each DPR is configured to have a 32-bit port on the PCI side, and a 16-bit port on the I/O side. See Figure 2 for a representation of the SDLC circuit.

The SDLC interface uses programmable PLL clock A as a reference frequency to sample the internal or external transmitter clock. Clock and data in and out comprise the four I/O lines of each SDLC channel. The two DPRs are partitioned into one block each for transmit and receive. The RAM blocks are used as circular buffers that have independently specified start and stop addresses and separate transmit and receive interrupts.

All the data I/O lines on the SDLC are programmable to be register controlled or state-machine controlled. Any or all of the bits can be used as a parallel port instead of being dedicated to a specific I/O protocol. Thirty-four differential I/O are provided at the front bezel (32 of the 34 at Pn4) for 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 100Ω. The termination resistors are in two-element packages to allow flexible termination options for custom formats and protocols. Optional pull-up/pull-down resistor packs can also be installed to provide a logic ‘1’ when the lines are not driven. The terminations and transceivers are programmable through the Xilinx device to provide the proper mix of outputs and inputs and terminations needed for a specific protocol implementation. The terminations are programmable for all I/O.

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

Interrupts are supported by the PMC BiSerial-III-SDLC. An interrupt can be configured to occur at the end of each transmitted message-frame, at the end of all message-frames transmitted, at the end of a received message-frame or when an abort character has been received. All interrupts can be individually masked, and a master interrupt enable is also provided to disable all interrupts simultaneously. The current interrupt status is available whether an individual interrupt is enabled or not making it possible to operate in polled mode. I2O interrupt processing is also implemented.

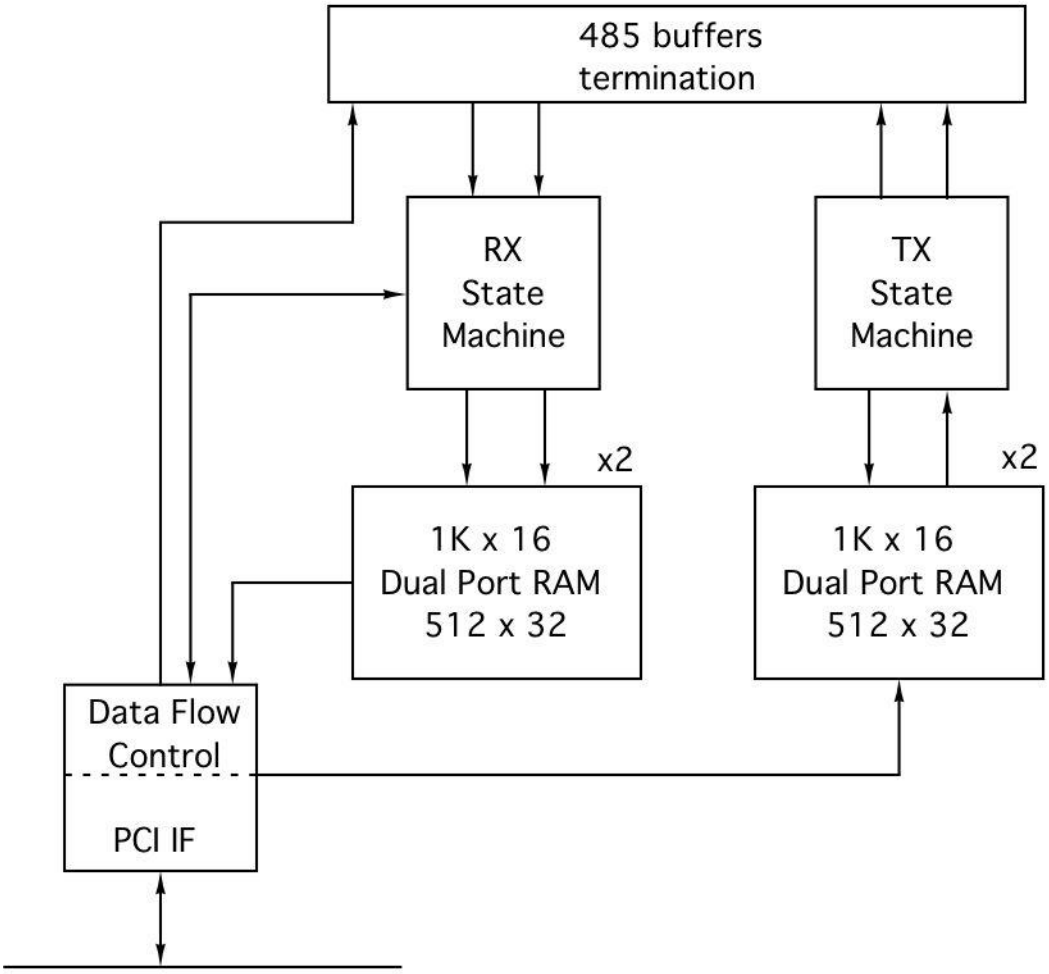


FIGURE 2 PMC BISERIAL-III SDLC BLOCK DIAGRAM

Theory of Operation

The PMC BiSerial-III-SDLC features a Xilinx FPGA. The FPGA contains all of the registers and protocol controlling elements of the BiSerial III design. Only the PLL, transceivers, and switches are external to the Xilinx device.

The PCI interface to the host CPU is controlled by a logic block within the Xilinx. The BiSerial III design requires one wait state for read or write cycles to any address. The PMC BiSerial-III is capable of supporting 40M Bytes per second into and out of the DPR. With a Windows® read/write loop better than 20 MB/sec is attained on most computers. The wait states refer to the number of clocks after the PCI core decode 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 BiSerial III can support many protocols. The PMC BiSerial-III-SDLC supports eight-channel full-duplex SDLC interfaces. This is a synchronous interface with separate clock and data inputs and outputs. Each message is delimited by eight-bit flag characters. The beginning flag and the ending flag enclose the SDLC frame. Both beginning and ending flags have the binary format 01111110. The ending flag for one frame may serve as the beginning flag for the next frame. Alternatively, the ending zero of an ending flag may serve as the beginning zero of a beginning flag, thus forming the pattern ‘011111101111110’. Also, the transmitter may insert multiple flags between frames to maintain the active state of the link if time fill between message frames is required. In order to avoid false flag detection from the data pattern, the SDLC interface uses zero insertion. If five consecutive ones appear anywhere in the data stream, a zero is inserted to avoid having six consecutive one bits. On the receive side, when five ones are received the sixth bit is monitored. If it is a zero, it is removed from the data stream, if it is a one then either a start/stop flag or an abort character (0xFE) has been detected. Any ending flag may be followed by a frame, by another flag, or by an idle condition. The idle condition is signaled by a minimum of 15 consecutive one bits. As long as one bits continue to be sent, the link remains in the idle state.

To send a message, write the message data to the transmit DPRs, specify the start and stop addresses and configuration control bits, then enable the transmitter. The state-machine will load the start address, send the beginning flag character and then send the data sequentially LSB first until the end address is reached and the ending flag is sent. As soon as the beginning flag is sent, the sending status bit will be asserted. At that time the ending address will be latched in the transmitter and new addresses can be written for the next message to be sent. This message will be sent as soon as the current message completes. If a new transmit starting address is not written, the transmitter will continue reading data with the next address after the stop address of the current frame. A new transmit end address must be written to trigger sending an additional message-frame.

If the TX clear is enabled, the transmitter will be automatically disabled and the TX interrupt will be asserted when no more message frames have been requested. If the TX clear is not enabled, the transmitter will remain enabled after the last message, but the TX interrupt will still be asserted. When multiple frames are being sent, the frame done interrupt will be asserted at the end of each message-frame. The TX interrupt will only occur after the last frame and the transmitter will wait, pointing at the next address after the end address. If additional data has been or is later written to the DPR, a new message can be started by entering a new end address (and optionally a new start address). The transmit state-machine will then start the new message and continue sending data until the new end address has been reached. If the end address of the message is less than the start address of the message, when the end of the DPR is reached the transmitter will proceed to the beginning of the DPR and continue until the end address is reached.

To receive a message the receiver must be enabled, but only the starting address of the receive buffer is specified. Data will be stored sequentially in the next address after that starting address and so on until the closing flag is detected. This will latch a receiver done interrupt status and can cause an interrupt if enabled. The last address that data was stored in is written to the starting address location for that message-frame. This allows any received message to be quickly accessed in the received data by reading the address pointer in the message start location, which points to the end address of the first message-frame. The memory location following the end of the first message-frame contains the end address of the second message-frame. This process can be repeated as many times as needed to find the message of interest. At the end of each frame, the end address is also latched and can be read from the control register as a read-only field, but this will be overwritten as subsequent frames complete. The transmit interrupt is mapped to the first interrupt line of the selected channel, the transmit frame done interrupt is mapped to the second interrupt line, the receive interrupt is mapped to the third interrupt line and the abort received interrupt is mapped to the fourth interrupt line of the selected channel.

When a frame completes and no more message-frames are pending, the bus can stay active by continually sending flags or it can go idle by sending ones. The SDLC Idle After Frame Done control bit determines this behavior for the transmitter. If this bit is not set and the bus remains active by sending multiple flags, the Repeated Flags Share Zero control bit determines whether the transmitter sends a ‘0111111001111110’ or a ‘011111101111110’ pattern while waiting for a new message-frame to be requested. When the transmitter is disabled the bus defaults to a high state, which is equivalent to the idle condition.

The PLL is configured to supply a 48 MHz signal on its clock A output. This is used to sample the transmit reference clock to detect transitions. These transitions are used to determine when to drive the next data bit onto the transmit data I/O line. The transmitter clock reference can be supplied by an external source or an internal clock reference provided by PLL clock B. For test purposes, a substitute external clock is created by routing the output from PLL clock B onto I/O 32 and 33 configured as outputs. These clocks may be connected externally to any or all selected channels for loopback testing. A control bit in each channels control register is used to select between these two options. When the internal clock mode is selected the transmit clock line is configured as an output, but when the external clock mode is selected the transmit clock line is configured as an input. The transmit data line is always an output and the receive clock and data lines are always inputs.

Address Map

BIS3_BASE	0x0000	0	Base control register
BIS3_ID	0x0004	1	ID register
BIS3_IO_DATA	0x0010	4	Data register 31 - 0
BIS3_IO_DIR	0x0014	5	Direction register 31 - 0
BIS3_IO_TERM	0x0018	6	Termination register 31 - 0
BIS3_IO_MUX	0x001C	7	Mux register 31 - 0
BIS3_IO_UCNTL	0x0020	8	Upper control register 33, 32
BIS3_SWITCH	0x0024	9	User switch value
BIS3_PLL_CMD	0x0028	10	PLL control register and read-back of PLL data
BIS3_PLL_RDBK	0x002C	11	PLL control register read-back
BIS3_SDLC_CNTL_0	0x0040	16	Chan 0 SDLC control read/write port
BIS3_SDLC_CNTL_1	0x0050	20	Chan 1 SDLC control read/write port
BIS3_SDLC_CNTL_2	0x0060	24	Chan 2 SDLC control read/write port
BIS3_SDLC_CNTL_3	0x0070	28	Chan 3 SDLC control read/write port
BIS3_SDLC_CNTL_4	0x0080	32	Chan 4 SDLC control read/write port
BIS3_SDLC_CNTL_5	0x0090	36	Chan 5 SDLC control read/write port
BIS3_SDLC_CNTL_6	0x00A0	40	Chan 6 SDLC control read/write port
BIS3_SDLC_CNTL_7	0x00B0	44	Chan 7 SDLC control read/write port
BIS3_IO_RDBK	0x00C0	48	External I/O read register
BIS3_IO_RDBKUPR	0x00C4	49	External I/O upper bits read register
BIS3_INT_STAT	0x00CC	51	Interrupt status and clear register
BIS3_I2OAR	0x00D4	53	I2O address storage register
BIS3_TX_MEM_0	0x01000		Dual-port RAM 0 read/write port
BIS3_RX_MEM_0	0x02000		Dual-port RAM 1 read/write port
BIS3_TX_MEM_1	0x03000		Dual-port RAM 2 read/write port
BIS3_RX_MEM_1	0x04000		Dual-port RAM 3 read/write port
BIS3_TX_MEM_2	0x05000		Dual-port RAM 4 read/write port
BIS3_RX_MEM_2	0x06000		Dual-port RAM 5 read/write port
BIS3_TX_MEM_3	0x07000		Dual-port RAM 6 read/write port
BIS3_RX_MEM_3	0x08000		Dual-port RAM 7 read/write port
BIS3_TX_MEM_4	0x09000		Dual-port RAM 8 read/write port
BIS3_RX_MEM_4	0x0A000		Dual-port RAM 9 read/write port
BIS3_TX_MEM_5	0x0B000		Dual-port RAM 10 read/write port
BIS3_RX_MEM_5	0x0C000		Dual-port RAM 11 read/write port
BIS3_TX_MEM_6	0x0D000		Dual-port RAM 12 read/write port
BIS3_RX_MEM_6	0x0E000		Dual-port RAM 13 read/write port
BIS3_TX_MEM_7	0x0F000		Dual-port RAM 14 read/write port
BIS3_RX_MEM_7	0x10000		Dual-port RAM 15 read/write port

FIGURE 3 PMC BISERIAL-III SDLC INTERNAL ADDRESS MAP

The address map provided is for the local decoding performed within the PMC BiSerial-III-SDLC. The addresses are all offsets from a base address, which is assigned by the system when the PCI bus is configured.

Programming

Programming the PMC BiSerial-III-SDLC requires only the ability to read and write data from the host. The base address of the module refers to the first user address for the slot in which the PMC is installed. This address is determined during system configuration of the PCI bus.

Depending on the software environment it may be necessary to set-up the system software with the PMC BiSerial-III "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 initialize the receiver buffer start address and enable the Rx channel. To transmit the software will need to load the message into the appropriate Dual Port RAM, set the transmitter buffer start and end address and any configuration parameters and enable the transmitter.

When a received message completes, the end address of the message will be written to the receiver buffer start address with the received data stored starting with the next address. The next message will be stored starting with the following address unless a new starting address has been written after the first message has begun. The end address of each received message can also be read from the address field of the channel control register, but this will be over-written when the next message completes.

Once the transmitter starts sending a message, a new end address (and optionally a new start address) can be written to send subsequent messages. Multiple messages can be loaded into the transmitter RAM and sent in any order desired.

The interrupt service routine should be loaded and the interrupt mask set. The interrupt service routine can be configured to respond to the channel interrupts on an individual basis. After the interrupt is received, the data can be retrieved. An efficient loop can then be implemented to fetch the data. New messages can be received even as the current one is read from the Dual Port RAM.

The TX interrupt indicates that a message has been sent and that the message has completed. If more than one interrupt is enabled, then the interrupt service routine (ISR) needs to read the status to see which source caused the interrupt. The status bits are latched, and are explicitly cleared by writing a one to the corresponding bit. It is a good idea to read the status register and write that value back to clear all the latched interrupt status bits before starting a transfer. This will insure that the interrupt status values read by the ISR came from the current transfer.

VendorId = 0xDCBA, CardId = 0x005A
Flash design ID = 0x0003, Current Flash revision = 0x0002

Register Definitions

BIS3_BASE

[$00] BiSerial III Base Control Register Port read/write

Base Control Register	
DATA BIT	**DESCRIPTION**
31-4	Spare
3	I2O CLR
2	I2O EN
1	Interrupt Set
0	Interrupt Enable Master

FIGURE 4 PMC BISERIAL-III SDLC BASE CONTROL REGISTER BIT MAP

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

Interrupt Enable Master: When '1' allows interrupts generated by the PMC-BiSerial-III-SDLC to be driven onto the carrier (INTA). When '0' the interrupts can be individually enabled and used for status without driving the backplane. Polled operation can be performed in this mode.

Interrupt Set: When '1' and the Master is enabled, this bit forces an interrupt request. This feature is useful for testing and software development.

I2O EN: When '1' allows the I2O interrupts to be activated. Interrupt requests are routed to the address stored in the I2O Address Register (I2OAR). When '0' the I2O function is disabled.

I2O CLR: When '1' this bit will cause the current data stored in the I2O collection register to be cleared. It is recommended that this register clear bit be used immediately before enabling I2O operation to prevent previously stored events from causing interrupts.

BIS3_ID

[$04] BiSerial III FLASH status/Driver Status Port read only

Design Number / FLASH Revision	
DATA BIT	**DESCRIPTION**
31-16	Design/Driver ID
15-0	FLASH revision

FIGURE 5 PMC BISERIAL-III SDLC DESIGN ID REGISTER BIT MAP

The Design/Driver ID for the SDLC project is 0x0003. The FLASH revision is currently 0x0001, but will be updated as features are added or revisions made.

BIS3_IO_DATA

[$10] BiSerial III Parallel Data Output Register read/write

Parallel Data Output Register	
DATA BIT	**DESCRIPTION**
31-0	parallel output data

FIGURE 6 PMC BISERIAL-III SDLC PARALLEL OUTPUT DATA BIT MAP

There are 32 potential output bits in the parallel port. The Direction, Termination, and Mux Control registers are also involved. When the direction is set to output, and the Mux control set to parallel port the bit definitions from this register are driven onto the corresponding parallel port lines.

This port is direct read/write of the register. The I/O side is read-back from the BIS3_IO_RDBK port. It is possible that the output data does not match the I/O data in the case of the Direction bits being set to input or the Mux control set to state-machine.

BIS3_IO_DIR

[$14] BiSerial III Direction Port read/write

Direction Control Port	
DATA BIT	**DESCRIPTION**
31-0	Parallel Port Direction Control bits

FIGURE 7 PMC BISERIAL-III SDLC DIRECTION CONTROL PORT

When set (‘1’) the corresponding bit in the parallel port is a transmitter. When cleared (‘0’) the corresponding bit is a receiver. The corresponding Mux control bits must also be configured for parallel port.

BIS3_IO_TERM

[$18] BiSerial III Termination Port read/write

Termination Control Port	
DATA BIT	**DESCRIPTION**
31-0	Parallel Port Termination Control bits

FIGURE 8 PMC BISERIAL-III SDLC TERMINATION CONTROL PORT

When set (‘1’) the corresponding I/O line will be terminated. When cleared (‘0’) the corresponding I/O line is not terminated. These bits are independent of the Mux control definitions. When a bit is set to be terminated; the analog switch associated with that bit is closed to create a parallel termination of approximately 100 Ω. In most systems the receiving side is terminated, and the transmitting side is not. The drivers can handle termination on both ends.

BIS3_IO_MUX

[$1C] BiSerial III Mux Port read/write

Multiplexor Control Port	
DATA BIT	**DESCRIPTION**
31-0	Parallel Port Mux Control bits

FIGURE 9 PMC BISERIAL-III SDLC MUX CONTROL PORT

When set (‘1’) the corresponding bit is set to State-Machine control. When cleared (‘0’) the corresponding bit is set to parallel port operation. The Mux control definition along with the Data, Direction and Termination registers allows for a bit-by-bit selection of operation under software control.

BIS3_IO_UCNTL

[$20] BiSerial III Upper Control Port read/write

Upper Bits Control Port	
DATA BIT	**DESCRIPTION**
25-24	Mux 33, 32
17-16	Termination 33, 32
9-8	Direction 33, 32
1-0	Data 32, 32

FIGURE 10 PMC BISERIAL-III SDLC UPPER CONTROL PORT

The BiSerial III has 34 transceivers. The upper control bits are concentrated within this register to cover the top 2 bits not controlled within the other control registers. The upper bits are only useable on the Bezel I/O connector. Pn4 has only 64 connections and doesn’t support the upper lines. The definitions are the same as the Data, Term, Dir and Mux port definitions for bit operation.

Data = Data transmitted when the Mux is set to ‘0’ and the direction is set to ‘1’. Termination when set to ‘1’ causes the parallel termination to be engaged. Setting the Mux control bits to ‘0’ creates a parallel port for those bits. Setting the Mux control bits to ‘1’ enables the state-machine to control the direction and data lines. The termination control is independent.

BIS3_IO_RDBK

[$C0] BiSerial III I/O Read-Back Port read only

I/O Read-Back Port	
DATA BIT	**DESCRIPTION**
31-0	I/O Data 31-0

FIGURE 11 PMC BISERIAL-III SDLC I/O READBACK PORT

The I/O lines can be read at any time. The value is not filtered in any way. If the transceivers are set to TX by the parallel port or state-machine then the read-back value will be the transmitted value. If the transceivers are set to receive then the port values will be those received by the transceivers from the external I/O.

BIS3_IO_RDBKUPR

[$C4] BiSerial III I/O Upper Read-Back Port read only

I/O Upper Read-Back Port	
DATA BIT	**DESCRIPTION**
1-0	I/O Data 33-32

FIGURE 12 PMC BISERIAL-III SDLC I/O READBACK PORT

The I/O lines can be read at any time. The value is not filtered in any way. If the transceivers are set to TX by the parallel port or state-machine then the read-back value will be the transmitted value. If the transceivers are set to receive then the port values will be those received by the transceivers from the external I/O. The upper bits are presented on this port.

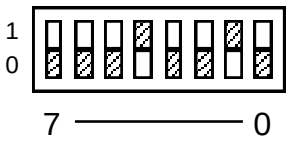
BIS3_SWITCH

[$24] BiSerial III Switch Port read only

User Switch Port	
DATA BIT	**DESCRIPTION**
31-24	Spare
23-16	sw7-0
15-0	Spare

FIGURE 13 PMC BISERIAL-III SDLC SWITCH PORT

The Switch Read Port has the user bits. The user bits are connected to the eight dip-switch positions. The switches allow custom configurations to be defined by the user and for the software to identify a particular board by its switch settings and to configure it accordingly.



The Dip-switch is marked on the silk-screen with the positions of the digits and the '1' and '0' definitions. The numbers are hex coded. The example shown would produce 0x12 when read (and shifted down).

BIS3_PLL_CMD, PLL_RDBK

[$28, 2C] BiSerial III PLL Control

PLL Command Register, PLL CMD Read-back

DATA BIT	DESCRIPTION
3	PLL Enable
2	PLL S2
1	PLL SCLK
0	PLL SDAT

FIGURE 14 PMC BISERIAL-III SDLC PLL CONTROL

The register bits for PLL Enable, PLL S2, and PLL SCLK are unidirectional from the Xilinx to the PLL – always driven. SDAT is open drain. The SDAT register bit when written low and enabled will be reflected with a low on the SDAT signal to the PLL. When SDAT is taken high or disabled the SDAT signal will be tri-stated by the Xilinx, and can be driven by the PLL. The SDAT register bit when read reflects the state of the SDAT signal between the Xilinx and PLL and can be in a different state than the written SDAT bit. To read back the contents of the CMD port use the RDBK port.

PLL Enable: When this bit is set to a one, SDAT is enabled. When set to '0' SDAT is tri-stated by the Xilinx.

PLL SCLK/SDAT: These signals are used to program the PLL over the I2C serial interface. SCLK is always an output whereas SDAT is bi-directional. When SDAT is to be read from the PLL

PLL S2: This is an additional control line to the PLL that can be used to select alternative pre-programmed frequencies.

The PLL is a separate device controlled by the Xilinx. The PLL has a fairly complex programming requirement which is simplified by using the Cypress® CyberClocks utility, and then programming the resulting control words into the PLL using this PLL Control port. The interface can be further simplified by using the Dynamic Engineering driver to take care of the low-level bit manipulation requirements.

BIS3_SDLC_CNTL7-0

[$B0, A0, 90, 80, 70, 60, 50, 40] BiSerial III SDLC Control Registers

SDLC Control Registers

DATA BIT	DESCRIPTION
31	Idle Detected/Clear (see note after description)
30	Abort Detected/Clear (see note after description)
29-25	spare
24	SDLC Internal Clock Select
23	Send an Abort (write only)
22	Load Transmit End Address (write only)
21	Load Transmit Start Address/SDLC Done
20	Load Receive Start Address/SDLC Sending Data
19	SDLC Idle After Frame Done
18-8	Address Input/ Receive End Address
7	Repeated Flags Share Zero
6	Received Abort Interrupt Enable
5	Receive Interrupt Enable
4	Transmit Frame Done Interrupt Enable
3	Transmit Interrupt Enable
2	Transmit Clear Enable
1	Receive Enable
0	Transmit Enable

FIGURE 15 PMC BISERIAL-III SDLC CONTROL REGISTERS

Transmit Enable: When this bit is a one the transmitter is enabled to send data starting with the address stored in the transmitter start-address register and continuing until the data at the address in the transmitter end-address register has been sent. When this bit is a zero the transmitter is disabled.

Receive Enable: When this bit is a one the receiver is enabled to receive data and store it in the dual-port RAM starting with the address stored in the receiver start-address register if it is the first message since the receiver was enabled, or in the next 16-bit address after the end-address of the last message if it is not. When this bit is a zero the receiver is disabled.

Transmit Clear Enable: When this bit is a one the transmit enable bit will be cleared when the transmitted message completes and there is not another message pending. When this bit is a zero the transmitter will remain enabled, but no more data will be sent until a new end address is loaded.

Transmit Interrupt Enable: When this bit is a one the transmitter interrupt is enabled. The interrupt will occur at when the transmit state-machine reaches the end address stored in the transmitter end-address register and there is not another message pending. When this bit is a zero the interrupt status will still be latched, but will not cause an interrupt to occur. The transmit interrupt is mapped to the first interrupt line in its channel block.

Transmit Frame Done Interrupt Enable: When this bit is a one the transmit frame done interrupt is enabled. This interrupt will occur when each message frame completes regardless of whether another message is pending. When this bit is a zero the interrupt status will still be latched, but will not cause an interrupt to occur. The transmit frame done interrupt is mapped to the second interrupt line in its channel block.

Receive Interrupt Enable: When this bit is a one the receiver interrupt is enabled. The interrupt will occur at the end of a message transmission, which is determined by the detection of a SDLC flag character (0x7e) after the message has started. When this bit is a zero the interrupt status will still be latched, but will not cause an interrupt to occur. The receive interrupt is mapped to the third interrupt line in its channel block.

Received Abort Interrupt Enable: When this bit is a one, the received abort interrupt is enabled. This interrupt will occur when an SDLC abort character (0x7f) is received. When this bit is a zero the abort interrupt status will still be latched, but will not cause an interrupt to occur. The received abort interrupt is mapped to the fourth interrupt line in its channel block.

Repeated Flags Share Zero: When this bit is a one and the transmitter is sending repeated flag characters, the last zero in each flag will also serve as the first zero in the next flag. This is only true for two successive flags, the last flag before data is sent will be sent entirely. When this bit is a zero, all eight bits of each flag will be sent regardless of adjacent characters.

Address Input/Receive End Address: This field is used with the three load address bits to specify address boundaries for the transmitter and receiver data buffers. When this field is read, it represents the address in which the last received data word from the last message-frame is stored. Note that this is a 16-bit address, bit 0 indicates which half of the appropriate long-word the last 16-bit word was stored (0 -> lower half, 1 -> upper half).

SDLC Idle After Frame Done: When this bit is a one, the SDLC link will go to the idle state (minimum of 15 consecutive ones) when message transmission completes. The link will remain high until a new message is requested. When this bit is zero and the transmitter remains enabled, the transmitter will send repeated flags until a new message is requested.

Load Receive Start Address/SDLC Sending Data: When this bit is a one the value in the address input field is loaded into the receiver start-address register. When this bit is a zero no action is taken. When this bit is read as a one, the transmitter is actively sending data. At this time new addresses can be written for the next message-frame to be sent. A new transmitter end address is required to queue a new message-frame. New transmit or receive start addresses are optional. If new start addresses are not written, the transmitter and/or receiver will continue reading/storing data at the next address after the end address of the last message frame. When this bit is a zero, the link is either idle, aborted or sending repeated flags.

Load Transmit Start Address/SDLC Frame Done: When this bit is a one the value in the address input field is loaded into the transmitter start-address register. When this bit is a zero no action is taken. When this bit is read as a one, it indicates that the last message has completed. This bit is latched and will be cleared by any write to this control register. An interrupt can be configured to occur when this bit goes high by asserting the transmit frame done interrupt enable. When this bit is read as a zero, a message-frame has not completed since the last write to the SDLC control register.

Load Transmit End Address (write only): When this bit is a one the value in the address input field is loaded into the transmitter end-address register. When this bit is a zero no action is taken.

Send an Abort (write only): When this bit is set to a one the transmit state-machine will send an abort character (0xfe) provided a transmission is currently in progress. When this bit is a zero normal operation will continue.

SDLC Internal Clock Select: When this bit is a one, the transmitter will use the internal transmit clock as a reference for sending SDLC data. When this bit is a zero, an external received clock will be used as the reference for data transmission.

Abort Detected/Clear: When an abort character is detected by the receiver, this status bit will be latched and can be cleared by writing a one back in this bit position. When this bit is a zero, no abort has been detected since the latch was last cleared.

Idle Detected/Clear: When an idle bus state is detected by the receiver, this status bit will be latched and can be cleared by writing a one back in this bit position. When this bit is a zero, the bus has not idled since the latch was last cleared.

Note: Writing Abort Clear or Idle Clear disables updating any other bits in the control register. So when these latched bits are cleared it must be the only action performed with that register access.

BIS3_INT_STAT

[$CC] BiSerial III Interrupt Status and Clear Register

Interrupt Status and Clear Register	
DATA BIT	**DESCRIPTION**
31-0	Channel Interrupt or Clear bit

FIGURE 16 PMC BISERIAL-III SDLC INTERRUPT STATUS REGISTER

Each bit is set when an interrupt occurs on the associated channel. Each bit can be cleared by writing to the register with the same bit position set (‘1’). You do not need to rewrite with a ‘0’ – the clearing action happens during the write.

This register is in parallel with the I2O interrupts. Usually only one or the other will be in use at a time. Both can be used if desired. Interrupt conditions are captured and processed in both places.

BIS3_I2OAR

[$D4] BiSerial III I20 Address Register

I2O Address Register	
DATA BIT	**DESCRIPTION**
31-0	Address

FIGURE 17 PMC BISERIAL-III SDLC I2O ADDRESS REGISTER

The physical address where the I2O interrupt status should be written to is stored in this register. When active interrupts are detected the I2O sequence is started. The PCI bus is requested, the hardware waits for the grant and then writes the captured status to the stored address. Please note that this is the direct hardware address, and not an indirect (translated) address.

The active bits are auto cleared and the process re-enabled for new active interrupts. Interrupts that occur during an I2O cycle are stored until the hardware is re-enabled and causes a second immediate processing cycle. The receiving hardware must be able to handle multiple interrupt status writes in close succession. A FIFO is ideal for the receiving hardware implementation.

Resource Mapping

I/O Mapping for SDLC Channels		Interrupt Mapping for SDLC Channels	
I/O line	SDLC Function	Int line	SDLC Function
I/O 0	Transmit Data 0	Int 0	Transmit Interrupt
I/O 1	Receive Data 0	Int 1	TX Frame Done
I/O 2	Transmit Clock 0	Int 2	Receive Interrupt
I/O 3	Receive Clock 0	Int 3	RX Abort Detected
I/O 4	Transmit Data 1	Int 4	Transmit Interrupt
I/O 5	Receive Data 1	Int 5	TX Frame Done
I/O 6	Transmit Clock 1	Int 6	Receive Interrupt
I/O 7	Receive Clock 1	Int 7	RX Abort Detected
I/O 8	Transmit Data 2	Int 8	Transmit Interrupt
I/O 9	Receive Data 2	Int 9	TX Frame Done
I/O 10	Transmit Clock 2	Int 10	Receive Interrupt
I/O 11	Receive Clock 2	Int 11	RX Abort Detected
I/O 12	Transmit Data 3	Int 12	Transmit Interrupt
I/O 13	Receive Data 3	Int 13	TX Frame Done
I/O 14	Transmit Clock 3	Int 14	Receive Interrupt
I/O 15	Receive Clock 3	Int 15	RX Abort Detected
I/O 16	Transmit Data 4	Int 16	Transmit Interrupt
I/O 17	Receive Data 4	Int 17	TX Frame Done
I/O 18	Transmit Clock 4	Int 18	Receive Interrupt
I/O 19	Receive Clock 4	Int 19	RX Abort Detected
I/O 20	Transmit Data 5	Int 20	Transmit Interrupt
I/O 21	Receive Data 5	Int 21	TX Frame Done
I/O 22	Transmit Clock 5	Int 22	Receive Interrupt
I/O 23	Receive Clock 5	Int 23	RX Abort Detected
I/O 24	Transmit Data 6	Int 24	Transmit Interrupt
I/O 25	Receive Data 6	Int 25	TX Frame Done
I/O 26	Transmit Clock 6	Int 26	Receive Interrupt
I/O 27	Receive Clock 6	Int 27	RX Abort Detected
I/O 28	Transmit Data 7	Int 28	Transmit Interrupt
I/O 29	Receive Data 7	Int 29	TX Frame Done
I/O 30	Transmit Clock 7	Int 30	Receive Interrupt
I/O 31	Receive Clock 7	Int 31	RX Abort Detected

Channel I/O Line Mapping

SDLC channel 0:

SDLC transmit data => I/O 0:	pin 1 +,	pin 35 –
SDLC receive data => I/O 1:	pin 2 +,	pin 36 –
SDLC transmit clock=> I/O 2:	pin 3 +,	pin 37 –
SDLC receive clock => I/O 3:	pin 4 +,	pin 38 –

SDLC channel 1:

SDLC transmit data => I/O 4:	pin 5 +,	pin 39 –
SDLC receive data => I/O 5:	pin 6 +,	pin 40 –
SDLC transmit clock=> I/O 6:	pin 7 +,	pin 41 –
SDLC receive clock => I/O 7:	pin 8 +,	pin 42 –

SDLC channel 2:

SDLC transmit data => I/O 8:	pin 9 +,	pin 43 –
SDLC receive data => I/O 9:	pin 10 +,	pin 44 –
SDLC transmit clock=> I/O 10:	pin 11 +,	pin 45 –
SDLC receive clock => I/O 11:	pin 12 +,	pin 46 –

SDLC channel 3:

SDLC transmit data => I/O 12:	pin 13 +,	pin 47 –
SDLC receive data => I/O 13:	pin 14 +,	pin 48 –
SDLC transmit clock=> I/O 14:	pin 15 +,	pin 49 –
SDLC receive clock => I/O 15:	pin 16 +,	pin 50 –

SDLC channel 4:

SDLC transmit data => I/O 16:	pin 17 +,	pin 51 –
SDLC receive data => I/O 17:	pin 18 +,	pin 52 –
SDLC transmit clock=> I/O 18:	pin 19 +,	pin 53 –
SDLC receive clock => I/O 19:	pin 20 +,	pin 54 –

SDLC channel 5:

SDLC transmit data => I/O 20:	pin 21 +,	pin 55 –
SDLC receive data => I/O 21:	pin 22 +,	pin 56 –
SDLC transmit clock=> I/O 22:	pin 23 +,	pin 57 –
SDLC receive clock => I/O 23:	pin 24 +,	pin 58 –

SDLC channel 6:

SDLC transmit data => I/O 24:	pin 25 +,	pin 59 –
SDLC receive data => I/O 25:	pin 26 +,	pin 60 –
SDLC transmit clock=> I/O 26:	pin 27 +,	pin 61 –
SDLC receive clock => I/O 27:	pin 28 +,	pin 62 –

SDLC channel 7:

SDLC transmit data => I/O 28:	pin 29 +,	pin 63 –
SDLC receive data => I/O 29:	pin 30 +,	pin 64 –
SDLC transmit clock=> I/O 30:	pin 31 +,	pin 65 –
SDLC receive clock => I/O 31:	pin 32 +,	pin 66 –

SDLC External Clock Test Resources

SDLC test clock => I/O 32:	pin 33 +,	pin 67 –
SDLC test clock => I/O 33:	pin 34 +,	pin 68 –

Interrupts

PMC BiSerial-III interrupts are treated as auto-vectored. When the software enters into an exception handler to deal with a PMC BiSerial-III interrupt the software must read the status register to determine the cause(s) of the interrupt, clear the interrupt request(s) and process accordingly. Power-on initialization will provide a cleared interrupt request and interrupts disabled.

For example, the PMC BiSerial-III TX state machine(s) generates an interrupt request when a transmission is complete, and the TX int enable and Master interrupt enable bits are set. The transmission is considered complete when the last bit is output from the output shift register.

The interrupt is mapped to INTA on the PMC connector, which is mapped to a system interrupt when the PCI bus configures. The source of the interrupt is obtained by reading BIS3_INT_STAT. The status remains valid until that bit in the status register is explicitly cleared.

When an interrupt occurs, the Master interrupt enable should be cleared, and the status register read to determine the cause of the interrupt. Next perform any processing needed to remove the interrupting condition, clear the latched bit and set the Master interrupt enable bit high again.

The individual enables operate after the interrupt holding latches, which store the interrupt conditions for the CPU. This allows for operating in polled mode simply by monitoring the BIS3_INT_STAT register. If one of the enabled conditions occurs, the interrupt status bit will be set, but unless the Master interrupt, and the channel interrupt enable is set, a system interrupt will not occur.

I2O interrupts are also available. Program the Address where the interrupt status should be written to in the I2OAR. Clear any stored interrupts in the I2O register, and then program the I2O enable to be set. The hardware will collect interrupt conditions, and write them to the address stored in the I2OAR. The interrupts will still need to be processed at the hardware level.

Loop-back

The Engineering kit has reference software, which includes an external loop-back test. The SDLC version of the PMC-BiSerial-III utilizes a 68 pin SCSI II front panel connector. The test requires an external cable with the following pins connected. Using our HDEterm68 test fixture make the following connections (TP2 unless noted). **Note**: TP1, 2 are both ordered as follows: 1, 35, 2, 36, 3, 37…32, 66, 33, 67, 34, 68.

SIGNAL	+	-	+	-
Chan 0 Data TX to RX	1	35	2	36
Chan 0 Clock TX to RX	3	37	4	38
Chan 1 Data TX to RX	5	39	6	40
Chan 1 Clock TX to RX	7	41	8	42
Chan 2 Data TX to RX	9	43	10	44
Chan 2 Clock TX to RX	11	45	12	46
Chan 3 Data TX to RX	13	47	14	48
Chan 3 Clock TX to RX	15	49	16	50
Chan 4 Data TX to RX	17	51	18	52
Chan 4 Clock TX to RX	19	53	20	54
Chan 5 Data TX to RX	21	55	22	56
Chan 5 Clock TX to RX	23	57	24	58
Chan 6 Data TX to RX	25	59	26	60
Chan 6 Clock TX to RX	27	61	28	62
Chan 7 Data TX to RX	29	63	30	64
Chan 7 Clock TX to RX	31	65	32	66

Additional I/O lines used for external clock tests-selectively connect to TX Clock lines Eight six-pin headers are installed in TP1 occupying the following positions: Pins 2-38, 6-42, 10-46, 14-50, 18-54, 22-58, 26-62 and 30-66.

External TX Clock Distribution Network to Channels 0-3

SIGNAL	+	-	+ (TP1)	- (TP1)
TX Clock A	33	67	2	38
			6	42
			10	46
			14	50

External TX Clock Distribution Network to Channels 4-7

SIGNAL	+	-	+ (TP1)	- (TP1)
TX Clock B	34	68	18	54
			22	58
			26	62
			30	66

Four short wires per header are added to the bottom of the HDEterm68.

Channel 0: TP1 2 to 36, TP1 38 to 4, TP1 3 to TP2 3 and TP1 37 to TP2 37
Channel 1: TP1 6 to 40, TP1 42 to 8, TP1 7 to TP2 7 and TP1 41 to TP2 41
Channel 2: TP1 10 to 44, TP1 46 to 12, TP1 11 to TP2 11 and TP1 45 to TP2 45
Channel 3: TP1 14 to 48, TP1 50 to 16, TP1 15 to TP2 15 and TP1 49 to TP2 49
Channel 4: TP1 18 to 52, TP1 54 to 20, TP1 19 to TP2 19 and TP1 53 to TP2 53
Channel 5: TP1 22 to 56, TP1 58 to 24, TP1 23 to TP2 23 and TP1 57 to TP2 57
Channel 6: TP1 26 to 60, TP1 62 to 28, TP1 27 to TP2 27 and TP1 61 to TP2 61
Channel 7: TP1 30 to 64, TP1 66 to 32, TP1 31 to TP2 31 and TP1 65 to TP2 65

Installing two shunts per channel on the headers in TP1 will connect the external clock to the respective channels, removing them allows internal clock tests to be performed.

Channel 0: 36-3 and 37-4
Channel 1: 40-7 and 41-8
Channel 2: 44-11 and 45-12
Channel 3: 48-15 and 49-16
Channel 4: 52-19 and 53-20
Channel 5: 56-23 and 57-24
Channel 6: 60-27 and 61-28
Channel 7: 64-31 and 65-32

PMC PCI Pn1 Interface Pin Assignment

The figure below gives the pin assignments for the PMC Module PCI Pn1 Interface on the PMC BiSerial-III. See the User Manual for your carrier board for more information. Unused pins may 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 18 PMC BISERIAL-III PN1 INTERFACE

PMC PCI Pn2 Interface Pin Assignment

The figure below gives the pin assignments for the PMC Module PCI Pn2 Interface on the PMC BiSerial-III. See the User Manual for your carrier board for more information. Unused pins may 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 19 PMC BISERIAL-III PN2 INTERFACE

BiSerial III Front Panel I/O Pin Assignment

The figure below gives the pin assignments for the PMC Module I/O Interface on the PMC BiSerial-III. Also, see the User Manual for your carrier board for more information. For customized version, or other options, contact Dynamic Engineering.

IO_0p	IO_0m	1	35
IO_1p	IO_1m	2	36
IO_2p	IO_2m	3	37
IO_3p	IO_3m	4	38
IO_4p	IO_4m	5	39
IO_5p	IO_5m	6	40
IO_6p	IO_6m	7	41
IO_7p	IO_7m	8	42
IO_8p	IO_8m	9	43
IO_9p	IO_9m	10	44
IO_10p	IO_10m	11	45
IO_11p	IO_11m	12	46
IO_12p	IO_12m	13	47
IO_13p	IO_13m	14	48
IO_14p	IO_14m	15	49
IO_15p	IO_15m	16	50
IO_16p	IO_16m	17	51
IO_17p	IO_17m	18	52
IO_18p	IO_18m	19	53
IO_19p	IO_19m	20	54
IO_20p	IO_20m	21	55
IO_21p	IO_21m	22	56
IO_22p	IO_22m	23	57
IO_23p	IO_23m	24	58
IO_24p	IO_24m	25	59
IO_25p	IO_25m	26	60
IO_26p	IO_26m	27	61
IO_27p	IO_27m	28	62
IO_28p	IO_28m	29	63
IO_29p	IO_29m	30	64
IO_30p	IO_30m	31	65
IO_31p	IO_31m	32	66
IO_32p	IO_32m	33	67
IO_33p	IO_33m	34	68

FIGURE 20 PMC BISERIAL-III FRONT PANEL INTERFACE

Applications Guide

Interfacing

The pin-out 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-III 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. The PMC BiSerial-III 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 III 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 (HDEterm68). The terminal block can mount on standard DIN rails.

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 are conceived and engineered for rugged industrial environments. The PMC BiSerial-III is constructed out of 0.062 inch thick High Temp FR4 material. The PC Boards are ROHS compliant. Dynamic Engineering has selected gold immersion processing to provide superior performance, and reliability (not subject to tin whisker issues).

Through hole and surface mounting of components are used. IC sockets use 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 is 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 III 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

Please refer to the warranty page on our website for the current warranty offered and options.

http://www.dyneng.com/warranty.html

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
150 DuBois, Suite C
Santa Cruz, CA 95060
(831) 457-8891 - Fax (831) 457-4793
support@dyneng.com

Specifications

Host Interface:	(PMC) PCI Mezzanine Card - 32 bit, 33 MHz
Serial Interface:	8 Full Duplex SDLC serial interfaces. 16-bit word size, LSB first.
TX Data rates generated:	40 MHz oscillator used to generate 48 MHz I/O clock sampling frequency, 3 MHz Internal transmit clock. SDLC – PLL clock B for external transmit clock test frequencies.
RX Data rates accepted:	SDLC rates 1-3 MHz accepted
Software Interface:	Control Registers, Status Ports, Dual Port RAM, Driver Available
Initialization:	Hardware reset forces all registers to 0.
Access Modes:	LW boundary Space (see memory map)
Wait States:	1 for all addresses
Interrupt:	TX interrupt at end of message transmission TX interrupt at end of frame transmission RX interrupt at end of message reception RX interrupt when abort received Software interrupt I2O interrupts
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:	2.17 W/^{o}C for uniform heat across PMC
Power:	Max. **TBD** mA @ 5V
Temperature range	0-70 standard, Extended Temperature available (-40 + 85)

Order Information

PMC BiSerial-III-SDLC-rev.A	PMC Module with 8 SDLC channels, 34 bit parallel port (overlaps with serial channels) RS-485 I/O. 32-bit data interface
Eng Kit–PMC BiSerial-III	HDEterm68 - 68 position screw terminal adapter http://www. dyneng. com/HDEterm68. html HDEcabl68 - 68 I/O twisted pair cable http://www. dyneng. com/HDEcabl68. html Technical Documentation, 1. PMC BiSerial-III Schematic 2. PMC BiSerial-III-SDLC Driver software and user application. Data sheet reprints are available from the manufacturer's web site

***Note**: The Engineering Kit is strongly recommended for first time* PMC BiSerial-III *purchases.*

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.