

# 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

# PC104p-BiSerial-III-TG1

4 Channel Instrumentation Receiver  
1 Channel Serial Transmitter

PC/104p Module

Revision A1  
Corresponding Hardware: Revision A  
10-2006-0801

**PC104p-BiSerial-III-TG1**  
4 Channel Instrumentation  
Receiver  
PC/104p Module

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

This document contains information of proprietary interest to Dynamic Engineering. It has been supplied in confidence and the recipient, by accepting this material, agrees that the subject matter will not be copied or reproduced, in whole or in part, nor its contents revealed in any manner or to any person except to meet the purpose for which it was delivered.

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 PC/104p Module stacks and compatible user-provided equipment. Connection of incompatible hardware is likely to cause serious damage.

©2006 by Dynamic Engineering.

Other trademarks and registered trademarks are owned by their respective manufactures.  
Revised June 6, 2006.



---

---

# Table of Contents

---

---

PRODUCT DESCRIPTION	6
THEORY OF OPERATION	9
ADDRESS MAP	11
PROGRAMMING	13
PC104P_BIS3_BASE	14
PC104P_BIS3_ID_SW	14
PC104P_BIS3_STATUS	15
PC104P_BIS3_DIR_TERM	16
PC104P_BIS3_PARDAT_485	17
PC104P_BIS3_PARDAT_485_RDBK	17
PC104P_BIS3_PARCNTL	18
PC104P_BIS3_BASE 0,1,2,3	19
PC104P_BIS3_INT 0,1,2,3	20
PC104P_BIS3_BURSTIN 0,1,2,3	21
PC104P_BIS3_BURSTOUT 0,1,2,3	22
PC104P_BIS3_FIFORW 0,1,2,3	23
PC104P_BIS3_TXAMTC 0,1,2,3	23
PC104P_BIS3_RXAFC 0,1,2,3	24
PC104P_BIS3_TX FIFO WORD COUNT 0,1,2,3	24
PC104P_BIS3_RX FIFO WORD COUNT 0,1,2,3	25
Interrupts	26
Loop-back	27
APPLICATIONS GUIDE	29
Interfacing	29
CONSTRUCTION AND RELIABILITY	30
THERMAL CONSIDERATIONS	30
WARRANTY AND REPAIR	31

<b>SERVICE POLICY</b>	<b>31</b>
<b>OUT OF WARRANTY REPAIRS</b>	<b>31</b>
<b>FOR SERVICE CONTACT:</b>	<b>31</b>
<b>SPECIFICATIONS</b>	<b>32</b>
<b>ORDER INFORMATION</b>	<b>33</b>
<b>SCHEMATICS</b>	<b>33</b>



---

---

# List of Figures

---

---

FIGURE 1	PC104P-BISERIAL-III BLOCK DIAGRAM	6
FIGURE 2	PC104P-BISERIAL-III-TG1 ADDRESS MAP	12
FIGURE 3	PC104P-BISERIAL-III-TG1 BASE CONTROL REGISTER BIT MAP	14
FIGURE 4	PC104P-BISERIAL-III-TG1 STATUS BIT MAP	14
FIGURE 5	PC104P-BISERIAL-III-TG1 INTERRUPT STATUS BIT MAP	15
FIGURE 6	PC104P-BISERIAL-III-TG1 DIRECTION TERMINATION CONTROL BIT MAP	16
FIGURE 7	PC104P-BISERIAL-III 485 DATA IO BIT MAP	17
FIGURE 8	PC104P-BISERIAL-III-TG1 485 DATA RDBK BIT MAP	17
FIGURE 9	PC104P-BISERIAL-III-TG1 PARALLEL CONTROL BIT MAP	18
FIGURE 10	PC104P-BISERIAL-III-TG1 DATA SOURCE BIT MAP	19
FIGURE 11	PC104P-BISERIAL-III-TG1 DATA SOURCE BIT MAP	20
FIGURE 12	PC104P-BISERIAL-III-TG1 BURST IN BIT MAP	21
FIGURE 13	PC104P-BISERIAL-III-TG1 BURST OUT BIT MAP	22
FIGURE 14	PC104P-BISERIAL-III-TG1 FIFO R/W BIT MAP	23
FIGURE 15	PC104P-BISERIAL-III-TG1 TX AMT COUNT BIT MAP	23
FIGURE 16	PC104P-BISERIAL-III-TG1 RX AF COUNT BIT MAP	24
FIGURE 17	PC104P-BISERIAL-III-TG1 TX FIFO COUNT BIT MAP	24
FIGURE 18	PC104P-BISERIAL-III-TG1 RX FIFO COUNT BIT MAP	25
FIGURE 19	PC104P-BISERIAL-III CONNECTOR PINOUT	28

# Product Description

The PC104p-BiSerial-III-TG1 is part of the PC/104p Module family of modular I/O components by Dynamic Engineering. The PC104p-BiSerial-III is capable of providing multiple serial protocols. The TG1 protocol implemented provides four channels of serial interface. The serial format is used with sensors. LSB first, 32 bit words within a 1024 bit frame using an alternating 512 bit strobe reference. Transitions on the falling edge; data stable on the rising edge of the clock. The hardware has 2Kx32 internal FIFOs for each of the 4 receive and 1 transmit ports. The memory is supported with DMA to allow for streaming data from the system into host memory.

The TG1 version is minimized and does not include the ADC, DAC, PLL, TTL or external FIFOs.

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.

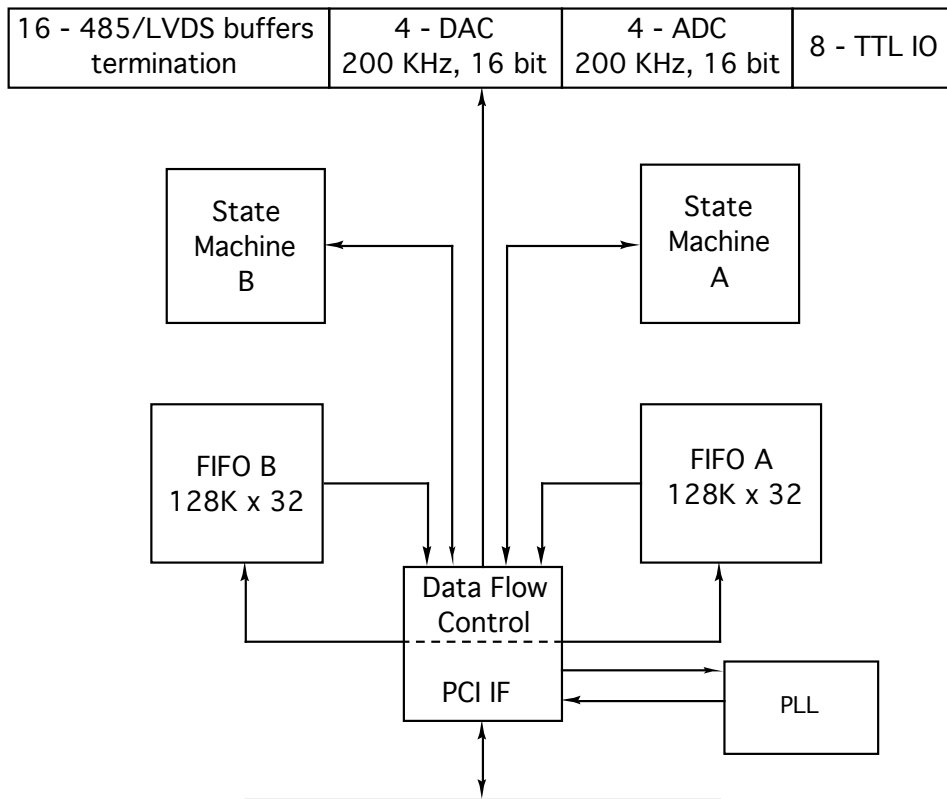


FIGURE 1

PC104P-BISERIAL-III BLOCK DIAGRAM



The configuration shown in Figure one 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 as much memory, and are more efficiently implemented using the internal FIFOs. Internal FIFOs can be configured using the block RAM within the Xilinx.

Sixteen differential I/O are provided for the serial/parallel signals. The drivers and receivers conform to the LVDS or RS-485 specification (exceeds RS-422 specification). The LVDS or 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 pullup/pulldown resistor packs can also be installed to provide a logic '1' on undriven lines.

The terminations and transceivers are programmable through the Xilinx device to provide the proper mix of outputs, inputs and terminations needed for a specific protocol implementation. The TG1 Serial interface uses 15 of the 16 differential IO. 0-11 are used for the 4 input ports [3 signals each] and 14-12 are used for the transmit port. All 16 bits can be programmed to be Parallel data or the alternate function – sensor data via the source control register. The terminations are programmable for all IO.

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

The PC104p-BiSerial-III conforms to the PC/104p standard. This guarantees compatibility with multiple PC/104p boards. Because the PC/104p may be mounted on different form factors, while maintaining plug and software compatibility, system prototyping may be done on one Carrier board, with final system implementation on a different one. For example the PCI2PC104p – PCI carrier for PC/104p can be used for development in a conventional PC. Later the hardware and software can be ported to the target.  
<http://www.dyneng.com/pci2pc104p.html>

The serial format for transmit and receive is specific to the sensors used with the TG1 design. The electrical format is RS422. The "LR" signal acts as a frame and toggles between low and high states. The data is valid both when low and when high. The data and clock are synchronized to provide 512 bits per "side" of LR and 1024 bits per "cycle" of the LR signal. The data is sent LSB first and is stable on the rising edge of the reference clock. The data is captured as 32 bit words – 16 words per side of LR [Left Right]

The receive data rate is tested at 2.5 Mhz. The system expected rate is 1.3 Mhz. The receiver design uses the PCI clock to sample the clock, data and LR signal, shift the data into a shift register based on the sampled clock and store into the FIFO assigned for that channel. The receive rate can be increased.

The transmitter uses a reference oscillator to operate. The oscillator value can be changed or the divisor used with the oscillator changed to provide alternate



frequencies. The onboard PLL is not used for the TG1. The PLL could be implemented to provide more programmable transmit frequencies and or a new reference rate for the receiver if higher frequencies are desired.

Interrupts are supported by the PC104p-BiSerial-III. An interrupt can be configured to occur at the end of a transmitted message. An interrupt can be set at the end of a reception. All interrupts are individually maskable and a master interrupt enable is also provided to disable all interrupts simultaneously.

The TG1 design is implemented with the idea of offloading the CPU as much as possible. Four channels of [receive] DMA are implemented with large buffer capabilities so that long link lists can be used to reduce the amount of CPU interaction required to capture large data streams.

# Theory of Operation

The PC104p-BiSerial-III-TG1 is designed for transferring data from one point to another with a simple serial protocol.

The PC104p-BiSerial-III-TG1 features a Xilinx Spartan III FPGA. The FPGA contains all of the registers and protocol controlling elements of the PC104p-BiSerial-III design. Only the transceivers, and switches are external to the Xilinx device in this application.

The PCI interface to the host CPU is controlled by a logic block within the Xilinx. The PC104p-BiSerial-III design requires one wait state for read or write cycles to any address. The PC104p-BiSerial-III is capable of supporting 40 MBytes per second into and out of the FIFO's with single word reads and writes. 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.

In reality in most systems the transfer rate will be 2-4 LW/uS for approximately 12Mbytes per second. With DMA the data is transferred at 33 Mhz with 32 bit words for a transfer rate of 132 Mbytes/sec transfer rate when active. With most PC systems there is enough system overhead to reduce the effective transfer rate to approximately 50% of the maximum or 66 Mbytes/sec.

With 4 channels running DMA and 66 Mbytes/sec available there are 16.5 Mbytes/sec [4.125 MLW/Sec] available per channel on the TG1. With 2.5 Mhz data received and converted to 32 bit words the effective receive rate is 78.1 K LW/sec. Plenty of margin for the data transfer to system memory.

The DMA channels are all independent allowing the data streams to be independently written to separate buffers in system memory. Most systems have only one DMA controller per card causing the data to have to be mixed within the file or for a lot of CPU overhead to reprogram the DMA for each transfer to occur as the data is ready. With the TG1 each channel is separate allowing larger transfers to happen with the data flow controlling when the data is transferred and the CPU available to do other processing tasks.

The FIFOs are in place for transmit channels in parallel with the receivers. Only the transmitter on channel 0 is implemented due to IO limitations and system requirements on the TG1. If your system can use the TG1 function and would operate using the transmitter then we can port this design to the PMC BiSerial III and instantiate the other transmitters. The PMC version has 32+ transceivers available.

[http://www.dyneng.com/pmc\\_biserial\\_III.html](http://www.dyneng.com/pmc_biserial_III.html)



The PC104p-BiSerial-III can support many protocols. The PC104p-BiSerial-III-TG1 uses a sensor protocol.

State machines within the FPGA control all transfers between the internal registers and FPGA logic, and the FPGA and the data buffers. The TX state machine reads from the transmit data register and loads the shift register before sending the data. The RX state machine receives data from the data buffers and takes care of moving data from the shift register into the RX storage.

Data is read from the TX channel FIFO and loaded into the shift register. The LSB is then present at the output of the data buffer. One bit period later the data is transitioned to the next value. The LSB+1 is now on the data lines. This process repeats until the first word is transferred. If more data is available, then the process repeats for the second word. Etc. The transmitter memory should be programmed with sufficient data to continue prior to enabling that channel to insure that the FIFO does not empty while additional data is loaded. Please refer to the register bit definitions for more details.

The data rate is set by an external oscillator for TX and by the system for RX. The transceivers are rated for 40 MHz. The interface is synchronous. The external system may or may not be operating when the TG1 is enabled to capture data. The external signals are sampled and operated on by a state-machine operating at the PCI clock rate [33 Mhz]. Independent of the externals the state-machine will behave properly – in the event of dropped reference clock, LR or data.

The receive function uses a free running shift register coupled with the receive state-machine to capture the data. The LR signal is synchronized and then sampled. When the channel receive state-machine is enabled it searches for the first transition from low to high or high to low. After the transition the shift register captures data for the next 32 [rising] clocks [first clock of the new LR state as the transition is on the falling edge]. After each group of 32 bits the data is captured into a storage register and the FIFO interface signaled that there is data available to store into the FIFO.

The 32 bit words are counted. After 16 are captured the hardware looks for the next transition and then starts to capture again. The transition should happen on the falling edge following the last bit captured causing a new 16 to be started on the next clock – no gaps in the data. If the transmitter runs dry and extends the state of the LR signal no loss in data or extra data capture will occur as the state-machine will wait for the transition then proceed to capture data. Each state where a delay is possible has an escape based on the enable to go back to the idle state – to abort the capture process.



# Address Map

PC104P_BIS3_BASE	0x0000	// 0	base control register offset
PC104P_BIS3_ID_SW	0x0004	// 1	ID & Switch Register offset
PC104P_BIS3_STATUS	0x0008	// 2	base Status register
PC104P_BIS3_DIR_TERM	0x000C	// 3	direction and termination register
PC104P_BIS3_PARDAT_485	0x0010	// 4	parallel 485 data IO DATA
PC104P_BIS3_PARDAT_485_RDBK	0x0014	// 5	parallel 485 data reg read-back
PC104P_BIS3_PARCNTL	0x0018	// 6	parallel data control register
PC104P_BIS3_TTL_IO	0x001C	// 7	TTL Data out and TTL Data in
PC104P_BIS3_TTL_RDBK	0x0020	// 8	TTL Data out register read-back
PC104P_BIS3_FIFO_A	0x0024	// 9	External FIFO A write,read port
PC104P_BIS3_FIFO_B	0x0028	// 10	External FIFO B write,read port
PC104P_BIS3_DAC0	0x002C	// 11	DAC 0 data port
PC104P_BIS3_DAC1	0x0030	// 12	DAC 1 data port
PC104P_BIS3_DAC2	0x0034	// 13	DAC 2 data port
PC104P_BIS3_DAC3	0x0038	// 14	DAC 3 data port
PC104P_BIS3_ADC0	0x003C	// 15	ADC 0 CMD/data port
PC104P_BIS3_ADC1	0x0040	// 16	ADC 1 CMD/data port
PC104P_BIS3_ADC2	0x0044	// 17	ADC 2 CMD/data port
PC104P_BIS3_ADC3	0x0048	// 18	ADC 3 CMD/data port
PC104P_BIS3_ASTAT	0x004C	// 19	DAC/ADC Status port
PC104P_BIS3_base0	0x0078	// 30	base control for channel 0
PC104P_BIS3_int0	0x007C	// 31	INT latch and status 0
PC104P_BIS3_burstin0	0x0080	// 32	burst in 0
PC104P_BIS3_burstout0	0x0084	// 33	burst out 0
PC104P_BIS3_fiforw0	0x0088	// 34	single read / write 0
PC104P_BIS3_txamtc0	0x008C	// 35	TX Almost Empty Count 0
PC104P_BIS3_rxafc0	0x0090	// 36	RX Almost Full Count 0
PC104P_BIS3_txfifowc0	0x0094	// 37	TX FIFO Word Count 0
PC104P_BIS3_rxfifowc0	0x0098	// 38	RX FIFO Word Count 0
PC104P_BIS3_spare0	0x009C	// 39	Spare 0
PC104P_BIS3_base1	0x00A0	// 40	base control for channel 1
PC104P_BIS3_int1	0x00A4	// 41	INT latch and status 1
PC104P_BIS3_burstin1	0x00A8	// 42	burst in 1
PC104P_BIS3_burstout1	0x00AC	// 43	burst out 1
PC104P_BIS3_fiforw1	0x00B0	// 44	single read / write 1
PC104P_BIS3_txamtc1	0x00B4	// 45	TX Almost Empty Count 1
PC104P_BIS3_rxafc1	0x00B8	// 46	RX Almost Full Count 1
PC104P_BIS3_txfifowc1	0x00BC	// 47	TX FIFO Word Count 1
PC104P_BIS3_rxfifowc1	0x00C0	// 48	RX FIFO Word Count 1
PC104P_BIS3_spare1	0x00C4	// 49	Spare 1
PC104P_BIS3_base2	0x00C8	// 50	base control for channel 2
PC104P_BIS3_int2	0x00CC	// 51	INT latch and status 2



```

PC104P_BIS3_burstin2      0x00D0    // 52  burst in 2
PC104P_BIS3_burstout2    0x00D4    // 53  burst out 2
PC104P_BIS3_fiforw2      0x00D8    // 54  single read / write 2
PC104P_BIS3_txamtc2      0x00DC    // 55  TX Almost Empty Count 2
PC104P_BIS3_rxafc2       0x00E0    // 56  RX Almost Full Count 2
PC104P_BIS3_txfifowc2    0x00E4    // 57  TX FIFO Word Count 2
PC104P_BIS3_rxfifowc2    0x00E8    // 58  RX FIFO Word Count 2
PC104P_BIS3_spare2       0x00EC    // 59  Spare 2

PC104P_BIS3_base3        0x00F0    // 60  base control for channel 3
PC104P_BIS3_int3         0x00F4    // 61  INT latch and status 3
PC104P_BIS3_burstin3     0x00F8    // 62  burst in 3
PC104P_BIS3_burstout3    0x00FC    // 63  burst out 3
PC104P_BIS3_fiforw3      0x00100   // 64  single read / write 3
PC104P_BIS3_txamtc3      0x00104   // 65  TX Almost Empty Count 3
PC104P_BIS3_rxafc3       0x00108   // 66  RX Almost Full Count 3
PC104P_BIS3_txfifowc3    0x0010C   // 67  TX FIFO Word Count 3
PC104P_BIS3_rxfifowc3    0x00110   // 68  RX FIFO Word Count 3
PC104P_BIS3_spare3       0x00114   // 69  Spare 3

//FIFO_TX0_SIZE 1        // n x 32 internal
FIFO_TG1_TX0_SIZE 2048   // 2048 x 32 internal FIFO
FIFO_TG1_TX1_SIZE 2048   // 2048 x 32 internal FIFO
FIFO_TG1_TX2_SIZE 2048   // 2048 x 32 internal FIFO
FIFO_TG1_TX3_SIZE 2048   // 2048 x 32 internal FIFO

//FIFO_RX0_SIZE 1        // n x 32 internal FIFO
FIFO_TG1_RX0_SIZE 2048   // 2048 x 32 internal FIFO
FIFO_TG1_RX1_SIZE 2048   // 2048 x 32 internal FIFO
FIFO_TG1_RX2_SIZE 2048   // 2048 x 32 internal FIFO
FIFO_TG1_RX3_SIZE 2048   // 2048 x 32 internal FIFO

```

FIGURE 2

PC104P-BISERIAL-III-TG1 ADDRESS MAP

The address map provided is for the local decoding performed within the PC104p-BiSerial-III-TG1. The addresses are all offsets from a base address, which is assigned by the system when the PCI bus is configured.

The VendorId = 0x10EE. The CardId = 0x0025.



# Programming

Programming the PC104p-BiSerial-III-TG1 requires only the ability to read and write data from the host. The base address is determined during system configuration of the PCI bus. The base address refers to the first user address for the slot in which the PMC is installed.

Depending on the software environment it may be necessary to set-up the system software with the PC104p-BiSerial-III-TG1 "registration" data. For example in WindowsNT there is a system registry, which is used to identify the resident hardware. Other OS may be more "plug and play".

In order to receive data the software is only required to enable the receiver for the channel(s) of interest. The data will be captured once the receiver is synchronized with the "LR" signal and the data stored into the associated FIFO. The data can be read based on the FIFO status or DMA programmed and enabled to provide automatic transfer of data to the system memory.

To transmit, data should be written into the channel O TX FIFO and then the transmitter enabled. The initial write before enable needs to be enough data to guarantee that the FIFO will not go empty before more data can be written.

Interrupts are used to help manage the DMA process. When a programmed transfer is completed the interrupt can be generated to alert the host to program a new transfer to a new location. The transfers can be programmed to be quite large and are independent for each channel allowing the CPU interaction to be minimized.

The Dynamic Engineering TG1 driver for Windows 2000 and XP manages the interaction and can set-up the DMA for you. The driver is easily integrated into a Visual C programming environment. Please refer to the driver manual for more information.



## Register Definitions

### PC104P\_BIS3\_BASE

PC104P BIS3 BASE 0x0000 // 0 base control register offset

BASE Control Register	
DATA BIT	DESCRIPTION
31-3	Spare and reserved bits
2	M_IO_EN
1	intforce
0	m_inten

FIGURE 3 PC104P-BISERIAL-III-TG1 BASE CONTROL REGISTER BIT MAP

M\_inten when set allows the PC104p-Biserial-III-TG1 to generate interrupts at the card level. There are additional local enables for each interrupt type at the channel level.

Intforce when set causes an interrupt to be generated to the system. Useful for debugging and software test. Please note that M\_inten must be enabled [set] for intforce to have an effect.

M\_IO\_EN when set ['1'] serves as a master IO enable for the receive channels. The separate channel enables also need to be enabled for the Master IO Enable to have effect. To synchronize channels, set the M\_IO\_EN to '0' then enable the channels of interest, then set to '1' so that all channels see the enable at the same time. For asynchronous operation set this bit to '1'.

### PC104P\_BIS3\_ID\_SW

PC104P BIS3 ID SW 0x0004 // 1 ID & Switch Register offset

Switch and Revision Port	
DATA BIT	DESCRIPTION
31-16	Spare
15-8	revision
7-0	switch in

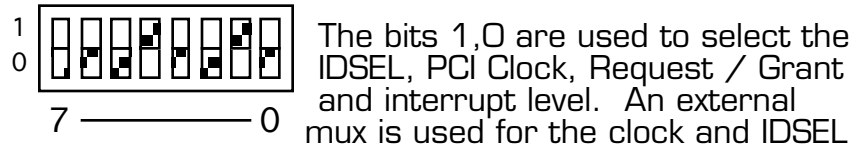
FIGURE 4 PC104P-BISERIAL-III-TG1 STATUS BIT MAP

The dip switch can be read with the lower eight bits of this port. Direct mapping.



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.



lines. The Request, Grant, and Interrupt logic is internal to the Xilinx. Bits 1,0 should be set to correspond to the level of the Biserial within the PC104p stack. The CPU would be the base level. Position "00" would be immediately above the CPU, position "01" would be the next slot up etc.

The revision is for the FLASH is currently 0x3. It is a good idea for your software to check the FLASH revision to make sure that they are both working from the same definitions.

### PC104P\_BIS3\_STATUS

PC104P BIS3 STATUS 0x0008 // 2 base Status register

DATA BIT		DESCRIPTION
31		int_stat
30-1		Spare
0		loc_int

FIGURE 5 PC104P-BISERIAL-III-TG1 INTERRUPT STATUS BIT MAP

LOC\_INT = Force\_int for this implementation. INT\_STAT is the masked version of FORCE\_INT. For designs implementing other non-channel resources more interrupt status will be available. Please also see the channel interrupts for more information.

## PC104P\_BIS3\_DIR\_TERM

PC104P BIS3 DIR TERM 0x000C // 3 direction and termination register

CONTROL		DIR_TERM REGISTER	
DATA	BIT	DESCRIPTION	
15-0		DIRection	15-0 0 = read 1 = drive
31-16		TERMination	15-0 1 = terminated

FIGURE 6 PC104P-BISERIAL-III-TG1 DIRECTION TERMINATION CONTROL BIT MAP

The direction for each of the 16 differential pairs is controlled through this port. The port defaults to zero, which corresponds to tri-stating the drivers and no terminations enabled.

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 tri-stated line will yield approximately 2.5V on each side of the tri-stated driver.

CONTROL	CORRESPONDING IO BITS
DIR_15-0	IO_15-0

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. If the Parallel Port is set to be an input with the direction bits then the corresponding termination bits should also be set.

CONTROL	CORRESPONDING IO BIT(S)
TERM_15.0	IO_15..0

## PC104P\_BIS3\_PARDAT\_485

PC104P BIS3 PARDAT 485 0x0010 // 4 parallel 485 data IO DATA

Parallel Data (485) IO Port	
DATA BIT	DESCRIPTION
31-16	Spare parallel output data
15-0	

FIGURE 7

PC104P-BISERIAL-III 485 DATA IO BIT MAP

There are 16 potential output bits in the parallel port. The Direction and Termination register sets the direction of the bits. When the direction is set to output and the source control is set to parallel port then the bit definitions from this register are driven onto the corresponding parallel port lines.

Writing to this register puts data onto the enabled data lines [direction set].

Reading from this port returns all of the IO lines. It is possible that the output data does not match the IO data in the case of the Direction bits being set to input.

## PC104P\_BIS3\_PARDAT\_485\_RDBK

PC104P BIS3 PARDAT 485 RDBK 0x0014 // 5 parallel 485 data reg read-back

Parallel Data Register Read-back Port	
DATA BIT	DESCRIPTION
15-0	15-0

FIGURE 8

PC104P-BISERIAL-III-TG1 485 DATA RDBK BIT MAP

To read the contents of the PARDAT\_485 port access this port. Read only. This is the direct read of the register rather than the IO signals.

## PC104P\_BIS3\_PARCNTL

PC104P\_BIS3\_PARCNTL 0x0018 // 6 parallel data control register

Parallel Port Control	
DATA BIT	DESCRIPTION
31-16	Spare
15-0	Parallel Port Source Definitions

FIGURE 9

PC104P-BISERIAL-III-TG1 PARALLEL CONTROL BIT MAP

Each of the Parallel Port bits has a corresponding source control bit. When the bit is set '1' the parallel data is used [PC104P\_BIS3\_PARDAT\_485]. When '0' the defined IO is used. The TG1 design uses bits 11-0 for the 4 receive ports, and has bits 14-12 available as a transmit port. Bit 15 does not have an IO definition. The control bits for 14-0 should be set to 0 to use the IO definitions instead of the parallel port.

Please note that the direction & termination control bits need to be set to make the port bits act as inputs or outputs. For the TG1 the bits 11-0 should be set to x000 to leave the lower bits as inputs for the IO function. Most likely bits 27-16 should be set to xFFF to terminate the inputs.

Addresses x1C – x4C are not used in the TG1 design.

PC104P_BIS3_TTL_IO	0x001C	// 7	TTL Data out and TTL Data in
PC104P_BIS3_TTL_RDBK	0x0020	// 8	TTL Data out register read-back
PC104P_BIS3_FIFO_A	0x0024	// 9	External FIFO A write, read port
PC104P_BIS3_FIFO_B	0x0028	// 10	External FIFO B write, read port
PC104P_BIS3_DAC0	0x002C	// 11	DAC 0 data port
PC104P_BIS3_DAC1	0x0030	// 12	DAC 1 data port
PC104P_BIS3_DAC2	0x0034	// 13	DAC 2 data port
PC104P_BIS3_DAC3	0x0038	// 14	DAC 3 data port
PC104P_BIS3_ADC0	0x003C	// 15	ADC 0 CMD/data port
PC104P_BIS3_ADC1	0x0040	// 16	ADC 1 CMD/data port
PC104P_BIS3_ADC2	0x0044	// 17	ADC 2 CMD/data port
PC104P_BIS3_ADC3	0x0048	// 18	ADC 3 CMD/data port
PC104P_BIS3_ASTAT	0x004C	// 19	DAC/ADC Status port

## PC104P\_BIS3\_BASE 0,1,2,3

PC104P BIS3 base0,1,2,3 //30,40,50,60 base control for channel 0,1,2,3

Parallel Port Control	
DATA BIT	DESCRIPTION
31-8	Spare
7	RX ENABLE
6	TX ENABLE
5	FORCE INT
4	MINTEN
3	DMA RDEN
2	DMA WREN
1	BYPASS
0	RST

FIGURE 10

PC104P-BISERIAL-III-TG1 DATA SOURCE BIT MAP

RST when '1' causes the FIFO for the channel to reset. TX and RX.

Bypass when '1' causes the TX FIFO to automatically be read and loaded into the RX FIFO to implement Loop-back testing. Set to '0' for normal operation.

DMA WREN and DMA RDEN are interrupt enables for the DMA transfers. Set to enable.

MINTEN is the channel master interrupt enable. Enabled when '1'.

Force Int when set and MINTEN is also set causes an interrupt to be generated from this channel.

TX Enable when set '1' starts the Transmit IO state-machine. The State-machine will idle until the Valid signal from the TX FIFO is true. Data will then be sent at 2.5 Mbits/ sec until the FIFO data is not valid when needed or the enable is set to '0'. Please note that channel 0 is the only channel with the transmitter instantiated.

RX Enable when set '1' starts the Receive IO state-machine. The state-machine will wait for an edge on the data strobe [LR] and then start to capture data. Each time 32 bits is captured data is written to the RX FIFO. The process continues until disabled '0'.

There are no channel interrupts from the IO section. It is envisioned that the DMA interrupts will be utilized with this design.

## PC104P\_BIS3\_INT 0,1,2,3

PC104P BIS3 int0,1,2,3 // 31 INT latch and status 0,1,2,3

Channel Status Port	
DATA BIT	DESCRIPTION
15	DMA RD INT
14	DMA WR INT
13	DMA RD ERR
12	DMA WR ERR
11	LOC_INT
10-8	SPARE
7	RX FF VALID
6	RX FF FULL
5	RX FF AFL
4	RX FF MT
3	SPARE
2	TX FF FULL
1	TX FF AMT
0	TX FF MT

FIGURE 11

PC104P-BISERIAL-III-TG1 DATA SOURCE BIT MAP

Each of the four channels has a similar status register. Reading from the register provides status. Writing to the register with bits 15-12 set will clear the stored condition.

TX FF MT is the Transmit FIFO Empty status bit. When the internal TX FIFO is empty this bit will be set.

TX FF AMT is the Transmit FIFO Almost Empty status bit. When the FIFO data level is at or below the programmed "Almost Empty" point this bit will be set.

TX FF FULL is the Transmit FIFO Full status bit. When the FIFO is full this bit is set.

RX FF MT is the Receive FIFO Empty status bit. When the internal receive FIFO is empty this bit is set.

RX FF AFL is the Receive FIFO Almost Full status bit. When the FIFO is full to the programmed almost full level or above this bit is set.

RX FF FULL is the Receive FIFO Full status bit. When the FIFO is full this bit is set.

RX FF VALID is a status bit which is set to indicate that there is valid data ready to be read from the FIFO. The FIFO has an external pipeline to support the DMA and back-up requirements. The data from the FIFO is pre-read into the pipeline to

be ready to transfer. When valid data is in the last stage of the pipeline the valid bit is set. Please note that the FIFO may be empty and this bit still set as the pipeline is emptied.

LOC\_INT is set when a channel IO interrupt or Force Interrupt has occurred. In this design the IO does not have interrupts leaving only the Force Interrupt condition.

DMA write is associated with writing data into the card and the TX direction. DMA read is associated with reading data from the card and the RX direction. The DMA hardware within the Biserial 3 takes care of the actual data transfer.

DMA WR ERR is set when an error occurs during a DMA write transfer. This can be a target or master abort or an incorrect direction bit in a descriptor pointer.

DMA RD ERR is set when an error occurs during a DMA read transfer.

DMA RD INT is set when a DMA read transfer completes.

DMA WR INT is set when a DMA write transfer completes.

### PC104P\_BIS3\_BURSTIN 0,1,2,3

PC104P BIS3 burstin0,1,2,3 // 32 burst in 0,1,2,3

Burst IN Port	
DATA BIT	DESCRIPTION
31-0	Burst in starting address

FIGURE 12

PC104P-BISERIAL-III-TG1 BURST IN BIT MAP

Burst In is a VHDL module designed by Dynamic Engineering to manage DMA transfers of data from host memory to the local memory. In the case of the PC104p-BiSerial-III the local memory is FIFO based.

The software will access the Burst In address – located in host memory, and supply the hardware with the location of the initial DMA descriptor. The hardware will then fetch the pointers to the data from the location supplied. The data will be read from the host memory and the transfers controlled by the FIFO level. The burst in module will attempt to keep the FIFO close to full and the IO module will read the data and transmit.

The process is compatible with scatter gather DMA techniques. The Dynamic Driver for Windows 2000 and XP will manage this process for you.



If you are writing your own driver, each list entry consists of three 32-bit words. The linked list descriptor order is:

DmaPciAddress	PCI address of data block
DmaLength	Length of data block
NextPointer	PCI address of next descriptor (bit 0 is set to indicate that this is the last list entry - bit1 is set if the transfer is from the device to PCI memory)

### PC104P\_BIS3\_BURSTOUT 0,1,2,3

PC104P\_BIS3\_burstout0,1,2,3 // 33 burst out 0,1,2,3

Burst IN Port	
DATA BIT	DESCRIPTION
31-0	Burst in starting address

FIGURE 13

PC104P-BISERIAL-III-TG1 BURST OUT BIT MAP

Burst Out is a VHDL module designed by Dynamic Engineering to manage DMA transfers of data from local memory to the host memory. In the case of the PC104p-BiSerial-III the local memory is FIFO based.

The software will access the Burst Out address and supply the hardware with the location of the initial DMA descriptor – located in host memory. The hardware will then fetch the pointers to the data from the location supplied. The data will be read from the local memory and the transfers controlled by the FIFO level. The burst out module will attempt to keep the FIFO close to empty and the IO module will load data into the FIFO as data is received from the external system.

The process is compatible with scatter gather DMA techniques. The Dynamic Driver for Windows 2000 and XP will manage this process for you.

The link list order is the same as for the Burst In process.

### PC104P\_BIS3\_FIFORW 0,1,2,3

PC104P BIS3 fiforw0,1,2,3 // 34 single read / write 0,1,2,3

Single access Read / Write Port	
DATA BIT	DESCRIPTION
31-0	FIFO Data

FIGURE 14

PC104P-BISERIAL-III-TG1 FIFO R/W BIT MAP

To do non- DMA read and write accesses to the FIFO's use this port. The TX FIFO can be written to and the RX FIFO read from for each channel. If the Bypass mode is enabled the data will automatically be moved from TX to RX FIFO's allowing for DMA or single word access loop-back testing. For small data transfers this port can be easier to use than setting up the DMA. For larger transfers or time critical operation the DMA method is preferred.

### PC104P\_BIS3\_TXAMTC 0,1,2,3

PC104P BIS3 txamtc0,1,2,3 // 35,45,55,65 TX Almost Empty Count 0,1,2,3

TX Almost Empty Count Port	
DATA BIT	DESCRIPTION
15-0	Count Data

FIGURE 15

PC104P-BISERIAL-III-TG1 TX AMT COUNT BIT MAP

The TX Almost Empty Count is programmed with this register. The Register is read-write. The count is used to determine the channels Almost Empty state. When this level is true the DMA will move more data into the FIFO.

### PC104P\_BIS3\_RXAFC 0,1,2,3

PC104P BIS3 rxafc0,1,2,3 // 36,46,56,66 RX Almost Full Count 0,1,2,3

RX Almost Full Count Port	
DATA BIT	DESCRIPTION
15-0	Count Data

FIGURE 16

PC104P-BISERIAL-III-TG1 RX AF COUNT BIT MAP

The RX Almost Full Count is programmed with this register. The Register is read-write. The count is used to determine the channels Almost Full state. When this level is true the DMA will move more data out of the FIFO.

### PC104P\_BIS3\_TX FIFO WORD COUNT 0,1,2,3

PC104P BIS3 txfifowc0,1,2,3 // 37,47,57,67 TX FIFO Word Count 0,1,2,3

TX FIFO Word Count	
DATA BIT	DESCRIPTION
15-0	Count Data

FIGURE 17

PC104P-BISERIAL-III-TG1 TX FIFO COUNT BIT MAP

The TX FIFO level can be read with this port. The level is compared with the programmed level to create the TX Almost Empty flag. This port is read only.

## PC104P\_BIS3\_RX FIFO WORD COUNT 0,1,2,3

PC104P\_BIS3\_rxfifowc0,1,2,3 // 38,48,58,68 RX FIFO Word Count 0,1,2,3

TX FIFO Word Count	
DATA BIT	DESCRIPTION
15-0	Count Data

FIGURE 18

PC104P-BISERIAL-III-TG1 RX FIFO COUNT BIT MAP

The RX FIFO level can be read with this port. The level is compared with the programmed level to create the RX Almost Full flag. This port is read only.

## Interrupts

PC104p-BiSerial-III-TG1 interrupts are treated as auto-vectored. When the software enters into an exception handler to deal with a PC104p-BiSerial-III-TG1 interrupt the software must read the status register(s) 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 PC104p-BiSerial-III-TG1 DMA state machines generate an interrupt request when a programmed transfer is complete and the interrupt enable and Master interrupt enable bits are set.

The interrupt is mapped to INTA [or B or C or D based on the switch setting] on the PC/104p connector, which is mapped to a system interrupt when the PCI bus configures. The source of the interrupt is obtained by reading the Interrupt Status register. 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 Interrupt Status register.

## Loop-back

The Engineering kit has reference software, which includes an external loop-back test. The TG1 version of the PC104p-BiSerial-III utilizes a 50 pin right angle header connector. The test requires an external cable [ribbon] with the following pins connected. Upper nibble to Lower 3 nibbles.

<u>SIGNALs</u>	<u>S</u>	<u>D0</u>	<u>D1</u>	<u>D2</u>	<u>D3</u>
IO12+=>0,3,6,9	25	1	7	13	19
IO12-	26	2	8	14	20
IO13+=>1,4,7,10	27	3	9	15	21
IO13-	28	4	10	16	22
IO14+=>2,5,8,11	29	5	11	17	23
IO14-	30	6	12	18	24

## PC104p-BiSerial-III-TG1 Header Pin Assignment

The figure below gives the pin assignments for the header connector on the PC104p-BiSerial-III design. Please note that the Analog and TTL IO are not installed on this version. GND\* is a plane which is tied to GND through a 0805 0Ω resistor. DC, AC or open are options. For customized version, or other options, contact Dynamic Engineering.

IO_OP	CLK_in0+	IO_0m	CLK_in0-	1	2
IO_1P	Data_In0+	IO_1m	Data_In0-	3	4
IO_2P	LR_in0+	IO_2m	LR_in0-	5	6
IO_3P	CLK_in1+	IO_3m	CLK_in1-	7	8
IO_4P	Data_In1+	IO_4m	Data_In1-	9	10
IO_5P	LR_in1+	IO_5m	LR_in1-	11	12
IO_6p	CLK_in2+	IO_6m	CLK_in2-	13	14
IO_7p	Data_In2+	IO_7m	Data_In2-	15	16
IO_8p	LR_in2+	IO_8m	LR_in2-	17	18
IO_9p	CLK_in3+	IO_9m	CLK_in3-	19	20
IO_10p	Data_In3+	IO_10m	Data_In3-	21	22
IO_11p	LR_in3+	IO_11m	LR_in3-	23	24
IO_12p	CLK_out0+	IO_12m	CLK_out0-	25	26
IO_13p	Data_out0+	IO_13m	Data_out0-	27	28
IO_14p	LR_out0+	IO_14m	LR_out0-	29	30
IO_15p	spare+	IO_15m	spare-	31	32
GND*		GND*		33	34
ADCO		TTLO		35	36
ADC1		TTL1		37	38
ADC2		TTL2		39	40
ADC3		TTL3		41	42
DACO		TTL4		43	44
DAC1		TTL5		45	46
DAC2		TTL6		47	48
DAC3		TTL7		49	50

FIGURE 19

PC104P-BISERIAL-III CONNECTOR PINOUT

# 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 ribbon 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 PC104p-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 PC104p-BiSerial-III does not contain special input protection. The connector is pinned out for a standard Header cable to be used. The twisted pairs are defined to match up with the PC104p-BiSerial-III pin definitions. It is suggested that this standard cable be used for most of the cable run.

**Custom cables** can be manufactured with discrete wire header and direct connection to your mating equipment. The TG1 version has a custom cable with header and 4 25 pin D style connectors to go to the sensors. All gold plated contacts with strain relief and heat shrink cable covering.

**Terminal Block.** We offer a high quality 50-screw terminal block that directly connects to the ribbon cable. The terminal block can mount on standard DIN rails. HDRterm50

[<http://www.dyneng.com/HDRterm50.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

PC/104p Modules were conceived and engineered for rugged industrial environments. The PC/104p-BiSerial is constructed out of 0.062 inch thick FR4 material.

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 PC/104p device is secured into the stack with high insertion force pins and four screws attached to the 4 stand-offs. The four screws provide significant protection against shock, vibration, and incomplete insertion.

The PC/104p Module provides a low temperature coefficient of  $1.7 \text{ W}/^{\circ}\text{C}$  for uniform heat. This is based upon the temperature coefficient of the base FR4 material of  $0.31 \text{ W}/\text{m}^{\circ}\text{C}$ , and taking into account the thickness and area of the PC/104p. The coefficient means that if 1.7 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 PC104p-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  
435 Park Dr.  
Ben Lomond, CA 95005  
831-336-8891  
831-336-3840 fax

[support@dyneng.com](mailto:support@dyneng.com)



## Specifications

Host Interface:	PC/104p - 32 bit PCI bus
Serial Interface:	4 RX and 1 TX serial instrumentation protocol ports.
Tx Data rates generated:	2.5 Mhz
Rx Data rates accepted:	At least 2.5 MHz
Software Interface:	Control Registers, Status Ports
Initialization:	Hardware Reset forces all registers to 0.
Access Modes:	LW boundary Space (see memory map)
Wait States:	1 for all addresses
Interrupt:	DMA Read and write interrupts for each channel Software interrupt
DMA:	8 channels of DMA, one per IO channel. Please note that only 1 of the TX channels is completely implemented.
Onboard Options:	All Options are Software Programmable
Interface Options:	50 pin ribbon cable or discrete wire 50 screw terminal block interface
Dimensions:	Standard Single PC/104p Module.
Construction:	FR4 Multi-Layer Printed Circuit, Through Hole and Surface Mount Components. Programmable parts are socketed.
Temperature Coefficient:	1.7 W/°C for uniform heat across PC/104p
Power:	Max. <b>TBD</b> mA @ 5V



## Order Information

PC104p-BiSerial-III-TG1

PC/104p Module with 4 receive ports for sensors

Eng Kit

HDRterm50 - 50 position screw terminal adapter

<http://www.dyneng.com/HDRterm50.html>

HDRcabl50 - 50 position ribbon cable

<http://www.dyneng.com/HDRribn50.html>

Technical Documentation,

1. PC104p-BiSerial-III Schematic

2. PC104p-BiSerial-III-TG1 Reference test

software

Data sheet reprints are available from the manufacturer's web site

Windows driver option.

**Note:** The Engineering Kit is strongly recommended for first time PC/104p-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.

All information provided is Copyright Dynamic Engineering

