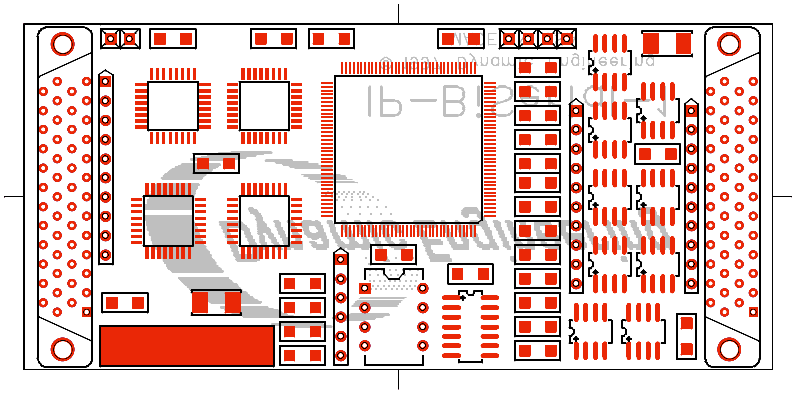
DYNAMIC ENGINEERING

435 Park Dr., Ben Lomond, Calif. 95005
831-336-8891 **Fax** 831-336-3840
Web Page http://www.dyneng.com
E-Mail sales@dyneng.com
engineering@dyneng.com
support@dyneng.com
Est. 1988

User Manual

IP-BiSerial-BA2

Bi-directional Serial Data Interface
IP Module



Revision A2
Corresponding Hardware: Revision A

IP-BiSerial-BA2
Bi-directional Serial Data Interface
IP Module

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

Manual Revision A2. Revised Jan 25th, 1999

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 IP Module carriers and compatible user-provided equipment. Connection of incompatible hardware is likely to cause serious damage.

Dynamic Engineering

Hardware and Software Design • Manufacturing Services

Table of Contents

List of Figures

Product Description

IP-BISERIAL-BA2 is part of the IP Module family of modular I/O components. The IP-BISERIAL-BA2 is capable of providing multiple serial protocols. The standard protocol implemented provides a Data, Clock and Strobe interface with Ready control. the -BA2 version is a custom modification. The main modifications include switching to lsb first format, adding parity generation and checking.

In addition to the BA2 version other custom interfaces are available. Please see our web page for current protocols offered. If you do not find it there 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 contact Dynamic Engineering with your custom application.

The IP-BISERIAL-BA2 supports both 8 and 32 Mhz. IP Bus operation. The IP Clock or an external reference is used to derive the reference clocks for the serial operation. Please be sure to select the proper clock divisors and source selector after reset to insure proper operation. Please refer to the programming section for details.

Both single ended and differential I/O are available on the serial signals. The differential drivers and receivers conform to the RS-485 specification (exceeds RS-422 specification). The RS-485 input signals are terminated with 180Ω. The single ended driver signal is characterized as an open drain driver with 24 mA of sink. For convenience A 2KΩ pull-up is supplied on board, for faster termination a second pull-up can be added at the receiving end of the circuit. Single ended signals are received through 33Ω resistors. Care should be taken with the single ended signals. Transients can damage the board.

All configuration registers support read and write operations for maximum software convenience. Word and byte operations are supported (please refer to the memory map).

The IP-BISERIAL-BA2 conforms to the VITA standard. This guarantees compatibility with multiple IP Carrier boards. Because the IP may be mounted on different form factors, while maintaining plug and software compatibility, system prototyping may be done on one IP Carrier board, with final system implementation on a different one.

The serial channels are supported by a 1K by 16 bit FIFO. The FIFO supports byte and word reads. A byte wide write path exists for loopback testing. The serial receive channel looks for data in 16 bit transfers plus parity. The received words are then loaded into the FIFOs. The data length loaded is determined by the strobe signal. The host can poll or wait for the message complete interrupt. The message can be read directly from the input FIFO.

The Output channel has a separate 1k x 16 FIFO. The FIFO can be written as words or bytes. The upper and lower bytes are read together and sent as a 16 bit data word. Parity is automatically generated and appended to each data word. The FIFO can be accessed directly for loop back testing. The data is available in a byte wide format when looped back.

The IP-BISERIAL-BA2 can create an external reference clock. Several rate divisors are selectable. Please refer to the clock selection section within the programming section for details.

Interrupts are supported by the IP-BISERIAL-BA2. The interrupt occurs at the end of the transmission whether data is received or sent or both. The interrupts are individually maskable. The vector is user programmable by a read/write register. The interrupt occurs on IntReq0. The FIFO status is available for the FIFO making it possible to operate in a polled mode.

Theory of Operation

The IP-BISERIAL-BA2 is designed for the purpose of transferring data from one point to another with a serial protocol.

The IP-BISERIAL-BA2 features a Xilinx FPGA. The FPGA contains all of the registers and protocol controlling elements of the BISERIAL design. Only the drivers, receivers, boot PROM and FIFOs are external to the Xilinx device.

The IP-BISERIAL-BA2 is a part of the IP Module family of modular I/O products. It meets the IP Module Vita Standard. Contact Dynamic Engineering for a copy of this specification. It is assumed that the reader is at least casually familiar with this document and logic design. In standard configuration it is a Type 1 mechanical with no components on the back of the board and one slot wide.

The bus interface to the host CPU is controlled by a logic block within the Xilinx device that contains the decoding and timing elements required to interface to the IP bus interface. The timing is referenced to the 8 or 32 MHz IP logic clock. The IP responds to the ID, INTSEL, and IO selects. The DMA and MEM control lines are connected to the Xilinx for future revisions, and are not used at this time. The BISERIAL design requires wait states for read or write cycles to any address. Hold cycles are supported as required by the host processor. Data remains enabled during a read until the host removes the SEL line. Local timing terminates a write cycle prior to the SEL being deasserted. If no hold cycles are requested by the host, the IP-BISERIAL-BA2 is capable of supporting 16+ MB per second data transfer rate with a 32 Mhz. reference rate.

The serial I/O can support many protocols. The -BA2 timing is shown in the next diagram. The clock is free running, the data is valid on the falling edge of the clock, and strobe frames the data. The timing is in reference to an external user supplied clock. The clock is input on the TX_RDY differential receiver line pair. The TX state-machine synchronizes the output transmission to this clock. There are small delays associated with receiving the external clock through the differential receiver and bringing the signal into the FPGA. The internal version of the clock is also retansmitted on the reference clock output.

A pair of state machines within the FPGA control all transfers between the FIFO and FPGA, and the FPGA and the data buffers. The TX state machine reads from the transmit FIFOs and loads the shift registers 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 FIFO.

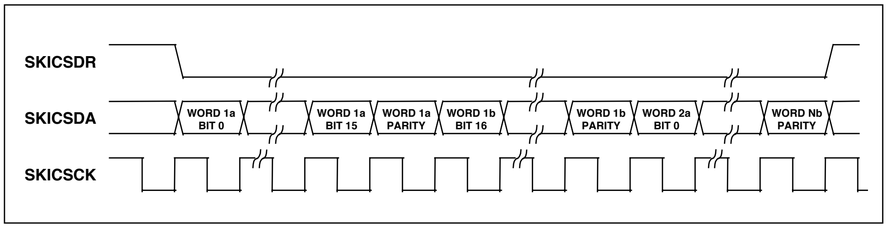


FIGURE 1 IP-BISERIAL-BA2 SERIAL PROTOCOL TIMING

When the start bit is detected high to begin the transfer, the data is read from the TX FIFO and loaded into the shift register. The LSB is then present at the output of the data buffer. The Strobe is activated at the same time. One half clock period later the Falling edge of the data clock is driven to the output clock buffer. One half clock period later the data is transitioned to the next value. The LSB+1 is now on the data lines. The process repeats until the first word is transferred. At the end of the word a 17th bit is added for parity. The parity bit is programmable to be odd or even.   Assuming that there is data to be sent in the lower FIFO a second word is read and loaded into the shift register and sent out onto the bus. The process is repeated until that word is transferred.   The transfer will continue until either FIFO is empty when it is time to load. The data stream is continuous.

The receive function is very similar. When the Strobe is detected low and the clock active then data is loaded into the receive shift register on the falling edge of the data clock. Once a word has been received the data is loaded into the receive FIFO. When the strobe goes inactive the transfer has been completed and an interrupt is generated to the host [if enabled].

The receiver checks for parity, over-run and framing errors. If an error is detected the appropriate bit in STAT1 is set, an interrupt is generated, and the reception aborts.

A counter keeps track of the number of words received. The counter counts once per 17 bit word received. The counter loads the count into a latch and then clears the counter to allow a second message to be received before the count is read. The count self clears when read through STAT1. The counter is 10 bits wide.

Address Map

Function		Offset	Width	Type
BIS_CNTL0	EQU	$00	byte on word boundary	read/write
BIS_CNTL1	EQU	$02	byte on word boundary	read/write
BIS_CNTL2	EQU	$04	byte on word boundary	read/write
BIS_VECTOR	EQU	$06	byte on word boundary	read/write
BIS_STAT0	EQU	$08	byte on word boundary	read
BIS_STAT1	EQU	$0A	byte on word boundary	read
BIS_STAT2	EQU	$0C	word on word boundary	read
BIS_RESET	EQU	$0E	word	write
BIS_TX_FIFO_0_W	EQU	$10	D15..8 byte or word	write
BIS_TX_FIFO_1_W	EQU	$11	D7..0 byte	write
BIS_TX_FIFO_0_R	EQU	$10	byte on word boundary	read
BIS_TX_FIFO_1_R	EQU	$14	byte on word boundary	read
BIS_RX_FIFO_0_W	EQU	$20	byte on word boundary	write
BIS_RX_FIFO_1_W	EQU	$24	byte on word boundary	write
BIS_RX_FIFO_0_R	EQU	$20	D15..8 byte or word	read
BIS_RX_FIFO_1_R	EQU	$21	D7..0 byte	read
BISERIAL_IDPROM	EQU	$80	byte on word boundary	read

FIGURE 2 IP-BISERIAL-BA2 INTERNAL ADDRESS MAP

The address map provided is for the local decoding performed within the IP-BISERIAL-BA2. The addresses are all offsets from a base address. The carrier board that the IP is installed into provides the base address and controls the “naming of the bytes”. We refer to the bytes following Motorola conventions..i.e. upper is D15-D8 and lower is D7-D0. When byte wide data is located on the lower byte then an odd address results or the use of a word access using only the lower byte of data. We prefer the word oriented approach because it is more consistent accross platforms.

Programming

Programming the IP-BISERIAL-BA2 requires only the ability to read and write data in the host's I/O space. The base address is determined by the IP Carrier board. This documentation refers to the address where the IO space for the slot that the IP is installed in as the base address.

In order to receive data the software is only required to enable the RX state machine, FIFOs, and set the parity properly. If desired, the interrupt can be enabled and the interrupt vector written to the vector register. Data will be loaded into the FIFOs as it is received.

A typical sequence would be to first write to the vector register with the desired interrupt vector. For example $40 is a valid user vector for the Motorola 680x0 family. Please note that some carrier boards do not use the interrupt vector. The interrupt service routine should be loaded and the mask should be set. When the start bit is set the hardware looks to make sure that the strobe is not active then begins looking for it to be active. In this manner the data received is protected from joining mid-message. Once a new strobe assertion is detected the data loading process begins. When the strobe is detected to be deasserted the interrupt request is asserted to let the host know that the data is available. The software can read the word count and set-up a loop counter to efficiently read the data from the FIFOs. If the next message starts and completes then the FIFO empty flag can be used to control the amount of data to read.

The end of transmission interrupt will indicate to the software that the message has been started and that the message has terminated. If both the TX and RX interrupts are enabled then the SW needs to read BIS_STAT1 to see which source caused the interrupt. Reading BIS_STAT1 will clear the interrupt status, and the INTACK cycle will clear the actual interrupt. The interrupt status can be read after the INTACK cycle. It is a good idea to read the status register to force the RX_INT and TX_INT bits to 0 before Start is enabled to insure that the RX_INT or TX_INT=1 value read by the interrupt service routine came from the current reception.

Before transmitting data the FIFOs are enabled and the data loaded. The BA-2 design has both internal and external clock selection capability. If the internal reference is used then the baud rate selections are used to select the output rate. If the external clock souce is selected then the rate is set by the external source. If the internal reference is selected then CLK_HI must be set to the proper level. CLK_HI selects [or not] the prescaler. Alternate baud rates are available if CLK_HI is set mis-matched to the IP reference rate. The parity bit can be programmed to be odd or even. The default is odd. The strobe width can be programmed to be standard or short to cause errors for test purposes. Once the complete message is loaded and the controls set properly the start bit can be set to cause the transfer to begin. If a slow clock rate is selected and a long message is to be sent then data can be loaded during transmission to save operational time. Care must be taken to make sure that the FIFOs do not become empty. When the TX interrupt is received the transmission has been completed and another message can be loaded. All that needs to happen with a second message is to load the FIFO and set the start bit.

Messages longer than 2K bytes can be accomodated by special ordering HW with larger FIFOs or by using the MT and Full flags on the FIFOs to poll during the transfer…fill the tx FIFO and when not full add more data until full. On the receive side poll and when not empty read the data until empty. the PAE and PAF flags are not implemented in the standard version of the hardware, but could be used to provide an almost empty interrupt to allow the TX side to operate in an interrupt driven mode with longer messages. Similarly the PAF could be used to provide an almost full interrupt to the receive side host to allow interrupt driven long message capability.

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

Register Definitions

BIS_CNTL0

$00 BISERIAL Control Register Port read/write

CONTROL REGISTER 0

DATA BIT	DESCRIPTION
7	CLK_HI_B
6	EXT_INT
5	R2
4	R1
3	R0
2	INT_SET
1	INT_EN_TX
0	STRT_TX

FIGURE 3 IP-BISERIAL-BA2 CONTROL REGISTER 0 BIT MAP

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

2. CLK_HI_B is used to let the HW know which IP reference clock is present to derive the TX Clock rate from. If running at 32 Mhz. CLK_HI_B should be set. If not set when the faster IP clock is used then the baud rates will all be off by a factor of 4 [lower than actual]. Depending on cable length the faster rates may work. If set when the IP clock rate is 8 Mhz then the Baud rates will also be a factor of 4 off [faster than actual].

3. EXT_INT is used to select the clock source for the transmitter to be the external clock or one derived from the IP clock. 0 = internal, 1 = external.

4. R2, R1 and R0 are the bit rate selection bits for generating the external reference clock. and the TX transmit clock.

Bit Pattern	**Divisor**	**clk hi & 32 or !clk hi & 8**
000	1	default = 4 MHz.
001	2	2 MHz.
010	4	1 Mhz.
011	8	500 KHz.
100	16	250 KHz.
101	32	125 KHz.

110	64	62.5 KHz.
111	128	31.25 Khz.

5. INT_SET us used to create an interrupt for test and software development purposes. Set the bit to cause the interrupt and clear the bit to remove the interrupt.

6. INT_EN_TX is the Interrupt Enable bit for the Transmit channel. The default state is off. If enabled and the master interrupt enable is also enabled then an interrupt is requested when the transmission is complete. The interrupt is cleared by reading the status register.

7. STRT_TX is set to send data. The bit is auto cleared at the end of a transmission.

BIS_CNTL1

[$02 BISERIAL Control Register Port read/write

CONTROL REGISTER 1

DATA BIT	DESCRIPTION
7	CLR_FIFO
6	SPARE
5	CLK_HI_WS
4	PARITY_TYPE_TX
3	PARITY_TYPE_RX
2	STROBE_WIDTH
1	INT_EN_RX
0	STRT_RX

FIGURE 4 IP-BISERIAL-BA2 CONTROL REGISTER 1 BIT MAP

1. CLR_FIFO is used to reset the FIFOs. The default state is reset. The FIFOs must be taken out of reset to be used to store data. Please refer to FTX_LD and FRX_LD [control reg 2].

2. CLK_HI_WS is used to let the hardware know which rate IP clock is present. If the card is plugged into a carrier board directly then the selection can remain '0' for no inserted waitstates. If the card is being tested on an extender cable then the waitstate should be selected with a value of '1'. The hardware only inserts waitstates if the bit is set, and it is a read cycle.

3. INT_EN_RX is used to enable the receive interrupt. The default is disabled. If enabled and the master interrupt enable is also enabled then an interrupt is requested when the Strobe returns to the off state [1]. The interrupt is cleared by reading the status register.

4. STRT_RX is used to enable the receive state machine to receive messages. Unlike the transmit state machine the start bit is not auto-cleared at the end of a transmission.

5. PARITY_TYPE_RX,TX are used to select the parity "style" generated by the TX state machine or tested for by the RX state machine. 0 = odd, 1 = even.

6. STROBE_WIDTH when set to 1 causes the TX state machine to generate a shorter than standard strobe width and should cause a framing error on the receiver side.

BIS_CNTL2

[$04 BISERIAL Control Register Port read/write

CONTROL REGISTER 2

DATA BIT	DESCRIPTION
7	TESTMODE
6	INT_EN
5	FRX_LD
4	FTX_LD
3	RX_RDY
2	Spare
1	TXTTL_422
0	RXTTL_422

FIGURE 5 IP-BISERIAL-BA2 CONTROL REGISTER 2 BIT MAP

1. TESTMODE is used to select the reference clock to the TX and RX FIFOs. In test mode a faster clock is provided to the FIFOs to allow the interface to keep up with the IP Bus requirements. Default is 0. Normal operation is 0.

2. INT_EN is the master interrupt enable. Default is 0. If set to 1 then either the RX or TX interrupts can occur based on the state machines and the state of the RX and TX interrupt enables. If the master interrupt enable is off [0] then no interrupts will be generated. The status register can see the interrupt requests from the RX and TX state machines to allow polled operation.

3. FRX_LD is tied to the RX FIFO WE2/_LD pin. FTX_LD is tied to the TX FIFO WE2/_LD pin.When the FIFOs are taken out of reset it is possible to set-up the FIFO to accept commands to program the way the programmable almost empty and programmable almost full signals operate. ***In the standard transfer mode these pins are set hi before***

CLR_FIFO is released to use as a second WE control pin. If the PAE and PAF flags are used for a different protocol then the FIFOs will require programming.

4. TXTTL_422 is the control used to select whether a TTL input of RS422 [485] input should be used for the RDY signal in the TX state machine. The default state is 422. A '1' selects TTL. Set to 0 for BA-2 use.

5. RX_TTL_422 is the control bit to select whether the data, clock, and strobe should be received as TTL or 422 signals. Default state = 0 = 422 [485] a '1' selects TTL inputs. Set to 0 for BA-2 use.

6. RX_RDY bit is used to set the state of the Receiver Ready Bit. If the system needs Receiver Input Ready control then program the transmitter to output when Ready is '1' and use the RX_RDY to grant or block transmission. Default is '0'. Can be used as a general purpose output bit.

BIS_Vector

[$06] BISERIAL Interrupt Vector Port
The Interrupt vector for the BISERIAL is stored in this byte wide register. This read/write register is initialized to 'xxFF' upon power-on reset or software reset. The vector is stored in the odd byte location [D7..0]. The vector should be initialized before the interrupt is enabled or the mask is lowered. The interrupt is automatically cleared when the CPU acknowledges the interrupt.

BIS_STAT0

[$08] BISERIAL Status Port [read only]

Data Bit	Status	
8	CLK_SEL_RX	1 = external clock source for Receiver, 0 = internal
7	RX_STB	1 = reception in progress
6	TX_STB	1 = transmission in progress
5	FTX_MT_0	0 = empty 1 = not empty
4	FTX_MT_1	0 = empty 1 = not empty
3	FTX_FF_0	0 = full, 1 = not full
2	FRX_MT_0	0 = empty, 1 = not empty
1	FRX_FF_0	0 = full 1 = not full
0	FRX_FF_1	0 = full 1 = not full

FIGURE 6 IP-BISERIAL-BA2 STATUS REG 0 BIT MAP

1. RX_STB & TX_STB are indicators that a data transfer is in progress.

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

3. CLK_SEL_RX is an indication of which reference clock the receiver is utilizing. If set then the external clock has been detected and used. If cleared then the external clock is not present and the local 8mhz clock is used instead. If the internal reference is used, transactions will process normally, but the data will be incorrect. This mode was added to keep the state-machine from becoming stuck when the external clock is not active.

BIS_STAT1

[$0A] BISERIAL Status Port [read only]

Data Bit	Status
0	TX_INT 1 = Interrupt pending
1	RX_INT 1 = Interrupt pending
2	Parity Error 1 = error detected
3	Frame Error 1 = error detected
4	Over Run Error 1 = error detected
5-7	unused read ‘0’

FIGURE 7 IP-BISERIAL-BA2 STATUS REG 1 BIT MAP

1. RX_INT & TX_INT are set when the respective interrupt conditions exist and the interrupts are enabled. The master interrupt can be disabled and still have the benefit of the status. The status is cleared when read.

2. Parity, Frame, and Over Run errors are tested for when a reception is in progress. If detected then the status bit is set and the reception is aborted. The word count shows the number of words loaded when aborted. Cleared on read of STAT1.

BIS_STAT2

[$0C] BISERIAL Status Port [read only]

Data Bit	Status	
0	CNT0	lsb of word counter
1	CNT1	
2	CNT2	
3	CNT3	
4	CNT4	
5	CNT5	
6	CNT6	
7	CNT7	
8	CNT8	
9	CNT9	MSB of word counter
10	MC_ERR	1 = new count written before old count read 0 = no error

FIGURE 8 IP-BISERIAL-BA2 STATUS REG 2 BIT MAP

1. Word count. The word count is updated for each word loaded into the FIFOs. Read the word count to determine the number of words to read. Cleared on read of STAT2. The counter is cleared when the new reception starts. The latch storing the count is loaded when the reception is completed. The counter is updated once per word received. The software has until the next message has been completed before the current count is over-written. The length of time is set by system constraints.

BISERIAL_RESET

[$0E] BISERIAL Reset Port

The user can, by accessing this port, cause the BISERIAL to reset all major functions. The Control register, and FIFO's are cleared by a write to this port. Any data pattern can be written.

BIS_TX_FIFO_0_W

[$10] BISERIAL FIFO byte 0 write

The BISERIAL supports byte writes to the data FIFOs. By writing a byte to this address only byte_0 is affected. D15..8 are loaded at this address. Word writes will load both TX FIFOs.

BIS_TX_FIFO_1_W

[$11] BISERIAL FIFO byte 1 write

The BISERIAL supports byte writes to the TX FIFOs. By writing a byte to this address only byte_1 is affected. If a word is written to BIS_TX_FIFO_0_W this byte is loaded as well.

BIS_TX_FIFO_0_R

[$10] BISERIAL FIFO byte 0 write

A loopback path is provided for the TX FIFOs to allow the host to read the data stored in the TX FIFOs. Both bytes are read back through the lower byte lane [D7..0]. Reading from this address fetches from the upper FIFO byte. Be sure to set the clock to Testmode before reading from this register. *Once the data is read from the FIFO the data is no longer available for transmission*.

BIS_TX_FIFO_1_R

[$14] BISERIAL FIFO byte 0 write

A loopback path is provided for the TX FIFOs to allow the host to read the data stored in the TX FIFOs. Both bytes are read back through the lower byte lane [D7..0]. Reading from this address fetches from the lower FIFO byte. Be sure to set the clock to Testmode before reading from this register. *Once the data is read from the FIFO the data is no longer available for transmission*.

BIS_RX_FIFO_0_W

[$20] BISERIAL FIFO byte 0 write

A loopback path is provided for the RX FIFOs to allow the host to load data into the RX FIFOs. Both bytes are written through the lower byte lane [D7..0]. Writing to this address loads the upper RX FIFO. This operation competes with and should not be performed during normal operation.

BIS_RX_FIFO_1_W

[$24] BISERIAL FIFO byte 1 write

A loopback path is provided for the RX FIFOs to allow the host to load data into the RX FIFOs. Both bytes are written through the lower byte lane [D7..0]. Writing to this address loads the lower RX FIFO. This operation competes with and should not be performed during normal operation.

BIS_RX_FIFO_0_R

[$20] BISERIAL FIFO byte 0 read
The data stored into FIFO_0 can be accessed through this port. Byte and word accesses are available. A word access will fetch data from both FIFO 0 and FIFO 1.

BIS_RX_FIFO_1_R

[$21] BISERIAL FIFO byte 1 read
The data stored into FIFO 1 can be accessed through this port. Only byte wide accesses are supported.

Interrupts

All IP Module interrupts are vectored. The vector from the IP-BISERIAL-BA2 comes from a vector register loaded as part of the initialization process. The vector register can be programmed to any 8 bit value. The default value is $FF which is sometimes not a valid user vector. The software is responsible for choosing a valid user vector.

The IP-BISERIAL-BA2 state machines generate an interrupt request when a transmission or reception is complete and the INTEN bits in the control registers are set. The transmission is considered complete when the strobe line is deactivated. The interrupt is mapped to interrupt request 0. The CPU will respond by asserting INT. The hardware will automatically supply the appropriate interrupt vector and clear the request when accessed by the CPU. The source of the interrupt is obtained by reading BIS_STAT1. The status remains valid until the status register is read. The interrupt status is auto-cleared when the status register is accessed. Some carrier boards prefetch data.
If your carrier board prefetches the interrupt status then the status may be cleared when the SW goes to look at it. If this is an issue then reading the BIS_STAT1 before BIS_STAT0 is usually a solution.

The interrupt level seen by the CPU is determined by the IP Carrier board being used. The master interrupt can be disabled or enabled through the BIS_CNTL2 register. The individual enables for TX and RX are controlable through BIS_CNTL0 and BIS_CNTL1. The enable operates before the interrupt holding latch which stores the request for the CPU. Once the interrupt request is set, the way to clear the request is to reset the board, service the request, or disable the interrupt. Toggling the interrupt enable low will clear the interrupt, the interrupt enable can be set back to enabled immediately. TX_INT_EN enables and clears the TX interrupt and RX_INT_EN enables and clears the RX interrupt request.

If operating in a polled mode and making use of the interrupts for status then the master interrupt should be disabled and the Rx or TX or both enabled. When BIS_STAT1 shows and interrupt pending the appropriate FIFO action can take place and the enable toggled to remove the interrupt request then one extra read of the BIS_STAT1 to make sure that the interrupt request is cleared before starting the next transfer. Reading the BIS_STAT1 register does clear the interrupt status, but if the source of the status is still pending [interrupt request] then the status can become set again before the SW has a chance to clear it out. Hence the necessity of one extra read for clearing purposes.

Power on initialization will provide a cleared interrupt request, interrupts disabled, and interrupt vector of $FF.

ID PROM

Every IP contains an ID PROM, whose size is at least 32 x 8 bits. The ID PROM aids in software auto configuration and configuration management. The user's software, or a supplied driver, may verify that the device it expects is actually installed at the location it expects, and is nominally functional. The ID PROM contains the manufacturing revision level of the IP. If a driver requires that a particular revision be present, it may check for it directly.

The location of the ID PROM in the host's address space is dependent on which carrier is used. Normally the ID PROM space is directly above the IPs I/O space, or at IP-base + $80. Macintosh drivers use the ID PROM automatically.

Standard data in the ID PROM on the IP-BISERIAL-BA2 is shown in the figure below. For more information on IP ID PROMs refer to the IP Module Logic Interface Specification, available from Dynamic Engineering.

Each of the modifications to the IP-BiSerial-IO board will be recorded with a new code in the DRIVER ID location. -BA2 is set to ‘2’.

Address	Data
01	ASCII "I" ($49)
03	ASCII "P" ($50)
05	ASCII "A" ($41)
07	ASCII "H" ($48)
09	Manufacturer ID ($1E)
0B	Model Number ($01)
0D	Revision ($A2)
0F	reserved (00)
11	Driver ID, low byte (02)
13	Driver ID, high byte (00)
15	No of extra bytes used (0C)
17	CRC (a3)

FIGURE 9 IP-BISERIAL-BA2 ID PROM

IP Module Logic Interface Pin Assignment

The figure below gives the pin assignments for the IP Module Logic Interface on the IP-BISERIAL-BA2. Pins marked n/c below are defined by the specification, but not used on the IP-BISERIAL-BA2. Also see the User Manual for your carrier board for more information.

GND		GND		1		26	
	CLK		+5V		2		27
Reset*		R/W*		3		28	
	D0		IDSEL*		4		29
D1		n/c		5		30	
	D2		MEMSEL*		6		31
D3		n/c		7		32	
	D4		IntSel*		8		33
D5		n/c		9		34	
	D6		IOSel*		10		35
D7		n/c		11		36	
	D8		A1		12		37
D9		n/c		13		38	
	D10		A2		14		39
D11		n/c		15		40	
	D12		A3		16		41
D13		IntReq0*		17		42	
	D14		A4		18		43
D15		n/c		19		44	
	BS0*		A5		20		45
BS1*		n/c		21		46	
	n/c		n/c		22		47
n/c		Ack*		23		48	
	+5V		n/c		24		49
GND		GND		25		50	

NOTE 1: The no-connect signals above are defined by the IP Module Logic Interface Specification, but not used by this IP. See the Specification for more information.

NOTE 2: The layout of the pin numbers in this table corresponds to the physical placement of pins on the IP connector. Thus this table may be used to easily locate the physical pin corresponding to a desired signal. Pin 1 is marked with a square pad on the IP Module.

FIGURE 10 IP-BISERIAL-BA2 LOGIC INTERFACE

IP Module IO Interface Pin Assignment

The figure below gives the pin assignments for the IP Module IO Interface on the IP-BISERIAL-BA2. Also see the User Manual for your carrier board for more information.

GND		ICSSKDR +		1		26	
	REFCLK_422+		ICSSKDR -		2		27
REFCLK_422-		GND		3		28	
	GND		RX_RDY_TTL		4		29
SYS_CLK_IN+		GND		5		30	
	SYS_CLK_IN-		RX_DATA_TTL		6		31
GND		GND		7		32	
	SKICSDA +		RX_CLK_TTL		8		33
SKICSDA -		GND		9		34	
	GND		RX_STB_TTL		10		35
SKICSCK +		GND		11		36	
	SKICSCK -		TX_RDY_TTL		12		37
GND		GND		13		38	
	SKICSDR +		GND		14		39
SKICSDR -		R_C_TTL_IN		15		40	
	GND		GND		16		41
RX_RDY_422+		GND		17		42	
	RX_RDY_422-		TX_DTA_TTL		18		43
GND		GND		19		44	
	ICSSKDA +		GND		20		45
ICSSKDA -		TX_CLK_TTL		21		46	
	GND		GND		22		47
ICSSKCK +		GND		23		48	
	ICSSKCK -		TX_STB_TTL		24		49
GND		GND		25		50	

NOTE 1: The layout of the pin numbers in this table corresponds to the physical placement of pins on the IP connector. Thus this table may be used to easily locate the physical pin corresponding to a desired signal. Pin 1 is marked with a square pad on the IP Module.

FIGURE 11 IP-BISERIAL-BA2 IO INTERFACE

BA2 uses the 422 versions of data, clock and strobe. The external clock reference is received on SYS_CLK_IN for this revision. TX refers to the transmit and RX refers to Receive relative to the BiSerial board.

Applications Guide

Interfacing

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 IP-BISERIAL-BA2 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. IP-BISERIAL-BA2 does not contain special input protection.

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, by applying voltage less than ground or more than +5 volts with the IP powered. With the IP unpowered, driven input voltages should be kept within .7 volts of ground potential.

Terminal Block. We offer a high quality 50 screw terminal block that directly connects to the flat cable. The terminal block mounts on standard DIN rails.

Many flat cable interface products are available from third party vendors to assist you in your system integration and debugging. These include connectors, cables, test points, 'Y's, 50 pin in-line switches, breakout boxes, etc.

Construction and Reliability

IP Modules were conceived and engineered for rugged industrial environments. The IP-BISERIAL-BA2 is constructed out of 0.062 inch thick FR4 material.

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

The IP Module connectors are keyed and shrouded with Gold plated pins on both plugs and receptacles. They are rated at 1 Amp per pin, 200 insertion cycles minimum. These connectors make consistent, correct insertion easy and reliable.

The IP is secured against the carrier with four metric M2 stainless steel screws. The heads of the screws are countersunk into the IP. The four screws provide significant protection against shock, vibration, and incomplete insertion. For most applications they are not required.

The IP Module provides a low temperature coefficient of 0.89 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 IP. The coefficient means that if 0.89 Watts are applied uniformly on the component side, then the temperature difference between the component side and solder side is one degree Celsius.

Thermal Considerations

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

Warranty and Repair

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

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

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

Service Policy

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

Out of Warranty Repairs

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

For Service Contact:

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

Specifications

Logic Interface:	IP Module Logic Interface
Serial Interface:	RS-485 RDY, Data, CLK, STB
TX CLK rates generated:	4 MHz, 2 Mhz, 1 MHz, 500 KHz, 250 KHz, 125 KHz, 62.5 KHz, 31.25 KHz
Software Interface:	Control Register, ID PROM, Vector Register, Status Port, FIFO
Initialization:	Hardware Reset forces all registers to 0. Software Reset Command resets the control register, and FIFO's.
Access Modes:	Word or Byte in I/O Space (see memory map) Word in ID Space Vectored interrupt
Access Time:	back-to-back cycles in 500ns (8Mhz.) or 125 nS (32 Mhz.) to/from FIFO
Wait States:	1 to ID space, 2 or 3 to IO or INT space depending on CLK_HI_WS selection
Interrupt:	Tx interrupt at end of transmission Rx interrupt at end of transmission
DMA:	No Logic Interface DMA Support implemented at this time
Onboard Options:	All Options are Software Programmable
Interface Options:	50 pin flat cable 50 screw terminal block interface User cable
Dimensions:	Standard Single IP Module. 1.8 x 3.9 x 0.344 (max.) inches
Construction:	FR4 Multi-Layer Printed Circuit, Through Hole and Surface Mount Components. Programmable parts are socketed.
Temperature Coefficient:	0.89 W/^{O}C for uniform heat across IP
Power:	Max. **220** mA @ 5V

Order Information

IP-BISERIAL-BA2	IP Module with 1 Tx and 1 Rx serial channel, Programmable data rates Standard protocol support, RS-485 drivers and receivers 16 bit IP interface
Tools for IP-BISERIAL-BA2	IP-Debug-Bus - IP Bus interface extender IP–Debug-IO - IO connector breakout