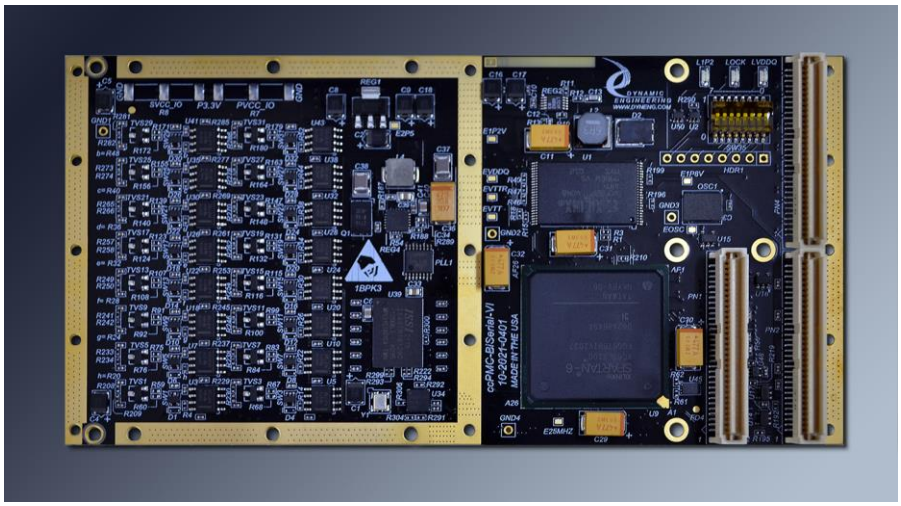
DYNAMIC ENGINEERING
150 DuBois St., Suite B&C Santa Cruz, CA 95060
(831) 457-889
https://www.dyneng.com
sales@dyneng.com
Est. 1988

User Manual

(cc)PMC-BiSerial-VI-GPIO Hardware Manual

General Purpose IO with COS Change of State detection

Manual Revision 01p2



ccPMC Model shown Rev 01

PMC: 10-2015-0604/5
ccPMC: 10-2021-0401/2

PMC-Biserial-VI-GPIO
ccPMC-Biserial-VI-GPIO
32 differential GPIO

Dynamic Engineering
150 DuBois St., Suite C
Santa Cruz, CA 95060
(831) 457-8891

Manual Revised 11/25/24

Table of Contents

List of Figures

Product Description

In embedded systems many of the interconnections are made with differential – RS-485 or LVDS signals. Depending on the system architecture an IP, PMC, or XMC will be the right choice to make the connection. With most architectures you have a choice as there are carriers for PCIe, PCI, cPCI, VPX, VME, PC/104p and other buses for XMC, PMC, and IP mezzanine modules.

Usually the choice is based on other system constraints as XMC, PMC, and IP can provide the IO you require. Dynamic Engineering would be happy to assist in your decision regarding architecture and other trade-offs with the XMC/ PMC / IP decision. Dynamic Engineering has carriers for XMC, IP, and PMC modules for most architectures, and is adding more as new solutions are requested and required by our customers.

The conduction cooled PMC - ccPMC-BiSerial-VI-GPIO and PMC compatible PMC-BiSerial-VI-GPIO have 32 independent digital IO. Hereafter referred to as BiSerial-VI-GPIO. The high density makes efficient use of slot resources. The IO is available for system connection through the front panel, via the rear [Pn4] connector, or both. A high density SCSI front panel connector provides the front panel IO [PMC only]. The IO lines are protected with 400 W TVS devices. The rear panel IO has a PIM and PIM Carrier available for rear panel wiring options in some systems.

In addition to the IO, a temperature sensor, internal FIFO with DMA, PLL with 4 clock references [1 used this application], Oscillator 32 MHz, Oscillator 100 MHz differential, plus 50 and 25 MHz references. Software selects between the PLLA input and 32 MHz after division to provide the Change of State reference clock. 256 Mbytes of DDR memory is also available for future revisions and customized versions.

Industrial temperature components are standard.

BiSerial-VI-GPIO provides 32 IO. Each IO is independent with programming options for direction, polarity, level or edge triggered with separate rising and falling enables. In addition, a choice of reference clocks with the local 50 MHz oscillator or programmable PLL. Filtered and direct IO access are also provided. Interrupts are programmable on a per line basis with 3 enables per line to cover the level, rising, and falling options.

Reference software for Win10 and Linux provide references for all of the modes of operation including setting up clocking, interrupts, using the parallel ports etc.

The HDEterm68 https://www.dyneng.com/HDEterm68.html

can be used as a breakout for the front or rear panel IO. The HDEcabl68 provides a convenient cable. https://www.dyneng.com/HDEcabl68.html Custom cables can be manufactured to your requirements. Please contact Dynamic Engineering with your specifications.

Each channel is programmable to be input or output on a channel-by-channel basis. All 32 IO channels can be used as interrupt generators. Interrupts are programmable to be based on rising, falling and change of state [both] conditions. The interrupts are maskable to allow polled operation as well.

The inputs are available unfiltered and after the transition detection. The transition detection is programmable for clock rate. The local 32 MHz oscillator, PCI or external clocks can be selected as the reference to the clock divider. The clock divider is programmable to use the reference rate or to divide it to a lower frequency.

All of the IO are routed through the FPGA to allow for custom applications that require hardware intervention or specific timing- for example an automatic address or data strobe to be generated.

The IO are implemented with RS-485 or LVDS transceivers. When enabled, the data stored in the output register is driven onto the line.

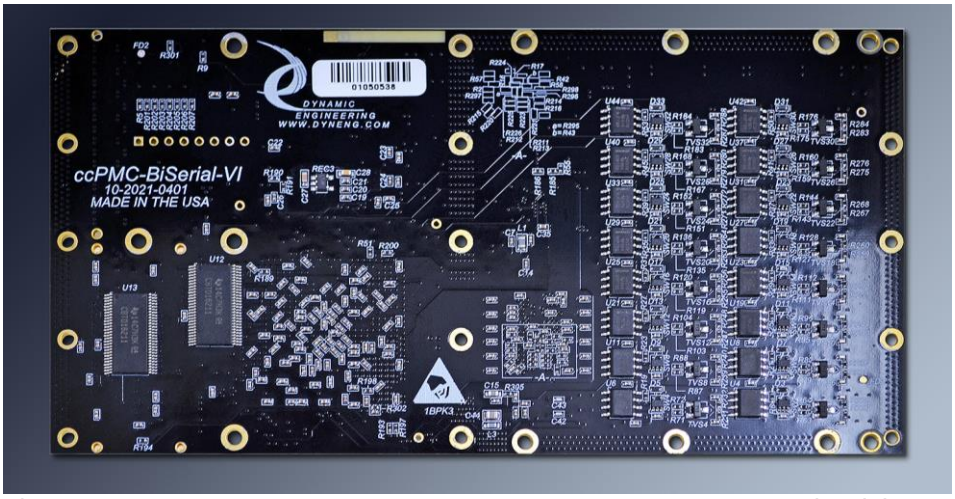


Figure 1 ccPMC-BiSerial-VI-GPIO REAR VIEW

The registers are mapped as 32 bit words. All registers are read-writeable. The Linux and Windows® compatible drivers are available to provide the system level interface for

this design. Use standard C/C++ to control your hardware or use the Hardware manual to make your own software interface. The software manuals are also available on-line. The Linux documentation is provided in-line with the source code.

The basic functions of parallel IO and COS capture are designed into the “GPIO” model.

PMC-BiSerial-VI and ccPMC-BiSerial-VI are part of the Mezzanine Module family of modular I/O components. ccPMC-BiSerial-VI conforms to the conduction cooled standard. PMC-BiSerial-VI conforms to the PMC standard. This guarantees compatibility with multiple Carrier boards. Because the mezzanine 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, use PCIeBPMCX1 to develop your SW in a standard PCIe slot and then port to your target HW.

Dynamic Engineering offers drivers and reference software for Windows®, Linux, and VxWorks. Drivers and reference SW are available AS-IS to clients of the (cc)PMC-BISERIAL-VI-GPIO. Support contracts are encouraged to help with integration and enhancements. https://www.dyneng.com/TechnicalSupportFromDE.pdf

Theory of Operation

BiSerial-VI-GPIO can be used for multiple purposes with applications in telecommunications, control, sensors, IO, test; anywhere multiple independent or coordinated IO are useful.

The designs feature the Xilinx Spartan 6 FPGA, RS-485 or LVDS Transceivers. The FPGA contains the PCI interface and control required for the GPIO interface.

The Xilinx design incorporates the “PCI Core” and additional modules for DMA in parallel with a direct register decoded programming model. The initial implementation provides an enhanced feature set based on the PMC-BiSerial-VI-GPIO design. Designs can be ported between the PMC and ccPMC implementations. Switching to TTL levels is also easy as the addressing and functions are very similar between the two implementations. Additional FLASH updates will provide new features.

The drivers are initialized to the receive state. The direction registers are used to program the channel to be a driver or not. Terminations are disabled at reset. Each IO line can be programmed to be terminated. Typically the receive side is terminated with 485 and LVDS.

Data written to the IO register can be placed on the bus. For an IO with the direction bit set: When a ‘0’ is written to any IO line register position the corresponding line is driven low. When a ‘1’ is written to any IO line register position that line is driven high.

If the direction bit is set to input, the level will be controlled by external devices. Open lines are recommended to be terminated to present the same voltage on both inputs [of the differential pair] and put the device in safe mode. The control register is read-writeable. The data register read returns the value of the data register. The direct read returns the value of the IO bus. The register read-back is independent of the bus; the data read will always match the data written. The IO data read will reflect the state of the bus and not necessarily the state of the on-board drivers.

The read-back registers are clocked at a programmable rate with an internal clock generator. The basic option is available under SW control. If special programming is needed please contact Dynamic Engineering for a custom FPGA implementation.

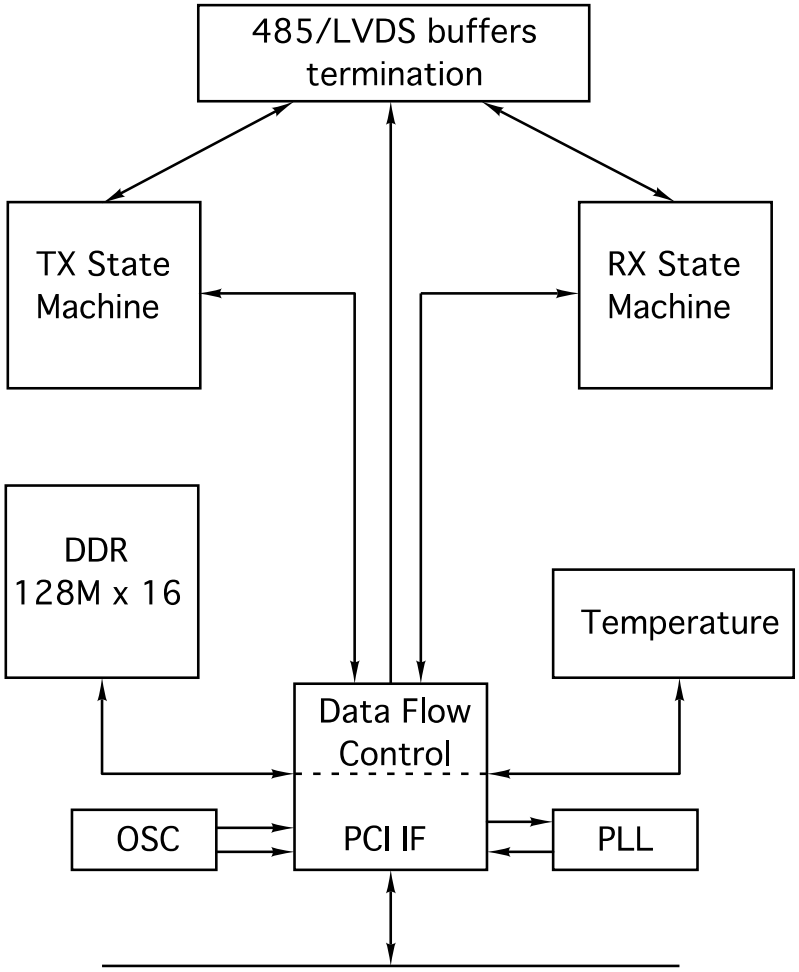


Figure 2 ccPMC-BiSerial-VI Block Diagram

Programming

Programming (cc)PMC-BISERIAL-VI-GPIO board 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 board is installed.

Once the initialization process has occurred, and the system has assigned addresses to BiSerial-VI-GPIO, software will need to determine what the address space is for the PCI interface [BAR0]. The offsets in the address table are relative to the system assigned BAR0 base address.

The next step is to initialize BiSerial-VI-GPIO. If the basic mode of direct read and write operations is to be used the default settings can be used except for setting the direction bits corresponding to the channels to transmit on.

If COS inputs are to be used the reference and divisor clocks may require programming. In many cases the default settings will work. In addition, the Rising, Falling, and Interrupt capabilities need to be programmed. Once the settings are in place it is recommended the receive state registers are written to for clearing purposes as the programming steps may cause phantom events to be captured.

One additional programming step will be to initialize the PLL to the user desired frequency if selected instead of the local oscillator.

For Windows™ and Linux systems the Dynamic Driver can be used. The driver will take care of finding the hardware and provide an easy-to-use mechanism to program the hardware. The Driver comes with reference software showing how to use the card and reference frequency files to allow the user to duplicate the test set-up used in manufacturing at Dynamic Engineering. Using simple, known to work routines is a good way to get acquainted with new hardware.

DMA is provided to support test of Dynamic Engineering carriers using DMA. The DMA function is not normally used with the current GPIO implementation. DMA could be added to the Tx, Rx or both to provide consistent rate of update for output or sample for input. Please contact Dynamic Engineering [engineering@dyneng.com] for any modifications we can make to support your project.

Firmware Updates

Revision 1.1: First release on ccPMC-BiSerial-VI.

Base Address Map

Function	Offset
// BiSerial-VI-GPIO definitions	
#define BiSerial-VI-GPIO_BASE	0x0000 // 0 Base control register offset
#define BiSerial-VI-GPIO_BASE1	0x0004 // 1 2nd Base control with Master Interrupt En
#define BiSerial-VI-GPIO_STATUS	0x0008 // 2 Status Register
#define BiSerial-VI-GPIO_SWITCH	0x000C // 3 Type, Revision, Dip Switch
#define BiSerial-VI-GPIO_TERM	0x0010 // 4 Set to enable termination
#define spare	0x0014 // 5 unused currently
#define BiSerial-VI-GPIO_DataOut0	0x0018 // 6 Data Out 31-0
#define PAR_TTL_GPIO_DataEn0	0x0020 // 8 Data Enable 31-0 '1' = Tx
#define PAR_TTL_GPIO_Polarity0	0x0028 // 10 Polarity Selection 31-0 '1' = inverted
#define PAR_TTL_GPIO_EdgeLevel0	0x0030 // 12 Edge or Level 31-0 '1' = Edge
#define PAR_TTL_GPIO_IntEnable0	0x0038 // 14 Interrupt Enables 31-0 '1' = Enabled
#define PAR_TTL_GPIO_Rising0	0x0040 // 16 Rising Edge Capture 31-0 '1' = Enabled
#define PAR_TTL_GPIO_Falling0	0x0048 // 18 Falling Edge Capture 31-0 '1' = Enabled
#define PAR_TTL_GPIO_CosRising0	0x0050 // 20 Status set when Rising Edge Captured 31-0
#define PAR_TTL_GPIO_CosFalling0	0x0058 // 22 Status set when Falling Edge Captured 31-0
#define PAR_TTL_GPIO_Direct0	0x0060 // 24 IO Data Synchronized and unfiltered 31-0
#define PAR_TTL_GPIO_Filtered0	0x0068 // 26 IO Data Synchronized, Polarized, Masked 31-0
#define PAR_TTL_GPIO_HalfDiv	0x0070 // 28 Divisor for COS clock reference
#define PAR_TTL_GPIO_TempData	0x0074 // 29 Current Temperature
#define PAR_TTL_GPIO_FIFO	0x0078 // 30 Single Word Access ⇔ FIFO 8Kx32
#define spare	0x007C // 31 Currently unused
#define PAR_TTL_WR_DMA_PNTR	0x0080 // 32 Write Physical Address to start DMA Tx
#define PAR_TTL_RD_DMA_PNTR	0x0084 // 33 Write Physical Address to start DMA Rx
#define spare	0x0088 // 34 reserved for Tx FIFO Count
#define spare	0x008C // 35 reserved for Rx FIFO Count

Figure 3 (cc)PMC-BISERIAL-VI-GPIO Address Map

The address map provided is for the local decoding performed within BiSerial-VI-GPIO. The addresses are all offsets from a base address. The upstream device provides the base address. Dynamic Engineering prefers a long-word oriented approach because it is more consistent across platforms.

The map is presented with the #define style to allow cutting and pasting into many compilers “include” files.

The host system will search the PCI bus to find the assets installed during power-on initialization.

ccPMC-BiSerial-VI-GPIO: VendorId = 0xDCBA and the CardId = 0x0072
PMC-BiSerial-VI-GPIO: VendorId = 0xDCBA and the CardId = 0x0073

The Type field has a unique number allowing SW to read the type currently connected to. The UserAp package makes use of this feature to print the correct board type on the menu. The OS uses the CardId to index into the INF and supply PMC or ccPMC in the card description in the device manager [Windows]. Linux and other systems also make use of the data.

Register Definitions

BiSerial-VI-GPIO_BASE

[$00 Base Control Register Port read/write]

DATA BIT	DESCRIPTION
31	TestClkOut
30-18	Spare
17	DMA_RdEn
16	DMA_WrEn
15	PLL_SDAT [write to PLL, read-back from PLL]
14	PLL_SCLK
13	PLL_EN
12	ClkCosSel
11-5	spare
4	Force Interrupt
3	spare
2	spare
1	IoRst
0	Spare

Figure 4 BiSerial-VI-GPIO Control port 0 Bit Map

The base control register for BiSerial-VI-GPIO is used for programming the PLL and other design level controls. Unused bits are reserved for additional new features. Unused bits should be programmed '0' to allow for future commonality.

IoRst When set '1' resets various HW functions. FIFOs, State-machines etc. In most cases the control registers are unaffected. Clear to '0' for normal operation.

Force Interrupt when '1' and the master enabled will cause an interrupt request. The interrupt can be cleared by clearing this bit or disabling the master interrupt enable or both. Force Interrupt is used for test and software development purposes.

PLL_EN: When this bit is set to a one, the signals used to program and read the PLL are enabled. PLL_SDAT is driven low when PLL_EN is '1' and PLL_SDAT is written '0' else Tristated. Place in tristate mode to read data back from the PLL.

PLL_SCLK/PLL_SDAT : These signals are used to program the PLL over the I^2C serial interface. SCLK is always an output whereas SDATA is bi-directional. Park the Data bit '1' to read back with SCLK.

The PLL is programmed with the output file generated by the Cypress PLL programming tool. [CY3672 R3.01 Programming Kit or CyberClocks R3.20.00 Cypress may update the revision from time to time.]

The .JED file is used by the Dynamic Driver to program the PLL. Programming the PLL is fairly involved and beyond the scope of this manual. For clients writing their own drivers it is suggested to get the Engineering Kit for this board including software, and to use the translation and programming files ported to your environment. This procedure will save you a lot of time. For those who want to do it themselves the Cypress PLL in use is the 22393. The output file from the Cypress tool can be passed directly to the Dynamic Driver [Linux or Windows] and used to program the PLL without user intervention.

The reference frequency for the PLL is 40 MHz.

JTAG signals are used to reprogram, verify, erase etc. the FLASH used to load the FPGA. JTAG Mux Sel when set causes the JTAG mux to connect the FPGA control to the FLASH device. When '0' the external programming header is selected. TDI, TMS, and CLK are used to control the actions of the state-machine within the FLASH and to transfer data. TDO is used to read the serial data returned from the FLASH. We are adding this feature to our drivers. Currently not supported.

TestClkOut is used to put a clock pattern on the output enabled bits. Used for test or to check basic IO operation. Normally disabled.

BiSerial-VI-GPIO_BASE1

[$04 Master Interrupt Control]

DATA BIT	DESCRIPTION
31-1	spare
0	Master Interrupt Enable

Figure 5 BiSerial-VI-GPIO Master Interrupt Enable

Master Interrupt Enable when '1' gates active interrupt requesting conditions onto Interrupt Request A. When set to '0' the interrupting functions are available as status but no interrupt request is generated by the card to allow for polled operation.

BiSerial-VI-GPIO_STATUS

[$08 Board level Status Port]

DATA BIT	DESCRIPTION
31	Interrupt Status
30-20	spare
19	Dma_RdInt
18	DMA_WrInt
17	DMA_RdErr
16	DMA_WrErr
15-5	spare
4	IntForce
3	LevelIntReq
2	CosFallingIntReq
1	CosRisingIntReq
0	local interrupt

Figure 6 BISERIAL-VI-GPIO Status Port Bit Map

Local Interrupt is set when any of the interrupt types is set – unmasked for non-DMA interrupt types.

CosRisingIntReq - This is the logical OR of the COS outputs for the Rising Edge condition. The RISING register will select which bits are enabled. If any of the enabled bits are active this bit is set. The status is captured before the master interrupt enable. If the master interrupt enable is set an interrupt will be generated if this condition is true.

CosFallingIntReq - This is the logical OR of the COS outputs for the Falling Edge condition. The Falling register will select which bits can be active [enabled]. If any of the enabled bits capture a falling edge this bit will be set. The status is captured before the master interrupt enable. If the master interrupt enable is set an interrupt will be generated if this condition is true.

LevelIntReq is set when an IO bit meets the level interrupt criterion specified by the Polarity, Edge/Level, and Level Interrupt enable. The status is captured before the master interrupt enable. If the master interrupt enable is set an interrupt will be generated if this condition is true.

IntForce is a copy of the IntForce bit from the Base Control Register. This status bit is included to allow a single register read to determine all interrupt types for the design. Clear in the base register. If the master interrupt enable is set an interrupt will be generated if this condition is true.

Write DMA Error Occurred: When a one is read, a write DMA error has been detected. This will occur if there is a target or master abort or if the direction bit in the next pointer of one of the chaining descriptors is a one. A zero indicates that no write DMA error has occurred. This bit is latched and can be cleared by writing back to the Status register with a one in this bit position.

Read DMA Error Occurred: When a one is read, a read DMA error has been detected. This will occur if there is a target or master abort or if the direction bit in the next pointer of one of the chaining descriptors is a zero. A zero indicates that no read DMA error has occurred. This bit is latched and can be cleared by writing back to the Status register with a one in this bit position.

Write DMA Interrupt Occurred: When a one is read, a write DMA interrupt is latched. This indicates that the scatter-gather list for the current write DMA has completed, but the associated interrupt has yet to be completely processed. A zero indicates that no write DMA interrupt is pending.

Read DMA Interrupt Occurred: When a one is read, it indicates that a read DMA interrupt is latched. This indicates that the scatter-gather list for the current read DMA has completed, but the associated interrupt has yet to be completely processed. A zero indicates that no read DMA interrupt is pending.

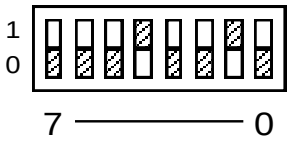
Interrupt Status – Set if the PCI interrupt is asserted. This bit can be checked to determine if this card is causing an interrupt to the system. If set the other bits can be checked to see which feature(s) of the board need to be serviced. Secondary reads to the COS etc. to determine the exact type of interrupt.

BiSerial-VI-GPIO _SWITCH
[$0C Switch and Revision number port read only]

DATA BIT	DESCRIPTION
31-24	Type
23-16	Revision Major
15-8	Revision Minor
7-0	DIP switch

Figure 7 BISERIAL-VI-GPIO Revision, Switch

The DIP Switch is labeled for bit number and '1' '0' in the silk screen. The DIP Switch can be read from this port and used to determine which BiSerial-VI-GPIO is being addressed in a system with multiple cards installed. The DIPswitch can also be used for other purposes – software revision etc. The switch shown would read back 0x12.



RevisionMajor and RevisionMinor are stored and allow SW to determine the feature set of the installed card. Major revisions are updated when large changes occur – new feature etc. Minor Revisions are updated anytime a new FLASH is released. The GPIO Major revision is x01 and the minor revision is x01 currently.

The Type field is provided to allow a read in user space to determine the card type the GPIO is implemented on.

Type	PCB
1	ccPMC-BiSerial-VI
2	PMC-BiSerial-VI

Figure 8 BISERIAL-VI-GPIO Type Table

BiSerial-VI-GPIO_DataOut0

[$18 Data IO Port read/write]

DATA BIT	DESCRIPTION
31-0	Data Out 31-0

Figure 9 BISERIAL-VI-GPIO Data IO Bit Map

The data to be transmitted is written to the Data Output Port side of the Data Register. Reading from this port will return the value of the port independent of other settings. Please see the Direct and Filtered ports for IO side data read.

The output bits are driven onto the IO for the bits that are enabled with the Enable control register, and when the master parallel enable is set. For bits without the Enable register bit set there are no side effects. The Enable register will act as a mask for the data register.

BiSerial-VI-GPIO_TERM

[$10 Data IO Port read/write]

DATA BIT	DESCRIPTION
31-0	Termination 31-0

Figure 10 BISERIAL-VI-GPIO Termination Bit Map

When set ‘1’ the analog switch is closed completing the circuit to put 100 ohms across the corresponding differential pair. When ‘0’ termination is disabled for that bit. Normally enabled for Rx defined bits.

BiSerial-VI-GPIO_DataEn0

[$20 Enable Register bits 31-0 read – write]

DATA BIT	DESCRIPTION
31-0	En31-0

Figure 11 BISERIAL-VI-GPIO Enable Bit Map

The 32 bits of the parallel port direction are controlled with this port. When reset this port is cleared 0x00000000. All IO are set to read [inputs]. To use one or more of the IO for outputs; program the corresponding Enable bit(s) to '1'.

Once a Direction bit is set to output the data in the corresponding output holding register bit is broadcast on that IO line. The data in the holding register will match the data in the data output register if the master parallel enable bit is set. If initial states are important you may want to program the initial data and enable it before enabling the direction bits.

BiSerial-VI-GPIO_Polarity0

[$28 Polarity Reg Port]

DATA BIT	DESCRIPTION
31-0	Polarity 31-0

Figure 12 BISERIAL-VI-GPIO Polarity Reg Bit Map

Each Polarity control bit corresponds to the associated IO bit. When '1' Inverted processing is selected. When '0' non-inverted processing of the IO bit is selected.

BiSerial-VI-GPIO_EdgeLevel0

[$30 EdgeLevel Reg Port]

DATA BIT	DESCRIPTION
31-0	EdgeLevel 31-0

Figure 13 BISERIAL-VI-GPIO EdgeLevel Reg Bit Map

Each EdgeLevel control bit corresponds to the associated IO bit. When '1' Edge processing is selected. When '0' level processing is selected. See Rising and Falling control registers when Edge is selected. See Polarity registers when level is selected.

BiSerial-VI-GPIO_IntEnable0

[$38 IntEnable Reg Port]

DATA BIT	DESCRIPTION
31-0	IntEnable 31-0

Figure 14 BISERIAL-VI-GPIO IntEnable Reg Bit Map

Each IntEnable control bit corresponds to the associated IO bit. When '1' the interrupt from that IO is enabled. Affects both Edge and Level selected IO types.

BiSerial-VI-GPIO_Rising0

$40 Rising Lower Control Register Port read/write

DATA BIT	DESCRIPTION
31-0	Rising 31-0

Figure 15 BISERIAL-VI-GPIO Rising Bit Map

The Rising control register bits correspond to the input data bits. All IO can be set-up for COS activity even if defined as an output. Please see EdgeLevel definition register. In most cases the output bits will be set to '0' for the Rising register. When set '1' and the corresponding input bit transitions from low to high the COS register of rising activity will have the corresponding bit set. If the separate interrupt enable bit is also set an interrupt can be generated. The Rising register is a control register. The COS data is read back separately.

BiSerial-VI-GPIO_Falling0

$48 Falling Lower Control Register Port read/write

DATA BIT	DESCRIPTION
31-0	Falling 31-0

Figure 16 BISERIAL-VI-GPIO Falling Lower Bit Map

The Falling control register bits correspond to the input data bits. All IO can be set-up for COS activity even if defined as an output. In most cases the output bits will be set to '0' for the Falling register. When set '1' and the corresponding input bit transitions from High to Low the COS register of falling activity will have the corresponding bit set. If the separate interrupt enable bit is also set an interrupt can be generated. The Falling register is a control register. The COS data is read back separately.

BiSerial-VI-GPIO_CosRising0

$50 COS Rising Lower Control Register Port read/write

DATA BIT	DESCRIPTION
31-0	COS Rising 31-0

Figure 17 BISERIAL-VI-GPIO COS Rising Bit Map

The COS Rising Edge data is available to read from this port. Write back to clear latched data.

BiSerial-VI-GPIO_CosFalling0

$58 COS Falling Lower Control Register Port read/write

DATA BIT	DESCRIPTION
31-0	COS Falling 31-0

Figure 18 BISERIAL-VI-GPIO COS Falling Bit Map

The COS Falling Edge data is available to read from this port. Write back to clear latched data.

BiSerial-VI-GPIO_Direct0

$60 Direct Data Lower Control Register Port read/write

DATA BIT	DESCRIPTION
31-0	Direct 31-0

Figure 19 BISERIAL-VI-GPIO Direct Read Bit Map

The IO is synchronized to the PCI reference clock and made available from this port. No filtering is performed.

BiSerial-VI-GPIO_Filtered0

$68 Filtered Data Lower Control Register Port read/write

DATA BIT	DESCRIPTION
31-0	Filtered 31-0

Figure 20 BISERIAL-VI-GPIO Filtered Read Bit Map

The IO is synchronized to the PCI reference clock, polarized and masked [POL and EdgeLevel].

BiSerial-VI-GPIO_HalfDiv

[$70 COS clock definition port read -write]

DATA BIT	DESCRIPTION
31-16	spare
15-0	DIVISOR

Figure 21 BISERIAL-VI-GPIO COS Clk Control Bit Map

DIVISOR[15-0] are the clock divisor select bits. The clock source is divided by a 16-bit counter. The output frequency is {reference / [2*N], n≥1. The counter operates from 1 ⇔ N. A pulse is generated when the counter reaches the end point, and the pulse used to create the output clock. The output clock is a square wave as a result.

The 100 MHz differential clock used for the DDR reference is used to create the 50 MHz reference used by the divider.

For a desired frequency of 1 MHz. The required divisor is 50 => N = 25.

Please see ClkCosSel to use this reference or a user programmed PLLA output. The Windows and Linux SW packages support user defined .jed files to program the PLL to custom frequencies. See PLL [Base Reg] for more information.

BiSerial-VI-GPIO_TempData

[$74 Temperature Interface port read -write]

DATA BIT	DESCRIPTION
31-24	spare
23-8	Lm75WriteData
7-5	Lm75Pointer [x80 = PTR Only]
4	Lm75Read
3-1	spare
0	Lm75Write

Figure 22 BISERIAL-VI-GPIO LM75 Interface Bit Map

LM75B is a 400 KHz I2C device. The 32 MHz reference is used to create an 80x reference to the controller built into the GPIO. Write a null pointer to initialize [x81]. The Write and Read bits are auto cleared when the operation is complete. After the initial write completes do a dummy read of the data [x10]. Once the read bit is cleared repeat for data. See reference SW for an example. We use a flag to go through the initialization cycle once. Data is read back in the field shown. After shifting down the data is byte swapped to be in the proper order. Test the sign bit to see if a negative number.

0.125 C is the bit value.

BiSerial-VI-GPIO_FIFO

[$78 FIFO Interface port read -write]

DATA BIT	DESCRIPTION
31-0	FIFO Data

Figure 23 BISERIAL-VI-GPIO FIFO Interface Bit Map

GPIO incorporates an 8Kx32 FIFO for DMA testing purposes. Data can be written to this port and later read from the same port. In addition, this FIFO is used with the DMA engine to load and unload data. Caution, reading data from the port removes it as it is a FIFO.

This port can be redirected to support IO requirements. Please contact Dynamic Engineering for this option

BiSerial-VI-GPIO_WR_DMA_PNTR

[$80 Write DMA Pointer (write only)

DMA Pointer Address Register	
Data Bit	Description
31-2	First Chaining Descriptor Physical Address
1	direction [0]
0	end of chain

Figure 24 BiSerial-VI-GPIO Write DMA pointer register

This write-only port is used to initiate a scatter-gather write [TX] DMA. When the address of the first chaining descriptor is written to this port, the DMA engine reads three successive long words beginning at that address. Essentially this data acts like a chaining descriptor value pointing to the next value in the chain.

The first is the address of the first memory block of the DMA buffer containing the data to read into the device, the second is the length in bytes of that block, and the third is the address of the next chaining descriptor in the list of buffer memory blocks. This process is continued until the end-of-chain bit in one of the next pointer values read indicates that it is the last chaining descriptor in the list.

All three values are on LW boundaries and are LW in size. Addresses for successive parameters are incremented. The addresses are physical addresses the HW will use on the PCI bus to access the Host memory for the next descriptor or to read the data to be transmitted. In most OS you will need to convert from virtual to physical. The length parameter is a number of bytes, and must be on a LW divisible number of bytes.

Status for the DMA activity can be found in the channel control register and channel status register.

Notes:

1. Writing a zero to this port will abort a write DMA in progress.
2. End of chain should not be set for the address written to the DMA Pointer Address Register. End of chain should be set when the descriptor follows the last length parameter.
3. The Direction should be set to ‘0’ for Burst In DMA in all chaining descriptor locations.

BiSerial-VI-GPIO_RD_DMA_PNTR

[$84] Read DMA Pointer (write only)

DMA Pointer Address Register	
Data Bit	Description
31-2	First Chaining Descriptor Physical Address
1	direction [1]
0	end of chain

Figure 25 BiSerial-VI-GPIO Read DMA pointer register

This write-only port is used to initiate a scatter-gather read [RX] DMA. When the address of the first chaining descriptor is written to this port, the DMA engine reads three successive long words beginning at that address. Essentially this data acts like a chaining descriptor value pointing to the next value in the chain.

The first is the address of the first memory block of the DMA buffer to write data from the device to, the second is the length in bytes of that block, and the third is the address of the next chaining descriptor in the list of buffer memory blocks. This process is continued until the end-of-chain bit in one of the next pointer values read indicates that it is the last chaining descriptor in the list.

All three values are on LW boundaries and are LW in size. Addresses for successive parameters are incremented. The addresses are physical addresses the HW will use on the PCI bus to access the Host memory for the next descriptor or to read the data to be transmitted. In most OS you will need to convert from virtual to physical. The length parameter is a number of bytes, and must be on a LW divisible number of bytes.

Status for the DMA activity can be found in the channel control register and channel status register.

Notes:

1. Writing a zero to this port will abort a write DMA in progress.
2. End of chain should not be set for the address written to the DMA Pointer Address Register. End of chain should be set when the descriptor follows the last length parameter.
3. The Direction should be set to ‘1’ for Burst Out DMA in all chaining descriptor locations.

LOOP-BACK & IO Connection Definitions - STD

(cc)PMC-BISERIAL-VI-GPIO can be used with direct end point cabling or with an interface. Dynamic Engineering uses HDEterm68 along with loop-back connections to accomplish loop-back.

The following table shows the connections used with HDEterm68 in the loop-back test. (cc)PMC-BiSerial-VI-GPIO uses 32 Differential IO.

Twisted Pair: Pins shown for P1 SCSI connector and match on HDEterm68
This table is for the ccPMC model and the rear IO version of PMC-BiSerial-VI-GPIO. We use PMC-UNIV-TEST to host the device for rear IO. Other carriers can be used and may cause an adjustment to the table depending on rear IO mapping.

Please note: *BiSerial numbers IO from 0. HDEterm68 numbers from 1. Names are per BiSerial side.*

Connect across the table “from to” Pn4/HDEterm68 numbers

IO-0P	1/1	IO-16P	33/17
IO-0N	3/35	IO-16N	35/51
IO-1P	2/2	IO-17P	34/18
IO-1N	4/36	IO-17N	36/52
IO-2P	5/3	IO-18P	37/19
IO-2N	7/37	IO-18N	39/53
IO-3P	6/4	IO-19P	38/20
IO-3N	8/38	IO-19N	40/54
IO-4P	9/5	IO-20P	41/21
IO-4N	11/39	IO-20N	43/55
IO-5P	10/6	IO-21P	42/22
IO-5N	12/40	IO-21N	44/56
IO-6P	13/7	IO-22P	45/23
IO-6N	15/41	IO-22N	47/57
IO-7P	14/8	IO-23P	46/24
IO-7N	16/42	IO-23N	48/58
IO-8P	17/9	IO-24P	49/25
IO-8N	19/43	IO-24N	51/59
IO-9P	18/10	IO-25P	50/26
IO-9N	20/44	IO-25N	52/60
IO-10P	21/11	IO-26P	53/27
IO-10N	23/45	IO-26N	55/61
IO-11P	22/12	IO-27P	54/28
IO-11N	24/46	IO-27N	56/62
IO-12P	25/13	IO-28P	57/29
IO-12N	27/47	IO-28N	59/63
IO-13P	26/14	IO-29P	58/30
IO-13N	28/48	IO-29N	60/64
IO-14P	29/15	IO-30P	61/31
IO-14N	31/49	IO-30N	63/65
IO-15P	30/16	IO-31P	62/32
IO-15N	32/50	IO-31N	64/66

UART5_TXP	9/17	UART5_RXP	10/18
UART5_TXN	43/19	UART5_RXN	44/20
UART5_CTSP	25/49	UART5_RTSP	26/50
UART5_CTSN	59/51	UART5_RTSN	60/52
UART6_TXP	11/21	UART6_RXP	12/22
UART6_TXN	45/23	UART6_RXN	46/24
UART6_CTSP	27/53	UART6_RTSP	28/54
UART6_CTSN	61/55	UART6_RTSN	62/56
UART7_TXP	13/25	UART7_RXP	14/26
UART7_TXN	47/27	UART7_RXN	48/28
UART7_CTSP	29/57	UART7_RTSP	30/58
UART7_CTSN	63/59	UART7_RTSN	64/60
UART8_TXP	15/29	UART8_RXP	16/30
UART8_TXN	49/31	UART8_RXN	50/32
UART8_CTSP	31/61	UART8_RTSP	32/62
UART8_CTSN	65/63	UART8_RTSN	66/64

Dynamic Engineering Drivers and Reference SW include loop-back tests using the above connections.

PMC PCI Pn1 Interface Pin Assignment

The figure below gives the pin assignments for the PMC Module PCI Pn1 Interface. 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.

TCK	-12V	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 26 (cc)PMC-BISERIAL-VI-GPIO Pn1 Interface

PMC PCI Pn2 Interface Pin Assignment

The figure below gives the pin assignments for the PMC Module PCI Pn2 Interface. 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		1	2
TMS	TDO	3	4
TDI	GND	5	6
GND		7	8
		9	10
	+3.3V	11	12
RST#	BUSMODE3#	13	14
+3.3V	BUSMODE4#	15	16
	GND	17	18
AD30	AD29	19	20
GND	AD26	21	22
AD24	+3.3V	23	24
IDSEL	AD23	25	26
+3.3V	AD20	27	28
AD18		29	30
AD16	C/BE2#	31	32
GND		33	34
TRDY#	+3.3V	35	36
GND	STOP#	37	38
PERR#	GND	39	40
+3.3V	SERR#	41	42
C/BE1#	GND	43	44
AD14	AD13	45	46
GND	AD10	47	48
AD8	+3.3V	49	50
AD7		51	52
+3.3V		53	54
	GND	55	56
		57	58
GND		59	60
	+3.3V	61	62
GND		63	64

Figure 27 (cc)PMC-BISERIAL-VI-GPIO Pn2 Interface

Applications Guide

Interfacing

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

ESD
Proper ESD handling procedures must be followed when handling the (cc)PMC-BISERIAL-VI-GPIO. The card is shipped in an anti-static, shielded bag. The card should remain in the bag until ready for use. When installing the card the installer must be properly grounded and the hardware should be on an anti-static workstation.

Start-up
Make sure that the "system" can see your hardware before trying to access it. Many BIOS will display the PCI devices found at boot up on a "splash screen" with the VendorID and CardId and an interrupt level. Look quickly, if the information is not available from the BIOS then a third party PCI device cataloging tool will be helpful.

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.

We provide the components. You provide the system. Only careful planning and practice can achieve safety and reliability. Inputs can be damaged by static discharge, or by applying voltage outside of the device rated voltages.

Construction and Reliability

Dynamic Engineering Modules are conceived and engineered for rugged industrial environments. (cc)PMC-BISERIAL-VI-GPIO is constructed out of 0.062-inch thick High-Temp ROHS compliant FR4 material.

ROHS and standard processing are available options.

Through-hole and surface-mount components are used. PMC connectors are rated at 1 Amp per pin, 100 insertion cycles minimum. These connectors make consistent, correct insertion easy and reliable.

PMCs are 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 PCB provides a (typical based on PMC) low temperature coefficient of 2.17 W/°C for uniform heat. This is based upon the temperature coefficient of the base FR4 material of 0.31 W/m-°C, and taking into account the thickness and area of the board. 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.

(cc)PMC-BISERIAL-VI-GPIO has internal thermal planes made up of heavy copper power and ground planes. The planes will spread the thermal load over the entire board to minimize hotspots and increase the “coolability”. The components are Industrial temperature rated or better. Thermal vias are added under components to tie in with the thermal plane directly. Where possible devices with thermal ties were chosen to allow direct connection to the ground plane.

Thermal Considerations

The (cc)PMC-BISERIAL-VI-GPIO design consists of CMOS circuits. The power dissipation due to internal circuitry is very low. It is possible to create 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. 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 Street, Suite B&C
Santa Cruz, CA 95060
831-457-8891
support@dyneng.com

Specifications

Host Interface (PCI):	PCI Interface 33 MHz. 32-bit
Serial Interfaces:	32 Differential IO channels. Each IO can be programmed to be input, or output. Inputs can be treated as Edge or Level sensitive. Filtering options include COS, Inversion, Selective interrupts.
Clk Rates supported:	COS programmed to use PLL or 50 MHz with programmable divider.
Software Interface:	Control Registers, FIFOs, and Status Ports
Initialization:	Hardware reset forces all registers to 0 except as noted
Access Modes:	Long-word boundary space (see memory map)
Interrupt:	All IO lines can be used as interrupt sources with programmable rising and or falling activity on IO line “COS”, DMA interrupts
DMA:	Independent controllers for each port TX and each port RX
Onboard Options:	All Options are Software Programmable
Interface Options :	Front or Rear IO [PMC] Rear IO ccPMC.. Front IO via P1 SCSI connector. Rear IO through Pn4.
Dimensions:	Standard Single PMC or ccPMC.
Construction:	High Temp ROHS compliant FR4 Multi-Layer Printed Circuit, Through-Hole and Surface-Mount Components
Temperature Coefficient:	2.17 W/^{o}C for uniform heat across PMC [similar for other formats]
Power	5V & 3.3V used. Local DC\DC converters for 2.5, 1.2, 1.5,.75V rails.

Order Information

Please refer to our (cc)PMC-BISERIAL-VI-GPIO webpage for the most up to date information:
https://www.dyneng.com/ccPMC-BiSerial-VI.html

(cc)PMC-BISERIAL-VI-GPIO	32 programmable differential IO. Independent direction, termination, COS, Interrupt, Polarity, Edge/Level operation. Direct and filtered data read-back. PLL and local oscillators for references.. non-ROHS assembly. Industrial temperature components standard.
-LVDS	Change to LVDS IO instead of RS485.
-RIO	Change to Pn4 IO instead of SCSI connector [PMC Model].
-CC	Add conformal coating option. Recommended for condensing or near condensing environments
-ROHS	Leaded solder is standard on this product. Add -ROHS for ROHS processing.
HDEterm68	https://www.dyneng.com/HDEterm68.html is available as a breakout or for loop-back purposes. Available with several options including connector orientation, DIN rails, Terminal Block, header strip.
HDEcabl68	SCSI cable suitable to interconnect PMC BiSerial III and HDEterm68. Available in various lengths. Twisted shielded construction.

Glossary

Acronyms and other specialized names and their meaning:

PMC	PCI Mezzanine Card - establishes common connectors, connections, size and other mechanical features.
PCI	Peripheral Component Interconnect – parallel bus from host to this device.
VendorID	Manufacturers number for PCI/PCIe boards. DCBA is Dynamic Engineering's ID.
CardID	Unique number assigned to design to distinguish between all designs of a particular vendor.
UART	Universal Asynchronous Receiver Transmitter. Common serialized data transfer with start bit, stop bit, optional parity, optional 7/8 bit data. Can be over any electrical interface. RS232 and RS422 are most common.
Baud	Used as the bit period for this document. Not strictly correct but is the common usage when talking about UARTs.
FIFO	First In First Out Memory
JTAG	Joint Test Action Group – a standard used to control serial data transfer for test and programming operations.
TAP	Test Access Port – basically a multi-state port that can be controlled with JTAG [TMS, TDI, TDO, TCK]. The TAP States are the states in the State machine controlled by the commands received over the JTAG link.

TMS	Test Mode State – this serial line provides the state switching controls. '1' indicates to move to the next state, '0' means stay put in cases where delays can happen, otherwise 0,1 are used to choose which branch to take. Due to complexity of state manipulation the instructions are usually precompiled. Rising edge of TCK valid.
TDI	Test Data In - this serial line provides the data input to the device controlled by the TMS commands. For example the data to program the FLASH comes on the TDI line while the commands to the state-machine to move through the necessary states comes over TMS. Rising edge of TCK valid.
TCK	Test Clock provides the synchronization for the TDI, TDO and TMS signals
TDO	Test Data Out is the shifted data out. Valid on the falling edge of TCK. Not all states output data.
Packet	Group of characters transferred. When the characteristics of a group of characters is known the data can be stored in packets, transferred as such and the system optimized as a result. Any number of characters can be sent.
Packed	When UART characters are always sent/received in groups of 4 allowing full use of host bus / FIFO bandwidth.
UnPacked	When UART characters are sent on an unknown basis requiring single character storage and transfer over the host bus.
MUX	Multiplexor – multiple signals multiplexed to one with a selection mechanism to control which path is active.
Flash	Non-volatile memory used on Dynamic Engineering boards to store FPGA configurations or BIOS.