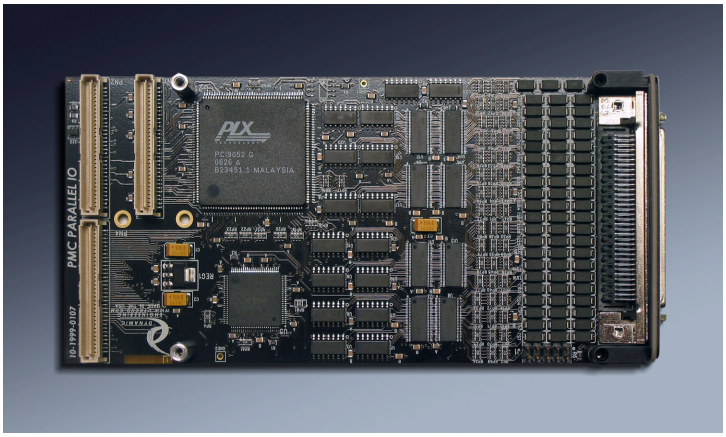
DYNAMIC ENGINEERING
150 DuBois St Suite C, Santa Cruz Ca 95060
831-457-8891
http://www.dyneng.com
sales@dyneng.com
Est. 1988

User Manual

PMC-PARALLEL-IO

"pario"

Digital Parallel Interface
PMC Module



Revision 11p1
Corresponding Hardware: Revision 11
10-1999-0111

PMC-PARALLEL-IO
Digital Parallel Interface
PMC Module

Dynamic Engineering
150 DuBois St. Suite C
Santa Cruz, CA 95060
831-457-8891
831-457-4793 FAX

Revised June 22, 2020

Table of Contents

PRODUCT DESCRIPTION 6

THEORY OF OPERATION 8

ADDRESS MAP 10

PROGRAMMING 14

Register Definitions 14
pario_reg0 14
pario_reg1 15
pario_io0 15
pario_io1 16
pario_cntl0 17
pario_cntl1 18
pario_Revision 19

PMC MODULE LOGIC INTERFACE PIN ASSIGNMENT 20

PMC MODULE LOGIC INTERFACE PIN ASSIGNMENT 21

PMC MODULE FRONT PANEL IO INTERFACE PIN ASSIGNMENT 22

PMC MODULE BACKPLANE IO INTERFACE PIN ASSIGNMENT 23

APPLICATIONS GUIDE 24

Interfacing 24

Construction and Reliability 25

Thermal Considerations 26

WARRANTY AND REPAIR 27
Service Policy 27

Out of Warranty Repairs 27
For Service Contact: 27

SPECIFICATIONS 28

List of Figures

FIGURE 1	PMC-PARALLEL-IO BLOCK DIAGRAM	9
FIGURE 2	PMC-PARALLEL-IO INTERNAL ADDRESS MAP	10
FIGURE 3	PMC-PARALLEL-IO PLX ADDRESS MAP	11
FIGURE 4	PMC-PARALLEL-IO CONTROL PORT 0 BIT MAP	14
FIGURE 5	PMC-PARALLEL-IO CONTROL PORT 1 BIT MAP	15
FIGURE 6	PMC-PARALLEL-IO CONTROL PORT 0 BIT MAP	15
FIGURE 7	PMC-PARALLEL-IO CONTROL PORT 1 BIT MAP	16
FIGURE 8	PMC-PARALLEL-IO CONTROL REGISTER 0 BIT MAP	17
FIGURE 9	PMC-PARALLEL-IO CONTROL REGISTER 1 BIT MAP	18
FIGURE 10	PMC-PARALLEL-IO REVISION	19
FIGURE 11	PMC-PARALLEL-IO PN1 INTERFACE	20
FIGURE 12	PMC-PARALLEL-IO PN2 INTERFACE	21
FIGURE 13	PMC-PARALLEL-IO FRONT PANEL INTERFACE	22
FIGURE 14	PMC-PARALLEL-IO PN4 INTERFACE	23

Product Description

In embedded systems many of the interconnections are made with single ended TTL or CMOS level signals. Depending on the system architecture an IP XMC, or PMC will be the right choice to make the connection. With most architectures you have a choice as there are carriers for cPCI, PCI, PCIe, cPCIe, VME, VPX, PC/104p, PCIe104 and other buses for both PMC and IP mezzanine modules.

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

If you are interested in an IP module solution please refer to the IP-Parallel-IO.
https://www.dyneng.com/ip_parallel_io.html

PMC-Parallel-IO has 64 independent digital IO. The high density makes efficient use of PMC position resources. The IO is available for system connection both through the front panel and via the rear [Pn4] connector. A high density 68 pin SCSI III front panel connector provides the front panel IO. The FPIO lines are protected with transorbs. The rear panel IO has a PIM and PIM Carrier available for rear panel wiring options. HDEterm68 can be used as a breakout for the front or rear panel IO. HDEcabl68 provides a convenient cable.
https://www.dyneng.com/HDEterm68.html
https://www.dyneng.com/HDEcabl68.html

Please note: PMC-Parallel-TTL and XMC-Parallel-TTL should be considered for new designs. The new designs have additional features i.e. all lines can be interrupters with COS, PLL support, internal memory. In addition newer software [OS] support is available for the updated models. *In order to serve our clients throughout their system life cycles; PMC-Parallel-IO is available for purchase and will continue to be available.*

Each IO is programmable to be input or output on a channel-by-channel basis. Two IO can be used as interrupt generators. Interrupts are programmable to be based on level or edge and active high or low. An external clock and clock enable can be used or the internal clock selected for capturing the Input channels. The PCI clock can be divided by several programmable divisors to provide the right sampling rate for your application.

12 of the IO are routed through the Xilinx CPLD [with Rev 11 switched from Altera to Xilinx] 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 driven low with open-drain high current drivers. The high side is set with pull-up resistors. IO 7-0 have 3 pull-up locations per IO line. IO 63-8 have 2 pull-up locations per line. The default is for 470 ohms. The multiple locations allow for pull-up strengths greater than 470 ohms and to stay within the resistor wattage capabilities. The multiple positions also allow for parallel combinations to create more options of specific pull-up values.

Please contact the factory to arrange for custom implementations. We will design and test your modifications. The new version of the board will be documented and assigned a modified part number to allow revision control and re-ordering in the future. Modifications can include changes to the Xilinx program, resistor values or other changes that you may need.

The Xilinx can be programmed via the PMC JTAG interface and programming adapter. Specialized files can be e-mailed and installed in the field should you want to customize the design.

IO registers are mapped as 32 bit words and support byte, word and 32 bit access. Internal registers [within Xilinx] are byte wide, LW aligned. The Windows® compatible driver package is 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 manual is also available on-line.

PMC-PARALLEL-IO is part of the PMC Module family of modular I/O components. PMC-PARALLEL-IO conforms to the PMC standard. This guarantees compatibility with multiple PMC Carrier boards. Because the PMC may be mounted on different form factors, while maintaining plug and software compatibility, system prototyping may be done on one PMC Carrier board, with final system implementation on a different one.

Theory of Operation

PMC-PARALLEL-IO can be used for multiple purposes with applications in telecommunications, control, sensors, IO, test; anywhere multiple independent IO are useful.

PMC-PARALLEL-IO features an industry standard PLX 9052, Xilinx CPLD, and high current driver devices. The CPLD contains the local interface and control required for the parallel interface. The PLX device acts as a bridge between the PCI and local bus. The basic initialization and PLX programming requirements are covered in this manual. Additional information is available in the PLX manual and may be of interest in some cases.

The Xilinx design contains several registers to allow user selection of Interrupt Enable, Rising and Falling edges, Pulse or Level, Clock divisor, and a new feature – FLASH revision.

Data written to the TX IO registers will be placed on the bus. The drivers are initialized to the off state and pull-ups on board hold the IO lines in the 'high' state. At the end of a PCI reset cycle the FPGA creates a special write cycle to initialize the registers with 0xffffffff.

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 un-driven by the local driver and the output level will be controlled by the termination resistor and any other drivers attached to that line. The control register is read-writeable. Because the register is independent of the bus; the data read will always match the data written allowing read-modify-write operations.

A separate read-back register provides access to the IO bus side of the drivers. The 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. If desired the internal clock can be replaced with an external source and an enable. The basic option is available under SW control. If special programming is needed please contact Dynamic Engineering for a custom CPLD implementation.

The register IO positions corresponding to 63-52 are routed through the FPGA and back out to the control lines. If desired, these lines can be programmed to create custom timing pulses etc. For example if the interface is to put out an address and then an address qualifier to strobe the address into the receiving hardware one of the 12 control lines can be programmed to create a pulse some time after the address for the IO registers is written to. The custom pulse will be more accurate for delay and duration than a SW timing solution. The number of accesses to the card can be reduced as well having the effect of greater throughput. Please contact Dynamic Engineering with your requirements.

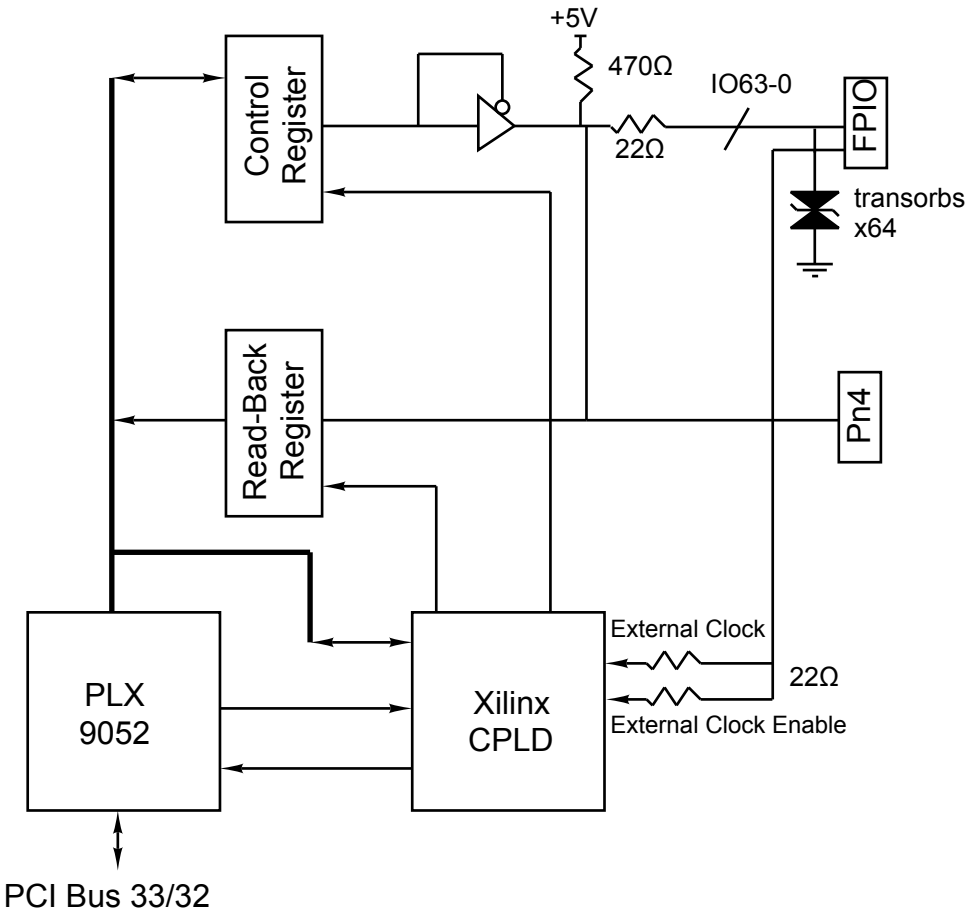


FIGURE 1 PMC-PARALLEL-IO BLOCK DIAGRAM

Address Map

Function	Offset	Width
// PMC relative addresses //		
#define pario_reg0	0x00	32
#define pario_reg1	0x04	32
#define pario_io0	0x10	32
#define pario_io1	0x14	32
#define pario_cntl0	0x20	8 [D7-0]
#define pario_cntl1	0x24	8 [D7-0]
#define pario_Revision	0x30	8 Major.Minor

FIGURE 2 PMC-PARALLEL-IO INTERNAL ADDRESS MAP

The address map provided is for the local decoding performed within PMC-Parallel-IO. The addresses are all offsets from a base address. The carrier board the PMC is installed into provides the base address. Dynamic Engineering prefers a long-word oriented approach because it is more consistent across platforms.

The host system will search the PCI bus to find the assets installed during power-on initialization. The VendorId = 0x10B5 and the CardId = 0x9050 for the PMC-Parallel-IO. If custom vendor and card data are needed we can install the optional EEPROM.

Once the initialization process has occurred and the system has assigned addresses to the PMC-Parallel-IO card then software will need to determine what the address space is for the PLX device [BAR0] and the rest of the hardware [BAR1].

The next step is to initialize the PMC-Parallel-IO. The PLX is first initialized to allow accesses and then the registers need to be set-up.

The PLX internal register addresses accessible from the PCI bus are in the next table.

```
/* PLX 9052 Addresses  Relative to base address of PLX */
#define PLX_LAS0R       0x00 // Local Address Space 0 Range
#define PLX_LAS1R       0x04 // Local Address Space 1 Range
#define PLX_LAS2R       0x08 // Local Address Space 2 Range
#define PLX_LAS3R       0x0c // Local Address Space 3 Range
#define PLX_LERR        0x10 // Local Expansion ROM Range
#define PLX_LAS0LBA     0x14 // Local Address Space 0 Local Base Add
#define PLX_LAS1LBA     0x18 // Local Address Space 1 Local Base Add
#define PLX_LAS2LBA     0x1c // Local Address Space 2 Local Base Add
#define PLX_LAS3LBA     0x20 // Local Address Space 3 Local Base Add
#define PLX_ERLBA       0x24 // Expansion ROM Local Base Address
#define PLX_LAS0BRD     0x28 // Local Address Space 0 Bus Region Desc
#define PLX_LAS1BRD     0x2c // Local Address Space 1 Bus Region Desc
#define PLX_LAS2BRD     0x30 // Local Address Space 2 Bus Region Desc
#define PLX_LAS3BRD     0x34 // Local Address Space 3 Bus Region Desc
#define PLX_ERBRD       0x38 // Expansion ROM Bus Region Descriptor
#define PLX_CS0BASE     0x3c // Chip Sel 0 Base Address Register Desc
#define PLX_CS1BASE     0x40 // Chip Sel 1 Base Address Register Desc
#define PLX_CS2BASE     0x44 // Chip Sel 2 Base Address Register Desc
#define PLX_CS3BASE     0x48 // Chip Sel 3 Base Address Register Desc
#define PLX_ICS         0x4c // Interrupt Control & Status
#define PLX_CNTRL       0x50 // user IO, PCI target Response, EEPROM
```

FIGURE 3 PMC-PARALLEL-IO PLX ADDRESS MAP

The following is the initialization sequence that we use for our test software in a Windows® environment. A Windows® driver is now available to support your project. We use MS Studio in conjunction with the driver to write our test software.

The following set-up code is provided to allow you to write your own initialization procedure if you are not using the Dynamic Engineering Driver. The driver automatically takes care of the initialization.

The initialization software will need to locate the hardware in the system and determine the base address assigned by the system. The VendorId = 0x10B5 and the CardId = 0x9050 for the PMC-Parallel-IO

Example of PLX Initialization Programming

Write 0x00000001 to PLX_LAS0BA (14h) // Local Address Space 0 Remap
Write 0x00800022 to PLX_LAS0BRD (28h) //Local Address Space control CS 0 en
Write 0x00000201 to PLX_CS0BASE (3Ch) // enable CS0, size = 256, locate at 0
Write 0x0 to PLX_ INTCSR (4Ch) // Interrupts disabled, PCI mode
Write 0x00200904 to PLX_CNTRL (50h)// PLX Control Register

Example of Xilinx Register Programming

// all bits off
Write 0xFFFFFFFF to pario_reg0
Write 0xFFFFFFFF to pario_reg1
// internal clock 8 mhz - input register reference rate
Write 0x04 to pario_cntl0
// internal interrupts off, level, active low
Write 0x00 to pario_cntl1

After the initialization the PLX is set-up with a small memory segment allocated to the PMC-Parallel-IO and the appropriate decoding enabled. All of the IO are in the "off" state, the input is being sampled and the interrupts are disabled. The hardware is now ready for use. The programming section will provide the details of the control registers and more about how to use the hardware.

Interrupts require the PLX device to be programmed to pass them from the PMC Parallel IO design through to the PCI bus, and the Xilinx control registers programmed to create an interrupt based on the IO state. The control of the interrupt capture is performed within the Xilinx device.

In level mode the PLX is programmed to pass the interrupts through to the PCI bus. The interrupt is cleared at the Xilinx device.

When using pulse mode, the Xilinx generates a pulse based on the IO transition [high or low as programmed] the PLX Bridge captures the pulse, and generates INTA. SW will need to clear the interrupt within the PLX for this mode.

For level mode (initialization):

1. Enable interrupts on the Xilinx by writing 0x44 to the PARIO_CNTL1
2. Enable interrupts on PLX writing 0x49 to PLX_INTCSR.

For Pulse/Edge Mode (initialization):

1. Enable interrupts and pulse/edge Mode by writing 0x66 to PARIO_CNTL1
2. Enable Interrupts on PLX and set chip to pulse/edge mode with high polarity (required for this mode) by writing 0x35B to PLX_INTCSR

Changing Modes:

When switching from pulse/edge mode to level (and vice versa), it is important to first disable the PCI interrupts on the PLX chip by writing a '0' to bit 6 on the PLX_INTCSR. The user must also ensure interrupts have not been triggered during mode change before reinitializing that same bit to '1'.

For example, when switching from level to pulse/edge mode (assuming current low polarity settings):

1. Disable PCI by writing 0x00 to the PLX_INTCSR
2. Write 0xFFFF_FFFF to the PARIO_REG0 and PARIO_REG1 registers.
3. Setup the Xilinx in pulse/edge mode by writing 0x66 to PARIO_CNTL1 (polarity choices may change this value).
4. Setup PLX by writing 0x31B to PLX_INTCSR
5. Because this change may trigger an interrupt, clear pulse/edge level interrupts by writing 0xF1B to PLX_INTCSR (which is the same values as before, but adding 1's to the pulse/edge mode clearing bits.
6. Finally, re-enable the PCI interrupts by writing 0x35B to PLX_INTCSR

NOTE: If interrupts are being triggered with the Xilinx in level mode, they need to be cleared by disabling and reenabling interrupts using the PARIO_CNTL1 bits 2 and 6. It is up to the SW to know when the bits can be reenabled based on IO state.

The Xilinx provides level and edge triggered capabilities on each of the two interrupt lines. The Xilinx provides interrupt masks for each level. Please refer to the programming section bit maps for more information. If not using interrupts with your application use the default setting or write 0x00 to the PLX_ICS register.

Programming

Programming PMC-PARALLEL-IO requires only the ability to read and write data in the host's PMC space. The base address is determined by the PMC Carrier board. This documentation refers to the address offset where BAR1 is allocated as the base address.

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

pario_reg0

[$00 parallel-io Control Register Port read/write]

CONTROL REGISTER 0	
DATA BIT	**DESCRIPTION**
31-0	IO31-IO0

FIGURE 4 PMC-PARALLEL-IO CONTROL PORT 0 BIT MAP

The lower 32 bits of the 64 IO lines are controlled through this port. The port is long word aligned and responds to byte, word, and long word accesses. The port is read-write capable. The port is on the control side of the IO lines. The read data will match the write data. The IO controls are active low with the bits controlling the output enables of the drivers. When high, the output for that line is off allowing the pull-ups on board to create a '1' on that line unless some other driver is active. The IO line side of the port is available through the IO read-back port.

pario_reg1

[$04 parallel-io Control Register Port read/write]

CONTROL REGISTER 0	
DATA BIT	**DESCRIPTION**
31-0	IO63-IO32

FIGURE 5 PMC-PARALLEL-IO CONTROL PORT 1 BIT MAP

The upper 32 bits of the 64 IO lines are controlled through this port. The port is long word aligned and responds to byte, word, and long word accesses. The port is read-write capable. The port is on the control side of the IO lines. The read data will match the write data. The IO controls are active low. When high, the output for that line is off allowing the pull-ups on board to create a '1' on that line unless some other driver is active. The IO line side of the port is available through the IO read-back port.

pario_io0

[$10 parallel-io Read Back Port read only]

CONTROL REGISTER 0	
DATA BIT	**DESCRIPTION**
31-0	IO31-IO0

FIGURE 6 PMC-PARALLEL-IO CONTROL PORT 0 BIT MAP

The lower 32 bits of the 64 IO lines are read through this port. The port is on the IO side of the IO lines. The data read may not match the control register data pattern as some of the bits which are set high may be driven low externally. The data is sampled with a user selectable clock rate. Please refer to the pario_cntl0 definition for more details.

pario_io1

[$14 parallel-io Read Back Port read only]

CONTROL REGISTER 0	
DATA BIT	**DESCRIPTION**
31-0	IO63-IO32

FIGURE 7 PMC-PARALLEL-IO CONTROL PORT 1 BIT MAP

The upper 32 bits of the 64 IO lines are read through this port. The port is on the IO side of the IO lines. The data read may not match the control register data pattern as some of the bits which are set high may be driven low externally. The data is sampled with a user selectable clock rate. Please refer to the pario_cntl0 definition for more details.

pario_cntl0

[$20 parallel-io Control Register Port read/write]

CONTROL REGISTER 0	
DATA BIT	**DESCRIPTION**
7	spare
6	CLKSEL_2
5	CLKSEL_1
4	CLKSEL_0
3	spare
2	INT_CLK_EN
1	CLK_EN_SEL
0	CLK_SEL

FIGURE 8 PMC-PARALLEL-IO CONTROL REGISTER 0 BIT MAP

The CLK_SEL bit is used to select either the internal or external source for the IO read-back register clock. 0 = internal 1 = external.

CLK_EN_SEL is used to pick the internal or external clock enable for the IO read-back registers. internal = 0, external = 1.

INT_CLK_EN is used to supply the clock enable for the IO read-back registers. 0 = disabled. 1 = enabled. Only has effect when the CLK_EN_SEL bit is set to internal.

Disabling will mean the data read from upper and lower are from the same sample time. Leaving enabled means reading from the upper will not be the same sample period as the lower since there is some delay between accesses.

CLKSEL_2,1,0 are the clock divisor select bits. The PMC clock is divided by a counter and the select bits pick which clock is used to drive the IO read-back registers. These bits have no effect when the external clock is selected.

CLKSEL2,1,0	divisor	typical clock rate [33 Mhz. PMC reference]
000	4	8.25 MHz.
001	8	4.125 MHz.
010	16	2.063 Mhz.
011	32	1.031 Mhz.
100	64	515.6 Khz.
101	128	257.8 Khz.
110	256	128.9 KHz.
111	512	64.5 KHz.

pario_cntl1

$24 Parallel-IO Control Register Port read/write

CONTROL REGISTER 1	
DATA BIT	**DESCRIPTION**
7	
6	SEL1_EN
5	SEL1_P/L
4	SEL1_HI/LO
3	
2	SEL0_EN
1	SEL0_P/L
0	SEL0_HI/LO

FIGURE 9 PMC-PARALLEL-IO CONTROL REGISTER 1 BIT MAP

CNTL1 register is used to control the action of the programmable interrupts. '0' refers to IRQ0 which is connected to LINT1 and '1' refers to IRQ1 which is connected to LINT2 on the PLX device. Both interrupts are mapped to INTA on the PMC interface. The source of the interrupt can be determined by reading the internal PLX register. The interrupt sources INT0 and INT1 correspond to IO0 and IO1 respectively. IO0/INT0 causes an interrupt on IRQ0/LINT1 and IO1/INT1 cause an interrupt on IRQ1/LINT2 when active and enabled.

SELx_HI/LO is used to select whether that interrupt level is active HI or active LO. 0 = active LO, falling edge 1 = active HI, rising edge. In level mode if active low is selected, when the IO is low an interrupt is generated with an active low signal.

SELx_P/L is used to select whether that interrupt level is pulsed or level. 0 = level. 1 = pulsed. Note that SEL_HI/LO affects the sense of the edge or level.

In level mode the state of the IO combined with the Hi/Lo setting is used to create an active low LINT signal.

Pulsed interrupts use the transition from high -> low or low -> high to create an active high pulse.

The PLX bridge will need to be programmed to either pass the interrupt request through [level] or capture the pulse and assert the interrupt.

SELx_EN enables the IO line interrupt. 0 = disabled, 1 = enabled.

Level sensitive interrupts are cleared by changing the external event or masking off with the enable bit until the trigger is no longer active. The state of the bits can be read through the IO read-back register.
Pulsed interrupts do not require clearing at the Xilinx. The event causing the interrupt is detected and a pulse generated. A second pulse will be generated when the programmed event happens again. The PLX side will need to be cleared in this case. In edge mode the pulse is captured by the bridge and held until cleared by SW.

pario_Revision

$30 Parallel-IO Revision Control Port read only

CONTROL REGISTER 1	
DATA BIT	**DESCRIPTION**
7-4	Revision Major
3-0	Revision Minor

FIGURE 10 PMC-PARALLEL-IO REVISION

With Revision 11 a new register is added to the design. The major revision will be updated to track the PCB or other larger changes. The minor revision is updated for smaller adjustments, minor corrections etc.

0xB.0 Switch to Xilinx CPLD
0xB.1 Correct edge triggered interrupt handling

PMC Module Logic Interface Pin Assignment

The figure below gives the pin assignments for the PMC Module PCI Pn1 Interface on the PMC-Parallel-IO. 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
GND	INTA#	3	4
		5	6
BUSMODE1#	+5V	7	8
		9	10
GND -		11	12
CLK	GND	13	14
GND -		15	16
	+5V	17	18
	AD31	19	20
AD28-	AD27	21	22
AD25-	GND	23	24
GND -	C/BE3#	25	26
AD22-	AD21	27	28
AD19	+5V	29	30
	AD17	31	32
FRAME#-	GND	33	34
GND	IRDY#	35	36
DEVSEL#	+5V	37	38
GND	LOCK#	39	40
		41	42
PAR	GND	43	44
	AD15	45	46
AD12-	AD11	47	48
AD9-	+5V	49	50
GND -	C/BE0#	51	52
AD6-	AD5	53	54
AD4	GND	55	56
	AD3	57	58
AD2-	AD1	59	60
	+5V	61	62
GND		63	64

FIGURE 11 PMC-PARALLEL-IO PN1 INTERFACE

PMC Module Logic Interface Pin Assignment

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

FIGURE 12 PMC-PARALLEL-IO PN2 INTERFACE

PMC Module Front Panel IO Interface Pin Assignment

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

EXT_CLK_EN	EXT_CLK	1	35
IO_31	IO_63	2	36
IO_30	IO_62	3	37
IO_29	IO_61	4	38
IO_28	IO_60	5	39
IO_27	IO_59	6	40
IO_26	IO_58	7	41
IO_25	IO_57	8	42
IO_24	IO_56	9	43
IO_23	IO_55	10	44
IO_22	IO_54	11	45
IO_21	IO_53	12	46
IO_20	IO_52	13	47
IO_19	IO_51	14	48
IO_18	IO_50	15	49
IO_17	IO_49	16	50
IO_16	IO_48	17	51
IO_15	IO_47	18	52
IO_14	IO_46	19	53
IO_13	IO_45	20	54
IO_12	IO_44	21	55
IO_11	IO_43	22	56
IO_10	IO_42	23	57
IO_9	IO_41	24	58
IO_8	IO_40	25	59
IO_7	IO_39	26	60
IO_6	IO_38	27	61
IO_5	IO_37	28	62
IO_4	IO_36	29	63
IO_3	IO_35	30	64
IO_2	IO_34	31	65
IO_1	IO_33	32	66
IO_0	IO_32	33	67
GND	GND	34	68

FIGURE 13 PMC-PARALLEL-IO FRONT PANEL INTERFACE

PMC Module Backplane IO Interface Pin Assignment

The figure below gives the pin assignments for the PMC Module IO Interface on PMC-Parallel_IO and routed to Pn4. Also see the User Manual for your carrier board for more information.

IO_0	IO_1	1	2
IO_2	IO_3	3	4
IO_4	IO_5	5	6
IO_6	IO_7	7	8
IO_8	IO_9	9	10
IO_10	IO_11	11	12
IO_12	IO_13	13	14
IO_14	IO_15	15	16
IO_16	IO_17	17	18
IO_18	IO_19	19	20
IO_20	IO_21	21	22
IO_22	IO_23	23	24
IO_24	IO_25	25	26
IO_26	IO_27	27	28
IO_28	IO_29	29	30
IO_30	IO_31	31	32
IO_32	IO_33	33	34
IO_34	IO_35	35	36
IO_36	IO_37	37	38
IO_38	IO_39	39	40
IO_40	IO_41	41	42
IO_42	IO_43	43	44
IO_44	IO_45	45	46
IO_46	IO_47	47	48
IO_48	IO_49	49	50
IO_50	IO_51	51	52
IO_52	IO_53	53	54
IO_54	IO_55	55	56
IO_56	IO_57	57	58
IO_58	IO_59	59	60
IO_60	IO_61	61	62
IO_62	IO_63	63	64

FIGURE 14 PMC-PARALLEL-IO PN4 INTERFACE

Applications Guide

Interfacing

The pin-out tables are displayed with the pins in the same relative order as the actual connectors. 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. Open Drain interface devices provide some immunity from and allow operation when part of the circuit is powered on and part is not. It is better to avoid the issue of going past the safe operating areas by powering the equipment together and by having a good ground reference.

Keep cables short. Flat cables, even with alternate ground lines, are not suitable for long distances. The PMC-Parallel-IO has transorbs for input protection. The connector is pinned out for a standard SCSI II/III cable to be used. It is suggested that this standard cable be used for most of the cable run.

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

We provide the components. You provide the system. Safety and reliability can be achieved only by careful planning and practice. Inputs can be damaged by static discharge, or by applying voltage outside of the particular device's rated voltages.

Construction and Reliability

PMC Modules were conceived and engineered for rugged industrial environments. PMC-Parallel-IO is constructed out of 0.062 inch thick ROHS compliant, high temperature FR4 material.

The PMC Module connectors are keyed and shrouded with Gold plated pins on both plugs and receptacles. They are rated at 1 Amp per pin, 100 insertion cycles minimum. These connectors make consistent, correct insertion easy and reliable.

The PMC is secured against the carrier with the connectors and front panel. If more security against vibration is required the stand-offs can be secured against the carrier.

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

Thermal Considerations

PMC-PARALLEL-IO 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. 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 St. Suite C
Santa Cruz, CA 95060
831-457-8891
support@dyneng.com

Specifications

Logic Interface:	PMC Logic Interface [PCI]
Digital Parallel IO:	64 discrete IO channels. Each individually controllable to be Input or Output
CLK rates supported:	8 MHz - 64 KHz reference input sample rate, plus external clock and clock enable
Software Interface:	Control Registers, IO registers, IO Read-Back registers
Initialization:	Programming procedure documented in this manual
Access Modes:	LW, Word or Byte to IO registers Word Control registers
Access Time:	Complete access with PCI overhead in less than 500 nS.
Interrupt:	2 IO lines can be used as interrupt sources with programmable level/edge, active Hi/LO and enables.
DMA:	No DMA Support implemented at this time
Onboard Options:	All Options are Software Programmable
Interface Options:	68 Pin SCSI III connector at front bezel User IO routed to Pn4
Dimensions:	Standard Single PMC Module.
Construction:	High Temperature, FR4 Multi-Layer Printed Circuit, Through Hole and Surface Mount Components.
Temperature Coefficient:	2.17 W/^{o}C for uniform heat across PMC
Power:	Typ. 460 mA @ 5V outputs off Add 10 mA per active low output for pull-up current drivers support 64 mA per IO line, higher currents are possible depending on load.

Order Information

extended temperature range -40 - 85ºC

PMC-Parallel-IO	PMC Module with 64 TTL IO Front and Rear IO
-FP	Only Bezel IO installed.
-RIO	Only Pn4 IO installed.
-ROHS	Switch to ROHS processing.
-CC	Add conformal coating

Related

PCI2PMC	PMC to PCI adapter to allow installation of PMC-Parallel-IO into a PCI system. http://www.dyneng.com/pci2pmc.html
PCIeBPMCX1	PMC to PCIe adapter to allow installation of PMC-Parallel-IO into a PCIe system. http://www.dyneng.com/pciebpmcx1.html
HDEterm68	68 position terminal block with two SCSI II connectors. PMC-Parallel-IO compatible. http://www.dyneng.com/HDEterm68.html
HDEcabl68	SCSI II/III cable compatible with FPIO on PMC Parallel IO. http://www.dyneng.com/HDEcabl68.html
PIM-Parallel-IO	PMC IO Module for PMC Parallel IO design. Provides FPIO in cPCI systems when used with a PIM Carrier http://www.dyneng.com/pim_parallel_io.shtml