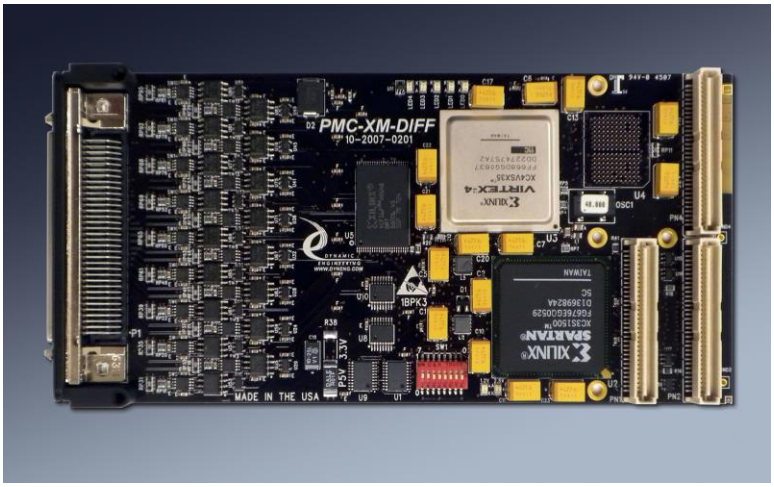
DYNAMIC ENGINEERING
150 DuBois, Suite C
Santa Cruz, CA 95060
(831) 457-8891 **Fax** (831) 457-4793
http://www.dyneng.com
sales@dyneng.com
Est. 1988

User Manual

PMC-XM-DIFF

Interface Module with Re-configurable I/O logic
RS-485 or LVDS or mixed
34 Differential Pairs at Bezel
32 Differential Pairs at Pn4



Revision B
Corresponding Hardware: Revision A/B
10-2007-0201/0202
Corresponding Firmware: Revision H

PMC-XM-DIFF
PMC based interface module
With Re-programmable I/O logic

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

Manual Revision B. Revised June, 17 2015

Table of Contents

PRODUCT DESCRIPTION 6

THEORY OF OPERATION 7

PROGRAMMING 8

ADDRESS MAP SPARTAN3 9

Register Definitions 10
XM_BASE 10
XM_BASE_USER_SWITCH 12
XM_CHAN0/1_CNTRL 14
XM_CHAN0/1_STATUS 16
XM_CHAN0/1_WR/RD_DMA_PNTR 19
XM_CHAN0/1_FIFO 19
XM_CHAN0/1_TX_AMT_LVL 20
XM_CHAN0/1_RX_AFL_LVL 20
XM_CHAN0/1_TX/RX_FIFO_COUNT 21

ADDRESS MAP: VIRTEX ATP DESIGN 22

Register Definitions 23
ATP_VTX_BASE 23
ATP_VTX_STATUS 25
ATP_VTX_IO_TERM, ATP_VTX_IO_DIR, ATP_VTX_IO_DOUT 26
ATP_VTX_IO_UPPER 26
ATP_VTX_IO_DIN 27
ATP_VTX_CHAN0/1_CNTRL 27
ATP_VTX_CHAN0/1_STATUS 29
ATP_VTX_TX0/1_FIFO 31
ATP_VTX_RX0/1_FIFO 31
ATP_VTX_TX0/1_DCOUNT 31
ATP_VTX_RX0/1_DCOUNT 32
ATP_VTX_TX0/1_AMT_CNT 32
ATP_VTX_RX0/1_AFL_CNT 33

I/O Data Loopback Testing 34

VIRTEX PIN OUT 35

PMC PCI PN1 INTERFACE PIN ASSIGNMENT 43

PMC PCI PN2 INTERFACE PIN ASSIGNMENT 44

FRONT PANEL I/O PIN ASSIGNMENT 45

PMC MODULE BACKPLANE IO INTERFACE PIN ASSIGNMENT 46

APPLICATIONS GUIDE 47

Interfacing 47

Construction and Reliability 48

Thermal Considerations 48

WARRANTY AND REPAIR 49

Service Policy 49
Out of Warranty Repairs 49

For Service Contact: 49

SPECIFICATIONS 50

ORDER INFORMATION 51

List of Figures

FIGURE 1 PMC-XM-DIFF BLOCK DIAGRAM 6
FIGURE 2 PMC-XM SPARTAN3 XILINX ADDRESS MAP 9
FIGURE 3 PMC-XM SPARTAN3 BASE CONTROL REGISTER 10
FIGURE 4 PMC-XM SPARTAN3 USER SWITCH PORT 12
FIGURE 5 PMC-XM SPARTAN3 STATUS REGISTER 13
FIGURE 6 PMC-XM SPARTAN3 CHANNEL CONTROL REGISTER 14
FIGURE 7 PMC-XM SPARTAN3 CHANNEL STATUS REGISTER 16
FIGURE 8 PMC-XM SPARTAN3 CHANNEL DMA POINTER REGISTER 19
FIGURE 9 PMC-XM SPARTAN3 CHANNEL FIFO PORT 19
FIGURE 10 PMC-XM SPARTAN3 CHANNEL TX ALMOST EMPTY REGISTER 20
FIGURE 11 PMC-XM SPARTAN3 CHANNEL RX ALMOST FULL REGISTER 20
FIGURE 12 PMC-XM SPARTAN3 CHANNEL TX/RX FIFO COUNT PORT 21
FIGURE 13 PMC-XM VIRTEX (ATP) XILINX ADDRESS MAP 22
FIGURE 14 PMC-XM VIRTEX (ATP) BASE CONTROL REGISTER 23
FIGURE 15 PMC-XM VIRTEX (ATP) BASE STATUS REGISTER 25
FIGURE 16 PMC-XM VIRTEX (ATP) I/O REGISTER 26
FIGURE 17 PMC-XM VIRTEX (ATP) I/O UPPER REGISTER 26
FIGURE 18 PMC-XM VIRTEX (ATP) I/O DIN REGISTER 27
FIGURE 19 PMC-XM VIRTEX (ATP) CHANNEL CONTROL REGISTER 27
FIGURE 20 PMC-XM VIRTEX (ATP) CHANNEL STATUS REGISTER 29
FIGURE 21 PMC-XM VIRTEX (ATP) CHANNEL TX FIFO PORT 31
FIGURE 22 PMC-XM VIRTEX (ATP) CHANNEL RX FIFO PORT 31
FIGURE 23 PMC-XM VIRTEX (ATP) CHANNEL TX FIFO COUNT PORT 31
FIGURE 24 PMC-XM VIRTEX (ATP) CHANNEL RX FIFO COUNT PORT 32
FIGURE 25 PMC-XM VIRTEX (ATP) TX FIFO ALMOST EMPTY PORT 32
FIGURE 26 PMC-XM VIRTEX (ATP) RX FIFO ALMOST FULL PORT 33
FIGURE 27 PMC-XM PN1 INTERFACE 43
FIGURE 28 PMC-XM PN2 INTERFACE 44
FIGURE 29 PMC-XM FRONT PANEL INTERFACE 45
FIGURE 30 PMC-XM PN4 INTERFACE 46

Product Description

The PMC-XM-DIFF features a Xilinx Spartan3-2000 676 pin FPGA to implement the PCI interface and two independent I/O channels. Each channel has a separate input and output scatter-gather DMA engine to move data to/from host memory over the local 32-bit 33 MHz PCI bus. A Xilinx Virtex4 668 pin FPGA interfaces between the Spartan3 and the I/O. The I/O can be populated with any combination of RS-485 or LVDS I/O devices.

Each I/O has separate direction, and termination controls to allow any combination of inputs and outputs. Impedance controlled and length matched within the mil [.001”] to allow for any user requirement.

Other features include on-board PLL, 8-position DIP Switch, Built in DMA, and user LED’s. Optional 1Mx36-bit QDDRII RAM and 13-bit Temperature Sensor are available.

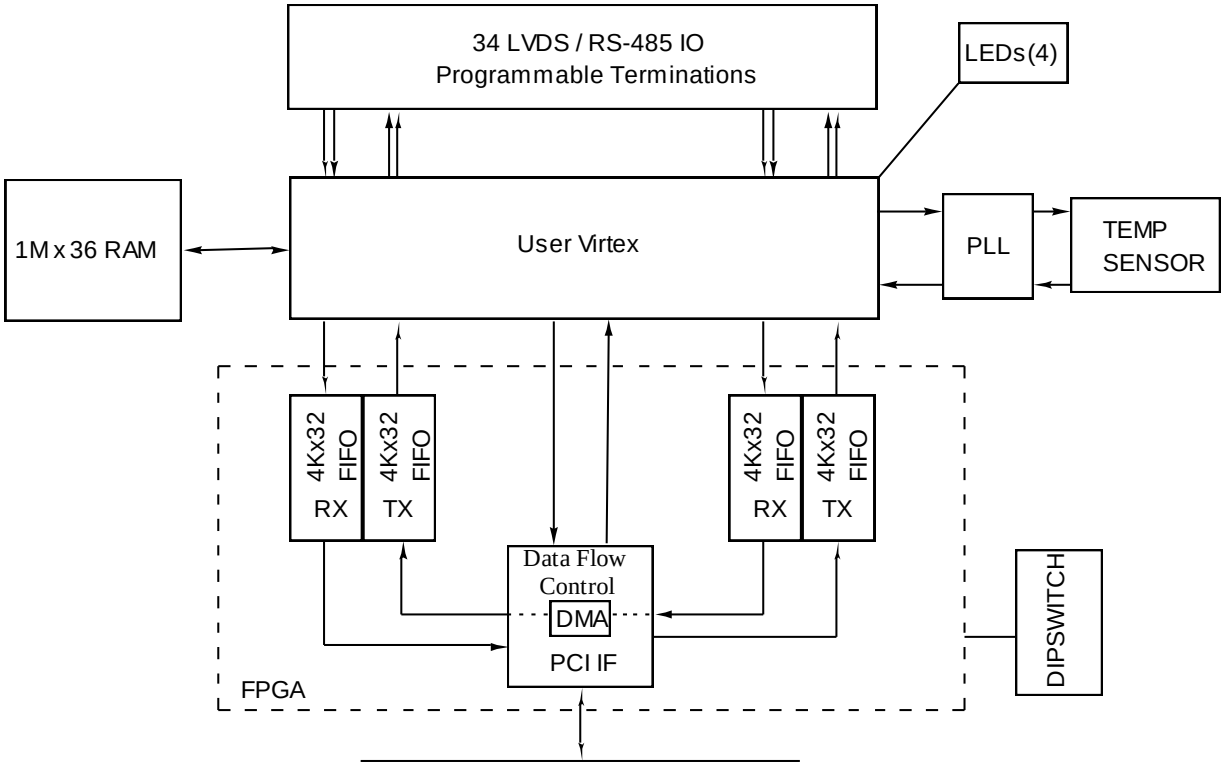


FIGURE 1 PMC-XM-DIFF BLOCK DIAGRAM

The engineering kit comes with a basic design for the Virtex consisting of the VHDL package used to generate the ATP implementation. The design includes decoding, DMA, two channels, I/O loop-back and more. The package can include the Windows® driver and reference application code. The reference software source files are provided and can be modified by the user to do whatever is desired. The package includes an auto design detection feature to automatically load Virtex device drivers corresponding to different Virtex designs. The user can change the design number and use the generic driver to access new features added to the client's implementation. The Virtex can be loaded from FLASH and overwritten with software. The reference package includes the Virtex load utilities, PLL programming software, and I/O loop-back tests.

Theory of Operation

The Spartan3 FPGA implements the PCI interface for the PMC-XM. Data is transferred to/from the PCI bus using single-word accesses for control/status or through the four scatter-gather DMA engines (two in and two out) for accessing the two I/O channels, each with a 4K x 32-bit transmit FIFO and a 4K x 32-bit receive FIFO.

Data transfer state-machines control the bidirectional bursting of data between the Spartan3 and the Virtex for the two I/O channels. The data is transferred across a 32-bit bidirectional data bus and Virtex control/status registers are addressed by an eight-bit address bus. The transfers are independently enabled from the Channel Control Registers in the Spartan3. In the Virtex ATP design used by Dynamic Engineering to test the PMC-XM hardware, there are also four corresponding 4K x 32-bit FIFOs to buffer the bursted data. Handshaking signals generated by the Virtex let the transfer state-machine know when to burst data and, when the FIFOs are near their limits, to move only single words. Two additional handshaking signals have been added between the Spartan3 and Virtex designs since the last version. These are the Virtex transmit FIFO full signals from each of the two channels. If you have an existing Virtex design, make sure that you add these output signals to that design. The previously existing Virtex status output bits two and three have been reused for these signals. See the Virtex reference design for the details.

A programmable PLL supplies two independent clock frequencies (maximum 200 MHz) to be used by the user. Digital clock managers (DCMs) in the Virtex FPGA can be used to further enhance the clock capabilities. An optional 1Mx36-bit QDDRII RAM is accessible by the Virtex for intermediate processing of I/O data and an optional 13-bit digital temperature sensor can be used to read the ambient temperature of the PMC-XM environment.

Scatter-gather DMA is accomplished by writing a list of memory descriptors to host memory. Each descriptor consists of three long-words: the physical address of a block of contiguous user memory, the length of that block and a pointer to the next list entry. The last word of each descriptor also contains two flag-bits that are replaced with zeros for the actual memory accesses. The LSB (Bit 0) is the end-of-chain bit. When this bit

is set, the current descriptor is the last in the list. The next bit (Bit 1) is the direction bit. When this bit is set, it indicates that the transfer is from the module to host memory. When this bit is zero, data is transferred from host memory to the PMC-XM.

The address of the first list entry is written to the DMA engine to begin DMA processing. The DMA continues until the list is complete and an interrupt is signaled to clean-up the transfer and potentially begin another. It is necessary that all memory pages that are to be accessed be physically resident in memory while the DMA is in progress.
The four DMA engines can all operate simultaneously. PCI bus access is arbitrated on a round-robin basis with a DMA engine relinquishing the bus at the end of each list entry transfer or when the corresponding FIFO gets close to full for the transmit or close to empty for the receive. The arbiter can also be configured to give priority to a channel that is approaching the FIFO limit (almost-empty for the transmit or almost-full for the receive).

Programming

Programming the PMC-XM requires only the ability to read and write data from the host. The base address is determined during system configuration of the PCI bus. The base address refers to the first user address for the slot in which the PMC is installed. The VendorId = 0x10EE. The CardId = 0x0024. Current revision = 0x08

Depending on the software environment it may be necessary to set-up the system software with the PMC-XM "registration" data. For example in WindowsNT there is a system registry, which is used to identify the resident hardware.

To use DMA it will be necessary to acquire a block of non-paged memory that is accessible from the PCI bus in which to store chaining descriptor list entries.

At Dynamic Engineering the PMC-XM-DIFF is tested in a Windows environment and we use the Dynamic Engineering Drivers to do the hardware accesses and manage the DMA's. We use MS Visual C++ in conjunction with the drivers to write our test software. Please consider purchasing the engineering kit for the PMC-XM; the software kit includes the drivers and our test suite.

The Spartan3 address space begins at address offset 0, the Virtex address space begins at offset 0x400.

Address Map Spartan3

Register Name	Offset	Description
XM_BASE	0x0000	// Base control register
XM_BASE_USER_SWITCH	0x0004	// User switch/Xilinx rev. read port
XM_BASE_STATUS	0x0008	// Interrupt status port
XM_CHAN0_CNTRL	0x0010	// Channel 0 Control register offset
XM_CHAN0_STATUS	0x0014	// Channel 0 Status read/latch clear port offset
XM_CHAN0_WR_DMA_PNTR	0x0018	// Channel 0 Write DMA physical address register
XM_CHAN0_RD_DMA_PNTR	0x001C	// Channel 0 Read DMA physical address register
XM_CHAN0_FIFO	0x0020	// Channel 0 FIFO offset for single word access
XM_CHAN0_TX_AMT_LVL	0x0024	// Channel 0 TX almost empty level register offset
XM_CHAN0_RX_AFL_LVL	0x0028	// Channel 0 RX almost full level register offset
XM_CHAN0_TX_FIFO_COUNT	0x002C	// Channel 0 TX FIFO count read port offset
XM_CHAN0_RX_FIFO_COUNT	0x0030	// Channel 0 RX FIFO count read port offset
XM_CHAN1_CNTRL	0x0040	// Channel 1 Control register offset
XM_CHAN1_STATUS	0x0044	// Channel 1 Status read/latch clear port offset
XM_CHAN1_WR_DMA_PNTR	0x0048	// Channel 1 Write DMA physical address register
XM_CHAN1_RD_DMA_PNTR	0x004C	// Channel 1 Read DMA physical address register
XM_CHAN1_FIFO	0x0050	// Channel 1 FIFO offset for single word access
XM_CHAN1_TX_AMT_LVL	0x0054	// Channel 1 TX almost empty level register offset
XM_CHAN1_RX_AFL_LVL	0x0058	// Channel 1 RX almost full level register offset
XM_CHAN1_TX_FIFO_COUNT	0x005C	// Channel 1 TX FIFO count read port offset
XM_CHAN1_RX_FIFO_COUNT	0x0060	// Channel 1 RX FIFO count read port offset

FIGURE 2 PMC-XM SPARTAN3 XILINX ADDRESS MAP

The address map provided is for the local decoding performed within the PMC-XM Spartan3 Xilinx. The addresses are all offsets from a base address. The base address and interrupt level are provided by the host in which the PMC-XM is installed.

The host system will search the PCI bus to find the assets installed during power-on initialization. The VendorId = 0x10EE and the CardId = 0x0024 for the PMC-XM. Interrupts are requested by the configuration space. PCIView and other third party utilities can be useful to see how your system is configured. Dynamic Engineering recommends using the Dynamic Engineering Drivers to take care of initialization and device registration.

Register Definitions

XM_BASE

[0x0000] Base Control Register (read/write)

Base Control Register

Data Bit	Description
31-17	Spare
16	Virtex File-Load Enable
15	Virtex Access State-Machine Reset
14-11	Spare
10	Virtex Interrupt Enable
9	Virtex Init
8	Virtex Reset
7	Virtex Flash Enable
6	Slave Serial Mode Enable
5	Virtex Program Init
4	Virtex Program Select
3	Flash Select
2	Flash Control
1	Force Interrupt
0	Master Interrupt Enable

FIGURE 3 PMC-XM SPARTAN3 BASE CONTROL REGISTER

All bits are active high and default to ‘0’ on reset or power-up.

Master Interrupt Enable: This bit enables the interrupts for the base portion of the XM design. When this bit is a ‘1’, the interrupt is enabled; and when this bit is a ‘0’ the interrupt is disabled. Currently the only interrupt source for this portion of the design is the Force Interrupt bit.

Force Interrupt: When this bit is ‘1’ and the Master Interrupt Enable is ‘1’, an interrupt will be generated. This bit is useful for software development and debugging.

Flash Control: When this bit is ‘1’, the Flash Select bit controls which Flash Prom is connected to the JTAG port. When this bit is ‘0’, the TRST# JTAG signal controls the selection. When TRST is grounded, the Virtex Flash is selected; when TRST is open, the signal is pulled high and the Spartan3 Flash is selected.

Flash Select: When Flash Control is set to ‘1’ this bit controls which Flash Prom is connected to the JTAG port. When Flash Select is ‘0’, the Virtex Flash is selected; when Flash Select is ‘1’, the Spartan3 Flash is selected. When Flash Control is ‘0’, this bit has no effect.

Virtex Program Select: When this bit is ‘1’, the Virtex Flash is controlled by the Virtex Flash Enable bit. When this bit is ‘0’, the Virtex Flash is controlled by the Virtex done bit.

Virtex Program Init: When this bit is set to ‘1’ it forces the Virtex to re-configure from the Flash Prom. When this bit is ‘0’, the Virtex can be re-configured by a bit-file load.

Slave Serial Mode Enable: When this bit is set to ‘1’, slave serial programming mode is selected on the Virtex. When this bit is ‘0’ master serial mode is selected. Slave serial mode is used when the Virtex is programmed from a file by the Spartan3 and master serial mode is used when the Virtex configures from the on-board flash.

Virtex Flash Enable: When this bit is ‘0’ and the Virtex Program Select bit is ‘1’, the Virtex flash is disabled so that the Spartan3 can program the Virtex from a bit-file.

Virtex Reset: When this bit is ‘1’, all the registers and FIFOs in the Virtex are reset. When this bit is ‘0’, the Virtex can resume normal operation.

Virtex Init: When set to ‘1’, this bit delays configuration when a configuration cycle has been initiated. When this bit transitions to ‘0’, the mode bits are sampled and the configuration can proceed. The bit then becomes a status bit, which is read from the Status register, a ‘0’ indicating a CRC error.

Virtex Interrupt Enable: This bit enables the Virtex interrupt for the base portion of the XM design. When this bit is a ‘1’, the Virtex interrupt is enabled; and when this bit is a ‘0’ the Virtex interrupt is disabled at the base level.

Virtex Access State-Machine Reset: When a ‘1’ is written to this bit, it causes all the Virtex access state-machines to be reset. This bit is not stored, so it is not necessary to write it back to zero to restore normal operation.

Virtex File-Load Enable: when set to ‘1’, begins the process of programming the Virtex device from a bit-file. The data must be read from the file and loaded into the TX0 FIFO. When the hardware detects that the load is complete this bit will be automatically cleared.

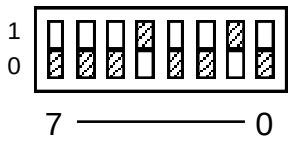
XM_BASE_USER_SWITCH

[0x0004] User Switch Port (read only)

Dip-Switch Port	
Data Bit	**Description**
31-16	Spare
15-8	Xilinx Design Revision Number
7-0	Sw7-0

FIGURE 4 PMC-XM SPARTAN3 USER SWITCH PORT

Sw7-0: The user switch is read through this read-only port. The bits are read as the lowest byte. Access the port as a long word and mask off the undefined bits. The dip-switch positions are defined in the silkscreen. For example the switch figure below indicates a 0x12.



Xilinx design revision number: The value of the second byte of this port is the rev. number of the Xilinx design (currently 0x08 - rev. H).

XM_BASE_STATUS
[0x0008] Status Register Read / Latch Clear Write

Status Register

Data Bit	Description
31	Interrupt Status
30-22	Spare
21	Virtex Status 1
20	Virtex Status 0
19-10	Spare
9	Virtex Init Status
8	Virtex Configuration Done
7	Virtex Interrupt 1 Active
6	Virtex Interrupt 0 Active
5	Channel 1 Interrupt Active
4	Channel 0 Interrupt Active
3-1	Spare
0	Local Interrupt Active

FIGURE 5 PMC-XM SPARTAN3 STATUS REGISTER

Local Interrupt Active: When read as a '1', a local interrupt condition is active. Currently, the only such condition is the Force Interrupt bit in the Base Control Register. A system interrupt will not occur unless the Master Interrupt Enable bit in the Base Control Register is also set. When read as a '0', no local interrupt conditions are active.

Channel 0/1 Interrupt Active: When read as a '1', the corresponding channel interrupt is active. When read as a '0', that interrupt is inactive.

Virtex Interrupt 0/1 Active: When read as a '1', the corresponding Virtex channel interrupt is active. When read as a '0', that interrupt is inactive.

Virtex Configuration Done: When read as a '1', the Virtex FPGA has successfully configured. When read as a '0', the Virtex configuration was not successful.

Virtex Init Status: When read as a '1' after the Virtex configuration, it indicates that a CRC error did not occur during the Virtex configuration. When read as a '0' after the Virtex configuration, it indicates that a CRC error occurred during the previous Virtex configuration. In this case the done bit should also be low.

Virtex Status 1-0: These bits are driven by the Virtex to indicate arbitrary status conditions. In the current Virtex ATP design they are both low, but they can be assigned for any purpose desired.

Interrupt Status: When read as a ‘1’, an enabled interrupt condition is active and a system interrupt should be asserted provided the master interrupt enable is set. When read as a ‘0’, no enabled interrupt condition is active.

XM_CHAN0/1_CNTRL

[0x0010, 0x0040] Channel Control Register (read/write]

Control Register

Data Bit	Description
31-14	Spare
13	RX FIFO Almost Full Interrupt Enable
12	TX FIFO Almost Empty Interrupt Enable
11	DMA Read Arbitration Priority Enable
10	DMA Write Arbitration Priority Enable
9	Virtex Interrupt Enable
8	Receive Enable
7	Transmit Enable
6	Force Interrupt
5	Master Interrupt Enable
4	DMA Read Enable
3	DMA Write Enable
2	FIFO Bypass
1	RX FIFO Reset
0	TX FIFO Reset

FIGURE 6 PMC-XM SPARTAN3 CHANNEL CONTROL REGISTER

TX/RX FIFO Reset: When one of these bits is ‘1’, the transmit/receive FIFO for the referenced channel is placed in a reset condition. When these bits are ‘0’, the FIFOs are in a normal operational state.

FIFO Bypass: When this bit is ‘1’, any data written to the transmit FIFO will be transferred to the receive FIFO as long as there is room in the receive FIFO. This facilitates FIFO loop-back testing. When this bit is ‘0’, data written to the transmit FIFO will remain in the FIFO until read by the data transfer state machine.

DMA Read/Write Enable: When one of these bits is ‘1’, the read/write DMA interrupt is enabled for the referenced channel. When one of these bits is ‘0’, that DMA interrupt is disabled.

Master Interrupt Enable: This bit enables the local interrupts for the referenced channel. When this bit is a ‘1’, the channel interrupts are enabled; and when this bit is a ‘0’ the interrupts are disabled.

Force Interrupt: When this bit is ‘1’ and the Master Enable is a ‘1’, a system interrupt will occur. This bit is useful for software development and debugging.

Transmit Enable: When this bit is ‘1’, the transfer state-machine is enabled to move data from the referenced channel transmit FIFO to the corresponding Virtex channel transmit FIFO. When this bit is ‘0’, the transmit transfer state machine is disabled.

Receive Enable: When this bit is ‘1’, the transfer state machine is enabled to move data from the referenced Virtex channel receive FIFO to the corresponding local receive FIFO. When this bit is ‘0’, the receive transfer state machine is disabled.

Virtex Interrupt Enable: When this bit is ‘1’, the corresponding Virtex interrupt (VINT0 for channel 0 or VINT1 for channel 1) is enabled to drive the channel interrupt. When this bit is ‘0’, the Virtex interrupt cannot cause a channel interrupt.

DMA Write Arbitration Priority Enable: When this bit is ‘1’, the write DMA for the referenced channel will receive priority if the TX FIFO has become almost empty as defined by the value stored in the TX_AMT_LVL register. When this bit is ‘0’, the DMA arbitration will follow round-robin arbitration priority.

DMA Read Arbitration Priority Enable: When this bit is ‘1’, the read DMA for the referenced channel will receive priority if the RX FIFO has become almost full as defined by the value stored in the RX_AFL_LVL register. When this bit is ‘0’, the DMA arbitration will follow round-robin arbitration priority.

TX FIFO Almost Empty Interrupt Enable: When this bit is set to a one, an interrupt will be generated when the transmit FIFO level is equal or less than the value specified in the TX_AMT_LVL register, provided the channel master interrupt enable is asserted. When this bit is zero, an interrupt will not be generated, but the status can still be read from the channel status register.

RX FIFO Almost Full Interrupt Enable: When this bit is set to a one, an interrupt will be generated when the receive FIFO level is equal or greater than the value specified in the RX_AFL_LVL register, provided the channel master interrupt enable is asserted. When this bit is zero, an interrupt will not be generated, but the status can still be read from the channel status register.

XM_CHAN0/1_STATUS

[0x0014, 0x0044] Channel Status Read / Latch Clear Write

Status Register

Data Bit	Description
31	Interrupt Active Status
30-18	Spare
17	Virtex Interrupt Active
16	Local Interrupt Active
15	Read DMA Interrupt Active
14	Write DMA Interrupt Active
13	Read DMA Error
12	Write DMA Error
11-10	Spare
9	Receive FIFO Almost Full (latched)
8	Transmit FIFO Almost Empty (latched)
7	Receive FIFO Valid
6	Receive FIFO Full
5	Receive FIFO Almost Full
4	Receive FIFO Empty
3	Transmit FIFO Valid
2	Transmit FIFO Full
1	Transmit FIFO Almost Empty
0	Transmit FIFO Empty

FIGURE 7 PMC-XM SPARTAN3 CHANNEL STATUS REGISTER

Transmit FIFO Empty: When read as a ‘1’, the corresponding transmit FIFO is empty. When read as a ‘0’, the FIFO has at least one word in it.

Transmit FIFO Almost Empty: When read as a ‘1’, the corresponding transmit FIFO is almost empty as determined by the value entered in the almost empty level register. When read as a ‘0’, there is more data in the FIFO than specified in the level register.

Transmit FIFO Full: When read as a ‘1’, the corresponding transmit FIFO is full. When read as a ‘0’, there is room for at least one more word in the FIFO.

Receive FIFO empty: When read as a ‘1’, the corresponding receive FIFO is empty. When read as a ‘0’, the FIFO has at least one word in it.

Receive FIFO Almost Full: When read as a ‘1’, the corresponding receive FIFO is almost full as determined by the value entered in the almost full level register. When read as a ‘0’, there is less data in the FIFO than specified in the level register.

Receive FIFO Full: When read as a ‘1’, the corresponding receive FIFO is full. When read as a ‘0’, there is room for at least one more word in the FIFO.

Receive FIFO Valid: When read as a ‘1’, there is valid receive data to read. When read as a ‘0’, there is no valid receive data. There is a four-deep pipeline on the output of the RX FIFO that will be filled before data is retained in the FIFO. Therefore even though the FIFO is empty there may actually be up to four long-words of valid receive data. This status bit indicates when there is valid data even though the FIFO is empty.

Transmit FIFO Almost Empty (latched): When a one is read, it indicates that the transmit FIFO data count has become less than or equal to the value in the TX_AMT_LVL register. A zero indicates that the FIFO has not become almost empty since this bit was last cleared. This bit is latched and can be cleared by writing to the Status register with a one in this bit position.

Receive FIFO Almost Full (latched): When a one is read, it indicates that the receive FIFO data count has become greater than or equal to the value in the RX_AFL_LVL register. A zero indicates that the FIFO has not become almost full since this bit was last cleared. This bit is latched and can be cleared by writing to the Status register with a one in this bit position.

Write DMA Error: When read as a ‘1’, 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. When read as a ‘0’, no error has occurred.

Read DMA Error: When read as a ‘1’, 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. When read as a ‘0’, no error has occurred.

Write DMA Interrupt Active: When read as a ‘1’, 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. When read as a ‘0’, no write DMA interrupt is pending.

Read DMA Interrupt Active: When read as a ‘1’, 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. When read as a ‘0’, no read DMA interrupt is pending

Local Interrupt Active: When read as a ‘1’, a local interrupt condition is active for the referenced channel. A channel interrupt will not occur unless the Master Interrupt Enable bit in the Channel Control Register is also set. When read as a ‘0’, no local interrupt conditions are active.

Virtex Interrupt Active: When read as a ‘1’, the corresponding Virtex interrupt (VINT0 for channel 0 or VINT1 for channel 1) is active. A system interrupt will not occur unless the Virtex Interrupt Enable in the Channel Control Register is set. When read as a ‘0’, the Virtex interrupt is inactive.

Interrupt Active Status: When read as a ‘1’, an enabled channel interrupt condition is active and a channel interrupt is asserted. When read as a ‘0’, no enabled channel interrupt is active.

XM_CHAN0/1_WR/RD_DMA_PNTR

[0x0018, 0x001C, 0x0048, 0x004C] DMA Address Register (Write only)

DMA Pointer Address Register	
Data Bit	**Description**
31-0	First Chaining Descriptor Physical Address

FIGURE 8 PMC-XM SPARTAN3 CHANNEL DMA POINTER REGISTER

These write-only ports are used to initiate scatter-gather DMAs. When the physical address of the first chaining descriptor is written to one of these ports, the corresponding DMA engine reads three successive long words beginning at that address. The first is the address of the first memory block of the DMA buffer, 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 a bit in one of the next pointer values read indicates that it is the end of the chain.

Note: Writing a zero to one of these ports will abort the associated DMA if one is in progress.

XM_CHAN0/1_FIFO

[0x0020, 0x0050] Write TX/Read RX FIFO Port

TX / RX FIFO Port	
Data Bit	**Description**
31-0	FIFO Data 31-0

FIGURE 9 PMC-XM SPARTAN3 CHANNEL FIFO PORT

Data written to this address is written into the transmit FIFO as long as the FIFO is not full. When this address is read a data-word is read from the receive FIFO. When the receive FIFO becomes empty, the last data-word that was in the FIFO will be returned.

XM_CHAN0/1_TX_AMT_LVL

[0x0024, 0x0054] TX Almost Empty Level Register (read/write)

TX Almost Empty Level Register	
Data Bit	**Description**
31-16	Spare
15-0	TX FIFO Almost Empty Level

FIGURE 10 PMC-XM SPARTAN3 CHANNEL TX ALMOST EMPTY REGISTER

This register specifies the level at which the transmit FIFO almost empty level will be asserted. When the number of data words in the transmit FIFO is less than or equal to this count the almost empty status will be asserted.

XM_CHAN0/1_RX_AFL_LVL

[0x0028, 0x0058] RX Almost Full Level Register (read/write)

RX Almost Full Level Register	
Data Bit	**Description**
31-16	Spare
15-0	RX FIFO Almost Full Level

FIGURE 11 PMC-XM SPARTAN3 CHANNEL RX ALMOST FULL REGISTER

This register specifies the level at which the receive FIFO almost full level will be asserted. When the number of data words in the receive FIFO is greater than or equal to this count the almost full status will be asserted.

XM_CHAN0/1_TX/RX_FIFO_COUNT

[0x002C, 0x0030, 0x005C, 0x0060] TX/RX FIFO Data Count Port (read only)

FIFO Data Count

Data Bit	Description
31-16	Spare
15-0	FIFO Data Words Stored

FIGURE 12 PMC-XM SPARTAN3 CHANNEL TX/RX FIFO COUNT PORT

These read-only register ports report the number of 32-bit data words in the corresponding transmit/receive FIFO and data pipeline (currently a maximum of 0x1001 for the transmit and 0x1004 for the receive).

Address Map: Virtex ATP Design

Register Name	Offset	Description
ATP_VTX_BASE	0x0400	// Base control register
ATP_VTX_STATUS	0x0404	// Interrupt status/clear port
ATP_VTX_IO_TERM	0x0410	// I/O termination control
ATP_VTX_IO_DIR	0x0414	// I/O direction control
ATP_VTX_IO_DOUT	0x0418	// I/O output data register
ATP_VTX_IO_UPPER	0x041C	// I/O term/dir/dout/din upper two bits
ATP_VTX_IO_DIN	0x0420	// I/O input data read port
ATP_VTX_CHAN0_CNTRL	0x0450	// Channel 0 Control register offset
ATP_VTX_CHAN0_STATUS	0x0454	// Channel 0 Status read/latch clear port offset
ATP_VTX_TX0_FIFO	0x0458	// Channel 0 TX FIFO offset for single word access
ATP_VTX_RX0_FIFO	0x045C	// Channel 0 RX FIFO offset for single word access
ATP_VTX_TX0_DCOUNT	0x0460	// Channel 0 TX FIFO count read port offset
ATP_VTX_RX0_DCOUNT	0x0464	// Channel 0 RX FIFO count read port offset
ATP_VTX_TX0_AMT_CNT	0x0468	// Channel 0 TX FIFO almost empty count offset
ATP_VTX_RX0_AFL_CNT	0x046C	// Channel 0 RX FIFO almost full count offset
ATP_VTX_CHAN1_CNTRL	0x04A0	// Channel 1 Control register offset
ATP_VTX_CHAN1_STATUS	0x04A4	// Channel 1 Status read/latch clear port offset
ATP_VTX_TX1_FIFO	0x04A8	// Channel 1 TX FIFO offset for single word access
ATP_VTX_RX1_FIFO	0x04AC	// Channel 1 RX FIFO offset for single word access
ATP_VTX_TX1_DCOUNT	0x04B0	// Channel 1 TX FIFO count read port offset
ATP_VTX_RX1_DCOUNT	0x04B4	// Channel 1 RX FIFO count read port offset
ATP_VTX_TX1_AMT_CNT	0x04B8	// Channel 1 TX FIFO almost empty count offset
ATP_VTX_RX1_AFL_CNT	0x04BC	// Channel 1 RX FIFO almost full count offset

FIGURE 13 PMC-XM VIRTEX (ATP) XILINX ADDRESS MAP

This address map is only valid for the ATP design supplied by Dynamic Engineering. The addresses are offset from the PCI address assigned to the card by the system PCI configuration utility.

Register Definitions

ATP_VTX_BASE

[0x0400] Base Control Register (read/write)

Base Control Register

Data Bit	Description
31–20	Spare
19	PLL SDAT Output
18	PLL S2/Suspend
17	PLL SCLK
16	PLL Enable
15–14	Spare
13–10	I/O Mux Select
9	I/O Enable
8	Reset DCM
7	Force Interrupt 1
6	Master Interrupt 1 Enable
5	Force Interrupt 0
4	Master Interrupt 0 Enable
3–0	LED 4–1

FIGURE 14 PMC-XM VIRTEX (ATP) BASE CONTROL REGISTER

LED 4–1: When one of these bits is set to a ‘1’, the corresponding LED will be lit. When the bit is a ‘0’, the LED will not be lit.

Master Interrupt 0/1 Enable: When this bit is ‘1’, the corresponding interrupt is enabled (VINT0 or VINT1). When this bit is ‘0’, the interrupt is disabled.

Force Interrupt 0/1: When this bit is ‘1’, and the corresponding interrupt enable is set, that interrupt will be asserted from the Virtex.

Reset DCM: When this bit is ‘1’, the DCM (Digital Clock Manager) will be manually reset. When this bit is ‘0’, the DCM will operate normally.

I/O Enable: When this bit is ‘1’, the I/O control and data lines will be controlled by I/O Mux Select and the channel I/O state-machines. When this bit is ‘0’, the I/O control and data lines will be controlled by the I/O registers described below (see figures 16-18).

I/O Mux Select: Each channel has two byte-wide input ports, two byte-wide output ports, an input clock line and an output clock line. When I/O Enable is asserted, this four-bit field controls the connections between these internal ports and the external I/O lines. See the section on I/O testing following the register definitions.

PLL Enable: When this bit is '1', the PLL interface circuit is enabled for reading or programming the PLL. When this bit is '0', the PLL interface circuit is disabled.

PLL SCLK: This bit is used to clock data into and out of the PLL.

PLL S2/Suspend: This bit is used to select alternative pre-programmed clock frequencies from the PLL. It is normally set to '0'.

PLL SDAT Output: This is where the PLL data state is specified when data is being written to the PLL. When the PLL is driving the data line this bit must be set to a '1'.

The PMC-XM has a PLL device which is programmed over an I²C bus to produce the desired frequencies.

The data line has a pull-up on the board. When the PLL is enabled and the I²C data bit is set to '0' in this register, the external line is driven low. When not enabled or when the I²C data bit is set to '1' in this register, the external line is tri-stated and pulled-up by the resistor. For a read operation the data should be set to '1' to allow the PLL to drive the data line.

The clock line for the PLL to be programmed is toggled along with the data to create a bit stream with a "software clock". Set the bit to the next state and toggle the clock line and repeat.

The upper selection bit can be set in the register and directly driven to the PLL. This allows the selection of alternative pre-programmed clock frequencies.

To read over the I²C bus a command is first written and then the bus read for the response. The I²C data input bit in the status register contains the state of the bus when read. The software will toggle the clock line and when the low-to-high transition is made, read the data bit then repeat until the entire message is captured.

The engineering kit contains the logic and software required to program the PLL and to read-back the internal register programming. The software to determine the frequency command words is available from Cypress Semiconductor. The PLL part number is CY22393FC. Cypress has a utility available for calculating the frequency command words for the PLL. http://www.dyneng.com/CyberClocks.zip is the URL for the Cypress software used to calculate the PLL programming words. The reference frequency is 40 MHz.

ATP_VTX_STATUS

[0x0404] Status Register (read only)

Base Status Register

Data Bit	Description
31-20	Spare
19	PLL SDAT Input
18	Intstat1
17	Intstat0
16	DCM Locked
15-8	Design ID
7-0	Design Rev.

FIGURE 15 PMC-XM VIRTEX (ATP) BASE STATUS REGISTER

Design ID/Rev.: These fields are read to determine which design and revision is programmed into the Virtex. This is used to determine the control/status register configuration and which driver to use to communicate with the design.

DCM Locked: When read as a '1', it indicates that the DCM is in a locked state and the clocks produced are functioning reliably. When read as a '0', it indicates that the DCM is not locked and therefore the clocking is not reliable.

Intstat0/1: When read as a '1', it indicates that the corresponding interrupt is active. When read as a '0', the interrupt is not active.

PLL SDAT Input: This is where the PLL data line is read when data is being read from the PLL. This line is used to read the register contents of the PLL.

ATP_VTX_IO_TERM, ATP_VTX_IO_DIR, ATP_VTX_IO_DOUT

[0x0410, 0x0414, 0x0418] I/O Register (read/write)

I/O Register	
Data Bit	**Description**
31-0	Control/Data

FIGURE 16 PMC-XM VIRTEX (ATP) I/O REGISTER

These three registers control the termination, direction and data output values for the I/O lines (31-0) on the PMC-XM when I/O enable is not asserted. Each bit controls these parameters for their respective I/O line. Terminations are active when the corresponding control bit is a '1' and inactive when the corresponding control bit is a '0'. An I/O line is an output when the corresponding direction control bit is a '1' and an input when the corresponding control bit is a '0'. The last two I/O lines (IO32, IO33) are controlled by the ATP_VTX_IO_UPPER register described below.

ATP_VTX_IO_UPPER

[0x041C] Status Register (read/write)

I/O Register	
Data Bit	**Description**
31-18	Spare
17-16	Data In 33-32
15-10	Spare
9-8	Termination 33-32
7-6	Spare
5-4	Direction 33-32
3-2	Spare
1-0	Data Out 33-32

FIGURE 17 PMC-XM VIRTEX (ATP) I/O UPPER REGISTER

The data output value, direction, termination and data input value for IO32 and IO33 are accessed through this register when I/O enable is not asserted. Each bit controls these parameters for their respective I/O line.

ATP_VTX_IO_DIN

[0x0420] I/O Register (read only)

I/O Register	
Data Bit	**Description**
31-0	Control/Data

FIGURE 18 PMC-XM VIRTEX (ATP) I/O DIN REGISTER

The value of the I/O input data lines is read from this register. Each bit represents the real-time data value present on its respective I/O line.

ATP_VTX_CHAN0/1_CNTRL

[0x0450, 0x04A0] Channel Control Register (read/write)

Channel Control Register	
Data Bit	**Description**
31-16	Transmit Byte Count
15-11	Spare
10	I/O Input select
9	I/O Output select
8	Receive Interrupt Enable
7	Transmit Interrupt Enable
6	Receive I/O Enable
5	Transmit I/O Enable
4	Receive FIFO Reset
3	Transmit FIFO Reset
2	Force Interrupt
1	Master Interrupt Enable
0	FIFO Bypass

FIGURE 19 PMC-XM VIRTEX (ATP) CHANNEL CONTROL REGISTER

FIFO Bypass: When this bit is ‘1’, any data written to the transmit FIFO will be transferred to the receive FIFO as long as there is room in the FIFO. This facilitates FIFO loop-back testing. When this bit is ‘0’, data written to the transmit FIFO will remain in the FIFO until explicitly read.

Master Interrupt Enable: When this bit is ‘1’, the corresponding interrupt is enabled (VINT0 for channel 0 or VINT1 for channel 1). When this bit is ‘0’, the interrupt is disabled. The base control interrupt enable bit must also be set to cause an interrupt.

Force Interrupt: When this bit is ‘1’, and the corresponding interrupt enables are set, that interrupt will be asserted from the Virtex. This bit has a parallel function with the force interrupt bits in the base control register.

Transmit/Receive FIFO Reset: Writing a ‘1’ to one of these bits will reset the corresponding FIFO. The value is not stored, so it isn’t necessary to write back a ‘0’ afterwards.

Transmit/Receive I/O Enable: When one of these bits is a ‘1’, the corresponding I/O function is enabled. The transmitter will read data from the transmit FIFO and send it out a byte at a time along with a gated clock derived from the PLL clock A for channel 0 or PLL clock B for channel 1. The receiver looks for a clock and stores byte-wide data and writes four bytes at a time to the receive FIFO.

Transmit/Receive Interrupt Enable: When one of these bits is a ‘1’, the corresponding I/O interrupt is enabled. The transmit interrupt occurs when the requested byte-count has been reached. The receive interrupt occurs when no I/O clocks are seen for four clock periods after a reception has commenced.

I/O Output select: When this bit is ‘1’, the ‘B’ output port of the referenced channel is selected to drive the I/O bus. When this bit is ‘0’, the ‘An’ output port of the referenced channel is selected to drive the I/O bus.

I/O Input select: When this bit is ‘1’, the ‘B’ input port of the referenced channel is selected to receive data from the I/O bus. When this bit is ‘0’, the ‘A’ input port of the referenced channel is selected to receive data from the I/O bus.

Transmit Byte Count: This 16-bit field specifies the number of bytes to send in a single transmission. The maximum number of bytes to send is 65,535.

ATP_VTX_CHAN0/1_STATUS

[0x0454, 0x04A4] Channel Status Register (read only)

Channel Status Register

Data Bit	Description
31-16	Spare
15	Channel Interrupt Asserted
14-12	Spare
11	Local Interrupt Condition Active
10	Receive Done Interrupt
9	Transmit Done Interrupt
8	Transmit Data Valid
7	Receive FIFO Full
6	Receive FIFO Almost Full
5	Receive FIFO Almost Empty
4	Receive FIFO Empty
3	Transmit FIFO Full
2	Transmit FIFO Almost Full
1	Transmit FIFO Almost Empty
0	Transmit FIFO Empty

FIGURE 20 PMC-XM VIRTEX (ATP) CHANNEL STATUS REGISTER

Transmit FIFO Empty: When read as a ‘1’, the corresponding transmit FIFO is empty. When read as a ‘0’, the FIFO has at least one word in it.

Transmit FIFO Almost Empty: When read as a ‘1’, the corresponding transmit FIFO is almost empty as determined by the almost empty field in the control register. When read as a ‘0’, there is more data in the FIFO than specified in the control register.

Transmit FIFO Almost Full: When read as a ‘1’, the corresponding transmit FIFO is almost full. The almost full level is hard-coded to 4078 words. When read as a ‘0’, there are less than this number of words in the FIFO. This status is also used to regulate the data transfer from the Spartan3.

Transmit FIFO Full: When read as a ‘1’, the corresponding transmit FIFO is full. When read as a ‘0’, there is room for at least one more word in the FIFO. This status is also used to regulate the data transfer from the Spartan3.

Receive FIFO Empty: When read as a ‘1’, the corresponding receive FIFO is empty. When read as a ‘0’, the FIFO has at least one word in it. This status is also used to regulate the data transfer from the Spartan3.

Receive FIFO Almost Empty: When read as a ‘1’, the corresponding receive FIFO is almost empty. The almost empty level is hard-coded to 18 words. When read as a ‘0’, there are more than this number of words in the FIFO. This status is also used to regulate the data transfer from the Spartan3.

Receive FIFO Almost Full: When read as a ‘1’, the corresponding receive FIFO is almost full as determined by the almost full field in the control register. When read as a ‘0’, there is less data in the FIFO than specified in the control register.

Receive FIFO Full: When read as a ‘1’, the corresponding receive FIFO is full. When read as a ‘0’, there is room for at least one more word in the FIFO.

Transmit Data Valid: When read as a ‘1’, the corresponding transmit FIFO has valid data at its output. This may be true even if the transmit empty flag is set if the last word has not been acquired by the I/O transmitter or the FIFO bypass state-machine.

Transmit Done Interrupt: When read as a ‘1’, it indicates that an I/O transmission has completed. This occurs when the requested byte-count has been reached. When read as a ‘0’, no requested transmission has completed since the status was last read.

Receive Done Interrupt: When read as a ‘1’, it indicates that an I/O reception has completed. This occurs when at least one byte has been received and then at least four I/O clock periods elapse without any data being received. When read as a ‘0’, either a reception has not begun, or it is still in progress since the status was last read.

Local Interrupt Condition Active: When read as a ‘1’, an enabled local interrupt condition is active for the corresponding interrupt (VINT0 for channel 0 or VINT1 for channel 1). When read as a ‘0’, the interrupt condition is not active.

Channel Interrupt Asserted: When read as a ‘1’, the corresponding channel interrupt is asserted (VINT0 for channel 0 or VINT1 for channel 1). When read as a ‘0’, the interrupt is not asserted.

ATP_VTX_TX0/1_FIFO

[0x0458, 0x04A8] TX FIFO Port (read/write)

TX FIFO Port	
Data Bit	**Description**
31-0	FIFO Data 31-0

FIGURE 21 PMC-XM VIRTEX (ATP) CHANNEL TX FIFO PORT

Data written to this address is written into the transmit FIFO as long as the FIFO is not full. When this address is read a data-word is read from the transmit FIFO. When the FIFO becomes empty, the last data-word that was in the FIFO will be returned.

ATP_VTX_RX0/1_FIFO

[0x045C, 0x04AC] RX FIFO Port (read/write)

RX FIFO Port	
Data Bit	**Description**
31-0	FIFO Data 31-0

FIGURE 22 PMC-XM VIRTEX (ATP) CHANNEL RX FIFO PORT

Data written to this address is written into the receive FIFO as long as the FIFO is not full. When this address is read a data-word is read from the receive FIFO. When the FIFO becomes empty, the last data-word that was in the FIFO will be returned.

ATP_VTX_TX0/1_DCOUNT

[0x0460, 0x04B0] TX FIFO Data Count Port (read only)

TX FIFO Data Count	
Data Bit	**Description**
31-12	Spare
11-0	FIFO Data Words Stored

FIGURE 23 PMC-XM VIRTEX (ATP) CHANNEL TX FIFO COUNT PORT

These read-only register ports report the number of 32-bit data-words in the corresponding transmit FIFO (currently a maximum of 4095 (0xFFF)).

ATP_VTX_RX0/1_DCOUNT

[0x0464, 0x04B4] RX FIFO Data Count Port (read only)

RX FIFO Data Count	
Data Bit	**Description**
31-12	Spare
11-0	FIFO Data Words Stored

FIGURE 24 PMC-XM VIRTEX (ATP) CHANNEL RX FIFO COUNT PORT

These read-only register ports report the number of 32-bit data words in the corresponding receive FIFO (currently a maximum of 4095 (0xFFF)).

ATP_VTX_TX0/1_AMT_CNT

[0x0464, 0x04B4] RX FIFO Data Count Port (read/write)

TX FIFO Almost Empty Count	
Data Bit	**Description**
31-12	Spare
11-0	Almost-Empty Threshold for Transmit FIFO

FIGURE 25 PMC-XM VIRTEX (ATP) TX FIFO ALMOST EMPTY PORT

This register specifies the number of transmit FIFO data-words used as an almost-empty threshold for the Transmit FIFO Almost Empty bit in the channel status register. If the number of data-words in the transmit FIFO is above this value, the almost-empty flag will be ‘0’, otherwise the flag will be ‘1’.

ATP_VTX_RX0/1_AFL_CNT

[0x0464, 0x04B4] RX FIFO Data Count Port (read/write)

RX FIFO Almost Full Count	
Data Bit	**Description**
31-12	Spare
11-0	FIFO Data Words Stored

FIGURE 26 PMC-XM VIRTEX (ATP) RX FIFO ALMOST FULL PORT

This register specifies the number of receive FIFO data-words used as an almost-full threshold for the Receive FIFO Almost Full bit in the channel status register. If the number of data-words in the receive FIFO is below this value, the almost-full flag will be '0', otherwise the flag will be '1'.

I/O Data Loopback Testing

Our test-fixture connects I/O (7-0) to I/O (15-8) and I/O (23-16) to I/O (31-24).
Also I/O (32) is connected to I/O (33).

Channel 0-output port ‘A’ drives I/O lines (7-0), when selected.
Channel 1-output port ‘A’ drives I/O lines (15-8), when selected.
Channel 0-output port ‘B’ drives I/O lines (23-16), when selected.
Channel 1-output port ‘B’ drives I/O lines (31-24), when selected.

The following table lists the I/O routing configurations for each I/O Mux Select value. Set the appropriate I/O enables and I/O input and output selects for each Mux Select value to accomplish the first or second data-transfer.

Mux	First Transfer	Second Transfer	Clock
0x0	=> chan0-A => chan0-A,	chan0-B => chan0-B,	clk0 IO32 => IO33 clk0
0x1	=> chan0-A => chan0-A,	chan0-B => chan0-B,	clk0 IO33 => IO32 clk0
0x2	=> chan1-A => chan1-A,	chan1-B => chan1-B,	clk1 IO32 => IO33 clk1
0x3	=> chan1-A => chan1-A,	chan1-B => chan1-B,	clk1 IO33 => IO32 clk1
0x4	=> chan0-A => chan0-B,	chan0-B => chan0-A,	clk0 IO32 => IO33 clk0
0x5	=> chan0-A => chan0-B,	chan0-B => chan0-A,	clk0 IO33 => IO32 clk0
0x6	=> chan1-A => chan1-B,	chan1-B => chan1-A,	clk1 IO32 => IO33 clk1
0x7	=> chan1-A => chan1-B,	chan1-B => chan1-A,	clk1 IO33 => IO32 clk1
0x8	=> chan0-A => chan1-A,	chan0-B => chan1-B,	clk0 IO32 => IO33 clk1
0x9	=> chan0-A => chan1-A,	chan0-B => chan1-B,	clk0 IO33 => IO32 clk1
0xA	=> chan1-A => chan0-A,	chan1-B => chan0-B,	clk1 IO32 => IO33 clk0
0xB	=> chan1-A => chan0-A,	chan1-B => chan0-B,	clk1 IO33 => IO32 clk0
0xC	=> chan0-A => chan1-B,	chan0-B => chan1-A,	clk0 IO32 => IO33 clk1
0xD	=> chan0-A => chan1-B,	chan0-B => chan1-A,	clk0 IO33 => IO32 clk1
0xE	=> chan1-A => chan0-B,	chan1-B => chan0-A,	clk1 IO32 => IO33 clk0
0xF	=> chan1-A => chan0-B,	chan1-B => chan0-A,	clk1 IO33 => IO32 clk0

Virtex Pin Out

The Virtex FPGA pin definitions are contained in the engineering kit and repeated here as a reference. The hardwired pins for power and ground are not shown.

Signal Name	Pin	Direction	I/O Standard
OSC	C13	Input	LVCMOS 3.3 V
VCLK66	A16	Input	LVCMOS 3.3 V
CLK66FB	C10	Output	LVCMOS 3.3 V
VD<0>	A3	Bidir	LVCMOS 3.3 V
VD<1>	B3	Bidir	LVCMOS 3.3 V
VD<2>	A4	Bidir	LVCMOS 3.3 V
VD<3>	B4	Bidir	LVCMOS 3.3 V
VD<4>	A5	Bidir	LVCMOS 3.3 V
VD<5>	B6	Bidir	LVCMOS 3.3 V
VD<6>	A6	Bidir	LVCMOS 3.3 V
VD<7>	B7	Bidir	LVCMOS 3.3 V
VD<8>	A7	Bidir	LVCMOS 3.3 V
VD<9>	B9	Bidir	LVCMOS 3.3 V
VD<10>	A8	Bidir	LVCMOS 3.3 V
VD<11>	B10	Bidir	LVCMOS 3.3 V
VD<12>	A9	Bidir	LVCMOS 3.3 V
VD<13>	B12	Bidir	LVCMOS 3.3 V
VD<14>	A10	Bidir	LVCMOS 3.3 V
VD<15>	B13	Bidir	LVCMOS 3.3 V
VD<16>	A11	Bidir	LVCMOS 3.3 V
VD<17>	B14	Bidir	LVCMOS 3.3 V
VD<18>	A12	Bidir	LVCMOS 3.3 V
VD<19>	B15	Bidir	LVCMOS 3.3 V
VD<20>	A15	Bidir	LVCMOS 3.3 V
VD<21>	B17	Bidir	LVCMOS 3.3 V
VD<22>	A17	Bidir	LVCMOS 3.3 V
VD<23>	B18	Bidir	LVCMOS 3.3 V
VD<24>	A18	Bidir	LVCMOS 3.3 V
VD<25>	B20	Bidir	LVCMOS 3.3 V
VD<26>	A19	Bidir	LVCMOS 3.3 V
VD<27>	B21	Bidir	LVCMOS 3.3 V
VD<28>	A20	Bidir	LVCMOS 3.3 V
VD<29>	B23	Bidir	LVCMOS 3.3 V
VD<30>	A21	Bidir	LVCMOS 3.3 V
VD<31>	B24	Bidir	LVCMOS 3.3 V

VADD<0>	C1	Input	LVCMOS 3.3 V
VADD<1>	C2	Input	LVCMOS 3.3 V
VADD<2>	C4	Input	LVCMOS 3.3 V
VADD<3>	C5	Input	LVCMOS 3.3 V
VADD<4>	C6	Input	LVCMOS 3.3 V
VADD<5>	C7	Input	LVCMOS 3.3 V
VADD<6>	C8	Input	LVCMOS 3.3 V
VADD<7>	D15	Input	LVCMOS 3.3 V
V_W	D3	Input	LVCMOS 3.3 V
V_R	D4	Input	LVCMOS 3.3 V
VACK	D5	Output	LVCMOS 3.3 V
VRST	D8	Input	LVCMOS 3.3 V
VDMA_W0	C11	Input	LVCMOS 3.3 V
VDMA_W1	C12	Input	LVCMOS 3.3 V
VDMA_R0	D13	Input	LVCMOS 3.3 V
VDMA_R1	C14	Input	LVCMOS 3.3 V
VDMA_RDY_W0	C17	Output	LVCMOS 3.3 V
VDMA_RDY_W1	C19	Output	LVCMOS 3.3 V
VDMA_RDY_R0	C15	Output	LVCMOS 3.3 V
VDMA_RDY_R1	C16	Output	LVCMOS 3.3 V
VDMA_FL_W0	D11	Output	LVCMOS 3.3 V
VDMA_FL_W1	D12	Output	LVCMOS 3.3 V
VDMA_MT_R0	C20	Output	LVCMOS 3.3 V
VDMA_MT_R1	C21	Output	LVCMOS 3.3 V
VINT0	D6	Output	LVCMOS 3.3 V
VINT1	D7	Output	LVCMOS 3.3 V
VSTAT<0>	D9	Output	LVCMOS 3.3 V
VSTAT<1>	D10	Output	LVCMOS 3.3 V
VSPARE<0>	D16	Input	LVCMOS 3.3 V
VSPARE<1>	D17	Input	LVCMOS 3.3 V
VSPARE<2>	D18	Input	LVCMOS 3.3 V
VSPARE<3>	D19	Input	LVCMOS 3.3 V
VSPARE<4>	D20	Input	LVCMOS 3.3 V
VSPARE<5>	D21	Input	LVCMOS 3.3 V
VSPARE<6>	D22	Input	LVCMOS 3.3 V
VSPARE<7>	D23	Input	LVCMOS 3.3 V
VSPARE<8>	D24	Input	LVCMOS 3.3 V

LED<0>	AE18	Bidir	LVCMOS 3.3 V
LED<1>	AF18	Bidir	LVCMOS 3.3 V
LED<2>	AC19	Bidir	LVCMOS 3.3 V
LED<3>	AF19	Bidir	LVCMOS 3.3 V
TERM<0>	AF20	Output	LVCMOS 3.3 V
TERM<1>	AB20	Output	LVCMOS 3.3 V
TERM<2>	W21	Output	LVCMOS 3.3 V
TERM<3>	J25	Output	LVCMOS 3.3 V
TERM<4>	L21	Output	LVCMOS 3.3 V
TERM<5>	M21	Output	LVCMOS 3.3 V
TERM<6>	K24	Output	LVCMOS 3.3 V
TERM<7>	L24	Output	LVCMOS 3.3 V
TERM<8>	M24	Output	LVCMOS 3.3 V
TERM<9>	N24	Output	LVCMOS 3.3 V
TERM<10>	P24	Output	LVCMOS 3.3 V
TERM<11>	J23	Output	LVCMOS 3.3 V
TERM<12>	K23	Output	LVCMOS 3.3 V
TERM<13>	L23	Output	LVCMOS 3.3 V
TERM<14>	M23	Output	LVCMOS 3.3 V
TERM<15>	N23	Output	LVCMOS 3.3 V
TERM<16>	P23	Output	LVCMOS 3.3 V
TERM<17>	V23	Output	LVCMOS 3.3 V
TERM<18>	J22	Output	LVCMOS 3.3 V
TERM<19>	K22	Output	LVCMOS 3.3 V
TERM<20>	M22	Output	LVCMOS 3.3 V
TERM<21>	N22	Output	LVCMOS 3.3 V
TERM<22>	P22	Output	LVCMOS 3.3 V
TERM<23>	J21	Output	LVCMOS 3.3 V
TERM<24>	K21	Output	LVCMOS 3.3 V
TERM<25>	V25	Output	LVCMOS 3.3 V
TERM<26>	V26	Output	LVCMOS 3.3 V
TERM<27>	R23	Output	LVCMOS 3.3 V
TERM<28>	R24	Output	LVCMOS 3.3 V
TERM<29>	R25	Output	LVCMOS 3.3 V
TERM<30>	U22	Output	LVCMOS 3.3 V
TERM<31>	U23	Output	LVCMOS 3.3 V
TERM<32>	U24	Output	LVCMOS 3.3 V
TERM<33>	U25	Output	LVCMOS 3.3 V

DIR<0>	W26	Output	LVCMOS 3.3 V
DIR<1>	AA26	Output	LVCMOS 3.3 V
DIR<2>	AD26	Output	LVCMOS 3.3 V
DIR<3>	W25	Output	LVCMOS 3.3 V
DIR<4>	Y25	Output	LVCMOS 3.3 V
DIR<5>	AB25	Output	LVCMOS 3.3 V
DIR<6>	AC25	Output	LVCMOS 3.3 V
DIR<7>	AD25	Output	LVCMOS 3.3 V
DIR<8>	W24	Output	LVCMOS 3.3 V
DIR<9>	Y24	Output	LVCMOS 3.3 V
DIR<10>	AA24	Output	LVCMOS 3.3 V
DIR<11>	AB24	Output	LVCMOS 3.3 V
DIR<12>	AC24	Output	LVCMOS 3.3 V
DIR<13>	AE24	Output	LVCMOS 3.3 V
DIR<14>	AF24	Output	LVCMOS 3.3 V
DIR<15>	W23	Output	LVCMOS 3.3 V
DIR<16>	AA23	Output	LVCMOS 3.3 V
DIR<17>	AB23	Output	LVCMOS 3.3 V
DIR<18>	AC23	Output	LVCMOS 3.3 V
DIR<19>	AD23	Output	LVCMOS 3.3 V
DIR<20>	AE23	Output	LVCMOS 3.3 V
DIR<21>	AF23	Output	LVCMOS 3.3 V
DIR<22>	V22	Output	LVCMOS 3.3 V
DIR<23>	W22	Output	LVCMOS 3.3 V
DIR<24>	Y22	Output	LVCMOS 3.3 V
DIR<25>	AB22	Output	LVCMOS 3.3 V
DIR<26>	AC22	Output	LVCMOS 3.3 V
DIR<27>	AD22	Output	LVCMOS 3.3 V
DIR<28>	AF22	Output	LVCMOS 3.3 V
DIR<29>	V21	Output	LVCMOS 3.3 V
DIR<30>	F18	Output	LVCMOS 3.3 V
DIR<31>	F19	Output	LVCMOS 3.3 V
DIR<32>	G19	Output	LVCMOS 3.3 V
DIR<33>	E18	Output	LVCMOS 3.3 V
PLL_REF	G1	Output	LVCMOS 3.3 V
PLL_SCLK	F1	Output	LVCMOS 3.3 V
PLL_S2	G2	Output	LVCMOS 3.3 V
PLL_SDAT	E1	Bidir	LVCMOS 3.3 V
PLL_CLKA	D2	Input	LVCMOS 3.3 V
PLL_CLKB	D1	Input	LVCMOS 3.3 V

IO<0>	E21	Bidir	LVCMOS 3.3 V
IO<1>	G23	Bidir	LVCMOS 3.3 V
IO<2>	G21	Bidir	LVCMOS 3.3 V
IO<3>	A22	Bidir	LVCMOS 3.3 V
IO<4>	C22	Bidir	LVCMOS 3.3 V
IO<5>	E22	Bidir	LVCMOS 3.3 V
IO<6>	G22	Bidir	LVCMOS 3.3 V
IO<7>	H22	Bidir	LVCMOS 3.3 V
IO<8>	A23	Bidir	LVCMOS 3.3 V
IO<9>	C23	Bidir	LVCMOS 3.3 V
IO<10>	E23	Bidir	LVCMOS 3.3 V
IO<11>	F23	Bidir	LVCMOS 3.3 V
IO<12>	A24	Bidir	LVCMOS 3.3 V
IO<13>	C24	Bidir	LVCMOS 3.3 V
IO<14>	E24	Bidir	LVCMOS 3.3 V
IO<15>	F24	Bidir	LVCMOS 3.3 V
IO<16>	C25	Bidir	LVCMOS 3.3 V
IO<17>	D25	Bidir	LVCMOS 3.3 V
IO<18>	E25	Bidir	LVCMOS 3.3 V
IO<19>	G25	Bidir	LVCMOS 3.3 V
IO<20>	H25	Bidir	LVCMOS 3.3 V
IO<21>	C26	Bidir	LVCMOS 3.3 V
IO<22>	D26	Bidir	LVCMOS 3.3 V
IO<23>	E26	Bidir	LVCMOS 3.3 V
IO<24>	F26	Bidir	LVCMOS 3.3 V
IO<25>	G26	Bidir	LVCMOS 3.3 V
IO<26>	H23	Bidir	LVCMOS 3.3 V
IO<27>	G24	Bidir	LVCMOS 3.3 V
IO<28>	H24	Bidir	LVCMOS 3.3 V
IO<29>	H26	Bidir	LVCMOS 3.3 V
IO<30>	H21	Bidir	LVCMOS 3.3 V
IO<31>	F20	Bidir	LVCMOS 3.3 V
IO<32>	G20	Bidir	LVCMOS 3.3 V
IO<33>	H20	Bidir	LVCMOS 3.3 V

Pinouts for Optional Temperature Sensor

CS_TEMP	Y23	Output	LVCMOS 3.3 V
SCK_TEMP	Y26	Output	LVCMOS 3.3 V
DIN_TEMP	AB26	Output	LVCMOS 3.3 V
DOUT_TEMP	AC26	Input	LVCMOS 3.3 V

Pinouts for Optional 1Mx36-bit QDDRII SRAM

QDR_CQ	U1	Input	HSTL_I_DCI 1.8 V
QDR_K	P3	Output	HSTL_I 1.8 V
QDR_KN	P2	Output	HSTL_I 1.8 V
QDR_W	L4	Output	HSTL_I 1.8 V
QDR_R	J7	Output	HSTL_I 1.8 V
QDR_IN<0>	AD6	Input	HSTL_I_DCI 1.8 V
QDR_IN<1>	AB5	Input	HSTL_I_DCI 1.8 V
QDR_IN<2>	AA8	Input	HSTL_I_DCI 1.8 V
QDR_IN<3>	Y8	Input	HSTL_I_DCI 1.8 V
QDR_IN<4>	V6	Input	HSTL_I_DCI 1.8 V
QDR_IN<5>	T6	Input	HSTL_I_DCI 1.8 V
QDR_IN<6>	P7	Input	HSTL_I_DCI 1.8 V
QDR_IN<7>	L7	Input	HSTL_I_DCI 1.8 V
QDR_IN<8>	P8	Input	HSTL_I_DCI 1.8 V
QDR_IN<9>	AD3	Input	HSTL_I_DCI 1.8 V
QDR_IN<10>	AC5	Input	HSTL_I_DCI 1.8 V
QDR_IN<11>	Y5	Input	HSTL_I_DCI 1.8 V
QDR_IN<12>	W5	Input	HSTL_I_DCI 1.8 V
QDR_IN<13>	U5	Input	HSTL_I_DCI 1.8 V
QDR_IN<14>	T7	Input	HSTL_I_DCI 1.8 V
QDR_IN<15>	R7	Input	HSTL_I_DCI 1.8 V
QDR_IN<16>	K5	Input	HSTL_I_DCI 1.8 V
QDR_IN<17>	M7	Input	HSTL_I_DCI 1.8 V
QDR_IN<18>	K2	Input	HSTL_I_DCI 1.8 V
QDR_IN<19>	M4	Input	HSTL_I_DCI 1.8 V
QDR_IN<20>	M1	Input	HSTL_I_DCI 1.8 V
QDR_IN<21>	P4	Input	HSTL_I_DCI 1.8 V
QDR_IN<22>	T3	Input	HSTL_I_DCI 1.8 V
QDR_IN<23>	U3	Input	HSTL_I_DCI 1.8 V
QDR_IN<24>	V2	Input	HSTL_I_DCI 1.8 V
QDR_IN<25>	AA3	Input	HSTL_I_DCI 1.8 V
QDR_IN<26>	AC1	Input	HSTL_I_DCI 1.8 V
QDR_IN<27>	L8	Input	HSTL_I_DCI 1.8 V
QDR_IN<28>	J2	Input	HSTL_I_DCI 1.8 V
QDR_IN<29>	R1	Input	HSTL_I_DCI 1.8 V
QDR_IN<30>	R8	Input	HSTL_I_DCI 1.8 V
QDR_IN<31>	U4	Input	HSTL_I_DCI 1.8 V
QDR_IN<32>	V1	Input	HSTL_I_DCI 1.8 V
QDR_IN<33>	T8	Input	HSTL_I_DCI 1.8 V
QDR_IN<34>	W4	Input	HSTL_I_DCI 1.8 V
QDR_IN<35>	AB1	Input	HSTL_I_DCI 1.8 V

QDR_OUT<0>	AD4	Output	HSTL_I 1.8 V
QDR_OUT<1>	AD5	Output	HSTL_I 1.8 V
QDR_OUT<2>	AB6	Output	HSTL_I 1.8 V
QDR_OUT<3>	Y6	Output	HSTL_I 1.8 V
QDR_OUT<4>	V7	Output	HSTL_I 1.8 V
QDR_OUT<5>	U7	Output	HSTL_I 1.8 V
QDR_OUT<6>	P6	Output	HSTL_I 1.8 V
QDR_OUT<7>	N5	Output	HSTL_I 1.8 V
QDR_OUT<8>	N7	Output	HSTL_I 1.8 V
QDR_OUT<9>	AC6	Output	HSTL_I 1.8 V
QDR_OUT<10>	Y4	Output	HSTL_I 1.8 V
QDR_OUT<11>	W6	Output	HSTL_I 1.8 V
QDR_OUT<12>	V5	Output	HSTL_I 1.8 V
QDR_OUT<13>	U6	Output	HSTL_I 1.8 V
QDR_OUT<14>	P5	Output	HSTL_I 1.8 V
QDR_OUT<15>	N4	Output	HSTL_I 1.8 V
QDR_OUT<16>	L6	Output	HSTL_I 1.8 V
QDR_OUT<17>	J6	Output	HSTL_I 1.8 V
QDR_OUT<18>	M2	Output	HSTL_I 1.8 V
QDR_OUT<19>	L3	Output	HSTL_I 1.8 V
QDR_OUT<20>	M8	Output	HSTL_I 1.8 V
QDR_OUT<21>	R4	Output	HSTL_I 1.8 V
QDR_OUT<22>	R2	Output	HSTL_I 1.8 V
QDR_OUT<23>	T4	Output	HSTL_I 1.8 V
QDR_OUT<24>	Y3	Output	HSTL_I 1.8 V
QDR_OUT<25>	W2	Output	HSTL_I 1.8 V
QDR_OUT<26>	Y2	Output	HSTL_I 1.8 V
QDR_OUT<27>	K1	Output	HSTL_I 1.8 V
QDR_OUT<28>	L1	Output	HSTL_I 1.8 V
QDR_OUT<29>	N3	Output	HSTL_I 1.8 V
QDR_OUT<30>	T1	Output	HSTL_I 1.8 V
QDR_OUT<31>	V4	Output	HSTL_I 1.8 V
QDR_OUT<32>	W1	Output	HSTL_I 1.8 V
QDR_OUT<33>	Y1	Output	HSTL_I 1.8 V
QDR_OUT<34>	AA1	Output	HSTL_I 1.8 V
QDR_OUT<35>	AD1	Output	HSTL_I 1.8 V
QDR_BWN<0>	K7	Output	HSTL_I 1.8 V
QDR_BWN<1>	K4	Output	HSTL_I 1.8 V
QDR_BWN<2>	K6	Output	HSTL_I 1.8 V
QDR_BWN<3>	N2	Output	HSTL_I 1.8 V

QDR_ADDR<0>	M6	Output	HSTL_I 1.8 V
QDR_ADDR<1>	J4	Output	HSTL_I 1.8 V
QDR_ADDR<2>	J5	Output	HSTL_I 1.8 V
QDR_ADDR<3>	K3	Output	HSTL_I 1.8 V
QDR_ADDR<4>	M5	Output	HSTL_I 1.8 V
QDR_ADDR<5>	AA4	Output	HSTL_I 1.8 V
QDR_ADDR<6>	AC2	Output	HSTL_I 1.8 V
QDR_ADDR<7>	AC3	Output	HSTL_I 1.8 V
QDR_ADDR<8>	AB3	Output	HSTL_I 1.8 V
QDR_ADDR<9>	AB4	Output	HSTL_I 1.8 V
QDR_ADDR<10>	AE4	Output	HSTL_I 1.8 V
QDR_ADDR<11>	AC4	Output	HSTL_I 1.8 V
QDR_ADDR<12>	AE3	Output	HSTL_I 1.8 V
QDR_ADDR<13>	AD2	Output	HSTL_I 1.8 V
QDR_ADDR<14>	AF3	Output	HSTL_I 1.8 V
QDR_ADDR<15>	AE6	Output	HSTL_I 1.8 V
QDR_ADDR<16>	AF4	Output	HSTL_I 1.8 V
QDR_ADDR<17>	AF6	Output	HSTL_I 1.8 V
RD_STB_IN	AF8	Input	HSTL_I_DCI 1.8 V
RD_STB_OUT	AF7	Output	HSTL_I 1.8 V

I/O Standard Acronyms:
LVCMOS – Low Voltage Complementary Metal Oxide Semiconductor
HSTL_I – High-speed Transceiver Logic Class I
DCI – Digitally Controlled Impedance

The pin names match with the schematic names and the names found throughout this manual. The engineering kit contains a reference project with the pin numbers defined and the bus interfaces implemented. A lot of time will be saved on the first implementation by starting with the reference design.

PMC PCI Pn1 Interface Pin Assignment

The figure below gives the pin assignments for the PMC Module PCI Pn1 Interface on the PMC-XM. See the User Manual for your carrier board for more information. Unused pins may be assigned by the specification and not needed by this design.

	-12V(unused)	1	2
GND	INTA#	3	4
		5	6
BUSMODE1#	+5V	7	8
		9	10
GND -		11	12
CLK	GND	13	14
GND -		15	16
	+5V	17	18
	AD31	19	20
AD28-	AD27	21	22
AD25-	GND	23	24
GND -	C/BE3#	25	26
AD22-	AD21	27	28
AD19	+5V	29	30
	AD17	31	32
FRAME#-	GND	33	34
GND	IRDY#	35	36
DEVSEL#	+5V	37	38
GND	LOCK#	39	40
		41	42
PAR	GND	43	44
	AD15	45	46
AD12-	AD11	47	48
AD9-	+5V	49	50
GND -	C/BE0#	51	52
AD6-	AD5	53	54
AD4	GND	55	56
	AD3	57	58
AD2-	AD1	59	60
	+5V	61	62
GND		63	64

FIGURE 27 PMC-XM PN1 INTERFACE

PMC PCI Pn2 Interface Pin Assignment

The figure below gives the pin assignments for the PMC Module PCI Pn2 Interface on the PMC-XM. See the User Manual for your carrier board for more information. Unused pins may be assigned by the specification and not needed by this design.

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

FIGURE 28 PMC-XM PN2 INTERFACE

Front Panel I/O Pin Assignment

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

IO_0P	IO_0N	1	35
IO_1P	IO_1N	2	36
IO_2P	IO_2N	3	37
IO_3P	IO_3N	4	38
IO_4P	IO_4N	5	39
IO_5P	IO_5N	6	40
IO_6P	IO_6N	7	41
IO_7P	IO_7N	8	42
IO_8P	IO_8N	9	43
IO_9P	IO_9N	10	44
IO_10P	IO_10N	11	45
IO_11P	IO_11N	12	46
IO_12P	IO_12N	13	47
IO_13P	IO_13N	14	48
IO_14P	IO_14N	15	49
IO_15P	IO_15N	16	50
IO_16P	IO_16N	17	51
IO_17P	IO_17N	18	52
IO_18P	IO_18N	19	53
IO_19P	IO_19N	20	54
IO_20P	IO_20N	21	55
IO_21P	IO_21N	22	56
IO_22P	IO_22N	23	57
IO_23P	IO_23N	24	58
IO_24P	IO_24N	25	59
IO_25P	IO_25N	26	60
IO_26P	IO_26N	27	61
IO_27P	IO_27N	28	62
IO_28P	IO_28N	29	63
IO_29P	IO_29N	30	64
IO_30P	IO_30N	31	65
IO_31P	IO_31N	32	66
IO_32P	IO_32N	33	67
IO_33P	IO_33N	34	68

FIGURE 29 PMC-XM FRONT PANEL INTERFACE

PMC Module Backplane IO Interface Pin Assignment

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

PIO_0P	PIO_1P	1	2
PIO_0N	PIO_1N	3	4
PIO_2P	PIO_3P	5	6
PIO_2N	PIO_3N	7	8
PIO_4P	PIO_5P	9	10
PIO_4N	PIO_5N	11	12
PIO_6P	PIO_7P	13	14
PIO_6N	PIO_7N	15	16
PIO_8P	PIO_9P	17	18
PIO_8N	PIO_9N	19	20
PIO_10P	PIO_11P	21	22
PIO_10N	PIO_11N	23	24
PIO_12P	PIO_13P	25	26
PIO_12N	PIO_13N	27	28
PIO_14P	PIO_15P	29	30
PIO_14N	PIO_15N	31	32
PIO_16P	PIO_17P	33	34
PIO_16N	PIO_17N	35	36
PIO_18P	PIO_19P	37	38
PIO_18N	PIO_19N	39	40
PIO_20P	PIO_21P	41	42
PIO_20N	PIO_21N	43	44
PIO_22P	PIO_23P	45	46
PIO_22N	PIO_23N	47	48
PIO_24P	PIO_25P	49	50
PIO_24N	PIO_25N	51	52
PIO_26P	PIO_27P	53	54
PIO_26N	PIO_27N	55	56
PIO_28P	PIO_29P	57	58
PIO_28N	PIO_29N	59	60
PIO_30P	PIO_31P	61	62
PIO_30N	PIO_31N	63	64

FIGURE 30 PMC-XM PN4 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 PMC-XM. 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 work-station.

Start-up
Make sure that the "system" can see your hardware before trying to access it. Many BI/OS 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 BI/OS then a third party PCI device cataloging tool will be helpful. We use PCIView.

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

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

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

The PMC is secured against the carrier with four screws attached to the 2 stand-offs and 2 locations on the front panel. The four screws provide significant protection against shock, vibration, and incomplete insertion.

The PMC Module provides a low temperature coefficient of 2.17 W/°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 PMC. The coefficient means that if 2.17 Watts are applied uniformly on the component side, then the temperature difference between the component side and solder side is one degree Celsius.

Thermal Considerations

The PMC-XM 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 AUTHORIZATI/ON (RMA) number. Carefully package the unit, in the original shipping carton if this is available, and ship prepaid and insured with the RMA number clearly written on the outside of the package. Include a return address and the telephone number of a technical contact. For out-of-warranty repairs, a purchase order for repair charges must accompany the return. Dynamic Engineering will not be responsible for damages due to improper packaging of returned items. For service on Dynamic Engineering Products not purchased directly from Dynamic Engineering contact your reseller. Products returned to Dynamic Engineering for repair by other than the original customer will be treated as out-of-warranty.

Out of Warranty Repairs

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

For Service Contact:

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

Specifications

Host Interface:	33 MHz/32-bit PCI Mezzanine Card
Access types:	Configuration and Memory space utilized
Clock rates supported:	33 MHz PMC, 33 MHz data transfer between Spartan3 and Virtex Local 40 MHz oscillator for PLL reference to provide two programmable frequencies
Memory	FIFO memory is provided to support DMA Four 4K x 32-bit FIFOs in Spartan3 and four 4K x 32-bit FIFOs in Virtex (depends on the Virtex design)
Software Interface:	Control/Status Registers within Spartan3 and Virtex
Initialization:	Hardware reset forces all registers to zero except as noted
Access Modes:	All registers on long-word boundary - Standard target accesses read and write to registers and memory - DMA access to memory
Access Time:	No wait states in DMA modes, one wait state in target access to Spartan3 Virtex accesses are user defined
Interrupt:	One interrupt to the PCI bus is supported with multiple sources. The interrupts are maskable and are supported with status registers.
Onboard Options:	All Options are Software Programmable.
Dimensions:	Standard Single PMC Module
Construction:	FR4 Multi-Layer Printed Circuit, Surface Mount Components
Power:	5V and 3.3V from PCI bus. Local 2.5V, 1.8V and 1.2V created with on-board power supplies
User	8 position software readable switch 4 software controllable LEDs 2 Power LEDs

Order Information

PMC-XM	http://www.dyneng.com/pmc_xm.html Standard version with two 16KB FIFOs per channel, standard XM timing and protocol.
PMC-XM-Eng-1	Engineering Kit for the PMC-XM Board-level schematics (PDF) and Sample Virtex design (VHDL)
PMC-XM-Eng-2	Engineering Kit for the PMC-XM Board-level schematics [PDF], Sample Virtex Design (VHDL), Software Drivers and Sample Test Application