IOP 480 Data Book

Order Number: IOP 480-SIL-DB-P1-1.0

Printed in the USA, October 1999

IOP 480 Data Book

Version 1.0

October 1999

Website: http://www.plxtech.com
Email: apps@plxtech.com
Phone: 408 774-9060
800 759-3735
Fax: 408 774-2169

Contents

Figures

Tables

Timing Diagrams

Registers

PREFACE

The information contained in this document should be considered preliminary. Although an effort has been made to keep the information accurate, there may be misleading or even incorrect statements made herein. The document is being written in parallel with actual chip development and, as such, it is subject to change. This description is intended to be a living document, to be updated throughout the IOP 480 I/O processor design effort. It provides a broad technical overview of the IOP 480 I/O processor.

SUPPLEMENTAL DOCUMENTATION

The following is a list of additional documentation to provide the reader with further information about the IOP 480 microprocessor and related subjects:

- *PCI Local Bus Specification, Revision 2.2*
  PCI Special Interest Group (PCI SIG)
  2575 NE Kathryn #17, Hillsboro, OR 97124 USA
  Tel: 800 433-5177 (domestic only) or 503 693-6232, Fax: 503 693-8344, http://www.pcisig.com
- *PCI Hot-Plug Specification, Revision 1.0*
  PCI Special Interest Group (PCI SIG)
  2575 NE Kathryn #17, Hillsboro, OR 97124 USA
  Tel: 800 433-5177 (domestic only) or 503 693-6232, Fax: 503 693-8344, http://www.pcisig.com
- *PCI Power Management Interface Specification, Revision 2.2*
  PCI Special Interest Group (PCI SIG)
  2575 NE Kathryn #17, Hillsboro, OR 97124 USA
  Tel: 800 433-5177 (domestic only) or 503 693-6232, Fax: 503 693-8344, http://www.pcisig.com
- *PICMG 2.1 R1.0 CompactPCI® Hot Swap Specification*
  PCI Industrial Computer Manufacturers Group (PICMG)
  c/o Rogers Communications, 401 Edgewater Place, Suite 500, Wakefield, MA 01880, USA
  Tel: 781 246-9318, Fax: 781 224-1239, http://www.picmg.org
- *Intelligent I/O (I_2O) Architecture Specification Rev 1.5*
  I_2O Special Interest Group
  404 Balboa Street, San Francisco, CA, 94118 USA
  Tel: 415 750-8352, Fax: 415 751-4829, http://www.i2osig.org

REVISION HISTORY

Date	Revision	Comment
7/30/1997	0.1	Red Book initial release.
3/6/1998	0.2	Revised Red Book to include detailed core and peripheral specifications.
4/15/1998	0.3	Revised Red Book to separate IBM and PLX material.
7/8/1998	0.31	Revised Red Book to combine IBM and PLX materials and incorporate engineering changes.
12/7/1998	0.90	Blue Book initial release. Incorporate Red Book engineering changes.

Date	Revision	Comment

1/27/1999 – 2/12/1999	0.92	Spacing problem addressed. Replaced TrueType fonts with PostScript fonts and eliminated nonbreaking spaces and pair kerning.
		Minor text corrections, updates and additional product documentation were addressed in the following sections. (Bulleted items address specific changes.)
		Section 1 "IOP 480 Integrated PowerPC I/O Processor"
		• Figure 1-1. IOP 480 Block Diagram, redrawn (UART and Serial EEPROM Controller)
		• 1.2.2.1 "Advanced Data Pipe Architecture"
		  • Local-to-Local DMA definition changed
		  • Discussion of Big/Little Endian Conversion, Mailbox Registers and Doorbell Registers eliminated
		• 1.2.8 "Direct Data Transfer Modes," added a direct data transfer mode
		• 1.4.3 "High-Performance PCI I_2O Design," mention of incorporation of the "Pull and Outbound" option
		Section 2 "Local Bus Interface"
		• 2.2.2 "Bus Regions," reduced seven bus regions to six.
		• TD 2-1. Big Endian Cycle, corrected AD[31:0], LCLK, ADS#, ALE, and LAD[31:0].
		• 2.4 "Local Signals," moved to become 2.3.
		Section 3 "PCI Bus Interface"
		• 3.5 "PCI Bus Protocol," simple protocol added.
		• TD 3-1. Direct Slave Write to 8-Bit Local Bus with One External (READY#) Wait State, inserted.
		• TD 3-2. Direct Slave PCI Write of 15 Lwords to 32-Bit Local Bus, Local Burst Enabled, One External (READY#) Wait State, Bterm Enabled, inserted.
		Section 4 "Direct Slave Operation"
		• Figure 4-4. Direct Slave Write, removed redundant LAD between IOP 480 and Local Bus.
		• TD 4-2. Direct Slave Read from 32-Bit Local Bus with Zero Wait States, corrected FRAME#, IRDY#, and TRDY#.
		• TD 4-7. Direct Slave Read from 32-Bit Local Bus with Zero Three Internal (WAIT#) Wait States, corrected LCLK.
		Section 5 "Direct Master Operation"
		• TD 5-6. Direct Master Burst Read of 12 Lwords with WAIT#, revised.
		• TD 5-7. Direct Master Configuration Write Type 0, revised.
		• TD 5-9. Direct Master PCI Dual Address Cycle, revised.
		• TD 5-10. Direct Master Writeback Cycle, deleted.
		Section 6 "IOP 480 CPU Bus Interface"
		6.4 "Accessing the PCI Bus," and 6.5 "Accessing the Local Bus," switched.

Date	Revision	Comment

1/27/1999 – 2/12/1999	0.92	**Section 7 "DMA Operation"**
		• Figures 7-4 and 7-5. Rearranged lines coming from "Local or Host Memory" block. Added Initialize DMA descriptors in either Host or Local Memory, and in 7.5, added second "PCI Address High" block.
		• Figures 7-8 and 7-9. DMA Data Transfer Operation, replaced BREQi with LHOLDREQ.
		• 7.1.9 "Local Bus EOT," added text on ending all transfers.
		• 7.1.16 "Local Bus Arbitration," replaced BOFF# with LHOLDREQ.
		• 7.1.17 "PCI Bus Arbitration," inserted and 7.1.18 and 7.1.19 incremented.
		• 7.2.8 "Local Bus Arbitration," BOFF# replaced with LHOLDREQ.
		• TD 7-9. DMA2 Unaligned Transfer, corrected LAD[31:0].
		Section 8 "Local Bus Internal Arbiter"
		• 8.2 "Initialization" and 8.4 "High-Priority Mode," BREQ replaced with BOFF#.
		• 8.5 "Performance Tuning," BOFF# added.
		Section 9 "PCI Bus Internal Arbiter"
		Section 10 "Reset and Initialization"
		• Table 10-1. Adapter and Host Modes Resets moved after Table 10-2.
		• Table 10-3. Serial EEPROM Load Registers, replaced table and decremented to Table 10-2.
		• 10.4.3 "Blank EEPROM Programming, added."
		• 10.5 "Internal Register Access," moved to Section 17 "Register Summary," now numbered 17.1.
		• 10.6 "IOP 480 CPU Boot" decremented to 10.5, with all following level 1 headings decremented.
		• Table 10-4. Contents of Registers After Reset, LE field, Big Endian replaced with Little Endian disabled (Big Endian).
		• 10.8 "IOP 480 Initialization from Local Bus moved to 10.4.1.
		• 10.9 "IOP 480 Initialization from PCI" moved to 10.7.1.
		Section 11 "Interrupts"
		11.2 "PCI Interrupts," numbering changed from 11.2.1.2 onward due to extra heading.
		Section 12 "Memory Controller"
		• Table 12-11. Refresh Arbitration, replaced.
		  • "De-assert LHOLDACK0" with "Assert BOFF#".
		  • "De-assert LHOLDACK1" with "None".
		• TD 12-10. SDRAM Initialization, corrected CMD (Internal).
		• TD 12-13. SDRAM Self-Refresh End, corrected CMD (Internal).
		• Table 12-19. Refresh Arbitration replaced.
		  • "De-assert LHOLDACK0" with "Assert BOFF#".
		  • "De-assert LHOLDACK1" with "None".
		• TD captions augmented with additional Wait State information.
		Section 14 "Compact Hot Swap"
		HS_CNTL, HS_NEXT, and HS_CSR changed to HSCNTL, HSNEXT, and HSCSR.
		Section 15 "Vital Product Data"
		• 15.2 VPD Capabilities Register.
		• "VPD Address" para changed to reflect bit change, referenced VPD_CAP.
		• "VPD Data" para referenced VPD_DATA.
		• TD captions were augmented with additional information.

Date	Revision	Comment

1/27/1999 – 2/12/1999	0.92	**Section 16 "Power Management"**
		Section 17 "Register Summary"
		• 10.5 "Internal Register Access," moved to Section 17 "Register Summary," now numbered 17.1, with all following level 1 headings incremented.
		• Table 17-3. Serial EEPROM Loading Sequence, "Totals" changed.
		• Minor revisions in register descriptions
		  • Register 17-145, C0PLAE changed to C1PLAE, C0PLAF changed to C1PLAF.
		  • Registers 17-148, 17-149, and 17-150, ***Reserved*** deleted.
		Section 21 "Timing Diagrams"
		The following timing diagrams were revised or supplemented:
		• 21-4. Big Endian Cycle, corrected AD[31:0], LCLK, ADS#, ALE, and LAD[31:0].
		• 21-12. Direct Slave Read from 32-Bit Local Bus with Zero Wait States, corrected FRAME#, IRDY#, and TRDY#.
		• 21-16. Direct Slave Read from 32-Bit Local Bus with Three Internal (WAIT#) Wait States, corrected LCLK.
		• 21-33. Direct Master Burst Read of 12 Lwords with WAIT#, corrected REQ#.
		• 21-46. DMA2 Unaligned Transfer, corrected LAD[31:0].
		• 21-71. SDRAM Initialization, corrected CMD (Internal).
		• 21-74. SDRAM Self-Refresh End, corrected CMD (Internal).
		• 21-2. VPD Write replaced with Register to Start VPD Write.
		• 21-3. VPD Read replaced with Register Read to Show Condition of VPD Read.
		• 21-36. Direct Master Dual Address Cycle, revised.
		• 21-37. Direct Master Writeback Cycle, deleted.
		• Memory TD captions augmented with additional Wait State information. Also, in level one headings substituted the word "cycle" with "access," to reflect nomenclature in Section 12 "Memory Controller."

Date	Revision	Comment

7/1999	0.96	**General Revisions to Document**
		• Deleted all references to IOCR register.
		• Changed references to “de-assert” command back to “negate” in IOP 480 CPU sections.
		• Added bleed tabs on section and appendix recto pages.
		• LHOLDACK1 is now LHOLDACK1/BREQ.
		Global to Timing Diagrams
		• Corrected signal names to use standard signal name, rather than internal signal name: “LHOLDACKREQ0” is now “LHOLDACK0/LDREQ,” “LHOLDREQACK0” is now “LHOLDREQ0/LHOLDACK,” “DACK12#” is now “DACK[2:1]#,” and “DREQ12#” is now “DREQ[2:1]#.”
		• Changed “LCLK-66” to “LCLK-60.”
		• Modified names in other sections to more closely correspond to their Section 21 counterparts.
		Section 1 “IOP 480 Integrated PowerPC I/O Processor”
		• Title of Section 1 changed to “Introduction.”
		• 1.2.8 Changed from “Direct Data Transfer Modes” to “Data Transfer Mechanisms.” Content also changed.
		Section 2 “Local Bus Interface”
		• 2.4.3.3 and Table 2-4, applied LHOLDACK1/BREQ changes.
		• 2.4.4 Local Chip Selects, added to document. Table 2-5 “Registers Defining Local Characteristics of all LCS Range Accesses,” added.
		• 2.7.4 “Internal IOP 480 CPU,” amplified with new text and diagram.
		  • Table 2-9 Internal IOP 480 CPU Big Endian Word, deleted.
		  • Table 2-10 IOP 480 CPU Big Endian Byte Ordering,” changed to reflect Internal CPU Bus addresses.
		  • Endian Swapping text and diagram, added.
		  • 2.7.4.2 Table 2-11 Internal IOP 480 CPU Little Endian Word, deleted.,
		  • Table 2-12 IOP 480 CPU Little Endian Byte Ordering,” changed to reflect Internal CPU Bus addresses. (Table numbering now resequenced.)
		• 2.8, inserted Table 2-8 Big Endian/Little Endian Byte Ordering. (Table numbering now resequenced.
		Section 3 “PCI Bus Interface”
		• No Changes.
		Section 4 “Direct Slave Operation”
		• 4.3 “Internal FIFOs,” added DS FIFO description.
		Section 5 “Direct Master Operation”
		• No Changes.
		Section 6 “IOP 480 CPU Bus Interface”
		• No Changes.
		Section 7 “DMA Operation”
		• 7.1.3, revised Figure 7-3 Block DMA Mode Initialization (Single Address or PCI Dual Address Cycle), transfer size reordered.
		• TD 7-8, DMA2 Flyby, changed to DMA2 Non-Flyby
		Section 8 “Local Bus Internal Arbiter”
		• Changed BOFF# to LHOLDACK1/BREQ in 8.2, 8.4, and 8.5.
		• Changed LHOLDREQ0 to LHOLDREQ0/LHOLDACK in 8.1 and TD 8-2.
		Section 9 “PCI Bus Internal Arbiter”
		• No Changes.

Date	Revision	Comment
7/1999	0.96	**Section 10 "Reset and Initialization"** • 10.4.3, revised first para to read that "…the IOP 480 uses the Fairchild 93CS66LEN serial EEPROM." • Table 10-3 Serial EEPROM Load Registers, added four addresses. • 10.4.4, supplemented text and title of section to include IOP 480 Initialization without a serial EEPROM. **Section 11 "Interrupts"** • 11.2.1.1, changed PINTENB[11] to PINTSTAT[11]. • Figure 11-1 INTA# Interrupt and Error Sources, replaced with Table 11-1 PCI Interrupts (INTA#). • Figure 11-4 INTO Interrupt and Error Sources, replaced with Table 11-2 Local Interrupts (INTO). Subsequent table numbering incremented, and figure numbering decremented. • 11.6.4 Deleted "as controlled by the IOCR[CIL} bit" in first para. **Section 12 "Memory Controller"** • All figures, added connection dots and input arrows, as appropriate. • 12.1 Flash memory and parallel serial EEPROMs, added. • 12.2.6 and 12.2.7, changed loading of MA[17:0] from "beginning" to "end." • Figure 12-3, changed 128K to 256K. • Added Figure (now 12-4, "Flash (One 8-Bit Device)"). • TDs 12-1 through 12-4 and 12-6, and 12-7, revised MA[17:0] signal. • TD 12-8, revised MA[17:0] and BLAST# signals. • TD 12-22, added new diagram, "Long SDRAM Burst Write." **Section 13 "Intelligent I/O (I_2O)"** • No changes. **Section 14 "Compact Hot Swap"** • Table 14-1 Reversed bits. **Section 15 "Vital Product Data"** • No changes. **Section 16 "Power Management"** • No changes. **Section 17 "Register Summary"** The following registers were revised or supplemented: • 17-9 bit 4, description revised. • 17-41 bit, description supplemented. • 17-48 through 17-55 bit 19:2, description revised. • 17-46 bits 2 and 3, description revised to "Reserved." • 17-63 bit 31, description supplemented and bit 3, description revised. • 17-67 bits 10 and 6, descriptions supplemented. Bit 7, BREQ info added and description changed. Bits 3:0, added multiplexed signal name to appropriate signal names. • 17-69 bits 27 through 17, descriptions revised. • 17-80 bit 15, description supplemented. • 17-90 bits 10 through 0, value after reset changed. • 17-109 added a note regarding DRAM registers. • 17-110 bit 15, description supplemented. • 17-126 bit 0, description supplemented. • 17-129 bits 31:0 changed to 31:2, and 1:0 bits made Reserved. • Channel 0 and Channel 1 Mode registers, bits 20, 19, and 15 descriptions revised and supplemented. • In all Bus Region Descriptor registers, changed bit 2 description "Byte Ordering" to "Direct Slave Byte Ordering" and bits 1:0 description, "Bus Width" to "Local Bus Width".

Date	Revision	Comment
7/1999	0.96	**Section 18 "IOP 480 Pin Description"** • Table 18-6 changed from "PCI Configuration Serial EEPROM Interface" to Serial EEPROM Interface." • Tables 18-6 through 18-12, pin types supplemented with mA value. • Table 18-7, function revised to read "…(the signals must be held high by the external pullup resistors). " • Table 18-8, LHOLDACK1 is now LHOLDACK1/BREQ and added note for this signal. • Figure 18-1, LHOLDACK1 is now LHOLDACK1/BREQ. **Section 19 "Electrical Specifications:** • 19.1, changed from "General Electrical Specifications" to "I/O Timing Specifications." • Figure 19-1, changed from "IOP 480 ALE Output Delay to the Local Clock," to "IOP 480 ALE Output Delay from the Local Clock (Min/Max Nanoseconds)," and figure redrawn. • Table 19-1, MAx output delay changed to 10.8. • Table 19-2, Input Voltage (VIN) revised. **Section 20 "Package"** • Figure 20-2, added missing pins. • Figure 20-4, changed height and width measurement to 28.000. **Section 21 "Timing Diagrams"** • TD 21-46, changed from "DMA2 Flyby" to "DMA2 Non-Flyby" • TDs 21-63 through 21-69, revised MA[17:0] signal. • TD 21-70, revised MA[17:0] and BLAST# signals. • TD 21-84. SDRAM Non-Page Mode Burst Read, 4-0-0-0 Wait States; A2C=1, PRCG=2; Master=IOP 480 • TD 21-84, added new diagram, "Long SDRAM Burst Write." **Section 22 "Serial Port Operation"** • Deleted all references to "DMA." • Deleted 22.2.4, "SPU Handshaking I/O Pair Selection." • Deleted last para of 22.7.1. **Section 23 "IOP 480 CPU Overview"** • No changes. **Section 24 "IOP 480 CPU Programming Model"** • No changes. **Section 25 "IOP 480 CPU Cache Operations"** • No changes. **Section 26 "IOP 480 CPU Debugging and JTAG Facilities"** • Minor text changes. • Changed 26.7.1.2 to read "three private instructions," not four. **Section 27 "IOP 480 CPU Memory Management"** • No changes. **Section 28 "IOP 480 CPU Instruction Set"** • Table 28-1, updated page numbers. **Section 29 "IOP 480 CPU Register Summary"** • No changes. **Appendices** • A and B, added missing page cross-references. • C and D, no changes. • E "Ordering Instructions," and F "PLX Technology Representatives and Distributors," added.

Date	Revision	Comment
10/1999	1.0	**Global** • Changed “PowerPC 401 CPU Core” to “PowerPC RISC CPU Core,” as appropriate. • Changed “60 MHz” to “66 MHz” for Local Bus and CPU speeds, including TIming Diagram LCLK signal names (“LCLK-60” and “LCLK-66” changed to “LCLK”, with no CPU speed listed) and refresh rate calculation. **Preface** • Updated additional documetation list to reflect correct addresses and contact numbers. **Section 1 “Introduction”** • Generally revised content to match current chip release information. • 1.1 “Highlights,” enhanced text with several small corrections. • Figure 1-1, added “32-Bit 66 MHz” to Local Bus and “32-Bit 33 MHz” to PCI Bus. • 1.3.1, changed “write only” to “scatter/gather.” • 1.3.2, changed “IBM PowerPC Embedded Tools Program” to “PLX Partners Program.” • Figure 1-6, removed CPU box. **Section 2 “Local Bus Interface”** • Table 2-8, added LAD row. **Section 10 “Reset and Initialization”** • Table 10-1, revised Mode input and output information for Adapter and Host modes. • 10.4, changed “264” to “132.” **Section 11 “Interrupts”** • 11.2.1.2, changed “INTO” to “INTI.” **Section 12 “Memory Controller”** • 12.1 and 12.2, added “external asynchronous FIFOs.” • TDs 12-1, 12-2, 12-3, 12-5, 12-7 and 12-8, changed. **Section 16 “Power Management”** • 16-5 “Power Consumption,” added. **Section 17 “Register Summary”** • Register 17-63, enhanced bit 4 description with serial EEPROM data. **Section 18 “IOP 480 Pin Description”** • Table 18-2, added note regarding use of internal and external pull-up and pull-down resistors. • Added new Table 18-3, “I/O Buffer Types.” Subsequent tables renumbered accordingly. • Table 18-5, supplemented INTA# information regarding Adapter and Host modes. **Section 19 “Electrical Specifications”** • Table 19-1, Specs rearranged, other PCI signals data and other local pins (not EEPROM) inserted, BTERM#, BLAST#, MAx, PCI bus signals, and other Local control signals data removed. Supplemented note with information regarding use of decoupling capacitors. • Added new Table 19-2, “IOP 480 Local Bus Driver Loading Derating,” and Table 19-3, “IOP 480 PCI Buffer Loading Derating.” Subsequent tables renumbered accordingly. **Section 21 “Timing Diagrams”** • TDs 21-63, 21-64, 21-65, 21-67, 21-69, and 21-70, changed. **Section 22 “Serial Port Operation”** • 22.4.1, removed + and - from equations. • Registers 22-7 and 22-9, added IE section. **Section 24 “IOP 480 CPU Programming Model”** • Minor text changes. **Section 29 “IOP 480 CPU Register Summary”** • Register 29-20 bit 13, enhanced description. **Appendix C “Terms and Definitions”** • Changed descriptions of Direct Slave and Direct Master. **Appendix E “Ordering Instructions”** • Table E-1, part numbers changed. **Appendix F “PLX Technology Representatives and Distributors”** • Updated all tables.

IOP 480

Integrated PowerPC I/O Processor

October 1999
Version 1.0

1 INTRODUCTION

1.1 HIGHLIGHTS

- PLX industry-leading advanced Data Pipe Architecture (DPA) PCI controller with dual DMA channels, programmable PCI Initiator and Target data transfer modes, and PCI messaging functions
- Local Bus DMA channel with Flyby DMA support
- 66 MHz 32-Bit PowerPC RISC CPU Core
- 32-Bit PCI interface operating up to 33 MHz
- Flexible Memory controller offering support of up to 256 MB of SDRAM or EDO DRAM of up to 66 MHz
- PCI Bus Arbiter supports up to three external PCI Masters and high priority mode Local Bus Arbiter supports two external Local Bus Masters and high priority mode
- Programmable interrupt controller with multiple timers and counters
- I_2O-*Ready* Messaging Unit with v2.0 performance extensions
- PCI Specification v2.2 Power Management features
- PCI Hot Plug compatible
- CompactPCI® Hot Swap *Friendly*
- PCI Dual Address Cycle (DAC) support
- 32-Bit Multiplexed Local Bus up to 66 MHz supports 8-, 16-, or 32-bit peripheral and memory devices with unlimited bursting up to 264 MB/s
- Debug Serial Port (Tx/Rx)
- IEEE 1149.1 JTAG port for test and debug
- 3.3V, 5V tolerant PCI and Local signaling supports universal PCI adapter designs
- 3.3V Core CMOS in 208-pin PQFP and 225-pin PBGA
- Industrial Temp Range operation
- 600 mW typical power consumption

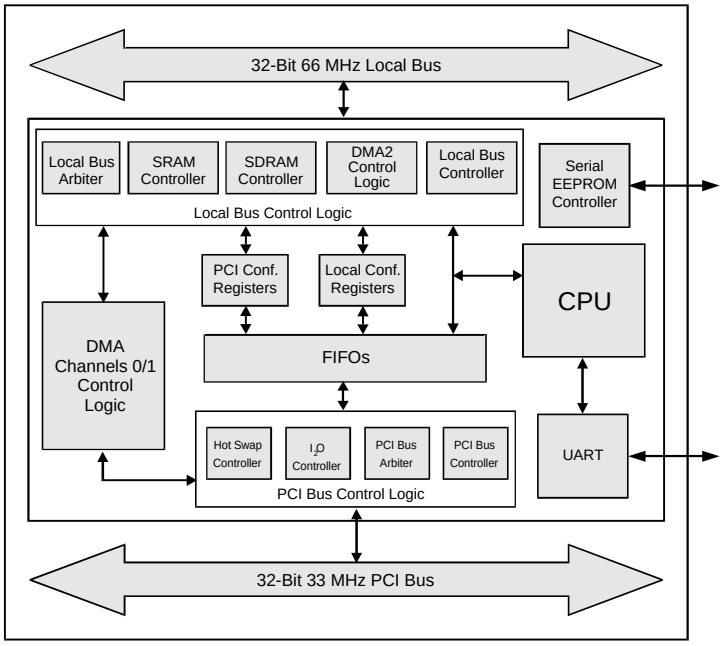

Figure 1-1. IOP 480 Block Diagram

1.2 FEATURES

1.2.1 PowerPC RISC Processor Core

- 66 MHz PowerPC RISC 32-Bit CPU core
- Thirty-two, 32-bit general purpose registers
- Code compatible with PowerPC User Instruction Set Architecture and Development Tools
- Separate 4 KB Instruction cache and Write-Back/ Write-Through 2 KB Data cache
- Hardware multiply and divide
- On-Chip clock generation and power management up to 66 MHz CPU core and 66 MHz Local Bus
- Four timers (64-bit time base, programmable interval timer, fixed interval timer, and watchdog timer)
- Memory Management Unit

1.2.2 On-Chip Peripheral Logic

1.2.2.1 Advanced Data Pipe Architecture

- PCI Specification version 2.2 (v2.2) compliant 32-bit, 33 MHz Bus Master Interface Controller with PCI Power Management features for adapters and embedded systems
- DPA Architecture includes two DMA engines, programmable Target and Initiator Data Transfer modes and PCI messaging functions
- Dual Independent DMA channels with flexible prioritization scheme
    - Direct hardware control of DMA
        - Demand mode DMA operation
        - Burst length control–BTERM
        - End of transfer (EOT)
    - Programmable burst length using thresholds including unlimited burst
    - Shuttle mode DMA channel support with automatic invalidation of used DMA descriptors
    - Unaligned transfer support
    - Supports PCI Bus Mastering from Local Slave-Only devices (and vice-versa)
    - Scatter/Gather List/Ring management
        - Descriptors can be found in PCI Bus memory or in Local Bus memory
        - Automatic polling for valid descriptors (PCI or Local Bus)
        - Automatically returns transfer count upon hardware (EOT) DMA termination
- Local-to-Local DMA
    - Flyby I/O-to-Memory transfers between Local devices
    - DMA initiated Burst reads and writes on the Local Bus (memory-to-memory)
- PCI Initiator Mode (Direct Master Mode)
    - Type 0 and Type 1 configuration cycles
    - Supports all PCI cycle types, including full support for Memory Write and Invalidate (MWI) cycles
    - Initiator READ prefetching
    - Burst length control
    - Unaligned transfer control
    - Endian swapping
- PCI Target Mode (Direct Slave Mode)
    - Multiple independent address spaces
    - Dynamic Local Bus width control
    - Target READ prefetching
    - Endian swapping
    - Local Bus priority control
    - PCI Latency Timer
    - Eight Local Bus chip selects
    - Supports both Memory-Mapped and I/O-Mapped Burst accesses from PCI-to-Local
- PCI Messaging
    - Incorporates an I_2O-*Ready* Messaging Unit, which is fully compatible with the PCI extensions of I_2O Architecture Specification v2.0 (Pull and Outbound option)
    - Complete messaging unit with two Doorbell registers, and eight Mailbox registers with interrupt capability
    - Automatically updating Queue Management pointers, which can be used for message passing under the I_2O protocol or a custom protocol
- VPD Support—Fully supports the Vital Product Data (VPD) PCI extension, which provides an alternate access method other than Expansion ROM for VPD. It is updated for distinguishing identical system boards.
- PCI Dual Address Cycle (DAC) support (64-Bit Address Space)—Supports PCI DAC beyond the 4-GB Address space, which can be used during IOP 480 Bus Master operations (DMA, PCI Initiator).
- PCI $\leftrightarrow$ Locally-sustained data transfers up to 132 MB/sec.
- Six FIFOs for zero wait-state burst in Master, Target, and DMA modes (32-words long for writes and 16 words long for reads).

- PCI Configuration Cycles—In PCI Initiator mode, the PCI controller can generate Type 0 and Type 1 PCI Configuration cycles to enable configuration of other PCI devices or cards in the system.
- Interrupt Generator—The PCI controller can generate PCI and Local interrupts to the internal interrupt controller from several sources, including PCI Interrupt (INTA#), System Error (SERR#), and Parity Error (PERR#).
- Asynchronous Bus Clocks—The Local Bus interface runs from a Local system clock and generates the necessary internal clocks. This clock is capable of running asynchronously to the PCI clock.
- The IOP 480 requires 3.3V Vcc and provides 3.3V signaling with 5V I/O tolerance on both the PCI and Local Buses. It can support universal PCI adapter designs.
- Serial EEPROM Interface—The IOP 480 contains an optional serial EEPROM interface, which can be used to load configuration information. This is useful for loading information unique to a particular adapter (such as, Network ID or Vendor ID).

1.2.2.2 Memory Controller Interface

- 66 MHz Memory-Bus capability
- Programmable timing, supporting either SDRAM, EDO DRAM, SRAM, parallel EEPROM, FIFO, Flash memory, and/or I/O peripheral chips
- High-Bandwidth bus (32-bit data bus)

1.2.2.3 Arbiters

- PCI Bus arbiter supports up to three external Masters
- Local Bus arbiter supports up to two external Masters
- High priority modes supported by both arbiters enables deterministic transfer

1.2.3 JTAG Interface

- IEEE 1149.1 JTAG Boundary Scan interface
- Used for Testing and PowerPC debug

1.2.4 Programmable Interrupt Controller

- Critical Interrupt input (from external devices)
- External Interrupt input (from external devices)
- Timer and debug event interrupts
- Interrupts programmable as either active-high or active-low, and maskable

1.2.5 Local Bus Interface

- Multiplexed 32-bit external bus operates up to 66 MHz
- Compatible with industry-leading DSPs, RISC processors, and a wide variety of I/O and memory devices
- Supports 8-, 16-, or 32-bit peripherals
- Big-Endian or Little-Endian device attachment
- Programmable wait states
- Up to four external programmable peripherals/ memory regions

1.2.6 Programmable Chip Selects

- Provides chip select pins for up to four external I/O or memory-mapped devices (non-SDRAM) connected to the Local Bus
- Each chip select is programmable for bus width and wait states

1.2.7 Serial Port

- Debug Serial UART port for communications with serial devices
- TTL-level RX and TX signals
- Used for controller debug or attaching external serial devices

1.2.8 Data Transfer Mechanisms

The IOP 480 supports four direct data transfer modes:

- **PCI Initiator**—Local External Master initiates cycle to PCI Target
- **PCI Target**—PCI Master initiates cycle to Local Slave, perhaps memory
- **Local Bus Master**—Access to internal registers
- **PCI Bus**—Access to:
    - Local Configuration/Runtime registers
    - PCI Configuration registers
    - I_2O Message queues
- **DMA**—Local-to-PCI**,** PCI-to-Local
- **CPU**—Access to:
    - SPU
    - PCI Target
    - Local Slave
    - All internal registers
- **DMA2**—Local Slave-to-Local Slave (Flyby or Non-Flyby
- **VPD**—Access to serial EEPROM

- **External Local Master**—Access to External Local Slave (as we now own the Local Bus)

1.3 COMPANY AND PRODUCT BACKGROUND

PLX Technology, Inc., the world leader in PCI-to-Local Bus I/O accelerator chips, supports more than 500 OEM customers in a wide variety of PCI applications. Customer applications include PC workstations and servers, PCI add-in boards, embedded PCI communication systems (such as routers and switches), and industrial PCI implementations (such as CompactPCI, PMC, and Passive Backplane PCI).

PLX Technology, Inc., is an active participant in industry standard committees, including the PCISIG®, I_2O SIG®, and PICMG®, and maintains active developer technology and cross-marketing partnerships with industry leaders, such as Intel, IBM, Hewlett-Packard, Motorola, Integrated Systems, WindRiver, and others.

Focused on providing complete solutions for PCI implementations, PLX provides design assistance to customers in the form of Reference Design kits and Software Development kits. Depending upon the application, these kits may include reference boards, API libraries, software debug tools, and sample device drivers with source, enabling customers to quickly bring new designs to production. New tools, application notes, FAQs, and information updates are constantly added to our website (www.plxtech.com) for the convenience of PLX customers. Our expertise and total solutions for the PCI interface allow customers to focus on adding value in their designs without worrying about the complexities of implementing PCI, I_2O, and CompactPCI.

1.3.1 IOP 480 I/O Processor General Description

The IOP 480 is designed to support a new class of products that are used in the embedded space—the I/O processor (IOP). Design requirements driving the development of the IOP include the following:

- Distributed processing architecture (off-load Host processor)
- High-Speed PCI system bus
- High-Performance, low-cost processor
- High-Performance burst mode data transfers
- Intelligent message passing support
- Efficient I/O transaction management
- Performance scalability

The IOP 480 Integrated PowerPC I/O Processor is also designed for applications where cost, space, power consumptionand performance are all equally critical. The IOP 480 provides a high level of integration, reducing chip count from five chips to one, thereby significantly reducing system component cost. The high integration results of the IOP 480 in a simplified board design, less power consumption and faster time-to-market solution. This cost-effective, general-purpose integrated processor targets system PCI interfaces in networking, telecommunications and other embedded markets. The IOP 480 can be used for control purposes in applications such as routers, switches, network storage applications, and image display systems.

The IOP 480 is a high-performance, low-cost, low-power, integrated I/O processor (IOP). It combines a PowerPC RISC CPU, SDRAM/SRAM/EDO Memory Controller, and a PCI v2.2-compliant Bus Master Controller to enable the development of low-cost intelligent PCI adapters and embedded Host PCI systems.

The IOP 480 CPU is a PowerPC core with integrated 4 KB instruction cache and 2 KB data cache. It is code-compatible with other members or the PowerPC 401xx, 403xx, and 60x processor families.

The IOP 480 PCI Bus Master design is based on the industry-standard PLX PCI 9054 I/O Accelerator device. Additional enhancements to the PCI 9054 design include the following:

- DMA Ring management
- Local-to-Local DMA mode
- A Memory controller
- PCI and Local Bus arbiters
- Additional enhancements to the I_2O-*Ready* messaging unit

At the heart of the IOP 480 Bus Master is the PLX Data Pipe Architecture (DPA) technology.

Many high-performance designs are adopting the PCI standard for use in their I/O subsystems. However, implementations vary, which can lead to inefficiencies. The PLX DPA is tuned for PCI efficiency. Its capabilities include significantly reduced management and

housekeeping overhead and intelligent DMA logic for scatter/gather protocols.

1.3.2 Development Tool Support

The IOP 480 is supported by a wide variety of development tools, including the IBM High C/C++ Compiler, IBM RISCWatch emulator, third party compilers, debuggers, Real Time Operating Systems, and other tools available through the PLX Partners Program.

The IOP 480 is fully compatible with PLX PCI SDK and I_2O SDK software development kits, which allow quick and easy development of high performance Local and host PCI software through standard APIs, I_2O messaging protocols, PCI debug tools, and example device drivers.

IOP 480 design support is provided through Reference Design Kits (RDK) which provide a flexible PCI development board, complete with Orcad schematics, documentation, and software. Simulation models for the IOP 480 are also available.

1.4 APPLICATIONS

1.4.1 PCI Adapter Cards

Major PCI adapter card applications for the IOP 480 include high performance communications, networking, disk control, multimedia and video adapters. The IOP 480 moves data between the host PCI Bus and the adapter Local Bus in several ways. First, the host processor may program the IOP 480 DMA controller to move data between the adapter memory and the host PCI Bus. Second, the IOP 480 can perform “PCI Initiator transfers,” whereby a Local mastering device accesses the PCI Bus directly through a PCI Master transfer. The IOP 480 supports slave (target) transfers in which another PCI device is the master. The IOP 480 also has a complete messaging unit with mailbox registers, doorbell registers, and queue management pointers, which can be used for message passing under the I_2O protocol or a custom protocol.

Adapter cards are the primary vehicle for I/O Processing and Host processor off-loading. (Refer to Figure 1-2.)

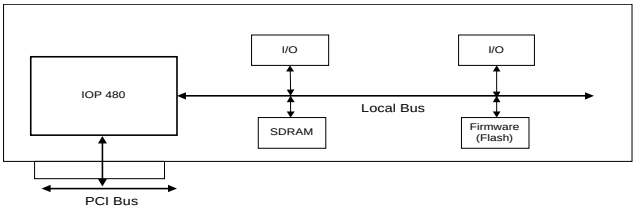

Figure 1-2. Sample PCI I/O Processor Adapter Design

1.4.2 PCI Host Embedded Systems

Another application for the IOP 480 lies in PCI host embedded systems, such as network hubs and routers, printer engines, set top boxes, and industrial equipment. In this configuration, all four of the above-mentioned data transfer modes are used. In addition, the IOP 480 supports Type 0 and Type 1 PCI configuration cycles. This allows the IOP 480 to configure the other PCI devices or cards in the system. The IOP 480 provides a PCI Bus arbiter with support for up to three PCI Bus master devices or PCI slots. (Refer to Figure 1-3.)

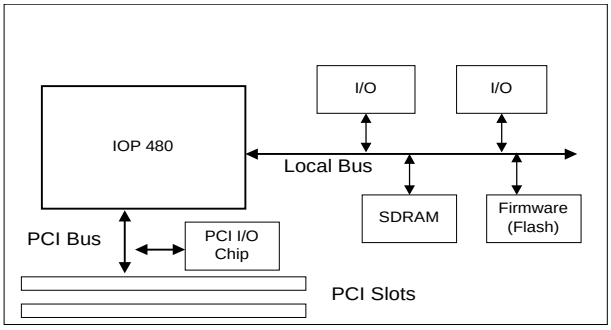

Figure 1-3. Sample PCI Host Embedded System Design

1.4.3 High-Performance PCI I_2O Design

As a member of the I_2O SIG, PLX helped define the I_2O specification, and was the first to offer a processor-independent I_2O implementation in the PCI 9080. The IOP 480 is the first second-generation IOP, incorporating the latest I_2O and PCI performance enhancements. Applications include high-performance storage controllers (RAID) and network interface cards (10/100baseT, ATM).

I_2O also provides an efficient solution for embedded designs. I_2O takes advantage of the PCI performance features, while providing a level of abstraction from both the host operating system and I/O subsystem. The I_2O design simplifies the upgrade path of the user's product to take advantage of future hardware and software performance enhancements. (Refer to Figure 1-4 and Figure 1-5.)

The IOP 480 incorporates the Pull and Outbound option, as specified in I_2O Specification v2.0. These options aid the development of Write-Only architecture.

Advantages:

- Split driver model
- Model coprocessor implementation
- Facilitates peer-to-peer transfers
- Manages batching interrupts
- Aides write-only architectures
- Aides Local Bus balancing

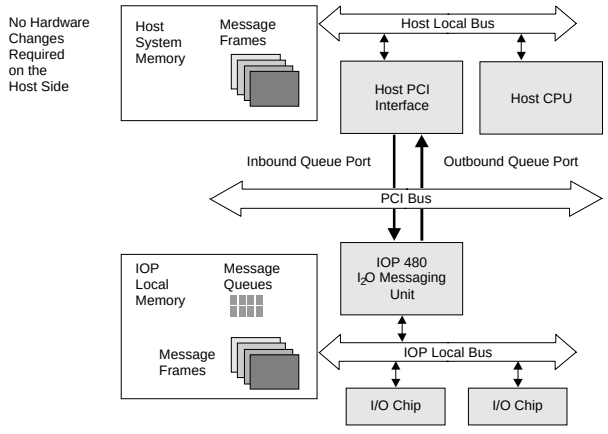

Figure 1-4. Typical I_2O Server/Adapter Card Design

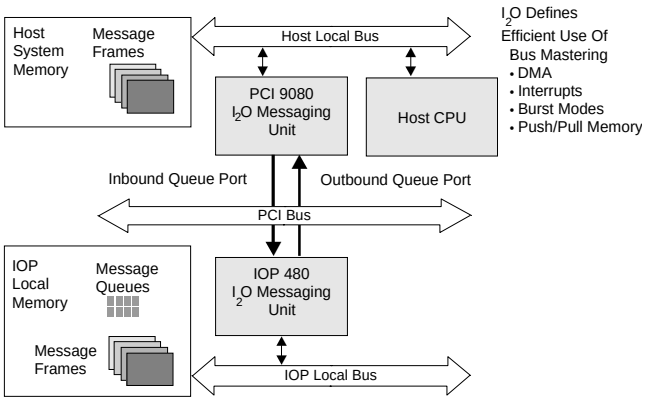

Figure 1-5. Typical I_2O Embedded System Design

1.4.4 High-Performance CompactPCI Adapter Design

Another key application for the IOP 480 is CompactPCI adapters for telecom and networking applications. These applications include high performance communications, such as WAN/LAN controller cards, high-speed modem cards, Frame Relay cards, and telephony cards for telecom switches and remote-access systems.

The IOP 480 has integrated key features to enable live-insertion of Hot Swap CompactPCI adapters. The IOP 480 PICMG 2.1 R1.0-compatible Hot Swap *Friendly* PCI interface includes both Hot Swap *Capable* and Hot Swap *Friendly* features. (Refer to Figure 1-6.)

1.4.4.1 Hot Swap Capable

- *PCI Specification v2.1* or better
- Tolerant of Vcc from early power
- Tolerant of asynchronous reset
- Tolerant of precharge voltage
- Limited I/O pin leakage at precharge voltage

1.4.4.2 Hot Swap Friendly

- Incorporates the Hot Swap Control/Status register (HSCSR)
- Incorporates an Extended Capability Pointer (ECP) mechanism
- Incorporates added resources for software control of ENUM#, the ejector switch, and the status LED, which indicates insertion and removal to the user

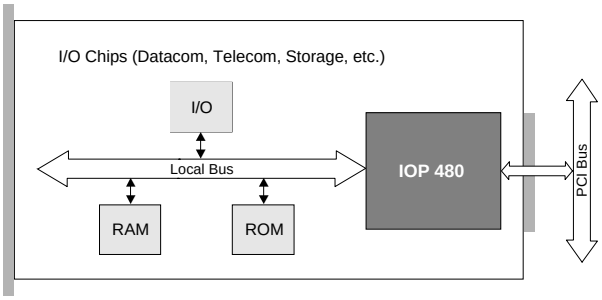

Figure 1-6. High-Performance CompactPCI Adapter

1.4.5 Real Time Application Design

To face the problems of real-time applications on the PCI Bus, The IOP 480 provides a high-priority mode. The PCI Bus provides a 132 MB/s of bandwidth. However, there is no built-in mechanism for a particular application to have a consistent slice of that bandwidth. If a plug-in card is gathering real-time data, even at relatively slow rates, large buffers are required on board to guarantee that the none of the data is dropped. Dropped data may appear as missed frames in a movie or clicks in an audio stream. (Refer to Figure 1-7.)

To address this problem, the IOP 480's arbiters can be put into a High-Priority mode. Instead of using the standard fairness algorithm, the IOP 480 can be set up to be deliberately biased towards a particular transfer. If the internal DMA Channels (Channels 0 and 1) are used to transfer data and the built-in local and PCI arbiters are set for high priority (and used), the IOP 480 delivers a guaranteed minimum bandwidth.

This amount of guaranteed bandwidth depends on several factors. The burst capability of the PCI and local targets, the relative speeds of the PCI and Local Buses, and the Bus Latency Timer settings all contribute to this guaranteed bandwidth. The number and traffic patterns of other masters on either bus do not effect this bandwidth. A 33 MHz system capable of bursts up to 16 words can easily achieve a 100 MB/s guaranteed minimum bandwidth with the IOP 480. (Refer to Figure 1-7.)

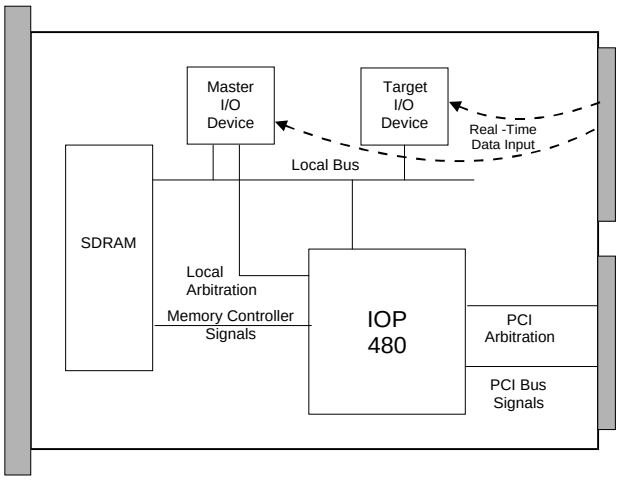

Figure 1-7. Sample Real-Time Application

1.4.6 Data Communications Design

In many datacom and telecom applications, the IOP 480 is tasked to work in conjunction with a processor (either in the IOP itself or externally) to transfer data to and from I/O chips. This usually means that the processor would set up DMA descriptor chains for both Transmit and Receive operations. To occupy a fixed block of memory, the tails of these chains are made to point back to the head, resulting in descriptor rings. The IOP 480 incorporates ring management specifically for this type of application. (Refer to Figure 1-8.)

The rings and the I/O chips may reside either on the PCI Bus or on the Local Bus. One of the IOP's DMA channels would be set for Local-to-PCI transfers and the other for PCI to local transfers (for separate transmit and receive rings). The IOP utilizes a valid bit in each DMA descriptor link to keep track of its location in the ring sequence. This Valid bit is automatically invalidated at the completion of each descriptor link. As the IOP circles through the rings, the managing processor can update the links and then again validate them. If the IOP 480 reaches an invalidate link, it waits for that link to become valid before processing.

In this manner, both the IOP and the managing processor can run at their fastest possible speed without interrupts. The end of a packet would be signaled to the IOP via the EOT# (end of transfer) pins. When the IOP encounters an EOT, instead of simply invalidating the current descriptor link, it writes the number of words remaining to be transferred (in the current link) to the current descriptor link location, giving an idea of packet size.

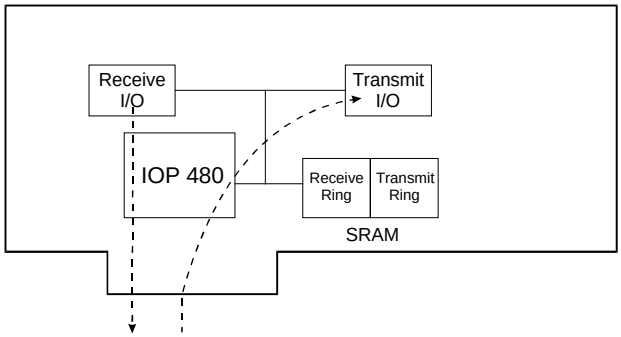

Figure 1-8. Data Communication Design