PowerPro (CA91L750)
PowerPC Memory Controller Manual

www.tundra.com

Trademarks

Tundra and the Tundra logo are registered trademarks of Tundra Semiconductor Corporation (“Tundra”). All other trademarks, services marks and logos (“Marks”) are the property of their respective holders including Tundra. Users are not permitted to use these Marks without the prior consent of Tundra or such third party owning the Mark. The absence of a Mark identifier is not a representation that a particular product or service name is not a Mark.

Copyright

Published in Canada

Disclaimer

Tundra assumes no responsibility for the accuracy or completeness of the information presented which is subject to change without notice. In no event will Tundra be liable for any direct, indirect, special, incidental or consequential damages, including lost profits, lost business or lost data, resulting from the use of or reliance upon the information, whether or not Tundra has been advised of the possibility of such damages.

Mention of non-Tundra products or services is for information purposes only and constitutes neither an endorsement nor a recommendation.

Product: PowerPro (CA91L750)
Title: PowerPro PowerPC Memory Controller Manual
Document: 80A5000_MA001_05
Status: Preliminary
Modification Date: February, 2000 Tundra Semiconductor Corporation

---

Revision History

80A5000_MA001_05, Preliminary, February 2001

This manual discusses the PowerPro device. This is a full revision of the 80A5000_MA001_04 manual. All of the chapters have been edited.

80A5000_MA001_04, Preliminary, August 2000

This manual discusses the PowerPro device. The following sections have been updated to reflect technical information changes:

- Chapter 6: “Dual UART Interface” on page 87
- Chapter 16: “ Registers” on page 153 (UART registers)

This document supersedes the previous *PowerPro PowerPC Memory Controller Manual*, document number 80A5000_MA001_03.

80A5000_MA001_03, Preliminary, July 2000

This manual discusses the PowerPro device. This is the first version of the preliminary *PowerPro PowerPC Memory Controller Manual*.

The following sections have been updated to reflect technical information changes:

- Chapter 4: “SDRAM Interface” on page 51
- Chapter 12: “Reset, Clock and Power-up Options” on page 111
- Chapter 16: “ Registers” on page 153
- Chapter 17: “Timing” on page 237
- Appendix A: “Typical Applications” on page 241

This document supersedes the previous *PowerPro PowerPC Memory Controller Manual*, document number 80A5000_MA001_02.

80A5000_MA001_02, Advance, May 2000

This manual discusses the PowerPro device. The following sections have been updated to reflect technical information changes:

- Chapter 12: “Reset, Clock and Power-up Options” on page 111
- Chapter 15: “Electrical Characteristics and Pin Information” on page 139

The following signals have been updated in the pin information chapter:

Ball	Function			
	0	1	2	3
Y5	PB_DBG[0]_	INT[20]	GPIO[16]	EE_A[24]
W6	PB_DBG[1]_	INT[21]	GPIO[17]	EE_A[23]

This document supersedes the previous *PowerPro PowerPC Memory Controller Manual*, document number 80A5000_MA001_01.

80A5000_MA001_01, Advance, April 2000

This manual discusses the PowerPro device.This is the first version of the *PowerPro PowerPC Memory Controller Manual.*

Contents

List of Figures

List of Tables

Corporate Profile

Tundra Semiconductor Corporation

Tundra Semiconductor Corporation (TSE:TUN) designs, develops, and markets advanced System Interconnect for use by the world's leading Internet and communications infrastructure vendors. Tundra chips provide the latest interface and throughput features to help these vendors design and deliver more powerful equipment in shorter timeframes. Tundra products are essential to a range of applications, including telecommunications, data communications, wireless communications, industrial automation, and ruggedized systems. Tundra headquarters are located in Kanata, Ontario, Canada, and sales offices are based in Mountain View, California and Maidenhead, U.K. Tundra sells its products worldwide through a network of direct sales personnel, independent distributors, and manufacturers' representatives. More information is available online at www.tundra.com.

Greater Demand, Greater Opportunity

The increasingly complex requirements placed on the Internet, intranets and extranets have created an insatiable demand for higher speed and greater capacity in communications networks. The evolution of converging communications networks requires higher levels of security and increasingly sophisticated network intelligence. These network demands, and the user expectations that drive them, have created a global need for well-managed and ever-increasing bandwidth.

Tundra helps meet those demands by creating underlying technology that enables the accelerated flow of voice, data, and video information over communications networks. Tundra products can be found in a broad range of applications, including telecommunications, data communications, wireless communications, industrial automation, and avionics. Communications infrastructure vendors rely on Tundra for off-the-shelf, standards-based, easy-to-deploy and highly scalable System Interconnect products.

Tundra System Interconnect

Tundra is *System Interconnect*. Tundra uses the term System Interconnect to refer to the technology used to connect all the components and sub-systems in almost any embedded system. This concept applies to the interfacing of functional elements (CPU, memory, I/O complexes, etc.) within a single-board system, and the interfacing of multiple boards in a larger system.

System Interconnect is a vital enabling technology for the networked world. The convergence of voice, video, and data traffic, the need for more secure communications, and the exploding demand for high-speed network access are putting communications infrastructure vendors under intense pressure to provide faster, well-managed bandwidth that also integrates smoothly with existing technology. Tundra System Interconnect helps these vendors address their customer needs. It enables them to build standards-based network equipment that can scale to multi-gigahertz speeds and also integrate with existing infrastructure.

Partnerships

Fundamental to the success of Tundra is its partnerships with leading manufacturers, including Motorola, Compaq and Texas Instruments. As a result of these alliances, Tundra devices greatly influence the design of customers' architectures. Customers are changing their designs to incorporate Tundra products. This highlights the commitment Tundra holds to be a significant part of its customers' success.

The Tundra design philosophy is one in which a number of strategic customers are invited to participate in the definition, design, test and early silicon supply phases of product development. Close working relationships with customers and clear product roadmaps ensure that Tundra can anticipate and meet the future directions and needs of communications systems designers and manufacturers.

Tundra Customers

Tundra semiconductor products are used by the world's leading communications infrastructure vendors, including Cisco, Motorola, Ericsson, Nortel, Lucent, IBM, Xerox, Hewlett-Packard, 3Com, Nokia, Siemens, Newbridge, Alcatel, Matsushita, OKI, Fujitsu, Samsung, and LGS.

Tundra Customer Support

Tundra is respected throughout the industry for its outstanding commitment to customer support. Tundra ensures that its customers can take immediate advantage of the company's products through its Applications Engineering Group, unmatched Design Support Tools (DST), and full documentation. Customer support also includes Web-based and telephone access to in-house technical resources.

Tundra System Interconnect … Silicon Behind the Network ™

Chapter 1: General Information

This chapter describes the main functions and features of PowerPro (CA91L750). It also discusses general document elements and technical support information. The following topics are discussed:

1.1 What is the PowerPro?

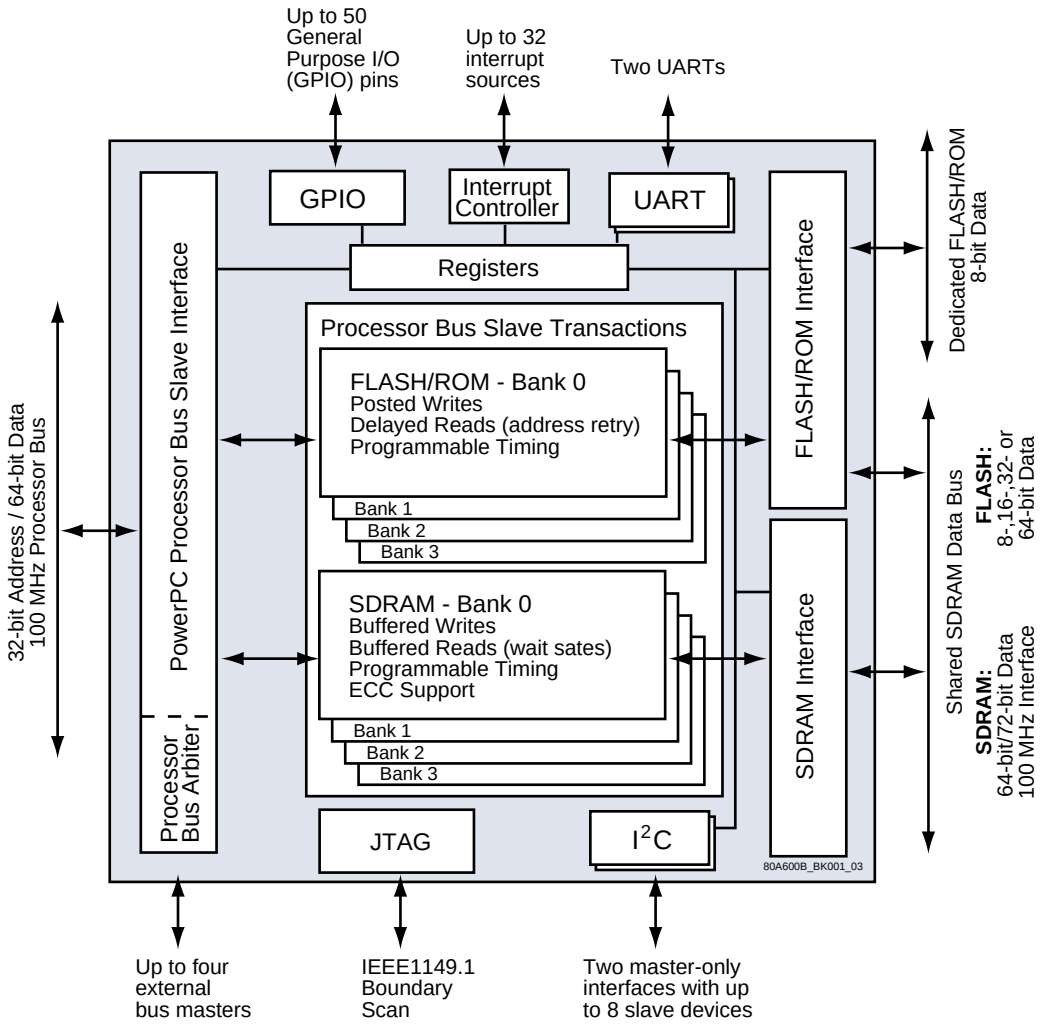
PowerPro is a memory controller for PowerPC processors. PowerPro is designed to interconnect with embedded PowerPC processors — the Motorola MPC8260 (PowerQUICC II), PowerPC 603e, PowerPC 740, PowerPC 750 and PowerPC 7400.

PowerPro can be used as a stand-alone device in non-PCI applications, or as a companion with the Tundra PowerPC-to-PCI bus switch — PowerSpan. Together, PowerSpan and PowerPro are a complete PowerPC chipset offering. With this chipset, Tundra implements a new modular approach to PowerPC system design. The chipset provides embedded designers with the ability to adapt PowerPC as a computing platform in the communications market.

PowerPro is designed for applications to leverage the Switched PCI architecture of PowerSpan in PowerPC applications. PowerPro enables ECC protection in PowerPC 750 and MPC8260 applications.

For more information on PowerSpan, please refer to the *PowerSpan PowerPC-to-PCI Bus Switch Manual*. This manual is available on the Tundra website at www.tundra.com.

Figure 1: PowerPro (CA91L750) Block Diagram



1.2 PowerPro Features

PowerPro (CA91L750) features are listed in the following sub-sections.

1.2.1 Processor Interface

- Direct-connect support for:
  - — MPC8260 (60x interface)
  - — PowerPC 603e, PowerPC 740, PowerPC 750, PowerPC 7400 (60x interface)
- 66 to 100 MHz bus frequency

- 32-bit address, 64-bit data
- Address and data parity
- De-coupled address and data bus operation
- Bus Slave
  - 64-bit port size
  - Eight programmable slave images for memory peripherals
  - Programmable register image
- Bus Arbiter
  - Supports up to four external bus masters
  - Two level fair arbitration scheme
  - Independent address and data bus arbitration
  - Programmable bus parking
  - Boot control

1.2.2 SDRAM Interface

- Operating at processor (60x) bus speed
- 64-bit interface for non-ECC operations
- 72-bit interface (64-bit data and 8 check-bits) for ECC applications
- ECC protection applied to the data path
- Page management for optimal read and write access times. The SDRAM Interface has the ability to have 32 logical banks open simultaneously.
- Four banks supported with two chip selects per bank, up to 1 Gbyte per bank.
- Programmable timing parameters per bank
- Programmable address mappings per bank

1.2.3 FLASH/ROM Interface

- Four banks of FLASH/ROM/SRAM
- Direct support for 8-, 16-, 32-, and 64-bit external peripherals
- Programmable timing per bank
- 64-bit packed reads for PowerPC bus accesses
- Programmable address mappings

1.2.4 Integrated Peripherals

- Two high-speed UARTs (DUART)
- I^2C Interface
- Programmable General Purpose Timer, four compare and four trigger settings.
- System Watchdog Timer

- 32 channel Interrupt controller, interrupts from external and internal (UART, I^2C, timers, ECC, errors) sources
- 50 General purpose I/O pins. These pins are multiplexed with other functions.
- JTAG support for board level testing

1.2.5 Registers

- Bidirectional interrupt pins (maskable/routable)

1.2.6 Packaging

- 324-pin HPBGA
  - 23 mm body size
  - 1.0 mm ball pitch

1.2.7 Operating Environment

- Commercial
- Industrial

1.3 PowerPro Benefits

PowerPro offers the following benefits to designers:

- Modular PowerPC design
- Increased through-put with memory system optimization
- Proven product testing in a hardware emulation environment
- Reduced design effort in both software and hardware PowerPC systems
- ECC protection in PowerPC 750 and MPC8260 applications.

1.4 PowerPro Typical Applications

PowerPro is targeted at communications infrastructure applications that use both PowerPC architectures, such as the following:

- ADSL concentrators
- CDMA base stations
- VoIP gateways
- Ethernet switches
- VPN equipment
- MPEG 2 encoders
- Exchange carrier switching equipment

Figure 2 illustrates a typical PowerPro application involving the PowerPC 750 and PowerSpan. In this diagram PowerPro is used in both a PowerPC and PCI system.

Figure 2: PowerPro, PowerPC and PCI Application

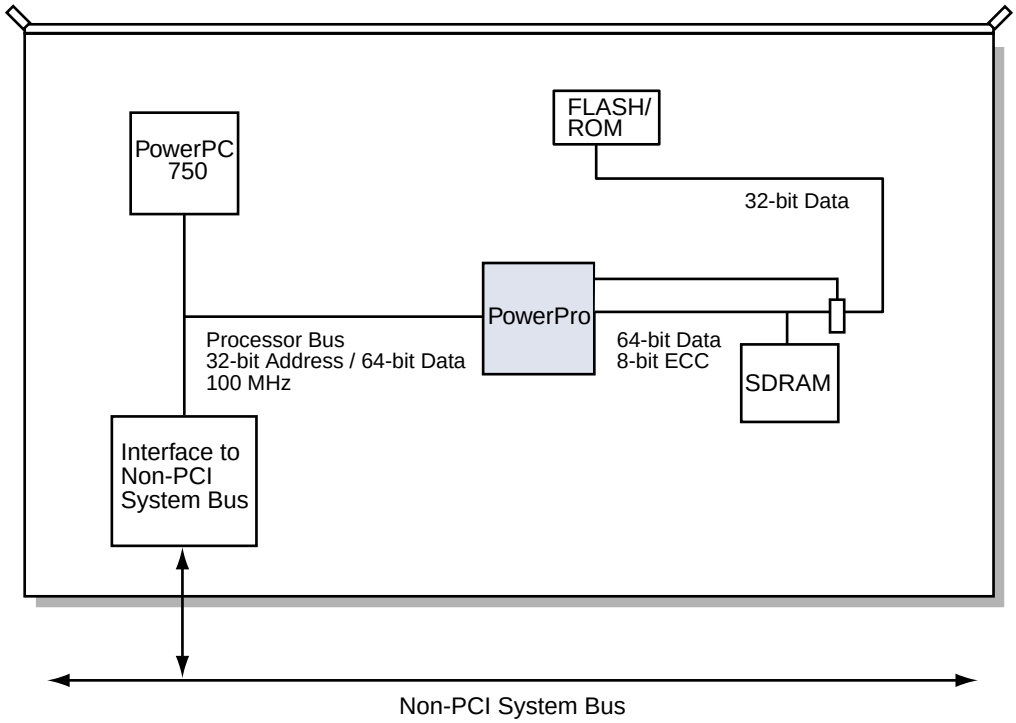
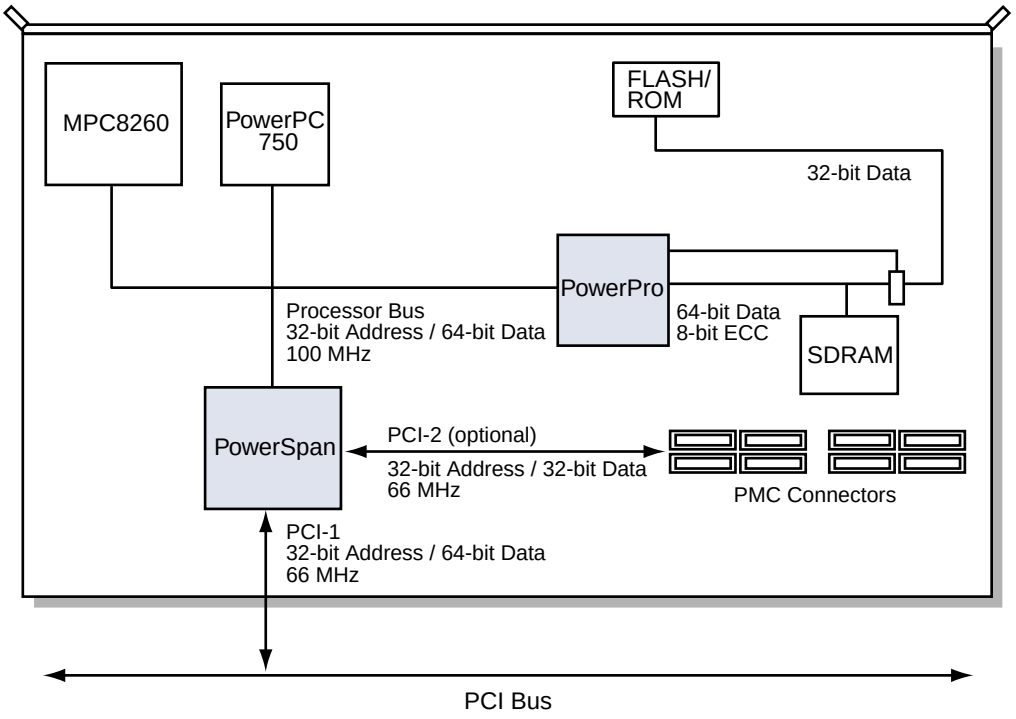


Figure 3 illustrates a typical PowerPro application involving the MPC8260 and the PowerPC 750. In this diagram PowerPro is used in both a PowerPC system.

Figure 3: PowerPro PowerPC Application



1.5 Document Conventions

1.5.1 Signals

Signals are either active high or active low. Active low signals are defined as true (asserted) when they are at a logic low. Similarly, active high signals are defined as true at a logic high. Signals are considered asserted when active and negated when inactive, irrespective of voltage levels. For voltage levels, the use of 0 indicates a low voltage while a 1 indicates a high voltage.

The following signal conventions are used:

- SIGNAL_: indicates an active low signal
- SIGNAL#: indicates an active low signal on the PCI bus

1.5.2 Bit Ordering

This document adopts the convention that the most significant bit is always the largest number (also referred to as *Little-Endian* bit ordering). For example, the PCI address/data bus consists of AD[31:0], where AD[31] is the most significant bit and AD[0] is the least-significant bit of the field.

1.5.3 Numeric Conventions

The following numeric conventions are used:

- Hexadecimal numbers are denoted by the suffix *h*. For example, 04h.
- Binary numbers are denoted by the suffix *b*. For example, 10b.

1.5.4 Typographic Conventions

The following typographic conventions are used:

- *Italic* type is used for the following purposes:
  - — Book titles: For example, *PowerPro PowerPC Memory Controller Manual.*
  - — Important terms: For example, when a device is granted access to the processor (60x) bus it is called the bus *master*.
  - — Undefined values: For example, GPIO ports are enabled by the setting the GPIO_*x* register.
- `Courier` type is used to represent a file name or text that appears on a computer display. For example, “run `loadext.exe` by typing it at a command prompt.”

1.5.5 Symbols

This symbol directs the reader to useful information or suggestions.

This symbol alerts the reader to procedures or operating levels that may result in misuse or damage to the device.

1.5.6 Document Status

Tundra technical documentation is classified as either Advance, Preliminary, or Final. These classifications are briefly explained:

- Advance: The Advance manual contains information that is subject to change. The Advance manual exists until device prototypes are available. This type of manual can be downloaded from our website at www.tundra.com.
- Preliminary: The Preliminary manual contains information about a device that is near production-ready, and is revised as required. The Preliminary manual exists until the device is released to production. This type of manual can be downloaded from our website.
- Final: The Final manual contains information about a final, customer-ready device. This type of manual can be downloaded from our website. It can also be ordered in print format by calling 613-592-0714 or 1-800-267-7231 (please ask for customer service), or by email at docs@tundra.com.

1.6 Related Documentation

Before you read this manual you should be familiar with the following:

- *PowerSpan PowerPC-to-PCI Bus Switch Manual*
- *MPC8260 User's Manual*

1.7 Customer Support Information

Tundra is dedicated to providing its customers with superior technical documentation and support. The following types of support are available:

- PowerPro Web Page: This web page briefly describes PowerPro's features, benefits, typical applications, and block diagram. For more information, please visit Tundra's website at www.tundra.com.
- PowerPro Design Support Tools (DST) Web Page: This web page contains the latest manual, manual addenda, application notes, design notes, and device errata. You can customize how the PowerPro DST web page is presented to you, and also register to receive email notification when a documentation resource changes.
- docs@tundra.com: You can order Tundra documentation manuals using this email address. Please include PowerPro in the subject header of your message.
- support@tundra.com: You can also send technical questions and feedback to Tundra using this email address. Please include PowerPro in the subject header of your message.
- Phone Support: Technical support staff can be reached at 613-592-0714 or 1-800-267-7231. Please ask for PowerPro technical support.
- Sales Support: Tundra has an extensive and experienced sales network that communicates information about its products. If you would like to locate a sales representative near you, please visit our website at www.tundra.com.
- Mailing Address:
  Tundra Semiconductor Corporation
  603 March Road
  Kanata, ON
  K2K 2M5

Chapter 2: Functional Overview

PowerPro (CA91L750) is a high-performance memory controller for embedded PowerPC processors. This chapter discusses main PowerPro functionality. The topic addressed in this chapter includes:

2.1 Overview

PowerPro is designed to interconnect with embedded PowerPC processors — the Motorola MPC8260, PowerPC 603e, PowerPC 740, PowerPC 750 and PowerPC 7400. Both the MPC8260 and the PowerPC 7400 must operate in processor (60x) compatible bus mode to be used in PowerPro systems.

PowerPro (CA91L750) can be used as a stand alone device in non-PCI applications or as a companion with the Tundra PowerPC to PCI bus switch — PowerSpan. When used with PowerSpan, the two devices deliver a complete, modular PowerPC chipset solution. For more information on PowerSpan, please refer to the *PowerSpan PowerPC-to-PCI Bus Switch Manual*. This manual is available on the Tundra website at www.tundra.com.

2.2 Architecture

PowerPro (CA91L750) operates in PowerPC systems and has a 64-bit, 100 MHz SDRAM Interface that allows the PowerPC 750 and PowerPC 740 to access memory on a high-speed bus. PowerPro (CA91L750) also has a 64-bit Processor Bus Interface and a FLASH/ROM Interface.

The PowerPro Processor Bus Interface responds to read and write requests from external bus masters to any of the four SDRAM memory banks, the four FLASH/ROM memory banks, and internal registers.

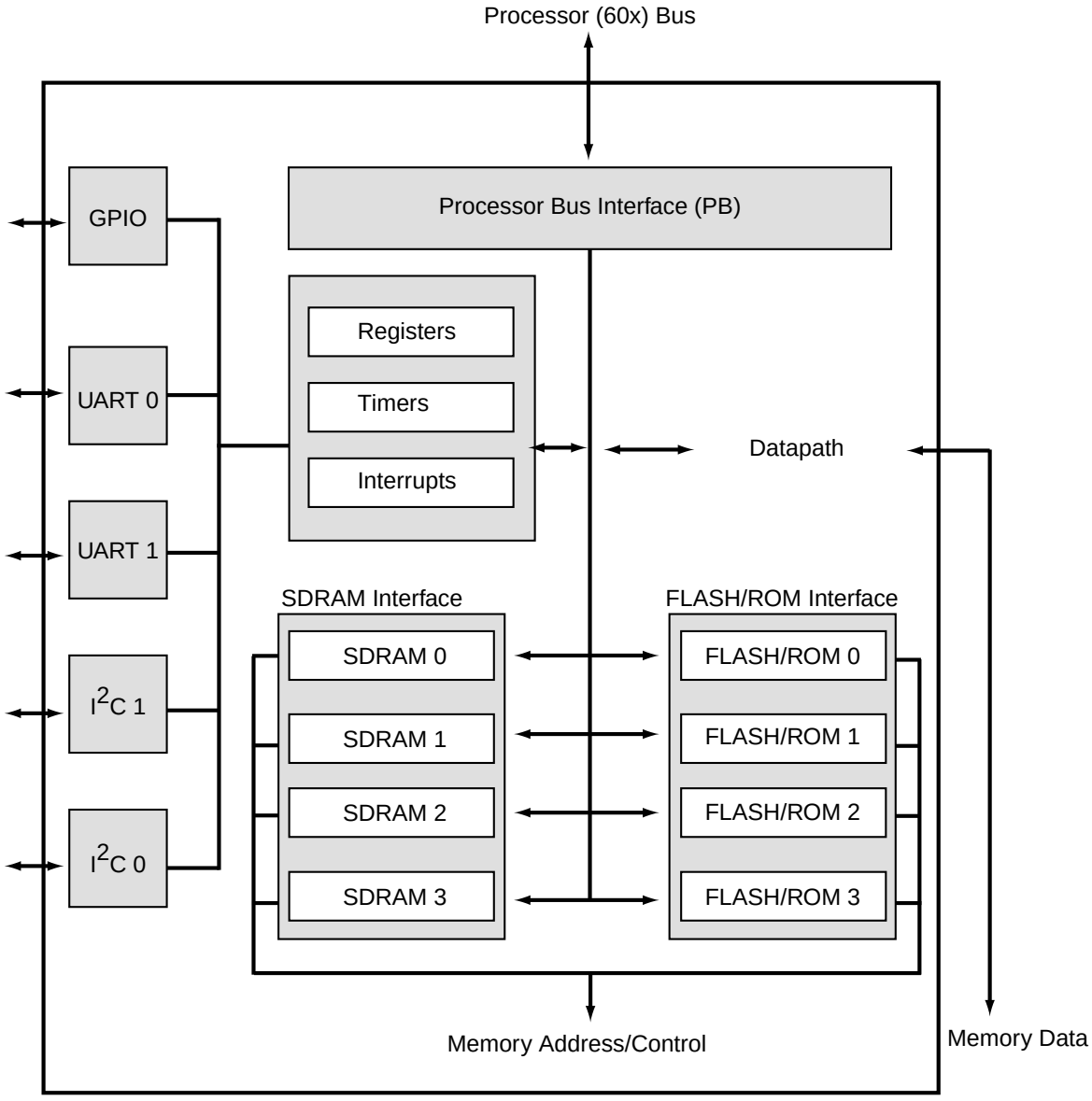
Reads and writes to the Processor Bus Interface are processed in the order that they are received by PowerPro.

PowerPro's main architectural elements include:

- Processor Bus (PB) Interface
  - 64-bit data bus
- Processor Bus arbiter
  - Four external bus masters
  - Hardware verified against PowerPC 750, PowerPC 740 and MPC8260 bus interfaces
- SDRAM Interface
  - 64-bit data bus
  - 100 MHz operating frequency
  - PC-100 compliant
- FLASH/ROM Interface
  - 8-,16-,32-, or 64-bit
- Registers
  - 8 byte reads
- I^2C Interface
- General Purpose I/O ports (GPIO)
- Dual UARTs (DUART)
- Interrupt controller
- JTAG

The functional diagram in Figure 4 outlines the major components of the PowerPro 64-bit architecture.

Figure 4: PowerPro (CA91L750) Dataflow Diagram



Each of these major blocks are discussed in the following sections. For more detailed information, refer to specific chapters in this manual.

Not all PowerPro functionality is available at one time. Many of the functions on PowerPro are multiplexed. Refer to Chapter 14: “Signal Description” on page 129 and Chapter 15: “Electrical Characteristics and Pin Information” on page 139 for more information on multiplexed functionality.

2.2.1 Processor Bus (PB) Interface

PowerPro (CA91L750) has a Processor Bus Interface that directly connects to a variety of PowerPC microprocessors. PowerPro (CA91L750) supports the following PowerPC microprocessors: MPC8260 (PowerQUICC II), PowerPC 603e, PowerPC 740, PowerPC 750 and PowerPC 400. The PB Interface operates at 100 MHz in PowerPC 740 and PowerPC 750 applications.

The PB Interface has a 32-bit address bus and 64-bit data bus. It is a slave only interface and supports single-beat and burst data transfers. The address and data buses are decoupled for piplined transactions, and also support MPC8260 extended cycles. Extended cycles on the MPC8260 offers more flexible bursting and more efficient use of the processor bandwidth.

The PowerPro PB Interface is connected to the processor (60x) bus. Please refer to Motorola documentation for specific information on the processor (60x) bus and its requirements.

For more information on the PB Interface, refer to Chapter 3: “Processor Bus Interface” on page 37.

2.2.1.1 Processor Bus Arbitration

The PB Interface has an integrated bus arbiter. The arbiter supports four external bus masters for applications involving multiple bus masters.

The PB arbiter implements two levels of priority. Devices programmed into a specific priority level operate in a round robin fashion. Each master has a register to determine its arbitration level for the address bus. The arbitration level for each master is programmable.

2.2.2 SDRAM Interface

PowerPro provides control functions and signals for JEDEC-compliant SDRAM devices.

PowerPro enables access to four (single or dual) DIMMs of SDRAM with two chip selects per DIMM. Both buffered (registered) and non-buffered DIMMs are supported. PowerPro has a maximum of 4 Gbytes of addressable SDRAM memory. All SDRAM parameters are selectable per bank, which allows each DIMM socket to contain a different variety of SDRAM. This capability gives each SDRAM optimized timing parameters.

For more information on the SDRAM Interface, refer to Chapter 4: “SDRAM Interface” on page 51.

2.2.2.1 ECC Protection

ECC protection is an alternative to simple parity detection. PowerPro offers ECC protection for the data path between PowerPro and system memory. ECC detects errors, and corrects single-bit errors in the 64-bit data path.

PowerPro enables ECC protection in MPC8260 and PowerPC 750 applications. The MPC8260 can enable ECC protection in certain applications, but when multiple processors are involved there is a potential for processor (60x) bus compliancy issues.

2.2.3 FLASH/ROM Interface

PowerPro supports four distinct banks of FLASH/ROM devices. The FLASH/ROM devices can either share the SDRAM data bus or have a single, dedicated 8-bit data bus. When the FLASH/ROM Interface shares the SDRAM data bus, the width of the bus can be configured as 8-, 16-, 32-, or 64-bit.

The FLASH/ROM Interface is a flexible interface with many multiplexing options to support a variety of address and data bus requirements.

Each of the four independent FLASH/ROM banks have individually programmable address images. Individual, internal chip-select machines drive each of the four FLASH/ROM banks. The chip-select machines arbitrate for required resources. This ability provides flexible system functionality.

For more information on the FLASH/ROM Interface, refer to Chapter 5: “FLASH/ROM Interface” on page 69.

2.2.4 Registers

The 512 byte control and status registers are used to program device specific parameters, as well as monitor settings. All registers are 32-bit and are accessible by external masters through the PB Interface. Register accesses can either be 4 or 8 bytes.

For more information on registers, refer to Chapter 16: “ Registers” on page 153.

2.2.5 I^2C Interface

PowerPro contains two master-only, I^2C bus compatible interfaces. Each interface supports up to eight I^2C slave devices.

The I^2C Interfaces are connected to serial presence detect EEPROMs; these are commonly found on DIMM modules.

In a system that contains both PowerSpan and PowerPro (CA91L750), the same EEPROM can be shared between the two devices.

For more information on the I^2C Interface, refer to Chapter 8: “I^2C Interface” on page 95.

2.2.6 General Purpose I/O Port

PowerPro has flexible general purpose I/O capability. Although all PowerPro pins are programmed with a primary purpose, in many instances these pins are not enabled because their functionality is not required in the system. These unused pins are assigned to the General Purpose I/O (GPIO) pool. All pins capable of GPIO are mapped in a GPIO register.

The GPIO port, combined with the general purpose timers available on PowerPro (CA91L750), enable software to control any low to medium speed device that is not time critical.

For more information on the GPIO Port, refer to Chapter 7: “General Purpose I/O Interface” on page 93.

2.2.7 UART Interface

PowerPro (CA91L750) has two, serial universal asynchronous receiver transmitter (UART) protocol interfaces. These dual UARTs (DUART) complete parallel-to-serial conversion of digital data which must be transmitted, and complete the serial-to-parallel conversion of digital data which has been transmitted.

For more information on the UART Interface, refer to Chapter 6: “Dual UART Interface” on page 87.

2.2.8 JTAG Interface

PowerPro (CA91L750) has a Joint Test Action Group (JTAG) Interface to facilitate boundary-scan testing. The JTAG Interface implements the five test port signals required to be fully compliant with IEEE 1449.1 specification. Refer to the *IEEE 1449.1 Boundary-scan Specification* for more information.

For more information on the JTAG Interface, refer to Chapter 13: “JTAG Interface” on page 125.

Chapter 3: Processor Bus Interface

This chapter outlines the functionality of the Processor Bus Interface. The following topics are discussed:

3.1 Overview

The PowerPro Processor Bus (PB) Interface is a slave only interface. The PB Interface data width is 64-bit and the address width is 32-bit. The maximum operating frequency is 100 MHz.

3.1.1 Interface Support

The PB Interface is specifically designed to support the following PowerPC devices:

- MPC8260 (PowerQUICC II)
- PowerPC 603e/740/750
- PowerPC 7400

The MPC8260 and the PowerPC 7400 must operate in processor (60x) compatible bus mode to be used in systems. The MPC8260 has the option to operate in Single MPC8260 bus mode; PowerPro does not support this mode. The MPC7400 has the option to operate in MPX mode; PowerPro does not support this mode.

The supported PowerPC processor interfaces are not identical. However, the PB Interface discusses the processor (60x) bus protocol used by all the supported processors. The following sections highlight how PowerPro (CA91L750) operates differently to address specific processor requirements. An example of different operation in PowerPC devices is the extended cycles with the MPC8260.

3.1.2 Terms

The following terms are used in the PB Interface description.

- **Address retry window**: This term refers to the clock following the assertion of PB_AACK_. It is the latest cycle that a snooping master can request for an address tenure re-run.
- **Window of opportunity***:* This term refers to the clock following the assertion of PB_ARTRY_. The retrying master must request the bus on this clock to ensure that it is the next bus owner. This allows it to perform the transactions required to maintain cache coherency.

3.2 Processor Bus Interface

3.2.1 Overview

The PowerPro PB Slave Interface claims processor (60x) bus transactions intended for the SDRAM Interface, FLASH/ROM Interface, or the PowerPro internal control and status registers. The slave claims the transactions based on the address decode information in PowerPro. Refer to “Address Decoding” on page 38 for information on base address registers that are programmed in PowerPro.

3.2.2 PowerPro as PB Slave

The operation of the PB Slave Interface is described by dividing the PB slave transaction into three different phases:

- **Address Phase**: This section discusses the decoding of processor (60x) bus accesses.
- **Data Phase**: This section describes control of transaction length.
- **Terminations**: This section describes the terminations supported by PowerPro (CA91L750), as well as exception handling.

Pull-up resistors are not required on the processor bus address (PB_A[0:31]) and data (PB_D[0:63]) signals to guarantee functional operation of PowerPro. However, adding resistors to the address and data signals minimizes the current drawn by PowerPro's tristated buffers when the bus is in an idle condition. The system designer must decide whether to add these resistors to the address bus and data bus.

3.2.3 Address Phase

3.2.3.1 Address Decoding

A PB slave image is the range of PB physical address space used to decode a PowerPro (CA91L750) access. A slave image is controlled by the information programmed in the base address registers. Each slave image monitors the Processor Bus Base Address (PB_REG_ADDR) register (see page 161). When the address falls into the configured address range, and the Processor Bus Transfer Type (PB_TT) is supported, PowerPro claims the address tenure. See “Transaction Types” on page 39 for more information on supported transaction types.

PowerPro (CA91L750) has the following slave (address) images:

- “PB Register Base Address” on page 161
- “SDRAM Memory Bank x Address” on page 175
- “ROM Memory Bank x Address” on page 181

The “x” symbol in the register naming indicates that there are various images for both the SDRAM slave images and FLASH/ROM slave images. The PB slave supports four SDRAM slave images and four FLASH/ROM slave images. The PB slave supports one register image. Refer to Chapter 16: “ Registers” on page 153 for more information on register access through the PB Interface.

A single dedicated address image can be connected to the general purpose timer trigger, or an interrupt, for debugging purposes. When the selected address appears on the processor (60x) bus the event is logged. This is independent of whether or not the address was destined for PowerPro. See Chapter 9: “Timers” on page 97 and Chapter 11: “Interrupt Controller” on page 105 for more information.

3.2.3.2 Claiming Transactions

The PB slave image claims transactions only under the following circumstances:

- the address falls into the configured slave image
- the transaction type is supported by PowerPro

3.2.3.3 Transaction Types

The PB slave only claims processor (60x) bus transactions with specific transaction types. The supported transaction types consist of address only, read, and write. These transaction types, and their binary codes, are described in Table 1.

PowerPro registers only accept single read cycles and single write cycles.

Address only transfer types are claimed to ensure PowerPro does not negatively impact cache control, reservation or ordering transactions on the processor (60x) bus. PowerPro handles address only cycles by asserting PB_AACK_ — no data transfer occurs.

Table 1: PowerPro PB Slave Transaction Types

PB_TT[0:4]	Name
Address only	
00000	Clean Block
00100	Flush Block
01000	Sync Block
01100	Kill Block
10000	eieio
11000	tlb invalidate
00001	lwarx
01001	tlb sync
01101	icbi
Read	
01010	Read
01110	Read with intent to modify
11010	Read Atomic
11110	Read with intent to modify atomic
01011	Read with no intent to cache
Write	
00010	Write with flush
00110	Write with kill
10010	Write with flush atomic

Since PowerPro does not have a cache, all read and write transfer types are treated the same. For example, a “Read with intent to modify” (PB_TT= 01110) is handled the same way as a “Read Atomic” (PB_TT= 11010).

3.2.3.4 Address Tenure

The processor bus has independent address and data tenures to support pipelined transactions. PowerPro operates in systems with up to one level of address pipelining. PowerPro requires that data tenures be kept in strict order with respect to address tenures.

Each slave in a processor (60x) bus system is responsible for the following actions:

- decoding the address broadcast by the master
- claiming the address tenure with the assertion of Address Acknowledge (PB_AACK_)
- managing the data termination signals during the data tenure

Address Acknowledge (PB_AACK_)

The PB slave uses PB_AACK_ to limit the level of address pipelining to one. The PB slave does not acknowledge subsequent address phases until it finishes its participation in the current data tenure. If the previous address phase was claimed by another slave, the PB slave does not acknowledge the current address phase until the previous slave completes its data tenure.

The earliest time that the PB Slave Interface asserts PB_AACK_ is two clocks after the Processor Bus Transfer Start (PB_TS_) signal is asserted low. This signal indicates the beginning of a new address bus tenure. The PB slave can also delay the assertion of PB_AACK_ in order to wait for external memory devices to access data.

Address Retry (PB_ARTRY_)

The PowerPro PB Slave Interface uses the Address Retry Enable (ARTRY_EN) bit, in the PB_GEN_CTRL register (see page 162), to control its use of PB_ARTRY_ during transactions. By default, the use of address retry is disabled. The PB slave supports a single read at a time when ARTRY_EN is disabled.

PB_ARTRY_ is never asserted in SDRAM accesses, but may be asserted (if enabled) in FLASH/ROM accesses and I2C accesses. When ARTRY_EN is enabled, the PB slave retries a processor (60x) bus master under the following conditions:

- register read during I2C load
- FLASH/ROM read when the transaction will take more than eight clocks
- FLASH/ROM write when the ROM machine is busy

When the assertion of PB_ARTRY_ is enabled, PB_ARTRY_ is only asserted the clock after PB_AACK_. This constraint places the assertion of the signal within the address retry window.

When PB_ARTRY_ is disabled — by setting the ARTRY_EN bit to 0 — the PB slave holds onto the bus after the assertion of PB_AACK_ until it is able to assert Processor Bus Transfer Acknowledge (PB_TA). The PB Slave acknowledges the address tenure with the PB_AACK_ signal and captures the address in the Delayed Read latch. The Delayed Read Request latch is de-allocated when the external processor (60x) bus master completes the transaction.

3.2.3.5 Address Parity

Address parity checking is supplied on each byte of the address bus. Parity is disabled on PowerPro by default.

Address parity bit assignments are defined in Table 2.

Table 2: PowerPro (CA91L750) PB Address Parity Assignments

Address Bus	Address Parity
PB_A[0:7]	PB_AP[0]
PB_A[8:15]	PB_AP[1]
PB_A[16:23]	PB_AP[2]
PB_A[24:31]	PB_AP[3]

When the PB slave detects an address parity error during its decode process it does not assert Address Acknowledge (PB_AACK_). Address Parity checking is enabled when the Address Parity Enable (AP_EN) bit is set in the PB_GEN_CTRL register (see page 162).

Even or odd parity is configured with the Parity (PARITY) bit in the PB_GEN_CTRL register.

3.2.4 Data Phase

3.2.4.1 Data Tenure

The processor (60x) bus protocol has independent address and data tenures to support pipelined transactions. PowerPro operates in systems with up to one level of address pipelining. PowerPro requires that data tenures be kept in strict order with respect to address tenures.

3.2.4.2 Transaction Length

The PB slave supports a super-set of the data transfer sizes supported by the embedded PowerPC family. All data transfer sizes supported by the PB slave are illustrated in Table 3. Burst transfers are indicated by the assertion of Processor Bus Transfer Burst (PB_TBST_). The shaded regions indicate transaction sizes unique to the MPC8260.

Table 3: PowerPro (CA91L750) PB Transfer Sizes

Transfer Size	Bytes	PB_TBST_	PB_TSIZ[0]	PB_TSIZ[1:3]
Byte	1	1	0	001
Half-word	2	1	0	010
Tri-byte	3	1	0	011
Word	4	1	0	100
Five bytes	5	1	0	101
Six bytes	6	1	0	110
Seven bytes	7	1	0	111

Table 3: PowerPro (CA91L750) PB Transfer Sizes

Transfer Size	Bytes	PB_TBST_	PB_TSIZ[0]	PB_TSIZ[1:3]
Double Word (DW)	8	1	0	000
Extended Double (MPC8260 only)	16	1	1	001
Extended Triple (MPC8260 only)	24	1	1	010
Burst (Quad DW)	32	0	0	010

3.2.4.3 Data Alignment

The PowerPro (CA91L750) port size is 64-bit. Embedded processor (60x) bus transfer sizes and alignments, as defined in Table 3 and Table 4, are supported by the PB Interface for transaction accesses. PowerPro register accesses must be 8 bytes or less.

Register accesses are usually restricted to 4 byte (32-bit) accesses. Any accesses larger than 4 bytes normally results in an error condition and the assertion of the Processor Bus Transaction Error Acknowledge (PB_TEA_) signal when PB_TEA_ generation is enabled in PowerPro. PowerPro allows 8 byte register accesses for systems that require 8 byte register reads, but the PB_TEA_ signal must be disabled in the TEA_EN bit, in the PB_GEN_CTRL register (see page 162).

The size of the register access is controlled with the Vector Increment field, in the Interrupt Vector Increment register (see page 214). The VINC[1] bit controls upper-word register access. When PB_TEA_ generation is disabled in PowerPro, an 8 byte register read returns the 4 byte register addressed repeated in the lower and upper word. When VINC[1] is set to 0 and an 8 byte read is performed, the register at offset (PB_REG_ADDR & 0xFF8) is returned (32-bit value) copied in the upper and lower 32-bit words. When VINC[1] is set to 1 and an 8-byte read is performed, the register at offset (PB_REG_ADDR & 0xff8) + 0x004 is returned (32-bit value) replicated in the upper and lower 32-bit words.

For example, an 8 byte read to REG_BADDR + 0x198 when VINC[1] is set to 0 returns the address {INT_VBADDR, INT_VBADDR}. An 8-byte read to REG_BADDR + 0x198 when VINC[1] is set to 1 returns address {INT_MISC, INT_MISC}. The TEA_EN bit, in the PB_GEN_CTRL register (see page 162), must be 0 in this mode. This setting disables the assertion of PB_TEA_ when a register access exceeds 4 bytes. Table 4 lists the size and alignment of transactions less than or equal to 8 bytes.

The information in Table 4 is independent of endian considerations and refers to byte lane control on the PB Interface.

Table 4: PowerPro PB Single Beat Data Transfers

Size	TSIZ[0:3]	A[29:31]	Data Bus Byte Lanes							
			0	1	2	3	4	5	6	7
Byte	0001	000	D0							
	0001	001		D1						
	0001	010			D2					
	0001	011				D3				
	0001	100					D4			
	0001	101						D5		
	0001	110							D6	
	0001	111								D7
Half word	0010	000	D0	D1						
	0010	001		D1	D2					
	0010	010			D2	D3				
	0010	100					D4	D5		
	0010	101						D5	D6	
	0010	110							D6	D7
Tri-byte	0011	000	D0	D1	D2					
	0011	001		D1	D2	D3				
	0011	100					D4	D5	D6	
	0011	101						D5	D6	D7
Word	0100	000	D0	D1	D2	D3				
	0100	100					D4	D5	D6	D7
Five bytes	0101	000	D0	D1	D2	D3	D4			
	0101	011				D3	D4	D5	D6	D7
Six bytes	0110	000	D0	D1	D2	D3	D4	D5		
	0110	010			D2	D3	D4	D5	D6	D7
Seven bytes	0111	000	D0	D1	D2	D3	D4	D5	D6	
	0111	001		D1	D2	D3	D4	D5	D6	D7
Double word	0000	000	D0	D1	D2	D3	D4	D5	D6	D7

3.2.4.4 Cache Line Size

The supported set of embedded PowerPC processors implement a 32-byte cache line size. Cache wrap reads are supported by the PB slave for burst and extended transactions.

PowerPC processors do not generate cache wrap writes.

3.2.4.5 Reads

All PB Slave reads are can be retried if the Address Retry enable (ARTRY_EN) bit is set in the PB_GEN_CTRL register (see Table 39 on page 162).

Retried Reads

An outstanding read is referred to as a retried read. The following steps outline what occurs during a read request:

1. Delayed Read Request: The PB slave latches transaction parameters and issues a retry (PB_ARTRY_).
2. Delayed Read Completion: The PB slave obtains the requested data and completion status on the destination bus.
3. Read Completion: The master repeats the transaction with the same parameters used for the initial request.

Any attempt by a PB master to complete the read transaction is retried by the PowerPro (CA91L750) PB slave when PB_ARTRY_ is enabled.

Discard Timer

The PowerPro PB slave image has a discard timer. When an external master does not claim data within 2^{15} clocks after data is read from the destination bus, the Delayed Read Request latch is de-allocated. This prevents deadlock conditions. Read buffer contents are cleared, but there is no error recorded and no interrupts are generated.

Address Retry (PB_ARTRY_)

For more information refer to “Address Retry (PB_ARTRY_)” on page 41.

3.2.4.6 Writes

Write data is treated independently from read data. A write to an image does not invalidate the contents of the read line buffer currently in use.

3.2.4.7 Data Parity

Data Parity is enabled by setting the Data Parity Enable (DP_EN) bit in the PB_GEN_CTRL register (see page 162). Even or odd parity is enabled by setting the Parity (PARITY) bit in the same register.

Parity generation and checking is provided for each byte of the data bus and for each data beat of the data tenure. Parity checking is disabled by default.

Data parity bit assignments are defined in Table 5.

Table 5: PowerPro PB Data Parity Assignments

Data Bus	Data Parity
PB_D[0:7]	PB_DP[0]
PB_D[8:15]	PB_DP[1]
PB_D[16:23]	PB_DP[2]
PB_D[24:31]	PB_DP[3]
PB_D[32:39]	PB_DP[4]
PB_D[40:47]	PB_DP[5]
PB_D[48:55]	PB_DP[6]
PB_D[56:63]	PB_DP[7]

The data parity bits, PB_DP[0:7], are driven to the correct values for even or odd parity by the PB slave during reads. When checking is enabled, the data parity bits, PB_DP[0:7], are checked by the PB slave during writes.

The detection of a data parity error does not affect the transaction. Data is still forwarded to the destination. For more information on PB Interface errors refer to Chapter 10: “Error Handling” on page 101.

3.2.5 Termination

The PB Interface uses the following signals to indicate termination of individual data beats and/or data tenures:

- PB_ARTRY_: This signal terminates the entire data tenure and schedules the transaction to be retried. No data is transferred, even if the signal is asserted at the same time as PB_TA_ or Processor Bus data Valid (PB_DVAL_).
- PB_TA_: This signal is asserted by the PB slave to indicate the successful transfer of a single beat transaction, or each 8-byte quantity transferred for a burst.
- PB_DVAL_: This signal is asserted by the PB slave to indicate the successful transfer of an 8 byte quantity within an extended transfer of 16 or 24 bytes. PB_TA_ is asserted with PB_DVAL_ on the transfer of the last 8 byte quantity. The PB slave does not use PB_TA_, or PB_DVAL_ to insert wait states.
- PB_TEA_: This signal indicates an unrecoverable error and causes the external master to immediately terminate the data tenure.

The PB Slave does not assert a data termination signal earlier than the *address retry window.*

3.2.5.1 Assertion of PB_TEA_

The assertion of PB_TEA_ can be enabled or disabled through the Transaction Error Acknowledge (TEA_EN) bit in the PB_GEN_CTRL register (see page 162).

In a development environment, the TEA_EN bit can be set in order to enable the assertion of PB_TEA_ to assist with software debugging. In a production environment, it can be useful to disable the TEA_EN bit. When PB_TEA_ is disabled, the PB_TA_ signal is asserted in its place.

When PB_TEA_ is disabled, incorrect data can be transferred in a system.

When a particular slave image cannot handle transactions involving more than 4 bytes, PowerPro asserts PB_TEA_. An example of the assertion of PB_TEA_ for unsupported reads is FLASH/ROM accesses that exceed 4 bytes.

3.2.5.2 Errors

The PowerPro PB Slave Interface detects the following error conditions:

- address parity
- data parity on writes
- illegal accesses

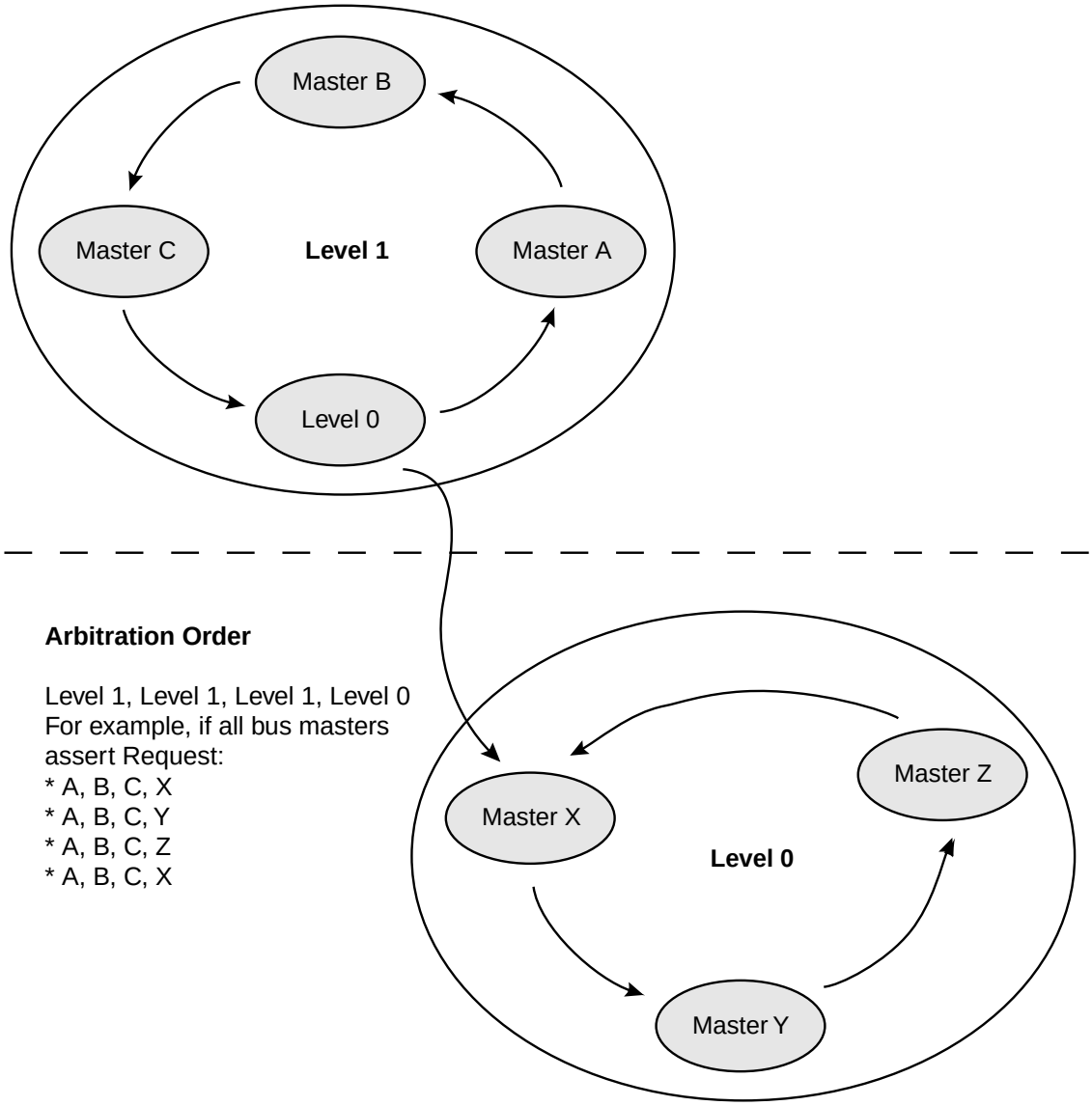
See Chapter 10: “Error Handling” on page 101 and Chapter 11: “Interrupt Controller” on page 105 for more information.

3.3 Processor Bus Interface Arbitration

The PowerPro PB arbiter supports requests for address bus and data bus ownership from four external masters. The PB arbiter is enabled or disabled through the External Master x Enable (Mx_EN) bit in the Processor Bus Arbiter Control (PB_ARB_CTRL) register (see page 164). The ability to disable the internal PowerPro arbiter makes it possible to use an external arbiter in a PowerPro system.

Figure 5 illustrates the arbitration system used in the PB Interface arbiter.

Figure 5: PB Arbitration Order



The PB arbiter implements two levels of priority. Devices programmed into a specific priority level operate in a round robin fashion. Each master has a External Master x Priority Level (Mx_PRI) bit in the PB_ARB_CTRL register to determine its arbitration level for the address bus. The arbitration level for each master can be reconfigured during system run-time.

3.3.1 Data Bus Arbitration

The arbiter samples PB_TT[3] when PB_TS_ is asserted to generate data bus requests. The arbiter grants the data bus to the current address bus owner by asserting PB_DBG_ the clock after PB_TS_ is asserted. The current data bus grant is negated when the requesting master has qualified the grant.

The Processor Bus Data Bus Grant ($\overline{\text{PB\_DBG}}$) signals are asserted under the following conditions:

- The assertion of the Processor Bus Request (PB_REQ_) signal when the bus is idle.
- After the assertion of the Processor Bus Transaction Start (PB_TS_) signal, the PB_DBG_ signal changes to the next requesting master or the parked master.

Requesting masters are required to qualify bus grants before beginning a data tenure.

The MPC8260 and other 60x bus agents require the system signal DBB_ to qualify data bus grants. The PowerPro PB Master does not require DBB_ to qualify data bus grants.

3.3.2 Address Arbitration

Requesting bus masters are required to qualify bus grants before beginning an address tenure. Some processor (60x) bus agents, including the MPC8260, require the system signal ABB_ to qualify address bus grants. The PowerPro PB Interface does not require ABB_ to qualify address bus grants.

3.3.2.1 Address-Only Cycles

The arbiter supports address only cycles. When PB_TT[3] is sampled low during PB_TS_, the arbiter does not grant the data bus.

The use of PB_TT[3] as a data bus request means that the PowerPro arbiter does not support the PowerPC instructions eciwx and ecowx.

3.3.3 Bus Parking

The PowerPro PB arbiter offers a flexible address bus parking scheme. When there is no master requesting the address bus, the arbiter can park on either the last bus master or a specific bus master.

The bus parking mode is determined by the Bus Park Mode (PARK) bit, in the Processor Bus Arbiter Control (PB_ARB_CTRL) register (see page 164). To park the bus on a specific master, the Bus Master to be Parked (BM_PARK) field in the PB_ARB_CTRL register must be set to the specific bus master for address parking.

Table 6 shows the codes for parking external bus masters.

Table 6: Parked PB Master

BM_PARK [1:0]	Parked PB Master	External Pins
00	M0	PB_BR[0]_/PB_BG[0]_
01	M1	PB_BR[1]_/PB_BG[1]_
10	M2	PB_BR[2]_/PB_BG[2]_
11	M3	PB_BR[3]_/PB_BG[3]_

The parked master does not drive any address bus signals until it generates a request to use the address bus.

3.4 Endian Conversion

PowerPro does not perform any endian conversion. PowerPC big-endian bit ordering is assumed for all transactions.

Chapter 4: SDRAM Interface

This chapter outlines the functionality of the SDRAM Interface. The topics addressed in this chapter include:

4.1 Overview

PowerPro supports many different types of Synchronous DRAM (SDRAM) memory. Both discrete memory modules and Dual Inline Memory Modules (DIMM) can be used in a PowerPro system.

SDRAM operating frequency is dependent on bus loading. Refer to Appendix A: "Typical Applications" on page 241 for more information.

PowerPro accesses a maximum of four DIMMs of SDRAM memory. PowerPro supports both buffered (registered) and non-buffered DIMMs. A DIMM can be either single or dual. Each SDRAM has two chip selects per DIMM, and supports four logical memory banks for a single DIMM and eight logical banks for a dual DIMM.

Buffered (registered) DIMMs place a register (buffer) between PowerPro's command/address line and the DIMM's command/address line.

4.2 Supported SDRAM Configurations

Each DIMM of SDRAM must be connected to a DIMM socket. There are a maximum of four logical memory banks per side of each DIMM. Each DIMM can be individually configured through the SDRAM registers. The ability to individual configure the memory means that each DIMM socket can contain different types of memory. The different memory configuration can be optimized for the timing requirements.

PowerPro has a separate controller for each bank of SDRAM supported within a DIMM. The separate controllers manage the memory attached to it. The controllers internally arbitrate for the SDRAM command bus, and chip-wide data bus. Each separate controller snoops the addresses of the processor for transactions that fall within the SDRAM address range. Refer to Chapter 16: “ Registers” on page 153 for more information on SDRAM address images.

Snooping a transaction refers to monitoring addresses driven by a bus master.

4.3 SDRAM Operation

Before any READ or WRITE commands can be issued to a bank within the SDRAM, a row in that bank must be opened. The ACTIVE command is used to open the bank and select both the bank and the row to be activated.

After opening a row - issuing an ACTIVE command - a READ or WRITE command can be used for that row of the SDRAM bank, subject to the tRCD operation specification. tRCD (min) must be divided by the clock period and rounded-up to the next whole number to determine the earliest clock edge after the ACTIVE command on which a READ or WRITE command can be entered.

A subsequent ACTIVE command to a different row in the same bank can only be issued after the previous active row has been closed. Closing a bank is accomplished with the PRECHARGE command. The minimum time interval between successive ACTIVE commands to the same bank is defined by tRC.

A subsequent ACTIVE command to another bank can be issued while the first bank is being accessed. This functionality results in a reduction of total row access overhead. The minimum time interval between successive ACTIVE commands to different banks is defined by tRRD.

4.4 Registers

The SDRAM Interface has one control register, and four bank specific registers.

The SDRAM registers include the following:

- “SDRAM Refresh Interval” on page 171: This register determines the refresh interval for SDRAM in units of processor bus clocks (PB_CLKs)

- "SDRAM Timing Parameters" on page 172: This register controls the setting of bits that dictate timing parameters for the device.
- "SDRAM Memory Bank x Address" on page 175: This register maps the four available SDRAM banks to the 60x address space.
- "SDRAM Memory Bank x Address Mask" on page 176: This register indicates the memory block size of the SDRAM bank address.
- "SDRAM Memory Bank x Control and Status" on page 178: This register contains the bits that control specific SDRAM bank functionality, such as ECC protection.

4.5 Initialization

The SDRAM Interface must be configured through the PowerPro Processor Bus Interface before it can be used in the system.

The following sequence is used to initialize the SDRAM Interface:

1. Determine the parameters that must be programmed in the memory module registers.
   a. If SDRAM DIMMs are being used in the system, the following action is taken:
   — Access the primary I^2C Interface and read the serial presence detect information from each SDRAM DIMM.
   b. If discrete memory modules (non-DIMMs) are being used the following action is taken:
   — If no serial presence detect is present on the primary I^2C Interface, the stored values in the SDRAM_CTRL register are used for the banks that are present. When the I^2C information is present, the I^2C information is used to program the SDRAM_CTRL register.
2. Program the Base Address (A[0:15]) field and Enable (ENABLE) bit in the SDRAM Memory Bank X Address (SD_Bx_ADDR) register (see page 175) and the Mask (M[0:15]) bit in the SDRAM Memory Bank X Address Mask (SD_Bx_MASK) register (see page 176). These registers must be programmed for all SDRAM banks used in the system.

All SDRAM memory images must be aligned to the size of the memory being used. For example, a 64-Mbyte device must be on a 64-Mbyte boundary, a 256-Mbyte device on a 256-Mbyte boundary. This relationship continues until the maximum device width is reached. Refer to Chapter 16: " Registers" on page 153 for more information.

3. Program the SDRAM Memory Bank X Control and Status (SD_Bx_CTRL) register (see page 178) with the appropriate parameters for the physical type of memory being used in the system. These parameters are determined from either the specification sheets for the memory itself, or from the primary I^2C Interface serial presence detect information.

The bits and fields which must be set in the SD_Bx_CTRL register include:

- T_RCD: This bit specifies the timing delay between the Activate (ACTV) command and the READ/WRITE command. This bit can be used to optimize SDRAM performance.
- T_RP: This field specifies the timing delay between the Precharge (PRE) command to the ACTV command. This field can be used to optimize SDRAM performance.
- T_RAS: This field specifies the timing delay between the ACTV command and the PRE command. This field can be used to optimize SDRAM performance.
- Number of Banks (NBANK): This bit selects the number of banks.
- Buffered SDRAM DIMM select (BUF): This field tells PowerPro if buffered or unbuffered SDRAM DIMMs are in the bank.
- Address Mapping Mode (A_MODE): This bit maps the address of the number of columns PowerPro is supporting.
- ECC Checking and Correction Enable (ECC_EN): This bit enables ECC checking and correction.
- ECC Correction Mode (ECC_CE): When this bit is enabled, PowerPro corrects all correctable errors.

Banks can be individually enabled for ECC protection. However, enabling ECC protection has no effect if the system has the Data Quality Mask Enable (DQM_EN) bit, in the SDRAM Timing Parameters (SD_TIMING) register (see page 172), set to 1. Selecting ECC protection on banks which do not have physical ECC memories attached results in continuous multi-bit ECC errors. All banks must have the ECC enable bit set to 1.

4. Program the SD_TIMING register.

   The bits and fields which must be set in the SD_TIMING register include:

   - Datapath Tune (TUNE): This setting is determined by the board layout, pin loading, and clock frequency. Refer to “SDRAM Datapath Tuning” on page 55 for clarification.
   - External Datapath (EX_DP): This bit must be set to connect the SDRAM data bus to the processor (60x) data bus.
   - CAS Latency (CL): This is a global parameter and is set to all the SDRAMs simultaneously.
   - SDRAM Timing Parameter (T_RC): This setting is determined from SDRAM specification and should be taken as the worst-case value — the longest delay — for all of the attached SDRAMs.

5. Ensure the minimum amount of time has elapsed from PORESET_. This time is dependent on the SDRAM memory specifications.

6. Set the Enable (ENABLE) bit, in the SD_TIMING register, to 1. This setting forces a power-on reset sequence to the SDRAMs. This sequence consists of the following:

   — precharge-all command

   — eight auto-refresh commands

   — mode register set command

   — eight auto-refresh commands

   Refer to “Commands” on page 56 for more information on SDRAM commands.

The worst case timing from all of the four possible enabled banks must be used during this sequence.

7. The SDRAMs are initialized.

4.5.1 SDRAM Datapath Tuning

High system performance is the result of the least amount of latency between the processor (60x) bus requesting memory, and that request being fulfilled. However, a heavily loaded and fast system is not able to meet timing criteria without pipeline stages inserted in the datapath. The TUNE bits in the SDRAM Timing register (see page 172) control the number of pipeline stages inserted in the datapath.

By pipelining the datapath, system performance and timing is enhanced.

If ECC correction is enabled on any bank, by default two pipeline stages are inserted in the data path. If ECC correction is not used on any memory bank, the following TUNE settings are available:

- TUNE = 00: no pipeline stages.

  If EX_DP = 0, then data flows from SDRAM to the processor (60x) bus through PowerPro in the same clock, with PowerPro acting as a data buffer. If EX_DP = 1, then it is assumed that the SDRAM data bus is connected directly to the 60x data bus; in this case the PowerPro will drives DVAL/TA but not D[0:63].

- TUNE = 01: one pipeline stage on the output data path.

  One stage of pipeline is inserted on the outgoing data path. For processor (60x) bus t0 SDRAM transfers (writes), the data to the SDRAM comes directly off of an internal PowerPro register. For SDRAM to processor (60x) bus transfers (reads), the data to the processor (60x) bus comes directly off of an internal PowerPro register.

- TUNE = 10: one pipeline stage on the input data path
- TUNE = 11: two pipeline stages

  One piplined stage is on the input path and one piplined stage is on the output path.

4.6 Commands

PowerPro issues commands specific to the SDRAM devices by encoding them on the following signals:

- SD_CS[0:3]_
- SD_CS[4:7]_
- SD_RAS_
- SD_CAS_
- SD_WE_

Table 7: SDRAM Commands

Name	Description	SD_CS_	SD_RAS_	SD_CAS_	SD_WE_
ACTIVE	Select bank and activate row	Low	Low	High	High
READ	Select bank and column and start read burst	Low	High	Low	High
WRITE	Select bank and column and start write burst	Low	High	Low	Low
PRECHARGE	De-activate row in bank or banks	Low	Low	High	Low
AUTO REFRESH	Refresh both banks from a on-chip refresh counter	Low	Low	Low	High

4.6.1 Standard SDRAM Commands

4.6.1.1 ACTIVE (ACTV)

The active command is used to open (or activate) a row in a particular bank for a subsequent access. This row remains active (or open) for accesses until a PRECHARGE command is issued to that bank. A PRECHARGE command must be issued before opening a different row.

4.6.1.2 READ

The READ command is used to initiate a burst read access to an active row.

4.6.1.3 WRITE

The WRITE command is used to initiate a burst write access to an active row

4.6.1.4 PRECHARGE

The PRECHARGE command is used to deactivate the open row in a particular bank or the open row in all banks. The bank(s) will be available for a subsequent row access a specified time (tRP) after the PRECHARGE command is issued. Once a bank has been precharged, it is in idle state and must be activated prior to any READ or WRITE commands being issued to that bank.

4.6.1.5 AUTO REFRESH

AUTOREFRESH is used during normal operation of the SDRAM and is similar to CAS_-BEFORE-RAS_ REFRESH in conventional DRAMs. This command is non-persistent, so it must be issued each time a refresh is required

4.6.2 Supported SDRAM Commands

Table 8 lists all of the SDRAM commands used by SDRAM devices. PowerPro supports a sub-set of these commands. The shaded areas of the table indicate commands that are not supported in PowerPro.

Table 8: SDRAM Commands

Command	Description
NOP	No operation
Mode Register Set	Load the Mode Register from SD_A[12:0]
Row Activate	Activate a row specified on SD_A[12:0]
Read	Column burst read
Read with Auto-Precharge	Column burst read with row precharge at the end of the transfer
Write	Column burst write
Write with Auto-Precharge	Column burst write with row precharge at the end of the transfer
Precharge	Precharge a single leaf
Precharge All	Precharge both leaves
Auto-Refresh	Refresh both banks from a on-chip refresh counter
Self-Refresh	Refresh autonomously
Power Down	Power down if both banks are pre-charged
Stop	Interrupt a read or write burst

4.7 Transactions

The SDRAM Interface supports both read transactions and write transactions to memory.

Cache wrap reads occur for burst and extended transactions to SDRAM. The supported embedded processors, MPC8260, PowerPC 740 and PowerPC 750, do not generate cache wrap writes.

4.7.1 Reads

There are three different scenarios which affect PowerPro SDRAM reads. The following categories list different areas that impact reads on PowerPro:

- The desired bank is open at the required address
- The desired bank is open, but at the incorrect address
- The desired bank is closed

These categories are summarized in the following sections.

4.7.1.1 Bank Open at the Required Address

Read performance is optimized for SDRAM page hits. Refer to “Page Mode” on page 62 for more information on page hits.

A page hit occurs when the current address falls within a row that is currently open. PowerPro follows these steps on a page hit on a read:

1. PowerPro decodes the transaction to determine if the transaction should be claimed. PowerPro latches the address, size and type of the transaction.
   - If the address falls in the SDRAM address range indicated by Base Address (A[0:15]) field, in the SDRAM Memory Base X Address (SD_Bx_ADDR) register, PowerPro claims the transaction.

The memory space allotted must be aligned with the block size of the attached memory. For example, a 128 Mbyte DIMM can only be mapped to addresses on a 128 Mbyte boundary.

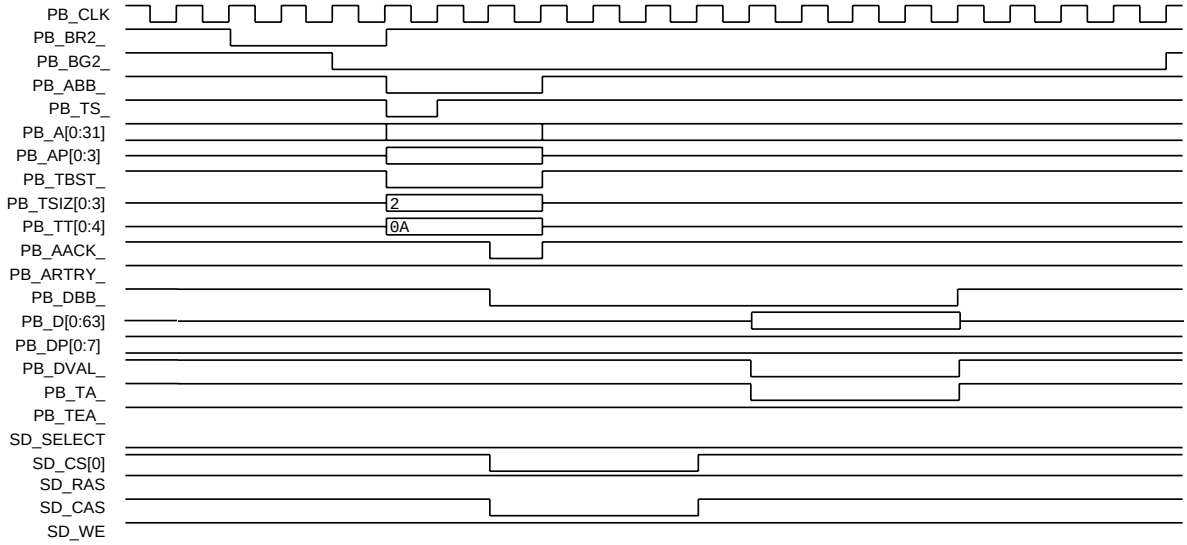
   - PowerPro determines whether or not the open pages are hit. If the open pages are hit the SDRAM Interface activates the appropriate bank by asserting its chip select for the next cycle.
2. In the next cycle PowerPro asserts SD_CAS_ and de-asserts SD_WE_, then places the column address on the requested address. This initiates the burst read cycle.
3. After the SD_CAS_ latency expires, the SDRAM device drives the data to PowerPro.
4. When the data is received, PowerPro initiates ECC protection (if enabled) and compares it against the data returned from SDRAM. Refer to Chapter 10: “Error Handling” on page 101 for more ECC protection information.

5. Assuming ECC protection indicates no error in the data, PowerPro drives the data to the processor (60x) bus.

Figure 6 shows a burst read when the bank is open to the required address. The Bank Management (BMGT) field is set to 0 in the SDRAM Memory Bank x Control and Status (page 178). When the BMGT field is set to 0, the memory bank is left open until a miss or refresh occurs. If the BMGT field is set to 0xF, the memory bank is closed after each access. The timing of this transaction is not effected if the BMGT field is set to 0 or 1.

Figure 6 through Figure 9 are meant to clarify information. However, they are based on generic information and are for example purposes only. They do not constitute accurate timing diagrams.

Figure 6: Burst Read with Memory Bank Open



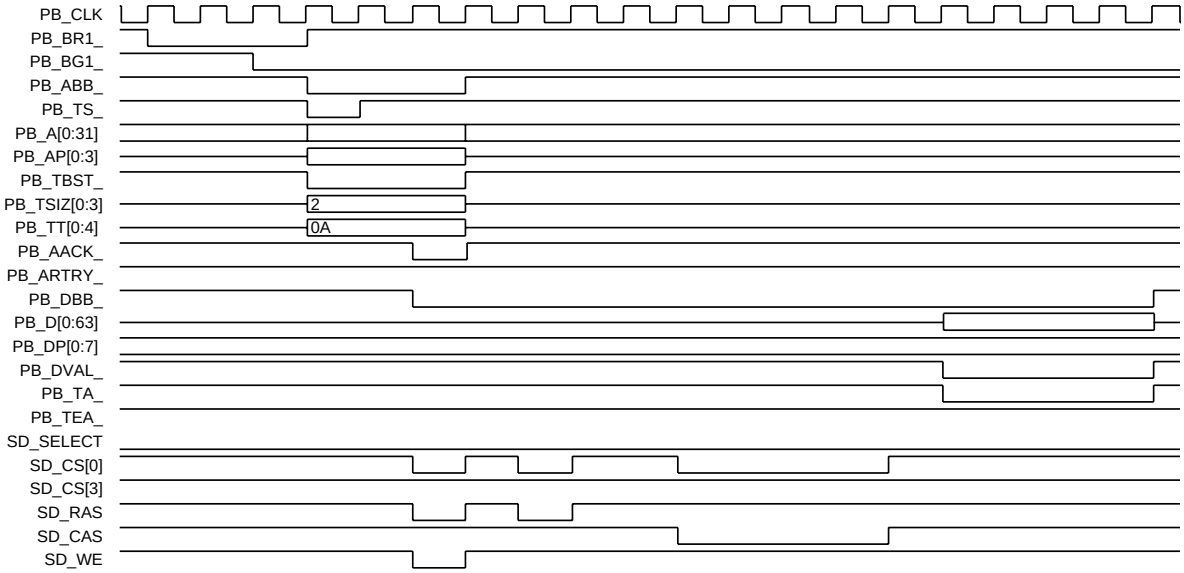
4.7.1.2 Bank Open at the Incorrect Address

In order to access data on a read when the required bank is detected as open, but is at the incorrect address, PowerPro follows these steps:

1. Close the current bank with a Precharge (PRE) command
2. Open the new bank with an Activate (ACTV) command
3. Issue a Read (READ) command
4. After the SD_CAS_ latency expires, the SDRAM device drives the data to PowerPro.
5. Assuming ECC protection indicates no error in the data, PowerPro drives the data to the processor (60x) bus.

Figure 7 shows a burst read when the bank is open to the required address. The Bank Management (BMGT) field is set to 0 in the SDRAM Memory Bank x Control and Status (page 178). When the BMGT field is set to 0, the memory bank is left open until a miss or refresh occurs. If the BMGT field is set to 0xF, the memory bank is closed after each access. The timing of this transaction is not effected if the BMGT field is set to 0 or 1.

Figure 7: Single Read with a Bank Open at the Incorrect Address



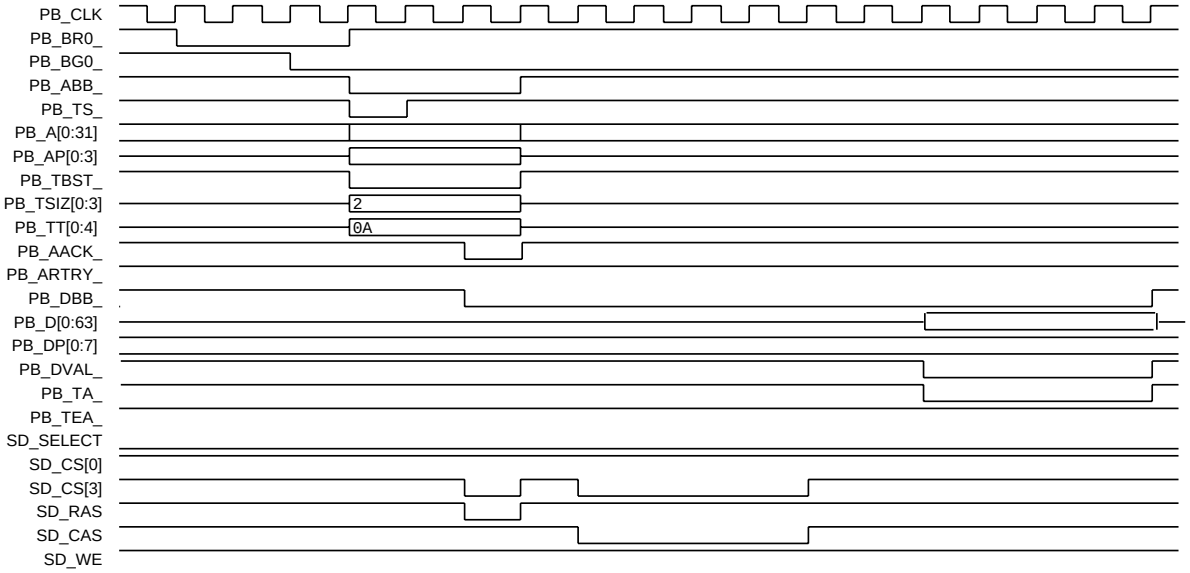
4.7.1.3 Bank Closed

In order to access data on a read when the required bank is closed, PowerPro follows these steps:

1. Assert the ACTV command to the SDRAM
2. Assert a READ command.
3. After the SD_CAS_ latency expires, the SDRAM device drives the data to PowerPro.
4. Assuming ECC protection indicates no error in the data, PowerPro drives the data to the processor (60x) bus.

Figure 8 shows a burst read when the bank is open to the required address. The Bank Management (BMGT) field is set to 0 in the SDRAM Memory Bank x Control and Status (page 178). When the BMGT field is set to 0, the memory bank is left open until a miss or refresh occurs. If the BMGT field is set to 0xF, the memory bank is closed after each access.

Figure 8: Burst Read with Memory Bank Closed



4.7.2 Writes

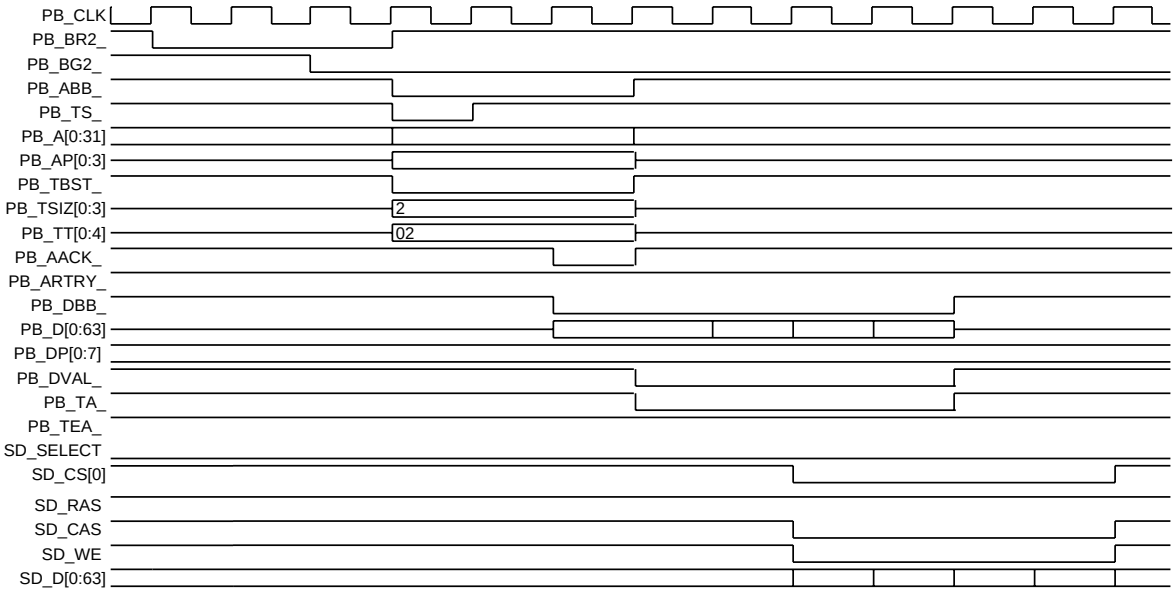
PowerPro writes are fully pipelined. From the processor (60x) bus perspective, PowerPro accepts four data writes after PB_AACK_ is asserted.

PowerPro follows these steps on a write:

1. PowerPro decodes the transaction to determine if the transaction must be claimed. PowerPro latches the address, size and type of the transaction.
   - If the address falls in the SDRAM address range indicated by Base Address (A[0:15]) field, in the SDRAM Memory Base X Address (SD_Bx_ADDR) register, PowerPro claims the transaction.
   - PowerPro determines whether or not the open pages are hit. If the open pages are hit then the SDRAM Interface activates the appropriate bank by asserting its chip select for the next cycle.
2. ECC protection (if enabled) generates the ECC code for the data to be written.
3. In the next cycle PowerPro asserts SD_CAS_ and asserts SD_WE_, then places the column address on the requested address. PowerPro drives the data to be written and its ECC code to the SDRAM devices.
4. PowerPro drives the new data on the data bus each cycle until the transaction is complete.

Figure 9 shows a single write when the bank is open to the required address. The Bank Management (BMGT) field is set to 0 in the SDRAM Memory Bank x Control and Status (page 178). When the BMGT field is set to 0, the memory bank is left open until a miss or refresh occurs. If the BMGT field is set to 0xF, the memory bank is closed after each access. The timing of this transaction is not effected if the BMGT field is set to 0 or 1.

Figure 9: Burst Write With Memory Bank Open



4.8 Page Mode

The SDRAM page mode size is 4 Kbyte. The SDRAM Interface supports back-to-back page mode. Pipelined accesses to system memory that lie in the same bank are treated as a continues burst as long as the addressed page is open.

The Bank Management (BMGT) bits, in the SDRAM Memory Bank X Control and Status (SD_Bx_CTRL) register, determine if pages are left open after access, or closed immediately with an auto-precharge command during the last read or write. A BMGT bit is assigned to each SDRAM. Pages left open by setting the BMGT bit to 0, remain open until an access to that page results in a miss or a Refresh command is driven to memory.

4.9 Refresh

SDRAM is dynamic memory and must be periodically refreshed. A refresh cycle is issued for the SDRAM memory. All enabled banks of SDRAM are refreshed simultaneously at the expiry of the refresh timer. The refresh interval is programmed in the SDRAM Refresh Interval (T[0:15]) field, in the SDRAM Refresh Interval (SD_REFRESH) register (see page 171).

SDRAM needs to be refreshed thousands of times per second or it looses its stored data.

The refresh interval for SDRAM is in units of processor bus clocks. The programmed number of clocks are counted, and at the end of the interval a refresh command is sent to all SDRAM banks. The register must be set to an appropriate interval for the memory attached and the processor (60x) bus clock frequency used in the system. If different memory devices are used in different banks, a refresh time appropriate to the worst case device — requiring the most frequent refresh intervals — must be assigned to all SDRAM devices.

The PowerPro refresh counter is designed to minimize the disruption of other memory accesses.

PowerPro collects a maximum of 16 refreshes while waiting for an inactive time to assert the refresh command. If, by 16 refresh periods no inactive time is detected, the refresh counter allows the current transaction to finish then takes priority over any pending transactions. The next transaction is overridden by refresh. Once the refresh starts, all pending refreshes — up to the maximum of 16 — are performed.

During a refresh command, all SDRAM and FLASH/ROM accesses are either delayed or retried until the refresh cycle completes.

4.10 ECC Protection

ECC protection can be enabled on the SDRAM Interface. Refer to Chapter 10: “Error Handling” on page 101 for information on ECC protection.

4.11 Endian Conversion

The PowerPC platform is big-endian. In a big-endian environment A[0] refers to the most significant bit. PowerPro is designed for a PowerPC system and uses the big-endian notation wherever possible. However, many available memories are specified with little-endian notation. All available SDRAM DIMMs and modules use A[0] as the least significant bit. However, PowerPro does not perform any endian conversion.

The PowerPro SDRAM and FLASH/ROM address connections, illustrated in Table 9 and Table 10, are labeled as little-endian to minimize confusion in connecting bits together. The tables map exactly what processor (60x) bus address pin connect to what SDRAM address pin in all modes. The general rule for connection is connect like-to-like when attaching SDRAM and FLASH/ROMs to PowerPro.

PowerPro does not perform any endian conversion. It is a big-endian device.

Table 9 outlines the SDRAM address map to the processor (60x) bus. The A_Mode refers to the A_Mode field in the SDRAM Memory Bank x Control register (see page 178).

4.12 Address Mapping

The information in this section focuses on software information required to program PowerPro. It is software specific information.

Address mapping occurs between the processor (60x) bus and SDRAM memory. The following parameters must be programmed in order for proper address mapping to occur:

- The address from the processor (60x) bus map to the SDRAM chip select must be programmed in the A[0:15] field. This programming is based on the Address Mode (A_MODE) bit, in the SD_Bx_CTRL register.
- The size of the memory installed must be programmed through the SD_Bx_MASK register.
- The number of physical and logical banks must be programmed through the Number of Banks (NBANK) bit in the SD_Bx_CTRL register.

Table 9 and Table 10 illustrate the mapping of processor bus address to physical SDRAM address.

Table 9: SDRAM Address to Processor (60x) Bus Mapping[a]

SDRAM Address (SD_A)	PB_AA_Mode 0	PB_AA_Mode 1	PB_AA_Mode 2	PB_AA_Mode 3	PB_AA_Mode 4	All Modes
	Row	Row	Row	Row	Row	Column
SD_A[12]	PB_A8	PB_A7	PB_A6	PB_A5	PB_A4	PB_A17
SD_A[11]	PB_A9	PB_A8	PB_A7	PB_A6	PB_A5	PB_A18
SD_A[10]/ AP	PB_A10	PB_A9	PB_A8	PB_A7	PB_A6	PB_AP
SD_A[9]	PB_A11	PB_A10	PB_A9	PB_A8	PB_A7	PB_A19
SD_A[8]	PB_A12	PB_A11	PB_A10	PB_A9	PB_A8	PB_A20
SD_A[7]	PB_A13	PB_A12	PB_A11	PB_A10	PB_A9	PB_A21
SD_A[6]	PB_A14	PB_A13	PB_A12	PB_A11	PB_A10	PB_A22
SD_A[5]	PB_A15	PB_A14	PB_A13	PB_A12	PB_A11	PB_A23
SD_A[4]	PB_A16	PB_A15	PB_A14	PB_A13	PB_A12	PB_A24
SD_A[3]	PB_A17	PB_A16	PB_A15	PB_A14	PB_A13	PB_A25
SD_A[2]	PB_A18	PB_A17	PB_A16	PB_A15	PB_A14	PB_A26
SD_A[1]	PB_A19	PB_A18	PB_A17	PB_A16	PB_A15	PB_A27
SD_A[0]	PB_A20	PB_A19	PB_A18	PB_A17	PB_A16	PB_A28

a. AP = 1 when Auto-Precharge command issued. At other times AP= 0.

The relationship between the processor (60x) bus address and the SDRAM chip selects and bank address pins (SD_BA[1:0]) involves the NBANK bit in the SD_Bx_CTRL register. This relationship is shown in Table 11. Table 11 illustrates the mapping between processor (60x) bus addresses and SD_A[12:0] for the row and column phases of an SDRAM access. The Mask (M) bit, in the SD_Bx_MASK register, indicates the size of the memory block (decode).

Table 10: SDRAM Chip Select and Bank Mapping

PB Address (PB_A) (see Table 11)	NBANK Setting			
	00	01	10	11
1	SD_BA[0]	SD_BA[0]	CS[x] or CS[x+1]	CS[x] or CS[x+1]
2	-	SD_BA[1]	SD_BA[0]	SD_BA[0]
3	-	-	-	SD_BA[1]

Table 11: Processor Bus to SDRAM Address Mapping

PB_A	A_Mode field setting in the SD_Bx_CTRL register					Mask (M) field setting in the SD_Bx_MASK register											
	A_ Mode 0[a]	A_Mode 1	A_Mode 2	A_Mode 3	A_Mode 4	0X800[b]	0xC00	0xE00	0xF00	0xF80	0xFC0	0xFE0	0xFF0	0xFF8	0xFFC	0xFFE	0xFFF
0						-[c]	-	-	-	-	-	-	-	-	-	-	-
1						1[d]	c	-	-	-	-	-	-	-	-	-	-
2						2	1	-	-	-	-	-	-	-	-	-	-
3						3	2	1	-	-	-	-	-	-	-	-	-
4					R-12		3	2	1	-	-	-	-	-	-	-	-
5				R-12	R-11			3	2	1	-	-	-	-	-	-	-
6			R-12	R-11	R-10				3	2	1	-	-	-	-	-	-
7		R-12	R-11	R-10	R-9					3	2	1	-	-	-	-	-
8	R-12	R-11	R-10	R-9	R-8						3	2	1	-	-	-	-
9	R-11	R-10	R-9	R-8	R-7							3	2	1	-	-	-
10	R-10	R-9	R-8	R-7	R-6								3	2	1	-	-
11	R-9	R-8	R-7	R-6	R-5									3	2	1	-
12	R-8	R-7	R-6	R-5	R-4										3	2	1
13	R-7	R-6	R-5	R-4	R-3											3	2
14	R-6	R-5	R-4	R-3	R-2												3
15	R-5	R-4	R-3	R-2	R-1												
16	R-4	R-3	R-2	R-1	R-0												

Table 11: Processor Bus to SDRAM Address Mapping

PB_A	A_Mode field setting in the SD_Bx_CTRL register					Mask (M) field setting in the SD_Bx_MASK register											
	A_ Mode 0[a]	A_Mode 1	A_Mode 2	A_Mode 3	A_Mode 4	0X800[b]	0xC00	0xE00	0xF00	0xF80	0xFC0	0xFE0	0xFF0	0xFF8	0xFFC	0xFFE	0xFFF
17	R-3	R-2	R-1	R-0	C-12												
18	R-2	R-1	R-0	C-11	C-11												
19	R-1	R-0	C-9	C-9	C-9												
20	R-0	C-8	C-8	C-8	C-8												
21	C-7	C-7	C-7	C-7	C-7												
22	C-6	C-6	C-6	C-6	C-6												
23	C-5	C-5	C-5	C-5	C-5												
24	C-4	C-4	C-4	C-4	C-4												
25	C-3	C-3	C-3	C-3	C-3												
26	C-2	C-2	C-2	C-2	C-2												
27	C-1	C-1	C-1	C-1	C-1												
28	C-0	C-0	C-0	C-0	C-0												
29																	
30																	
31																	

a. The A_Mode table heading indicates the setting in the A_Mode field of the SD_Bx_CTRL register.
b. The hexadecimal value in this table heading represents the setting in the Mask (M) bit, in the SD_Bx_MASK register.
c. This table value represents no value.
d. This table value represents the setting in the NBANK bit of the SD_Bx_CTRL register.

Chapter 5: FLASH/ROM Interface

This chapter discusses the functions of the FLASH/ROM Interface. The topics addressed in this chapter include:

5.1 Overview

PowerPro supports four distinct banks of FLASH/ROM devices. The interface enables many multiplexing options to support a variety of address and data bus requirements. Each of the four independent FLASH/ROM banks have individually programmable address images. FLASH/ROM devices of 8-, 16-, 32-, and 64-bit data widths can be attached to the SDRAM data bus. Alternatively, a separate 8-bit data bus can be used either to attach a FLASH/ROM data path or as extra address lines

Each FLASH/ROM bank has an individually programmable image with unique address ranges, bus widths, addressing modes and timing parameters. The programmable images have separate parameters and an internal separate machine to drive the image. There are four chip selects; each of the four chip-select machines arbitrates for required resources.

When this document discusses FLASH/ROM transactions it is referring to all ROM-like devices with asynchronous interfaces including FLASH, EEPROM, and SRAM.

5.2 FLASH/ROM Signals

FLASH/ROM memory space is separate from SDRAM memory space. The FLASH/ROM chip selects activate the appropriate FLASH/ROM bank when the address falls within one of the FLASH/ROM address ranges.

Table 12 shows the FLASH/ROM Interface signals.

Table 12: FLASH/ROM Interface Signals

Pin Name	Pin Type	Description
SD_A[12:0] Multiplexed with: EE_A[12:0]	Output	1. SDRAM Address 2. EEPROM Address
SD_BA[1:0] Multiplexed with: EE_A[14:13]	Output	1. SDRAM Bank Address 2. EEPROM Address
SD_SELECT Multiplexed with: EE_SELECT/ EE_AL[0]/ GPIO[23]	Tristate bidirectional	1. SDRAM Bank Select: External FET switch 2. EEPROM Buffer Select 4. EEPROM Address Latch 0: For time-multiplexing the EEPROM address, the first address phase is latched qualified with this signal 3. General Purpose I/O
EE_AL1_ Multiplexed with: GPIO[24]	Tristate bidirectional	1. EEPROM Address Latch 1: For time-multiplexing the EEPROM address, the second address phase is latched qualified with this signal 2. General Purpose I/O.
EE_OE_ Multiplexed with: GPIO[25]	Tristate bidirectional	1. External Memory Output Enable 2. General Purpose I/O
$\overline{\text{EE\_WE\_}}$ Multiplexed with: GPIO[26]	Tristate bidirectional	1. External Memory Write Enable 2. General Purpose I/O

Table 12: FLASH/ROM Interface Signals

Pin Name	Pin Type	Description
EE_RNW Multiplexed with: GPIO[27]	Tristate bidirectional	1. EEPROM Read not Write: Active 1 during an EEPROM read, 0 at all other times. 2. General Purpose I/O
EE_AL2 Multiplexed with: GPIO[28]_	Tristate bidirectional	1. EEPROM Address Latch **2**: For time-multiplexing the EEPROM address, the third address phase is to be latched qualified with this signal. 2. General Purpose I/O
EE_READY Multiplexed with: GPIO[29]	Tristate bidirectional	1. External Memory Ready Input Indicator: Tells PowerPro when FLASH is ready on the data bus so PowerPro knows when to sample it. 2. General Purpose I/O
EE_CS[0:3]_ Multiplexed with: GPIO[30:33]	Tristate bidirectional	1. External Memory Chip Select: One per bank. 2. General Purpose I/O
EE_DATA[0:7] Multiplexed with: EE_A[23:15]/ EE_A[31:24]/ EE_A[23:16]/ EE_A[15:8]/ EE_A[7:0]/ INT[0:7]/ GPIO[34:41]/ PWRUP[0:7]	Tristate bidirectional	1. ROM Data [0:7] 2. ROM upper (MSB) address bits [23:15] 3. ROM address bits [31:24] (time-multiplexed). 4. ROM address bits [23:16] (time-multiplexed). 5. ROM address bits[15:8] (time-multiplexed). 6. ROM address bits[7:0] (time-multiplexed). 7. Interrupt inputs[0:7] 8. General Purpose I/O 7. Power-Up Options: Only latched during power-on reset
UART0_TX Multiplexed with: INT[8]/ GPIO[42]	Tristate bidirectional	1. Primary UART Transmit Line 2. Interrupt Controller Input 3. General Purpose I/O
UART0_RX Multiplexed with: INT[9]/ GPIO[43]	Tristate bidirectional	1. Primary UART Receive Line 2. Interrupt Controller Input 3. General Purpose I/O

All of the GPIO signal pins can also be configured as General Purpose I/O (GPIO) pins for applications where the generic FLASH/ROM Interface controller does not perform the function desired in the end application. GPIO programming overrides normal ROM function.

When configuring GPIO pins it is possible to disable the FLASH/ROM Interface. By enabling GPIO functionality, the multiplexed pins that are dedicated to the FLASH/ROM Interface can be disabled.

5.2.1 Time-Multiplexed SDRAM Signals

The following signals are primarily SDRAM signals, but depending on the FLASH/ROM mode the signals can be time-multiplexed with the FLASH/ROM Interface.

Table 13: Memory Signals

Signal Name	Signal Type	Description
SD_A[12:0] Multiplexed with: EE_A[12:0]	Output	SD_BA[1:0] and SD_A[12:0] are always output as FLASH/ROM address [14:0], and can be time multiplexed as FLASH/ROM address [29:15].
SD_BA[1:0] Multiplexed with: EE_A[14:13]	Output	
SD_ECC[0:7] Multiplexed with: SD_DQM[0:7]	Tristate bidirectional	Always output as FLASH/ROM byte enables during FLASH/ROM cycles. No useful information is presented when accessing a 8-bit wide device. A 16-bit wide device can use SD_ECC/DQM[0:1] as byte enables. For a 32-bit device, byte enables appear on SD_ECC/DQM[0:3], and a 64-bit devices uses all of SD_ECC/DQM[0:7] for byte enables. In all cases, a 1 on these lines indicates that the particular byte should be written, while a 0 indicates that the byte should not be written (it must be masked).
SD_D[0:63]	Tristate bidirectional	An 8-bit wide FLASH/ROM can have its data port connected to SD_D[0:7], a 16-bit wide ROM must use SD_D[0:15], a 32-bit wide FLASH/ROM must use SD_D[0:31], and a 64-bit wide FLASH/ROM must use SD_D[0:63].
SD_SELECT Multiplexed with: EE_SELECT/ EE_AL[0]/ GPIO[23]	Tristate bidirectional	This signal is used in heavily loaded SDRAM configurations to control a FET switch connected between data lines on bank 0/1 and bank 2/3. This signal toggles and can be used for selecting a FET switch, or buffer, between FLASH/ROM or SDRAM accesses. This is required when the SD_D port is shared between the FLASH/ROM and the SDRAM devices. This signal is high when an access is made to SDRAM bank 2/3. The signal is low when an access is made to SDRAM bank 0/1 or FLASH/ROM devices. This signal is also used as EE_AL[0] during FLASH/ROM Address Mode 11. When used as EE_AL[0], it must be connected to an external latch to hold FLASH/ROM address [0:7].

5.2.2 Time-Multiplexed Processor Bus Signals

The following signals are primarily Processor Bus Interface signals, but depending on the FLASH/ROM mode the signals can be time-multiplexed with the FLASH/ROM Interface.

Table 14: PB Signals

Pin Name	Pin Type	Description
PB_DP[0:7] Multiplexed with: PB_DBG[2:3]_/ EE_A[28:23]/ GPIO[4:11]	Tristate bidirectional	When data parity is not used on the processor (60x) bus, these pins can be connected to FLASH/ROM address [28:23], [29:24], or [20:15] depending on the FLASH/ROM configuration selected.
PB_BR[0:1]_ Multiplexed with: EE_A[28:27] _/ GPIO[12:13]_	Tristate bidirectional	When the PB Interface arbiter is not used, these pins can be connected to FLASH/ROM address [28:23], [29:24], or [20:15] depending on the FLASH/ROM configuration selected.
PB_BG[0:1]_ Multiplexed with: EE_A[26:25]_/ GPIO[14:15]_	Tristate bidirectional	
PB_DBG[0:1] Multiplexed with: INT[20:21]_ / EE_A[24:23]_ GPIO[16:17]_	Tristate bidirectional	

5.3 Data Port

The FLASH/ROM Interface can be connected to two different data ports: SD_D[0:63] (time-shared with SDRAM devices) or EE_DATA[0:7] (dedicated 8-bit port). Up to 32-bits of FLASH/ROM address are available.

Table 15 summarizes the four available FLASH/ROM addressing modes. Table 15 and Table 16 illustrate the address depending on the ROM addressing mode selected.

Signals PB_DP, PB_BR_, PB_BG_, and PB_DBG_ (see Table 14) have primary functions other than providing the FLASH/ROM address. When the on-board processor (60x) bus arbiter is used, PB_BR_, PB_BG_, and PB_DBG_ are not available for FLASH/ROM addresses. When data parity is enabled on the processor (60x) bus PB_DP[2:7] are not available for FLASH/ROM addresses.

5.4 Address Mapping

The address and block size of each FLASH/ROM image are programmable through the ROM Memory Bank X Address (EE_Bx_ADDR) register and the ROM Memory Bank X Address Mask (EE_Bx_MASK) register. Refer Chapter 16: “ Registers” on page 153 for a description of these registers.

5.4.1 Multiplexed Address Signals

The least significant FLASH/ROM address lines (EE_A[14:0]) are shared with the SDRAM lines SD_A[12:0], SD_BA[1:0] for fifteen lines.

The most significant address lines appear in two places. First, if the power-up option for processor (60x) bus parity is programmed to disable parity, and GPIO[4:11] ports are disabled, then PB_DP[0:7] is configured as an output to EE_A[20:15]. Second, if INT[0:7] are masked and GPIO[34:41] are disabled, then EE_A[20:15] are output.

When the FLASH/ROM image is configured to use the port as a data input, then the port is turned around on read cycles coinciding with the assertion of EE_OE_. On write cycles the port changes from address to write data with the assertion of EE_WE_. The signal EE_DATA can be used as a multiplexed Most Significant Bit (MSB) address and data port by capturing MSB addresses in an external latch tied to EE_CS_, EE_OE_, or EE_WE_. This configuration is dependent on the programming of the EE_Bx_CTRL register (see page 185).

The two most significant address bits, EE_A[21] and EE_ADDR[22] are shared with UART #0. If UART #0 is disabled, INT[8] and INT[9] are masked and the corresponding GPIO ports are disabled, then the Most Significant Bit (MSB) EEPROM addresses are output on these lines.

Multiplexing FLASH/ROMs on SDRAM lines can overload the SDRAM lines and cause the SDRAM Interface to be unable to operate at the required frequency. The exact board layout and attachment of FLASH/ROMs depends on frequency requirements and loading.

The following bullets outline different configurations and address possibilities for the 8-, 16-, 32-, and 64-bit FLASH/ROM devices.

5.4.1.1 Connection summary

- 8-bit devices
  - FLASH/ROM DATA[0:7] can occupy:
    - EE_DATA[0:7]: dedicated FLASH/ROM 8-bit data bus
    - SD_D[0:7]: a section of the SDRAM data bus
- 16-bit devices
  - FLASH/ROM DATA[0:15] can only be connected to:
    - SD_D[0:15]: a section of the SDRAM data bus
- 32-bit devices
  - FLASH/ROM DATA[0:31] can only be connected to:
    - SD_D[0:31]: a section of the SDRAM data bus
- 64-bit devices
  - FLASH/ROM DATA[0:63] can only be connected to:
    - SD_D[0:63]: a section of the SDRAM data bus
- EE_A[14:0] are time-shared with SD_A[12:0] and SD_BA[1:0].
- ROM ADDRESS[28:23], [29:24], or [20:15] appear on PB_DP[2:7] if PB parity is disabled and GPIO[6:11] are disabled.
- ROM ADDRESS[28:23], [29:24], or [20:15] appear on {PB_BR[0:1], PB_BG[0:1], PB_DBG[0:1]} if the internal PB arbiter is disabled.
- ROM ADDRESS[22:15], [31:24], [23:16], [15:8], and [7:0] can appear on EE_DATA[0:7] depending on the FLASH/ROM address mode. If EE_DATA is used as a data port as well as in the cycle, the address only appears at the beginning of the cycle. If EE_DATA is used, then an external latch is needed to hold the MSB address, assuming these bits are needed to connect to the FLASH/ROM device.

Table 15 shows how the FLASH/ROM addresses are mapped in PowerPro. The Ref column in the table refers to the information in Table 16 on page 78.

Table 15: FLASH/ROM Address Mapping

Address Mode	Data Port	Ref	Control Pins	SD_A[14:0]	EE_DATA[0:7]	Extra[0:5]
00	0	A	<none>	EE_A[14:0]	<data>	EE_A[20:15]
	1	B	<none>	EE_A[14:0]	EE_A[22:15]	EE_A[28:23]
01	0	C	<none>	EE_A[14:0]	<data>	EE_A[20:15]
		D	EE_AL1	EE_A[29:15]	EE_A[15:8]	EE_A[20:15]
	1	E	<none>	EE_A[14:0]	EE_A[22:15]	EE_A[28:23]
		F	EE_AL1	EE_A[29:15]	EE_A[15:8]	EE_A[28:23]
10	0	G	<none>	EE_A[14:0]	<data>	EE_A[29:24]
		H	EE_AL1	EE_A[29:15]	EE_A[15:8]	EE_A[29:24]
		I	EE_AL2	EE_A[14:0]	EE_A[23:16]	EE_A[29:24]
	1	J	<none>	EE_A[14:0]	EE_A[7:0]	EE_A[29:24]
		K	EE_AL1	EE_A[29:15]	EE_A[15:8]	EE_A[29:24]
		L	EE_AL2	EE_A[14:0]	EE_A[23:16]	EE_A[29:24]
11	0	M	<none>	EE_A[14:0]	<data>	EE_A[29:24]
		N	EE_AL1	EE_A[29:15]	EE_A[15:8]	EE_A[29:24]
		O	EE_AL2	EE_A[14:0]	EE_A[23:16]	EE_A[29:24]
		P	SD_SELECT	EE_A[14:0]	EE_A[7:0]	EE_A[29:24]
	1	Q	<none>	EE_A[14:0]	EE_A[7:0]	EE_A[29:24]
		R	EE_AL1	EE_A[29:15]	EE_A[15:8]	EE_A[29:24]
		S	EE_AL2	EE_A[14:0]	EE_A[23:16]	EE_A[29:24]
		T	SD_SELECT	EE_A[14:0]	EE_A[31:24]	EE_A[29:24]

Table 16 shows how PowerPro pins interact with FLASH/ROM addresses. The Ref column in Table 15 refers to the information in Table 16.

Table 16: PowerPro Pin to FLASH/ROM Address Mapping

Pin	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
PB_BR[0]	20	28	20	20	28	28	29	29	29	29	29	29	29	29	29	29	29	29	29	29
PB_BR[1]	19	27	19	19	27	27	28	28	28	28	28	28	28	28	28	28	28	28	28	28
PB_BG[0]	18	26	18	18	26	26	27	27	27	27	27	27	27	27	27	27	27	27	27	27
PB_BG[1]	17	25	17	17	25	25	26	26	26	26	26	26	26	26	26	26	26	26	26	26
PB_ DBG[0]	16	24	16	16	24	24	25	25	25	25	25	25	25	25	25	25	25	25	25	25
PB_ DBG[1]	15	23	15	15	23	23	24	24	24	24	24	24	24	24	24	24	24	24	24	24
EE_ DATA[0]	-d-	22	-d-	15	22	15	-d-	15	23	7	15	23	-d-	15	23	7	7	15	23	31
EE_ DATA[1]	-d-	21	-d-	14	21	14	-d-	14	22	6	14	22	-d-	14	22	6	6	14	22	30
EE_ DATA[2]	-d-	20	-d-	13	20	13	-d-	13	21	5	13	21	-d-	13	21	5	5	13	21	29
EE_ DATA[3]	-d-	19	-d-	12	19	12	-d-	12	20	4	12	20	-d-	12	20	4	4	12	20	28
EE_ DATA[4]	-d-	18	-d-	11	18	11	-d-	11	19	3	11	19	-d-	11	19	3	3	11	19	27
EE_ DATA[5]	-d-	17	-d-	10	17	10	-d-	10	18	2	10	18	-d-	10	18	2	2	10	18	26
EE_ DATA[6]	-d-	16	-d-	9	16	9	-d-	9	17	1	9	17	-d-	9	17	1	1	9	17	25
EE_ DATA[7]	-d-	15	-d-	8	15	8	-d-	8	16	0	8	16	-d-	8	16	0	0	8	16	24
SD_BA[1]	14	14	14	29	14	29	14	29	14	14	29	14	14	29	14	14	14	29	14	14
SD_BA[0]	13	13	13	28	13	28	13	28	13	13	28	13	13	28	13	13	13	28	13	13
SD_A[12]	12	12	12	27	12	27	12	27	12	12	27	12	12	27	12	12	12	27	12	12
SD_A[11]	11	11	11	26	11	26	11	26	11	11	26	11	11	26	11	11	11	26	11	11
SD_A[10]	10	10	10	25	10	25	10	25	10	10	25	10	10	25	10	10	10	25	10	10
SD_A[9]	9	9	9	24	9	24	9	24	9	9	24	9	9	24	9	9	9	24	9	9
SD_A[8]	8	8	8	23	8	23	8	23	8	8	23	8	8	23	8	8	8	23	8	8
SD_A[7]	7	7	7	22	7	22	7	22	7	7	22	7	7	22	7	7	7	22	7	7
SD_A[6]	6	6	6	21	6	21	6	21	6	6	21	6	6	21	6	6	6	21	6	6
SD_A[5]	5	5	5	20	5	20	5	20	5	5	20	5	5	20	5	5	5	20	5	5

Table 16: PowerPro Pin to FLASH/ROM Address Mapping

Pin	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
SD_A[4]	4	4	4	19	4	19	4	19	4	4	19	4	4	19	4	4	4	19	4	4
SD_A[3]	3	3	3	18	3	18	3	18	3	3	18	3	3	18	3	3	3	18	3	3
SD_A[2]	2	2	2	17	2	17	2	17	2	2	17	2	2	17	2	2	2	17	2	2
SD_A[1]	1	1	1	16	1	16	1	16	1	1	16	1	1	16	1	1	1	16	1	1
SD_A[0]	0	0	0	15	0	15	0	15	0	0	15	0	0	15	0	0	0	15	0	0

5.5 Transactions

5.5.1 Processor Bus Transactions

The FLASH/ROM Interface supports all valid processor (60x) bus transactions. All returned data is internally buffered in PowerPro before it is returned to the requesting master on the processor (60x) bus.

When PB_ARTRY_ is enabled (through the ARTRY Enable (ARTRY_EN) bit), the requesting master is retried until PowerPro has gathered all the requested data. Writes to FLASH/ROM devices are buffered within PowerPro.

Access to SDRAM through PowerPro is closed if PowerPro is processing a FLASH/ROM read or write transaction. Access to the SDRAM through PowerPro is not closed for the processor (60x) bus while PowerPro is processing a FLASH/ROM write, but it is closed during a FLASH/ROM read if PB_ARTRY_ is not enabled.

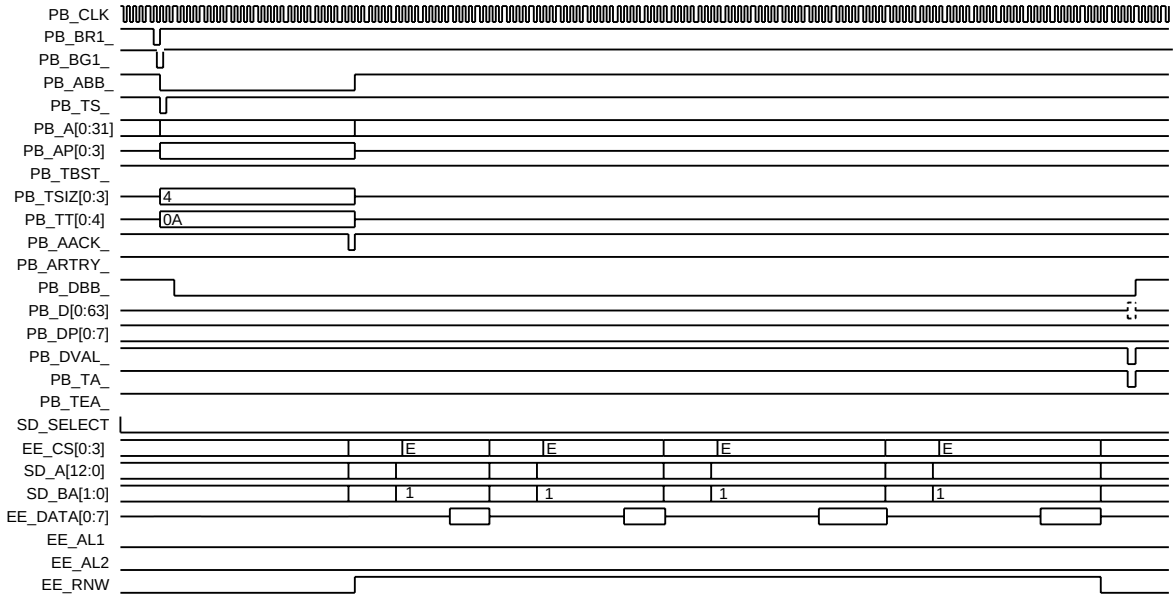
5.5.2 Reads

Reading a FLASH/ROM device involves driving the address, EE_OE_ and EE_CS_. The return interval of the data depends on the speed of the FLASH/ROM device.

The Read Not Write (EE_RNW) signal is asserted during a FLASH/ROM read cycle, and EE_WE_ is asserted during a FLASH/ROM write cycle. Both of these signals are used to enable external logic to be selected during FLASH/ROM read and write cycles.

Figure 10 shows a 32-bit FLASH write to an 8-bit port.

Figure 10: FLASH Read



5.5.2.1 Wait States

The definition of address-to-data wait states are the number of cycles between the assertion of EE_CS_ and the arrival of data from the FLASH/ROM device on the EE_A_ address signals. The definition of recovery wait states are the number of cycles between the arrival of data on EE_A_ and the address for the next FLASH/ROM transaction.

Address-to-data wait states are programmed in the First Wait (FWT) bit in the EE_Bx_CTRL register (see page 185).

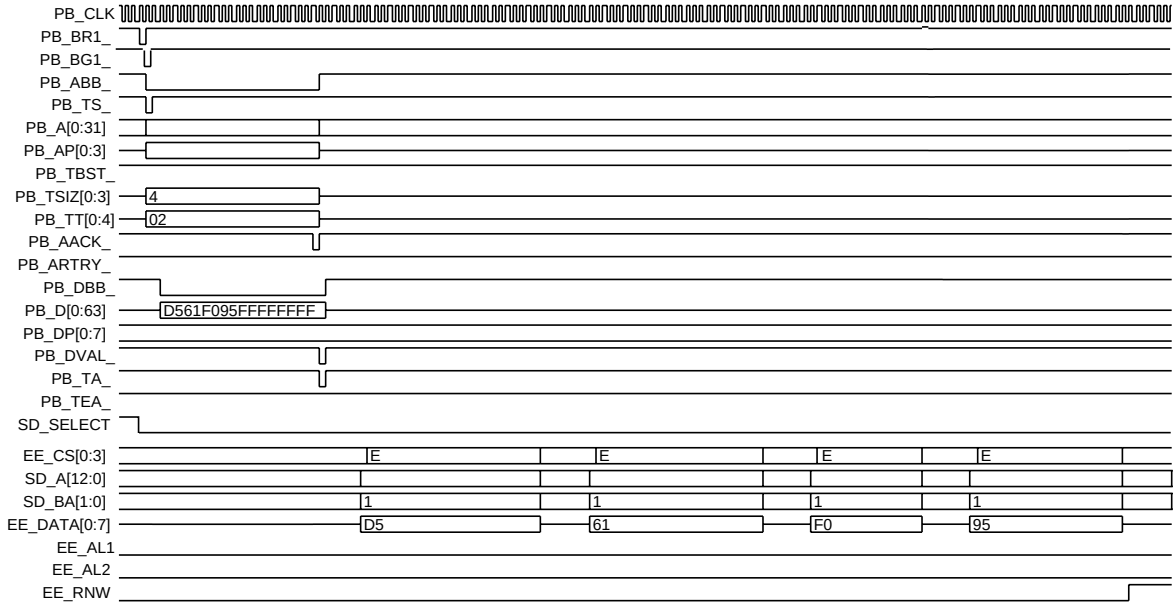
5.5.3 Writes

Address-to-data and recovery wait states for reads and writes are identical and are programmed in the First Wait (FWT) bit in the EE_Bx_CTRL register (see page 185).

The signal EE_RNW is asserted during an FLASH/ROM read cycle; EE_WE is asserted during a FLASH/ROM write cycle. Both of these signals are used to enable external logic to be selected during FLASH/ROM read and write cycles.

Figure 11 shows a 32-bit FLASH write to an 8-bit port.

Figure 11: FLASH Write



5.6 Connecting FLASH/ROM to PowerPro

Each of the four FLASH/ROM chip-select machines controls one of EE_CS[0:3] signals. These signals, plus EE_OE_, and EE_WE_ (where appropriate) are directly connected to their equivalent signals on the FLASH/ROM device.

Devices which have a READY output to control data selection can connect to EE_READY. This enables PowerPro to utilize READY. The signal EE_SELECT_ is active throughout the cycle and should be connected to external transceivers to remove load from the shared SDRAM data and address buses.

5.6.1 Typical Configurations

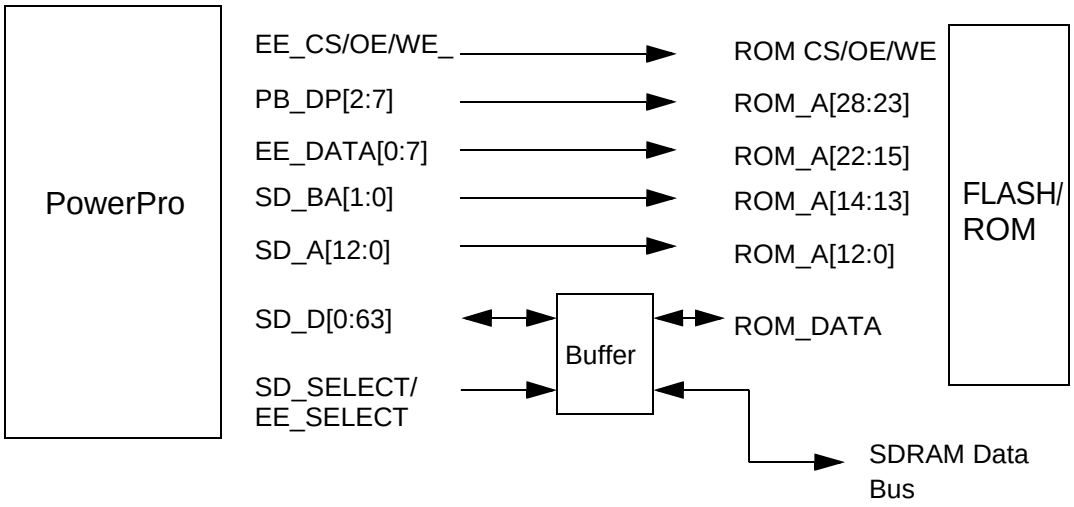
Figures 12-7 illustrate options for connecting a FLASH/ROM device to PowerPro. These diagrams show potential system configurations.

5.6.1.1 Configuration One

The following items outline the configuration displayed in Figure 12:

- The PORT bit, in the ROM Memory Bank X Control (see page 185), is set to 1. Setting this bit to 1 selects the SDRAM data bus as the FLASH/ROM data port.
- The MUX bit, in the ROM Bank Address register (see page 181), is set to 00. Setting this bit to 000 defines the address mapping between the SDRAM Interface and the PB Interface.
- Data parity in the Processor bus Interface is disabled, by setting the Data parity enable (DP_EN) bit, in the Processor Bus General Control register (see page 162) to 0.
- SD_SELECT signal (see page 134) is used to control an external buffer to off-load the FLASH/ROM from the SDRAM data bus.

Figure 12: Configuration One

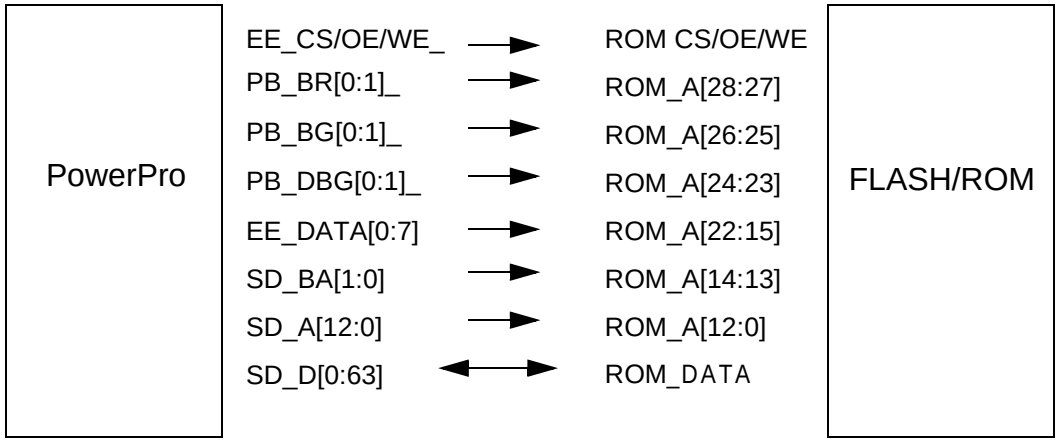


5.6.1.2 Configuration Two

The following items outline the configuration displayed in Figure 13:

- The PORT bit, in the ROM Memory Bank X Control (see page 185), is set to 1. Setting this bit to 1 selects the SDRAM data bus as the FLASH/ROM data port.
- The MUX bit, in the ROM Bank Address register (see page 181), is set to 00. Setting this bit to 000 defines the address mapping between the SDRAM Interface and the PB Interface.
- The internal PowerPro arbiter is disabled, by setting the External Master Enable (Mx_EN) bit, in the Processor Bus Arbiter Control register (see page 164), to 0.

Figure 13: Configuration Two

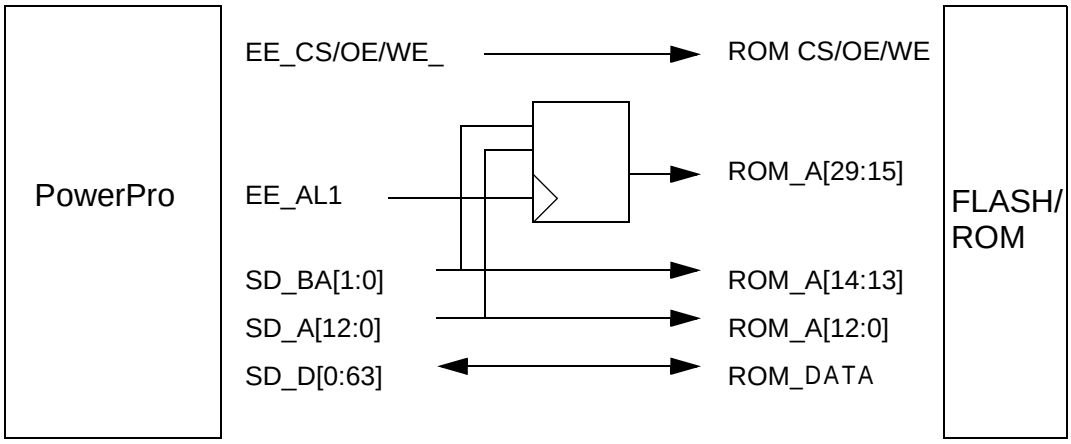


5.6.1.3 Configuration Three

The following items outline the configuration displayed in Figure 14:

- The PORT bit, in the ROM Memory Bank X Control (see page 185), is set to 1. Setting this bit to 1 selects the SDRAM data bus as the FLASH/ROM data port.
- The MUX bit, in the ROM Bank Address register (see page 181, is set to 10. Setting this bit to 010 defines the address mapping between the SDRAM Interface and the PB Interface.
- EE_DATA used for GPIO or interrupts
- The address is latched with positive-edge triggered external latch

Figure 14: Configuration Three

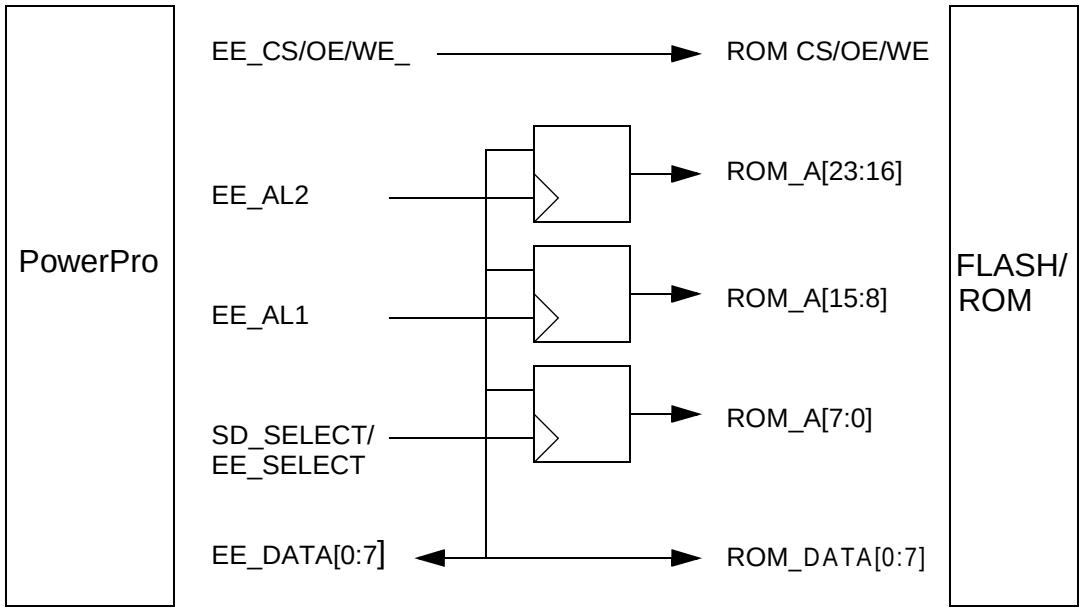


5.6.1.4 Configuration Four

The following items outline the configuration displayed in Figure 15.

- The PORT bit, in the ROM Memory Bank X Control (see page 185), is set to 0. Setting this bit to 1 selects the SDRAM data bus as the FLASH/ROM data port.
- The MUX bit, in the ROM Bank Address register (see page 181), is set to 11. Setting this bit to 11 ensures all three parts of the address appear on EE_DATA[0:7] lines.
- All connections made using EE_DATA, as extra loading (in this case), cannot be tolerated on the SDRAM bus.
- The address is latched by external positive-edge triggered latch

Figure 15: Configuration Four



Chapter 6: Dual UART Interface

This chapter outlines the functionality of PowerPro's dual UARTs. The topics addressed in this chapter include:

6.1 Overview

PowerPro has two serial interfaces which use the UART protocol for communication. The UARTs perform serial-to-parallel conversion on data characters received from a peripheral device, and parallel-to-serial conversion on data characters received from the processor. The processor can read the complete status of the UART at any time during the functional operation. Status information reported includes the type and condition of the transfer operations being performed by the UART, as well as any error conditions, for example parity, overrun, framing, or break interrupt.

The UARTs include a receive and transmit control, and a user programmable processor interrupt system. Interrupts can be programmed to the user's requirements, minimizing the computing required to handle the communications link.

6.2 Registers

UART registers are accessed through the processor. PowerPro UART operations, including transmission and reception, are controlled by the following registers:

- UARTx Receive/Transmit Data (UARTx_RX_TX) register
- UARTx Interrupt Enable (UARTx_IER) register

- UARTx Interrupt Status / FIFO Control (UARTx_ISTAT_FIFO) register

The UARTx_ISTAT_FIFO register has different functionality depending on whether the register is read or if it is written. Table 17 on page 89 shows the register in its read only state and its write only state. In the “ Registers” on page 153 the register bits are explained first in the right only state (Table 96 on page 222) and then in their write only state (Table 98 on page 225).

- UARTx Line Control (UARTx_LCR) register
- UARTx Modem Control (UARTx_RT) register
- UARTx Line Status (UARTx_LSR) register
- UARTx Modem Status (UARTx_MSR) register

- UARTx Scrachpad (UARTx_SCR) register

Table 17: Summary of UART Register

Bit	Register Address											
	0 DLAB=0	0 DLAB=0	1 DLAB=0	2	2	3	4	5	6	7	0 DLAB=1	1 DLAB=1
	UARTx_ RX_TX (read only)	UARTx_ RX_TX (write only)	UARTx_ IER	UARTx_ ISTAT_ FIFO (read only)	UARTx_ ISTAT_ FIFO (write only)	UARTx_ LCR	UARTx_ RT	UARTx_ LSR	UARTx_ MSR	UARTx_ SCR	Divisor Latch	Divisor Latch
	RX	TX	IER	ISTAT	FIFO	LCR	MCR	LSR	MSR	SCR	DLM	DLL
0	Data Bit 0[a]	Data Bit 0	Enable Received Data Available Interrupt (ERBFI)	0 if interrupt is pending	FIFO Enable	Word Length Select Bit (0) (WLEN)	Reserved	Data Ready (DR)	Reserved	Bit 0	Bit 8	Bit 0
1	Data Bit 1	Data Bit 1	Enable Transmitter Holding Register Empty Interrupt (ETBEI)	Interrupt ID Bit (0)	RCVR FIFO Reset	Word Length Select Bit (1) (WLEN)	Reserved	Overrun Error (OE)	Reserved	Bit 1	Bit 9	Bit 1
2	Data Bit 2	Data Bit 2	Enable Receiver Line Status Interrupt (ELSI)	Interrupt ID Bit (1)	XMIT FIFO Reset	Number of Stop Bits (STB)	Reserved	Parity error (PE)	Reserved	Bit 2	Bit 10	Bit 2

Table 17: Summary of UART Register

Bit	Register Address											
	0 DLAB=0	0 DLAB=0	1 DLAB=0	2	2	3	4	5	6	7	0 DLAB=1	1 DLAB=1
	UARTx_ RX_TX (read only)	UARTx_ RX_TX (write only)	UARTx_ IER	UARTx_ ISTAT_ FIFO (read only)	UARTx_ ISTAT_ FIFO (write only)	UARTx_ LCR	UARTx_ RT	UARTx_ LSR	UARTx_ MSR	UARTx_ SCR	Divisor Latch	Divisor Latch
	RX	TX	IER	ISTAT	FIFO	LCR	MCR	LSR	MSR	SCR	DLM	DLL
3	Data Bit 3	Data Bit 3	0	Interrupt ID Bit (2)	Reserved	Parity Enabled (PEN)	Reserved	Framing error (FE)	Reserved	Bit 3	Bit 11	Bit 3
4	Data Bit 4	Data Bit 4	0	0	Reserved	Even Parity Select (EPS)	Reserved	Break Interrupt (BI)	Reserved	Bit 4	Bit 12	Bit 4
5	Data Bit 5	Data Bit 5	0	0	Reserved	Stick Parity (SP)	Reserved	Transmitter Holding Register (THRE)	Reserved	Bit 5	Bit 13	Bit 5
6	Data Bit 6	Data Bit 6	0	FIFO Enabled	RCVR Trigger (LSB)	Set Break (SB)	Reserved	Transmitter Empty (TEMT)	Reserved	Bit 6	Bit 14	Bit 6
7	Data Bit 7	Data Bit 7	0	FIFO Enabled	RCVR Trigger (LSB)	Divisor Latch Access Bit (DLAB)	Reserved	FIFO in RCVR FIFO	Reserved	Bit 7	Bit 15	Bit 7

a. Bit 0 is always the least significant bit. It is the first to be serially transmitted or received.

6.2.1 Receive/Transmit Data Register

The UARTx_RX_TX register enables the PowerPro to receive read data or send write data. Refer to UARTx Receive/Transmit Data register (page 216) for more information.

When the Divisor Latch Access (DLAB) bit is set, writes to UART offset 0x0 and 0x1 set the baud rate divisor. When the DLAB bit is set to 1, the UARTx_RX_TX and UARTx_IER registers are not accessible. The register space is shared between the UARTx_DLM register and the UARTx_Rx_Tx at offset 1B0 (Table 92 on page 216 and Table 93 on page 218) and the UART1_DLL register and the UARTx_IER register (Table 94 on page 219 and Table 95 on page 221) at offset 1B1.

Writing 1 to the DLAB bit enables the UARTx_DLM or UARTx_DLL (depending on reads and writes) and disables the other registers that share their register offset.

6.2.2 Interrupt Enable Register

The UARTx_IER register enables UART interrupts. Each interrupt can individually activate an interrupt output signal. The UART interrupts that can be enabled include:

- Enable Received Data Available Interrupt: Activated by setting the ERBFI bit.
- Enable Transmitter Holding Register Empty Interrupt: Activated by setting the ETBEI bit.
- Enable Receiver Line Status Interrupt: Activated by setting the ELSI bit.

Refer to the UARTx Interrupt Enable register (page 219) for more information.

6.2.3 Interrupt Status and FIFO Control Register

The UARTx_ISTAT_FIFO register is used for providing interrupt status information to the processor, as well as performing FIFO control operations. FIFO operations include setting the receiver FIFO trigger levels and enabling the FIFOs. Refer to UARTx Interrupt Status/FIFO control register (page 222) for more information.

6.2.4 Line Control Register

The UARTx_LCR register specifies the format of the asynchronous data communications exchange and enables the Divisor Latch Access bit. This register enables parity, and even parity. Refer to the UARTx Line Control register (page 227) for more information.

6.2.5 Modem Control Register

The UARTx_RT register functionality is not supported in PowerPro.

6.2.6 Line Status Register

The UARTx_LSR registers provides status information to the processor concerning the data transfer. It logs parity error, receiver FIFO errors and framing errors. This register is intended for read operations. Refer to the UARTx Line Status register (page 230) for more information.

6.2.7 Modem Status Register

The UARTx_MSR register functionality is not supported in PowerPro.

6.2.8 Scratchpad Register

The UARTx_SCR is an 8-bit read and write register. This register does not control UART operation, but is designed to hold temporary data.

6.3 Clocking

The UARTs run synchronously with the processor bus clock (PB_CLK). The UARTs do not contain their own clock generator or crystal input. The baud rate selection is unique to PowerPro.

The PowerPro UARTs are controlled by the Baud Rate Divisor Latches (B) bit in the UARTx _DLM or the UARTx_DLL registers (see page 218 and page 221). The divisor is a self-resetting, free running 32-bit counter which allows the UART's logic to continue when it has reached zero.

With a 100 MHz clock, the baud rate is programmable from 100 Mbaud (one bit every 10 ns) to 0.023 baud — one bit every 42 seconds.

6.3.1 Baud Rate Setting

In order to determine the correct setting, the system clock frequency and the required baud (bits per second) rate must be equated. The output frequency of the Baud Generator is represented in the following equation:

- (Frequency/ (16 * Baud))-1

Due to errors introduced by rounding numbers, baud rates may not be attainable at certain PB_CLK frequencies.

The two 8-bit latches store the divisor in a 16-bit binary format. These divisor latches must be loaded during initialization to ensure proper operation of the Baud Generator. Upon loading either of the divisor latches, a 16-bit Baud counter is immediately loaded.

Chapter 7: General Purpose I/O Interface

This chapter outlines the functionality of the General Purpose I/O port. The topics addressed in this chapter include:

7.1 Overview

PowerPro features a flexible, General Purpose I/O (GPIO) interface. GPIO functionality is multiplexed with other features on PowerPro. Although all pins on PowerPro have a primary purpose, in many instances these pins are not used in their primary role. For example, a system may not require two UARTS and two I^2C ports. These pins, when not being used for their primary purpose, are assigned to the GPIO pool. All pins capable of GPIO have a mapping in the GPIO register (see page 235).

Refer to "Pin-out Information" on page 115 for pin multiplexing information.

7.2 GPIO Register

The GPIO port has three pins and a single register controlling its functionality. The following bullets outline the GPIO register bits:

- GPIO Enable (ENABLE[0:7]): This bit controls whether the GPIO port is activated.
- GPIO Mask (MASK[0:7]): This bit enables or masks writes.
- GPIO Direction (DIR[0:7]): This bit controls whether the GPIO port is an output or an input.
- GPIO Data (DATA[0:7]): This bit is a write value, which controls the value the pin assumes when it is an output.

Refer to "General Purpose I/O" on page 235 for more information on the GPIO register.

7.3 GPIO Signals

PowerPro has a series of multiplexed signals that can be programmed as GPIO signals. GPIO signals are present on all PowerPro interfaces. Refer to “Signal Description” on page 129 for more information on multiplexed PowerPro signals.

7.4 Reads

Reading from the data port returns the pin’s current value. When the pin is an input, the input value is returned. When the pin is configured as an output, the value output on the pin is returned.

Reading from a GPIO port, even if that port is not enabled, returns the value on the pin at the time the read command is executed.

7.5 Writes

In order to set pins bit-wise without affecting other pins in the same register, a write mask is provided. When the write mask value is 0, the enable, direction, and write data values are ignored. When the write mask value is 1, the enable, direction, and write data values are written.

The write mask has no effect on reads.

The GPIO, coupled with the general purpose timers, can enable software to control any low-to-medium speed device. The GPIO can control anything on the system with a non-time critical protocol.

7.6 Activating GPIO Functionality

After HRESET_ is negated, the GPIO pins are driven either low or high (depending on the individual signal requirements for signal activation) until they can be programmed to be enabled and are made inputs. GPIO signals must be programmed as GPIO signals before a GPIO interface can operate correctly.

For example, if signal GPIO[15], which is multiplexed with the PB_BG[1]_ signal, if it was required as a GPIO signal the following steps occur:

- GPIO[15] is floated during HRESET_.
- HRESET_ is deasserted.
- Power-up option determines that the arbiter is not enabled
  - The GPIO[15] signal is multiplexed with the PB_BG[0:1]_. Only one of the multiplexed functions is available. In order to enable the GPIO[15] pin, the PB_BG[0:1]_ signal — and therefore the arbiter — cannot be enabled.
- GPIO[15]_ is driven low.

Chapter 8: I2C Interface

This chapter outlines the functionality of the I2C Interface in PowerPro.The topics addressed in this chapter include:

8.1 Overview

PowerPro has two master-only, I2C bus compatible interfaces that each support a maximum of eight I2C slave devices. Refer to the *I2C Specification* for more information on the I2C protocol and the requirements for I2C bus compatible devices.

The I2C Interface is used for the initialization of registers after reset, and for reading serial presence detect data from DIMMs. PowerPro also provides a mechanism for the user to perform master read and write operations to EEPROMs or other I2C compatible slave devices.

The two I2C interfaces are classified as primary and secondary. The primary I2C Interface is connected to serial presence detect EEPROMS commonly found on DIMM modules. The secondary I2C interface is intended for general use. An example application could be using the interface for templates sensing.

The I2C signals can be multiplexed with Interrupt and GPIO functionality. When the signals are multiplexed, I2C functionality is no longer available on PowerPro.

The PowerPro I2C interfaces support the following features:

- I2C 7-bit device addressing
- Standard mode (up to 100 kbits/s)
- Single read/write (random read, byte write)
- Sequential read during post reset load sequence

The interface consists of two pins: I2Cx_SDA and I2Cx_SCL. I2Cx_SDA is a bidirectional open drain pin for transferring address, control, and data bits. I2Cx_SCL is the clock output for the I2C slave devices. I2Cx_SCL is derived from the processor clock. At the maximum Processor Bus Clock (PB_CLK) frequency of 100 MHz, the I2Cx_SCL clock rate is 100 kHz.

PowerPro does not support multiple masters on the same I2C bus. However, through the GPIO ports it is possible to accomplish multiple masters on the same I2C bus in software.

8.2 Bus Master Transactions

The I2C interfaces can perform master reads and writes from all external interfaces. These I2C transactions are generated by accessing the I2Cx Control and Status (I2Cx_CSR) register (see page 193). This register can be used to access EEPROMs or perform arbitrary single byte transfers to other I2C compatible devices. The I2C_CSR register contains the following fields:

- EEPROM Address (ADDR)
- Data (DATA)
- Device Code (DEV_CODE)
- Chip Select (CS)
- Read/Write (RW)
- Active (ACT)
- Error (ERR)

8.2.1 EEPROM Address

The 8-bit EEPROM ADDR field specifies the address for byte writes and random reads. The 8-bit DATA field is the source for writes and destination for reads. DEV_CODE is the 4-bit field that specifies the I2C device type. The default setting is 1010b. This is the code for EEPROMs. CS is the 3-bit field used to select one of the eight slaves on the I2C bus. The DEV_CODE and CS fields from the I2C 7-bit device address.

8.2.2 Active Bit

When the ACT bit is set, it means a transfer is in progress and the register is in read-only mode. After performing a write or read access, ACT bit must be polled until it is negated before performing other transfers. The ACT bit is also asserted during power-up EEPROM load.

8.2.3 Errors

When PowerPro is unable to complete an I2C access, the Error (ERR) bit, in the I2Cx_CSR register, is set when the Act (ACT) bit, in the I2Cx_CSR, is negated. The ERR bit must be cleared before attempting another access.

Chapter 9: Timers

This chapter outlines the functionality of PowerPro timers.The topics addressed in this chapter include:

9.1 Overview

PowerPro has both a single general purpose timer and a watchdog timer.

9.2 General Purpose Timer

PowerPro features a single, free-running 32-bit counter as general purpose system timer.

The general purpose timer is controlled by a series of register. The registers include the following:

- General Purpose Timer Base Count (GPT_COUNT) register
- General Purpose Timer Capture Events (GPT_CAPTURE) register
- General Purpose Timer Interrupt Control (GPT_IEN) register
- General Purpose Timer Interrupt Status (GPT_ISTATUS) register
- General Purpose Timer Trigger (GPT_Tx) register
- General Purpose Timer Compare x (GPT_Cx) register
- General Purpose Timer Compare Mask x (GPT_Mx)

The functionality of these registers are discussed in the following sections.

9.2.1 Base Count

The General Purpose Timer Base Count (GPTC) field, in the GPT_COUNT register (see page 198), contains the current value of the base count. All general purpose timer functions work from the value in the GPTC field.

The counter increments once per system clock (PB_CLK). All general purpose timer functions are based-on the reference count.

The GPTC bit must be written to in order set a new value in the case count. This resets the counter to the new input value.

There is no start or stop mechanism on the base count. It is constantly in use in PowerPro.

9.2.2 Capture Events

PowerPro has four capture time registers to capture the current value of the general purpose timer when an event occurs. The time of the free running counter (GPTC bit) is copied into the appropriate capture register when an enabled capture event occurs. The four capture registers are GPT_T*x*.

There are four different types of capture events which cause the GPTC bit value to be copied into the capture register. The following events cause the value to be copied:

1. A software event caused by setting one or more of Software Capture Event (SEVT[0:3]) bits, in the GPT_CAPTURE register (see page 199), to 1.
2. Activity on one of the four SDRAM banks, combined with SDRAM Bank Address Match Capture Enable (SD_AM) bits, in the GPT_CAPTURE register, being set. For example, to capture the timer time of the last activity on SDRAM bank number two, set SD_AM[2] to 1.
3. Activity on one of the four FLASH/ROM banks, combined with the setting of the ROM Bank Address Match Capture Enable (EE_AM) bits, in the GPT_CAPTURE register.
4. A match on the processor (60x) bus address match register, combined with Processor (60x) Bus Address Match Capture Enable (PB_AM) bits, in the GPT_CAPTURE register, being set.

Matching on this register does not require that the matched address lie within the address space normally claimed by PowerPro.

9.2.3 Compare Events

PowerPro has four compare registers which provide an event indicating the general purpose timer has reached and passed the compare time. The event could be an interrupt or register status setting.

Four registers are compared against the current value of the GPTC[0:31] bits. A compare event is generated when the current value of the general purpose timer counter matches the compare value of General Purpose Timer Compare Value (CT[0:31]) bits, in the GPT_CT register (see page 204). The compare event is logged in the Compare Status (CSTAT) bits, in the GPT_ISTATUS register (see page 202). An interrupt can also be generated when the feature is enabled through the GPT Compare Interrupt Enable (C_IEN) field, in the General Purpose Timer Interrupt Control (GPT_IEN) register (see page 201).

9.2.3.1 Qualifying Compare Events

The compare time is qualified with the GPT Timer Compare Mask (CM) field, in the General Purpose Timer Compare Mask x (GPT_Cx) register. When a corresponding mask bit is clear, that bit is used in the compare. When the CM field is set, that bit is ignored and assumed to always to match.

For example, to be notified every time the counter rolled over to 0xxx3_0000 (where ‘x’ is unimportant value) program the following values:

- CT[0:31] = 0x0003_0000
- CM[0:31] = 0x0003_FFFF

9.3 Watchdog Timer

PowerPro’s watchdog timer catches faults in real time operating systems. The watchdog timer monitors the operation of a system and forces it to act correctly if it begins to act incorrectly. Incorrect behavior could include miss-fetched instructions which cause the system to begin executing code from non-existent or improper memory locations.

Errors in system behavior can be caused by electrical noise power line, static electric discharge, power interruption, voltage drop or a variety of other issues.

9.3.1 Enabling the Timer

The watchdog timer is enabled by setting the Enable (ENABLE) bit, in the Watchdog Timer Control (WD_CTRL) register (see page 195).

9.3.2 Time Counts

At a 100 MHz (10 ns) clock period, the 32-bit watchdog timer gives about 42 seconds for a maximal setting. Reading the Current Watchdog Timer Count (WDC[0:31]) field in the Watchdog Timer Count (WD_COUNT) register (see page 197), returns the current value of the watchdog timer.

9.3.2.1 Time-outs

When enabled, the watchdog timer counts down from the value in the Watchdog Timer Initial Value (WDT[0:31]) bits, in the Watchdog Timer Timeout (WD_TIMEOUT) register (see page 196), to zero. When the to counter reaches the zero value a watchdog time-out interrupt is asserted. Refer to Chapter 11: “Interrupt Controller” on page 105 for more information on interrupts.

9.3.3 Resetting the Timer

When the Watchdog Timer Count Reset (WD_RST) bit, in the WD_CTRL register, is set to 1, the watchdog timer is reset back to value in the WDT field, in the WD_TIMEOUT register.

The watchdog timer can also be reset to the value in the WDT field by setting the ENABLE bit, in the WD_CTRL register, to 1 or by changing (writing to) the WDT field.

If the watchdog timer functionality is not required, the timer can be used as a general purpose timer.

Chapter 10: Error Handling

This chapter outlines the error handling functionality of the PowerPro and describes how PowerPro handles different error conditions.The topics addressed in this chapter include:

10.1 Overview

PowerPro has error detection, reporting and recovery for both the Processor Bus (PB) Interface and the SDRAM Interface.

Errors detected by PowerPro are reported to the processor (60x) bus through the assertion of the PB_TEA_ signal. When the Transaction Error Acknowledge Enable (TEA_EN) bit is set in the Processor Bus General Control (PB_GEN_CTRL) register (see page 162), processor (60x) bus and SDRAM non-correctable errors generate the PB_TEA_ signal.

PowerPro latches the address and type of transaction that caused the error in the status registers to assist diagnostic and error handling software. Processor Bus (PB) Interface and SDRAM Interface errors are logged in their corresponding register fields. PB Interface transaction errors are captured in the Processor Bus Error Attribute (PB_ERR_ATTR) register while the address where the error occurred is logged in the Processor Bus Address error Log (PB_AERR) register. Refer to the "Processor Bus Interface Errors" on page 102 for more information.

SDRAM ECC errors are logged in the ECC Uncorrectable Error Flag (ECC_UC) bit and the ECC Correctable Error Occurred Flag (ECC_CO[0:7]) field in the SDRAM Memory Bank X Control and Status (SD_BX_CTRL) register. Refer to "SDRAM Interface Errors" on page 103 for more information.

PowerPro logs an error in its error registers when one of the following conditions occur:

- invalid register access

- uncorrectable error
- data parity error on the processor (60x) bus

PowerPro does not log the transaction that caused a correctable, single-bit ECC error

When interrupts are enabled, PB Interface and SDRAM Interface error conditions can be routed to one of two external interrupt outputs. Refer to Chapter 11: “Interrupt Controller” on page 105 for more information.

10.2 Processor Bus Interface Errors

The PB Interface detects and reports the following errors:

- address parity
- data parity
- bus errors
  - — invalid transaction type
  - — invalid addressing for a memory space

10.2.1 Address Parity Errors

Address parity is enabled by setting the Address Parity Enable (AP_EN) bit, in the in the Processor Bus General Control (PB_GEN_CTRL) register (see page 162). Address parity errors are logged in the Processor Bus Address Error Log (PB_AERR) register (see page 168).

10.2.2 Data Parity Errors

Data parity is enabled by setting the Data Parity enable (DP_EN) bit in the PB_GEN_CTRL register. A data parity error is logged when the Data Parity Error (DPAR) bit is set in the PB_ERR_ATTR register (see page 166).

10.2.3 Bus Errors

When an unsupported processor (60x) bus transaction occurs, the Transaction Type Error Log (TT_ERR[0:4]) bit in the PB_ERR_ATTR register (see page 166) is set to indicate the error type. Refer to “Transaction Types” on page 39 for information on supported PB Interface transaction types.

10.2.3.1 Error Status

When the Error Status (ES) bit, in the PB_ERRCS register, is set it means an error has been logged and the contents of the Processor Bus Transaction Type Error Log (TT_ERR) field, Processor Bus SIZ Error Log (SIZ_ERR) field and PB_ERR_ADDR register are valid. Information in the log cannot be changed while ES is set. ES must be cleared, by writing a one to the bit, in order for the error log registers to capture future errors.

When the ES bit is clear and the PB address match interrupt is set, TT_ERR, SIZ_ERR, PB_ERR_ADDR contain information on the transaction which triggered the PB match address interrupt. This information is overwritten by a processor (60x) bus error. This address logging and address match mechanism is designed to be a system level debugging tool.

10.2.4 What PB Errors Indicate

There are two types of PB Interface errors: programming problems resulting in invalid transaction types, and programming problems resulting in sizes invalid for the memory space. Both of these errors are potentially fatal errors to a system.

10.3 SDRAM Interface Errors

The SDRAM Interface detects and reports the following errors:

- ECC correctable error
- ECC non-correctable error

10.3.1 ECC Errors

PowerPro enables ECC protection in MPC8260 and PowerPC 750 applications. The MPC8260 can enable ECC protection in certain applications, but when multiple processors are involved there is a potential for processor (60x) bus compliancy issues.

PowerPro supports ECC protection for the data path between PowerPro and system memory. ECC enables PowerPro to detect errors in the memory data path, as well as correct single-bit errors in the 64-bit data path. The ECC logic in PowerPro detects and corrects all single-bit errors and detects all double-bit errors.

10.3.1.1 Enabling ECC Protection

In order to enable PowerPro's ECC protection functionality, the ECC Global Enable (ECC_EN) bit must be set to 0 in the SDRAM Memory Bank x Control and Status register (see page 178). With this bit set to 1, and a DIMM with ECC functionality is in the memory bank, ECC correction is enabled in PowerPro.

When ECC correction is enabled, the ECC Uncorrectable Error (ECC_UC) bit, in the SDRAM Memory Bank X Control Status (SD_Bx_CTRL) register (see page 185), indicates if an uncorrectable error occurred. The ECC Correctable Error (ECC_CO) field flags if a correctable error occurred, and in which byte lane.

When the memory bank is in ECC correction mode, by setting the ECC Correction Enable (ECC_CE) bit in the SD_Bx_CTRL register, and the ECC Checking and Correction Enable (ECC_EN) bit is enabled, any single-bit correctable errors are corrected.

When ECC is enabled and a byte write to SDRAM memory is performed, PowerPro does not assert a response on the processor (60x) bus until the SDRAM accesses — reads followed by 64-bit write— are completed. PowerPro does not assert PB_ARETRY_ on the processor (60x) bus for an SDRAM access; PowerPro inserts wait states instead of PB_ARETRY_.

When ECC_CE is disabled, single-bit correctable errors are logged, but the uncorrected (invalid) data is returned. The ECC_CO field logs the ECC error and the byte lane where the error occurred. Writing a 1 clears the ECC_CO bit.

10.3.2 ECC Error Logging

In a PowerPro system, whenever a uncorrectable error passes through the Processor Bus Interface, the address, transaction type, and transaction size are logged to an internal register. This information is held in the register until that register is cleared. This ability enables the identification and debugging of software errors.

10.3.2.1 ECC Uncorrectable Error Logging

When the ECC Uncorrectable Error (ECC_UC) bit, in the SDRAM Memory Bank X Control Status (SD_Bx_CTRL) register (see page 185), is set to 1 it indicates that an uncorrectable error occurred. PowerPro logs the address, transaction type, and transaction size of the ECC uncorrectable error. Writing a 1 clears the ECC_UC bit

10.3.2.2 ECC Correctable Error Logging

When a ECC correctable error occurs, PowerPro sets the ECC_CO bit. This bit indicates that a correctable ECC error did occur and on which byte lane. PowerPro does not log the address, transaction type, and transaction size when a ECC correctable error occurs. Writing a 1 clears the ECC_CO bit.

10.3.3 What ECC Errors Indicate

The SDRAM Interface produces only two errors: correctable and non-correctable ECC errors. A high frequency of correctable ECC errors can indicate faulty memory, a faulty motherboard, or a faulty system chip. A non-correctable error can have its transaction retried for correction. A series of non-correctable errors indicates a faulty board, faulty memory, electrical interface problems, or a clock frequency too high for the current board configuration and operating conditions.

Chapter 11: Interrupt Controller

This chapter outlines the interrupt functionality of PowerPro.The topics addressed in this chapter include:

11.1 Overview

PowerPro has a 32-input interrupt controller. Each of the 32 interrupt sources can be mapped to generate one of two external interrupt outputs. When the Interrupt Enable (IE) bit, in the Interrupt Controller Enable (INT_ENABLE) register (see page 208), is set and an interrupt occurs, that interrupt generates either Processor Bus Interrupt Out (PB_INT_OUT) or External Interrupt Out (EXT_INT_OUT). The type of interrupt signaled is dependent on the setting in Interrupt Generation Type (IGTYPE) field, in the Interrupt Controller Cycle Generation Type (INT_GENERATE) register (see page 209).

Each of the 32 interrupt sources can be specified as edge or level sensitive. When the interrupt source is edge sensitive, an interrupt event can be generated on the presence of a positive or negative edge. When the interrupt source is level sensitive, an interrupt event can be continuously generated on the presence of a high or low level.These features are set in the Interrupt Controller Trigger Type (INT_TRIGGER) register (see page 211) and the Interrupt Controller Polarity (INT_POLARITY) register (see page 210).

11.2 Interrupt Sources

Interrupt sources are classified as either interrupts from normal device operation, or from a device exception.

11.2.1 Interrupts from Transaction Exceptions

Bus transaction type exceptions occur due to address parity errors, data parity errors or bus errors. When an error occurs PowerPro tracks the direction of the transaction through the interrupt enabling and status function.

Please refer to Chapter 10: “Error Handling” on page 101 for more information.

11.3 Interrupt Registers

PowerPro interrupt status and enabling, as well as message passing through mailboxes and doorbells are controlled by the interrupt registers. Table 18 provides a description of the PowerPro registers controlling these functions.

Table 18: Interrupt Register Description

Register Type	Register Description and Operation
Status	The status register bits cover all of the interrupt sources supported in PowerPro and indicate active interrupt sources when set (see “Interrupt Status” on page 107). With a some exceptions, all bits in these registers are read and cleared by setting (“R/Write 1 to Clear”)
Enable	The enable register bits cover all of the interrupt sources supported by PowerPro. (see “Interrupt Enabling” on page 108). With some exceptions, all bits in these registers are Read/Write.
Mapping	This series of registers allow each interrupt source to be mapped to a specific interrupt output pin. The mapping definitions are provided in Table 22 (see “Interrupt Mapping” on page 108)

11.3.1 Interrupt Status

When an interrupt source becomes active, the relevant status bit is set in one of the interrupt status registers. The status of each of the interrupt channels is reported in two registers: Interrupt Controller Status (INT_STATUS) and Interrupt Controller Masked Status (INT_MSTATUS). INT_STATUS (see page 206) reports the status of the interrupt sources regardless of INT_ENABLE settings. INT_MSTATUS (see page 207) masks INT_STATUS with INT_ENABLE to provide masked status results. Writing a 1 to INT_STATUS clears the associated interrupt flag.

Table 19: Register Description for Interrupt Controller Status

Name	Type	Reset By	Reset State	Function
STAT[0:31]	R/W1Clr	PB_RST	0	Interrupt status. Note that this status is independent of INT_ENABLE. This register reports the interrupt status regardless of that interrupt being enabled. 0 = Interrupt has not occurred 1 = Interrupt has occurred Write 1 to clear interrupt.

Table 20: Register Description for Interrupt Controller Masked Status

Name	Type	Reset By	Reset State	Function
MSTAT[0:31]	R	PB_RST	0	Masked interrupt status, result of INT_STATUS and INT_ENABLE. 0 = Interrupt has not occurred or is masked 1 = Interrupt has occurred and is not masked

11.3.2 Interrupt Enabling

The Interrupt Controller Enable (INT_ENABLE) register (see page 208) enables interrupt generation to the processor bus. The interrupt is enabled by setting the Interrupt Enable (IE) bit. When the IE bit is not set, no interrupt is sent to the processor. However the status of the interrupt sources are still detected..

Table 21: Register Description for Interrupt Controller Enable

Name	Type	Reset By	Reset State	Function
IE[0:31]	R/W	PB_RST	0	Interrupt Enable. 0 = Interrupt Disabled: generation to the processor is suppressed, but detection is still active. 1 = Interrupt Enabled: causes an interrupt to be generated to the processor.

11.3.3 Interrupt Mapping

A variety of internal and external events are mapped to each bit of the interrupter. Table 22 numbers the bit position that corresponds to each interrupt source.

Table 22: Interrupt Register Map

Bit	Description
0	External interrupt input, shared with EE_DATA[0]
1	External interrupt input, shared with EE_DATA[1]
2	External interrupt input, shared with EE_DATA[2]
3	External interrupt input, shared with EE_DATA[3]
4	External interrupt input, shared with EE_DATA[4]
5	External interrupt input, shared with EE_DATA[5]
6	External interrupt input, shared with EE_DATA[6]
7	External interrupt input, shared with EE_DATA[7]
8	External interrupt input, shared with UART0_TX
9	External interrupt input, shared with UART0_RX
10	External interrupt input, shared with UART1_TX
11	External interrupt input, shared with UART1_RX
12	External interrupt input, shared with I2C0_SCLK
13	External interrupt input, shared with I2C0_SDA
14	External interrupt input, shared with I2C1_SCLK

Table 22: Interrupt Register Map

Bit	Description
15	External interrupt input, shared with I2C1_SDA
16	External interrupt input, shared with SD_CS[4]
17	External interrupt input, shared with SD_CS[5]
18	External interrupt input, shared with SD_CS[6]
19	External interrupt input, shared with SD_CS[7]
20	External interrupt input, shared with PB_DBG0
21	External interrupt input, shared with PB_DBG1
22	Watchdog timer time-out
23	UART #0
24	UART #1
25	PB Address Match
26	I^2C0 and I^2C1 ACT bit
27	GPT Capture or Trigger
28	SDRAM ECC uncorrectable error detected
29	SDRAM ECC correctable error detected
30	PB address parity error
31	PB data parity error

11.4 Software Debugging

The following series of registers are provided to help in software debugging

- Interrupt Controller Vector Base Address register
- Interrupt Controller Vector Increment register
- Interrupt Controller Incremented Vector Base Address register
- Interrupt Controller Software Set register

11.4.1 Interrupt Controller Vector Base Address Register

These registers can be used as a pointer to an interrupt service routine. The following equation shows the constantly regenerated value:

Vector base address (INT_VBADDR[0:31]) + interrupt number (interrupt #) * increment amount (VINC)

When the registers are used in this fashion, the set of interrupt service routines is placed at INT_VBADDR[0:31]. Each interrupt service routine is separated from its neighbor by the Vector Increment (VINC) field, in the INT_VBADDR register (see page 212). For example, VINC: 0x100, 0x200, 0x400, or 0x800 in address space. Reading this register is a convenient way of finding the code to handle a generic PB_INT_ signal.

The Software Interrupt (SINT) field, in the Interrupt Controller Software Set register, is designed for software debugging. When a bit in INT_SOFTSET is programmed to one, the same effect as the corresponding interrupt is realized within the chip.

Chapter 12: Reset, Clock and Power-up Options

This chapter outlines the reset, clock and power-up functionality of PowerPro. The topics addressed in this chapter include:

12.1 Reset

The PowerPro reset design enables it to be used in both MPC8260-compatible systems and with PowerPC 603e, PowerPC 740, PowerPC 750, and PowerPC 7400 processors. PowerPro has two reset signals: power-on reset (PORESET_) and hard reset (HRESET_). The PORESET_ signal is a power-on reset that resets PowerPro. The PORESET signal must be asserted for 300ms in order to reset the PowerPro PLL. The HRESET_ signal is initiated from the processor (60x) bus.

All internal logic is synchronously reset at the de-assertion of HRESET_. While HRESET_ is asserted, all bidirectional output buffers are tristated asynchronously — without the presence of a clock.

12.1.1 Reset Signals

Table 23 lists PowerPro (CA91L750) reset signals.

Table 23: PowerPro Reset Pins

Pin Name	Direction	Description
PORESET_	Input	Power-on reset, active low. This signal enables the PLL to lock.
HRESET_	Tristate bidirectional	Processor (60x) bus reset, active low. Resets PowerPro.

12.1.2 PORESET_

The PORESET_ signal is an input only signal. The PORESET_ signal must be asserted with the initial application of power and held asserted for 300 ms after a stable clock is present at the PB_CLK input. With initial power-up, PowerPro enters an undefined state until the PORESET_ signal is asserted. All logic within PowerPro is reset when PORESET_ is asserted.

The PORESET_ signal must be asserted for 300 ms in order to allow time for the internal Phase Lock Loop (PLL) to lock. During the 300 ms a clock must be both present and stable on PB_CLK.

The power-up options (see page 120) are only latched upon the release (positive edge) of PORESET_. The system board must properly control the PORESET_ signal during the power-on cycle, and make sure that the required power-up options are present on EE_DATA[0:7] at the end of the PORESET_ assertion period.

12.1.3 HRESET_

The HRESET_ signal is a processor (60x) bus signal. This signal is used to perform a power-on hard reset of the devices connected to the processor bus. Many devices on the processor bus can simultaneously drive this signal. Every device connected to the processor bus has the capability of asserting this signal to reset its processor bus interface.

The power-up option on signal EE_DATA[4] determines whether the reset sequence corresponding to a configuration master is used, or the reset sequence corresponding to a configuration slave is used. If PowerPro latches EE_DATA[4] = 0 at the negation of PORESET_, PowerPro is configured as a configuration master and drives HRESET_ low until the configuration cycle is complete. Refer to page 112 for more information on configuration master and slave devices.

12.1.3.1 Configuration Master and Slave Devices

The terms configuration master and configuration slave originate from the reset sequence used by the MPC8260. The MPC8260 does not use the traditional power-up method of weak pull-ups or pull-downs on input pins. In systems with an MPC8260, a device on the processor (60x) bus — the reset configuration master — reads a 32-bit configuration word from a FLASH/ROM device it controls, and applies it to PB_D[0:31] while HRESET_ is asserted. The configuration master reads seven of these configuration words from FLASH/ROM, places the 32-bit value for each word on PB_D[0:31] and asserts PB_A[0:6] to signal to the configuration slaves the presence of the configuration words on the processor (60x) data bus.

In order to reset the system so that PowerPro remains configured and does not revert to its default setting, both PORESET_ and HRESET_ must be asserted. If only HRESET_ is asserted to reset the system, PowerPro powers-up in its default configuration.
PowerPro is a configuration slave by default.

A processor (60x) bus device configured as a configuration slave is designed to recognize the assertion of one of A[0:6] to be the configuration word it must use, and latch that word as a 32-bit power-up option.

An MPC8260 reset configuration master holds $\overline{\text{HRESET}}$ asserted until the reset configuration cycle is complete. During this time, it first reads four bytes out of a FLASH/ROM attached to it and places these bytes on PB_D[0:31]. When the data on PB_D[0:31] is stable, it asserts one of PB_A[0:6]. It then de-asserts the address line, reads the next four bytes out of it's local FLASH/ROM, and continues the process until seven 32-bit words are read from FLASH/ROM and placed separately on PB_D[0:31]. For each of the seven words placed on PB_D[0:31], a unique bit from PB_A[0:6] is asserted to allow up to seven devices on the processor (60x) bus to be configured. The configuration master then reads a 32-bit word from FLASH/ROM and configures itself. The exact meaning of each of the 32-bits in the configuration word is unique to the device being configured.

PowerPro is able to act as either a configuration master or a configuration slave during the HRESET_ assertion period based on the PowerPro power-up option EE_DATA[4].

There can only be one device assuming the configuration master role on a processor (60x) bus system bus at one time.

PowerPro as Configuration Master

When PowerPro is acting as a configuration master, there must be a FLASH/ROM connected to the PowerPro on EE_CS[0]. The FLASH/ROM connected to PowerPro can have its data port connected to either the EE_DATA[0:7] — making the data port 8-bit — or to the SDRAM data bus — making the data port 8-,16-,32-, or 64-bit. These options are selectable through power-up options. Refer to Chapter 5: "FLASH/ROM Interface" on page 69 for more information.

PowerPro uses the lowest 64 bytes of memory in the FLASH/ROM to store both the seven configuration words applied sequentially to PB_D[0:31] during the HRESET_ cycle, and the configuration word used to set the PowerPro's base address register. Table 24 illustrates the FLASH/ROM memory map used by the PowerPro when acting as a configuration master.

In order to reset the system so that PowerPro remains configured and does not revert to its default setting, both PORESET_ and HRESET_ must be asserted. If only HRESET_ is asserted to reset the system, PowerPro powers-up in its default configuration.
PowerPro is a configuration slave by default.

Table 24: PowerPro ROM Memory Map — as Reset Configuration Master

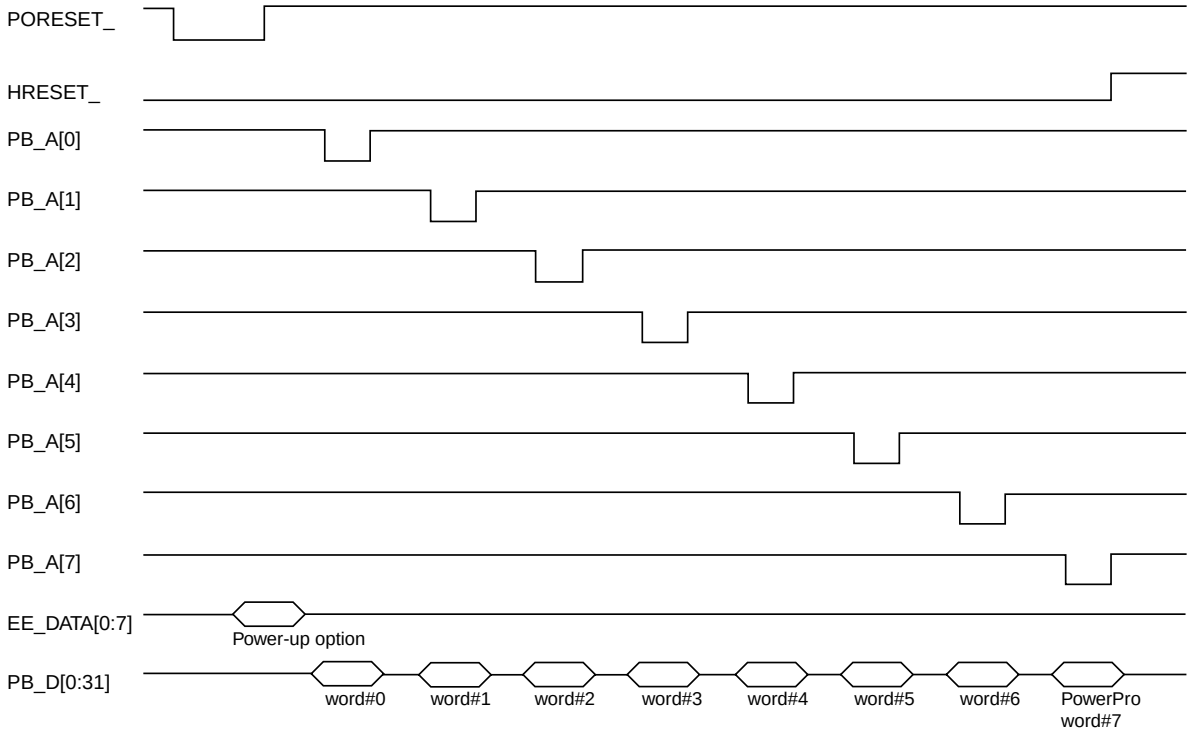
FLASH/ROM Address	Value
0x00000 - 0x00003	Configuration word placed on PB_D[0:31] coinciding with assertion PB_A[0] in HRESET_.
0x00004 - 0x00007	Not used
0x00008 - 0x0000B	Configuration word placed on PB_D[0:31] coinciding with assertion PB_A[1] in HRESET_.
0x0000C - 0x0000F	Not used
0x00010 - 0x00013	Configuration word placed on PB_D[0:31] coinciding with the assertion of PB_A[2] in HRESET_.
0x00014 - 0x00017	Not used
0x00018 - 0x0001B	Configuration word placed on PB_D[0:31] coinciding with the assertion of PB_A[3] in HRESET_.
0x0001C - 0x0001F	Not used
0x00020 - 0x00023	Configuration word placed on PB_D[0:31] which coincides with the assertion of PB_A[4] in HRESET_.
0x00024 - 0x00027	Not used
0x00028 - 0x0002B	Configuration word placed on PB_D[0:31] which coincides with the assertion of PB_A[5] in HRESET_.
0x0002C - 0x0002F	Not used
0x00030 - 0x00033	Configuration word placed on PB_D[0:31] which coincides with assertion of PB_A[6] in HRESET_.
0x00034 - 0x00037	Not used
0x00038 - 0x0003B	Configuration word used to configure PowerPro. D[0:23] is used to set PowerPro's register base address. D[24:31] must be set to the same values as the byte that was placed on EE_DATA[0:7] as power-up options.
0x0003C - 0x0003F	Not used

As a reset configuration master, PowerPro configures seven devices on the processor (60x) bus with a 32-bit word for each device. PowerPro also configures itself with the eighth configuration word. The register base address is the only value which is set using this option.

When PowerPro is configured during a HRESET_ assertion as a configuration master, the byte latched from EE_DATA[0:7] is overwritten with the configuration word value from PB_D[24:31]. In this situation, the appropriate value must be placed in the least significant byte of the configuration word.

Figure 16 illustrates the actions taken by PowerPro as configuration master. Information is latched from EE_DATA when PORESET_ is deasserted. The latched information that says whether PowerPro is a configuration master or configuration slave in the system. The FLASH/ROM address information in Table 24 is represented as word transfers in Figure 16.

Figure 16: Power-On Reset Sequence - PowerPro as Configuration Master



Note:
Power-up options latched on EE_DATA[0:7] on the rising edge of PORESET_

The reset configuration can occur with different data widths. The following figures show the configuration with a 16-bit data width and a 32-bit data width.

Figure 17: PowerPro as Configuration Master with a 16-bit Data Width

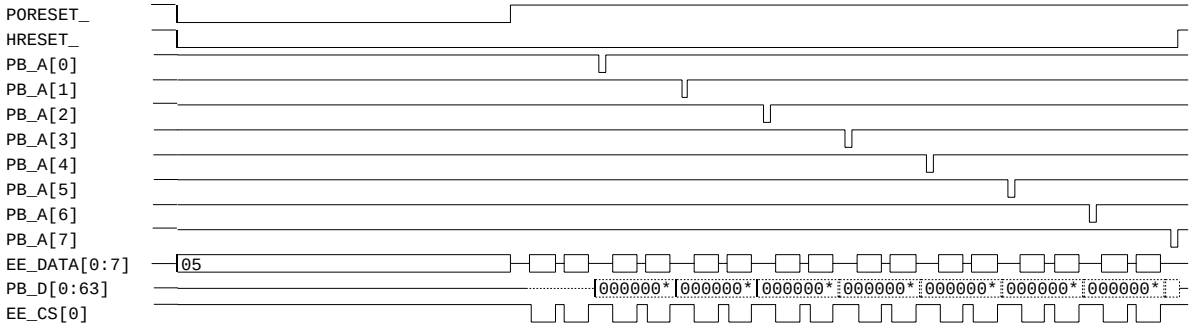


Figure 18: PowerPro as Configuration Master with a 32-bit Data Width

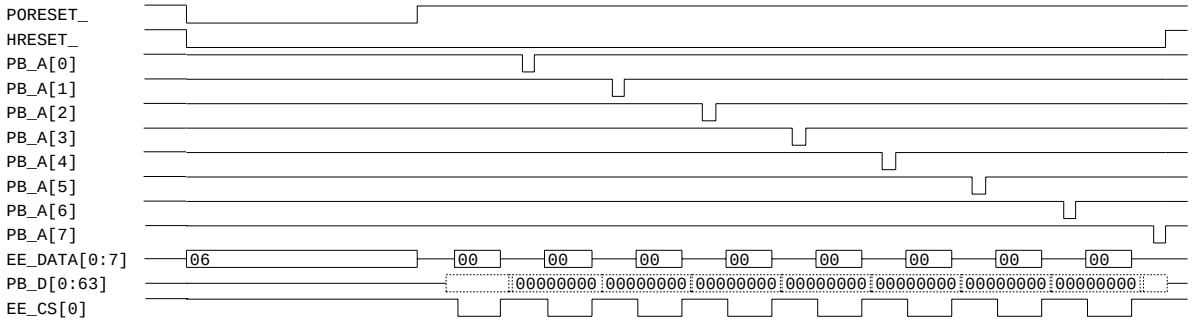
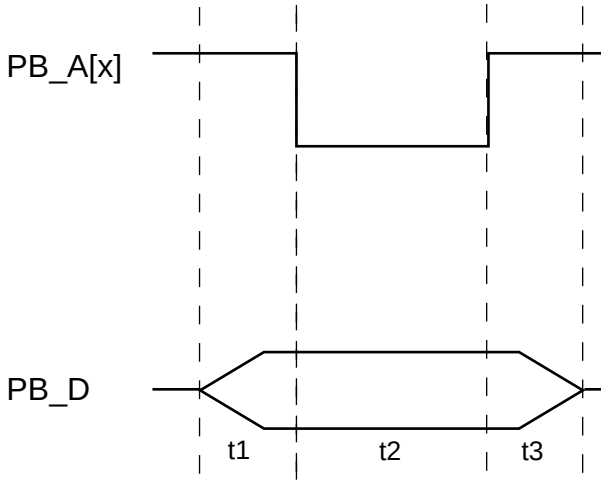


Figure 19 outlines one cycle in the power-up sequence. One PB_A[x] and PB_D cycle is highlighted. The x in PB_A[x] designates that this is a generic transaction but the diagram can be applied to any of the address and data lines.

Figure 19: PowerPro as Configuration Master - One Cycle



Note:
t1 = minimum of five clocks
t2 = minimum of five clocks
t3 = minimum of five clocks

PowerPro as Configuration Slave

PowerPro is a configuration slave by default when it is not configured as a configuration master. When PowerPro is a configuration slave the power-up options present on EE_DATA[5:7] are used to select which PB address line (0-7) PowerPro uses to latch its configuration word. If no configuration master is present, PowerPro uses the default value for the register base address.

When PowerPro is configured as a configuration slave, the 32-bit word latched into PowerPro is used to set the register base address (PB_D[0:23]). The 32-bit word latched into PowerPro is also used to replace the power-up options normally latched on EE_DATA[0:7] with the value held in the configuration word at PB_D[24:31].

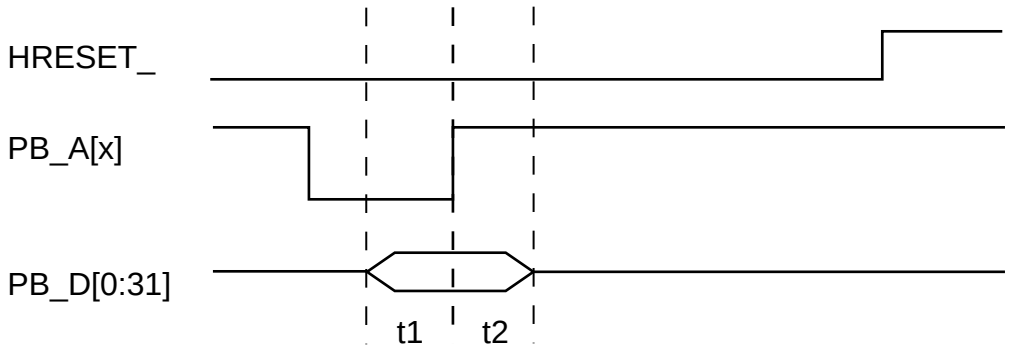
The value of the 32-bit configuration word which configures PowerPro is the same if it is acting as a configuration master and configuring itself, or acting as a configuration slave and latches this word from PB_D[0:31] when the selected PB_A[0:7] is asserted.

Systems which do not use the MPC8260 style reset configuration word mechanism provided in PowerPro must set PowerPro to act as a configuration slave, and ensure that the PB_A[0:7] pins are pulled high during the HRESET_ cycle.

The proper power-up configuration options must be set on EE_DATA[0:7] during the assertion of PORESET_.

Figure 20 illustrates PowerPro latching the configuration word for the register base address.

Figure 20: PowerPro as Configuration Slave



Note:
t1 = one clock
t2 = one clock

Configuration Addresses

Table 25 displays the different configuration addresses present on EE_DATA[5:7] that are used to select which PB address line (0-7). The Configuration Address Least Significant Bit (LSB) is EE_DATA[5]. The Configuration Address Most Significant Bit (MSB) is EE_DATA[7].

Table 25: Configuration Addresses

Configuration Address	EE_DATA[5:7]
0	000
1	100
2	010
3	110

Table 25: Configuration Addresses

Configuration Address	EE_DATA[5:7]
4	001
5	101
6	011
7	111

12.1.4 Power-on Reset Sequence

When PORESET_ is asserted, PowerPro performs self-configuration steps. The self-configuration steps are different of PowerPro is configured as a configuration slave or as a configuration master.

12.1.4.1 Reset Sequence For PowerPro as Configuration Slave

PowerPro has a series of steps that are followed when it is configured as a configuration slave. The following steps show the PowerPro's reset sequence:

1. Registers are loaded with their defined reset values.
2. Any registers with power-up reset options associated to them have the power-up reset value latched at the rising edge of HRESET_.
3. During the assertion of HRESET_ PowerPro monitors the PB_A[x] signal. The value for PB_A[x] is selected by EE_DATA[5:7] at the negation of PORESET_. If PB_A[x] is toggled, PowerPro latches the configuration word.

When PowerPro is the configuration slave, the reset configuration word is derived from the device driving the processor (60x) bus.

12.1.4.2 Reset Sequence For PowerPro as Configuration Master

PowerPro has a series of steps that are followed when it is configured as a configuration master. The steps of the PowerPro's reset sequence are as follows:

1. Registers are loaded with their defined reset values.
2. Any registers with power-up reset options tied to them have the power-up reset value latched at the rising edge of HRESET_.
3. PowerPro assumes that a boot FLASH/ROM is connected to EE_CS[0]. PowerPro reads the configuration words from system FLASH/ROM configuration table (refer to Table 24 and Figure 16).

When PowerPro is the configuration master, all reset configuration words are derived from the FLASH/ROM connected to EE_CS[0].

4. PowerPro applies the configuration words sequentially to PB_D[0:31].
5. The last configuration word (0x0003C) is used to configure PowerPro. This final word configures the Processor Bus Register Base Address register (see page 161).

12.2 Clocks

The PowerPro logic contains a single clock, the Processor Bus Clock (PB_CLK). All logic is synchronous to this clock. An internal PLL aligns the internal and external clocks.

The clock input enables PowerPro to be synchronized to the processor (60x) bus. PowerPro has a dedicated PLL designed to eliminate clock tree insertion delay. PowerPro requires the input clock to be at the specified frequency before reset is removed. The PLL is reset during the assertion of the PORESET_ signal. PORESET_ must be asserted for 300 ms in order to allow time for the internal Phase Lock Loop (PLL) to lock. During this time a clock must present and stable on PB_CLK. The PLLs are not locked until after the de-assertion of PORESET_.

The clocks to all devices on the processor (60x) bus and the SDRAM Interface must be balanced. Skew between these clocks must be accounted for when calculating timing requirements.

When PowerPro is in a system that does not have the PORESET_ signal, the PowerPro PORESET_ signal must be connected to the HRESET_ signal. For example, the MPC8260 has the PORESET_ signal, but the PowerPC 750 does not have the PORESET_ signal. In PowerPro applications that have a PowerPC 750, but not a MPC8260, PowerPro's PORESET_ signal must be connected to HRESET_ signal.

12.3 Power-up

A number of PowerPro features must be configured by the completion of the power-up reset sequence in order to ensure proper operation. PowerPro has multiplexed system pins to configure the power-up options. During the assertion of the PORESET_ signal, specific system pins are multiplexed as power-on reset inputs (refer to Table 26).

Figure 16 shows power-up options that are latched continuously by PowerPro while PORESET_ is de-asserted and HRESET _is asserted. Stable values must be present on the multiplexed system pins during the last 10 PB_CLKs before PORESET_ is de-asserted. The power-up option levels are usually provided by an external transceiver.

When PowerPro is configured during a HRESET_ assertion as a configuration master, the byte latched from EE_DATA[0:7] is overwritten with the configuration word value from PB_D[24:31]. In this situation, the appropriate value should be placed in the least significant byte of the configuration word.

12.3.1 System Boot

PowerPro can have a system boot FLASH/ROM attached to it. The boot FLASH/ROM must be attached to EE_CS[0] in order to communicate with PowerPro.

12.3.1.1 Attaching System Boot FLASH/ROM Memory

PowerPro reads the configuration word from a system boot FLASH/ROM device that can be attached to EE_CS[0].

When the reset sequence ends and there is a system boot FLASH/ROM is attached to EE_CS[0], the following register bits and fields are set by default:

- EE_DATA[0:7] signals are enabled as the dedicated FLASH/ROM data port.
  - The PORT bit in the EE_B0_CTRL register is set to 1.
- The MUX bit in the EE_B0_ADDR register is set to 11.
- The ENABLE bit in the EE_B0_ADDR register is set to 1.
- The A[0:23] field is set to 0xFFF001

12.3.1.2 System Boot FLASH/ROM Data Width

The PORT bit sets which FLASH/ROM data port the FLASH/ROM Interface is using: either the dedicated 8-bit FLASH/ROM data port or the SDRAM data bus. The MUX bit determines the FLASH/ROM multiplexing and on what data port the FLASH/ROM address appears. These two registers work together in the FLASH/ROM Interface in order to select the data ports used and address locations.

When EE_DATA[5] is used to set the Port (PORT) bit, in the ROM Memory Bank X Control (EE_BX_CTRL) register (see page 185), it also causes the ROM Address Multiplexing (MUX) 0 bit to be set in the ROM Memory Bank X Address (EE_BX_ADDR) register (see page 181).

When the PORT bit is to 0 — which enables EE_DATA[0:7] as the FLASH/ROM data port — the addressing mode register EE_B0_ADDR[MUX] is set to 11. Setting the Port bit to 1 — which enables the SDRAM data bus (SD_D) as the FLASH/ROM data port — causes the MUX bit to be set to 01.

This configuration limits how the boot FLASH/ROM can be connected to PowerPro. See Chapter 5: “FLASH/ROM Interface” on page 69 and Chapter 16: “ Registers” on page 153 for more information on FLASH/ROM addressing modes.

12.3.1.3 System Boot FLASH/ROM Memory not Required in the System

When no system boot FLASH/ROM is required in a system, EE_DATA[5:7] must be set to 001. This sets the boot FLASH/ROM to use EE_DATA[0:7] for a data port, and sets an invalid port width of 16-bits. The invalid data port disables the FLASH/ROM Interface.

Table 26 lists the multiplexed system pins, and the power-up options selected by them. Most power-up options have a read-only register associated with them.

Table 26: Power-Up Pin Assignments

Power-Up option	Selection	System Pin	Status Register
PLL Control	Enable internal PLL	EE_DATA[0] = 0	PB_GEN_CTRL [PLL_EN]
	Disable internal PLL	EE_DATA[0] = 1	
Boot FLASH/ROM Ready Throttling	Boot FLASH/ROM is not ready throttled. EE_READY is not used when accessing the boot FLASH/ROM on EE_CS[0].	EE_DATA[1] = 0	EE_B0_CTRL[RE]
	Boot FLASH/ROM is ready throttled. EE_READY is used to determine when data is available.	EE_DATA[1] = 1	
PB Arbiter	PB arbiter disabled	EE_DATA[2:3] = 00	PB_ARB_CTRL [Mx_EN]
	PB arbiter channel #0-1 enabled	EE_DATA[2:3] = 01 (= 0 for disabled)	
	PB arbiter channel #0-2 enabled. Address and data parity on the processor (60x) bus is disabled.	EE_DATA[2:3] = 10	
	PB arbiter channel #0-3 enabled. Address and data parity on the processor (60x) bus is disabled.	EE_DATA[2:3] = 11	

Table 26: Power-Up Pin Assignments

Power-Up option	Selection	System Pin	Status Register
MPC8260 Configuration Master: RSTCONF	Act as a reset configuration master from the FLASH/ROM attached to the PowerPro.	EE_DATA[4] = 0	n/a
	Act as a reset configuration slave, with (optionally) EE_DATA[5:7] indicating which address line to listen on.	EE_DATA[4] = 1	
Boot FLASH/ROM Port	Boot FLASH/ROM data port connected to EE_DATA[0:7]	EE_DATA[5] = 0	EE_B0_CTRL[PORT] EE_B0_ADDR[MUX]
	Boot FLASH/ROM data port connected to SD_D[0:7] / SD_D[0:15] / SD_D[0:31] or SD_D[0:63] depending on width.	EE_DATA[5] = 1	EE_DATA[5] = 0 EE_B0_ADDR[MUX]= 11 or EE_DATA[5] =1 EE_B0_ADDR[MUX]= 01
Boot FLASH/ROM Width: 8, 16, 32 or 64-bit	Select the boot FLASH/ROM width. Selecting EE_DATA[6:7] == 01b and EE_DATA[5] = 0 disables boot FLASH/ROM selection. PowerPro does not respond to 0xFFF0_0100 and above.	EE_DATA[6:7] 00 = 8-bit 01 = 16-bit 10 = 32-bit 11 = 64-bit	EE_B0_CTRL[WIDTH]

Chapter 13: JTAG Interface

This chapter outlines the JTAG functionality of the PowerPro. The topics addressed in this chapter include:

13.1 Overview

PowerPro provides a Joint Action Group (JTAG) Interface for boundary scan testing. The JTAG Interface IEEE 1149.1 compliant and implements the five test port signals required by the IEEE 1149.1 specification. For more information on JTAG operation, refer to the *IEEE 1149.1 Boundary Scan Specification*.

The Boundary Scan Description Language (BSDL) file for PowerPro is available on the Tundra website — www.tundra.com.

13.2 Interface Description

The JTAG Interface consists of a the following:

- Test Access Port interface
- Test Access Port (TAP) controller
- Instruction register
- Boundary Scan register
- Bypass register
- Idcode register

Each of these areas are described in detail in the following sections

13.3 JTAG Signals

PowerPro has dedicated JTAG signals. Table 27 lists the signals and describes their functionality.

Table 27: Test Signals

Pin Name	Pin Type	Description
JT_TCK	Input (LVTTL)	**Test Clock**: JTAG signal. Used to clock state information and data into and out of the device during boundary scan.
JT_TMS	Input (LVTTL) (Internal pull-up)	**Test Mode Select**: JTAG signal. Used to control the state of the Test Access Port controller
JT_TDI	Input (LVTTL) (Internal pull-up)	**Test Data Input**: JTAG signal. Used in conjunction with TCK to shift data and instructions into the Test Access Port (TAP) in a serial bit stream.
JT_TDO	Tristate output	**Test Data Output**: JTAG signal. Used in conjunction with TCK to shift data and instructions into the Test Access Port (TAP) in a serial bit stream.
JT_TRST[a]_	Input (LVTTL) (Schmitt trigger) (Internal pull-up)	**Test Reset**: JTAG signal. Asynchronous reset for the JTAG controller. This pin must be asserted during the power-up reset sequence to ensure that the Boundary Scan Register elements are configured for normal system operation. During normal operation of the device (when JTAG is not operational) this signal must be pulled low.
TEST_ON	Input (Internal pull-down)	**Test Enable**: Enables manufacturing test. During normal operation of the device (when JTAG is not operational) this signal must be pulled low.
BIDIR_CTRL	Input	**Bi-directional Control**: Manufacturing test pin.During normal operation of the device (when JTAG is not operational) this signal must be pulled low.

a. In order to ensure that the signal is pulled low when PowerPro is in normal operation, the signal can be connected to the PORESET_ signal.

13.3.1 JTAG Registers

PowerPro supports the mandatory JTAG registers. These registers include:

- Instruction register
- Boundary Scan register
- Bypass register
- Idcode register

The following sub-sections describe each of the JTAG registers.

13.3.1.1 Instruction Register

The 8-bit JTAG instruction register is a instruction and status register. As TAP controller instructions are scanned through the TDI input, the TAP controller status bits are scanned out through the TDO output.

13.3.1.2 Boundary Register

The PowerPro JTAG Interface has a chain of registers dedicated to boundary scan operation. These registers are not shared with any other functional registers of the PowerPro.

The boundary scan register chain includes registers controlling the direction of input and output drivers as well as the registers reflecting the signal value received or driven.

The JTAG controller must be reset to ensure that the Boundary Scan Register allows PowerPro to operate in system mode. The JTAG controller can be reset asynchronously with the assertion of JT_TRST_.

13.3.1.3 Bypass Register

The Bypass register is a single scan register used to bypass the boundary scan latches of PowerPro during boundary scan operations involving components other than PowerPro.

13.3.1.4 Idcode Register

The Idcode register contains device specific information. For example, the device manufacturer, part number, revision, and other device specific information is coded into this register.

13.4 TAP Controller

PowerPro has a standard JTAG TAP controller which controls instruction and data scan operations. The JT_TMS signal controls the state transitions of the TAP controller.

Chapter 14: Signal Description

This chapter outlines signals and signal descriptions of the PowerPro. The topics addressed in this chapter include:

14.1 Signal Overview

PowerPro uses 263 signals (excluding power and ground pins). The pin count for each functional blocks consists of the following:

- **Processor Bus signals**: 133
- **Memory signals**: 107
- **Miscellaneous signals**: 17
- **Test signals**: 6

When several functions are multiplexed onto one pin, the pin takes the name of the primary function. A primary function is defined as the function that the pin takes by default.

The GPIO Interface is inactive by default when the GPIO signals are multiplexed with system functionality.

Table 28 defines the different types of signal classification.

Table 28: Signal Type Definitions

Signal type	Signal type definition
Input	Standard input only signal.
Output	Standard output only signal.
Tristate output	Standard tristate output only signal.
Open drain	Open drain output that allows multiple devices to share as a wire-OR
Tristate bidirectional	Tristate input/output signal.
Bidirectional open drain	Open drain input/output which allows multiple devices to share as a wire or when it is used as output.

The signal description tables outline buffer types for the signals - either CMOS 1, CMOS 2, CMOS 3 or TTL. Refer to “Electrical Characteristics and Pin Information” on page 139 for more information of the electrical characteristics associated with the individual buffer types.

14.2 Processor Bus Signals

This section describes the PowerPro Processor Bus (PB) signal grouping. These signals are used to interface to the MPC8260, PPC 740, PPC 750, PPC 7400 and PPC 603e. Signals in this group are 3.3V LVTTL compatible. This interface is not 5V tolerant.

Table 29 summarizes the PB Interface signals. All primary operations are in bold. Signals with electrical characteristics different from the remainder of the group are placed at the end of Table 29.

Table 29: PB Signals

Pin Name	Pin Type	Description
PB_CLK	Input	**Processor Bus Clock**: All devices intended to interface with the bus processor side of the PowerPro must be synchronized to this clock. The PB_CLK can operate up to 100 MHz.
PORESET_	Input Buffer Type: TTL (Internal pull-down)	**Power-on reset**: Resets PowerPro.
HRESET_	Tristate bidirectional Buffer Type: CMOS 2	**Processor Bus Reset**: Resets all circuits on the PowerPro. The HRESET_ signal is a processor (60x) bus signal. This signal is used to perform a power-on hard reset of the devices connected to the processor bus.

Table 29: PB Signals

Pin Name	Pin Type	Description
PB_TS_	Input Buffer Type: TTL	**Transfer Start**: Indicates the beginning of a new address bus tenure.
PB_A[0:31]	Tristate bidirectional PB_A[0:7] Buffer Type: CMOS 2 PB_A[8:23] Buffer Type: CMOS 1 PB_A[24:32] Buffer Type: TTL	**Address Bus**: Address for the current bus cycle. PB_A[0:7] is driven during MPC8260 Master Configuration Cycles. At all other times the signal is an input.
PB_AP[0:3] Multiplexed with: PB_BR[2:3]_ + PB_BG[2:3]_ / GPIO[0:3]	Tristate bidirectional Buffer Type: CMOS 2	1. **Address Parity**: The processor address bus master drives this signal to indicate the parity of the address bus. 2. If parity checking is not being used, then the PowerPro can be used as a 4 channel arbiter on the PowerPC bus. In this case, these pins are used as the 3rd and 4th request and grant lines. The arbiter must be enabled in four channel mode (rather than the default two channel mode) for these pins to assume this function. 3. General Purpose I/O.
PB_TT[0:4]	Input Buffer Type: TTL	**Transfer Type**: The bus master drives these pins to specify the type of the transaction.
PB_TBST_	Input Buffer Type: TTL	**Transfer Burst**: The bus master asserts this pin to indicate that the current transaction is a burst transaction
PB_TSIZ[0:3]	Input Buffer Type: TTL	**Transfer Size**: Indicates the number of bytes to be transferred during a bus cycle.
PB_AACK_	Tristate bidirectional Buffer Type: CMOS 3	**Address Acknowledge**: A processor bus slave asserts this signal to indicate that it identified the address tenure. Assertion of this signal terminates the address tenure.
PB_ARTRY_	Tristate bidirectional Buffer Type: CMOS 2	**Address Retry**: Assertion of this signal indicates that the bus transaction should be retried by the processor bus master.

Table 29: PB Signals

Pin Name	Pin Type	Description
PB_D[0:63]	Tristate bidirectional Buffer Type: CMOS 3	**Data Bus**: These 64 pins are the Processor Data lines.
PB_DP[0:7] Multiplexed with: PB_DBG[2:3]_/ EE_A[28:23]/ GPIO[4:11]	Tristate bidirectional Buffer Type: CMOS 2	1. **Data Parity**: The processor data bus slave drives on reads, master drives on write to indicate the parity of the data bus. 2. Processor Bus Data Bus Grant: Enables the third and fourth channels of PowerPro's PowerPC bus arbiter. 3. EEPROM Address: If data parity is not used and GPIO is disabled for these pins, ROM address EE_A[28:23] is enabled. 4. General Purpose I/O
PB_DVAL_	Tristate bidirectional Buffer Type: CMOS 3	**Data Valid**: Indicates if the data beat is valid on PB_D[0:63]. Only used in MPC8260 systems for extended cycles.
PB_TA_	Tristate bidirectional Buffer Type: CMOS 3	**Transfer Acknowledge**: Indicates that a data beat is valid on the data bus. For single beat transfers, it indicates the termination of the transfer. For burst transfers, it is asserted four times to indicate the transfer of four data beats with the last assertion indicating the termination of the burst transfer.
PB_TEA_	Tristate bidirectional Buffer Type: CMOS 3	**Transfer Error Acknowledge**: Indicates a bus error
PB_INT_	Output Buffer Type: CMOS 2	**Processor Bus Interrupt**: Interrupt to local processor.
PB_BR[0:1]_ Multiplexed with: EE_A[28:27]_ / GPIO[12:13]_	Tristate bidirectional Buffer Type: CMOS 2	1. **Processor Bus Request**: These pins are the first two channels bus request lines of the Processor Bus Interface arbiter. 2. EEPROM Address: If the arbiter and GPIO are both disabled, EE_A[28:27] is enabled. 3. General purpose I/O

Table 29: PB Signals

Pin Name	Pin Type	Description
PB_BG[0:1]_ Multiplexed with: EE_A[26:25]_/ GPIO[14:15]	Tristate bidirectional Buffer Type: CMOS 2	1. **Processor Bus Grant**: These pins are the first two channels bus grant lines of the Processor Bus Interface arbiter. 2. EEPROM Address: If the arbiter and GPIO are both disabled, EE_A[26:25] is enabled. 3. General Purpose I/O
PB_DBG[0:1]_ Multiplexed with: INT[20:21]_ / EE_A[24:23]_ GPIO[16:17]_	Tristate bidirectional Buffer Type: CMOS 2	1. **Processor Bus Data Bus Grant**: These pins are the first two channels data bus grant lines of the Processor Bus Interface arbiter. 2. Interrupt Controller Input 3. EEPROM Address: If the arbiter, GPIO, and interrupt [23:24] are all disabled, EE_A[24:23] is enabled. 3. General Purpose I/O

14.3 Memory Signals

This section describes PowerPro signals used to memory interfaces. PowerPro supports SDRAM, EEPROM, FLASH, ROM, and SRAM memory.

Table 30 summarizes all memory signals. All primary operations are in bold. Signals with electrical characteristics different from the remainder of the group are placed at the end of Table 30.

Table 30: Memory Signals

Pin Name	Pin Type	Description
SD_RAS_	Output Buffer Type: CMOS 2	**SDRAM Command Bus**: RAS
SD_CAS_	Output Buffer Type: CMOS 2	**SDRAM Command Bus**: CAS
SD_WE_	Output Buffer Type: CMOS 2	**SDRAM Command Bus**: WE
SD_A[12:0] Multiplexed with: EE_A[12:0]	Output Buffer Type: CMOS 2	1. **SDRAM Address** 2. EEPROM Address
SD_BA[1:0] Multiplexed with: EE_A[14:13]	Output Buffer Type: CMOS 2	1. **SDRAM Bank Address** 2. EEPROM Address
SD_ECC[0:7] Multiplexed with: SD_DQM[0:7]	Tristate bidirectional Buffer Type: CMOS 3	1. **SDRAM ECC Protection**: If ECC correction used, there are eight ECC correction bits. 2.SDRAM DQM: If ECC is not being used, these are tied to the SDRAM DQM lines 0-7.
SD_CS[0:3]_ Multiplexed with: GPIO[19:22]	Tristate bidirectional Buffer Type: CMOS 1	1. **SDRAM Chip Select**: Two chip selects per memory DIMM 2. General Purpose I/O
SD_CS[4:7]_ Multiplexed with: INT[16:19]	Tristate bidirectional Buffer Type: CMOS 1	1. **SDRAM Chip Select**: Two chip selects per memory DIMM 2. Interrupt Controller Input
SD_D[0:63]	Tristate bidirectional Buffer Type: CMOS 2	**SDRAM / Peripheral Data Lines**

Table 30: Memory Signals

Pin Name	Pin Type	Description
SD_SELECT Multiplexed with: EE_SELECT/ EE_AL[0]/ GPIO[23]	Tristate bidirectional Buffer Type: CMOS 3	1. **SDRAM Bank Select**: External FET switch 2. EEPROM Buffer Select 4. EEPROM Address Latch 0: For time-multiplexing the EEPROM address, the first address phase is to be latched qualified with this signal 3. General Purpose I/O
EE_AL1_ Multiplexed with: GPIO[24]	Tristate bidirectional Buffer Type: CMOS 1	1. **EEPROM Address Latch 1**: For time-multiplexing the EEPROM address, the second address phase is to be latched qualified with this signal 2. General Purpose I/O.
EE_OE_ Multiplexed with: GPIO[25]	Tristate bidirectional Buffer Type: CMOS 2	1. **External Memory Output Enable** 2. General Purpose I/O
EE_WE_ Multiplexed with: GPIO[26]	Tristate bidirectional Buffer Type: CMOS 2	1. **External Memory Write Enable** 2. General Purpose I/O
EE_RNW Multiplexed with: GPIO[27]	Tristate bidirectional Buffer Type: CMOS 2	1. **EEPROM Read not Write**: Active 1 during an EEPROM read, 0 at all other times. 2. General Purpose I/O
EE_AL2 Multiplexed with: GPIO[28]_	Tristate bidirectional Buffer Type: CMOS 1	1. **EEPROM Address Latch 2**: For time-multiplexing the EEPROM address, the third address phase is to be latched qualified with this signal. 2. General Purpose I/O
EE_READY Multiplexed with: GPIO[29]	Tristate bidirectional Buffer Type: CMOS 2	1. **External Memory Ready Input Indicator** 2. General Purpose I/O
EE_CS[0:3]_ Multiplexed with: GPIO[30:33]	Tristate bidirectional Buffer Type: CMOS 1	1. **External Memory Chip Select**: One per bank. 2. General Purpose I/O

14.4 Miscellaneous Signals

This section describes PowerPro signals not necessarily dedicated to either the PB interface or memory interfaces.The miscellaneous signals have a variety of electrical capabilities which are indicated in Table 31.

Table 31: Miscellaneous Signals

Pin Name	Pin Type	Description
EE_DATA[0:7] Multiplexed with: EE_A[23:15]/ EE_A[31:24]/ EE_A[23:16]/ EE_A[15:8]/ EE_A[7:0]/ INT[0:7]/ GPIO[34:41]/ PWRUP[0:7]	Tristate bidirectional Buffer Type: CMOS 2	1. **ROM Data [0:7]** 2. ROM upper (MSB) address bits [23:15] 3. ROM address bits [31:24] (time-multiplexed). 4. ROM address bits [23:16] (time-multiplexed). 5. ROM address bits[15:8] (time-multiplexed). 6. ROM address bits[7:0] (time-multiplexed). 7. Interrupt inputs[0:7] 8. General Purpose I/O 7. Power-Up Options: Only latched during power-on reset
EXT_INT_ Multiplexed with: GPIO[18]	Tristate bidirectional Buffer Type: CMOS 2	1. **Extra Interrupt Output:** Intended to be tied to the MCP pin on a 60x processor. 2. General Purpose I/O
UART0_TX Multiplexed with: INT[8]/ GPIO[42]	Tristate bidirectional Buffer Type: CMOS 2	1. **Primary UART Transmit Line** 2. Interrupt Controller Input 3. General Purpose I/O
UART0_RX Multiplexed with: INT[9]/ GPIO[43]	Tristate bidirectional Buffer Type: CMOS 2	1. **Primary UART Receive Line** 2. Interrupt Controller Input 3. General Purpose I/O
UART1_TX Multiplexed with: INT[10]/ GPIO[44]	Tristate bidirectional Buffer Type: CMOS 2	1. **Secondary UART Transmit Line** 2. Interrupt Controller Input 3. General Purpose I/O
UART1_RX Multiplexed with: INT[11]/ GPIO[45]	Tristate bidirectional Buffer Type: CMOS 2	1. **Secondary UART Receive Line** 2. Interrupt Controller Input 3. General Purpose I/O

Table 31: Miscellaneous Signals

Pin Name	Pin Type	Description
I2C0_SCLK Multiplexed with: INT[12]/ GPIO[46]	Bidirectional open drain (5V tolerant) Buffer Type: CMOS 1	1. **I^2C Clock:** SDRAM serial presence detect**.** 2. Interrupt Controller Input 3. General Purpose I/O.
I2C0_SDA Multiplexed with: INT[13]/ GPIO[47]	Bidirectional open drain (5V tolerant) Buffer Type: CMOS 1	1. I **I^2C Data:** SDRAM serial presence detect. 2. Interrupt Controller Input 3. General Purpose I/O.
I2C1_SCLK Multiplexed with: INT[14]/ GPIO[48]	Bidirectional open drain (5V tolerant) Buffer Type: CMOS 1	1. **I^2C Clock**: Boot EEPROM. 2. Interrupt Controller Input 3. General Purpose I/O
I2C1_SDA Multiplexed with: INT[15]/ GPIO[49]	Bidirectional open drain (5V tolerant) Buffer Type: CMOS 1	1. **I^2C Data**: Boot EEPROM. 2. Interrupt Controller Input 3. General Purpose I/O
VDDCORE	Supply	**Core VDD**: Nominally 2.5 volts
VDDIO	Supply	**IO VDD**: Nominally 3.3 volts
VSS	Supply	Ground
PLL_VDDA	Supply	**Analog VDD**: Voltage supply pin to the analog circuits in the Phase Locked Loop (2.5V).
PLL_VSSA	Supply	**Analog VSS**: Voltage ground pin to the analog circuits in the Phase Locked Loop (2.5V).

14.5 Test Signals

This section describes PowerPro signals used to support silicon or board level testing.

Table 32: Test Signals

Pin Name	Pin Type	Description
JT_TCK	Input Buffer Type: TTL	**Test Clock**: JTAG signal. Used to clock state information and data into and out of the device during boundary scan.
JT_TMS	Input Buffer Type: TTL (Internal pull-up)	**Test Mode Select**: JTAG signal. Used to control the state of the Test Access Port controller
JT_TDI	Input Buffer Type: TTL (Internal pull-up)	**Test Data Input**: JTAG signal. Used in conjunction with TCK to shift data and instructions into the Test Access Port (TAP) in a serial bit stream.
JT_TDO	Tristate output Buffer Type: CMOS 2	**Test Data Output**: JTAG signal. Used in conjunction with TCK to shift data and instructions into the Test Access Port (TAP) in a serial bit stream.
JT_TRST[a]_	Input Buffer Type: TTL (Internal pull-up)	**Test Reset**: JTAG signal. Asynchronous reset for the JTAG controller. This pin must be asserted during the power-up reset sequence to ensure that the Boundary Scan Register elements are configured for normal system operation. During normal operation of the device (when JTAG is not operational) this signal must be pulled low.
TEST_ON	Input Buffer Type: TTL (Internal pull-down)	**Test Enable**: Enables manufacturing test. During normal operation of the device (when JTAG is not operational) this signal must be pulled low.
BIDIR_CTRL	Input Buffer Type: TTL	**Bi-directional Control**: Manufacturing test pin. During normal operation of the device (when JTAG is not operational) this signal must be pulled low.

a. In order to ensure that the signal is pulled low when PowerPro is in normal operation, the signal can be connected to the PORESET_ signal.

Chapter 15: Electrical Characteristics and Pin Information

This chapter describes the electrical characteristics of PowerPro. It also details the pin information of the device. The topics addressed in this chapter include:

15.1 Electrical Characteristics

15.1.1 Non-PCI Electrical Characteristics

The following table, Table 33, specifies the required DC characteristics of all PowerPro signal pins. Refer to "Signal Description" on page 129 for information on the buffer type for each signal.

Table 33: PowerPro PBGA Electrical Characteristics - CMOS Buffer

Symbol	Parameter	Condition	Min	Max	Units
V_{IL}	Input low voltage		0	0.3 x VDD_IO	V
V_{IH}	Input high voltage		0.7 x VDD_IO	VDD_IO	V
I_{OZ}	Tristate leakage Current	V_{OUT}=VDD or VSS	-10	+10	µA
V_{OH}	Output high voltage	CMOS 1		I_{OH}= 8	mA
		CMOS 2		I_{OH}= 12	mA
		CMOS 3		I_{OH}= 24	mA
V_{OL}	Output low voltage	CMOS 1	I_{OL}= -8		mA
		CMOS 2	I_{OL}= -12		mA
		CMOS 3	I_{OL}= -24		mA

Table 34: PowerPro PBGA Electrical Characteristics - TTL Buffer[a]

Symbol	Parameter	Condition	Min	Max	Units
V_{IL}	Input low voltage		0.0	0.8	V
V_{IH}	Input high voltage		2.0	VDD_IO	V
L_{IL}	Input leakage Current	No pull-up or pull-down resistance (V_{IN}=VDD or VSS)	- 10	+ 10	µA
L_{IL}_PU	Input leakage Current (internal pull-up)	High	- 10	+ 10	µA
		Low	TBD	TBD	µA
L_{IL}_PD	Input leakage Current (internal pull-down)	High	TBD	TBD	µA
		Low	- 10	+ 10	µA

a. All TTL buffer signals are inputs.

15.2 Pin Information

Table 35 shows the functions associated with a particular package ball number for the PowerPro 324 PBGA. The table includes all multiplexed pins. Any pin that only has one function is not multiplexed.

15.2.1 324 PBGA Pin to Package Cross-reference

The default function of the pin is the first function listed in Table 35.

Table 35: PowerPro Pin to Package Cross-references

Ball	Functions								
	0	1	2	3	4	5	6	7	8
Processor Bus Interface									
W8	PB_A[00]								
AA5	PB_A[01]								
Y6	PB_A[02]								
AB5	PB_A[03]								
AA6	PB_A[04]								
Y7	PB_A[05]								
AB6	PB_A[06]								
AA7	PB_A[07]								
W9	PB_A[08]								
Y8	PB_A[09]								
AB7	PB_A[10]								
AA8	PB_A[11]								
AB8	PB_A[12]								
Y9	PB_A[13]								
AA9	PB_A[14]								
AB9	PB_A[15]								
Y10	PB_A[16]								
AA10	PB_A[17]								
AB10	PB_A[18]								
AB11	PB_A[19]								
Y11	PB_A[20]								
AA11	PB_A[21]								
AB12	PB_A[22]								

Table 35: PowerPro Pin to Package Cross-references

Ball	Functions								
	0	1	2	3	4	5	6	7	8
AA12	PB_A[23]								
Y12	PB_A[24]								
AB13	PB_A[25]								
AA13	PB_A[26]								
Y13	PB_A[27]								
AB14	PB_A[28]								
AA14	PB_A[29]								
AB15	PB_A[30]								
Y14	PB_A[31]								
C5	PB_AACK_								
AA15	PB_AP[0]	PB_BR[2]_	GPIO[0]						
W14	PB_AP[1]	PB_BR[3]_	GPIO[1]						
AB16	PB_AP[2]	PB_BG[2]_	GPIO[2]						
Y15	PB_AP[3]	PB_BG[3]_	GPIO[3]						
B4	PB_ARTRY_								
AB3	PB_BG[0]_	EE_A[26]	GPIO[14]						
AA4	PB_BG[1]_	EE_A[25]	GPIO[15]						
W7	PB_BR[0]_	EE_A[28]	GPIO[12]						
Y4	PB_BR[1]_	EE_A[27]	GPIO[13]						
A3	PB_CLK								
C6	PB_D[00]								
B5	PB_D[01]								
A4	PB_D[02]								
D6	PB_D[03]								
A1	PB_D[04]								
B2	PB_D[05]								
C3	PB_D[06]								
D4	PB_D[07]								
B1	PB_D[08]								
G4	PB_D[09]								
C2	PB_D[10]								

Table 35: PowerPro Pin to Package Cross-references

Ball	Functions								
	0	1	2	3	4	5	6	7	8
E4	PB_D[11]								
D3	PB_D[12]								
C1	PB_D[13]								
D2	PB_D[14]								
F4	PB_D[15]								
E3	PB_D[16]								
H4	PB_D[17]								
E2	PB_D[18]								
F3	PB_D[19]								
E1	PB_D[20]								
F2	PB_D[21]								
G3	PB_D[22]								
F1	PB_D[23]								
G2	PB_D[24]								
J4	PB_D[25]								
H3	PB_D[26]								
G1	PB_D[27]								
H2	PB_D[28]								
J3	PB_D[29]								
H1	PB_D[30]								
J2	PB_D[31]								
J1	PB_D[32]								
K3	PB_D[33]								
K2	PB_D[34]								
K1	PB_D[35]								
L1	PB_D[36]								
L2	PB_D[37]								
L3	PB_D[38]								
M1	PB_D[39]								
M2	PB_D[40]								
M3	PB_D[41]								

Table 35: PowerPro Pin to Package Cross-references

Ball	Functions								
	0	1	2	3	4	5	6	7	8
N1	PB_D[42]								
N2	PB_D[43]								
N3	PB_D[44]								
P1	PB_D[45]								
P2	PB_D[46]								
R1	PB_D[47]								
P3	PB_D[48]								
P4	PB_D[49]								
T1	PB_D[50]								
R3	PB_D[51]								
T2	PB_D[52]								
U1	PB_D[53]								
T3	PB_D[54]								
U2	PB_D[55]								
V1	PB_D[56]								
R4	PB_D[57]								
U3	PB_D[58]								
V2	PB_D[59]								
W1	PB_D[60]								
Y1	PB_D[61]								
U4	PB_D[62]								
V3	PB_D[63]								
Y5	PB_DBG[0]_	INT[20]	GPIO[16]	EE_A[24]					
W6	PB_DBG[1]_	INT[21]	GPIO[17]	EE_A[23]					
T4	PB_DP[0]	PB_DBG[2]_	EE_A[23]	GPIO[4]					
W2	PB_DP[1]	PB_DBG[3]_	EE_A[24]	GPIO[5]					
V4	PB_DP[2]	EE_A[28]	GPIO[6]						
W3	PB_DP[3]	EE_A[27]	GPIO[7]						
Y2	PB_DP[4]	EE_A[26]	GPIO[8]						
AA1	PB_DP[5]	EE_A[25]	GPIO[9]						
AB1	PB_DP[6]	EE_A[24]	GPIO[10]						

Table 35: PowerPro Pin to Package Cross-references

Ball	Functions								
	0	1	2	3	4	5	6	7	8
AA2	PB_DP[7]	EE_A[23]	GPIO[11]						
Y3	PB_DVAL_								
AA3	PB_INT_								
W4	PB_TA_								
B7	PB_TBST_								
AB2	PB_TEA_								
AB4	PB_TS_								
A6	PB_TSIZ[0]								
C7	PB_TSIZ[1]								
B6	PB_TSIZ[2]								
D8	PB_TSIZ[3]								
C9	PB_TT[0]								
B8	PB_TT[1]								
D9	PB_TT[2]								
A7	PB_TT[3]								
C8	PB_TT[4]								
N19	PORESET_								
D7	HRESET_								
SDRAM Interface									
W15	SD_A[00]	EE_A[00]							
AB19	SD_A[01]	EE_A[01]							
Y17	SD_A[02]	EE_A[02]							
AA18	SD_A[03]	EE_A[03]							
W17	SD_A[04]	EE_A[04]							
W16	SD_A[05]	EE_A[05]							
Y18	SD_A[06]	EE_A[06]							
AA19	SD_A[07]	EE_A[07]							
AB20	SD_A[08]	EE_A[08]							
W18	SD_A[09]	EE_A[09]							
Y19	SD_A[10]	EE_A[10]							
AA20	SD_A[11]	EE_A[11]							

Table 35: PowerPro Pin to Package Cross-references

Ball	Functions								
	0	1	2	3	4	5	6	7	8
AB21	SD_A[12]	EE_A[12]							
AB22	SD_BA[0]	EE_A[13]							
AA21	SD_BA[1]	EE_A[14]							
AB18	SD_CAS_								
V20	SD_CS[0]	GPIO[19]							
U19	SD_CS[1]	GPIO[20]							
R19	SD_CS[2]	GPIO[21]							
V21	SD_CS[3]	GPIO[22]							
U20	SD_CS[4]	INT[16]							
V22	SD_CS[5]	INT[17]							
U21	SD_CS[6]	INT[18]							
T20	SD_CS[7]	INT[19]							
U22	SD_D[00]								
T21	SD_D[01]								
P19	SD_D[02]								
R20	SD_D[03]								
T22	SD_D[04]								
R21	SD_D[05]								
P20	SD_D[06]								
R22	SD_D[07]								
P21	SD_D[08]								
P22	SD_D[09]								
N20	SD_D[10]								
N21	SD_D[11]								
N22	SD_D[12]								
M21	SD_D[13]								
M20	SD_D[14]								
M22	SD_D[15]								
L22	SD_D[16]								
L21	SD_D[17]								
L20	SD_D[18]								

Table 35: PowerPro Pin to Package Cross-references

Ball	Functions								
	0	1	2	3	4	5	6	7	8
K22	SD_D[19]								
K21	SD_D[20]								
K20	SD_D[21]								
J22	SD_D[22]								
J21	SD_D[23]								
H22	SD_D[24]								
J20	SD_D[25]								
H21	SD_D[26]								
J19	SD_D[27]								
G22	SD_D[28]								
H20	SD_D[29]								
G21	SD_D[30]								
F22	SD_D[31]								
F21	SD_D[32]								
H19	SD_D[33]								
E22	SD_D[34]								
F20	SD_D[35]								
E21	SD_D[36]								
D22	SD_D[37]								
F19	SD_D[38]								
E20	SD_D[39]								
D21	SD_D[40]								
G19	SD_D[41]								
C22	SD_D[42]								
E19	SD_D[43]								
D20	SD_D[44]								
C21	SD_D[45]								
B22	SD_D[46]								
A22	SD_D[47]								
B21	SD_D[48]								
C20	SD_D[49]								

Table 35: PowerPro Pin to Package Cross-references

Ball	Functions								
	0	1	2	3	4	5	6	7	8
D19	SD_D[50]								
A21	SD_D[51]								
B20	SD_D[52]								
D16	SD_D[53]								
C19	SD_D[54]								
D18	SD_D[55]								
A20	SD_D[56]								
B19	SD_D[57]								
C18	SD_D[58]								
A19	SD_D[59]								
D15	SD_D[60]								
B18	SD_D[61]								
C17	SD_D[62]								
A18	SD_D[63]								
W19	SD_ECC[0]	SD_DQM[0]							
AA22	SD_ECC[1]	SD_DQM[1]							
Y21	SD_ECC[2]	SD_DQM[2]							
W20	SD_ECC[3]	SD_DQM[3]							
T19	SD_ECC[4]	SD_DQM[4]							
V19	SD_ECC[5]	SD_DQM[5]							
Y22	SD_ECC[6]	SD_DQM[6]							
W21	SD_ECC[7]	SD_DQM[7]							
AB17	SD_RAS_								
B17	SD_ SELECT	EE_ SELECT	EE_AL[0]	GPIO[23]					
AA17	SD_WE_								
FLASH/ROM Interface									
C16	EE_AL[1]	GPIO[24]							
C15	EE_AL[2]	GPIO[28]_							
B15	EE_CS[0]_	GPIO[30]							
A15	EE_CS[1]_	GPIO[31]							

Table 35: PowerPro Pin to Package Cross-references

Ball	Functions								
	0	1	2	3	4	5	6	7	8
C14	EE_CS[2]	GPIO[32]							
B14	EE_CS[3]	GPIO[33]							
D12	EE_DATA[0]	EE_A[22]	EE_A[31]	EE_A[23]	EE_A[15]	EE_A[07]	INT[0]	GPIO[34]	PWRUP[0]
A14	EE_DATA[1]	EE_A[21]	EE_A[30]	EE_A[22]	EE_A[14]	EE_A[06]	INT[1]	GPIO[35]	PWRUP[1]
C13	EE_DATA[2]	EE_A[20]	EE_A[29]	EE_A[21]	EE_A[13]	EE_A[05]	INT[2]	GPIO[36]	PWRUP[2]
B13	EE_DATA[3]	EE_A[19]	EE_A[28]	EE_A[20]	EE_A[12]	EE_A[04]	INT[3]	GPIO[37]	PWRUP[3]
A13	EE_DATA[4]	EE_A[18]	EE_A[27]	EE_A[19]	EE_A[11]	EE_A[03]	INT[4]	GPIO[38]	PWRUP[4]
C12	EE_DATA[5]	EE_A[17]	EE_A[26]	EE_A[18]	EE_A[10]	EE_A[02]	INT[5]	GPIO[39]	PWRUP[5]
B12	EE_DAT[A6]	EE_A[16]	EE_A[25]	EE_A[17]	EE_A[09]	EE_A[01]	INT[6]	GPIO[40]	PWRUP[6]
A12	EE_DATA[7]	EE_A[15]	EE_A[24]	EE_A[16]	EE_A[08]	EE_A[00]	INT[7]	GPIO[41]	PWRUP[7]
A17	EE_OE_	GPIO[25]							
A16	EE_READY	GPIO[29]							
D14	EE_RNW	GPIO[27]							
B16	EE_WE_	GPIO[26]							
Miscellaneous									
K4	BIDIR_CTRL								
A5	EXT_INT_	GPIO[18]							
C10	I2C0_SDA	INT[13]	GPIO[47]						
B9	I2C1_SDA	INT[15]	GPIO[49]						
B10	I2C0_SCLK	INT[12]	GPIO[46]						
A9	I2C1_SCLK	INT[14]	GPIO[48]						
Y20	JT_TCK								
W5	JT_TDI								
W22	JT_TDO								
D1	JT_TMS								
A8	JT_TRST_								
AA16	TEST_ON								
C11	UART0_RX	INT[9]	GPIO[43]						
A11	UART0_TX	INT[8]	GPIO[42]						
A10	UART1_RX	INT[11]	GPIO[45]						
B11	UART1_TX	INT[10]	GPIO[44]						

Table 35: PowerPro Pin to Package Cross-references

Ball	Functions								
	0	1	2	3	4	5	6	7	8
PLL									
C4	VDD_APLL								
A2	VDD_DPLL								
D5	VSS_APLL								
B3	VSS_DPLL								
Power									
D10	VDD_IO								
D11	VDD_IO								
D13	VDD_IO								
K19	VDD_IO								
L4	VDD_IO								
L19	VDD_IO								
M4	VDD_IO								
M19	VDD_IO								
N4	VDD_IO								
W11	VDD_IO								
W12	VDD_IO								
W13	VDD_IO								
R2	VDD_CORE								
Y16	VDD_CORE								
G20	VDD_CORE								
D17	VDD_CORE								
Ground									
J9	VSS								
J10	VSS								
J11	VSS								
J12	VSS								
J13	VSS								
J14	VSS								
K9	VSS								

Table 35: PowerPro Pin to Package Cross-references

Ball	Functions								
	0	1	2	3	4	5	6	7	8
K10	VSS								
K11	VSS								
K12	VSS								
K13	VSS								
K14	VSS								
L9	VSS								
L10	VSS								
L11	VSS								
L12	VSS								
L13	VSS								
L14	VSS								
M9	VSS								
M10	VSS								
M11	VSS								
M12	VSS								
M13	VSS								
M14	VSS								
N9	VSS								
N10	VSS								
N11	VSS								
N12	VSS								
N13	VSS								
N14	VSS								
P9	VSS								
P10	VSS								
P11	VSS								
P12	VSS								

Table 35: PowerPro Pin to Package Cross-references

Ball	Functions								
	0	1	2	3	4	5	6	7	8
P13	VSS								
P14	VSS								
No Connect									
W10	NO_ CONNECT[1]								

Chapter 16: Registers

This chapter outlines the registers of the PowerPro. The topics addressed in this chapter include:

16.1 Overview

PowerPro has a 512Kbyte register space designed to control behavior and monitor status.

16.2 Register Access

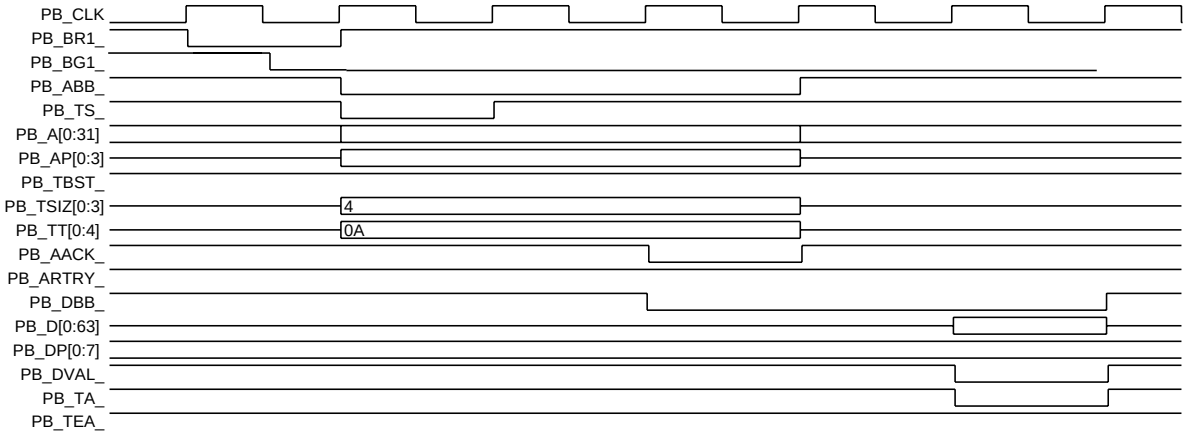
PowerPro registers are accessible by external masters through the PB Interface. Most registers are 32-bit registers. The UART registers are 8-bit. Writes are byte enabled. All bytes are returned on reads. The PowerPro supports only single beat register accesses.

Register writes to "write 1 to set/clear" status bits may not be reflected by an immediate register read.

16.2.1 Register Reads

Figure 21 shows a single 32-bit read access. The cycle is performed from an external master and the processor (60x) bus frequency is 100Mhz.

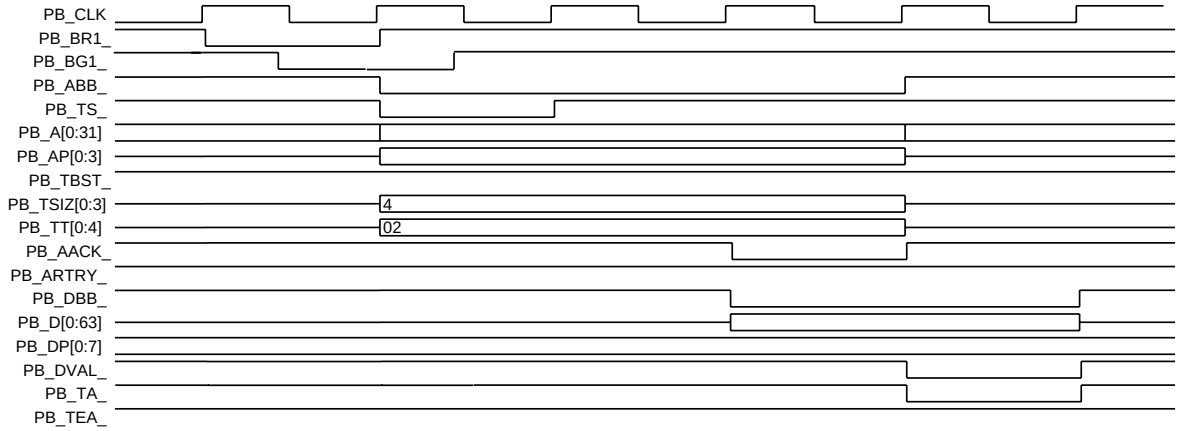
Figure 21: Register Read



16.2.2 Register Writes

and a single 32-bit write access. The cycle was performed from an external master. The PB bus frequency is 100Mhz.

Figure 22: Register Write



16.2.3 Register Image

PowerPro internal registers are accessed through an image specified by the Processor Bus Base Address (PB_REG_ADDR) register (see page 161). When an address matches the programmed address in PB_REG_ADDR, and the Transfer Type is supported, PowerPro registers are accessed.

The default value of PB_REG_ADDR is 0xFFFF_FE00 to 0xFFFF_FFFF and is specified at reset. However, the value can be programmed to relocate the registers elsewhere in the memory map. The value of PB_REG_ADDR can also be altered by a power-up configuration option.

Register access transactions are limited to a maximum of 8 bytes. The selected bytes are changed during writes. All bytes are returned on reads.

Refer to “Data Alignment” on page 43 for more information on setting the register access size from 4 bytes (default) to 8 bytes.

Bursting to the registers is not supported.

16.2.3.1 Termination

When a register access exceeds 8 bytes, or the external master attempts to burst to the registers, the PB Slave asserts Processor Bus Transfer error Acknowledge (PB_TEA_). This signal indicates a bus error and terminates the transaction.

16.2.3.2 SDRAM Memory Images

SDRAM memory images route memory mapped transactions to SDRAM. These images never assert PB_ARTRY_. Any valid processor (60x) bus transaction is supported, including MPC8260 extended cycles.

16.2.3.3 FLASH/ROM Memory Images

FLASH/ROM memory images route memory mapped transactions to EEPROM, Flash, SRAM and peripheral bus. Any valid processor (60x) bus cycle can be mapped to any size of EEPROM device, with PowerPro buffering the cycle information and performing multiple FLASH/ROM transactions as required.

When enabled, PB_ARTRY_ can be asserted by PowerPro during reads or writes when the FLASH/ROM Interface is busy processing a slow cycle.

16.3 Register Reset

The register space contains a single reset domain - the Processor bus Interface registers.

Registers are configured after reset through the Configuration Master or Configuration Slave configuration cycle. Refer to “Power-up” on page 120 for more information.

16.4 Register Descriptions

The PowerPro register space can be divided into the following groups:

- Processor Bus
- SDRAM
- FLASH/ROM / SRAM / Peripheral
- Error Handling
- Watchdog Timer
- General Purpose Timer
- Interrupt Controller

- UART
- General Purpose I/O (GPIO)

These groupings are shown in Table 36. The register map shows the offsets of all registers when accessed by the Processor Bus Register Image (PB_REG_BADDR) register.

Table 36: PowerPro Register Map

Offset (HEX)	Register Mnemonic	Register Name	Page
Processor Bus Registers			
0x000	PB_REG_ADDR	PB Base Address for registers (Can be altered via Power-up)	page 161
0x004	PB_GEN_CTRL	PB General Control and Configuration	page 162
0x008	PB_ARB_CTRL	PB Arbiter Control	page 164
0x00C	PB_ERR_ATTR	PB Error Log - Attributes	page 166
0x010	PB_ERR_ADDR	PB Error Log - Address	page 168
0x014	PB_AM	PB Address Match - Address	page 169
0x018	PB_AM_MASK	PB Address Match - Mask	page 170
0x01C	PowerPro Reserved		-
SDRAM Registers			
0x020	SD_REFRESH	SDRAM Refresh interval	page 171
0x024	SD_TIMING	SDRAM Timing Adjustment	page 172
0x028-03C	PowerPro Reserved		-
0x040	SD_B0_ADDR	SDRAM Bank 0 Base Address	page 175
0x044	SD_B0_MASK	SDRAM Bank 0 Base Address Compare Mask	page 176
0x048	SD_B0_CTRL	SDRAM Bank 0 Control	page 178
0x04C	PowerPro Reserved		-
0x050	SD_B1_ADDR	SDRAM Bank 1 Base Address	page 175
0x054	SD_B1_MASK	SDRAM Bank 1 Base Address Compare Mask	page 176
0x058	SD_B1_CTRL	SDRAM Bank 1 Control	page 178
0x05C	PowerPro Reserved		-
0x060	SD_B2_ADDR	SDRAM Bank 2 Base Address	page 175
0x064	SD_B2_MASK	SDRAM Bank 2 Base Address Compare Mask	page 176
0x068	SD_B2_CTRL	SDRAM Bank 2 Control	page 178

Table 36: PowerPro Register Map

Offset (HEX)	Register Mnemonic	Register Name	Page
0x06C	PowerPro Reserved		-
0x070	SD_B3_ADDR	SDRAM Bank 3 Base Address	page 175
0x074	SD_B3_MASK	SDRAM Bank 3 Base Address Compare Mask	page 176
0x078	SD_B3_CTRL	SDRAM Bank 3 Control	page 178
0x07C	PowerPro Reserved		-
FLASH/ROM Registers			
0x080	EE_B0_ADDR	ROM Bank 0 Base Address	page 181
0x084	EE_B0_MASK	ROM Bank 0 Base Address Compare Mask	page 183
0x088	EE_B0_CTRL	ROM Bank 0 Timing and Control	page 185
0x08C	PowerPro Reserved		-
0x090	EE_B1_ADDR	ROM Bank 1 Base Address	page 181
0x094	EE_B1_MASK	ROM Bank 1 Base Address Compare Mask	page 183
0x098	EE_B1_CTRL	ROM Bank 1 Timing and Control	page 185
0x09C	PowerPro Reserved		-
0x0A0	EE_B2_ADDR	ROM Bank 2 Base Address	page 181
0x0A4	EE_B2_MASK	ROM Bank 2 Base Address Compare Mask	page 183
0x0A8	EE_B2_CTRL	ROM Bank 2 Timing and Control	page 185
0x0AC	PowerPro Reserved		-
0x0B0	EE_B3_ADDR	ROM Bank 3 Base Address	page 181
0x0B4	EE_B3_MASK	ROM Bank 3 Base Address Compare Mask	page 183
0x0B8	EE_B3_CTRL	ROM Bank 3 Timing and Control	page 185
0x0BC	PowerPro Reserved		-
0x0C0	I2C0_CSR	I^2C Interface #0 (primary) Control and Status	page 193
0x0C4	I2C1_CSR	I^2C Interface #1 (secondary) Control and Status	page 193
0x0C8-0EC	PowerPro Reserved		-
Watchdog Timer Registers			-
0x0F0	WD_CTRL	Watchdog Timer Control	page 195
0x0F4	WD_TIMEOUT	Watchdog Timer Time-out Value	page 196

Table 36: PowerPro Register Map

Offset (HEX)	Register Mnemonic	Register Name	Page
0x0F8	WD_COUNT	Watchdog Timer Current Count	page 197
0x0FC	PowerPro Reserved		-
General Purpose Timer Registers			
0x100	GPT_COUNT	GPT Time Base Count	page 198
0x104	GPT_CAPTURE	GPT Capture Triggers	page 199
0x108	PowerPro Reserved		-
0x10C	GPT_INT	GPT Capture / Compare Timer Interrupt Control	page 201
0x110	GPT_ISTATUS	GPT Capture / Compare Timer Interrupt Status	page 202
0x114-11C	PowerPro Reserved		-
0x120	GPT_T0	GTP Capture #0 Time	page 203
0x124	GPT_T1	GPT Capture #1 Time	page 203
0x128	GPT_T2	GPT Capture #2 Time	page 203
0x12C	GPT_T3	GPT Capture #3 Time	page 203
0x130-13C	PowerPro Reserved		-
0x140	GPT_C0	GPT Compare #0 Time	page 204
0x144	GPT_C1	GPT Compare #1 Time	page 204
0x148	GPT_C2	GPT Compare #2 Time	page 204
0x14C	GPT_C3	GPT Compare #3 Time	page 204
0x150-15C	PowerPro Reserved		-
0x160	GPT_M0	GPT Compare #0 Time Mask	page 205
0x164	GPT_M1	GPT Compare #1 Time Mask	page 205
0x168	GPT_M2	GPT Compare #2 Time Mask	page 205
0x16C	GPT_M3	GPT Compare #3 Time Mask	page 205
0x170-17C	PowerPro Reserved		-
Interrupt Controller Registers			-
0x180	INT_STATUS	Interrupt Status	page 206
0x184	INT_MSTATUS	Interrupt Masked Status	page 207
0x188	INT_ENABLE	Interrupt Enable	page 208
0x18C	INT_GENERATE	Interrupt Generation Type	page 209

Table 36: PowerPro Register Map

Offset (HEX)	Register Mnemonic	Register Name	Page
0x190	INT_POLARITY	Interrupt Polarity	page 210
0x194	INT_TRIGGER	Interrupt Trigger Type	page 211
0x198	INT_VBADDR	Interrupt Vector Base Address	page 212
0x19C	INT_VINC	Interrupt Vector Increment	page 213
0x1A0	INT_VECTOR	Interrupt Vector Address	page 214
0x1A4	INT_SOFTSET	Interrupt Software Set	page 215
0x1AC	PowerPro Reserved		-
UART Registers			
0x1B0	UART0_Rx_Tx	UART0 Receive / Transmit Data	page 216
0x1B1	UART0_IER	UART0 Interrupt Enable	page 219
0x1B2	UART0_ISTAT_FIFO	UART0 Interrupt Status / FIFO Control	page 222
0x1B3	UART0_LCR	UART0 Line Control	page 227
0x1B4	UART0_MCR	UART0 Modem Control	page 229
0x1B5	UART0_LSR	UART0 Line Status	page 230
0x1B6	UART0_MSR	UART0 Modem Status	page 233
0x1B7	UART0_SCR	UART0 Scratchpad	page 234
0x1B8	PowerPro Reserved		
0x1C0	UART1_Rx_Tx	UART1 Receive / Transmit Data	page 216
0x1C1	UART1_IER	UART1 Interrupt Enable	page 219
0x1C2	UART1_ISTAT_FIFO	UART1 Interrupt Status / FIFO Control	page 222
0x1C3	UART1_LCR	UART1 Line Control	page 227
0x1C4	UART1_MCR	UART1 Modem Control	page 229
0x1C5	UART1_LSR	UART1 Line Status	page 230
0x1C6	UART1_MSR	UART1 Modem Status	page 233
0x1C7	UART1_SCR	UART1 Scratchpad	page 234
0x1C8	PowerPro Reserved		
0x1CC-1DC	PowerPro Reserved		-
General Purpose I/O Registers			

Table 36: PowerPro Register Map

Offset (HEX)	Register Mnemonic	Register Name	Page
0x1E0	GPIO_A	GPIO - Enable, Mask, Direction, Data [0:7]	page 235
0x1E4	GPIO_B	GPIO - Enable, Mask, Direction, Data [8:15]	page 235
0x1E8	GPIO_C	GPIO - Enable, Mask, Direction, Data [16:23]	page 235
0x1EC	GPIO_D	GPIO - Enable, Mask, Direction, Data [24:31]	page 235
0x1F0	GPIO_E	GPIO - Enable, Mask, Direction, Data [32:39]	page 235
0x1F4	GPIO_F	GPIO - Enable, Mask, Direction, Data [40:47]	page 235
0x1F8	GPIO_G	GPIO - Enable, Mask, Direction, Data [48:49]	page 235
0x1FC-	PowerPro Reserved		-

The following table describes the abbreviations used in the register descriptions.

Table 37: Abbreviations used in Register Descriptions

Abbreviation	Description
PORESET_	Processor Bus Reset
R/W	Read/Write
R	Read Only
R/Write 1 to Clear	Read/Write 1 to Clear
Write 1 to Set	Read 0/Write 1 to Set (Writing a 1 triggers an event)
0	Reset value is 0.
1	Reset value is 1.
PWRUP	Register bit loaded as a power-up option.
PowerPro Reserved	Do not write - Read back is undefined.
Reserved	Do not write - Read back undefined.

Table 38: PB Register Base Address

Register Name: PB_REG_ADDR	Register Offset: 000

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	BA[0:7]							
8-15	BA[8:15]							
16-23	BA[16:23]							Reserved
24-31	Reserved							

PB_REG_ADDR Description

Name	Type	Reset By	Reset State	Function
BA[0:23]	R/W	PORESET_	all 1	PowerPro memory mapped register base address.

PowerPro registers occupy a 512 bytes portion of the processor (60x) bus memory map. The base address of the 512 bytes portion is defined by this register. By default, after reset the registers occupy 0xFFFF_FE00 to 0xFFFF_FFFF.

The actual memory address of any register is simply the BA field (this register) + register offset. This register can be automatically loaded from a power-up configuration option. Refer to Chapter 12: “Reset, Clock and Power-up Options” on page 111 for more information.

The BA field is 24 bits. However, only 23 bits must be used. Bit 23 must have a value of 0.

Table 39: Processor Bus General Control Register

Register Name: PB_GEN_CTRL	Register Offset: 004

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	ECC_ TEST	Reserved						PLL_EN
8-15	Reserved							
16-23	Reserved							
24-31	Reserved			TEA_EN	ARTRY_ EN	DP_EN	AP_EN	PARITY

PB_GEN_CTRL Description

Name	Type	Reset By	Reset State	Function
ECC_TEST	R/W	PORESET_	0	Allows testing of ECC function. 0 = normal operation 1 = all write data masked. Writes to D[0:7] are mapped to the eight ECC check bits.
PLL_EN	R	PWRUP	n/a	Internal PLL enabled or disabled.
TEA_EN	R/W	PORESET_	0	Suppress PB_TEA_ generation 0 = PowerPro does not assert PB_TEA_ 1 = PowerPro asserts PB_TEA_ as required.
ARTRY_EN	R/W	PORESET_	0	Address retry enable 0=PB Slave never asserts PB_ARTRY_ 1=PB Slave asserts PB_ARTRY_ as required
DP_EN	R/W	PORESET_	0	Data Parity Enable 0 = Data parity checking disabled 1 = Data parity checking enabled
AP_EN	R/W	PORESET_	0	Address Parity Enable 0 = Address parity checking disabled 1 = Address parity checking enabled
PARITY	R/W	PORESET_	0	Parity 0 = Odd Parity, 1 = Even Parity

TEA_EN: If this bit is cleared, PowerPro does not assert PB_TEA_. Error conditions are signaled exclusively with interrupts when TEA_EN is not set.

ARTRY_EN: This bit controls the assertion of PB_ARTRY_ during transactions. When ARTRY_EN is set, the PB Slave Interface retries a processor (60x) bus master under the following conditions:

- Register read during I^2C load
- FLASH/ROM read when the transaction will take more than eight clocks
- FLASH/ROM write when the FLASH/ROM is busy
- SDRAM access when there is a pending FLASH/ROM read or write which uses the SDRAM control signals

ARTRY_EN is cleared by default. There is improved processor (60x) bus utilization by setting ARTRY_EN.

DP_EN, AP_EN: If this bit is cleared, PowerPro does not check the parity pins for the proper parity value. PowerPro drives parity on master writes and slave read cycles. Parity checking is disabled by default. If the external parity pins are disabled (parity not connected), then these pins become read-only values of 0.

Table 40: Processor Bus Arbiter Control Register

Register Name: PB_ARB_CTRL	Register Offset: 008

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	Reserved							
8-15	Reserved				M3_EN	M2_EN	M1_EN	M0_EN
16-23	Reserved				M3_PRI	M2_PRI	M1_PRI	M0_PRI
24-31	Reserved					PARK	BM_PARK	

PB_ARB_CTRL Description

Name	Type	Reset By	Reset State	Function
Mx_EN	R	PORESET_	PWRUP	External Master x Enable 0 = External requests ignored 1 = External requests recognized
Mx_PRI	R/W	PORESET_	0	External Master x Priority Level 0 = Low Priority 1 = High Priority
PARK	R/W	PORESET_	0	Bus Park Mode 0 = Park on last bus master 1 = Park on specific master
BM_PARK	R/W	PORESET_	0	Bus Master to be Parked 00 = External Master 0 01 = External Master 1 10 = External Master 2 11 = External Master 3

When the PB Interface arbitration control register is enabled, PowerPro controls the parameters of the processor bus arbiter. If address or data parity is being used, the arbiter is limited to two channels. If address and data parity are not used in the system, the arbiter is four channels. If any of the four arbiter channels are disabled, the pins these channels would have used can be accessed through the GPIO mechanism.

Mx_EN: When set, the arbiter recognizes address bus requests for this processor (60x) bus master. When cleared, the arbiter ignores address bus requests from this master.

Mx_PRI: Determines the arbitration priority for external masters.

PARK: When set, the arbiter parks the address bus on the processor (60x) bus master programmed in the BM_PARK field. When cleared, the arbiter parks the address bus on the last processor (60x) bus master to be granted the bus.

BM_PARK: Identifies the master to be parked, according to the following table:

Table 41: Parked PB Master

BM_PARK [1:0]	Parked PB Master	External Pins
00	M0	PB_BR[0]_/PB_BG[0]_
01	M1	PB_BR[1]_/PB_BG[1]_
10	M2	PB_BR[2_]/PB_BG[2]_
11	M3	PB_BR[3]_/PB_BG[3]_

Table 42: Processor Bus Error Attribute Register

Register Name: PB_ERR_ATTR	Register Offset: 00C

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	REG	DPAR	ECC_UC	Reserved			MES	ES
8-15	Reserved							
16-23	TT_ERR					Reserved		
24-31	SIZ_ERR				Reserved			

PB_ERR_ATTR Description

Name	Type	Reset By	Reset State	Function
REG	R	PORESET_	0	Register Invalid Access occurred and is logged
DPAR	R	PORESET_	0	PB Data Parity Error occurred and is logged
ECC_UC	R	PORESET_	0	ECC Uncorrectable Error occurred and is logged
MES	R	PORESET_	0	Multiple Error Status 1 = a second error occurred before the first error could be cleared.
ES	R/Write 1 to Clear	PORESET_	0	Error Status 0 = no error currently logged 1 = error currently logged Write 1 to clear all status bits in this register.
TT_ERR[0:4]	R	PORESET_	0	Processor bus Transaction Type Error Log
SIZ_ERR[0:3]	R	PORESET_	0	Processor bus SIZ field Error Log

The processor bus interface logs errors when it detects either a maximum parity error or the assertion of PB_TEA_. Logged error conditions include: invalid register access and invalid FLASH/ROM access.

ES: When the ES bit is set, it means an error has been logged and the contents of the TT_ERR, SIZ_ERR and AERR are valid. Information in the log cannot be changed while ES is set. Clearing the ES by writing a one to the bit allows the error log registers to capture future errors.

If ES is clear and the PB address match interrupt is set, TT_ERR, SIZ_ERR, PB_ERR_ADDR contain information on the transaction which triggered the PB match address interrupt. This information will be overwritten by a genuine 60x bus error. This address logging and address match mechanism should be useful as a system level debugging tool.

MES: Determines if multiple errors occur. The processor bus error logs are not be overwritten when MES is set. Clearing ES also clears MES.

Table 43: Processor Bus Address Error Log

Register Name: PB_ERR_ADDR	Register Offset: 010

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	A							
8-15	A							
16-23	A							
24-31	A							

PB_ERR_ADDR Description

Name	Type	Reset By	Reset State	Function
A[0:31]	R	PORESET_	0	Processor address error log

The processor bus interface logs errors when it detects: parity error or assertion of PB_TEA_ conditions.

A: The address of a processor bus transaction that generates an error condition is logged in this register.

Table 44: Processor Bus Address Match

Register Name: PB_AM_ADDR	Register Offset: 014

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	A							
8-15	A							
16-23	A							
24-31	A							

PB_AM_ADDR Description

Name	Type	Reset By	Reset State	Function
A[0:31]	R/W	PORESET_	0	Address Match Address

The PB_AM_ADDR register provides system debugging capabilities to PowerPro. If a processor (60x) transaction (not necessarily claimed by the PowerPro) matches the Address Match Address field, and is qualified by the Address Match Mask field in the PM_AM_ADDR register, then the event can generate an address match interrupt. The event can also be captured in the general purpose timer count value.

The register M field qualifies the address provided, enabling a range of addresses to be matched. The address bit will be compared if the corresponding mask bit is set. If the mask bit is masked (cleared), that address bit is ignored and, therefore, always match.

For example, for transactions that match the address range 0x5590_3000 - 0x5590_3FFF the following values are programmed:

- PB_AM_ADDR.A[0:31] = 0x5590_3000
- PM_AM_ADDR.M[0:31] = 0xFFFF_F000

Table 45: Processor Bus Address Match Mask

Register Name: PB_AM_MASK	Register Offset: 018

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	M							
8-15	M							
16-23	M							
24-31	M							

PB_AM_MASK Description

Name	Type	Reset By	Reset State	Function
M[0:31]	R/W	PORESET_	0xFFFF_FFFF	Address Match Mask

The PB_AM_MASK register provides system debugging capabilities to PowerPro. If a processor (60x) transaction (not necessarily claimed by the PowerPro) matches the Address Match Address field, and is qualified by the Address Match Mask field in the PM_AM_ADDR register, then the event can generate an address match interrupt. The event can also be captured in the general purpose timer count value.

The register M field qualifies the address provided, enabling a range of addresses to be matched. The address bit will be compared if the corresponding mask bit is set. If the mask bit is masked (cleared), that address bit is ignored and, therefore, always match.

For example, for transactions that match the address range 0x5590_3000 - 0x5590_3FFF the following values are programmed:

- PB_AM_ADDR.A[0:31] = 0x5590_3000
- PM_AM_ADDR.M[0:31] = 0xFFFF_F000

Table 46: SDRAM Refresh Interval

Register Name: SD_REFRESH	Register Offset: 020

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	Reserved							
8-15	Reserved							
16-23	T							
24-31	T							

SD_REFRESH Description

Name	Type	Reset By	Reset State	Function
T[0:15]	R/W	PORESET_	0x061A	SDRAM Refresh interval (in units of processor (60x) bus clocks)

T: Determines the refresh interval for SDRAM in units of processor bus clocks (PB_CLKs). The programmed number of clocks are counted, and at the end of the interval a REFRESH command is sent to all SDRAM banks. The register must be set to an appropriate interval for the memory attached and the PB_CLK frequency the system is using.

There is a minimum refresh interval for SDRAM. In the SD_REFRESH register the minimum refresh interval is T[0:15] = 0040h. SD_REFRESH[24] is automatically set to 1 if T[0:15] is less than or equal to 003Fh.

The refresh interval is determined by the PB_CLK frequency, and the refresh interval of the memory devices used in the system. If different memory devices are used in different banks, a refresh time appropriate to the worst case device must be programmed..

Worst case device means the device requiring the most frequent refresh interval.

Table 47: SDRAM Timing Parameters

Register Name: SD_TIMING	Register Offset: 024

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	ENABLE	DQM_EN	Reserved	CL	Reserved	EX_DPs	TUNE	
8-15	Reserved			T_RC				
16-23	Reserved							
24-31	Reserved							

SD_TIMING Description

Name	Type	Reset By	Reset State	Function
ENABLE	R/W	PORESET_	0	Enables SDRAM access. Setting to a 1 causes a reset sequence to be sent to all SDRAMs enabled and configured. All SDRAM parameters must be correctly set before enabling this bit.
DQM_EN	R/W	PORESET_	0	Data Quality Mask enable/ECC Global enable 1 = DQM enabled and ECC globally disabled. This setting overrides the individual bank settings for ECC (refer to the ECC_EN bit in the SDRAM_Bx_CTRL on page page 178). With this setting ECC[0:7] pins are used as DQM pins. 0= DQM disabled and ECC globally enabled, but may be individually disabled per bank. ECC[0:7] pins are used for ECC correction. DQM[0-7] connected to DQM0 - all byte writes (ECC or non-ECC) are done using read-modify-write.

SD_TIMING Description

Name	Type	Reset By	Reset State	Function
CL	R/W	PORESET_	0	CAS Latency 0 = 2 PB_CLKs 1 = 3 PB_CLKs
EX_DP	R/W	PORESET_	0	External datapath. When set, the SDRAM data bus is connected directly to the processor (60x) data bus. In this case, PowerPro only drives DVAL/TA, not D[0:63]. **Note:** When EX_DP is set to 1, the TUNE bits are set to 00.
TUNE[0:1]	R/W	PORESET_	00	PowerPC to SDRAM datapath tune bits. 00 = no pipelined stage 01 = one piplined stage on the output data path 10 = one pipeline stage on the input data path 11 = two pipeline stages, one on the input path and one on the output path **Note:** When EX_DP is set to 1, the TUNE bits are set to 00.
T_RC	R/W	PORESET_	0x0F	t_{RC} SDRAM timing parameter

This register contains settings global to all four SDRAM banks.

ENABLE: By default, all four SDRAM banks are disabled (not able to be accessed). For all attached memory banks, SD_Bx_ADDR, SD_Bx_MASK, and SD_Bx_CTRL need to be set appropriate to the type of memory installed. Then, the ENABLE bit in this register is to be set to a one, which will cause a reset sequence (consisting of a precharge-all, eight refresh cycles, a mode register set, and eight more refresh cycles) to be sent to all attached and enabled SDRAM banks. Individual SDRAM banks cannot be enabled or disabled once this bit is set.

TUNE: The highest system performance is a result of the least amount of latency between the processor (60x) bus requesting memory and that request being filled. However, a heavily loaded and fast system is not able to meet timing criteria without pipeline stages inserted in the datapath. The TUNE bits controls the number of pipeline stages inserted in the datapath.

When ECC correction is enabled on any bank, two pipeline stages are inserted in the data path by default. If ECC correction is not used on any memory bank, then the following settings are available:

- 00 = no pipeline stages. If EX_DP = 0, then data flows from SDRAM to the 60x bus through the PowerPro in the same clock, with the PowerPro acting as a simple data buffer. If EX_DP = 1, then it is assumed that the SDRAM data bus is connected directly to the 60x data bus; in this case the PowerPro drives DVAL/TA but not D[0:63].
- 01 = one pipeline stage on the output data path. One stage of pipeline is inserted on the outgoing data path. For 60x->SDRAM transfers (writes), the data to the SDRAM comes directly off of an internal PowerPro register. For SDRAM to processor (60x) bus transfers (reads), the data to the processor (60x) bus comes directly off of an internal PowerPro register.
- 10 = one pipeline stage on the input data path.
- 11 = two pipeline stages, one on the input path and one on the output path.

EX_DP: External Datapath. When set, PowerPro does not drive the processor (60x) data bus. When this bit is set, ECC correction and processor (60x) data parity are not available.

When EX_DP is set to 1, the TUNE bits are set to 00.

T_RC: SDRAM t_{RC} timing parameter; largest of all memory banks connected.

Table 48: SDRAM Memory Bank *x* Address

Register Name: SD_B*x*_ADDR	Register Offset: 040/050/060/070

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	A							
8-15	A							
16-23	Reserved							
24-31	Reserved							ENABLE

SD_B*x*_ADDR Description

Name	Type	Reset By	Reset State	Function
A[0:15]	R/W	PORESET_	0000/ 0100/ 0200/ 0300	Base address of SDRAM DIMM #0/1/2/3. The base address of a bank is qualified by the mask to determine if a given address is within range.
ENABLE	R/W	PORESET_	0	Bank (DIMM) Enable 0 = slot vacant 1 = slot filled with SDRAM module

The four available SDRAM banks are mapped to the 60x address space through the Address (A) field. This field indicates the memory space base address and the Mask (M) field, in the SD_Bx_MASK register, indicates the memory block size. Note that this requires that the memory space allocated be aligned with the size of the memory attached. For example a 128 Mbyte DIMM can only be mapped to addresses on a 128 Mbyte boundary.

The M field specifies which bits of the base address are used in the comparison, and which are ignored. A mask bit setting of 1 indicates that the corresponding bit is used in the comparison, while a mask bit setting of 0 indicates that the corresponding bit is not used (masked) in the address comparison.

For example, if a 64 Mbyte memory is attached to SDRAM bank #0 mapping this physical memory into the 60x address space (64 MByte aligned) of 0x3400_0000 - 0x37FF_FFFF requires the following settings:

- SD_B0_ADDR[0:15] = 0x3400
- SD_B0_MASK[0:15] = 0xFC00: indicates a 64 MByte block size

Table 49: SDRAM Memory Bank *x* Address Mask

Register Name: SD_B*x*_MASK	Register Offset: 044/054/064/074

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	M							
8-15	M							
16-23	Reserved							
24-31	Reserved							

SD_B*x*_MASK Description

Name	Type	Reset By	Reset State	Function
M[0:15]	R/W	PORESET_	0xFFFF	Mask to qualify bank address. Only bits selected in the mask are used for the address compare.

The four available SDRAM banks are mapped to the 60x address space through the Address (A) field which indicates the memory space base address and the Mask (M) field, in the SD_Bx_MASK register, indicates the memory block size. Note that this requires that the memory space allocated be aligned with the size of the memory attached. For example a 128 Mbyte DIMM can only be mapped to addresses on a 128 Mbyte boundary.

The M field specifies which bits of the base address are used in the comparison, and which are ignored. A mask bit setting of 1 indicates that the corresponding bit is used in the comparison, while a mask bit setting of 0 indicates that the corresponding bit is not used (masked) in the address comparison.

For example, if a 16 Mbyte memory is attached to SDRAM bank 2, mapping this physical memory into the processor (60x) bus address space (16 Mbyte aligned) of 0x5900_0000 - 0x59FF_FFFF requires the following settings:

- SD_B2_ADDR[0:15] = 0x5900
- SD_B2_MASK[0:15] = 0xFF00: indicates a 16 Mbyte block size

The settings represented in Table 50 are examples. They do not form a comprehensive list of all possible settings

Table 50: Memory Map to 60X Address Space

M[0:15]	Memory Size
0x8000	2 Gbyte
0xC000	1 Gbyte
0xE000	512 Mbyte
0xF000	256 Mbyte
0xF800	128 Mbyte
0xFC00	64 Mbyte
0xFE00	32 Mbyte
0xFF00	16 Mbyte
0xFFFF	64 kbyte

Table 51: SDRAM Memory Bank *x* Control and Status

Register Name: SD_B*x*_CTRL	Register Offset: 048/058/068/078

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	ECC_EN	ECC_CE	BUF	NBANK		A_MODE[0:2]		
8-15	T_RCD	Reserved	T_RP		Reserved	T_RAS[0:2]		
16-23	BMGT[0:3]				Reserved			ECC_UC
24-31	ECC_CO							

SD_B*x*_CTRL Description

Name	Type	Reset By	Reset State	Function
ECC_EN	R/W	PORESET_	0	ECC Checking and Correction 0 = disabled. ECC not used. 1 = enabled. ECC correction used.
ECC_CE	R/W	PORESET_	0	ECC Correction Mode 0 = disabled. Correctable errors are not corrected. 1 = enabled. Correctable errors are corrected.
BUF	R/W	PORESET_	0	Buffered (registered) SDRAM DIMM select 1 = Buffered DIMM used in this bank. 0 = Unbuffered DIMM used in this bank.
NBANK	R/W	PORESET_	11	Number of banks / chip selects 00 = 1 physical bank, 2 logical banks 01 = 1 physical bank, 4 logical banks 10 = 2 physical banks, 2 logical banks 11 = 2 physical banks, 4 logical banks
A_MODE	R/W	PORESET_	000	Addressing Mapping Mode: (see Chapter 4: “SDRAM Interface” on page 51) 000 = Mode 0 - 8 column bits 001 = Mode 1 - 9 column bits 010 = Mode 2 - 10 column bits 011 = Mode 3 - 11 column bits 100 = Mode 4 - 12 column bits others = reserved.

SD_B*x*_CTRL Description

Name	Type	Reset By	Reset State	Function
T_RCD	R/W	PORESET_	0	ACTV to READ/WRITE delay 0 = 2 clk 1 = 3 clk
T_RP[0:1]	R/W	PORESET_	01	PRE to ACTV delay 00 = reserved 01 = 2 clk 10 = 3 clk 11 = 4 clk.
T_RAS[0:2]	R/W	PORESET_	001	ACTV to PRE delay 000 = reserved 001 = 5 clk 010 = 6 clk 011 = 7 clk 100 = 8 clk 1xx = reserved.
BMGT[0:3]	R/W	PORESET_	1111	Bank Management, one bit per bank 0 = bank left open until miss or refresh 1 = bank always closed after access
ECC_UC	R/Write 1 to Clear	PORESET_	0	ECC uncorrectable Error Flag 0 = no uncorrectable error has occurred. 1 = uncorrectable error occurred. Write 1 to clear.
ECC_CO[0:7]	R/Write 1 to Clear	PORESET_	0	ECC correctable error occurred flag, byte lane 0-7 0 = no correctable error has occurred in this byte lane. 1 = correctable error occurred in this byte lane.

BUF: This bit is set to a 1 if a registered SDRAM DIMM is used, and 0 if an unregistered DIMM is used. A registered SDRAM DIMM contains a register on the control and address lines to reduce loading and enhance the timing margins for high speed operation.

NBANK: Two chip selects are provided for each SDRAM bank. This bit is to be set to 11 if both chip selects are to be used (DUAL bank DIMM), or a 00 if a single chip select is to be used (SINGLE bank DIMM).

A_MODE: Defines the mapping between processor (60x) bus address and SD_A[0:15], SD_B[0:1], and CS[0:1]. For more information, refer to Chapter 4: “SDRAM Interface” on page 51.

T_RCD, T_RP, T_RAS: These bits control timing to optimize SDRAM performance. These timings are specified as number of processor bus clocks (PB_CLKs). To determine the correct setting, consult the SDRAM data sheet and follow these steps:

1. Determine the SDRAM timing numbers in ns.
2. Divide the PB_CLK clock period
3. Round-up the number to the next integer number of PB_CLK periods

For example, if T_xx is specified as 17 ns and the PB_CLK period is 10 ns, the correct setting of T_xx is two PB_CLKs.

BMGT: There is one bank management bit per bank. When the BMGT bit is set to 0, the memory bank is left open until there is an access miss or a refresh. When the BMGT bit is set to 1, the memory bank is always closed after an access.

ECC_xx: When the DQM_EN field, in the SD_TIMING register, is not set and an ECC DIMM is used in the memory bank, ECC correction can be enabled. When ECC correction is enabled, the ECC_UC bit, in the SD_Bx_CTRL register, indicates if an uncorrectable error occurred. The ECC_CO[0:7] field indicates if a correctable error occurred and if correctable error occurred, the field indicates which byte lane has the error.

When the bank is in ECC mode and the ECC_CE bit is enabled, any single bit correctable errors are corrected before they are passed through. When the ECC_CE bit is disabled single bit correctable errors are logged, but the uncorrected (invalid) data is returned. Writing a 1 to the affected ECC_CO bit clears the bit.

The ECC pins must be connected to PowerPro in order to enable ECC protection.

Table 52: ROM Memory Bank *x* Address

Register Name: EE_B*x*_ADDR	Register Offset: 080/090/0A0/0B0

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	A							
8-15	A							
16-23	A							
24-31	Reserved					MUX		ENABLE

EE_B*x*_ADDR Description

Name	Type	Reset By	Reset State	Function
A[0:23]	R/W	PORESET_	see Table 53	FLASH/ROM Bank x Base Address. Qualified by EE_Bx_MASK.
MUX[29:30]	R/W	PORESET_	11	FLASH/ROM address multiplexing Determines where the FLASH/ROM address appears: 00 = EE_A[14:0] appears on {SD_BA[1:0], SD_A[12:0]}, while EE_A[22:15] appears on EE_DATA[0:7]. 01 = EE_A[29:15] appears on {SD_BA[1:0], SD_A[12:0]} when EE_AL1 is high. When EE_AL1 is low, EE_A[14:0] appears on {SD_BA[1:0], SD_A[12:0]}. 10 = EE_A[14:0] appears on {SD_BA[1:0], SD_A[12:0]} when EE_AL1 and EE_AL2 are low. EE_A[14:0] also appears on {SD_BA[1:0], SD_A[12:0]} when EE_AL2 is high; EE_A[29:15] appears on {SD_BA[1:0], SD_A[12:0]} when EE_AL2 is high. 11 = Same as ‘10’ setting, with the addition that EE_A[23:16] appears on EE_DATA[7:0] when EE_AL2 is high. Also EE_A[14:0] appears on {SD_BA[1:0], SD_A[12:0]} when EE_SELECT is high.
ENABLE	R/W	PORESET_	0	Bank Enable 0 = slot vacant 1 = slot filled with ROM like device

Table 53 shows the reset states for the A field in the EE_Bx_ADDR registers.

Table 53: Reset state of the A field in all EE_Bx_ADDR Registers

Register	A field Reset State
EE_B0_ADDR	0xFFF001
EE_B1_ADDR	0xFFFFFF
EE_B2_ADDR	0xFFFFFF
EE_B3_ADDR	0xFFFFFF

The four available FLASH/ROM banks are mapped to the processor (60x) bus address space through the A field in the EE_Bx_ADDR register and the Mask field of the EE_Bx_MASK register. The A field indicates the memory space base address. The M field indicates the memory block size.

This design requires that the memory space allocated be aligned with the size of the memory attached. For example, a 128 Mbyte DIMM can only be mapped to addresses on a 128 Mbyte boundary.

The M field specifies which bits of the base address are used in the comparison, and which are ignored. A mask bit setting of ‘1’ indicates that the corresponding bit is used in the comparison, while a mask bit setting of ‘0’ indicates that the corresponding bit is not used (it is masked) in the address comparison.

For example, if a 4 Mbyte memory is attached to FLASH/ROM bank 0. In order to map this physical memory into the processor (60x) bus address space (4 Mbyte aligned) of 0x1230_0000 - 0x123F_FFFF, the following settings must be used:

- EE_B0_ADDR[0:23] = 0x123000
- EE_B0_MASK[0:23] = 0xFFF000 (indicates a 1Mbyte block size)

Table 54: ROM Memory Bank x Address Mask

Register Name: EE_Bx_MASK	Register Offset: 084/094/0A4/0B4

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	M							
8-15	M							
16-23	M							
24-31	Reserved							

EE_Bx_MASK Description

Name	Type	Reset By	Reset State	Function
M[0:23]	R/W	PORESET_	see Table 55	FLASH/ROM address mask Used to qualify EE_Bx_ADDR.

Table 55 shows the reset states for the M field in the EE_Bx_MASK registers.

Table 55: Reset state of the M Field in all EE_Bx_MASK Registers

Register	A Field Reset State
EE_B0_MASK	0xFFF000
EE_B1_MASK	0xFFFFFF
EE_B2_MASK	0xFFFFFF
EE_B3_MASK	0xFFFFFF

The four available FLASH/ROM banks are mapped to the processor (60x) bus address space through the A field in the EE_Bx_ADDR register and the Mask field in the EE_Bx_MASK register. The A field indicates the memory space base address. The M field indicates the memory block size.

This design requires that the memory space allocated be aligned with the size of the memory attached. For example, a 128 Mbyte DIMM can only be mapped to addresses on a 128 Mbyte boundary.

The M field specifies which bits of the base address are used in the comparison, and which are ignored. A mask bit setting of 1 indicates that the corresponding bit is used in the comparison, while a mask bit setting of 0 indicates that the corresponding bit is not used (it is masked) in the address comparison.

For example, if a 1 Mbyte memory is attached to ROM bank 0. In order to map this physical memory into the processor (60x) bus address space (1 Mbyte aligned) of 0x1230_0000 - 0x123F_FFFF, the following settings must be used:

- EE_B0_ADDR[0:23] = 0x123000
- EE_B0_MASK[0:23] = 0xFFF000 (indicates a 1Mbyte block size)

Table 56: ROM Memory Bank *x* Control

Register Name: EE_B*x*_CTRL	Register Offset: 088/098/0A8/0B8

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	BM	FWE	WAIT					
8-15	CSON		OEON		WEON		PORT	WEOFF
16-23	THRD				THWR			
24-31	FWT				RE	ARE	WIDTH	

EE_B*x*_CTRL Description

Name	Type	Reset By	Reset State	Function
BM	R/W	PORESET_	see Table 57	Burst Mode 0 = Burst mode disable 1 = Burst mode enable
FWE	R/W	PORESET_	see Table 58	Flash Write Enable
WAIT[0:5]	R/W	PORESET_	see Table 59	With BM=0, WAIT = wait states on all non-burst transfers. Number of clocks from address valid to the deassertion of CS_ is 1+WAIT. With BM=1, WAIT = next wait.
CSON[0:1]	R/W	PORESET_	see Table 60	Chip select ON timing; with respect to address valid. 00 = CS_ valid with address valid. 01 = CS_ valid 1 clock after address valid. 10 = CS_ valid 2 clocks after address valid. 11 = CS_ valid 3 clocks after address valid.
OEON[0:1]	R/W	PORESET_	see Table 61	Output Enable ON Timing, measured with respect to CS_ assertion. 00 = OE_ valid with chip select valid. 01 = OE_ valid 1 clock after chip select valid. 10 = OE_ valid 2 clocks after chip select valid. 11 = OE_ valid 3 clocks after chip select valid.

EE_B*x*_CTRL Description

Name	Type	Reset By	Reset State	Function
WEON[0:1]	R/W	PORESET_	see Table 62	Write Enable ON timing, measured with respect to CS_ assertion. 00 = WE_ valid with chip select valid. 01 = WE_ valid 1 clock after chip select valid. 10 = WE_ valid 2 clocks after chip select valid. 11 = WE_ valid 3 clocks after chip select valid.
PORT	R/W	PORESET_	see Table 63	FLASH/ROM data port 0 = use dedicated 8-bit port 1 = use SDRAM data bus
WEOFF	R/W	PORESET_	see Table 64	Write Enable OFF timing, measured with respect to EE_CS_ de-assertion. 0 = EE_WE_ de-asserts with chip select de-assertion. 1 = EE_WE_ de-asserts 1 clock prior to chip select de-assertion.
THRD[0:3]	R/W	PORESET_	see Table 65	Transfer hold on reads Number of cycles that CS_ and related signals are held at the end of a read cycle; also the minimum time before the next access can occur following a read.
THWR[0:3]	R/W	PORESET_	see Table 66	Transfer hold on writes Number of cycles that CS_ and related signals are held at the end of a write cycle; also the minimum time before the next access can occur following a write.
FWT[0:3]	R/W	PORESET_	see Table 67	First Wait (BM = 1) Initial wait states on subsequent accesses of all burst transfers. The number of cycles from address valid to the next address valid including the time for latching the ROM/Perhipheral address is 1+FWT for the first access to a bursting device.
RE	R/W	PORESET_	see Table 68	Ready Enable 0 = Throttling via READY input is disabled. 1 = Throttling via READY input is enabled.

EE_B*x*_CTRL Description

Name	Type	Reset By	Reset State	Function
ARE	R/W	PORESET_	see Table 69	Asynchronous Ready 0 = READY input is synchronous (data sampled 1 clock after READY input is sampled). 1 = READY input is asychronous (data sampled 3 clocks after READY input is sampled).
WIDTH[0:1]	R/W	PORESET_	see Table 70	Bank width 00 = 8-bit (only option if PORT = 0). 01 = 16-bit 10 = 32-bit 11 = 64-bit

PowerPro provides access to FLASH/ROM banks. Each of the FLASH/ROM banks has a general purpose chip select machine controlling READ, WRITE, OE, WE, and CS outputs. PowerPro can control any type of ROM, EEPROM, FLASH, SRAM or SRAM/ROM-like device through the settings in the EE_Bx_CTRL register.

The EE_B1_CTRL register is involved in the PowerPro's power-up options. Refer to "System Boot" on page 121 for more information.

For more information on attaching and configuring FLASH/ROM devices to PowerPro. refer to Chapter 5: "FLASH/ROM Interface" on page 69.

Table 57 shows the reset states for the BM field in the EE_Bx_CTRL registers.

Table 57: Reset state of the BM Field in all EE_Bx_CTRL Registers

Register	BM Field Reset State
EE_B0_CTRL	0
EE_B1_CTRL	0
EE_B2_CTRL	0
EE_B3_CTRL	0

Table 58 shows the reset states for the FWE field in the EE_Bx_CTRL registers.

Table 58: Reset state of the FWE Field in all EE_Bx_CTRL Registers

Register	FWE Field Reset State
EE_B0_CTRL	0
EE_B1_CTRL	0
EE_B2_CTRL	0
EE_B3_CTRL	0

Table 59 shows the reset states for the WAIT field in the EE_Bx_CTRL registers.

Table 59: Reset state of the WAIT Field in all EE_Bx_CTRL Registers

Register	WAIT Field Reset State
EE_B0_CTRL	0x010100
EE_B1_CTRL	0x000000
EE_B2_CTRL	0x000000
EE_B3_CTRL	0x000000

Table 60 shows the reset states for the CSON field in the EE_Bx_CTRL registers.

Table 60: Reset state of the CSON Field in all EE_Bx_CTRL Registers

Register	CSON field Reset State
EE_B0_CTRL	0x01
EE_B1_CTRL	0x00
EE_B2_CTRL	0x00
EE_B3_CTRL	0x00

Table 61 shows the reset states for the OEON field in the EE_Bx_CTRL registers.

Table 61: Reset state of the OEON Field in all EE_Bx_CTRL Registers

Register	OEON field Reset State
EE_B0_CTRL	0x10
EE_B1_CTRL	0x00
EE_B2_CTRL	0x00
EE_B3_CTRL	0x00

Table 62 shows the reset states for the WEON field in the EE_Bx_CTRL registers.

Table 62: Reset state of the WEON Field in all EE_Bx_CTRL Registers

Register	WEON Field Reset State
EE_B0_CTRL	0x01
EE_B1_CTRL	0x00
EE_B2_CTRL	0x00
EE_B3_CTRL	0x00

Table 63 shows the reset states for the PORT field in the EE_Bx_CTRL registers.

Table 63: Reset state of the PORT Field in all EE_Bx_CTRL Registers

Register	PORT Field Reset State
EE_B0_CTRL	Power-up option
EE_B1_CTRL	0
EE_B2_CTRL	0
EE_B3_CTRL	0

Table 64 shows the reset states for the WEOFF field in the EE_Bx_CTRL registers.

Table 64: Reset state of the WEOFF field in all EE_Bx_CTRL Registers

Register	WEOFF Field Reset State
EE_B0_CTRL	1
EE_B1_CTRL	0
EE_B2_CTRL	0
EE_B3_CTRL	0

Table 65 shows the reset states for the THRD field in the EE_Bx_CTRL registers.

Table 65: Reset state of the THRD field in all EE_Bx_CTRL Registers

Register	THRD Field Reset State
EE_B0_CTRL	0x0100
EE_B1_CTRL	0x0000
EE_B2_CTRL	0x0000
EE_B3_CTRL	0x0000

Table 66 shows the reset states for the THWR field in the EE_Bx_CTRL registers.

Table 66: Reset state of the THWR field in all EE_Bx_CTRL Registers

Register	THWR Field Reset State
EE_B0_CTRL	0x0100
EE_B1_CTRL	0x0000
EE_B2_CTRL	0x0000
EE_B3_CTRL	0x0000

Table 67 shows the reset states for the FWT field in the EE_Bx_CTRL registers.

Table 67: Reset state of the FWT field in all EE_Bx_CTRL Registers

Register	FWT Field Reset State
EE_B0_CTRL	0x1011
EE_B1_CTRL	0x000
EE_B2_CTRL	0x000
EE_B3_CTRL	0x000

Table 68 shows the reset states for the RE field in the EE_Bx_CTRL registers.

Table 68: Reset state of the RE field in all EE_Bx_CTRL Registers

Register	RE Field Reset State
EE_B0_CTRL	Power-up option
EE_B1_CTRL	0
EE_B2_CTRL	0
EE_B3_CTRL	0

Table 68 shows the reset states for the ARE field in the EE_Bx_CTRL registers.

Table 69: Reset state of the ARE field in all EE_Bx_CTRL Registers

Register	ARE Field Reset State
EE_B0_CTRL	0
EE_B1_CTRL	0
EE_B2_CTRL	0
EE_B3_CTRL	0

Table 68 shows the reset states for the WIDTH field in the EE_Bx_CTRL registers.

Table 70: Reset state of the WIDTH field in all EE_Bx_CTRL Registers

Register	WIDTH Field Reset State
EE_B0_CTRL	Power-up option
EE_B1_CTRL	0x00
EE_B2_CTRL	0x00
EE_B3_CTRL	0x00

.

Table 71: I2C*x*_CSR

Register Name: I2C*x*_CSR	Register Offset: 0C0, 0C4

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	ADDR							
8-15	DATA							
16-23	DEV_CODE				CS			RW
24-31	ACT	ERR	Reserved					

I2C*x*_CSR Description

Name	Type	Reset By	Reset State	Function
ADDR[0:7]	R/W	PORESET_	0	Specifies I^2C device to be addressed.
DATA[0:7]	R/W	PORESET_	0	Specifies the required data for a write. Holds the data at the end of a read
DEV_CODE [0:3]	R/W	PORESET_	0b1010	Device Select. I^2C 4-bit device code.
CS[2:0]	R/W	PORESET_	0	Chip Select.
RW	R/W	PORESET_	0	0 = read 1 = write.
ACT	R	PORESET_	0	I^2C Interface active 0 = not active 1 = active
ERR	R/W1clr	PORESET_	0	Error 0 = no error, 1 = error condition.

This register supports the PowerPro I^2C Interface.

ACT: A I^2C bus cycle is initiated by writing to this register. Software must wait for the ACT bit to be 0 before starting a new I^2C bus cycle. When the ACT bit is 1, writes to this register have no effect, and the DATA field is undefined.

The ACT bit is set due to the following reasons:

1. The I^2C Interface is busy servicing a read or a write as a result of a write to this register.
2. The I^2C Interface is busy loading registers at the end of reset

ERR: This bit is set if PowerPro is unable to complete a requested EEPROM read or write. This bit is valid when ACT is low. This bit must be cleared before attempting another EEPROM access.

Table 72: Watchdog Timer Control

Register Name: WD_CTRL	Register Offset: 0F0

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	ENABLE	Reserved						
8-15	Reserved							
16-23	Reserved							
24-31	Reserved							WD_RST

WD_CTRL Description

Name	Type	Reset By	Reset State	Function
ENABLE	R/W	PORESET_	0	Watchdog timer enable.
WD_RST	W1Set	PORESET_	0	Watchdog timer count reset. Resets the watchdog count back to WD_TIMEOUT.

The watchdog timer provides fault catching in real time operating systems.

When watchdog timer is enabled, the timer counts down from the Watchdog Timer Initial Value (WDT) in the WD_TIMEOUT register (see Table 73 on page 196). The timer counts down from the value in the WDT field to 0. If the counter reaches 0, a watchdog time-out interrupt is asserted. The WD_RST bit resets the watchdog timer to the value in the WDT field when a 1 is written to the bit.

At a 100 MHz (10 ns) clock period, the 32-bit watchdog timer has about 42 seconds as a maximum setting. Reading the Current Watchdog Timer Count (WDC) in the Watchdog Timer Count (WD_COUNT) register returns the current value of the watchdog timer.

Any of the following actions reload the watchdog timer:

- setting ENABLE to 1
- setting WD_RST to 1
- writing to WDT reloads the watchdog timer to the original WDT value

When the watchdog timer functionality is not required, this counter can be used as a general purpose timer

Table 73: Watchdog Timer Timeout

Register Name: WD_TIMEOUT	Register Offset: 0F4

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	WDT							
8-15	WDT							
16-23	WDT							
24-31	WDT							

WD_TIMEOUT Description

Name	Type	Reset By	Reset State	Function
WDT[0:31]	R/W	PORESET_	0	Watchdog timer initial value. The watchdog timer counts down from this value to zero. If it reaches zero, a non-maskable interrupt is asserted.

The watchdog timer provides fault catching in real time operating systems.

When watchdog timer is enabled, the timer counts down from the Watchdog Timer Initial Value (WDT) in the WD_TIMEOUT register (see Table 73 on page 196). The timer counts down from the value in the WDT field to 0. If the counter reaches 0, a watchdog time-out interrupt is asserted. The WD_RST bit resets the watchdog timer to the value in the WDT field when a 1 is written to the bit.

At a 100 MHz (10 ns) clock period, the 32-bit watchdog timer has about 42 seconds as a maximum setting. Reading the Current Watchdog Timer Count (WDC) in the Watchdog Timer Count (WD_COUNT) register returns the current value of the watchdog timer.

Any of the following actions reload the watchdog timer:

- setting ENABLE to 1
- setting WD_RST to 1
- writing to WDT reloads the watchdog timer to the original WDT value

When the watchdog timer functionality is not required, this counter can be used as a general purpose timer

Table 74: Watchdog Timer Count

Register Name: WD_COUNT	Register Offset: 0F8

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	WDC							
8-15	WDC							
16-23	WDC							
24-31	WDC							

WD_COUNT Description

Name	Type	Reset By	Reset State	Function
WDC[0:31]	R	PORESET_ /WD_RST	The value of the WDT field in the WD_TIMEOUT register (see Table 73 on page 196)	Current watchdog timer count. If the count reaches zero and the watchdog timer is enabled, a non-maskable interrupt is generated.

Table 75: General Purpose Timer Base Count

Register Name: GPT_COUNT	Register Offset: 100

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	GPTC							
8-15	GPTC							
16-23	GPTC							
24-31	GPTC							

GPT_COUNT Description

Name	Type	Reset By	Reset State	Function
GPTC[0:31]	R/W	PORESET_	0	Base count which all general purpose timer functions work from.

The general purpose system timer is a free running 32-bit counter.

GPTC: This field has the current value of the counter. The counter increments once per system clock. All GPT functions are based-on this reference count. Writing to GPTC resets the counter to the new value. There is no start or stop mechanism on the counter; it is continuously running.

Table 76: General Purpose Timer Capture Events

Register Name: GPT_CAPTURE	Register Offset: 104

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	SEVT				Reserved			
8-15	SD_AM				Reserved			
16-23	EE_AM				Reserved			
24-31	PB_AM				Reserved			

GPT_CAPTURE Description

Name	Type	Reset By	Reset State	Function
SEVT[0:3]	R/W	PORESET_	0	Software Capture Event. When a 1 is written to the bit location corresponding with a capture timer, an event is generated which implements the event timer. Programming a 1 captures the current value of GPT_COUNT into the corresponding capture timer register.
SD_AM[0:3]	R/W	PORESET_	0	SDRAM Bank Address Match Capture Enable. Activity on each of the four available SDRAM banks can be captured in the four corresponding capture timers. 0 = SDRAM bank activity is not captured. 1 = SDRAM bank activity time is captured.

GPT_CAPTURE Description

Name	Type	Reset By	Reset State	Function
EE_AM[0:3]	R/W	PORESET_	0	FLASH/ROM Bank Address Match Capture Enable. Activity on each of the four available FLASH/ROM banks can be captured in the four corresponding capture timers. 0 = FLASH/ROM bank activity is not captured. 1 = FLASH/ROM bank activity time is captured.
PB_AM[0:3]	R/W	PORESET_	0	Processor (60x) bus address match capture enable. If this bit is set, and a transaction happens on the processor (60x) bus which matches PB_AM_ADDR qualified with PB_AM_MASK, the time of that occurrence is captured in the corresponding capture register.

PowerPro has four capture time registers. The time of the free running counter GPT_COUNT.GPTC is copied into the appropriate capture register if an enabled capture event occurs. The four capture registers are GPT_TT0 - GPT_TT3.

There are different types of capture events which cause the GPTC value to be copied into the capture register. The different types of capture events are as follows:

- A software capture event, which is caused by setting one or more of SEVT[0:3] bits to one.
- Activity on one of the four SDRAM banks combined with SD_AM[0:3] bits being set. For example, to capture the timer time that the last activity occurred on SDRAM bank #2, set SD_AM[2] to 1.
- Activity on one of the four FLASH/ROM banks combined with EE_AM[0:3] bits being set.
- A match on the processor (60x) bus address match register, combined with PB_AM[0:3] bits being set.

A match on this register does not necessarily require that the matched address lie within address space normally claimed by PowerPro

Table 77: General Purpose Timer Interrupt Control

Register Name: GPT_INT	Register Offset: 10C

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	T_IEN				Reserved			
8-15	Reserved							
16-23	C_IEN				Reserved			
24-31	Reserved							

GPT_INT Description

Name	Type	Reset By	Reset State	Function
T_IEN[0:3]	R/W	PORESET_	0	GPT Trigger Interrupt Enable 0 = Interrupt disabled for GPT Trigger 1 = Interrupt enabled for GPT Trigger
C_IEN[0:3]	R/W	PORESET_	0	GPT Compare Interrupt Enabled 0 = Interrupt disabled for GPT Compare 1 = Interrupt enabled for GPT Compare

General purpose timer triggers or compare events can be mapped to cause an interrupt to occur. If the corresponding event occurs (a timer trigger event or compare match) and the interrupt enable bit is set, and interrupt for that event is be generated.

Table 78: General Purpose Timer Interrupt Status

Register Name: GPT_ISTATUS	Register Offset: 110

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	T_STAT				Reserved			
8-15	Reserved							
16-23	C_STAT				Reserved			
24-31	Reserved							

GPT_ISTATUS Description

Name	Type	Reset By	Reset State	Function
T_STAT[0:3]	R/W1Clr	PORESET_	0	Trigger status 0 = No trigger has occurred. 1 = Trigger Occurred.
C_STAT[0:3]	R/W1Clr	PORESET_	0	Compare status 0 = Compare match has not occurred. 1 = Compare match has occurred.

The status of trigger and compare events are stored here. When a trigger or capture event occurs, the corresponding bit is set to 1. The bit remains at one until cleared by having a 1 written to it.

Table 79: General Purpose Timer Trigger x

Register Name: GPT_Tx	Register Offset: 120/124/128/12C

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	TT							
8-15	TT							
16-23	TT							
24-31	TT							

GPT_Tx Description

Name	Type	Reset By	Reset State	Function
TT[0:31]	R	PORESET_	0	General purpose timer trigger time. The time that the last trigger event occurred is stored in this register.

The Four trigger registers capture the time a trigger event occurs. The trigger register stores the time that the general purpose timer was at when the corresponding trigger occurred. Trigger events are enabled or disabled through the GPT_CAPTURE register (see page 199).

Table 80: General Purpose Timer Compare x

Register Name: GPT_Cx	Register Offset: 140/144/148/14C

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	CT							
8-15	CT							
16-23	CT							
24-31	CT							

GPT_Cx Description

Name	Type	Reset By	Reset State	Function
CT[0:31]	R/W	PORESET_	0xFFFF	General purpose timer compare value. This value is compared against the current GPT value. A match triggers the compare GPT event.

The four compare registers compare against the current value of the GPTC[0:31] field in the GPT_COUNT register. If the current value of the general purpose timer counter matches the compare value of GPT_CTx.CT[0:31], a compare event is generated. The compare event is logged in CSTAT bit in the GPT_ISTATUS. If C_IEN enabled in the CPT_INT register, an interrupt is also generated.

CT: The CT field is qualified with the compare mask. When a corresponding mask bit is clear, that bit is used in the CT field. If it is set, the bit is ignored and assumed to always to match.

For example, to receive notification every time the counter passes to 0xxx3_0000 (x is any value) the following settings are program:

- CT[0:31] = 0x0003_0000
- CM[0:31] = 0xFFF3_0000

A match occurs when the counter reaches 0x0003_0000, 0x0007_0000, 0x000A_0000, etc. For example, the time 0x0003_1000 does not match.

Table 81: General Purpose Timer Compare Mask x

Register Name: GPT_Mx	Register Offset: 160/164/168/16C

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	CM							
8-15	CM							
16-23	CM							
24-31	CM							

GPT_Mx Description

Name	Type	Reset By	Reset State	Function
CM[0:31]	R/W	PORESET_	0	General purpose timer compare mask.

The four compare registers compare against the current value of the GPTC[0:31] field in the GPT_COUNT register. If the current value of the general purpose timer counter matches the compare value of GPT_CTx.CT[0:31], a compare event is generated. The compare event is logged in CSTAT bit in the GPT_ISTATUS. If C_IEN enabled in the CPT_INT register, an interrupt is also generated.

CT: The CT field is qualified with the compare mask. When a corresponding mask bit is clear, that bit is used in the CT field. If it is set, the bit is ignored and assumed to always to match.

For example, to receive notification every time the counter passes to 0xxx3_0000 (x is any value) the following settings are program:

- CT[0:31] = 0x0003_0000
- CM[0:31] = 0xFFF3_0000

A match occurs when the counter reaches 0x0003_0000, 0x0007_0000, 0x000A_0000, etc. For example, the time 0x0003_1000 does not match.

Table 82: Interrupt Controller Status

Register Name: INT_STATUS	Register Offset: 180

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	STAT							
8-15	STAT							
16-23	STAT							
24-31	STAT							

INT_STATUS Description

Name	Type	Reset By	Reset State	Function
STAT[0:31]	R/W1Clr	PORESET_	0	Interrupt status. This status is independent of INT_ENABLE; this register reports the interrupt status regardless of that interrupt being enabled. 0 = interrupt has not occurred 1 = interrupt has occurred Write 1 to clear interrupt.

PowerPro has a 32-input interrupt controller. A variety of internal and external events are mapped to each bit of the interrupter. For more information refer to Chapter 11: “Interrupt Controller” on page 105.

The status of each of the interrupt channels is reported in two registers: INT_STATUS and INT_MSTATUS. INT_STATUS reports the status of the interrupt sources regardless of INT_ENABLE settings, while INT_MSTATUS masks INT_STATUS with INT_ENABLE to provide masked status results. Writing a 1 to INT_STATUS clears the associated interrupt flag.

Table 83: Interrupt Controller Masked Status

Register Name: INT_MSTATUS	Register Offset: 184

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	MSTAT							
8-15	MSTAT							
16-23	MSTAT							
24-31	MSTAT							

INT_MSTATUS Description

Name	Type	Reset By	Reset State	Function
MSTAT[0:31]	R	PORESET_	0	Masked interrupt status, result of INT_STATUS and INT_ENABLE. 0 = interrupt has not occurred or is masked 1 = interrupt has occurred and is not masked

Table 84: Interrupt Controller Enable

Register Name: INT_ENABLE	Register Offset: 188

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	IE							
8-15	IE							
16-23	IE							
24-31	IE							

INT_ENABLE Description

Name	Type	Reset By	Reset State	Function
IE[0:31]	R/W	PORESET_	0x0000	Interrupt Enable 0 = interrupt disabled, generation to the processor is suppressed, but detection is still active. 1 = interrupt enabled. This causes an interrupt to be generated to the processor.

IE: Each of the 32 interrupt sources can be mapped to generate one of two external interrupt outputs. If the corresponding bit in the IE field is set and an interrupt occurs, that interrupt generates either the PB_INT signal or EXT_INT_OUT signal depending on the setting in Interrupt Generation Type (IGTYPE) field in the INT_GENERATE register (see page 209).

When PowerPro is initialized (at power-up) these interrupts can transition. This causes interrupts to be pending during initialization.

Table 85: Interrupt Controller Cycle Generation Type

Register Name: INT_GENERATE	Register Offset: 18C

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	IGTYPE							
8-15	IGTYPE							
16-23	IGTYPE							
24-31	IGTYPE							

INT_GENERATE Description

Name	Type	Reset By	Reset State	Function
IGTYPE[0:31]	R/W	PORESET_	0x0000	Interrupt Generation type. 0 = Generate a PB_INT_ signal 1 = Generate an EXT_INT_ signal

IGTYPE: Each of the 32 interrupt sources can be mapped to generate one of two external interrupt outputs. If the corresponding bit in the Interrupt Enable (IE) field in the INT_ENABLE register is set and an interrupt occurs, that interrupt generates either the PB_INT_ signal or EXT_INT_ signal. The signal generation depends on the setting in Interrupt Generation Type (IGTYPE) field in the INT_GENERATE register (see page 209).

Table 86: Interrupt Controller Polarity

Register Name: INT_POLARITY	Register Offset: 190

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	IPOL							
8-15	IPOL							
16-23	IPOL							
24-31	IPOL							

INT_POLARITY Description

Name	Type	Reset By	Reset State	Function
IPOL[0:31]	R/W	PORESET_	0xFFFF_FFFF	Interrupt Polarity 0 = active low (level sensitive), or negative edge. 1 = active high (level sensitive), or positive edge.

Each of the 32 interrupt sources can be specified as edge or level sensitive. If the sources are specified as edge sensitive, an interrupt event can be generated on the presence of a positive or negative edge. If the sources are specified as level sensitive, an interrupt event can be continuously generated on the presence of a high or low level.

Interrupts that are generated internal to PowerPro, for example the UARTs, have different polarity and triggering attributes. These different attributes must be accounted for when the IPOL field is programmed.

Table 87: Interrupt Controller Trigger Type

Register Name: INT_TRIGGER	Register Offset: 194

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	ITTYPE							
8-15	ITTYPE							
16-23	ITTYPE							
24-31	ITTYPE							

INT_TRIGGER Description

Name	Type	Reset By	Reset State	Function
ITTYPE[0:31]	R/W	PORESET_	0xFFFF_FFFF	Interrupt Trigger Type. 0 = level sensitive 1 = edge sensitive

Each of the 32 interrupt sources can be specified as edge or level sensitive. If the sources are specified as edge sensitive, an interrupt event can be generated on the presence of a positive or negative edge. If the sources are specified as level sensitive, an interrupt event can be continuously generated on the presence of a high or low level..

Interrupts that are generated internal to PowerPro, for example the UARTs, have different polarity and triggering attributes. These different attributes must be accounted for when the ITTYPE field is programmed.

Table 88: Interrupt Controller Vector Base Address

Register Name: INT_VBADDR	Register Offset: 198

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	VA							
8-15	VA							
16-23	VA							
24-31	VA							

INT_VBADDR Description

Name	Type	Reset By	Reset State	Function
VA[0:18]	R/W	PORESET_	0	Vector Base Address Used for generating the V field in the INT_VECTOR register.
VA[19:31]	R	PORESET_	0	Vector Base Address Used for generating the V field in the INT_VECTOR register.

Table 89: Interrupt Controller Vector Increment

Register Name: INT_VINC	Register Offset: 19C

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	Reserved							PRI
8-15	Reserved							
16-23	Reserved							
24-31	Reserved						VINC	

INT_VINC Description

Name	Type	Reset By	Reset State	Function
PRI	R/W	PORESET_	0	Priority ordering. 0 = INT 0 is most significant. 1 = INT 31 is most significant.
VINC[0:1]	R/W	PORESET_	0	Vector Increment. The calculated interrupt vector is the vector base address + increment amount * interrupt number. 00 = 0x100 is space between ISR routines 01 = 0x200 is space between ISR routines 10 = 0x400 is space between ISR routines 11 = 0x800 is space between ISR routines

VINC[1] has the dual function of controlling the upper-word register access. Normally, register access is restricted to 4 byte (32-bit) accesses, with anything larger resulting in an error condition which generates TEA (when TEA generation is enabled in the PowerPro). If TEA generation is disabled in PowerPro, an eight byte register read returns the four byte register addressed replicated in the lower and upper word. When VINC[1] = 0 and an eight byte read is performed, the register at offset (PB_REG_ADDR & 0xff8) is returned (32-bit value) replicated in the upper and lower 32-bit words. When VINC[1] = 1 and an eight byte read is performed, the register at offset (PB_REG_ADDR & 0xff8) + 0x004 is returned (32-bit value) replicated in the upper and lower 32-bit words.

For example, an eight byte read to REG_BADDR + 0x198 when VINC[1] = 0 returns {INT_VBADDR, INT_VBADDR}. An eight byte read to REG_BADDR + 0x198 when VINC[1] = 1 returns {INT_VINC, INT_VINC}. The TEA_EN bit, in the PB_GEN_CTRL register, must be 0 in this mode.

Table 90: Interrupt Controller Incremented Vector Base Address

Register Name: INT_VECTOR	Register Offset: 1A0

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	V							
8-15	V							
16-23	V							
24-31	V							

INT_VECTOR Description

Name	Type	Reset By	Reset State	Function
V[0:31]	R	PORESET_	0	Interrupt vector. INT_VECTOR = INT_VBADDR + (int #) * INT_VINC.

This register is provided for software use. It is a constantly regenerated value of INT_VBADDR[0:31] + (interrupt #) * INT_VINC.

The register can be used as a pointer to an interrupt service routine. If the register is used in this way, the set of interrupt service routines must be placed at INT_VBADDR[0:31]. Each interrupt service routine is separated by VINC - 0x100, 0x200, 0x400, or 0x800 in address space. Reading this register is a useful way of finding the code to handle a generic PB_INT signal.

Table 91: Interrupt Controller Software Set

Register Name: INT_SOFTSET	Register Offset: 1A4

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	SINT							
8-15	SINT							
16-23	SINT							
24-31	SINT							

INT_SOFTSET Description

Name	Type	Reset By	Reset State	Function
SINT[0:31]	R/W	PORESET_	0	Software Interrupt. Writing a 1 to a bit position has the same effect as toggling the corresponding interrupt line.

The INT_SOFTSET register is uses as a software debugging tool. Programming a bit to 1 in INT_SOFTSET has the effect as toggling the corresponding interrupt line.

Toggling means setting a bit.

Table 92: UART*x* Receive / Transmit Data

Register Name: UARTx_RX_TX	Register Offset: 1B0/1C0

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	D							

UART*x*_RBR_THR Description

Name	Type	Reset By	Reset State	Function
D[0:7]	R/W	PORESET_	0	Read: receive data Write: send data

The UARTx_RX_TX register is located at register offset 1B0/1C0 when the Divisor Latch Access (DLAB) bit in the UARTx Line Control register (see page 227) is set to 0. When the (DLAB) bit is set to 1 and register offset 1B0/1C0 is accessed the UARTx_ DLM register is accessed. Refer to Table 93 on page 218 for more information on this register.

When the Divisor Latch Access (DLAB) bit is set, writes to UART offset 0x0 and 0x1 set the baud rate divisor. When the DLAB bit is set to 1, the UARTx_RX_TX and UARTx_IER registers are not accessible. The register space is shared between the UARTx_DLM register and the UARTx_Rx_Tx at offset 1B0 (Table 92 on page 216 and Table 93 on page 218) and the UART1_DLL register and the UARTx_IER register (Table 94 on page 219 and Table 95 on page 221) at offset 1B1.

Writing 1 to the DLAB bit enables the UARTx_DLM or UARTx_DLL (depending on reads and writes) and disables the other registers that share their register offset.

A byte written to this register is transmitted by the UART when the Transmitter Holding (THRE) bit in the UARTx Line Status (LSR) register (see page 230) is set to 0. This setting indicates there is space in the transmitting FIFO. Writing to this register when the THRE bit is set to a 1 causes data loss.

The UART has a 16-byte entry FIFO. If this FIFO is enabled, the THRE bit must be checked to make sure it is set to 0. After the THRE bit is checked the 16-byte writes can be started.

If the Data ready Flag (DR) bit in the LSR register is set, then a read to this register returns the byte of data received. Reads to this register when the DR bit is set to 0 return unpredictable data.

UART registers are only byte-accessible.

Table 93: UART*x* Divisor Latch (DLM)

Register Name: UART*x*_DLM	Register Offset: 1B0/1C0

Bits	Function							
	8	9	10	11	12	13	14	15
8-15	B							

UART*x*_DLM Description

Name	Type	Reset By	Reset State	Function
B[8:15]	R/W	PORESET_		Baud Rate Divisor Latches

The UARTx_DLM register is located at register offset 1B0/1C0 when the Divisor Latch Access (DLAB) bit in the UARTx Line Control register (see page 227) is set to 1. When the DLAB bit is 0 the UARTx_ RX_TX register is accessed from this register offset. Refer to Table 92 on page 216 for more information on this register.

When the Divisor Latch Access (DLAB) bit is set, writes to UART offset 0x0 and 0x1 set the baud rate divisor. When the DLAB bit is set to 1, the UARTx_RX_TX and UARTx_IER registers are not accessible. The register space is shared between the UARTx_DLM register and the UARTx_Rx_Tx at offset 1B0 (Table 92 on page 216 and Table 93 on page 218) and the UARTx_DLL register and the UARTx_IER register (Table 94 on page 219 and Table 95 on page 221) at offset 1B1.

Writing 1 to the DLAB bit enables the UARTx_DLM or UARTx_DLL (depending on reads and writes) and disables the other registers that share their register offset.

The output frequency of the Baud Generator is represented in the following equation:

- (Frequency/ (16 * Baud))-1

Two 8-bit latches store the divisor in a 16-bit binary format. These divisor latches must be loaded during initialization to ensure proper operation of the Baud Generator. Upon loading either of the Divisor Latches, a 16-bit Baud counter is immediately loaded.

Table 94: UART*x* Interrupt Enable

Register Name: UART*x*_IER	Register Offset: 1B1/1C1

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	ERBFI	ETBEI	ELSI	Reserved				

UART*x*_IER Description

Name	Type	Reset By	Reset State	Function
ERBFI	R/W	PORESET_	0	Enable Received Data Available Interrupt
ETBEI	R/W	PORESET_	0	Enable Transmitter Holding Register Empty Interrupt
ELSI	R/W	PORESET_	0	Enable Receiver Line Status Interrupt

The UARTx_IER register is located at register offset 1B1/1C1 when the Divisor Latch Access (DLAB) bit in the UARTx Line Control register (see page 227) is set to 0. When the (DLAB) bit is set to 1 and register offset 1B1/1C1 is accessed the UARTx_DLL register is accessed. Refer to Table 95 on page 221 for more information on this register.

When the Divisor Latch Access (DLAB) bit is set, writes to UART offset 0x0 and 0x1 set the baud rate divisor. When the DLAB bit is set to 1, the UARTx_RX_TX and UARTx_IER registers are not accessible. The register space is shared between the UARTx_DLM register and the UARTx_Rx_Tx at offset 1B0 (Table 92 on page 216 and Table 93 on page 218) and the UART1_DLL register and the UART1_IER register (Table 94 on page 219 and Table 95 on page 221) at offset 1B1.

Writing 1 to the DLAB bit enables the UARTx_DLM or UARTx_DLL (depending on reads and writes) and disables the other registers that share their register offset.

This register enables the three types of UART interrupts. Each interrupt can individually activate the interrupt controller. It is possible to totally disable the interrupt system by resetting bits 0 through 2 of the Interrupt Enable Register (IER). Setting bits of the IER register to a logic 1, enables the selected interrupt(s). Disabling an interrupt prevents it from activating the interrupt controller. All other system functions operate in their normal manner, including the setting of the Line Status registers. For more information on PowerPro interrupts, please refer to Chapter 11: “Interrupt Controller” on page 105.

ERBFI: This bit enables the Received Data Available Interrupt (and timeout interrupts in the FIFO mode) when set to logic 1.

ETBEI: This bit enables the Transmitter Holding Register Empty Interrupt when set to logic 1.

ELSI: This bit enables the Receiver Line Status Interrupt when set to logic 1.

UART registers are only byte-accessible.

Table 95: UART*x* Divisor Latch (DLL)

Register Name: UART*x*_DLL	Register Offset: 1B1/1C1

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	B							

UART*x*_DLL Description

Name	Type	Reset By	Reset State	Function
B[0:7]	R/W	PORESET_		Baud Rate Divisor Latches

The UARTx_DLL register is located at register offset 1B1/1C1 when the Divisor Latch Access (DLAB) bit in the UARTx Line Control register (see page 227) is set to 1. When the DLAB bit is 0 the UARTx_ IER register is accessed from this register offset. Refer to Table 94 on page 219 for more information on this register.

When the Divisor Latch Access (DLAB) bit is set, writes to UART offset 0x0 and 0x1 set the baud rate divisor. When the DLAB bit is set to 1, the UARTx_RX_TX and UARTx_IER registers are not accessible. The register space is shared between the UARTx_DLM register and the UARTx_Rx_Tx at offset 1B0 (Table 92 on page 216 and Table 93 on page 218) and the UART1_DLL register and the UARTx_IER register (Table 94 on page 219 and Table 95 on page 221) at offset 1B1.

Writing 1 to the DLAB bit enables the UARTx_DLM or UARTx_DLL (depending on reads and writes) and disables the other registers that share their register offset.

The output frequency of the Baud Generator is represented in the following equation:

- (Frequency/ (16 * Baud))-1

Two 8-bit latches store the divisor in a 16-bit binary format. These Divisor Latches must be loaded during initialization to ensure proper operation of the Baud Generator. Upon loading either of the Divisor Latches, a 16-bit Baud counter is immediately loaded.

Table 96: UARTx Interrupt Status / FIFO Control (Read Only)

Register Name: UARTx_ISTAT_FIFO	Register Offset: 1B2/1C2

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	B0	B1	B2	B3	Reserved		B6	B7

UART*x*_ISTAT_FIFO Description

Name	Type	Reset By	Reset State	Function
B0	R/W	PORESET_	0	Read: 0 if interrupt pending Write: FIFO enable
B1	R/W	PORESET_	0	Read: Interrupt ID bit 0 Write: Receiver FIFO reset
B2	R/W	PORESET_	0	Read: Interrupt ID bit 1 Write: Transmit FIFO reset
B3	R/W	PORESET_	0	Read: Interrupt ID bit 2
B6	R/W	PORESET_	0	Read: FIFOs enabled Write: Receiver trigger (LSB)
B7	R/W	PORESET_	0	Read: FIFOs enabled Write: Receiver trigger (MSB)

The UARTx_ISTAT_FIFO register is considered to be interrupt status register for reads and a FIFO control register for writes. Refer to Table 98 for more information on the FIFO control bit information.

The UARTx_ISTAT_FIFO register has different functionality depending on whether the register is read or if it is written. Table 17 on page 89 shows the register in its read only state and its write only state. In the “ Registers” on page 153 the register bits are explained first in the right only state (Table 96 on page 222) and then in their write only state (Table 98 on page 225).

In order to provide minimum software overhead during data character transfers, the UART prioritizes interrupts into three levels and records these in the Interrupt Status/ FIFO Control register. The levels of interrupt conditions in order of priority are as follows:

- Receiver Line Status
- Received Data Ready

- Transmitter Holding Register Empty

When the processor accesses the UART*x*_ISTAT_FIFO, the UART stops all interrupts and indicates the highest priority pending interrupt to the interrupt controller. While this interrupt controller access is occurring, the UART records new interrupts, but does not change its current indication until the access is complete.

Table 97 highlights the UART interrupt control functions.

Table 97: Interrupt Control Functions

Interrupt Identification Register			Interrupt Set and Reset Functions			
Bit 2	**Bit 1**	**Bit 0**	**Priority Level**	**Interrupt Type**	**Interrupt Source**	**Interrupt Reset Control**
0	0	1	-	None	None	-
1	1	0	Highest	Receiver line status	Overrun error or parity error or framing error or break error	Reading the Line Status register
1	0	0	Second	Received Data Available	Receiver data available or trigger level reached	Reading the Receiver Buffer register or the FIFO drops below the trigger level
0	1	0	Third	Transmitter Holding Register Empty	Transmitter Holding register empty	Reading the UART*x*_ISTAT_FIFO register (if it is the source of the interrupt) or writing into the Transmitter Holding register

Bit 0: This bit can be used in a prioritized interrupt environment to indicate whether an interrupt is pending. When bit 0 is a logic 0, an interrupt is pending and the UART*x*_ISTAT_FIFO contents can be used as a pointer to the appropriate interrupt service routine. When bit 0 is a logic 1, no interrupt is pending.

Bit 1: Bits 1 and 2 are used to identify the highest priority interrupt pending.

Bit 2: Bits 1 and 2 are used to identify the highest priority interrupt pending.

Bit 3: In the FIFO mode this bit is set along with bit 2 when a timeout interrupt is pending.

Bit 4: PowerPro Reserved

Bit 5: PowerPro Reserved

Bit 6: Bits 6 and 7 are set when FCR0 = 1

Bit 7: Bits 6 and 7 are set when FCR0 =1

Table 98: UARTx Interrupt Status / FIFO Control (Write Only)

Register Name: UARTx_ISTAT_FIFO	Register Offset: 1B2/1C2

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	B0	B1	B2	B3	Reserved		B6	B7

UART*x*_ISTAT_FIFO Description

Name	Type	Reset By	Reset State	Function
B0	R/W	PORESET_	0	Read: 0 if interrupt pending Write: FIFO enable
B1	R/W	PORESET_	0	Read: Interrupt ID bit 0 Write: Receiver FIFO reset
B2	R/W	PORESET_	0	Read: Interrupt ID bit 1 Write: Transmit FIFO reset
B3	R/W	PORESET_	0	Read: Interrupt ID bit 2
B6	R/W	PORESET_	0	Read: FIFOs enabled Write: Receiver trigger (LSB)
B7	R/W	PORESET_	0	Read: FIFOs enabled Write: Receiver trigger (MSB)

The UARTx_ISTAT_FIFO register is considered to be interrupt status register for reads and a FIFO control register for writes. Refer to Table 96 for more information on the interrupt status bit descriptions.

The UARTx_ISTAT_FIFO register has different functionality depending on whether the register is read or if it is written. Table 17 on page 89 shows the register in its read only state and its write only state. In the “ Registers” on page 153 the register bits are explained first in the right only state (Table 96 on page 222) and then in their write only state (Table 98 on page 225).

This register is used to enable the FIFOs, clear the FIFOs and set the RCVR FIFO trigger level.

Bit 0: Writing a 1 to FCR0 enables both the XMIT and RCVR FIFOs. Resetting FCR0 clears all bytes in both FIFOs.

Bit 1: Writing a 1 to FCR1 clears all bytes in the RCVR FIFO and resets its counter logic to 0. The shift register is not cleared. The 1 that is written to this bit position is self-clearing.

Bit 2: Writing a 1 to FCR2 clears all bytes in the XMIT FIFO and resets its counter logic to 0. The shift register is not cleared. The 1 that is written to this bit position is self-clearing.

Bit 3: PowerPro reserved (write)

Bit 4: PowerPro reserved

Bit 5: PowerPro reserved

Bit 6: FCR6 and FCR7 are used to set the trigger level for the RCVR FIFO interrupt.

Bit 7: FCR6 and FCR7 are used to set the trigger level for the RCVR FIFO interruptl

Table 99: UART*x* Line Control

Register Name: UART*x*_LCR	Register Offset: 1B3/1C3

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	WLEN		STB	PEN	EPS	SP	SB	DLAB

UART*x*_LCR Description

Name	Type	Reset By	Reset State	Function
WLEN[0:1]	R/W	PORESET_	0	Word Length Bits
STB	R/W	PORESET_	0	Number of Stop Bits
PEN	R/W	PORESET_	0	Parity Enable
EPS	R/W	PORESET_	0	Even Parity Select
SP	R/W	PORESET_	0	Stick Parity
SB	R/W	PORESET_	0	Set Break
DLAB	R/W	PORESET_	0	Divisor Latch Access Bit

The UARTx_LCR specifies the format of the asynchronous data communications exchange and sets the Divisor Latch Access bit. The contents of the UARTx_LCR can also read. The read capability simplifies system programming and eliminates the need for separate storage in system memory of the line characteristics.

UART registers are only byte-accessible.

WLEN[0:1]: These two bits specify the number of bits in each transmitted or received serial character. The encoding for the WLEN bits is shown in Table 100.

Table 100: WLEN Coding

WLEN[0]	WLEN[1]	Character Length
0	0	5 bits
0	1	6 bits
1	0	7 bits
1	1	8 bits

STB: This bit specifies the number of Stop bits transmitted and received in each serial character. If the STB bit is a logic 0, one Stop bit is generated in the transmitted data. If the STB bit is a logic 1 when a 5-bit word length is selected through bits 0 and 1, one and a half Stop bits are generated. If the STB bit is a logic 1 when either a 6-, 7-, or 8-bit word length is selected, two Stop bits are generated. The receiver checks the first Stop-bit only, regardless of the number of Stop bits selected.

PEN: This bit is the Parity Enable bit. When the PEN bit is a logic 1, a Parity bit is generated (transmit data) or checked (receive data) between the last data word bit and Stop bit of the serial data. The Parity bit is used to produce an even or odd number of 1s when the data word bits and the Parity bit are added.

EPS: This bit is the Even Parity Select bit. When the PEN bit is a logic 1 and the EPS bit is a logic 0, an odd number of logic 1s is transmitted or checked in the data word bits and Parity bit. When the PEN bit is a logic 1 and the EPS bit is a logic 1, an even number of logic 1s is transmitted or checked.

SP: This bit is the Stick Parity bit. When the PEN bit, the EPS bit and the SP bit are logic 1, the Parity bit is transmitted and checked as a logic 0. If the PEN bit and the SP bit are 1 and the EPS bit is a logic 0 then the Parity bit is transmitted and checked as a logic 1.

If the SP bit is a logic 0 Stick Parity is disabled.

SB: This bit is the Break Control bit. It causes a break condition to be transmitted to the receiving UART. When it is set to a logic 1, the serial output (SOUT) is forced to the Spacing (logic 0) state. The break is disabled by setting the SB bit to a logic 0.

DLAB: This bit is the Divisor Latch Access Bit (DLAB). It must be set high (logic 1) to access the Divisor Latches of the Baud Generator during a Read or Write operation. It must be set low (logic 0) to access the UARTx_RX_TX and UARTx_IER registers.

When the Divisor Latch Access (DLAB) bit is set, writes to UART offset 0x0 and 0x1 set the baud rate divisor. When the DLAB bit is set to 1, the UARTx_RX_TX and UARTx_IER registers are not accessible. The register space is shared between the UART0_DLM register and the UARTx_Rx_Tx at offset 1B0 (Table 92 on page 216 and Table 93 on page 218) and the UARTx_DLL register and the UARTx_IER register (Table 94 on page 219 and Table 95 on page 221) at offset 1B1.

Writing 1 to the DLAB bit enables the UARTx_DLM or UARTx_DLL (depending on reads and writes) and disables the other registers that share their register offset.

Table 101: UART*x* Modem Control

Register Name: UART*x*_MCR	Register Offset: 1B4/1C4

Bits	Function							
	0	1	2	3	4	5	6	7
0-7								

UART*x*_MCR Description

Name	Type	Reset By	Reset State	Function

The UARTx_MCR register is not supported by PowerPro.

Table 102: UART*x* Line Status

Register Name: UART*x*_LSR	Register Offset: 1B5/1C5

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	DR	OE	PE	FE	BI	THRE	TEMT	ERF

UART*x*_LSR Description

Name	Type	Reset By	Reset State	Function
DR	R/W	PORESET_	0	Data Ready
OE	R/W	PORESET_	0	Overrun Error
PE	R/W	PORESET_	0	Parity Error
FE	R/W	PORESET_	0	Framing Error
BI	R/W	PORESET_	0	Break Interrupt
THRE	R/W	PORESET_	0	Transmitter Holding Register
TEMT	R/W	PORESET_	0	Transmitter Empty
ERF	R/W	PORESET_	0	Error In Receiver FIFO

This register provides status information to the processor concerning the data transfer.

The UARTx_LSR register is intended for read operations only. Writing to this register is not recommended. Writing to this register is intended only for testing purposes. The bits in this register cannot be written to in FIFO mode.

DR: This bit is the receiver Data Ready (DR) indicator. The DR bit is set to a logic 1 whenever a complete incoming character has been received and transferred into the Receiver Buffer Register or the FIFO. The DR bit is reset to a logic 0 by reading all of the data in the UARTx_RX_TX register or the FIFO.

OE: This bit is the Overrun Error (OE) indicator. The OE bit indicates that data in the Receiver Buffer Register was not read by the processor before the next character was transferred into the Receiver Buffer Register, thereby destroying the previous character. The OE indicator is set to a logic 1 upon detection of an overrun condition and reset whenever the processor reads the contents of the Line Status Register. If the FIFO mode data continues to fill the FIFO beyond the trigger level, an overrun error occurs only after the FIFO is full and the next character has been completely received in the shift register. OE is indicated to the processor as soon as it happens. The character in the shift register is overwritten, but it is not transferred to the FIFO.

PE: This bit is the Parity Error (PE) indicator. The PE bit indicates that the received data character does not have the correct even or odd parity, as selected by the even-parity select bit. The PE bit is set to a logic 1 upon detection of a parity error and is reset to a logic 0 whenever the processor reads the contents of the Line Status Register. In the FIFO mode this error is associated with the particular character in the FIFO it applies to. This error is revealed to the processor when its associated character is at the top of the FIFO.

FE: This bit is the Framing Error (FE) indicator. The FE bit indicates that the received character did not have a valid Stop bit. The FE bit is set to a logic 1 whenever the Stop bit following the last data bit or parity bit is detected as a logic 0 bit (Spacing level). The FE indicator is reset whenever the processor reads the contents of the Line Status Register. In the FIFO mode this error is associated with the particular character in the FIFO it applies to. This error is revealed to the processor when its associated character is at the top of the FIFO. The UART attempts to resynchronize after a framing error. To do this the UART assumes that the framing error was due to the next start bit, so it samples this start bit twice and then takes in the data.

BI: This bit is the Break Interrupt (BI) indicator. The BI bit is set to a logic 1 whenever the received data input is held in the Spacing (logic 0) state for longer than a full word transmission time (that is, the total time of Start bit + data bits + Parity + Stop bits). The BI indicator is reset whenever the interrupt controller reads the contents of the Line Status Register. In the FIFO mode this error is associated with the particular character in the FIFO it applies to. This error is revealed to the processor when its associated character is at the top of the FIFO. When break occurs only one zero character is loaded into the FIFO. The next character transfer is enabled after SIN goes to the marking state and receives the next valid start bit.

Bits 1 through 4 are error conditions that produce a Receiver Line interrupt whenever any of the corresponding conditions are detected and the interrupt is enabled.

THRE: This bit is the Transmitter Holding Register Empty (THRE) indicator. The THRE bit indicates that the UART is ready to accept a new character for transmission. In addition, this bit causes the UART to issue an interrupt to the interrupt controller when the Transmit Holding register Empty Interrupt enable is set high. The THRE bit is set to a logic 1 when a character is transferred from the Transmitter Holding Register into the Transmitter Shift register. The bit is reset to logic 0 concurrently with the loading of the Transmitter Holding register by the interrupt controller. In the FIFO mode The THRE bit is set when the XMIT FIFO is empty; it is cleared when at least 1 byte is written to the XMIT FIFO.

TEMT: This bit is the Transmitter Empty (TEMT) indicator. The TEMT bit is set to a logic 1 whenever the UARTx_RX_TX register is empty. It is reset to a logic 0 whenever the UARTx_RX_TX register contains a data character. In the FIFO mode this bit is set to 1 whenever the transmitter FIFO is empty.

ERF: The ERF bit is set when there is at least one parity error, framing error or break indication in the FIFO. The ERF bit is cleared when the processor reads the LSR register, if there are no subsequent errors in the FIFO.

UART registers are only byte-accessible.

Table 103: UART*x* Modem Status

Register Name: UART*x*_MSR	Register Offset: 1B6/1C6

Bits	Function							
	0	1	2	3	4	5	6	7
0-7								

UART*x*_MSR Description

Name	Type	Reset By	Reset State	Function

The UARTx_MSR register is not supported by PowerPro.

Table 104: UART*x* Scratchpad Register

Register Name: UART*x*_SCR	Register Offset: 1B7/1C7

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	SR							

UART*x*_SCR Description

Name	Type	Reset By	Reset State	Function
SR[0:7]	R/W	PORESET_	0	Scratchpad Register

This register is an 8-byte read/write scratch register (memory). It performs no function, and doe not control the UART. It is intended to be used to hold temporary data.

UART registers are only byte-accessible.

Table 105: General Purpose I/O

Register Name: GPIO_*x*	Register Offset: 1E0/1E4/1E8/1EC/ 1F0/1F4/1F8

Bits	Function							
	0	1	2	3	4	5	6	7
0-7	ENABLE							
8-15	MASK							
16-23	DIR							
24-31	DATA							

GPIO_ *x* Description

Name	Type	Reset By	Reset State	Function
ENABLE[0:7]	R/W	PORESET_	0	GPIO Enable 0 = disable corresponding GPIO port 1 = enable corresponding GPIO port
MASK[0:7]	R/W	PORESET_	0	GPIO Mask 1 = enable corresponding GPIO write 0 = mask corresponding GPIO write
DIR[0:7]	R/W	PORESET_	(see Table 106)	GPIO Direction 1 = corresponding GPIO port is input (read) 0 = corresponding GPIO port is output (write)
DATA[0:7]	R/W	PORESET_	0	GPIO data DIR = 0 Holds read data from GPIO port. Writes have no effect. DIR = 1 Holds write data output to GPIO port. Reads return the data that is being output on the corresponding GPIO port.

GPIO functionality is multiplexed with many other functions on PowerPro. Critical PowerPro pins are multiplexed with GPIO capability. Care must be taken when pins are programmed as GPIO pins. There is a potential to create contention in PowerPro that can cause the device to fail or to require excess power.

Table 106 shows the reset states for the DIR field in the GPIO_x registers.

Table 106: Reset state of the DIR field in all GPIO_x registers

Register	DIR Field Reset State
GPIO_A	0xF0
GPIO_B	0x0F
GPIO_C	0xFF
GPIO_D	0xFF
GPIO_E	0xFF
GPIO_F	0xFF
GPIO_G	0xC0

PowerPro has general purpose I/O capability. Although all pins on the device have a primary purpose, in many instances these pins are required for their primary purpose. For example, a system may not need two UARTS or two I2C ports. These pins, when not being used for their primary purpose, are assigned to the General Purpose I/O pool. All pins capable of GPIO are mapped in a GPIO register.

Refer to Chapter 15: “Electrical Characteristics and Pin Information” on page 139for information on GPIO mapping.

ENABLE: This bit enables the use of the GPIO port.

DIR: This bit controls if the signal is an output or an input.

DATA: This field controls the value the pin has when it is an output. Reading from the data port returns the value the pin currently assumes. If the pin is an input, the input value is returned. If the pin is configured as an output, the value output on the pin is returned.

MASK: The Write Mask bit sets the pins bit-wise without affecting other pins in the same register. If the write mask value is 0, the ENABLE, DIR, and DATA values are ignored. If the write mask value is 1, then the ENABLE, DIR, and DATA values are written. The write mask has no effect on reads.

Reading from a GPIO port, even if that port is not enabled, returns the value on the pin at the time the read command was executed.

Software can control any low to medium speed device by using the GPIO and the general purpose timers. For example, the GPIO port and general purpose timers can control I^2C ports and RAMs. They can control any other devices are low to medium speed with non-time critical protocols.

Chapter 17: Timing

This chapter outlines the timing information and requirements of PowerPro. The topics addressed in this chapter include:

17.1 Overview

This chapter describes the timing information for the PowerPro device and contains complete AC timing information for PowerPro. Input setup and hold requirements are listed, as well as output setup and hold timings. These timings are divided into 5 pF increments as a function of capacitive loading on the signal pin.

A desired configuration must have all signals analyzed on the processor (60x) bus and the SDRAM bus. Different configurations have different critical signals. For example the number and type of processors and memory in a system

17.2 SDRAM and Processor (60x) Bus Timing Information

17.2.1 PowerPro Input Setup / Hold Times

Table 107: PowerPro Input Setup / Hold Times

Signal	PowerPro Input Setup Time	PowerPro Input Hold Time	Units
PB_TS_ PB_ARTRY_	3.8	0.1	ns
PB_A PB_AP PB_TT PB_TBST_ PB_TSIZ PB_AACK_	3.6	0.1	ns
PB_D PB_DP PB_TA PB_TEA_	3.0	0.0	ns
PB_BR	2.4	0.0	ns
SD_ECC	2.5	0.0	ns
SD_D	3.9	0.0	ns

17.2.2 PowerPro Output Setup / Hold Times

Table 108: PowerPro Output Setup / Hold Times

Load (pF)[a]	SD_CS[b c]	SD Cmd	SD Address	SD Data	SD ECC / DQM	PB_AACK	PB_ARTRY	PB dack	PB_Data	Units
5 pF	6.4	4.7	5.0	4.6	4.8	4.8	5.1	6.0	5.7	ns
10 pF	6.9	5.1	5.4	4.8	5.0	5.0	5.5	6.2	6.2	ns
15 pF	7.4	5.7	6.0	5.0	5.2	5.2	6.1	6.4	6.7	ns
20 pF	7.9	6.2	6.5	5.2	5.4	5.4	6.6	6.6	7.2	ns
25 pF	8.5	6.6	6.9	5.4	5.6	5.6	7.0	6.8	7.6	ns
30 pF	9.2	6.9	7.2	5.7	5.9	5.9	7.3	7.1	7.9	ns
35 pF	9.8	7.3	7.6	5.9	6.1	6.1	7.6	7.3	8.3	ns
40 pF	10.3	7.7	8.0	6.2	6.4	6.4	8.0	7.6	8.7	ns
45 pF	10.8	8.1	8.4	6.5	6.7	6.7	8.5	7.9	9.1	ns
50 pF	11.4	8.5	8.8	6.8	7.0	7.0	8.9	8.2	9.6	ns
55 pF	12.0	9.0	9.3	7.0	7.2	7.2	9.3	8.4	10.0	ns
60 pF	12.5	9.4	9.7	7.2	7.4	7.4	9.7	8.6	10.4	ns
65 pF	13.1	9.8	10.1	7.5	7.7	7.7	10.1	8.9	10.8	ns
70 pF	13.7	10.2	10.5	7.6	7.8	7.8	10.5	9.0	11.2	ns
75 pF	14.3	10.6	10.9	7.9	8.1	8.1	11.0	9.3	11.6	ns
80 pF	14.8	11.0	11.3	8.1	8.3	8.3	11.4	9.5	12.0	ns
85 pF	15.4	11.4	11.7	8.3	8.5	8.5	11.8	9.7	12.4	ns
90 pF	16.0	11.8	12.1	8.5	8.7	8.7	12.2	9.9	12.8	ns
95 pF	16.5	12.2	12.5	8.8	9.0	9.0	12.6	10.2	13.3	ns

a. All timing information is provided in a lumped capacitive load with the size indicated on the table. The table indicates measured timing worst case industrial at 2.0 V with worst case internal PowerPro induced clock skew and jitter applied.

b. The cell headings refer to the following signals: SD Chip Select = SD_CS, SD Command = CS_RAS, SD_CAS, SD_WE, SD Address = SD_A[12:0], SD_BA[1:0], SD Data = SD_D[0:63], SD ECC = SD_ECC / DQM [0:7], PB AACK = PB_AACK_, PB ARTRY = PB_ARTRY_, PB dack = PB_TA_, PB_DVAL_, PB_TEA_, PB_BG_, PB_DBG_, PB Data = PB_D, PB_DP

c. The SD Chip Select numbers provided need to be used for SD_CS where a third party address retry occurs on the processor (60x) bus. In a system where third party address retry never occurs, the timing for SD_CS can be reduced (in all loadings) by 1.0 ns.

17.2.3 Output Hold Time Calculation

Output hold times for the pin groups are calculated as follows:

- SDRAM Chip Select Hold = (load, pF) * 0.031 + 1.3
- SDRAM Command, Address hold = (load, pF) * 0.021 + 1.5
- SDRAM Data, ECC hold = (load, pF) * 0.012 + 1.4
- PB ARTRY, PB AACK, Dack = (load, pF) * 0.012 + 1.6
- PB Data = (load, pF) * 0.021 + 1.3

Appendix A: Typical Applications

This chapter outlines typical PowerPro applications. The topic addressed in this chapter include:

A.1 Overview

This chapter some processor (60x) bus and SDRAM system configurations supported by PowerPro. The configurations are discussed in terms of the timing implications for PowerPro operation.

The PowerPro SDRAM and Processor Bus interfaces run from a single synchronous clock. The limiting factor on system timing can be caused from either or both of these interfaces. The factors that effect timing include bus loading, clock distribution, board layout, and external device speed. These factors are addressed throughout this chapter.

Common system examples are described in the following sections. The system examples are divided into different configurations on the processor (60x) bus and different configurations on the SDRAM bus.

The processor (60x) bus connection examples contain the Motorola MPC8260, PowerPC 750, PowerSpan, and PowerPro devices. Micron SDRAM modules are used throughout the SDRAM configuration examples. These examples have been chosen to illustrate timing requirements on a standard group of components. Other processor (60x) bus components, memory modules and memory configurations can be used with PowerPro. However, when different components are used in a system the timing between the external device and PowerPro must be verified.

In all of the examples below, 1.0 ns is used as an aggregate system clock skew and jitter. Internal PowerPro clock jitter is incorporated in the timing numbers in Chapter 17: “SDRAM and Processor (60x) Bus Timing Information” on page 238. Tight system constraints can decrease the 1.0 ns number, which results in a correspondingly faster operational frequencies. The opposite is also true, where systems have a large clock skew or system jitter component a number greater than 1.0 ns must be used in the calculations.

A.2 Application Specific Timing

A.2.1 Processor (60x) Bus Interface

PowerPro is designed to operate synchronously at 100 MHz with the PowerPC 750. PowerPro also operates up to 83 MHz with the currently available Tundra PowerSpan device. PowerPro’s operational frequency is dependent on the system configuration. The frequency dependency includes board layout, external device speed, bus loading and clock distribution. PowerPro I/O timing is dependent on system configuration factors, however all internal paths operate at 100 MHz.

When attached to an processor (60x) bus master, PowerPro has PB_D, PB_DP, PB_ARTRY_, PB_TA_, and PB_TEA_ as outputs from PowerPro to the processor (60x) bus master. PowerPro has PB_TS_, PB_TT_, PB_TSIZ_, PB_TBST_, PB_A, PB_AP, PB_AACK_, PB_ARTRY_, PB_TA_, PB_TEA_, PB_D, and PB_DP as inputs from the processor (60x) bus master.

A.2.1.1 PowerPC 750 Timing

The following tables describe the important input and output timing signals for the PowerPC 750 in PowerPro applications. Refer to the *Motorola MPC750A Hardware Specification* for more timing information.

PowerPC 750 Input Timing

Table 109 describes the significant PowerPC 750 input timing.

Table 109: PowerPC 750 Input Timing

Signal	PowerPC 750 Input Setup Time	PowerPC 750 Input Hold Time	Units
D DP	2.5	0.0	ns
ARTRY_ TA_ TEA_	2.5	0.0	ns

PowerPC 750 Output Timing

Table 110 describes the significant PowerPC 750 output timing in a lumped 50pF load.

Table 110: PowerPC 750 Output Timing (50pF load)

Signal	PowerPC 750 Output Setup Time	PowerPC 750 Output Hold Time	Units
TS_ TT_ TSIZ_ TBST A AP AACK_ ARTRY_ TA_ TEA_ D DP	5.0	1.0	ns

A.2.1.2 MPC8260 Timing

The following tables describe the important input and output timing signals for the MPC8260 in PowerPro applications. Refer to the *Motorola MPC8260 Hardware Specification* for more timing information.

MPC8260 Input Timing

Table 111 describes the significant MPC8260 input timing.

Table 111: MPC8260 Input Timing

Signal	MPC8260 Input Setup Time	MPC8260 Input Hold Time	Units
AACK_ ARTRY_ TA_ TEA_ DBG_ BG_ BR_	6.0	1.0	ns
D	5.0	1.0	ns
DP	8.0	1.0	ns
All other pins	5.0	1.0	ns

MPC8260 Output Timing

Table 112 describes the significant MPC8260 output timing.

Table 112: MPC8260 Output timing

Signal	MPC 8260 Output Setup Time	MPC 8260 Output Hold Time	Units
DVAL_ TA_ TEA_	10.0	1.0	ns
A TS_ TT TSIZ TBST_ AP AACK_ ARTRY_ CI_ GBL_ WT_	8.0	1.0	ns
D DP	8.0	1.0	ns
All other pins	7.0	1.0	ns

A.2.1.3 PowerSpan Timing

The following tables describe the important input and output timing signals for PowerSpan in PowerPro applications. Refer to the *PowerSpan PowerPC-to-PCI Bus Switch Manual* for more timing information.

PowerSpan Input Timing

Table 113 describes the significant PowerSpan input timing.

Table 113: PowerSpan Input Timing

Signal	PowerSpan Input Setup Time	PowerSpan Input Hold Time	Units
PB_TS_ PB_TT PB_TSIZ_ PB_TBST_ PB_A PB_AP PB_AACK_ PB_ARTRY_ PB_TA_ PB_TEA_ PB_D PB_DP	3.2	0.5	ns
PB_BR_	3.4	0.5	ns

PowerSpan Output Timing

Table 114 describes the significant PowerSpan output timing.

Table 114: PowerSpan Output Timing

Signal	PowerSpan Output Setup Time	PowerSpan Output Hold Time	Units
PB_A PB_AP PB_TSIZ_ PB_TT_ PB_TBST_ PB_D PB_DP	7.3	1.7	ns
PB_TS_ PB_TA_ PB_DVAL_ PB_TEA_ PB_AACK_	6.6	2.0	ns
PB_ARTRY_	6.1	2.0	ns
PB_BG_ PB_DBG_	6.8	1.4	ns

A.2.1.4 PowerPro Timing

The PowerPro input and output timing are summarized in Chapter 17: “Timing” on page 237. The PowerPro timing is provided for a range of capacitive loads, while the PowerPC 750 is specified into 50pF lumped load and PowerSpan is specified into 35pF lumped load.

A.2.1.5 PowerPro Configuration Examples

The following sections describe specific application examples of PowerPro systems.

PowerPro System with PowerSpan and PowerPC 750.

In a system with PowerPro, PowerSpan, and the PowerPC 750 the loading on all shared pins is approximately 20 pF. Table 115 illustrates the output timing of PowerPro

Table 115: Output Timing of PowerPro

Signal	Input Setup	Input Hold	Output Setup	Output Hold	Units
Address Attributes	3.8	0.1	N/A	N/A	ns
PB_AACK_	3.6	0.1	5.4	1.8	ns
PB_ARTRY_	3.8	0.1	6.6	1.8	ns
PB_TA_, PB_DVAL_	3.0	0.0	6.6	1.8	ns
PB_D	3.0	0.0	7.2	1.7	ns

The limiting timing factor in this system example is the Address attributes from PowerSpan to PowerPro. The numbers when added together (7.3 + 3.8 + 1.0 = 12.1 ns) produce the frequency of approximately 83 MHz.

PowerPro System with PowerSpan and MPC8260.

In a system with PowerPro, PowerSpan, and the MPC8260 the loading on all shared pins is approximately 20 pF. Table 116 illustrates the output timing of PowerPro.

Table 116: Output Timing of PowerPro

Signal	Input Setup	Input Hold	Output Setup	Output Hold	Units
Address Attributes	3.8	0.1	N/A	N/A	ns
PB_AACK_	3.6	0.1	5.4	1.8	ns
PB_ARTRY_	3.8	0.1	6.6	1.8	ns
PB_TA_	3.0	0.0	6.6	1.8	ns
PB_DVAL_	3.0	0.0	6.6	1.8	ns
PB_D	3.0	0.0	7.2	1.7	ns

The limiting timing factor in this system example is the Data Acknowledge attributes from the MPC8260 to the PowerPro. The numbers, when added together (10.0 + 3.0 + 1.0 = 14.0 ns), produce the frequency of approximately 71 MHz.

PowerPro System with a Single PowerPC 750

In a system with PowerPro and a single PowerPC 750 the loading on all shared pins is approximately 12 pF. Table 117 illustrates the output timing of PowerPro.

Table 117: Output Timing of PowerPro

Signal	Input Setup	Input Hold	Output Setup	Output Hold	Units
Address Attributes	3.8	0.1	N/A	N/A	ns
PB_AACK_	3.6	0.1	5.1	1.8	ns
PB_ARTRY_	3.8	0.1	5.7	1.8	ns
PB_TA_	3.0	0.0	6.3	1.8	ns
PB_DVAL_	3.0	0.0	6.3	1.8	ns
PB_D	3.0	0.0	6.4	1.7	ns

The limiting timing factor in this system example is the output propagation delay on the PB_D bus from PowerPro. The numbers when, added together (6.4 + 2.5 + 1.0 = 9.9 ns), produce the frequency of approximately 101 MHz.

A.2.2 SDRAM Interface

PowerPro is designed to operate at 100MHz. PowerPro runs synchronously with the Processor Bus Interface using two registered SDRAM DIMMs. Four registered DIMMs enable operation of over 83 MHz. PowerPro's operational frequency is dependent on the system configuration. The frequency dependency includes board layout, external device speed, bus loading and clock distribution. PowerPro I/O timing is dependent on system configuration factors, however all internal paths operate at 100 MHz.

It is possible to use unbuffered DIMMs, but the operational frequency is reduced.

The examples below show some sample configurations and the SDRAM timings attainable with the PowerPro as a memory controller. Note that the devices on the 60x bus may further limit the frequency possible in the system, as the SDRAM memory bus is synchronous to the 60x bus. System design techniques such as advancing the clock of the PowerPro relative to the SDRAMs are possible to further increase the frequency attainable from some configurations, since the PowerPro tends to have greater slack on hold times and input setup than it does on output delay.

Standard SDRAM memory timing is as follows (times in ns):

Table 118: SDRAM Timing

Frequency	Access Time	Setup Time	Hold Time	Units
PC-66	9	3	1	ns
PC-100	6	2	1	ns
PC-133	5.4	1.5	0.8	ns

A.2.2.1 SDRAM Configuration Examples

The following sections describe specific application examples of PowerPro systems.The track routing load is estimated to be 4pF in the following examples.

PowerPro System with Two Modules of SDRAM Memory

A system configured with PowerPro and two modules of SDRAM memory provides 1 Gbyte of memory (512 Mbyte per module). The memory used in this example is registered PC-100 DIMMs by Micron Technology, Inc. (MT36LSDF6472-1xx). Please refer to the Micron website at www.micron.com for more information.

The following criteria is used to equate signal loading in this example:

- Chip Select load
  - 8 pF per DIMM, two connections for each DIMM per CS line = 16 pF
  - 16 pF + the track routing load
- Address, Command load
  - (8 pF per DIMM * 2) + the track routing load
- Data, ECC load
  - (16 pF per DIMM * 2) + the track routing load

Table 119 outlines PowerPro SDRAM timing. Refer to Chapter 17: “Timing” on page 237 for more information.

Table 119: SDRAM Timing

Signal	Input Setup	Input Hold	Output Setup	Output Hold	Units
SD_CS	N/A	N/A	7.9	1.9	ns
SD_Command	N/A	N/A	6.2	1.9	ns
SD_Address	N/A	N/A	6.5	1.9	ns
SD_Data	3.9	0.0	5.9	1.9	ns
SD_ECC	2.5	0.0	6.1	1.9	ns

The system clock skew and variability is approximately 1.0 ns. Because this system example uses PC-100 memory the critical path is data output from SDRAM memory to PowerPro data input. The numbers when added together (6.0 ns + 3.9 ns + 1.0 ns = 10.9 ns) produce the frequency of approximately 91 MHz. Using PC-133 memory as an example, the critical datapath timing limits system operation to (5.4 ns + 3.9 ns + 1.0 ns = 10.3 ns) approximately 97 MHz.

PC-100 is specified with a 50 pF lumped load. The example in this section uses a 32 pF load. This reduces the SDRAM data valid time from 6.0 ns to 5.5 ns for PC-100, and from 5.4 ns to 5.1 ns for PC133. In this situation a critical path is the SD_CS signal (7.5 + 2.0 + 1.0 = 10.5 ns) at approximately 95 MHz. For PC-133 memory (7.5 + 1.5 + 1.0 = 10.0 ns) the 35 pF loading achieves operation of 100 MHz.

PowerPro System with Four Modules of SDRAM Memory

A system configured with PowerPro and four modules of SDRAM memory provides 2 Gbyte of memory (512 MB per module) using registered PC-100 DIMMs (Micron DIMMs MT36LSDF6472-1xx). Using similar 1 Gbyte DIMMs, four Gbytes of memory can be attached in this configuration.

The following criteria is used to equate signal loading in this example:

- Chip Select load
  - (8 pF per DIMM, two connections for each DIMM per CS line) + the track routing load
- Address, Command load
  - (8 pF per DIMM * 4) + the track routing load
- Data, ECC load
  - (16 pF per DIMM * 4) + the track routing load

Table 120 outlines PowerPro SDRAM timing. Refer to Chapter 17: “Timing” on page 237 for more information

Table 120: SDRAM Timing

Signal	Input Setup	Input Hold	Output Setup	Output Hold	Units
SD_CS	N/A	N/A	7.9 ns	1.9 ns	ns
SD_Command	N/A	N/A	7.4 ns	2.2 ns	ns
SD_Address	N/A	N/A	7.7 ns	2.2 ns	ns
SD_Data	3.9	0	7.6 ns	2.2 ns	ns
SD_ECC	2.5	0	7.8 ns	2.2 ns	ns

Using PC-100 SDRAM, the limiting factor in this system is the data connection from memory to the PowerPro (6.0 + 3.9 + 1.0 = 10.9 ns) approximately 92 MHz. Using PC-133 SDRAM, the limiting factor remains the memory to PowerPro datapath, and the frequency attainable is (5.4 + 3.9 + 1.0 = 10.3 ns) approximately 97 MHz.

PowerPro with One module of SDRAM Memory

The MT9LSDT872A-10E SDRAM is used in this system example. The MT9LSDT872A-10E is a dual-sided, PC-100, unbuffered DIMM. The device provides 128 Mbyte memory.

The following criteria is used to equate signal loading in this example:

- Chip Select load
  - (30 pF per DIMM, two connections per DIMM per CS line) + the track routing load
- Address, Command
  - (70 pF per DIMM) + the track routing load
- Data, ECC load
  - (15 pF per DIMM) + the track routing load

Table 121 outlines PowerPro SDRAM timing. Refer to Chapter 17: “Timing” on page 237 for more information

Table 121: SDRAM Timing

Signal	Input Setup	Input Hold	Output Setup	Output Hold	Units
SD_CS_	n/a	n/a	13.0	3.2	ns
SD_Command	n/a	n/a	10.6	3.0	ns
SD_A	n/a	n/a	10.9	3.0	ns
SD_D	3.9	0.0	5.2	1.6	ns
SD_ECC_	3.9	0.0	5.4	1.6	ns

This system uses an external, high-powered buffer which reduces the SD_CS timing by over 2.5 ns to under 10.5 ns. The limiting factor is the timing on the command and address lines. Using PC-100 memory (10.9 + 2.0 + 1.0 = 13.9 ns) the operational frequency is approximately 72 MHz. Using PC-133 memory (10.9 + 1.5 + 1.0 = 13.0 ns) the operational frequency is approximately 75 MHz.

For memory configurations that are loaded greater than the examples in this chapter, buffering or load switching using FET switches is recommended.

A.3 PowerPro Access to Little-endian Device

PowerPro's SDRAM Interface uses the SD_DATA signals to transfer information throughout the system. In some instances, the DATA signals must be configured to meet the needs of a specific system design.

The following section outlines the requirements of signal connection for the big-endian device PowerPro with a little-endian device.

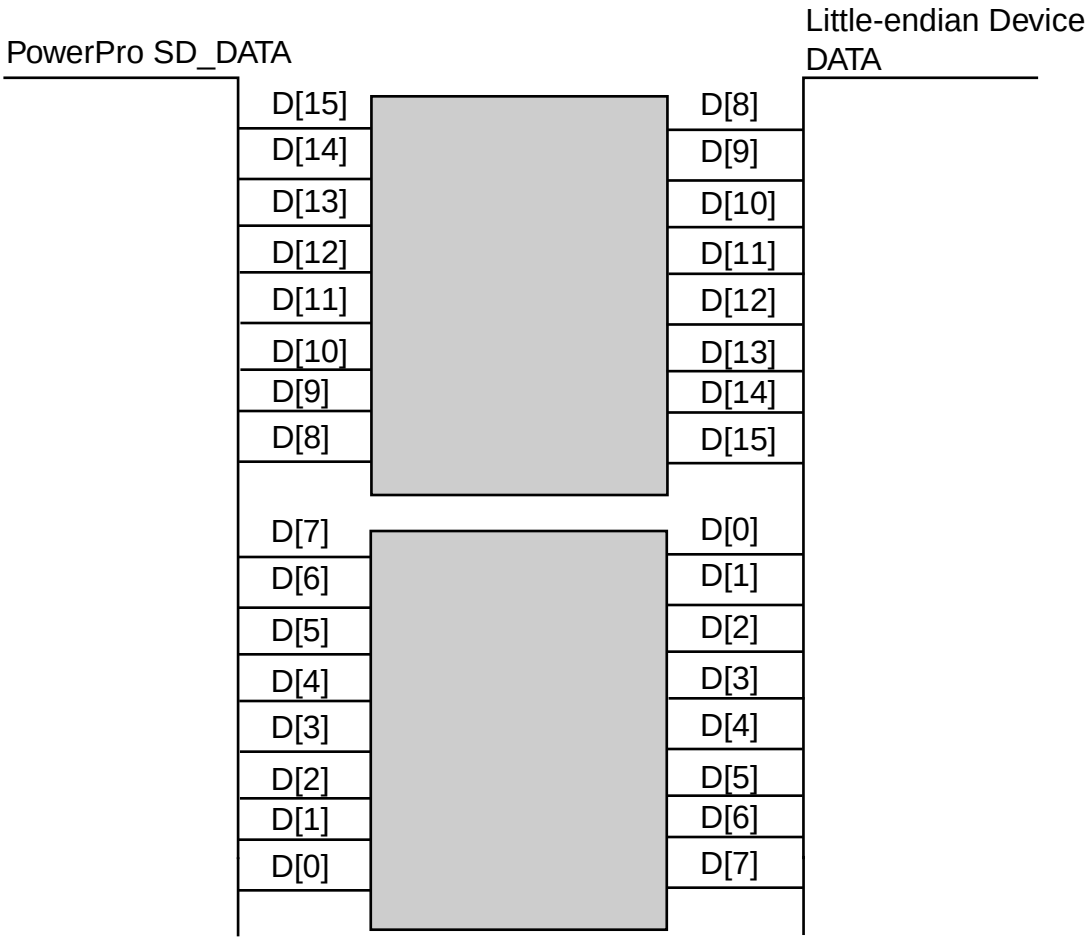
17.2.4 SDRAM Interface Connection to a Little-endian Device

17.2.4.1 Data Bus Connection

PowerPro is a big endian device which, in some applications, must be connected to little-endian devices. PowerPro does not perform any endian conversion. When PowerPro is connected to a little-endian device the two devices must compensate for the different endian protocols by arranging their connection to reflect data path requirements.

Figure 23 illustrates the connection that is required between PowerPro and a little-endian device data signals in order for data to flow with the proper ordering.

Figure 23: Data Bus Connection Between PowerPro and Little-endian Device



With this configuration a write of a byte of 0xAB to address location 0, the little-endian device D[7:0] equals 0xAB. If this connection is not used in a system, the D[7:0] signals will read 0xBA. They will be mapped with the improper endian ordering.

17.2.4.2 Additional Signal Connection

In order for the two devices to effectively transfer data in this application, there are additional signals which must be connected in the systems. Table 122 outlines the signals that are connected in this application.

Table 122: Connection Between PowerPro and Little-endian Device

PowerPro SDRAM Interface	Little-endian Device
SD_ECC[0]_	CE[1] Used to access Data [7:0]
SD_ECC[1]_	CE[2] Used to access Data [15:8]

In this configuration, the SD_ECC[1:0]_ signals are enabled as byte enables and are used to connect to little-endian device signals. The ECC protection functionality of PowerPro must be disabled in order for the signals to be used as byte enables. In order to globally disable ECC protection and enable these signals as byte enables, the Data Quality Mask Enable (DQM_EN) bit must be set to 1 in the SDRAM Timing Parameters register (see page 172).

Appendix B: Operating and Storage Conditions

This chapter outlines the power consumption, operating, and storage conditions of PowerPro. The topics addressed in this chapter include:

B.1 Overview

This chapter describes the power consumption, operating conditions and thermal characteristics of PowerPro.

The following list is a summary of PowerPro hardware parameters:

- Package: Surface mount 324 plastic ball grid array
- Core power supply (VDD_CORE): 2.5 V +/- 10% DC
- I/O power supply (VDD_IO): 3.3 V +/- 10% DC

B.2 Power Consumption

The PowerPro is an I/O limited device and because of this the I/O buffers dominate the power consumption of the device. The following assumptions are made when estimating the power consumption of PowerPro:

- Clock rate of 100MHz
- Data rate is 50MHz (single edge clocking)

B.2.1 I/O Power Consumption

Since the SDRAM Interface has the largest number of I/Os switching at any one time, the FLASH/ROM and processor (60x) bus interfaces are ignored. When PowerPro is driving either the SDRAM or the processor (60x) interface, it does not drive the other interface simultaneously. Only one interface is driving at any one time.

The SDRAM Interface consists of 14 address bits, four control bits, and up to 72 data/ECC bits. A total of 90 I/Os switch in PowerPro. Because the loading of the SDRAM Interface varies from application to application, a standard load of 50pF is assumed.

A standard loading of 50 pF is above average for most registered DIMM applications

The equation for the power consumption assumes 90 I/Os switching at 50MHz into 50pF. These assumptions output a value of 1.5 W as the total power consumption for PowerPro.

Table 123: Power Consumption Distribution

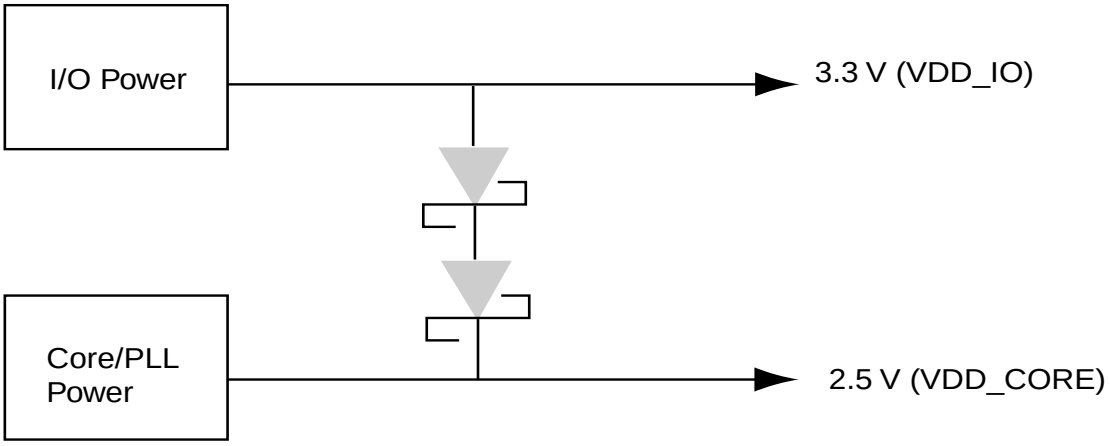
Parameter	Units
I/O Power Consumption	1.25 W
Core Power Consumption	0.25 W

B.3 Power-up Sequencing

Tundra recommends the use of a bootstrap diode between the power rails. The bootstrap diodes that are used in the system must be configured so that a nominal Core Supply Voltage (VDD_CORE) is sourced from the I/O Supply Voltage (VDD_IO)) until the power supply is active. In Figure 24, two Schottky barrier diodes are connected in series. Each of the diodes has a forward voltage (V_F) of 0.6 V at high currents which provides a 1.2 V current drop. This drop maintains 2.1 V on the 2.5 V power line. Once the Core/PLL power supply stabilizes at 2.5 V, the bootstrap diode(s) are reverse biased with small leakage current.

The forward voltage must be effective at the current levels required by PowerPro (< 1 amp). Do not use diodes with only a nominal V_F.

Figure 24: Bootstrap Diodes for Power-up Sequencing



B.4 Decoupling Capacitors

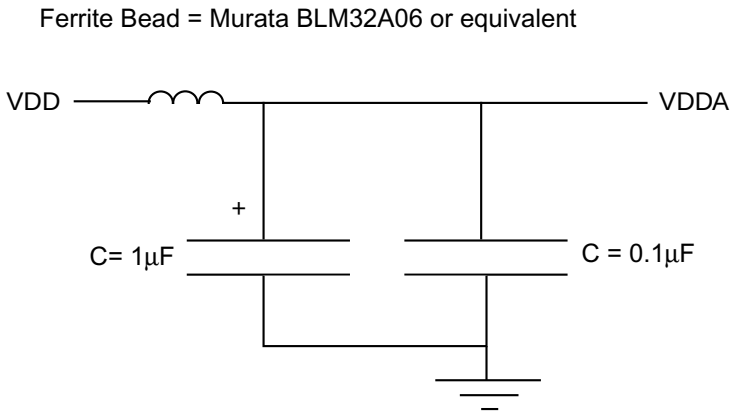
PowerPro requires the core voltage (VDD_CORE) and I/O voltage (VDD_IO) be decoupled to reduce switching noise. One bulk capacitor of 10uF is recommended for the core and I/O voltage supplies. Every third pair of power and ground pins must be decoupled with a 0.1uF surface mount capacitor to reduce high frequency switching noise.

The track lengths from the power and ground pins to the capacitors must be kept as short as possible. Based on this requirement, four 0.1uF capacitors are required for the I/O supply and two 0.1uF capacitors for the core supply. In order to keep the track lengths to the capacitors as short as possible, integrated capacitor components can be used in the application. It is possible to obtain components which have four 0.1uF capacitors in a 0612 size package. Other quantities and values of capacitors may be used at the designer's discretion.

B.5 PLL Filtering

VDDA is the voltage supply pin to the analog circuits in the PLL. Noise on VDDA can cause phase jitter at the output of the PLL. To provide isolation from the noisy internal digital circuitry, a filter circuit can be placed on VDDA (see Figure 25).

Figure 25: PLL Power Filter



All wire lengths must be kept short in order to minimize coupling from other signals.

B.6 Operating Conditions

Table 124 lists the recommended operating conditions for PowerPro.

Table 124: Recommended Operating Conditions

Characteristic		Symbol	Recommended Value	Unit
Core supply voltage		VDD_CORE	2.5V +/- 5%	V
PLL supply voltage		VDD_APLL	2.5V +/- 5%	V
I/O supply voltage		VDD_IO	3.3V +/- 5%	V
Input Voltage	Processor Bus	V_{in}	GND to VDD_IO	V
	Memory Bus	V_{in}	GND to VDD_IO	V
	JTAG Signals	V_{in}	GND to VDD_IO	V
Commercial Operation		t_{COM}	0 to +70	°C
Industrial Operation		t_{IND}	-40 to +85	°C

B.7 Thermal Characteristics

The maximum ambient temperature of PowerPro is calculated as follows:

$Ta \leq Tj - \theta_{ja} * P$

Where:

T_a = Ambient temperature (°C)

T_j = Maximum PowerPro Junction temperature (°C) = 125°C

θ_{ja} = Ambient to Junction Thermal Impedance (°C / W) see Table 125.

P = PowerPro power consumption (W).

The ambient to junction thermal impedance (θ_{ja}) is dependent on the air flow in meters per second over PowerPro.

Table 125: 3.3 Volt Package Thermal Resistance

Air Flow (m/s)	Package	Unit
	23 mm	
0	26.8	θ_{ja} °C/W
1	24.8	θ_{ja} °C/W
2	22.9	θ_{ja} °C/W

Appendix C: Mechanical and Ordering Information

This chapter outlines mechanical and ordering information for PowerPro. The topics addressed in this chapter include:

C.1 Mechanical Information

Figure 26: 324 PBGA Package — Bottom View

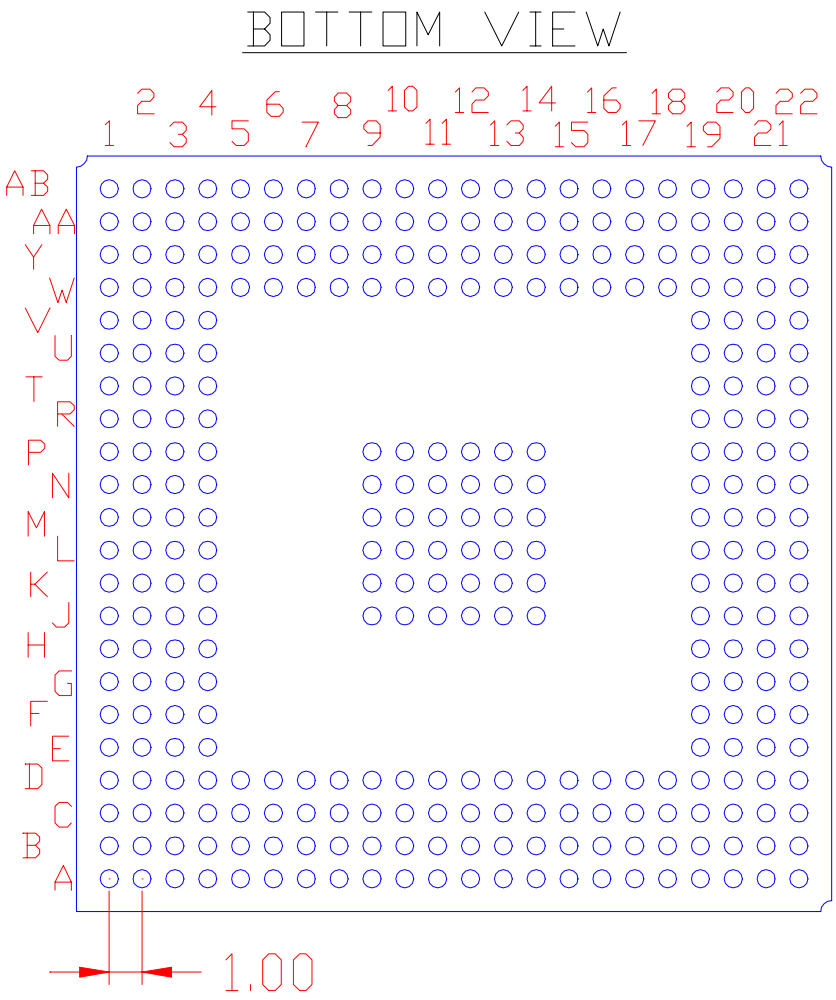


Figure 27: 324 PBGA — Top and Side View

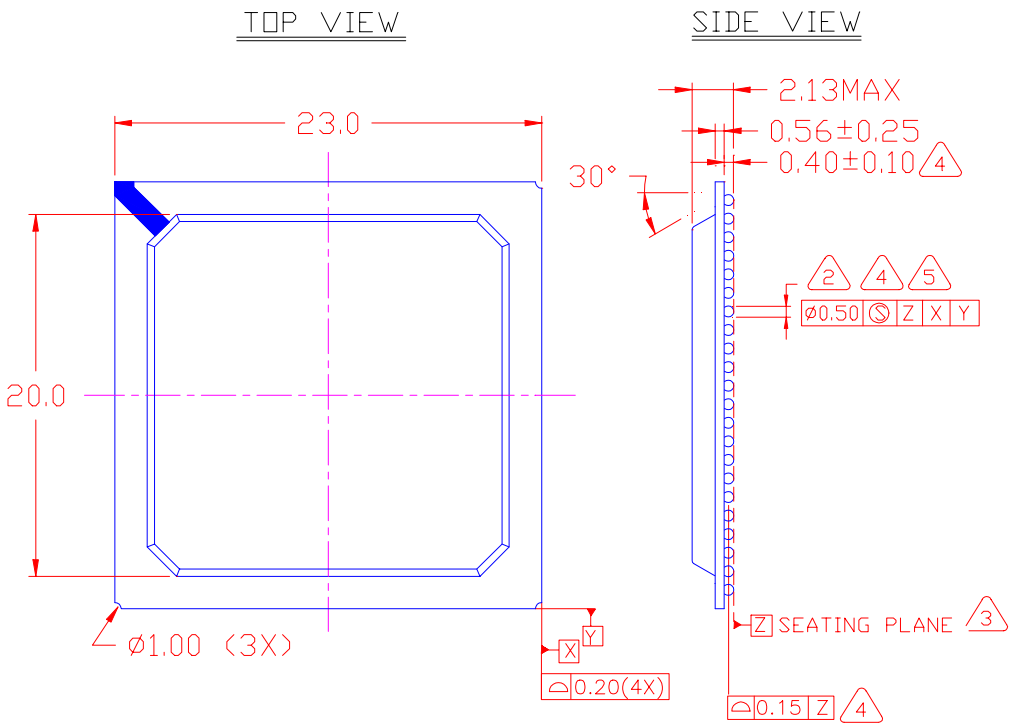


Table 126: Mechanical Information Notes

Number	Description
1	All dimensions conform to ANSI Y14.5-1994. The dimensions are in millimeters (mm).
2	The dimension is measured at the maximum solder ball diameter parallel to the primary datum Z.
3	The primary datum Z and the seating plane are defined by the spherical crowns of the solder balls.
4	The part conforms to Jedec Registered Outline drawing MO-151, Variation AAJ-1 EXCEPT for these dimensions.
7	Pad size is 0.40mm diameter. It is recommended that PCBs have the same pad size.

C.2 Ordering Information

Tundra products are designated by a part number. When ordering, refer to products by their full part number. for detailed mechanical drawings or alternative packaging requirements, please contact Tundra directly. Refer to Chapter 1: “General Information” on page 21 for contact information.

Table 127: PowerPro (CA91L750) Ordering Information

Part Number	Frequency	Voltage	Temperature	Package
CA91L750-100CLZ	100 MHz	3.3V	0°C to 75°C	PBGA

Industrial temperature PowerPro devices will be available when the device reaches production status.

Glossary

Bank

A bank is defined as a memory region defined with a base register and a bank size register. A physical bank of memory is controlled by a single chip select. A DIMM could be comprised of a single bank or dual banks.

Column

A column refers to a portion of memory within an SDRAM device. An SDRAM device can be thought of as a grid with rows and column. Once a row is activated, any column within that row can be accessed multiple times without reactivating the row. Columns are activated with the SD_CAS_ signal.

Cycle

The term cycle refers to a single data beat.

DIMM

A DIMM is an acronym for Dual Inline Memory Module. A DIMM is a physical card comprising multiple memory devices. The card can be populated on one or both sides. A DIMM can be a single bank or a dual bank DIMM.

Leaf

SDRAM use multiple banks within the device operating in an interleaved mode. 16-Mbit SDRAM devices contain two internal banks. An internal bank is referred to as a leaf.

Logical memory banks

This refers to the logical, or virtual, memory separations in the actual SDRAM memory in the DIMM modules. Logical memory refers to an imaginary set of locations, or addresses, where data can be stored. It is imaginary in the sense that the memory area is not the same as the real physical memory composed of transistors.
PowerPro supports four DIMMs of memory. A DIMM could be comprised of a single bank or dual banks. PowerPro supports either 4 logical banks of memory for a single DIMM or 8 logical banks for a dual DIMM.

Master

When discussing bus ownership, this manual uses the term master to indicate bus owner

Page A page is a row of memory. Once a row is activated, any column within that row can be accessed multiple times without having to reactivate the row. This is referred to “keeping the page open”. While it depends on the SDRAM device configuration, PowerPro supports 2 Kbytes for 64-bit wide memory page size. When an SDRAM’s physical configuration supports a larger page size, PowerPro breaks it up into a smaller 2-Kbyte page size.

Physical memory banks This refers to the physical, or actual, SDRAM memory in the DIMM modules. PowerPro supports four DIMMs of memory. A DIMM could be comprised of a single bank or dual banks. This means PowerPro supports one or two physical banks of memory on each DIMM.

Processor (60x) bus Processor (60x) bus refers to the interface between PowerPro and the Processor, to which PowerPro is connected. The processor bus is the 60x bus.

Reset When the term “reset” is used, it includes the signals that can reset PowerPro. The signals are: HRESET_ and PORESET_.

Row A row is a portion of memory within the SDRAM device. An SDRAM device can be thought of as a grid with rows and columns. Once a row is activated, any column within that row can be accessed multiple times without reactivating the row. Rows are activated with the SD_RAS_ signal.

SDRAM SDRAM stands for Synchronous Dynamic Random Access memory. SDRAM is a type of DRAM that can run at much higher clock than conventional memory. SDRAM synchronizes itself with the processor bus and is capable of running at processor bus speed.

Slave The term slave indicates the address accessed by the bus master

Transaction A transaction is composed of one or more cycles.

Index

G

I

J

M

N

P

R

S

T

U

W