

# PowerPC™

## *Advance Information*

# MPC8240 Integrated Processor Technical Summary

This document provides an overview of the MPC8240 PowerPC™ integrated processor for high-performance embedded systems. The MPC8240 is a cost-effective, general-purpose integrated processor for applications using PCI in networking infrastructure, telecommunications, and other embedded markets. It can be used for control processing in applications such as network routers and switches, mass storage subsystems, network appliances, and print and imaging systems.

For errata or revisions of this document, refer to the website at <http://www.mot.com/powerpc>.

## 1.1 MPC8240 Integrated Processor Overview

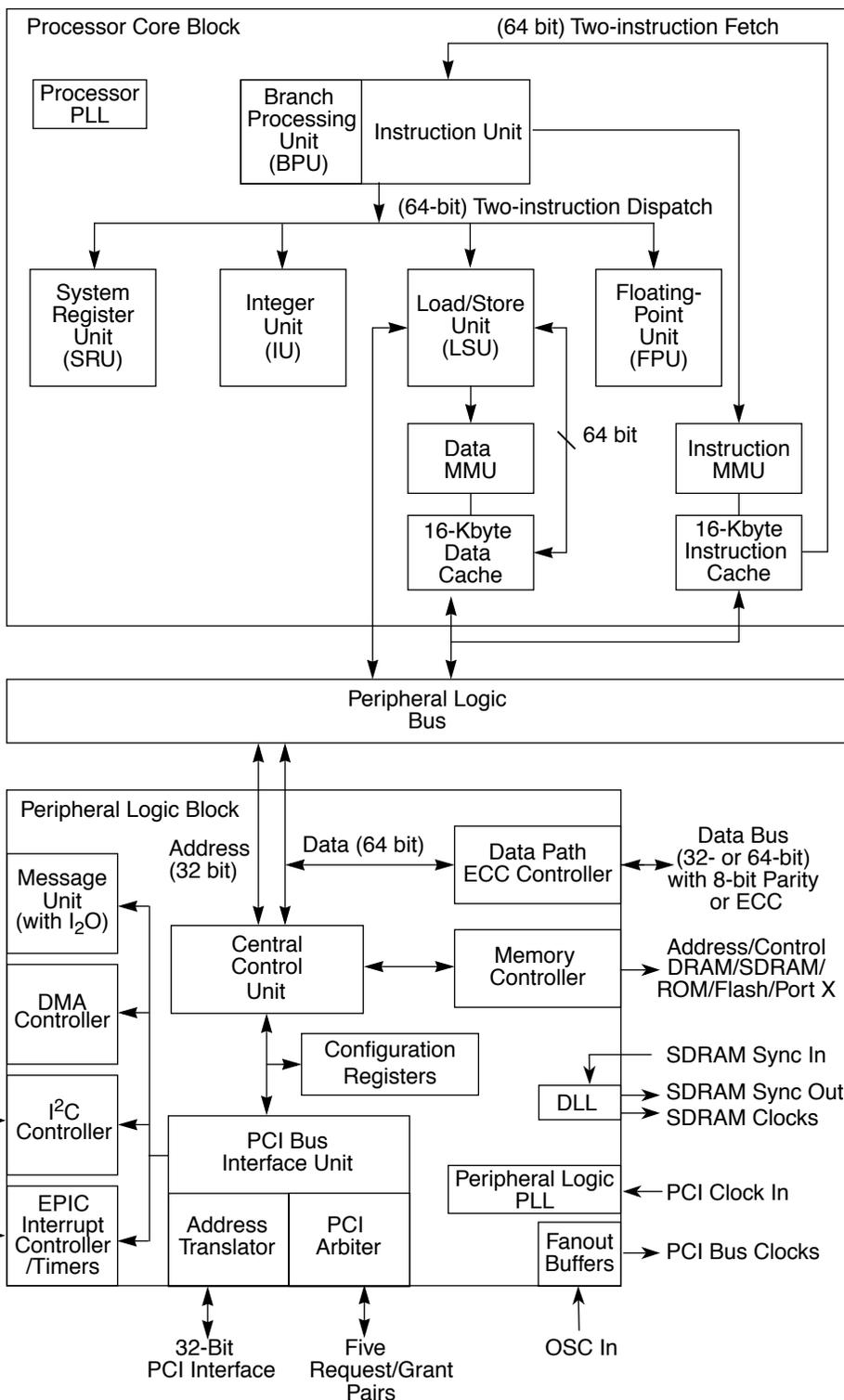
The MPC8240 integrated processor is comprised of a peripheral logic block and a 32-bit superscalar PowerPC processor core, as shown in Figure 1-1.



MPC8240

Additional features:

- JTAG/COP Interface
- Power Management



**Figure 1-1. MPC8240 Integrated Processor Functional Block Diagram**

The peripheral logic integrates a PCI bridge, memory controller, DMA controller, EPIC interrupt controller, a message unit (and I<sub>2</sub>O controller), and an I<sup>2</sup>C controller. The processor core is a full-featured, high-performance processor with floating-point support, memory management, 16-Kbyte instruction cache, 16-

**MPC8240 Integrated Processor Overview**

Kbyte data cache, and power management features. The integration reduces the overall packaging requirements and the number of discrete devices required for an embedded system.

The MPC8240 contains an internal peripheral logic bus that interfaces the processor core to the peripheral logic. The core can operate at a variety of frequencies, allowing the designer to trade off performance for power consumption. The processor core is clocked from a separate PLL, which is referenced to the peripheral logic PLL. This allows the microprocessor and the peripheral logic block to operate at different frequencies, while maintaining a synchronous bus interface. The interface uses a 64- or 32-bit data bus (depending on memory data bus width) and a 32-bit address bus along with control signals that enable the interface between the processor and peripheral logic to be optimized for performance. PCI accesses to the MPC8240's memory space are passed to the processor bus for snooping when snoop mode is enabled.

The processor core and peripheral logic are general-purpose in order to serve a variety of embedded applications. The MPC8240 can be used as either a PCI host or PCI agent controller.

### 1.1.1 MPC8240 Integrated Processor Features

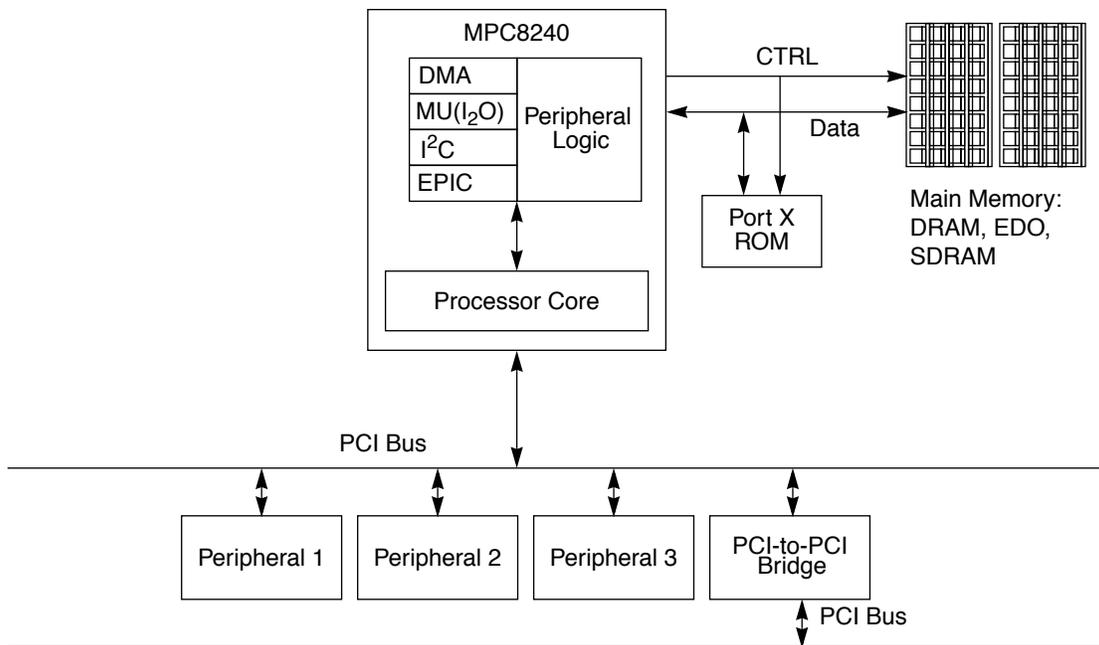
Major features of the MPC8240 are as follows:

- Peripheral logic
  - Memory interface
    - Programmable timing supporting either FPM DRAM, EDO DRAM or SDRAM
    - High-bandwidth bus (32/64-bit data bus) to DRAM
    - Supports one to eight banks of 4-, 16-, 64-, or 128-Mbit memory devices
    - Supports 1-Mbyte to 1-Gbyte DRAM memory
    - Contiguous memory mapping
    - 16 Mbytes of ROM space
    - 8-, 32-, or 64-bit ROM
    - Write buffering for PCI and processor accesses
    - Supports normal parity, read-modify-write (RMW), or ECC
    - SDRAM data-path buffer
    - Low voltage TTL logic (LVTTL)
    - Port X: 8-, 32-, or 64-bit general-purpose I/O port using ROM controller interface with address strobe
  - 32-bit PCI interface operating up to 66 MHz
    - PCI 2.1-compatible
    - PCI 5.0-V tolerance
    - Support for PCI locked accesses to memory
    - Support for accesses to all PCI address spaces
    - Selectable big- or little-endian operation
    - Store gathering of processor-to-PCI write and PCI-to-memory write accesses
    - Memory prefetching of PCI read accesses
    - Selectable hardware-enforced coherency
    - PCI bus arbitration unit (five request/grant pairs)

- PCI agent mode capability
  - Address translation unit
  - Internal configuration registers accessible from PCI
- Two-channel integrated DMA controller
  - Supports direct mode or chaining mode (automatic linking of DMA transfers)
  - Supports scatter gathering—read or write discontinuous memory
  - Interrupt on completed segment, chain, and error
  - Local-to-local memory
  - PCI-to-PCI memory
  - PCI-to-local memory
  - Local-to-PCI memory
- Message unit
  - Two doorbell registers
  - Inbound and outbound messaging registers
  - I<sub>2</sub>O message controller
- I<sup>2</sup>C controller with full master/slave support
- Embedded programmable interrupt controller (EPIC)
  - Five hardware interrupts (IRQs) or 16 serial interrupts
  - Four programmable timers
- Integrated PCI bus and SDRAM clock generation
- Programmable memory and PCI bus output drivers
- Debug features
  - Memory attribute and PCI attribute signals
  - Debug address signals
  - $\overline{MIV}$  signal: Marks valid address and data bus cycles on the memory bus.
  - Error injection/capture on data path
  - IEEE 1149.1 (JTAG)/test interface
- Processor core
  - High-performance, superscalar processor core
  - Integer unit (IU), floating-point unit (FPU) (user enabled or disabled), load/store unit (LSU), system register unit (SRU), and a branch processing unit (BPU)
  - 16-Kbyte instruction cache
  - 16-Kbyte data cache
  - Lockable L1 cache—entire cache or on a per-way basis
  - Dynamic power management

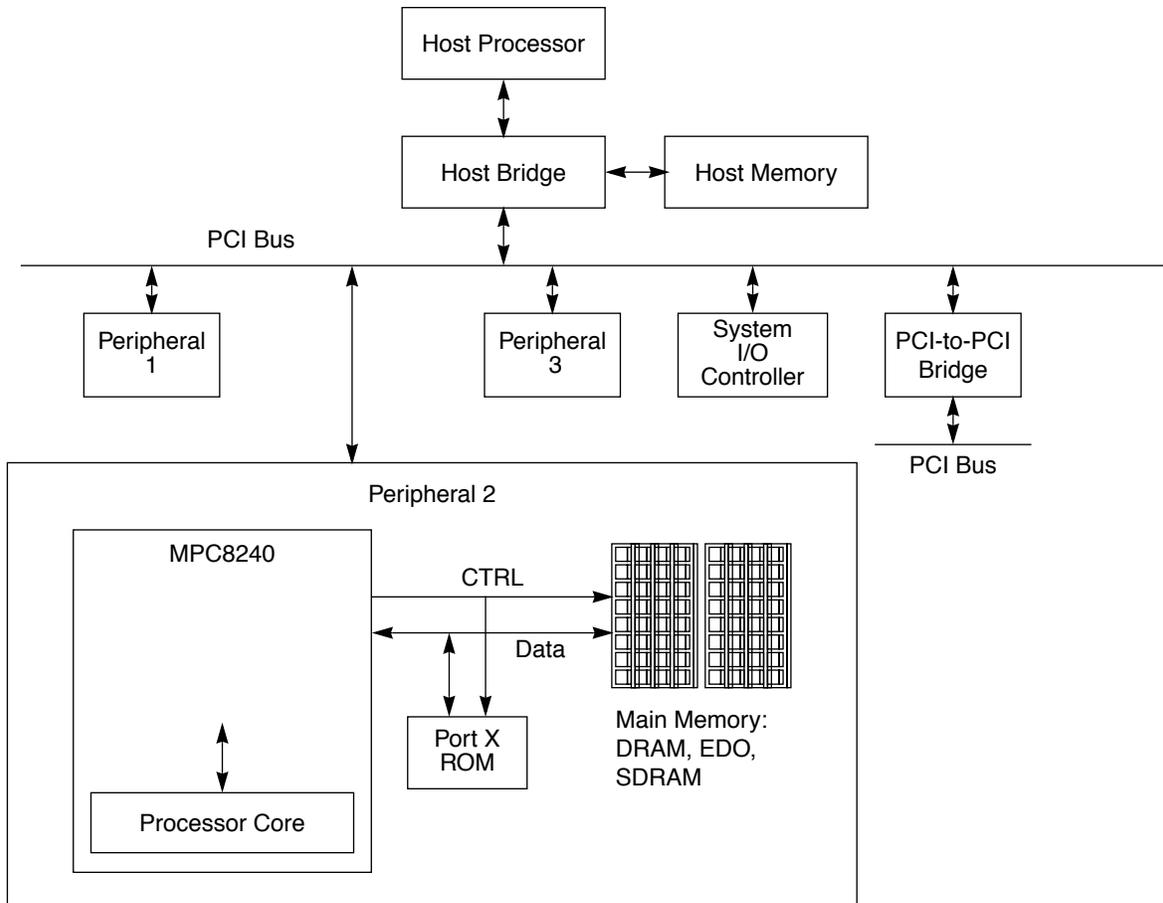
## 1.1.2 MPC8240 Integrated Processor Applications

The MPC8240 can be used for control processing in applications such as routers, switches, multi-channel modems, network storage, image display systems, enterprise I/O processor, Internet access device (IAD), disk controller for RAID systems, and copier/printer board control. Figure 1-2 shows the MPC8240 in the role of host processor.



**Figure 1-2. System Using an Integrated MPC8240 as a Host Processor**

Figure 1-3 shows the MPC8240 in a distributed I/O processor application.



**Figure 1-3. Embedded System Using an MPC8240 as a Distributed Processor**

Figure 1-4 shows the MPC8240 as a peripheral processing device. The PCI-to-PCI bridge shown could be of the PCI type 0 variety. The MPC8240 would not be part of the system configuration map. This configuration is useful in applications such as RAID controllers, where the I/O devices shown are SCSI controllers, or multi-port network controllers where the devices shown are Ethernet controllers.

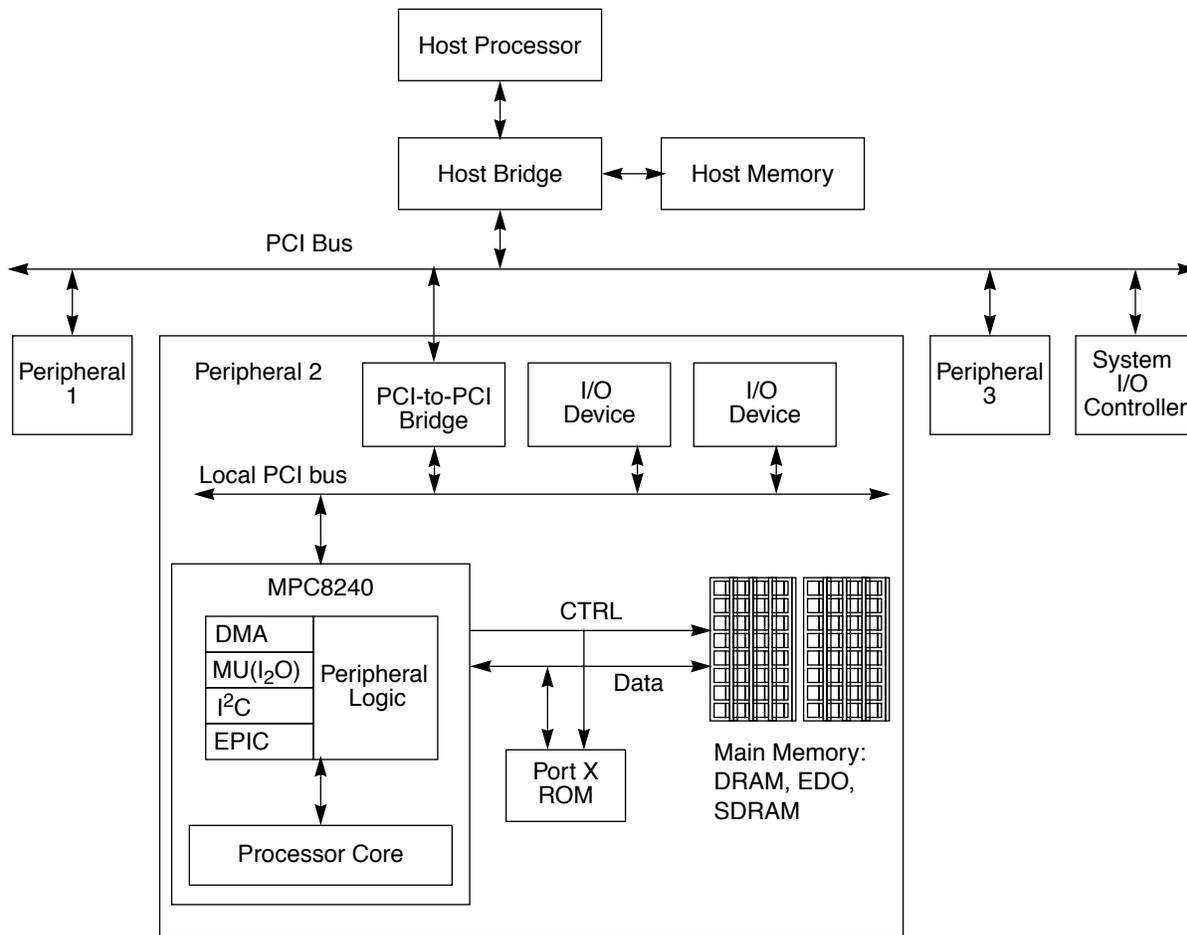


Figure 1-4. Embedded System Using an MPC8240 as a Peripheral Processor

## 1.2 Processor Core Overview

The MPC8240 contains an embedded version of the PowerPC 603e™ processor. For detailed information regarding the processor refer to the following:

- *MPC603e & EC603e User's Manual* (Those chapters that describe the programming model, cache model, memory management model, exception model, and instruction timing)
- *PowerPC Microprocessor Family: The Programming Environments for 32-Bit Microprocessors*

This section is an overview of the processor core, provides a block diagram showing the major functional units, and describes briefly how those units interact. For more information, refer to Chapter 2, "PowerPC Processor Core," in the *MPC8240 User's Manual*.

The processor core is a low-power implementation of the PowerPC microprocessor family of reduced instruction set computing (RISC) microprocessors. The processor core implements the 32-bit portion of the PowerPC architecture, which provides 32-bit effective addresses, integer data types of 8, 16, and 32 bits, and floating-point data types of 32 and 64 bits.

The processor core is a superscalar processor that can issue and retire as many as three instructions per clock. Instructions can execute out of order for increased performance; however, the processor core makes completion appear sequential.

The processor core integrates five execution units—an integer unit (IU), a floating-point unit (FPU), a branch processing unit (BPU), a load/store unit (LSU), and a system register unit (SRU). The ability to execute five instructions in parallel and the use of simple instructions with rapid execution times yield high efficiency and throughput. Most integer instructions execute in one clock cycle. On the processor core, the FPU is pipelined so a single-precision multiply-add instruction can be issued and completed every clock cycle.

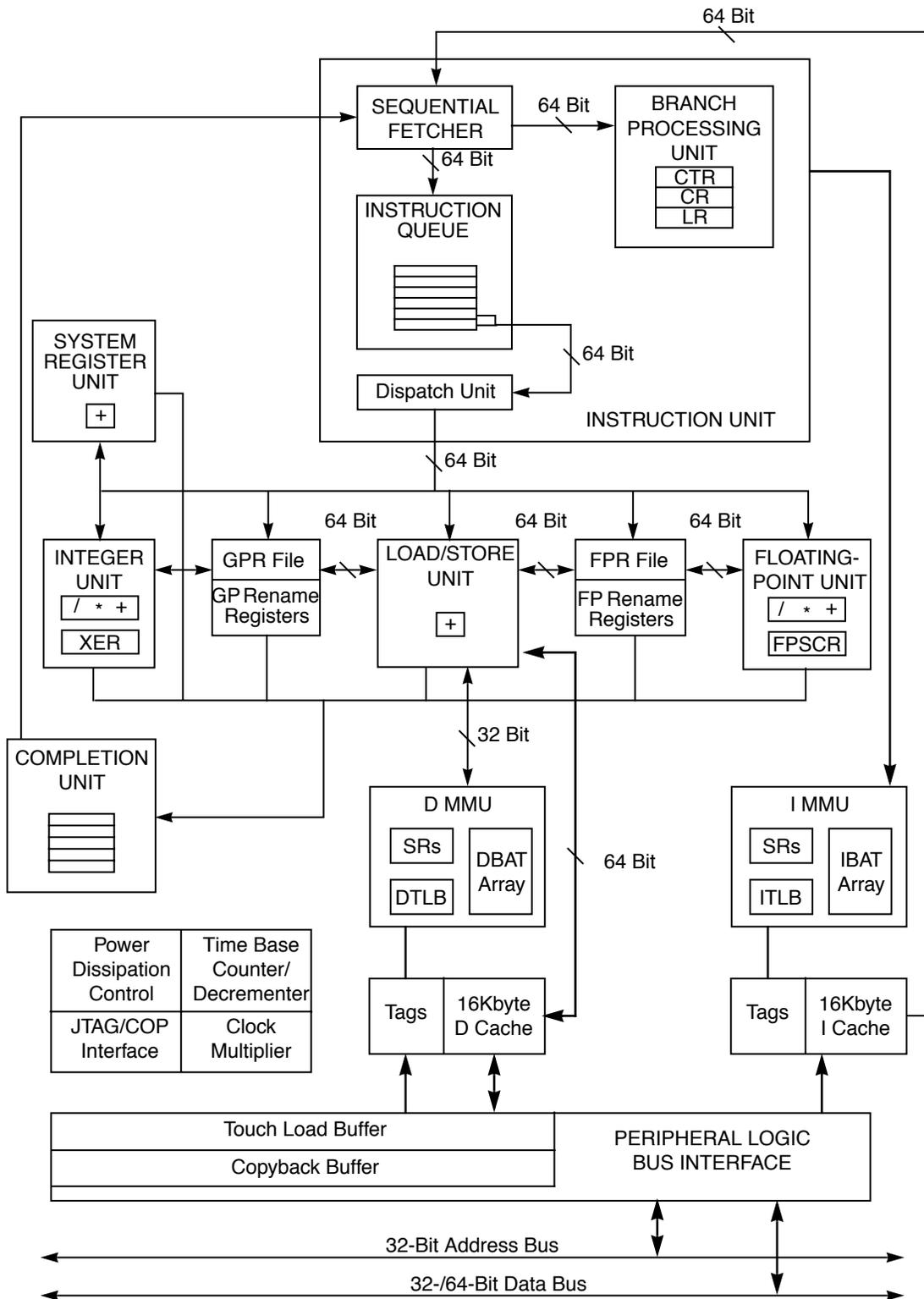
The processor core supports integer data types of 8, 16, and 32 bits, and floating-point data types of 32 and 64 bits.

The processor core provides independent on-chip, 16-Kbyte, four-way set-associative, physically addressed caches for instructions and data and on-chip instruction and data memory management units (MMUs). The MMUs contain 64-entry, two-way set-associative, data and instruction translation lookaside buffers (DTLB and ITLB) that provide support for demand-paged virtual memory address translation and variable-sized block translation. The TLBs and caches use a least recently used (LRU) replacement algorithm. The processor also supports block address translation through the use of two independent instruction and data block address translation (IBAT and DBAT) arrays of four entries each. Effective addresses are compared simultaneously with all four entries in the BAT array during block translation. In accordance with the PowerPC architecture, if an effective address hits in both the TLB and BAT array, the BAT translation takes priority.

As an added feature to the processor core, the MPC8240 can lock the contents of one to three ways in the instruction and data cache (or an entire cache). For example, this allows embedded applications to lock interrupt routines or other important (time-sensitive) instruction sequences into the instruction cache. It allows data to be locked into the data cache, which may be important to code that must have deterministic execution.

The processor core has a selectable 32- or 64-bit data bus and a 32-bit address bus. The processor core supports single-beat and burst data transfers for memory accesses, and supports memory-mapped I/O operations.

Figure 1-5 provides a block diagram of the MPC8240 processor core that shows how the execution units (IU, FPU, BPU, LSU, and SRU) operate independently and in parallel. Note that this is a conceptual diagram and does not attempt to show how these features are physically implemented on the chip.



**Figure 1-5. MPC8240 Integrated Processor Core Block Diagram**

## 1.3 Peripheral Logic Bus

The MPC8240 contains an internal peripheral logic bus that interfaces the processor core to the peripheral logic. The core can operate at a variety of frequencies allowing the designer to balance performance and power consumption. The processor core is clocked from a separate PLL, which is referenced to the peripheral logic PLL. This allows the microprocessor and the peripheral logic to operate at different frequencies while maintaining a synchronous bus interface.

The processor core-to-peripheral logic interface includes a 32-bit address bus, a 32- or 64-bit data bus as well as control and information signals. The peripheral logic bus allows for internal address-only transactions as well as address and data transactions. The processor core control and information signals include the address arbitration, address start, address transfer, transfer attribute, address termination, data arbitration, data transfer, data termination, and processor state signals. Test and control signals provide diagnostics for selected internal circuits.

The peripheral logic interface supports bus pipelining, which allows the address tenure of one transaction to overlap the data tenure of another. PCI accesses to the memory space are monitored by the peripheral logic bus to allow the processor to snoop these accesses (when snooping not explicitly disabled).

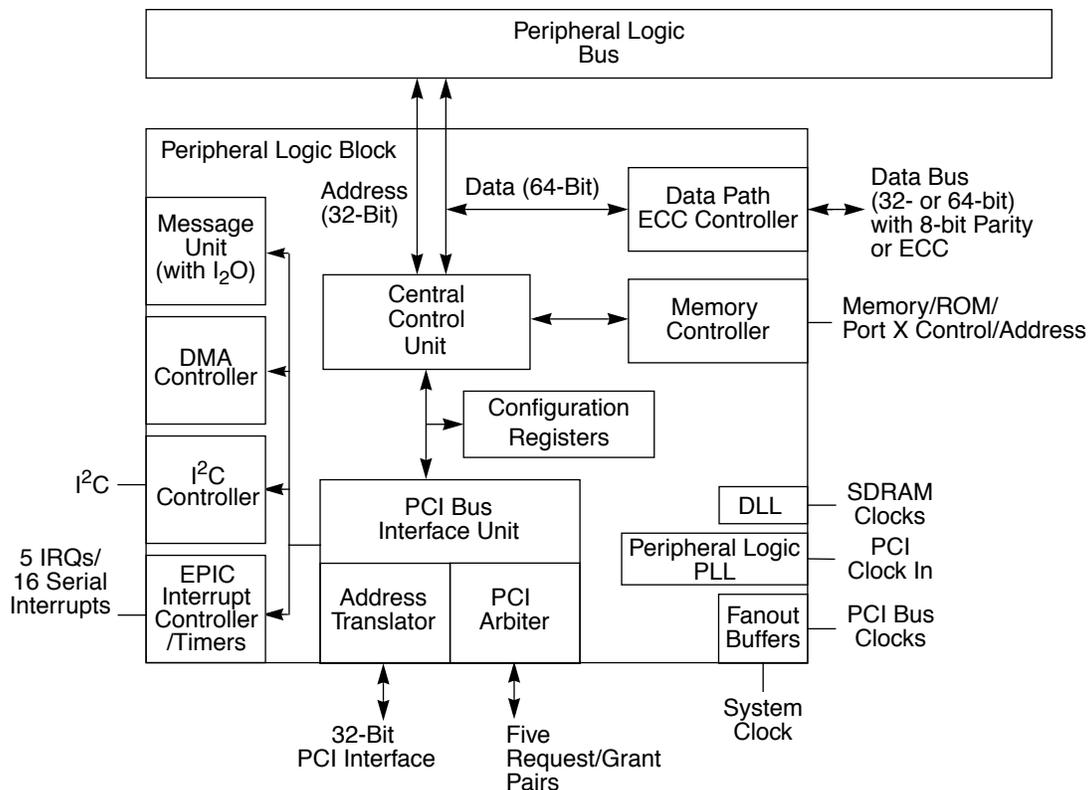
As part of the peripheral logic bus interface, the processor core's data bus is configured at power-up to either a 32- or 64-bit width. When the processor is configured with a 32-bit data bus, memory accesses on the peripheral logic bus interface allow transfer sizes of 8, 16, 24, or 32 bits in one bus clock cycle. Data transfers occur in either single-beat transactions, or two-beat or eight-beat burst transactions, with a single-beat transaction transferring as many as 32 bits. Single- or double-beat transactions are caused by noncached accesses that access memory directly (that is, reads and writes when caching is disabled, caching-inhibited accesses, and stores in write-through mode). Eight-beat burst transactions, which always transfer an entire cache line (32 bytes), are initiated when a line is read from or written to memory.

When the peripheral logic bus interface is configured with a 64-bit data bus, memory accesses allow transfer sizes of 8, 16, 24, 32, or 64 bits in one bus clock cycle. Data transfers occur in either single-beat transactions or four-beat burst transactions. Single-beat transactions are caused by noncached accesses that access memory directly (that is, reads and writes when caching is disabled, caching-inhibited accesses, and stores in write-through mode). Four-beat burst transactions, which always transfer an entire cache line (32 bytes), are initiated when a block is read from or written to memory.

## 1.4 Peripheral Logic Overview

The peripheral logic block integrates a PCI bridge, memory controller, DMA controller, EPIC interrupt controller/timers, a message unit with an Intelligent Input/Output (I<sub>2</sub>O) message controller, and an Inter-Integrated Circuit (I<sup>2</sup>C) controller. The integration reduces the overall packaging requirements and the number of discrete devices required for an embedded system.

Figure 1-6 shows the major functional units within the peripheral logic block. Note that this is a conceptual block diagram intended to show the basic features rather than an attempt to show how these features are physically implemented.



**Figure 1-6. MPC8240 Peripheral Logic Block Diagram**

### 1.4.1 Peripheral Logic Features

Major features of the peripheral logic are as follows:

- Peripheral logic bus
  - Supports various operating frequencies and bus divider ratios
  - 32-bit address bus, 64-bit data bus
  - Supports full memory coherency
  - Decoupled address and data buses for pipelining of peripheral logic bus accesses
  - Store gathering on peripheral logic bus-to-PCI writes
- Memory interface
  - 1 Gbyte of RAM space, 16 Mbytes of ROM space
  - High-bandwidth, 64-bit data bus (72 bits including parity or ECC)
  - Supports fast page mode DRAMs, extended data out (EDO) DRAMs, or synchronous DRAMs (SDRAMs)
  - Supports 1 to 8 banks of DRAM/EDO/SDRAM with sizes ranging from 1 to 128 Mbytes per bank
  - Supports page mode SDRAMs—four open pages simultaneously
  - DRAM/EDO configurations support parity or error checking and correction (ECC); SDRAM configurations support ECC
  - ROM space may be split between the PCI bus and the memory bus (8 Mbytes each)

- Supports 8-bit asynchronous ROM, or 32- or 64-bit burst-mode ROM
- Supports writing to flash ROM
- Configurable data path
- Programmable interface timing
- PCI interface
  - Compatible with PCI Local Bus Specification, Revision 2.1
  - Supports PCI locked accesses to memory using the  $\overline{\text{LOCK}}$  signal and protocol
  - Supports accesses to all PCI address spaces
  - Selectable big- or little-endian operation
  - Store gathering on PCI writes to memory
  - Selectable memory prefetching of PCI read accesses
  - Interface operates at up to 66 MHz
  - Parity support
- Supports concurrent transactions on peripheral logic bus and PCI buses

### 1.4.2 Peripheral Logic Functional Units

The peripheral logic consists of the following major functional units:

- Peripheral logic bus interface
- Memory interface
- PCI interface
  - PCI bus arbitration unit
  - Address maps and translation
  - Big-and little-endian modes
  - PCI agent capability
  - PCI bus clock buffers and bus ratios
- DMA controller
- Message unit
  - Doorbell registers
  - Message registers
  - I<sub>2</sub>O support (circular queues)
- Embedded programmable interrupt controller (EPIC) with four timers
- I<sup>2</sup>C interface

### 1.4.3 Memory System Interface

The MPC8240 memory interface controls processor and PCI interactions to main memory. It supports a variety of DRAM, and flash or ROM configurations as main memory. The MPC8240 supports FPM (fast page mode), EDO (extended data out) and synchronous DRAM (SDRAM). The maximum supported memory size is 1 Gbyte of DRAM or SDRAM and 16 Mbytes of ROM/flash. SDRAM must comply with the JEDEC SDRAM specification.

The MPC8240 implements Port X, a flexible memory bus interface that facilitates the connection of general purpose I/O devices. The Port X functionality allows the designer to connect external registers,

## Peripheral Logic Overview

communication devices, and other such devices directly to the MPC8240. Some devices may require a small amount of external logic to properly generate address strobes, chip selects, and other signals.

The MPC8240 is designed to control a 32-bit or 64-bit data path to main memory DRAM or SDRAM. For a 32-bit data path, the MPC8240 can be configured to check and generate byte parity using four parity bits. For a 64-bit data path, the MPC8240 can be configured to support parity or ECC checking and generation with eight parity/syndrome bits checked and generated.

The MPC8240 supports DRAM or SDRAM bank sizes from 1 to 128 Mbytes and provides bank start address and end address configuration registers, but MPC8240 does not support mixed DRAM/SDRAM configurations. MPC8240 can be configured so that appropriate row and column address multiplexing occurs according to the accessed memory bank. Addresses are provided to DRAM and SDRAM through a 13-bit interface for DRAM and 14-bit interface for SDRAM.

Two bank selects, one write enable, one output enable, and up to 21 address signals are provided for ROM/flash systems.

### 1.4.4 Peripheral Component Interface (PCI)

The PCI interface for the MPC8240 is compatible with the Peripheral Component Interconnect Specification Revision 2.1. Mode-selectable, big- to little-endian conversion is supplied at the PCI interface. The MPC8240 provides an interface to the PCI bus running at speeds up to 66 MHz.

The MPC8240's PCI interface can be configured as host or agent. In host mode the interface acts as the main memory controller for the system and responds to all host memory transactions.

In agent mode, the MPC8240 can be configured to respond to a programmed window of PCI memory space. A variety of initialization modes are provided to boot the device.

#### 1.4.4.1 PCI Bus Arbitration Unit

The MPC8240 contains a PCI bus arbitration unit, which reduces the need for an equivalent external unit, thus lowering system complexity and cost. It has the following features:

- The unit can be disabled to allow a remote arbitration unit to be used.
- Five external arbitration signal pairs. The MPC8240 is the sixth member of the arbitration pool.
- The bus arbitration unit allows fairness as well as a priority mechanism.
- A two-level round-robin scheme is used in which each device can be programmed within a pool of a high- or low-priority arbitration. One member of the low-priority pool is promoted to the high-priority pool. As soon as it is granted the bus, it returns to the low-priority pool.

#### 1.4.4.2 Address Maps and Translation

The MPC8240's processor bus supports memory-mapped accesses. The address space is divided between memory and PCI according to one of two allowable address maps. An in-bound and out-bound PCI address translation mechanism is provided to support the use of the MPC8240 in agent mode. Note that only map B is supported in agent mode.

When the MPC8240 is used as a peripheral processor and not the primary host, it is possible that only a portion of the memory controlled by it may be visible to the system. In addition, it may be necessary to map the local memory to different system memory address space. The address translation unit handles the mapping of both in-bound and out-bound transactions for these cases.

#### 1.4.4.3 Byte Ordering

The MPC8240 allows the processor to run in either big- or little-endian mode (except for the initial boot code which must run in big-endian mode).

#### 1.4.4.4 PCI Agent Capability

In certain applications the embedded system architecture dictates that the MPC8240 act as peripheral processor. In this case the peripheral logic must not act like a host bridge for the PCI bus. Instead it functions as a configurable device that is accessed by a host bridge. This capability allows multiple MPC8240 devices to coexist with other PCI peripheral devices on a single PCI bus. The MPC8240 contains PCI 2.1-compatible configuration capabilities.

#### 1.4.5 DMA Controller

The integrated DMA controller contains two independent units, each capable of performing the following types of transfers:

- PCI-to-local memory
- Local-to-PCI memory
- PCI-to-PCI memory
- Local-to-local memory

The DMA controller allows chaining through local memory-mapped chain descriptors. Transfers can be scatter gathered and misaligned. Interrupts are provided on completed segment, chain, and error conditions.

#### 1.4.6 Message Unit (MU)

Many embedded applications require handshake algorithms to pass control, status, and data information from one owner to another. This is made easier with doorbell and message registers. The MPC8240 has a message unit (MU) that implements doorbell and message registers as well as an I<sub>2</sub>O interface. The MU has many conditions that can cause interrupts and uses the EPIC unit to signal external interrupts to the PCI interface and internal interrupts to the processor core.

##### 1.4.6.1 Doorbell Registers

The MPC8240 MU contains one 32-bit inbound doorbell register and one 32-bit outbound doorbell register. The inbound doorbell register allows a remote processor to set a bit in the register from the PCI bus. This, in turn, generates an interrupt to the processor core.

The processor core can write to the outbound register, causing the outbound interrupt signal  $\overline{INTA}$  to assert, thus interrupting the host processor. Once  $\overline{INTA}$  is generated, it can be cleared only by the host processor by writing ones to the bits that are set in the outbound doorbell register.

##### 1.4.6.2 Inbound and Outbound Message Registers

The MPC8240 contains two 32-bit inbound message registers and two 32-bit outbound message registers. The inbound registers allow a remote host or PCI master to write a 32-bit value, causing an interrupt to the processor core. The outbound registers allow the processor core to write an outbound message which causes the outbound interrupt signal  $\overline{INTA}$  to assert.

##### 1.4.6.3 Intelligent Input/Output Controller (I<sub>2</sub>O)

The intelligent I/O specification is an open standard that defines an abstraction layer interface between the OS and subsystem drivers. Messages are passed between the message abstraction layer from one device to another.

The I<sub>2</sub>O specification describes a system as being made up of host processors and input/output platforms (IOPs). The host processor is a single processor or a collection of processors working together to execute a homogenous operating system. An IOP consists of a processor, memory, and I/O interfaces. The IOP functions separately from other processors within the system to handle system I/O functions.

The I<sub>2</sub>O controller of the MU enhances communication between hosts and IOPs within a system. The MU maintains a set of FIFO buffers located in local IOP memory. Four queues are provided: two for inbound messages and two for outbound messages. The inbound message queues are used to transfer messages from a remote host or IOP to the processor core. The outbound queues are used to transfer messages from the processor core to the remote host. Messages are transferred between the host and the IOP using PCI memory-mapped registers. The MPC8240's I<sub>2</sub>O controller facilitates moving the messages to and from the inbound and outbound registers and local IOP memory. Interrupts signal the host and IOP to indicate the arrival of new messages.

### 1.4.7 Inter-Integrated Circuit (I<sup>2</sup>C) Controller

The I<sup>2</sup>C serial interface has become an industry de-facto standard for communicating with low-speed peripherals. Typically, it is used for system management functions and EEPROM support. The MPC8240 contains an I<sup>2</sup>C controller with full master and slave functionality.

### 1.4.8 Embedded Programmable Interrupt Controller (EPIC)

The integrated hardware interrupt controller reduces the overall component count in embedded applications. The embedded programmable interrupt controller (EPIC) is designed to collect external and internal hardware interrupts, prioritize them, and deliver them to the processor core.

The module operates in two modes:

- In direct mode, five level- or edge-triggered interrupts can be connected directly to an MPC8240.
- The MPC8240 provides a serial delivery mechanism for when more than five external interrupt sources are needed. The serial mechanism allows for up to 16 interrupts to be serially scanned into the MPC8240. This mechanism increases the number of interrupts without increasing the number of pins.

The outbound interrupt request signal,  $\overline{L\_INT}$ , is used to signal interrupts to the host processor when the MPC8240 is configured for agent mode. The MPC8240 EPIC includes four programmable timers that can be used for system timing or for generating periodic interrupts.

### 1.4.9 Integrated PCI Bus and SDRAM Clock Generation

There are two clocking solutions directed towards different system requirements. For systems where the MPC8240 is the host controller with a minimum number of clock loads, clock fan-out buffers are provided on chip.

For systems requiring more clock fan out or where the MPC8240 is an agent device, external clock buffers may be used.

### 1.4.10 Bus Ratios

The MPC8240 requires a single clock input signal, PCI\_SYNC\_IN, which can be driven by the PCI clock fan-out buffers—specifically the PCI\_SYNC\_OUT output. PCI\_SYNC\_IN can also be driven by an external clock driver.

PCI\_SYNC\_IN is driven by the PCI bus frequency. An internal PLL, using PCI\_SYNC\_IN as a reference, generates an internal peripheral bus clock that is used for the internal logic. The peripheral bus clock frequency is selectable by the MPC8240 PLL configuration signals (PLL\_CFG[0–4]) to be a multiple of the PCI\_SYNC\_IN frequency.

The MPC8240 provides an on-chip delay-locked loop (DLL) that can supply an external memory bus clock to SDRAM banks. The memory bus clock is of the same frequency and synchronous with the internal peripheral bus clock.

The internal clocking of the processor core is generated from and synchronized to the internal peripheral bus clock by means of a second PLL. The core's PLL provides multiples of the internal processor core clock rates as specified in the *MPC8240 Hardware Specification*.

## 1.5 Power Management

The MPC8240 provides both automatic and program-controllable power reduction modes for progressive reduction of power consumption.

The MPC8240 has independent power management functionality for both the processor core and the peripheral logic. The MPC8240 provides hardware support for three levels of programmable power reduction for both the processor and the peripheral logic. Doze, nap, and sleep modes are invoked by register programming—HID0 in the case of the processor core and configuration registers in the case of the peripheral logic block. The processor and peripheral logic blocks are both fully static, allowing internal logic states to be preserved during all power-saving modes. The following sections describe the programmable power modes.

### 1.5.1 Programmable Processor Power Management Modes

Table 1-1 summarizes the programmable power-saving modes for the processor core. These are very similar to those available in the MPC603e device.

**Table 1-1. Programmable Processor Power Modes**

PM Mode	Functioning Units	Activation Method	Full-Power Wake Up Method
Full power	All units active	—	—
Full power (with DPM)	Requested logic by demand	By instruction dispatch	—
Doze	Bus snooping Data cache as needed Decrementer timer	Controlled by software (write to HID0)	External asynchronous exceptions (assertion of $\overline{SMI}$ or $\overline{int}$ ), Decrementer exception Hard or soft reset Machine check exception ( $\overline{mcp}$ )
Nap	Decrementer timer	Controlled by software (write to HID0) and qualified with $\overline{QACK}$ from peripheral logic	External asynchronous exceptions (assertion of $\overline{SMI}$ , or $\overline{int}$ ) Decrementer exception Negation of $\overline{QACK}$ by peripheral logic Hard or soft reset Machine check exception ( $\overline{mcp}$ )
Sleep	None	Controlled by software (write to HID0) and qualified with $\overline{QACK}$ from peripheral logic	External asynchronous exceptions (assertion of $\overline{SMI}$ , or $\overline{int}$ ) Negation of $\overline{QACK}$ by peripheral logic Hard or soft reset Machine check exception ( $\overline{mcp}$ )

### 1.5.2 Peripheral Logic Power Management Modes

The following subsections describe the power management modes of the peripheral logic. Table 1-2 summarizes the programmable power-saving modes for the peripheral logic block.

**Table 1-2. Programmable Peripheral Logic Power Modes**

PM Mode	Functioning Units	Activation Method	Full-Power Wake Up Method
Full power	All units active	—	—
Doze	PCI address decoding and bus arbiter System RAM refreshing Processor bus request and NMI monitoring EPIC unit I <sup>2</sup> C unit PLL	Controlled by software (write to PMCR)	PCI access to memory Processor bus request Assertion of NMI Interrupt to EPIC Hard Reset
Nap	PCI address decoding and bus arbiter System RAM refreshing Processor bus request and NMI monitoring EPIC unit I <sup>2</sup> C unit PLL	Controlled by software (write to PMCR) and processor core in nap or sleep mode	PCI access to memory <sup>1</sup> Processor bus request Assertion of NMI Interrupt to EPIC Hard Reset
Sleep	PCI bus arbiter System RAM refreshing (can be disabled) Processor bus request and NMI monitoring EPIC unit I <sup>2</sup> C unit PLL (can be disabled)	Controlled by software (write to PMCR) and processor core in nap or sleep mode	Processor bus request Assertion of NMI Interrupt to EPIC Hard Reset

<sup>1</sup> A PCI access to memory in nap mode does not cause  $\overline{QACK}$  to negate; subsequently, it does not wake up the processor core. Thus the processor won't snoop this access. After servicing the PCI access, the peripheral logic automatically returns to the nap mode.

## 1.6 Debug Features

The MPC8240 includes the following debug features:

- Memory attribute and PCI attribute signals
- Debug address signals
- $\overline{MIV}$  signal: Marks valid address and data bus cycles on the memory bus.
- Error injection/capture on data path
- IEEE 1149.1 (JTAG)/test interface

### 1.6.1 Memory Attribute and PCI Attribute Signals

The MPC8240 provides additional information corresponding to memory and PCI activity on several signals to assist with system debugging. The two types of attribute signals are described as follows:

- The memory attribute signals are associated with the memory interface and provide information concerning the source of the memory operation being performed by the MPC8240.
- The PCI attribute signals are associated with the PCI interface and provide information concerning the source of the PCI operation being performed by the MPC8240.

## 1.6.2 Memory Debug Address

When enabled, the debug address provides software disassemblers a simple way to reconstruct the 30-bit physical address for a memory bus transaction to DRAM and SDRAM, ROM, FLASH, or PortX. For DRAM or SDRAM, these 16 debug address signals are sampled with the column address and chip-selects. For ROMs, FLASH, and PortX devices, the debug address pins are sampled at the same time as the ROM address and can be used to recreate the 24-bit physical address in conjunction with ROM address. The granularity of the reconstructed physical address is limited by the bus width of the interface; double-words for 64-bit interfaces, words for 32-bit interfaces, and bytes for 8-bit interfaces.

## 1.6.3 Memory Interface Valid ( $\overline{MIV}$ )

The memory interface valid signal,  $\overline{MIV}$ , is asserted whenever FPM, EDO, SDRAM, FLASH, or ROM addresses or data are present on the external memory bus. It is intended to help reduce the number of bus cycles that logic analyzers must store in memory during a debug trace.

## 1.6.4 Error Injection/Capture on Data Path

The MPC8240 provides hardware to exercise and debug the ECC and parity logic by allowing the user to inject multi-bit stuck-at faults onto the peripheral logic or memory data/parity busses and to capture the data/parity output on receipt of an ECC or parity error.

## 1.6.5 IEEE 1149.1 (JTAG)/Test Interface

The processor core provides IEEE 1149.1 functions for facilitating board testing and debugging. The IEEE 1149.1 test interface provides a means for boundary-scan testing the processor core and the board to which it is attached.



**Home Page:**

[www.freescale.com](http://www.freescale.com)

**email:**

[support@freescale.com](mailto:support@freescale.com)

**USA/Europe or Locations Not Listed:**

Freescale Semiconductor  
 Technical Information Center, CH370  
 1300 N. Alma School Road  
 Chandler, Arizona 85224  
 (800) 521-6274  
 480-768-2130

[support@freescale.com](mailto:support@freescale.com)

**Europe, Middle East, and Africa:**

Freescale Halbleiter Deutschland GmbH  
 Technical Information Center  
 Schatzbogen 7  
 81829 Muenchen, Germany  
 +44 1296 380 456 (English)  
 +46 8 52200080 (English)  
 +49 89 92103 559 (German)  
 +33 1 69 35 48 48 (French)  
[support@freescale.com](mailto:support@freescale.com)

**Japan:**

Freescale Semiconductor Japan Ltd.  
 Headquarters  
 ARCO Tower 15F  
 1-8-1, Shimo-Meguro, Meguro-ku  
 Tokyo 153-0064, Japan  
 0120 191014  
 +81 2666 8080

[support.japan@freescale.com](mailto:support.japan@freescale.com)

**Asia/Pacific:**

Freescale Semiconductor Hong Kong Ltd.  
 Technical Information Center  
 2 Dai King Street  
 Tai Po Industrial Estate,  
 Tai Po, N.T., Hong Kong  
 +800 2666 8080

[support.asia@freescale.com](mailto:support.asia@freescale.com)

**For Literature Requests Only:**

Freescale Semiconductor  
 Literature Distribution Center  
 P.O. Box 5405  
 Denver, Colorado 80217  
 (800) 441-2447  
 303-675-2140  
 Fax: 303-675-2150  
[LDCForFreescaleSemiconductor@hibbertgroup.com](mailto:LDCForFreescaleSemiconductor@hibbertgroup.com)

RoHS-compliant and/or Pb- free versions of Freescale products have the functionality and electrical characteristics of their non-RoHS-compliant and/or non-Pb- free counterparts. For further information, see <http://www.freescale.com> or contact your Freescale sales representative.

For information on Freescale's Environmental Products program, go to <http://www.freescale.com/epp>.

Information in this document is provided solely to enable system and software implementers to use Freescale Semiconductor products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document. Freescale Semiconductor reserves the right to make changes without further notice to any products herein. Freescale Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale Semiconductor assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters which may be provided in Freescale Semiconductor data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts. Freescale Semiconductor does not convey any license under its patent rights nor the rights of others. Freescale Semiconductor products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Freescale Semiconductor product could create a situation where personal injury or death may occur. Should Buyer purchase or use Freescale Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold Freescale Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Freescale Semiconductor was negligent regarding the design or manufacture of the part.

