



# **PowerPC 750CX Supplement to the PowerPC 750 RISC Microprocessor User Manual**

**Version 1.1**

**October 31, 2000**

**IBM Microelectronics Division**



## Notices

Before using this information and the product it supports, be sure to read the general information on the back cover of this document.

## Trademarks

The following are trademarks of International Business Machines Corporation in the United States, or other countries, or both:

IBM  
AIX

IBM Logo  
PowerPC 750

PowerPC  
RISCWatch

Other company, product, and service names may be trademarks or service marks of others.

This document is a preliminary edition of *PowerPC 750CX Supplement to the PowerPC 750 RISC Microprocessor User Manual*. Make sure you are using the correct edition for the level of the product.

This document contains information on a new product under development by IBM. IBM reserves the right to change or discontinue this product without notice.

© International Business Machines Corporation 2000.  
Portions hereof ©International Business Machines Corporation, 1991-2000.  
All rights reserved.



## Table of Contents

Differences between the PowerPC 750CX and PowerPC 750 PID 8p Microprocessors .....	1
Preface .....	1
Overview .....	1
Programming Model .....	1
Performance .....	2
Power, Voltage, Frequency .....	3
Package .....	3
Start-up Functions .....	5
Bus Pull-up Resistor Requirements .....	5
Summary of PowerPC 750CX Version Differences .....	6
1.0. PowerPC 750CX Overview .....	7
1.1. PowerPC 750CX Microprocessor Overview .....	7
1.2. PowerPC 750CX Microprocessor Features .....	10
1.2.1. Overview of PowerPC 750CX Microprocessor Features .....	10
1.2.2. Instruction Flow .....	12
1.2.2.1. Instruction Queue and Dispatch Unit .....	12
1.2.2.2. Branch Processing Unit (BPU) .....	13
1.2.2.3. Completion Unit .....	14
1.2.2.3.1. Independent Execution Units .....	14
1.2.2.3.2. Integer Units (IUs) .....	14
1.2.2.3.3. Floating-Point Unit (FPU) .....	14
1.2.2.3.4. Load/Store Unit (LSU) .....	15
1.2.2.3.5. System Register Unit (SRU) .....	15
1.2.3. Memory Management Units (MMUs) .....	15
1.2.4. L1 (Level 1) Instruction and Data Caches .....	16
1.2.5. L2 (Level 2) Cache Implementation .....	18
1.2.6. System Interface/Bus Interface Unit (BIU) .....	18
1.2.7. Signals .....	20
1.2.8. Signal Configuration .....	21
1.2.9. Clocking .....	22
1.3. PowerPC 750CX Microprocessor: Implementation .....	22
1.4. PowerPC Registers and Programming Model .....	24
1.5. Instruction Set .....	29
1.5.1. PowerPC Instruction Set .....	29
1.5.2. PowerPC 750CX Microprocessor Instruction Set .....	30
1.6. On-Chip Cache Implementation .....	30
1.6.1. PowerPC Cache Model .....	30
1.6.2. PowerPC 750CX Microprocessor Cache Implementation .....	31
1.7. Exception Model .....	31
1.7.1. PowerPC Exception Model .....	31
1.7.2. PowerPC 750CX Microprocessor Exception Implementation .....	32
1.8. Memory Management .....	34
1.8.1. PowerPC Memory Management Model .....	34
1.8.2. PowerPC 750CX Microprocessor Memory Management Implementation .....	34
1.9. Instruction Timing .....	35
1.10. Power Management .....	36
1.11. Thermal Management .....	37
1.12. Performance Monitor .....	38
Document History File .....	39



## Differences between the PowerPC 750CX and PowerPC 750 PID 8p Microprocessors

### Preface

The primary objective of this document is to define the functionality of the PowerPC 750CX RISC Microprocessor and serve as an interim document for use by software and hardware developers with reference to the existing "PowerPC 740 and PowerPC 750 RISC Microprocessor User's Manual." In addition this document will highlight the differences and new features associated with the PowerPC 750CX microprocessor.

### Overview

The PowerPC 750CX processor is derived from the PowerPC 750 (PID 8p) design. The 750 PID8p is a 32-bit implementation of the PowerPC architecture in a 0.22 micron CMOS technology with six levels of copper interconnect. The PowerPC 750CX is a technology remap of the 750 (using a 0.18 micron CMOS technology), with a reduced pin 60x bus interface. The PowerPC 750CX RISC Microprocessor, also referred to as the 750CX for the rest of this document, incorporates a 256KB L2 cache on-chip, eliminating the 750's backside L2 bus.

In addition to these major differences, the 750CX varies from the 750 in a number of other ways, which are defined in the following sections.

- Programming Model
- Performance
- Power, Voltage, Frequency
- Package
- Start-up Functions
- Bus Pull-up Resistor Requirements

### Programming Model

The processor version number for both the 750 and the 750CX is 0x0008. The processor revision level for the 750 (PID8p) is 0x8302 for the latest release. Refer to Table "Summary of Differences" on page 6, for the appropriate processor version revision level. While the 750 has an on-chip L2 cache controller, with a tag array that can support external L2 cache sizes of 256KB, 512KB, or 1MB; the 750CX has integrated a 256KB L2 cache array on-chip.

While the 750's off-chip cache is accessed over a backside bus that is generally run at or less than the speed of the processor, the 750CX's on-chip cache runs at the same speed as the processor. This results in a system performance for the 750CX that is comparable to a 750 configuration having a larger external cache. The internal L2 cache uses ECC to correct some, and detect the remaining, single-bit errors.

The effect of having an internal cache on the programming model is simply that some of the bits in the L2 Cache Register (L2CR) are no longer needed in the 750CX. The unused bits of the L2CR are those that are used to configure the external cache in the 750. The bit definitions for the L2CR for both the 750 and the 750CX are shown below.

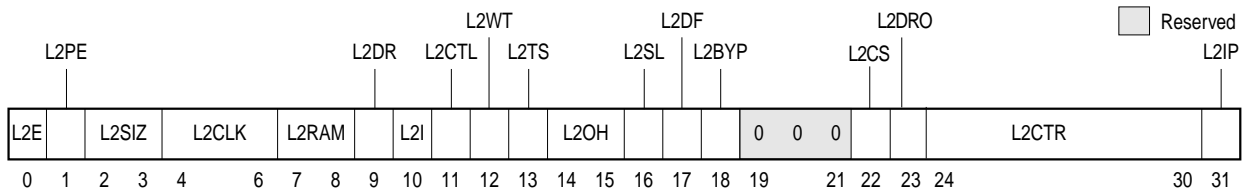


Figure 1. L2CR for the 750

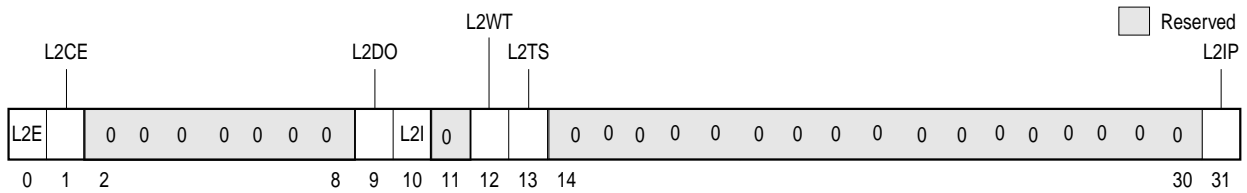


Figure 2. L2CR for the 750CX

### Performance

In addition to having an internal L2 cache, there are three other enhancements to the design that improve the performance of the PowerPC 750CX over the PowerPC 750.

- Additional FPU reservation station.
- Wider L1 data cache reload bus.
- Higher precision results from the reciprocal estimate instructions.

The floating-point execution unit in the PowerPC 750 has a single reservation station, which caused it to stall whenever the three stages of the FPU execution pipeline are full. A second reservation station has been added to the PowerPC 750CX FPU which eliminates this stall, resulting in higher (by 10% or more) throughput for optimized floating-point intensive applications.

The data bus width for the bus interface unit (BIU) accesses of the L1 data cache array is 64 bits on the PowerPC 750. To cast out or to reload a 256-bit cache line requires four access cycles. On the 750CX, this bus has been expanded to 256 bits. As a result, cache line data bursts can be read from or written to the cache array in a single cycle, reducing cache contention between the BIU and the load-store unit.

**Note:** For the first release of the 750CX (DD1.0), the data cache reload bus width is only 64 bits.

On the PowerPC 750, the floating reciprocal estimate single (**fres**) instruction provides an estimate of the reciprocal of its input value to a precision of 8 bits and the floating reciprocal square root estimate (**frsqрте**) instruction provides an estimate of the square root of its input value to a precision of 5 bits.

Both of these computations are improved on the PowerPC 750CX to yield results which contain 12 bits of precision. This reduces, or eliminates the need for iterative refinement of these values in applications which use these functions.

## Performance Comparison

	750	750CX
L1 data bus width	64 bits	256 bits
FPU reservation stations	1	2
fpres precision	8 bits	12 bits
frsqrte precision	5 bits	12 bits

## Power, Voltage, Frequency

Due to the technology remap, the PowerPC 750CX differs from the PowerPC 750 in its power, voltage, and frequency characteristics. These differences are summarized in the following table. Additionally, the dynamic power management (DPM) facility that is available on the PowerPC 750 is not available on the PowerPC 750CX for the initial revision level DD2.X.

## Power, Voltage, Frequency Differences

	PowerPC 750	PowerPC 750CX
Silicon Technology	0.22 micron	0.18 micron
Core Voltage	1.8V to 2.05V	1.71V to 1.89V
I/O Voltage	1.8V, 2.5V, or 3.3V	1.8V or 2.5V
Power disipation at 400MHz, 1.89V, 85°C	4.3W	4.2W (including L2 cache)
Core Speed Range	300MHz to 500MHz	350MHz to 500MHz
Maximum bus speed	100MHz	133MHz
DPM	Enabled	Disabled
<b>Note:</b> The PowerPC 750CX only supports up to 2.5V input and output voltages (I/O).		

## Package

Due to the internal L2 cache, the PowerPC 750CX does not have the PowerPC 750's backside L2 bus. In addition, the 750CX eliminates other pins, including several of the optional 60x bus protocol pins. Complete details regarding the package and pin assignments should be obtained from the "PowerPC 750CX Microprocessor Datasheet." The result is a reduced pin count package, as summarized in the table below.

## Pin Count Differences

	PowerPC 750	PowerPC 750CX
Package	360 pin DBGA	256 pin PBGA
Signal I/Os	258	139



The 119 pins removed from the PowerPC 750 to get the 750CX package are as follows.

- 98 signals are associated with the backside L2 bus.
- 12 signals are for parity on the 60x address (4 pins) and data (8 pins) busses.
- 4 signals are optional 60x bus protocol signals -  $\overline{ABB}$ ,  $\overline{DBB}$ ,  $\overline{DBDIS}$ ,  $\overline{DRTRY}$ .
- 4 signals for interrupt, debug and MP support -  $\overline{RSRV}$ ,  $\overline{SMI}$ ,  $\overline{TBEN}$ ,  $\overline{TLBISYNC}$
- CLKOUT is shared with  $\overline{CKSTP\_OUT}$  (see revision level note below)
- VOLTDET

The reduced pin count 60x bus on the 750CX eliminates optional functionality that has seldom been used. The  $\overline{ABB}$  signal is used to coordinate the handoff of address bus ownership in some systems, but is redundant with other signals ( $\overline{TS}$  and  $\overline{AACK}$ ) in the protocol. Similarly,  $\overline{DBB}$  can be used as an alternative way to coordinate data bus ownership. The  $\overline{DBDIS}$  signal, when asserted, causes the processor to suspend driving the data bus for some period of time.  $\overline{DRTRY}$  is used to pace the data beats on systems with slow memory access times.

**REVISION LEVEL NOTE ( $\overline{CLKOUT}/\overline{CKSTP\_OUT}$ ):** On the 750CX for DD1.0,  $\overline{CKSTP\_OUT}$  is removed and CLKOUT is present. Beginning with DD2.0,  $\overline{CKSTP\_OUT}$  and CLKOUT share a pin. Functionality for both is still available depending on the setting of the  $HID0[BCLK]$  and  $HID0[ECLK]$  register bits.

### HID0[BCLK] and HID0[ECLK] $\overline{CKSTP\_OUT}$ Configuration

HRESET	HID0[ECLK]	HID0[BCLK]	$\overline{CKSTP\_OUT}$
Asserted	x	x	Not Applicable
Negated	0	0	$\overline{CKSTP\_OUT}$
Negated	0	1	SYSCLK/ 2
Negated	1	0	Processor Core
Negated	1	1	SYSCLK

**REVISION LEVEL NOTE ( $\overline{DBWO}$ ):** On the 750CX for DD1.0 and DD2.0, the  $\overline{DBWO}$  signal is missing; this signal is present on the 750CX starting with DD2.1, but shared with L2\_TSTCLK.

Operation of this pin is dependent upon the LSSD\_MODE. When the LSSD\_MODE pin is low, the  $\overline{DBWO}/L2\_TSTCLK$  pin is set to the L2\_TSTCLK function which is used during the manufacturing process for testing. When the LSSD\_MODE pin is pulled to the high state, the  $\overline{DBWO}/L2\_TSTCLK$  pin is set to the  $\overline{DBWO}$  which is identical to those descriptions given in earlier versions of the PowerPC 750 RISC Microprocessor User Manuals.

Of the miscellaneous missing pins:

- CLKOUT is a test pin.
- $\overline{TLBISYNC}$  is used to synchronize multiple processors when the page table is modified.
- $\overline{TBEN}$  allows an external device to disable the internal time base register from incrementing.
- $\overline{SMI}$  can be used to signal a particular (system management) type of external interrupt.
- $\overline{RSRV}$  is intended to be used to signal an external L2 cache controller that a reservation is active in the processor.



## Start-up Functions

Several pins are used to configure certain processor modes during start-up. These pins are sampled at the negation of the  $\overline{\text{HRESET}}$  signal, to select the desired mode.

First, in the PowerPC 750, the  $\overline{\text{DRTRY}}$  signal must be active low at start-up to determine whether the processor will run in 'no-DRTRY' mode. In this mode, the bus interface unit can forward incoming data one bus cycle earlier to the load/store unit, knowing that the data will not be invalidated by a succeeding assertion of  $\overline{\text{DRTRY}}$ .

In the PowerPC 750CX, the  $\overline{\text{DRTRY}}$  pin is absent. This means that there is no way to post-invalidate received data, and so the PowerPC 750CX is always configured in 'no-DRTRY' mode.

Next, in the PowerPC 750, the  $\overline{\text{TLBISYNC}}$  signal is sampled at start-up to determine whether the processor will run in '32-bit data bus' mode. In this mode, only four bytes of data are transferred on the data bus at any one time, requiring a two-beat transaction to transfer a double word, and an eight-beat burst transaction to transfer a cache line. In the PowerPC 750CX, the  $\overline{\text{TLBISYNC}}$  pin is absent. In this case, the start-up function previously implemented by  $\overline{\text{TLBISYNC}}$  is performed by the  $\overline{\text{QACK}}$  signal.

These processor configuration differences are summarized in the following table.

## Start-up Differences

	PowerPC 750	PowerPC 750CX
noDRTRY mode	sample $\overline{\text{DRTRY}}$ to select	always selected
32-Bit Mode	sample $\overline{\text{TLBISYNC}}$ to select	sample $\overline{\text{QACK}}$ to select

## Bus Pull-up Resistor Requirements

On the PowerPC 750, the address bus, transfer attributes, and data bus may float in the high impedance state during inactive states while the bus is being held in the high impedance state by another device. This has the potential to cause excessive power draw in the bus receivers, both on the processor and on other devices attached to these signals. To insure that these signals do not float, pull-up resistors can be attached to these signals.

Starting with DD2.0, the PowerPC 750CX has added level protection to their I/O cell. Level protection is a function which avoids any floating I/O and meta-stability. The level protect "keeper" circuit locks the I/O to the closest voltage rail (that is, if the I/O voltage level is closer to OVDD, the circuit pulls the I/O level to OVDD; if the I/O level is closer to GND, the I/O level is pulled low). Approximately 100 $\mu$ a is required to overcome the keeper. (Refer to the PowerPC 750CX RISC Microprocessor Datasheet for more details.)

**REVISION LEVEL NOTE:** On the 750CX for DD1.0, the address bus, transfer attributes, and data bus signals float during periods of inactivity.

## Summary of PowerPC 750CX Version Differences

The revision level notes shown in this document are summarized below.

### Summary of Differences

	DD1.0	DD2.0	DD2.1	DD2.2	DD2.3e	DD2.4
PVR	0x00080100	0x00080100	0x00082201	0x00082202	0x00082213	0x00082204
L1 Data Bus Width	64-Bits	256-Bits	256-Bits	256-Bits	256-Bits	256-Bits
DPM	Enabled	Disabled	Disabled	Disabled	Enabled	Enabled
Test Pin	$\overline{\text{CHKSTP\_OUT}}$ removed	$\overline{\text{CLKOUT/CHKSTP\_OUT}}$ share a pin	$\overline{\text{CLKOUT/CHKSTP\_OUT}}$ share a pin	$\overline{\text{CLKOUT/CHKSTP\_OUT}}$ share a pin	$\overline{\text{CLKOUT/CHKSTP\_OUT}}$ share a pin	$\overline{\text{CLKOUT/CHKSTP\_OUT}}$ share a pin
32-Bit Mode	Not available	Available	Available	Available	Available	Available
Floating Bus	Pull-ups required	No pull-ups	No pull-ups	No pull-ups	No pull-ups	No pull-ups
$\overline{\text{DBWO}}$ Pin	Absent	Absent	Present	Present	Present	Present

## 1.0 PowerPC 750CX Overview

PowerPC 750CX is an implementation of the PowerPC architecture with enhancements to improve the floating point performance and the data transfer capability. This chapter provides an overview of the PowerPC 750CX microprocessor features, including a block diagram showing the major functional components. It also provides information about how PowerPC 750CX implementation complies with the PowerPC™ architecture definition.

### 1.1 PowerPC 750CX Microprocessor Overview

This section describes the features and general operation of PowerPC 750CX and provides a block diagram showing major functional units. The PowerPC 750CX is an implementation of the PowerPC microprocessor family of reduced instruction set computer (RISC) microprocessors with extensions to improve the floating point performance. The PowerPC 750CX 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 single and double-precision. The PowerPC 750CX is a superscalar processor that can complete two instructions simultaneously. It incorporates the following six execution units:

- Floating-point unit (FPU)
- Branch processing unit (BPU)
- System register unit (SRU)
- Load/store unit (LSU)
- Two integer units (IUs): IU1 executes all integer instructions. IU2 executes all integer instructions except multiply and divide instructions.

The ability to execute several instructions in parallel and the use of simple instructions with rapid execution times yield high efficiency and throughput for PowerPC 750CX-based systems. Most integer instructions execute in one clock cycle. The FPU is pipelined, it breaks the tasks it performs into subtasks, and then executes in three successive stages. Typically, a floating-point instruction can occupy only one of the three stages at a time, freeing the previous stage to work on the next floating-point instruction. Thus, three single-precision floating-point instructions can be in the FPU execute stage at a time. Double-precision add instructions have a three-cycle latency; double-precision multiply and multiply-add instructions have a four-cycle latency.

Figure 3 on page 10 shows the parallel organization of the execution units (shaded in the diagram). The instruction unit fetches, dispatches, and predicts branch instructions. Note that this is a conceptual model that shows basic features rather than attempting to show how features are implemented physically.

PowerPC 750CX has independent on-chip, 32-Kbyte, eight-way set-associative, physically addressed L1 caches for instructions and data and independent instruction and data memory management units (MMUs). Each MMU has a 128-entry, two-way set-associative translation lookaside buffer (DTLB and ITLB) that saves recently used page address translations. Block address translation is done through the four-entry instruction and data block address translation (IBAT and DBAT) arrays, defined by the PowerPC architecture. During block translation, effective addresses are compared simultaneously with all four BAT entries.

For information about the L1 cache, see Chapter 3, “Instruction and Data Cache Operation” in the PowerPC 740 and PowerPC 750 RISC Microprocessor Family User’s Manual.

The L2 cache is implemented with an on-chip, two-way set-associative tag memory, and an on-chip 256 Kbyte SRAM with ECC for data storage.



PowerPC 750CX has a 32-bit address bus and a 64-bit data bus. Multiple devices compete for system resources through a central external arbiter. PowerPC 750CX's three-state cache-coherency protocol (MEI) supports the modified, exclusive and invalid states, a compatible subset of the MESI (modified/exclusive/shared/invalid) four-state protocol, and it operates coherently in systems with four-state caches. PowerPC 750CX supports single-beat and burst data transfers for external memory accesses and memory-mapped I/O operations. The system interface is described in Chapter 7 and 8 in the PowerPC 740 and PowerPC 750 RISC Microprocessor Family User's Manual.

PowerPC 750CX has four software-controllable power-saving modes. Three static modes, doze, nap, and sleep, progressively reduce power dissipation. When functional units are idle, a dynamic power management mode causes those units to enter a low-power mode automatically without affecting operational performance, software execution, or external hardware. PowerPC 750CX also provides a thermal assist unit (TAU) and a way to reduce the instruction fetch rate for limiting power dissipation. Power management is described in Chapter 10, "Power and Thermal Management" in the PowerPC 740 and PowerPC 750 RISC Microprocessor Family User's Manual.

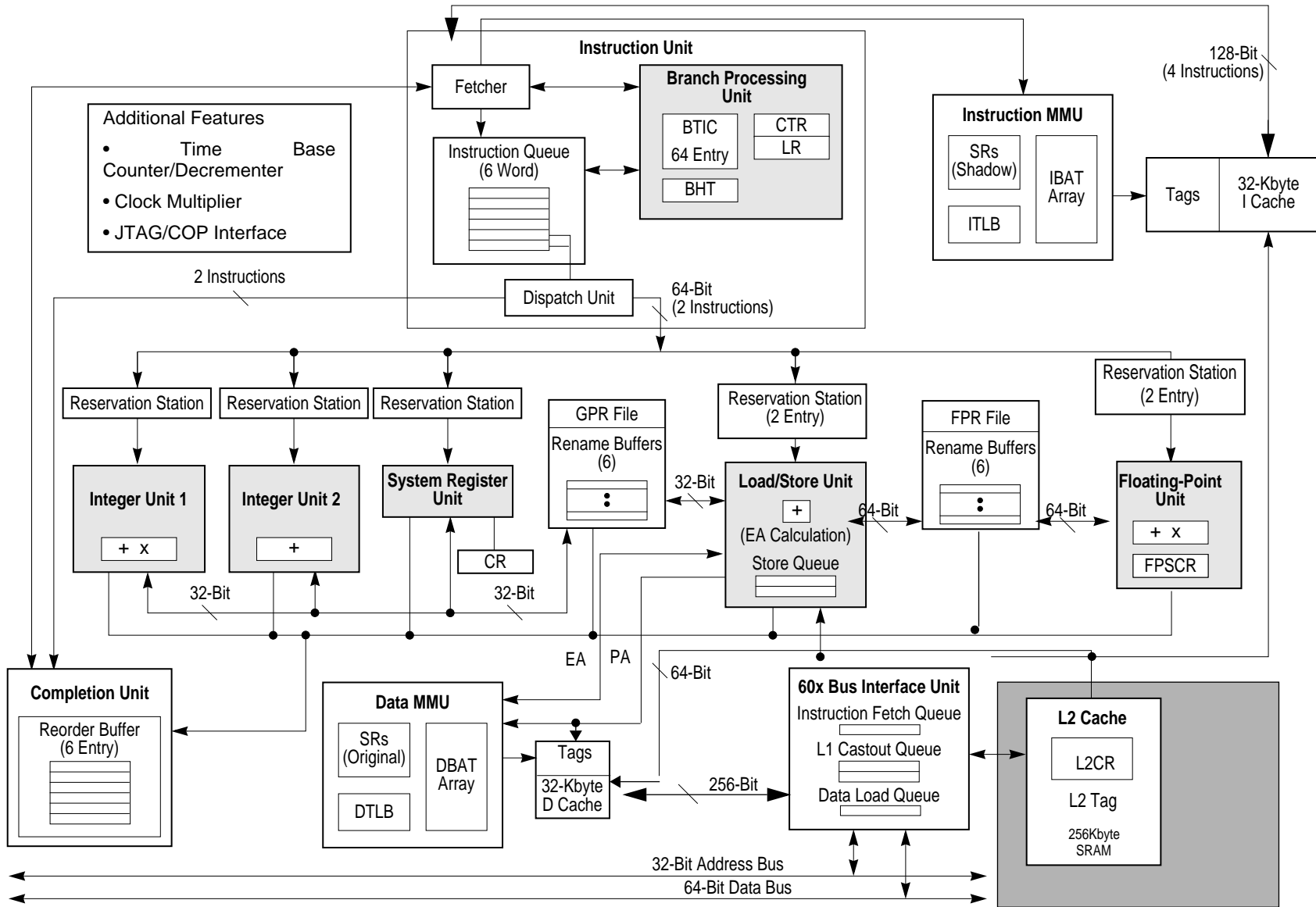


Figure 3. PowerPC 750CX Microprocessor Block Diagram



## 1.2 PowerPC 750CX Microprocessor Features

This section lists features of PowerPC 750CX. The interrelationship of these features is shown in Figure 3 on page 10.

### 1.2.1 Overview of PowerPC 750CX Microprocessor Features

Major features of PowerPC 750CX are as follows.

- High-performance, superscalar microprocessor.
  - As many as four instructions can be fetched from the instruction cache per clock cycle.
  - As many as three instructions can be dispatched per clock. (if one is a branch)
  - As many as six instructions can execute per clock (including two integer instructions).
  - Single-clock-cycle execution for most instructions.
- Six independent execution units and two register files.
  - BPU featuring both static and dynamic branch prediction.
    - 64-entry (16-set, four-way set-associative) branch target instruction cache (BTIC), a cache of branch instructions that have been encountered in branch/loop code sequences. If a target instruction is in the BTIC, it is fetched into the instruction queue a cycle sooner than it can be made available from the instruction cache. Typically, if a fetch access hits the BTIC, it provides the first two instructions in the target stream.
    - 512-entry branch history table (BHT) with two bits per entry for four levels of prediction—not-taken, strongly not-taken, taken, strongly taken.
    - Branch instructions that do not update the count register (CTR) or link register (LR) are removed from the instruction stream.
  - Two integer units (IUs) that share thirty-two GPRs for integer operands.
    - IU1 can execute any integer instruction.
    - IU2 can execute all integer instructions except multiply and divide instructions (multiply, divide, shift, rotate, arithmetic, and logical instructions). Most instructions that execute in the IU2 take one cycle to execute. The IU2 has a single-entry reservation station.
  - Three-stage FPU.
    - Fully IEEE 754-1985-compliant FPU for both single- and double-precision operations.
    - Supports non-IEEE mode for time-critical operations.
    - Hardware support for denormalized numbers.
    - Two-entry reservation station.
    - Thirty-two 64-bit FPRs for single or double-precision operands.
  - Two-stage LSU.
    - Two-entry reservation station.
    - Single-cycle, pipelined cache access.
    - Dedicated adder performs EA calculations.
    - Performs alignment and precision conversion for floating-point data.
    - Performs alignment and sign extension for integer data.
    - Three-entry store queue.
    - Supports both big- and little-endian modes.
    - Supports data type conversion with indexed scaling.
  - SRU handles miscellaneous instructions.
    - Executes CR logical and Move to/Move from SPR instructions (**mtspr** and **mfspir**).
    - Single-entry reservation station.

- Rename buffers.
  - Six GPR rename buffers.
  - Six FPR rename buffers.
  - Condition register buffering supports two CR writes per clock.
- Completion unit.
  - The completion unit retires an instruction from the six-entry reorder buffer (completion queue) when all instructions ahead of it have been completed, the instruction has finished execution, and no exceptions are pending.
  - Guarantees sequential programming model (precise exception model).
  - Monitors all dispatched instructions and retires them in order.
  - Tracks unresolved branches and flushes instructions from the mispredicted branch.
  - Retires as many as two instructions per clock.
- Separate on-chip L1 instruction and data caches (Harvard architecture).
  - 32-Kbyte, eight-way set-associative instruction and data caches.
  - Pseudo least-recently-used (PLRU) replacement algorithm.
  - 32-byte (eight-word) cache block.
  - Physically indexed/physical tags. (Note that the PowerPC architecture refers to physical address space as real address space.)
  - Cache write-back or write-through operation programmable on a per-page or per-block basis.
  - Instruction cache can provide four instructions per clock; data cache can provide two words per clock
  - Caches can be disabled in software.
  - Caches can be locked in software.
  - Data cache coherency (MEI) maintained in hardware.
  - The critical double word is made available to the requesting unit when it is burst into the line-fill buffer. The cache is nonblocking, so it can be accessed during this operation.
- On-chip 1:1 L2 cache.
  - 256 Kbyte on-chip ECC SRAMs.
  - On-chip 2-way set-associative tag memory.
- Separate memory management units (MMUs) for instructions and data.
  - 52-bit virtual address; 32-bit physical address.
  - Address translation for 4-Kbyte pages, variable-sized blocks, and 256-Mbyte segments.
  - Memory programmable as write-back/write-through, cacheable/noncacheable, and coherency enforced/coherency not enforced on a page or block basis.
  - Separate IBATs and DBATs (four each) also defined as SPRs.
  - Separate instruction and data translation lookaside buffers (TLBs).
    - Both TLBs are 128-entry, two-way set associative, and use LRU replacement algorithm.
    - TLBs are hardware-reloadable (that is, the page table search is performed in hardware).
- Bus interface features include the following.
  - Selectable bus-to-core clock frequency ratios of 2x, 2.5x, 3x, 3.5x, 4x, 4.5x .. 8x and 10x. (2x to 8x, all half-clock multipliers in-between).
  - A 64-bit, split-transaction external data bus with burst transfers.
  - Support for address pipelining and limited out-of-order bus transactions.
  - Single-entry load queue.
  - Single-entry instruction fetch queue.
  - Two-entry cache castout queue.
  - No- $\overline{DRTRY}$  mode eliminates the  $\overline{DRTRY}$  signal from the qualified bus grant. This allows the forwarding of data during load operations to the internal core one bus cycle sooner than if the use of  $\overline{DRTRY}$  is enabled.



- Multiprocessing support features include the following:
  - Hardware-enforced, three-state cache coherency protocol (MEI) for data cache.
  - Load/store with reservation instruction pair for atomic memory references, semaphores, and other multiprocessor operations
- Power and thermal management
  - Three static modes, doze, nap, and sleep, progressively reduce power dissipation:  
**Note:** Power management modes are disabled for current 750CX offering
    - Doze—All the functional units are disabled except for the time base/decrementer registers and the bus snooping logic.
    - Nap—The nap mode further reduces power consumption by disabling bus snooping, leaving only the time base register and the PLL in a powered state.
    - Sleep—All internal functional units are disabled, after which external system logic may disable the PLL and SYSCLK.
  - Thermal management facility provides software-controllable thermal management. Thermal management is performed through the use of three supervisor-level registers and an PowerPC 750CX-specific thermal management exception.
  - Instruction cache throttling provides control of instruction fetching to limit power consumption.
- Performance monitor can be used to help debug system designs and improve software efficiency.
- In-system testability and debugging features through JTAG boundary-scan capability.

## 1.2.2 Instruction Flow

As shown in Figure 3 on page 10, the PowerPC 750CX instruction unit provides centralized control of instruction flow to the execution units. The instruction unit contains a sequential fetcher, six-entry instruction queue (IQ), dispatch unit, and BPU. It determines the address of the next instruction to be fetched based on information from the sequential fetcher and from the BPU.

See Chapter 6, “Instruction Timing” in the PowerPC 740 and PowerPC 750 RISC Microprocessor Family User’s Manual for more information.

The sequential fetcher loads instructions from the instruction cache into the instruction queue. The BPU extracts branch instructions from the sequential fetcher. Branch instructions that cannot be resolved immediately are predicted using either PowerPC 750CX-specific dynamic branch prediction or the architecture-defined static branch prediction.

Branch instructions that do not affect the LR or CTR are removed from the instruction stream. The BPU folds branch instructions when a branch is taken (or predicted as taken); branch instructions that are not taken, or predicted as not taken, are removed from the instruction stream through the dispatch mechanism.

Instructions issued beyond a predicted branch do not complete execution until the branch is resolved, preserving the programming model of sequential execution. If branch prediction is incorrect, the instruction unit flushes all predicted path instructions, and instructions are fetched from the correct path.

### 1.2.2.1 Instruction Queue and Dispatch Unit

The instruction queue (IQ), shown in Figure 3 on page 10, holds as many as six instructions and loads up to four instructions from the instruction cache during a single processor clock cycle. The instruction fetcher continuously attempts to load as many instructions as there were vacancies in the IQ in the previous clock cycle. All instructions except branch instructions are dispatched to their respective execution units from the bottom two positions in the instruction queue (IQ0 and IQ1) at a maximum rate of two instructions per cycle. Reservation stations are provided for the IU1, IU2, FPU, LSU, and SRU. The dispatch unit checks for source and



destination register dependencies, determines whether a position is available in the completion queue, and inhibits subsequent instruction dispatching as required.

Branch instructions can be detected, decoded, and predicted from anywhere in the instruction queue. For a more detailed discussion of instruction dispatch, see Section 6.6.1 in the PowerPC 740 and PowerPC 750 RISC Microprocessor Family User's Manual.

### 1.2.2.2 Branch Processing Unit (BPU)

The BPU receives branch instructions from the sequential fetcher and performs CR lookahead operations on conditional branches to resolve them early, achieving the effect of a zero-cycle branch in many cases.

Unconditional branch instructions and conditional branch instructions in which the condition is known can be resolved immediately. For unresolved conditional branch instructions, the branch path is predicted using either the architecture-defined static branch prediction or PowerPC 750CX-specific dynamic branch prediction. Dynamic branch prediction is enabled if  $HID0[BHT] = 1$ .

When a prediction is made, instruction fetching, dispatching, and execution continue from the predicted path, but instructions can not write back results to architected registers until the prediction is determined to be correct (resolved).

When a prediction is incorrect, the instructions from the incorrect path are flushed from the processor and processing begins from the correct path.

PowerPC 750CX allows a second branch instruction to be predicted; instructions from the second predicted instruction stream can be fetched but cannot be dispatched.

Dynamic prediction is implemented using a 512-entry branch history table (BHT), a cache that provides two bits per entry that together indicate four levels of prediction for a branch instruction—not-taken, strongly not-taken, taken, strongly taken. When dynamic branch prediction is disabled, the BPU uses a bit in the instruction encoding to predict the direction of the conditional branch. Therefore, when an unresolved conditional branch instruction is encountered, PowerPC 750CX executes instructions from the predicted target stream although the results are not committed to architected registers until the conditional branch is resolved. This execution can continue until a second unresolved branch instruction is encountered.

When a branch is taken (or predicted as taken), the instructions from the untaken path must be flushed and the target instruction stream must be fetched into the IQ. The BTIC is a 64-entry cache that contains the most recently used branch target instructions, typically in pairs. When an instruction fetch hits in the BTIC, the instructions arrive in the instruction queue in the next clock cycle, a clock cycle sooner than they would arrive from the instruction cache. Additional instructions arrive from the instruction cache in the next clock cycle. The BTIC reduces the number of missed opportunities to dispatch instructions and gives the processor a one-cycle head start on processing the target stream.

The BPU contains an adder to compute branch target addresses and three user-control registers—the link register (LR), the count register (CTR), and the CR. The BPU calculates the return pointer for subroutine calls and saves it into the LR for certain types of branch instructions. The LR also contains the branch target address for the Branch Conditional to Link Register (**bcldr**) instruction. The CTR contains the branch target address for the Branch Conditional to Count Register (**bcctr**) instruction. Because the LR and CTR are SPRs, their contents can be copied to or from any GPR. Because the BPU uses dedicated registers rather than GPRs or FPRs, execution of branch instructions is largely independent from execution of integer and floating-point instructions.



### 1.2.2.3 Completion Unit

The completion unit operates closely with the dispatch and execution unit. Instructions are fetched and dispatched in program order. At the point of dispatch, the program order is maintained by assigning each dispatched instruction a successive entry in the six-entry completion queue. The completion unit tracks instructions from dispatch through execution and retires them in program order from the two bottom entries in the completion queue (CQ0 and CQ1).

Instructions cannot be dispatched to an execution unit unless there is a vacancy in the completion queue and there is a rename register available to be assigned to the instruction. Branch instructions that do not update the CTR or LR are removed from the instruction stream and do not take an entry in the completion queue. Instructions that update the CTR and LR follow the same dispatch and completion procedures as non-branch instructions, except that they are not issued to an execution unit.

An instruction is retired when it is removed from the completion queue and its results are written to architected registers from their rename registers (GPRs, FPRs, LR, and CTR). In-order retirement ensures the correct architectural state when PowerPC 750CX must recover from a mispredicted branch or any exception. Retiring an instruction removes it from the completion queue and returns the rename register assigned to it by the dispatch unit to the available queue to be reused again by another instruction being dispatched.

For a more detailed discussion of instruction completion, see Section 6.6.1 on Page 6-29, in the PowerPC 740 and PowerPC 750 RISC Microprocessor Family User's Manual.

#### 1.2.2.3.1 Independent Execution Units

In addition to the BPU, PowerPC 750CX has the following five execution units.

- Two Integer Units (IUs)
- Floating-Point Unit (FPU)
- Load/Store Unit (LSU)
- System Register Unit (SRU)

Each is described in the following sections.

#### 1.2.2.3.2 Integer Units (IUs)

The integer units IU1 and IU2 are shown in Figure 3 on page 10. The IU1 can execute any integer instruction; the IU2 can execute any integer instruction except multiplication and division instructions. Each IU has a single-entry reservation station that can receive instructions from the dispatch unit and operands from the GPRs or the rename buffers.

Each IU consists of three single-cycle subunits—a fast adder/comparator, a subunit for logical operations, and a subunit for performing rotates, shifts, and count-leading-zero operations. These subunits handle all one-cycle arithmetic instructions; only one subunit can execute an instruction at a time.

The IU1 has a 32-bit integer multiplier/divider as well as the adder, shift, and logical units of the IU2. The multiplier supports early exit for operations that do not require full 32- x 32-bit multiplication.

Each IU has a dedicated result bus (not shown in Figure 3 on page 10) that connects to rename buffers.

#### 1.2.2.3.3 Floating-Point Unit (FPU)

The FPU, shown in Figure 3 on page 10, is designed as a three stage pipelined processing unit, where the first stage is for multiply, the second stage is for add and the third stage is for normalize. A single-precision multiply-add operation requires one cycle through put and three cycles latency. (an sp instruction spends one cycle in each stage of the FPU). A double-precision multiply requires two cycles in the multiply stage and one cycle for add, and another cycle for normalize. A double-precision multiply-add has a two cycle through put and a four cycle latency. As instructions are dispatched to the FPU's reservation station, source operand data can be accessed from the FPRs or from the FPR rename buffers. Results in turn are written to the rename buffers and are made available to subsequent instructions. Instructions pass through the reservation station in dispatch order. Thirty-two 64-bit floating-point registers are provided to support floating-point operations. Stalls due to contention for FPRs are minimized by automatic allocation of the six floating-point rename registers. PowerPC 750CX writes the contents of the rename registers to the appropriate FPR when floating-point instructions are retired by the completion unit.

PowerPC 750CX supports all IEEE 754 floating-point data types (normalized, denormalized, NaN, zero, and infinity) in hardware, eliminating the latency incurred by software exception routines. (Note that "exception" is also referred to as "interrupt" in the architecture specification.)

#### 1.2.2.3.4 Load/Store Unit (LSU)

The LSU executes all load and store instructions and provides the data transfer interface between the GPRs, FPRs, and the cache/memory subsystem. The LSU functions as a two stage pipe-lined unit where the effective address is translated on the first cycle and the L1 cache is accessed on the second cycle. If the instruction hits in the L1 cache the data is available at the end of the second cycle. Additionally, the LSU performs data alignment, and provides sequencing for load/store string and multiple register instructions.

Load and store instructions are issued in program order; however, some memory accesses can occur out of order. Synchronizing instructions can be used to enforce strict ordering if necessary. When there are no data dependencies and the guarded bit for the page or block is cleared, a maximum of one out-of-order cacheable load operation can execute per cycle, with a two-cycle total latency on a L1 cache hit. Data returned from the cache is held in a rename register for subsequent instructions and until the completion logic commits the value to a GPR or FPR. Stores cannot be executed out of order and are held in the store queue until the completion logic signals that the store operation is to be completed to memory. PowerPC 750CX executes store instructions with a maximum throughput of one per cycle and a three-cycle total latency to the L1 data cache. The time required to perform the actual load or store operation depends on the processor/bus clock ratio and whether the operation involves the L1 cache, the L2 cache, system memory, or an I/O device.

#### 1.2.2.3.5 System Register Unit (SRU)

The SRU executes various system-level instructions, as well as condition register logical operations and move to/from special-purpose register instructions. To maintain system state, most instructions executed by the SRU are execution-serialized; that is, the instruction is held for execution in the SRU until all previously issued instructions have executed. Results from execution-serialized instructions executed by the SRU are not available or forwarded for subsequent instructions until the instruction completes.

### 1.2.3 Memory Management Units (MMUs)

PowerPC 750CX's MMUs support up to 4 Petabytes ( $2^{52}$ ) of virtual memory and 4 Gigabytes ( $2^{32}$ ) of physical memory for instructions and data. The MMUs also control access privileges for these spaces on block and page granularities. Referenced and changed status is maintained by the processor for each page to support demand-paged virtual memory systems.



The LSU with the aid of the MMU translates effective addresses for data loads and stores; the LSU first calculates effective addresses and the MMU translates it to determine the correct physical address for the memory access and provides the necessary control and protection information to complete the access.

PowerPC 750CX supports the following types of memory translation.

- Real addressing mode—In this mode, translation is disabled by clearing bits in the machine state register (MSR): MSR[IR] for instruction fetching or MSR[DR] for data accesses. When address translation is disabled, the physical address is identical to the effective address.
- Page address translation—translates the effective address by using virtual to physical method.
- Block address translation—translates the effective address by using BATs (128 Kbytes to 256 Mbytes).

If translation is enabled, the appropriate MMU translates the higher-order bits of the effective address into physical address bits. The lower-order address bits (that are untranslated and therefore, considered both logical and physical) are directed to the L1 caches where they form the index into the eight-way set-associative tag array. After translating the address, the MMU passes the higher-order physical address bits to the cache and the cache lookup completes. For caching-inhibited accesses or accesses that miss in the cache, the untranslated lower-order address bits are concatenated with the translated higher-order address bits; the resulting 32-bit physical address is used by the memory unit and the system interface, which accesses external memory.

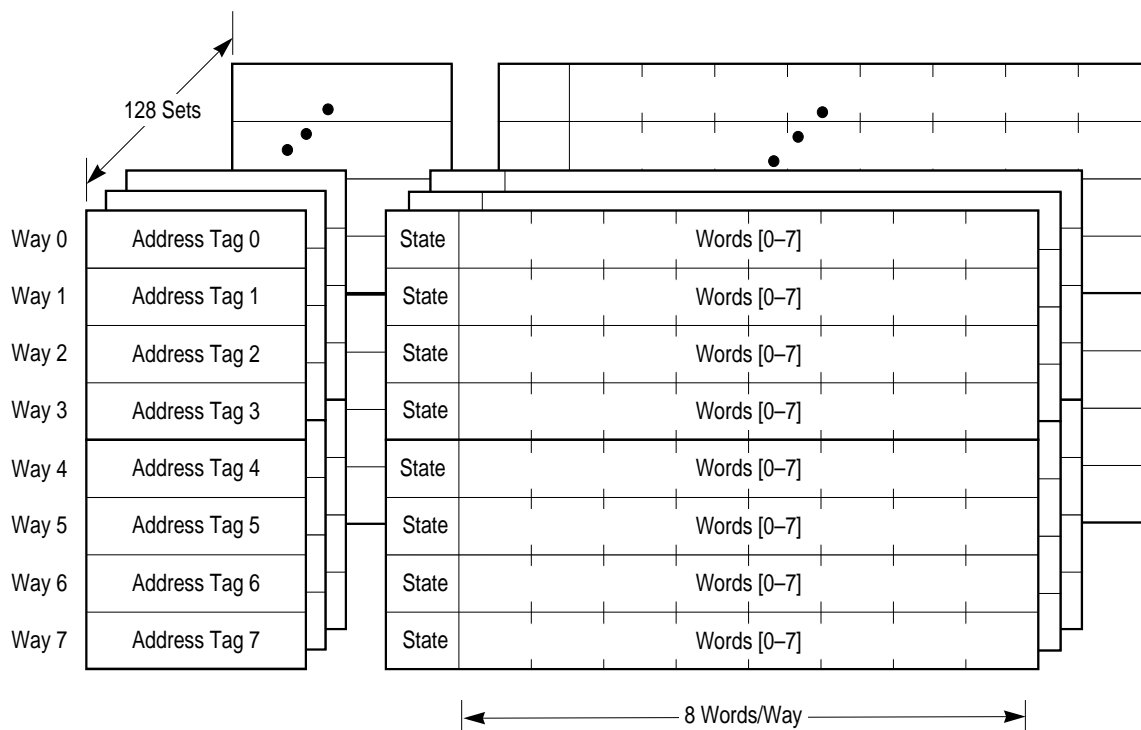
The TLBs store page address translations for recent memory accesses. For each access, an effective address is presented for page and block translation simultaneously. If a translation is found in both the TLB and the BAT array, the block address translation in the BAT array is used. Usually the translation is in a TLB and the physical address is readily available to the L1 cache. When a page address translation is not in a TLB, hardware searches for one in the page table following the model defined by the PowerPC architecture.

Instruction and data TLBs provide address translation in parallel with the L1 cache access, incurring no additional time penalty in the event of a TLB hit. PowerPC 750CX's TLBs are 128-entry, two-way set-associative caches that contain instruction and data address translations. PowerPC 750CX automatically generates a TLB search on a TLB miss.

### 1.2.4 L1 (Level 1) Instruction and Data Caches

PowerPC 750CX implements separate instruction and data caches. Each cache is 32-Kbyte and eight-way set associative. As defined by the PowerPC architecture, they are physically indexed. Each cache block contains eight contiguous words from memory that are loaded from an 8-word boundary (that is, bits EA[27–31] are zeros); thus, a cache block never crosses a page boundary. An entire cache block can be updated by a four-beat burst load. Misaligned accesses across a page boundary can incur a performance penalty. Caches are nonblocking, write-back caches with hardware support for reloading on cache misses. The critical double word is transferred on the first beat and is simultaneously written to the cache and forwarded to the requesting unit, minimizing stalls due to load delays. The cache being loaded is not blocked to internal accesses while the load completes.

PowerPC 750CX L1 cache organization is shown in Figure 4 on page 17.



**Figure 4. Cache Organization**

Within one cycle, the data cache provides double-word access to the LSU. Like the instruction cache, the data cache can be invalidated all at once or on a per-cache-block basis. The data cache can be disabled and invalidated by clearing HID0[DCE] and setting HID0[DCFI]. The data cache can be locked by setting HID0[DLOCK]. To ensure cache coherency, the data cache supports the three-state MEI protocol. The data cache tags are single-ported, so a simultaneous load or store and a snoop access represent a resource collision. If a snoop hit occurs, the LSU is blocked internally for one cycle to allow the eight-word block of data to be copied to the write-back buffer.

The data bus width for bus interface unit (BIU) accesses of the L1 data cache array is 64 bits on the 750. To cast out or to reload a 256 bit cache line requires four access cycles. On the 750CX, this bus has been expanded to 256 bits. As a result, cache line data bursts can be read from or written to the cache array in a single cycle, reducing cache contention between the BIU and the load-store unit.

Within one cycle, the instruction cache provides up to four instructions to the instruction queue. The instruction cache can be invalidated entirely or on a cache-block basis. The instruction cache can be disabled and invalidated by clearing HID0[ICE] and setting HID0[ICFI]. The instruction cache can be locked by setting HID0[ILOCK]. The instruction cache supports only the valid/invalid states.

PowerPC 750CX also implements a 64-entry (16-set, four-way set-associative) branch target instruction cache (BTIC). The BTIC is a cache of branch instructions that have been encountered in branch/loop code sequences. If the target instruction is in the BTIC, it is fetched into the instruction queue a cycle sooner than it can be made available from the instruction cache. The BTIC contains the first two instructions at the target of the branch. The BTIC can be disabled and invalidated through software.

For more information and timing examples showing cache hit and cache miss latencies, see Section 6.3.2, "Instruction Fetch Timings" in the PowerPC 740 and PowerPC 750 RISC Microprocessor Family User's Manual.



### 1.2.5 L2 (Level 2) Cache Implementation

The L2 cache is a unified cache that receives memory requests from both the L1 instruction and data caches independently. The L2 cache is implemented with an on-chip, two-way, set-associative tag memory, and with a 256-Kbyte, on-chip SRAM for data storage. The L2 cache normally operates in write-back mode and supports system cache coherency through snooping.

The L2 cache is organized into 64-byte lines, which in turn are subdivided into 32-byte sectors (blocks), the unit at which cache coherency is maintained.

The L2 cache controller contains the L2 cache control register (L2CR) and the L2 cache tag array. The L2CR register includes bits to manage the L2 cache. The cache is two-way set-associative with 2K tags per way. Each sector (32-byte cache block) has its own valid and modified status bits.

Requests from the L1 cache generally result from instruction misses, data load or store misses, write-through operations, or cache management instructions. Misses from the L1 caches are looked up in the L2 tags and serviced by the L2 cache if they hit; they are forwarded to the bus interface if they miss.

The L2 cache can accept multiple, simultaneous accesses but only processes one request per cycle. The L1 instruction cache can request an instruction at the same time that the L1 data cache is requesting one load and two store operations. The L2 cache also services snoop requests from the bus. If there are multiple pending requests to the L2 cache, snoop requests have highest priority. The next priority consists of load and store requests from the L1 data cache. The next priority consists of instruction fetch requests from the L1 instruction cache.

For more information, see Chapter 9, "L2 Cache Interface Operation" in the PowerPC 740 and PowerPC 750 User Manual.

### 1.2.6 System Interface/Bus Interface Unit (BIU)

As described in the preface section, the 750CX uses a reduced system signal set, which eliminates some 60X Bus optional protocol pins. The system designer needs to make note of these differences.

The address and data buses operate independently; address and data tenures of a memory access are decoupled to provide a more flexible control of memory traffic. The primary activity of the system interface is transferring data and instructions between the processor and system memory. There are two types of memory accesses:

- Single-beat transfers—These memory accesses allow transfer sizes of 8, 16, 24, 32, or 64 bits in one bus clock cycle. Single-beat transactions are caused by uncacheable read and write operations that access memory directly (that is, when caching is disabled), cache-inhibited accesses, and stores in write-through mode.
- Four-beat burst (32 byte) instruction/data transfers—Burst transactions, which always transfer an entire cache block (32 bytes), are initiated when an entire cache block is transferred. Because the first-level caches on PowerPC 750CX are write-back caches, burst-read memory, burst operations are the most common memory accesses, followed by burst-write memory operations, and single-beat (noncacheable or write-through) memory read and write operations.

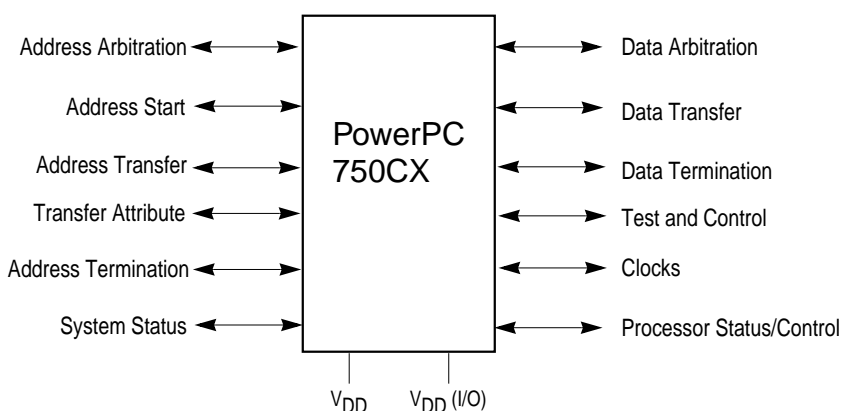
PowerPC 750CX also supports address-only operations, variants of the burst and single-beat operations, (for example, atomic memory operations and global memory operations that are snooped), and address retry activity (for example, when a snooped read access hits a modified block in the cache). The broadcast of some address-only operations is controlled through HID0[ABE]. I/O accesses use the same protocol as memory accesses.

Access to the system interface is granted through an external arbitration mechanism that allows devices to compete for bus mastership. This arbitration mechanism is flexible, allowing PowerPC 750CX to be integrated into systems that implement various fairness and bus parking procedures to avoid arbitration overhead.

Typically, memory accesses are weakly ordered—sequences of operations, including load/store string and multiple instructions, do not necessarily complete in the order they begin—maximizing the efficiency of the bus without sacrificing data coherency. PowerPC 750CX allows read operations to go ahead of store operations (except when a dependency exists, or in cases where a noncacheable access is performed), and provides support for a write operation to go ahead of a previously queued read data tenure (for example, letting a snoop push be enveloped between address and data tenures of a read operation). Because PowerPC 750CX can dynamically optimize run-time ordering of load/store traffic, overall performance is improved.

The system interface is specific for each PowerPC microprocessor implementation.

PowerPC 750CX signals are grouped as shown in Figure 5. Test and control signals provide diagnostics for selected internal circuits.



**Figure 5. System Interface**

The system interface supports address pipelining, which allows the address tenure of one transaction to overlap the data tenure of another. The extent of the pipelining depends on external arbitration and control circuitry. Similarly, PowerPC 750CX supports split-bus transactions for systems with multiple potential bus masters—one device can have mastership of the address bus while another has mastership of the data bus. Allowing multiple bus transactions to occur simultaneously increases the available bus bandwidth for other activity.

PowerPC 750CX's clocking structure supports a wide range of processor-to-bus clock ratios.



## 1.2.7 Signals

PowerPC 750CX's signals are grouped as follows.

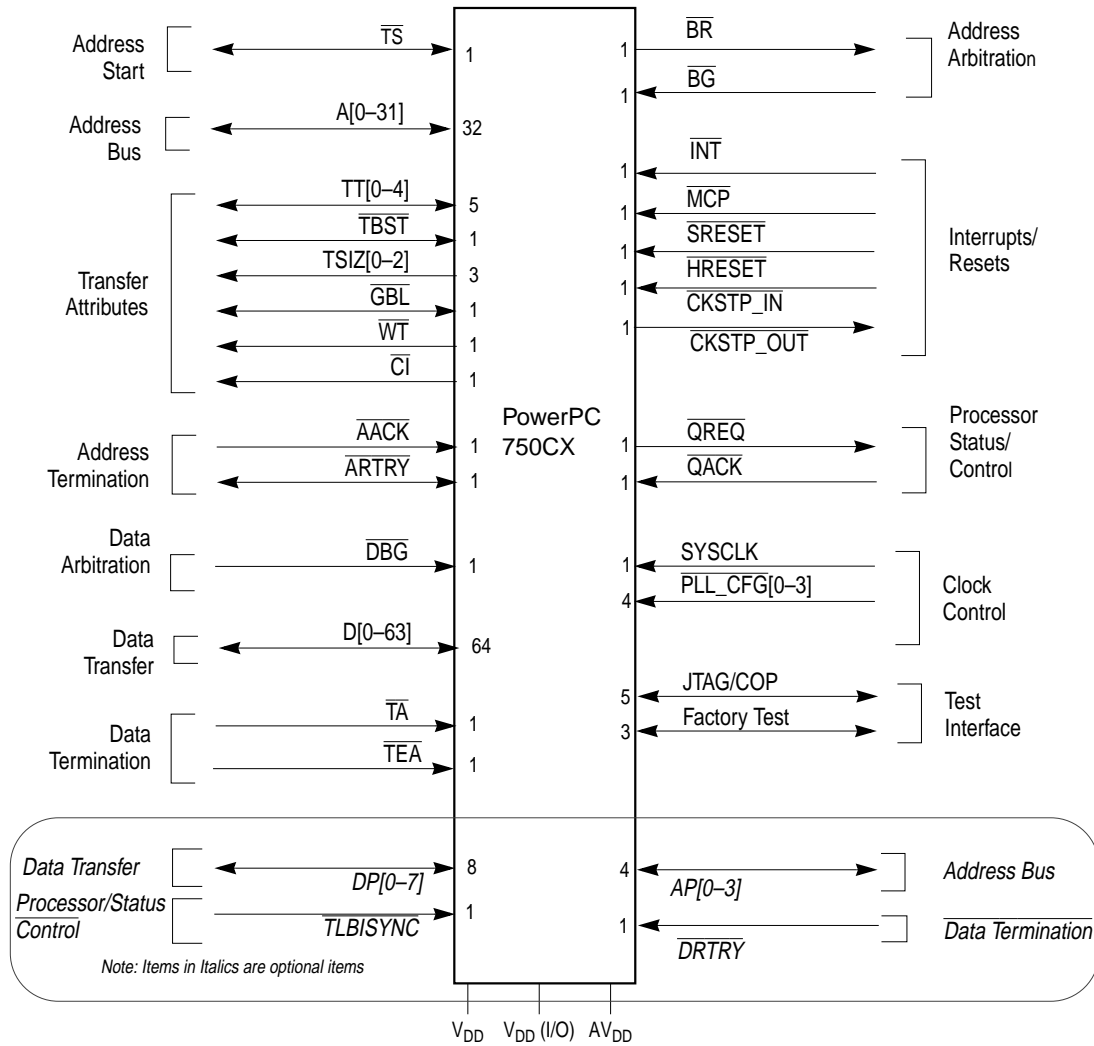
- Address arbitration signals—PowerPC 750CX uses these signals to arbitrate for address bus mastership.
- Address start signals—These signals indicate that a bus master has begun a transaction on the address bus.
- Address transfer signals—These signals include the address bus signals. They are used to transfer the address. There is no address parity.
- Transfer attribute signals—These signals provide information about the type of transfer, such as the transfer size and whether the transaction is bursted, write-through, or caching-inhibited.
- Address termination signals—These signals are used to acknowledge the end of the address phase of the transaction. They also indicate whether a condition exists that requires the address phase to be repeated.
- Data arbitration signals—PowerPC 750CX uses these signals to arbitrate for data bus mastership.
- Data transfer signals—There are 64 bit-lanes to transfer 8 bytes of data. There is no data parity.
- Data termination signals—Data termination signals are required after each data beat in a data transfer. In a single-beat transaction, a data termination signal also indicates the end of the tenure; in burst accesses, data termination signals apply to individual beats and indicate the end of the tenure only after the final data beat. They also indicate whether a condition exists that requires the data phase to be repeated.
- Interrupt signals—These signals include the interrupt signal, checkstop signals, and both soft reset and hard reset signals. These signals are used to generate interrupt exceptions and, under various conditions, to reset the processor.
- Processor status/control signals—These signals are used to set the reservation coherency bit, enable the time base, and other functions.
- Miscellaneous signals—These signals are used in conjunction with such resources as secondary caches and the time base facility.
- JTAG/COP interface signals—The common on-chip processor (COP) unit provides a serial interface to the system for performing board-level boundary scan interconnect tests.
- Clock signals—These signals determine the system clock frequency. These signals can also be used to synchronize multiprocessor systems.

**Note:** A bar over a signal name indicates that the signal is active low—for example,  $\overline{ARTRY}$  (address retry) and  $\overline{TS}$  (transfer start). Active-low signals are referred to as asserted (active) when they are low and negated when they are high. Signals that are not active low, such as TT[0–4] (transfer type signals) are referred to as asserted when they are high and negated when they are low.



### 1.2.8 Signal Configuration

Figure 6. shows PowerPC 750CX's logical pin configuration. The signals are grouped by function.



**Figure 6. PowerPC 750CX Microprocessor Signal Groups**

Signal functionality is described in detail in Chapter 7 “Signal Descriptions” and Chapter 8, “Bus Interface Operations” in the PowerPC 740 and PowerPC 750 RISC Microprocessor Family User’s Manual.

**Note:** The 750CX has a reduced set of signals from the PowerPC 750 PID8 series of microprocessors. The PowerPC 750CX Datasheet, should be referenced for the complete signal pins present on the PowerPC 750CX.



## 1.2.9 Clocking

PowerPC 750CX requires a single system clock input, SYSCLK, that represents the bus interface frequency. Internally, the processor uses a phase-locked loop (PLL) circuit to generate a master core clock that is frequency-multiplied and phase-locked to the SYSCLK input. This core frequency is used to operate the internal circuitry.

The PLL is configured by the PLL\_CFG[0–3] signals, which select the multiplier that the PLL uses to multiply the SYSCLK frequency up to the internal core frequency. The feedback in the PLL guarantees that the processor clock is phase locked to the bus clock, regardless of process variations, temperature changes, or parasitic capacitances.

The PLL also ensures a 50% duty cycle for the processor clock.

PowerPC 750CX supports various processor-to-bus clock frequency ratios, although not all ratios are available for all frequencies. Configuration of the processor/bus clock ratios is displayed through a PowerPC 750CX-specific register, HID1. For information about supported clock frequencies, see the PowerPC 750CX Datasheet (hardware specifications).

## 1.3 PowerPC 750CX Microprocessor: Implementation

The PowerPC architecture is derived from the POWER architecture (Performance Optimized With Enhanced RISC architecture). The PowerPC architecture shares the benefits of the POWER architecture optimized for single-chip implementations. The PowerPC architecture design facilitates parallel instruction execution and is scalable to take advantage of future technological gains.

This section describes the PowerPC architecture in general, and specific details about the implementation of PowerPC 750CX as a low-power, 32-bit member of the PowerPC processor family. The structure of this section follows the organization of the user's manual; each subsection provides an overview of each chapter.

- Registers and programming model—Section 1.4, "PowerPC Registers and Programming Model," on Page 24 describes the registers for the operating environment architecture common among PowerPC processors and describes the programming model. It also describes the registers that are unique to PowerPC 750CX. The information in this section is described more fully in Chapter 2, "Programming Model" in the PowerPC 740 and PowerPC 750 User Manual.
- Instruction set and addressing modes—Section 1.5, "Instruction Set," on Page 29 describes the PowerPC instruction set and addressing modes for the PowerPC operating environment architecture, defines the PowerPC instructions implemented in PowerPC 750CX, and describes new instruction set extensions to improve the performance of single-precision floating-point operations and the capability of data transfer. The information in this section is described more fully in Section 1.0, "PowerPC 750CX Overview," on Page 7.
- Cache implementation—Section 1.6, "On-Chip Cache Implementation," on Page 30 describes the cache model that is defined generally for PowerPC processors by the virtual environment architecture. It also provides specific details about PowerPC 750CX cache implementation. The information in this section is described more fully in Section 1.0, "PowerPC 750CX Overview," on Page 7 and Chapter 9, "L2 Cache Interface Operation" in the PowerPC 740 and PowerPC 750 User Manual.
- Exception model—Section 1.7, "Exception Model," on Page 31 describes the exception model of the PowerPC operating environment architecture and the differences in PowerPC 750CX exception model. The information in this section is described more fully in Chapter 4, "Exceptions" in the PowerPC 740 and PowerPC 750 User Manual.

- Memory management—Section 1.8, "Memory Management," on Page 32 describes generally the conventions for memory management among the PowerPC processors. This section also describes PowerPC 750CX's implementation of the 32-bit PowerPC memory management specification. The information in this section is described more fully in Chapter 5, "Memory Management" in the PowerPC 740 and PowerPC 750 User Manual.
- Instruction timing—Section 1.9, "Instruction Timing," on Page 35 provides a general description of the instruction timing provided by the superscalar, parallel execution supported by the PowerPC architecture and PowerPC 750CX. The information in this section is described more fully in Chapter 6, "Instruction Timing" in the PowerPC 740 and PowerPC 750 User Manual.
- Power management—Section 1.10, "Power Management," on Page 36 describes how the power management can be used to reduce power consumption when the processor, or portions of it, are idle. The information in this section is described more fully in Chapter 10, "Power and Thermal Management" in the PowerPC 740 and PowerPC 750 User Manual.
- Thermal management—Section 1.11, "Thermal Management," on Page 37 describes how the thermal management unit and its associated registers (THRM1–THRM3) and exception can be used to manage system activity in a way that prevents exceeding system and junction temperature thresholds. This is particularly useful in high-performance portable systems, which cannot use the same cooling mechanisms (such as fans) that control overheating in desktop systems. The information in this section is described more fully in Chapter 10, "Power and Thermal Management" in the PowerPC 740 and PowerPC 750 User Manual.
- Performance monitor—Section 1.12 on Page 38 describes the performance monitor facility, which system designers can use to help bring up, debug, and optimize software performance. The information in this section is described more fully in Chapter 11, "Performance Monitor" in the PowerPC 740 and PowerPC 750 User Manual.

The following sections summarize the features of PowerPC 750CX, distinguishing those that are defined by the architecture from those that are unique to PowerPC 750CX implementation.

The PowerPC architecture consists of the following layers, and adherence to the PowerPC architecture can be described in terms of which of the following levels of the architecture is implemented.

- PowerPC user instruction set architecture (UISA)—Defines the base user-level instruction set, user-level registers, data types, floating-point exception model, memory models for a uniprocessor environment, and programming model for a uniprocessor environment.
- PowerPC virtual environment architecture (VEA)—Describes the memory model for a multiprocessor environment, defines cache control instructions, and describes other aspects of virtual environments. Implementations that conform to the VEA also adhere to the UISA, but may not necessarily adhere to the OEA.
- PowerPC operating environment architecture (OEA)—Defines the memory management model, supervisor-level registers, synchronization requirements, and the exception model. Implementations that conform to the OEA also adhere to the UISA and the VEA.

The PowerPC architecture allows a wide range of designs for such features as cache and system interface implementations. PowerPC 750CX implementations support the three levels of the architecture described above. For more information about the PowerPC architecture, see the *PowerPC Microprocessor Family: The Programming Environments* manual.

Specific features of PowerPC 750CX are listed in Section 1.2, "PowerPC 750CX Microprocessor Features," on Page 10.



## 1.4 PowerPC Registers and Programming Model

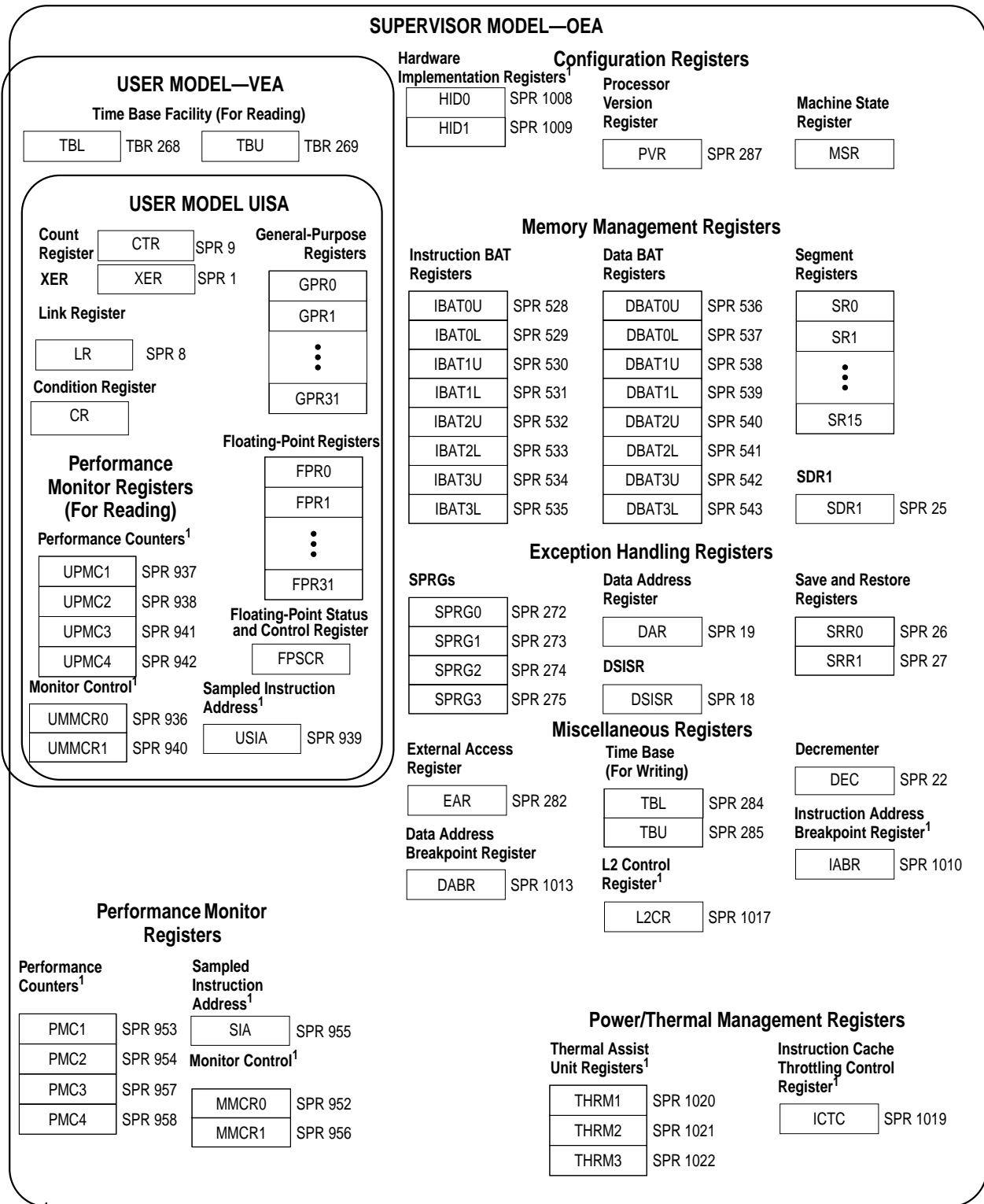
The PowerPC architecture defines register-to-register operations for most computational instructions. Source operands for these instructions are accessed from the registers or are provided as immediate values embedded in the instruction opcode. The three-register instruction format allows specification of a target register distinct from the two source operands. Load and store instructions transfer data between registers and memory.

PowerPC processors have two levels of privilege—supervisor mode of operation (typically used by the operating system) and user mode of operation (used by the application software). The programming models incorporate 32 GPRs, 32 FPRs, special-purpose registers (SPRs), and several miscellaneous registers. Each PowerPC microprocessor also has its own unique set of hardware implementation-dependent (HID) registers.

Having access to privileged instructions, registers, and other resources allows the operating system to control the application environment (providing virtual memory and protecting operating-system and critical machine resources). Instructions that control the state of the processor, the address translation mechanism, and supervisor registers can be executed only when the processor is operating in supervisor mode.

Figure 7 on page 25 shows all PowerPC 750CX registers available at the user and supervisor level. The numbers to the right of the SPRs indicate the number that is used in the syntax of the instruction operands to access the register.

For more information, see Chapter 2, "Programming Model" in the PowerPC 740 and PowerPC 750 User Manual.



These registers are processor- specific registers. They may not be supported by other PowerPC processors.

**Figure 7. PowerPC 750CX Microprocessor Programming Model—Registers**



The following tables summarize the PowerPC registers implemented in PowerPC 750CX; describe registers (excluding SPRs) defined by the architecture.

### Architecture-Defined Registers (Excluding SPRs)

Register	Level	Function
CR	User	The condition register (CR) consists of eight four-bit fields that reflect the results of certain operations, such as move, integer and floating-point compare, arithmetic, and logical instructions, and provide a mechanism for testing and branching.
FPRs	User	The 32 floating-point registers (FPRs) serve as the data source or destination for floating-point instructions. These 64-bit registers can hold single- or double-precision floating-point values.
FPSCR	User	The floating-point status and control register (FPSCR) contains the floating-point exception signal bits, exception summary bits, exception enable bits, and rounding control bits needed for compliance with the IEEE-754 standard.
GPRs	User	The 32 GPRs serve as the data source or destination for integer instructions.
MSR	Supervisor	The machine state register (MSR) defines the processor state. Its contents are saved when an exception is taken and restored when exception handling completes. PowerPC 750CX implements MSR[POW], (defined by the architecture as optional), which is used to enable the power management feature. PowerPC 750CX-specific MSR[PM] bit is used to mark a process for the performance monitor.
SR0–SR15	Supervisor	The sixteen 32-bit segment registers (SRs) define the 4-Gbyte space as sixteen 256-Mbyte segments. PowerPC 750CX implements segment registers as two arrays—a main array for data accesses and a shadow array for instruction accesses; see Figure 3 on page 10. Loading a segment entry with the Move to Segment Register ( <b>mtsr</b> ) instruction loads both arrays. The <b>mfsr</b> instruction reads the master register, shown as part of the data MMU in Figure 3 on page 10.

The OEA defines numerous special-purpose registers that serve a variety of functions, such as providing controls, indicating status, configuring the processor, and performing special operations. During normal execution, a program can access the registers, shown in Figure 7 on page 25, depending on the program's access privilege (supervisor or user, determined by the privilege-level (PR) bit in the MSR). GPRs and FPRs are accessed through operands that are part of the instructions. Access to registers can be explicit (that is, through the use of specific instructions for that purpose such as Move to Special-Purpose Register (**mtspr**) and Move from Special-Purpose Register (**mfspr**) instructions) or implicit, as the part of the execution of an instruction. Some registers can be accessed both explicitly and implicitly.

In PowerPC 750CX, all SPRs are 32 bits wide. Table “Architecture-Defined SPRs Implemented,” describes the architecture-defined SPRs implemented by PowerPC 750CX. In the *PowerPC Microprocessor Family: The Programming Environments* manual, these registers are described in detail, including bit descriptions.

Section 2.1.1 on Page 2-1 in the PowerPC 740 and PowerPC 750 User Manual describes how these registers are implemented in PowerPC 750CX. In particular, this section describes which features the PowerPC architecture defines as optional are implemented on PowerPC 750CX.

## Architecture-Defined SPRs Implemented

Register	Level	Function
LR	User	The link register (LR) can be used to provide the branch target address and to hold the return address after branch and link instructions.
BATs	Supervisor	The architecture defines 16 block address translation registers (BATs), which operate in pairs. There are four pairs of data BATs (DBATs) and four pairs of instruction BATs (IBATs). BATs are used to define and configure blocks of memory.
CTR	User	The count register (CTR) is decremented and tested by branch-and-count instructions.
DABR	Supervisor	The optional data address breakpoint register (DABR) supports the data address breakpoint facility.
DAR	User	The data address register (DAR) holds the address of an access after an alignment or DSI exception.
DEC	Supervisor	The decremter register (DEC) is a 32-bit decrementing counter that provides a way to schedule decremter exceptions.
DSISR	User	The DSISR defines the cause of data access and alignment exceptions.
EAR	Supervisor	The external access register (EAR) controls access to the external access facility through the External Control In Word Indexed ( <b>eciwx</b> ) and External Control Out Word Indexed ( <b>ecowx</b> ) instructions.
PVR	Supervisor	The processor version register (PVR) is a read-only register that identifies the processor.
SDR1	Supervisor	SDR1 specifies the page table format used in virtual-to-physical page address translation.
SRR0	Supervisor	The machine status save/restore register 0 (SRR0) saves the address used for restarting an interrupted program when a Return from Interrupt ( <b>rfi</b> ) instruction executes.
SRR1	Supervisor	The machine status save/restore register 1 (SRR1) is used to save machine status on exceptions and to restore machine status when an <b>rfi</b> instruction is executed.
SPRG0–SPRG3	Supervisor	SPRG0–SPRG3 are provided for operating system use.
TB	User: read Supervisor: read/write	The time base register (TB) is a 64-bit register that maintains the time of day and operates interval timers. The TB consists of two 32-bit fields—time base upper (TBU) and time base lower (TBL).
XER	User	The XER contains the summary overflow bit, integer carry bit, overflow bit, and a field specifying the number of bytes to be transferred by a Load String Word Indexed ( <b>lswx</b> ) or Store String Word Indexed ( <b>stswx</b> ) instruction.



Table “Implementation-Specific Registers,” describes the SPRs in PowerPC 750CX that are not defined by the PowerPC architecture.

Section 2.1.2 on Page 2-8 in the PowerPC 740 and PowerPC 750 User Manual gives detailed descriptions of these registers, including bit descriptions.

### Implementation-Specific Registers

Register	Level	Function
HID0	Supervisor	The hardware implementation-dependent register 0 (HID0) provides checkstop enables and other functions.
HID1	Supervisor	The hardware implementation-dependent register 1 (HID1) allows software to read the configuration of the PLL configuration signals.
IABR	Supervisor	The instruction address breakpoint register (IABR) supports instruction address breakpoint exceptions. It can hold an address to compare with instruction addresses in the IQ. An address match causes an instruction address breakpoint exception.
ICTC	Supervisor	The instruction cache-throttling control register (ICTC) has bits for controlling the interval at which instructions are fetched into the instruction buffer in the instruction unit. This helps control PowerPC 750CX’s overall junction temperature.
L2CR	Supervisor	The L2 cache control register (L2CR) is used to configure and operate the L2 cache.
MMCR0–MMCR1	Supervisor	The monitor mode control registers (MMCR0–MMCR1) are used to enable various performance monitoring interrupt functions. UMMCR0–UMMCR1 provide user-level read access to MMCR0–MMCR1.
PMC1–PMC4	Supervisor	The performance monitor counter registers (PMC1–PMC4) are used to count specified events. UPMC1–UPMC4 provide user-level read access to these registers.
SIA	Supervisor	The sampled instruction address register (SIA) holds the EA of an instruction executing at or around the time the processor signals the performance monitor interrupt condition. The USIA register provides user-level read access to the SIA.
THRM1, THRM2	Supervisor	THRM1 and THRM2 provide a way to compare the junction temperature against two user-provided thresholds. The thermal assist unit (TAU) can be operated so that the thermal sensor output is compared to only one threshold, selected in THRM1 or THRM2.
THRM3	Supervisor	THRM3 is used to enable the TAU and to control the output sample time.
UMMCR0–UMMCR1	User	The user monitor mode control registers (UMMCR0–UMMCR1) provide user-level read access to MMCR0–MMCR1.
UPMC1–UPMC4	User	The user performance monitor counter registers (UPMC1–UPMC4) provide user-level read access to PMC1–PMC4.
USIA	User	The user sampled instruction address register (USIA) provides user-level read access to the SIA register.



## 1.5 Instruction Set

All PowerPC instructions are encoded as single-word (32-bit) opcodes. Instruction formats are consistent among all instruction types, permitting efficient decoding to occur in parallel with operand accesses. This fixed instruction length and consistent format greatly simplify instruction pipelining.

For more information, see Chapter 2, "Programming Model" in the PowerPC 740 and PowerPC 750 User Manual.

### 1.5.1 PowerPC Instruction Set

The PowerPC instructions are divided into the following categories.

- Integer instructions—These include computational and logical instructions.
  - Integer arithmetic instructions
  - Integer compare instructions
  - Integer logical instructions
  - Integer rotate and shift instructions
- Floating-point instructions—These include floating-point computational instructions, as well as instructions that affect the FPSCR.
  - Floating-point arithmetic instructions
  - Floating-point multiply/add instructions
  - Floating-point rounding and conversion instructions
  - Floating-point compare instructions
  - Floating-point status and control instructions
- Load/store instructions—These include integer and floating-point load and store instructions.
  - Integer load and store instructions
  - Integer load and store multiple instructions
  - Floating-point load and store
  - Primitives used to construct atomic memory operations (**lwarx** and **stwcx.** instructions)
- Flow control instructions—These include branching instructions, condition register logical instructions, trap instructions, and other instructions that affect the instruction flow.
  - Branch and trap instructions
  - Condition register logical instructions
- Processor control instructions—These instructions are used for synchronizing memory accesses and management of caches, TLBs, and the segment registers.
  - Move to/from SPR instructions
  - Move to/from MSR
  - Synchronize
  - Instruction synchronize
  - Order loads and stores
- Memory control instructions—To provide control of caches, TLBs, and SRs.
  - Supervisor-level cache management instructions
  - User-level cache instructions
  - Segment register manipulation instructions
  - Translation lookaside buffer management instructions

This grouping does not indicate the execution unit that executes a particular instruction or group of instructions.



Integer instructions operate on byte, half-word, and word operands. Floating-point instructions operate on single-precision (one word) and double-precision (one double word) floating-point operands. The PowerPC architecture uses instructions that are four bytes long and word-aligned. It provides for byte, half-word, and word operand loads and stores between memory and a set of 32 GPRs. It also provides for word and double-word operand loads and stores between memory and a set of 32 floating-point registers (FPRs).

Computational instructions do not modify memory. To use a memory operand in a computation and then modify the same or another memory location, the memory contents must be loaded into a register, modified, and then written back to the target location with distinct instructions.

PowerPC processors follow the program flow when they are in the normal execution state; however, the flow of instructions can be interrupted directly by the execution of an instruction or by an asynchronous event. Either kind of exception may cause one of several components of the system software to be invoked.

Effective address computations for both data and instruction accesses use 32-bit unsigned binary arithmetic. A carry from bit 0 is ignored in 32-bit implementations.

## 1.5.2 PowerPC 750CX Microprocessor Instruction Set

PowerPC 750CX instruction set is defined as follows.

- PowerPC 750CX provides hardware support for all 32-bit PowerPC instructions.
- PowerPC 750CX implements the following instructions optional to the PowerPC architecture.
  - External Control In Word Indexed (**eciwx**).
  - External Control Out Word Indexed (**ecowx**).
  - Floating Select (**fsel**).
  - Floating Reciprocal Estimate Single-Precision (**fres**). Error < 1/4000.
  - Floating Reciprocal Square Root Estimate (**frsqrte**). Error < 1/4000.
  - Store Floating-Point as Integer Word (**stfiw**).

## 1.6 On-Chip Cache Implementation

The following subsections describe the PowerPC architecture's treatment of cache in general, and PowerPC 750CX-specific implementation, respectively. A detailed description of PowerPC 750CX L1 cache implementation is provided in Chapter 3, "Instruction and Data Cache Operation" in the PowerPC 740 and PowerPC 750 User Manual. The 750CX also contains an on-chip L2 unified cache.

### 1.6.1 PowerPC Cache Model

The PowerPC architecture does not define hardware aspects of cache implementations. For example, PowerPC processors can have unified caches, separate instruction and data caches (Harvard architecture), or no cache at all. PowerPC microprocessors control the following memory access modes on a page or block basis:

- Write-back/write-through mode
- Caching-inhibited mode
- Memory coherency

The caches are physically addressed, and the data cache can operate in either write-back or write-through mode, as specified by the PowerPC architecture.

The PowerPC architecture defines the term ‘cache block’ as the cacheable unit. The VEA and OEA define cache management instructions that a programmer can use to affect cache contents.

## 1.6.2 PowerPC 750CX Microprocessor Cache Implementation

PowerPC 750CX cache implementation is described in Section 1.2.4, “L1 (Level 1) Instruction and Data Caches,” on Page 16 and Section 1.2.5, “L2 (Level 2) Cache Implementation,” on Page 18. The BPU also contains a 64-entry BTIC that provides immediate access to cached branch-target instructions. For more information, see Section 1.2.2.2, “Branch Processing Unit (BPU),” on Page 13.

## 1.7 Exception Model

The following sections describe the PowerPC exception model and PowerPC 750CX implementation. A detailed description of PowerPC 750CX exception model is provided in Chapter 4, “Exceptions” in the PowerPC 740 and PowerPC 750 User Manual.

### 1.7.1 PowerPC Exception Model

The PowerPC exception mechanism allows the processor to interrupt the instruction flow to handle certain situations caused by external signals, errors, or unusual conditions arising from the instruction execution. When exceptions occur, information about the state of the processor is saved to certain registers, and the processor begins execution at an address (exception vector) predetermined for each exception. Exception processing occurs in supervisor mode.

Although multiple exception conditions can map to a single exception vector, a more specific condition may be determined by examining a register associated with the exception—for example, the DSISR and the FPSCR. Additionally, some exception conditions can be explicitly enabled or disabled by software.

The PowerPC architecture requires that exceptions be handled in program order; therefore, although a particular implementation may recognize exception conditions out of order, they are handled in order. When an instruction-caused exception is recognized, any unexecuted instructions that appear earlier in the instruction stream, including any that are undispatched, are required to complete before the exception is taken, and any exceptions those instructions cause must also be handled first; likewise, asynchronous, precise exceptions are recognized when they occur but are not handled until the instructions currently in the completion queue successfully retire or generate an exception, and the completion queue is emptied.

Unless a catastrophic condition causes a system reset or machine check exception, only one exception is handled at a time. For example, if one instruction encounters multiple exception conditions, those conditions are handled sequentially. After the exception handler handles an exception, the instruction processing continues until the next exception condition is encountered. Recognizing and handling exception conditions sequentially guarantees that exceptions are recoverable.

When an exception is taken, information about the processor state before the exception was taken is saved in SRR0 and SRR1. Exception handlers must save the information stored in SRR0 and SRR1 early to prevent the program state from being lost due to a system reset and machine check exception or due to an instruction-caused exception in the exception handler, and before enabling external interrupts.

The PowerPC architecture supports four types of exceptions.

- Synchronous, precise—These are caused by instructions. All instruction-caused exceptions are handled precisely; that is, the machine state at the time the exception occurs is known and can be completely restored. This means that (excluding the trap and system call exceptions) the address of the faulting instruction is provided to the exception handler and that neither the faulting instruction nor subsequent instructions in the code stream will complete execution before the exception is taken. Once the exception



is processed, execution resumes at the address of the faulting instruction (or at an alternate address provided by the exception handler). When an exception is taken due to a trap or system call instruction, execution resumes at an address provided by the handler.

- Synchronous, imprecise—The PowerPC architecture defines two imprecise floating-point exception modes, recoverable and nonrecoverable. Even though PowerPC 750CX provides a means to enable the imprecise modes, it implements these modes identically to the precise mode (that is, enabled floating-point exceptions are always precise).
- Asynchronous, maskable—The PowerPC architecture defines external and decremter interrupts as maskable, asynchronous exceptions. When these exceptions occur, their handling is postponed until the next instruction, and any exceptions associated with that instruction, completes execution. If no instructions are in the execution units, the exception is taken immediately upon determination of the correct restart address (for loading SRR0). As shown in the Table “PowerPC 750CX Microprocessor Exception Classifications,” PowerPC 750CX implements additional asynchronous, maskable exceptions.
- Asynchronous, nonmaskable—There are two nonmaskable asynchronous exceptions: system reset and the machine check exception. These exceptions may not be recoverable, or may provide a limited degree of recoverability. Exceptions report recoverability through the MSR[RI] bit.

### 1.7.2 PowerPC 750CX Microprocessor Exception Implementation

PowerPC 750CX exception classes described above are shown in the Table “PowerPC 750CX Microprocessor Exception Classifications.” Although exceptions have other characteristics, such as priority and recoverability, describes categories of exceptions PowerPC 750CX handles uniquely, includes no synchronous imprecise exceptions; although the PowerPC architecture supports imprecise handling of floating-point exceptions, PowerPC 750CX implements these exception modes precisely.

#### PowerPC 750CX Microprocessor Exception Classifications

Synchronous/Asynchronous	Precise/Imprecise	Exception Type
Asynchronous, non-maskable	Imprecise	Machine check, system reset
Asynchronous, maskable	Precise	External, decremter, system management, performance monitor, and thermal management interrupts
Synchronous	Precise	Instruction-caused exceptions

Table “Exceptions and Conditions,” lists PowerPC 750CX exceptions and conditions that cause them. Exceptions specific to PowerPC 750CX are indicated.

#### Exceptions and Conditions

Exception Type	Vector Offset (hex)	Causing Conditions
Reserved	00000	—
System reset	00100	Assertion of either HRESET or SRESET or at power-on reset
1. PowerPC 750CX-specific		

**Exceptions and Conditions (Continued)**

Exception Type	Vector Offset (hex)	Causing Conditions
Machine check	00200	Assertion of $\overline{TEA}$ during a data bus transaction, assertion of $\overline{MCP}$ , an address, data or L2 double bit error. MSR[ME] must be set.
DSI	00300	As specified in the PowerPC architecture. For TLB misses on load, store, or cache operations, a DSI exception occurs if a page fault occurs.
ISI	00400	As defined by the PowerPC architecture.
External interrupt	00500	MSR[EE] = 1 and $\overline{INT}$ is asserted.
Alignment	00600	<ul style="list-style-type: none"> <li>• A floating-point load/store, <b>stmw</b>, <b>stwcx</b>, <b>lmw</b>, <b>lwarx</b>, <b>eciw</b>x or <b>ecow</b>x instruction operand is not word-aligned.</li> <li>• A multiple/string load/store operation is attempted in little-endian mode.</li> <li>• The operand of <b>dcbz</b> is in memory that is write-through-required or caching-inhibited or the cache is disabled</li> </ul>
Program	00700	As defined by the PowerPC architecture.
Floating-point unavailable	00800	As defined by the PowerPC architecture.
Decrementer	00900	As defined by the PowerPC architecture, when the most significant bit of the DEC register changes from 0 to 1 and MSR[EE] = 1.
Reserved	00A00–00BFF	—
System call	00C00	Execution of the System Call ( <b>sc</b> ) instruction.
Trace	00D00	MSR[SE] = 1 or a branch instruction completes and MSR[BE] = 1. Unlike the architecture definition, <b>isync</b> does not cause a trace exception
Reserved	00E00	PowerPC 750CX does not generate an exception to this vector. Other PowerPC processors may use this vector for floating-point assist exceptions.
Reserved	00E10–00EFF	—
Performance monitor <sup>1</sup>	00F00	The limit specified in a PMC register is reached and MMCR0[ENINT] = 1
Instruction address breakpoint <sup>1</sup>	01300	IABR[0–29] matches EA[0–29] of the next instruction to complete, IABR[TE] matches MSR[IR], and IABR[BE] = 1.
Reserved	01400–016FF	—
Thermal management interrupt <sup>1</sup>	01700	Thermal management is enabled, the junction temperature exceeds the threshold specified in THRM1 or THRM2, and MSR[EE] = 1.
1. PowerPC 750CX-specific		



## Exceptions and Conditions (Continued)

Exception Type	Vector Offset (hex)	Causing Conditions
Reserved	01800–02FFF	—
1. PowerPC 750CX-specific		

### 1.8 Memory Management

The following subsections describe the memory management features of the PowerPC architecture, and PowerPC 750CX implementation, respectively. A detailed description of PowerPC 750CX MMU implementation is provided in Chapter 5, "Memory Management" in the PowerPC 740 and PowerPC 750 User Manual.

#### 1.8.1 PowerPC Memory Management Model

The primary functions of the MMU are to translate logical (effective) addresses to physical addresses for memory accesses and to provide access protection on blocks and pages of memory. There are two types of accesses generated by PowerPC 750CX that require address translation—instruction accesses, and data accesses to memory generated by load, store, and cache control instructions.

The PowerPC architecture defines different resources for 32- and 64-bit processors; PowerPC 750CX implements the 32-bit memory management model. The memory-management model provides 4 Gbytes of logical address space accessible to supervisor and user programs with a 4-Kbyte page size and 256-Mbyte segment size. BAT block sizes range from 128 Kbyte to 256 Mbyte and are software selectable. In addition, it defines an interim 52-bit virtual address and hashed page tables for generating 32-bit physical addresses.

The architecture also provides independent four-entry BAT arrays for instructions and data that maintain address translations for blocks of memory. These entries define blocks that can vary from 128 Kbytes to 256 Mbytes. The BAT arrays are maintained by system software.

The PowerPC MMU and exception model support demand-paged virtual memory. Virtual memory management permits execution of programs larger than the size of physical memory; demand-paged implies that individual pages are loaded into physical memory from system memory only when they are first accessed by an executing program.

The hashed page table is a variable-sized data structure that defines the mapping between virtual page numbers and physical page numbers. The page table size is a power of 2, and its starting address is a multiple of its size. The page table contains a number of page table entry groups (PTEGs). A PTEG contains eight page table entries (PTEs) of eight bytes each; therefore, each PTEG is 64 bytes long. PTEG addresses are entry points for table search operations.

Setting MSR[IR] enables instruction address translations and MSR[DR] enables data address translations. If the bit is cleared, the respective effective address is the same as the physical address.

#### 1.8.2 PowerPC 750CX Microprocessor Memory Management Implementation

PowerPC 750CX implements separate MMUs for instructions and data. It implements a copy of the segment registers in the instruction MMU; however, read and write accesses (**mfsr** and **mtsr**) are handled through the segment registers implemented as part of the data MMU. PowerPC 750CX MMU is described in Section 1.2.3, "Memory Management Units (MMUs)," on Page 15.

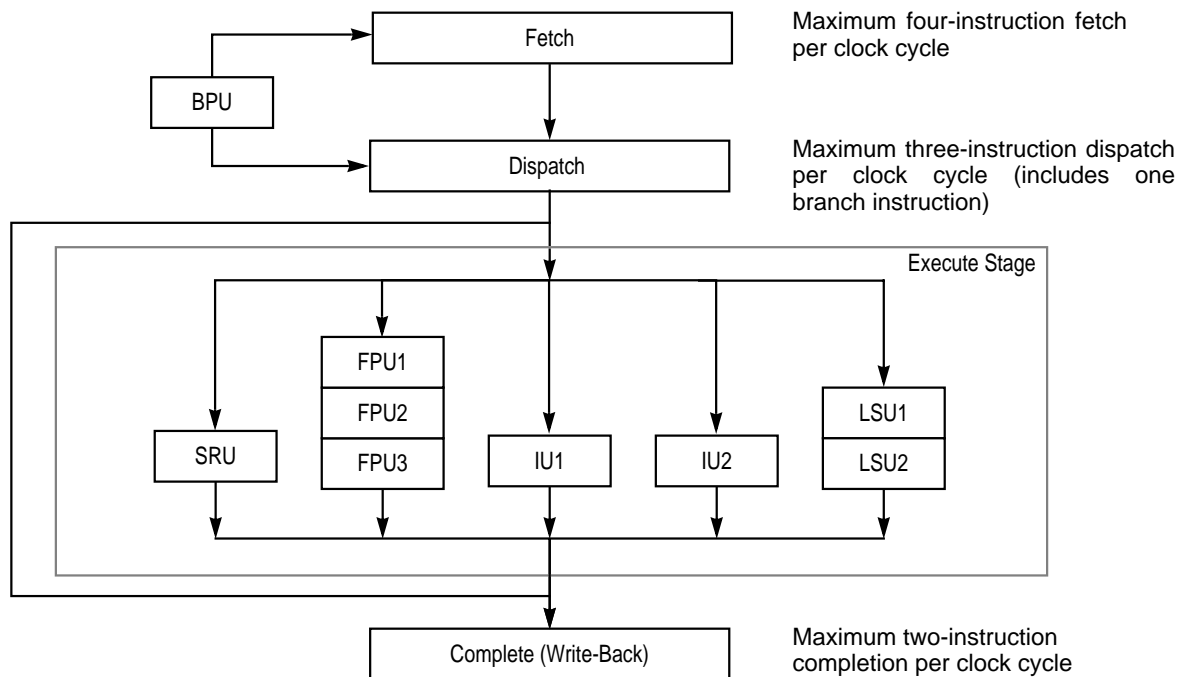
The R (referenced) bit is updated in the PTE in memory (if necessary) during a table search due to a TLB miss. Updates to the changed (C) bit are treated like TLB misses. A complete table search is performed and the entire TLB entry is rewritten to update the C bit.

### 1.9 Instruction Timing

PowerPC 750CX is a pipelined, superscalar processor. A pipelined processor is one in which instruction processing is divided into discrete stages, allowing work to be done on different instructions in each stage. For example, after an instruction completes one stage, it can pass on to the next stage leaving the previous stage available to the subsequent instruction. This improves overall instruction throughput.

A superscalar processor is one that issues multiple independent instructions into separate execution units, allowing instructions to execute in parallel. PowerPC 750CX has six independent execution units, two for integer instructions, and one each for floating-point instructions, branch instructions, load and store instructions, and system register instructions. Having separate GPRs and FPRs allows integer, floating-point calculations, and load and store operations to occur simultaneously without interference. Additionally, rename buffers are provided to allow operations to post execution results for use by subsequent instructions without committing them to the architected FPRs and GPRs.

As shown in Figure 8, the common pipeline of PowerPC 750CX has four stages through which all instructions must pass—fetch, decode/dispatch, execute, and complete/write back. Some instructions occupy multiple stages simultaneously and some individual execution units have additional stages. For example, the floating-point pipeline consists of three stages through which all floating-point instructions must pass.



**Figure 8. Pipeline Diagram**

**Note:** Figure 8 does not show features, such as reservation stations and rename buffers that reduce stalls and improve instruction throughput.

The instruction pipeline in PowerPC 750CX has four major pipeline stages, described as follows.



- The fetch pipeline stage primarily involves retrieving instructions from the memory system and determining the location of the next instruction fetch. The BPU decodes branches during the fetch stage and removes those that do not update CTR or LR from the instruction stream.
- The dispatch stage is responsible for decoding the instructions supplied by the instruction fetch stage and determining which instructions can be dispatched in the current cycle. If source operands for the instruction are available, they are read from the appropriate register file or rename register to the execute pipeline stage. If a source operand is not available, dispatch provides a tag that indicates which rename register will supply the operand when it becomes available. At the end of the dispatch stage, the dispatched instructions and their operands are latched by the appropriate execution unit.
- Instructions executed by the IUs, FPU, SRU, and LSU are dispatched from the bottom two positions in the instruction queue. In a single clock cycle, a maximum of two instructions can be dispatched to these execution units in any combination. When an instruction is dispatched, it is assigned a position in the six-entry completion queue. A branch instruction can be issued on the same clock cycle for a maximum three-instruction dispatch.
- During the execute pipeline stage, each execution unit that has an executable instruction executes the selected instruction (perhaps over multiple cycles), writes the instruction's result into the appropriate rename register, and notifies the completion stage that the instruction has finished execution. In the case of an internal exception, the execution unit reports the exception to the completion pipeline stage and (except for the FPU) discontinues instruction execution until the exception is handled. The exception is not signaled until that instruction is the next to be completed. Execution of most floating-point instructions is pipelined within the FPU allowing up to three instructions to be executing in the FPU concurrently. The FPU stages are multiply, add, and round-convert. Execution of most load/store instructions is also pipelined. The load/store unit has two pipeline stages. The first stage is for effective address calculation and MMU translation and the second stage is for accessing the data in the cache.
- The complete pipeline stage maintains the correct architectural machine state and transfers execution results from the rename registers to the GPRs and FPRs (and CTR and LR, for some instructions) as instructions are retired. As with dispatching instructions from the instruction queue, instructions are retired from the two bottom positions in the completion queue. If completion logic detects an instruction causing an exception, all following instructions are cancelled, their execution results in rename registers are discarded, and instructions are fetched from the appropriate exception vector.

Because the PowerPC architecture can be applied to such a wide variety of implementations, instruction timing varies among PowerPC processors.

For a detailed discussion of instruction timing with examples and a table of latencies for each execution unit, see Chapter 6 "Instruction Timing," in the PowerPC 740 and PowerPC 750 User Manual.

## 1.10 Power Management

PowerPC 750CX provides four power modes, selectable by setting the appropriate control bits in the MSR and HID0 registers. The four power modes are as follows.

- Full-power—This is the default power state of PowerPC 750CX. PowerPC 750CX is fully powered and the internal functional units are operating at the full processor clock speed. If the dynamic power management mode is enabled, functional units that are idle will automatically enter a low-power state without affecting performance, software execution, or external hardware.
- Doze—All the functional units of PowerPC 750CX are disabled except for the time base/decrementer registers and the bus snooping logic. When the processor is in doze mode, an external asynchronous interrupt, a system management interrupt, a decremter exception, a hard or soft reset, or machine check brings PowerPC 750CX into the full-power state. PowerPC 750CX in doze mode maintains the PLL in a



fully powered state and locked to the system external clock input (SYSCLK) so a transition to the full-power state takes only a few processor clock cycles.

- **Nap**—The nap mode further reduces power consumption by disabling bus snooping, leaving only the time base register and the PLL in a powered state. PowerPC 750CX returns to the full-power state upon receipt of an external asynchronous interrupt, a system management interrupt, a decremter exception, a hard or soft reset, or a machine check input ( $\overline{MCP}$ ). A return to full-power state from a nap state takes only a few processor clock cycles. When the processor is in nap mode, if  $\overline{QACK}$  is negated, the processor is put in doze mode to support snooping.
- **Sleep**—Sleep mode minimizes power consumption by disabling all internal functional units, after which external system logic may disable the PLL and SYSCLK. Returning PowerPC 750CX to the full-power state requires the enabling of the PLL and SYSCLK, followed by the assertion of an external asynchronous interrupt, a system management interrupt, a hard or soft reset, or a machine check input ( $\overline{MCP}$ ) signal after the time required to relock the PLL.

Chapter 10, "Power and Thermal Management" in the PowerPC 740 and PowerPC 750 User Manual provides information about power saving and thermal management modes for PowerPC 750CX.

## 1.11 Thermal Management

PowerPC 750CX's thermal assist unit (TAU) provides a way to control heat dissipation. This ability is particularly useful in portable computers, which, due to power consumption and size limitations, cannot use desktop cooling solutions such as fans. Therefore, better heat sink designs coupled with intelligent thermal management is of critical importance for high performance portable systems.

Primarily, the thermal management system monitors and regulates the system's operating temperature. For example, if the temperature is about to exceed a set limit, the system can be made to slow down or even suspend operations temporarily in order to lower the temperature.

The thermal management facility also ensures that the processor's junction temperature does not exceed the operating specification. To avoid the inaccuracies that arise from measuring junction temperature with an external thermal sensor, PowerPC 750CX's on-chip thermal sensor and logic tightly couples the thermal management implementation.

The TAU consists of a thermal sensor, digital-to-analog convertor, comparator, control logic, and the dedicated SPRs described in Section 1.4, "PowerPC Registers and Programming Model," on Page 24. The TAU does the following.

- Compares the junction temperature against user-programmable thresholds.
- Generates a thermal management interrupt if the temperature crosses the threshold.
- Enables the user to estimate the junction temperature by way of a software successive approximation routine.

The TAU is controlled through the privileged **mtspr/mfspr** instructions to the three SPRs provided for configuring and controlling the sensor control logic, which function as follows.

- **THRM1** and **THRM2** provide the ability to compare the junction temperature against two user-provided thresholds. Having dual thresholds gives the thermal management software finer control of the junction temperature. In single threshold mode, the thermal sensor output is compared to only one threshold in either **THRM1** or **THRM2**.



- THRM3 is used to enable the TAU and to control the comparator output sample time. The thermal management logic manages the thermal management interrupt generation and time multiplexed comparisons in the dual threshold mode as well as other control functions.

Instruction cache throttling provides control of PowerPC 750CX's overall junction temperature by reducing the rate at which instructions are fetched. This feature is accessed through the ICTC register.

Chapter 10, "Power and Thermal Management" in the PowerPC 740 and PowerPC 750 User Manual provides information about power saving and thermal management modes for PowerPC 750CX.

## 1.12 Performance Monitor

PowerPC 750CX incorporates a performance monitor facility that system designers can use to help bring up, debug, and optimize software performance. The performance monitor counts events during execution of code, relating to dispatch, execution, completion, and memory accesses.

The performance monitor incorporates several registers that can be read and written to by supervisor-level software. User-level versions of these registers provide read-only access for user-level applications. These registers are described in Section 1.4, "PowerPC Registers and Programming Model," on Page 24. Performance monitor control registers, MMCR0 or MMCR1, can be used to specify which events are to be counted and the conditions for which a performance monitoring interrupt is taken. Additionally, the sampled instruction address register, SIA (USIA), holds the address of the first instruction to complete after the counter overflowed.

Attempting to write to a user-read-only performance monitor register causes a program exception, regardless of the MSR[PR] setting.

When a performance monitoring interrupt occurs, program execution continues from vector offset 0x00F00.

Chapter 11, "Performance Monitor" in the PowerPC 740 and PowerPC 750 User Manual, describes the operation of the performance monitor diagnostic tool incorporated in PowerPC 750CX.

## Document History File

The following changes have been made to this document.

Date	Page	Description
6/19/00		Initial Release.
10/31/00	1	Changed reference to PVR to Table , "Summary of Differences" on page 6.
	2	Changed Figures 1 and 2 (L2CR) final bit from L2CS to L2IP.
	2	Add Note regarding 750CX DD1.0 Data cache reload bus width is 64-bits.
	3	Updated Power, Voltages, Frequency table (added Core Speed Range and Max. Bus Speed.
	3,4	Updated Packaging section (pins removed, $\overline{CKSTP\_OUT}$ , and DBWO).
	5	Updated Bus Pull-up Resistor Requirements (Level Protection).
	5	Deleted Dynamic Power Management Disabled section.
	6	Updated Summary of Differences table.
21	Updated Figure 6, "PowerPC 750CX Microprocessor Signal Groups" on page 21.	



© International Business Machines Corporation 2000

Printed in the United States of America

10/31/00

All Rights Reserved

The information contained in this document is subject to change without notice. The products described in this document are NOT intended for use in implantation or other life support applications where malfunction may result in injury or death to persons. The information contained in this document does not affect or change IBM's product specifications or warranties. Nothing in this document shall operate as an express or implied license or indemnity under the intellectual property rights of IBM or third parties. All information contained in this document was obtained in specific environments, and is presented as illustration. The results obtained in other operating environments may vary.

While the information contained herein is believed to be accurate, such information is preliminary, and should not be relied upon for accuracy or completeness, and no representations or warranties of accuracy or completeness are made.

THE INFORMATION CONTAINED IN THIS DOCUMENT IS PROVIDED ON AN "AS IS" BASIS. In no event will IBM be liable for any damages arising directly or indirectly from any use of the information contained in this document.

IBM Microelectronics Division  
1580 Route 52, Bldg. 504  
Hopewell Junction, NY  
12533-6531

The IBM home page can be found at <http://www.ibm.com>

The IBM Microelectronics Division home page can be found at <http://www.chips.ibm.com>