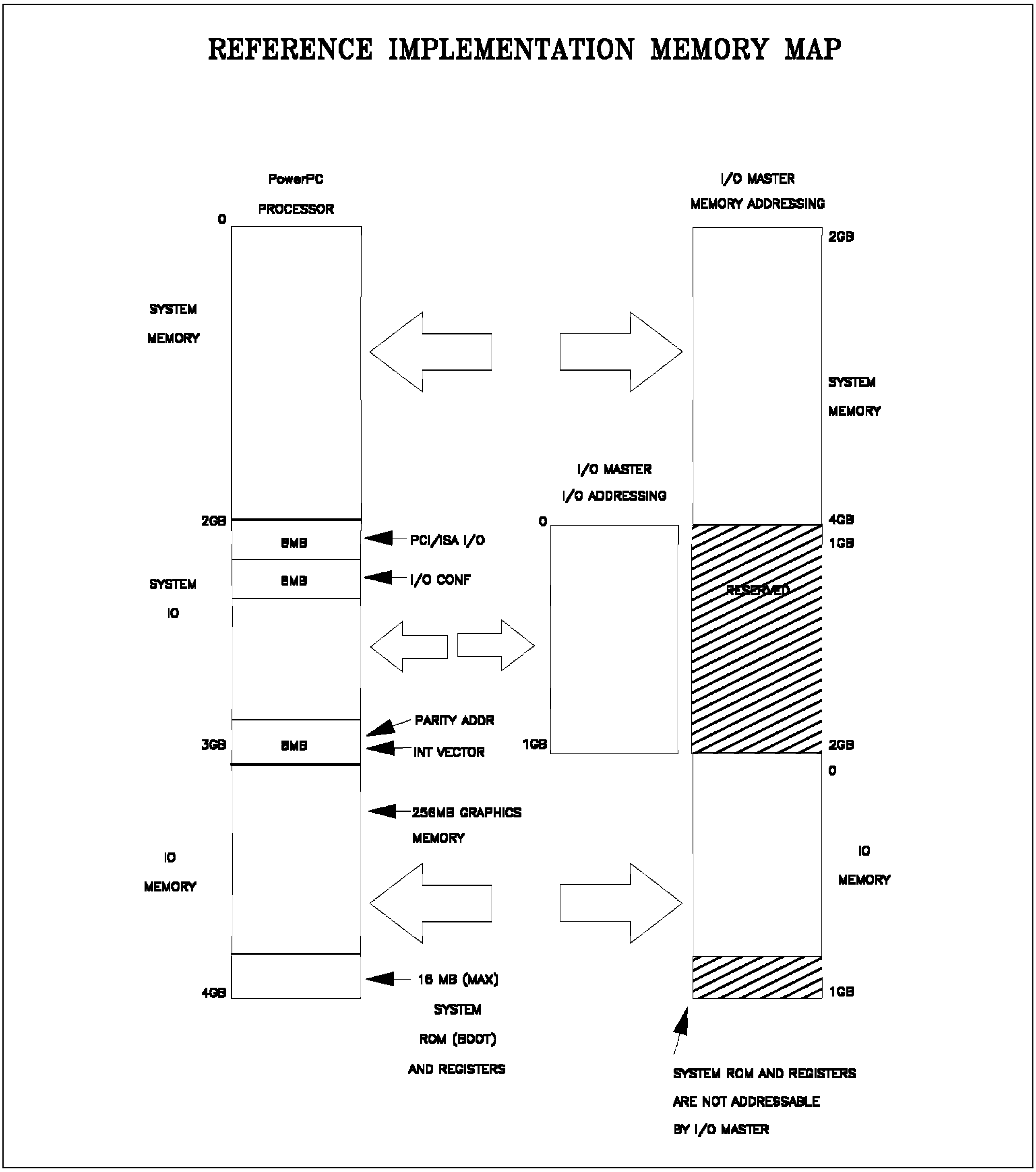

Figure 19. Reference Implementation Memory Map

The System Memory is addressed by the operating system and user applications executing in the PowerPC processor. Access to System Memory by the PowerPC processor is in the 0 to 2 GB range. System Memory accessed by a I/O master will be addressed in the 2 to 4 GB range, as shown on the right side of the figure.

The System I/O address space is used for input and output to the devices attached to the secondary or tertiary I/O buses (e.g. PCI and ISA). The first 8 MB of the System I/O address space are used for most system I/O. This address space includes interfaces to devices such as system registers, timers, parallel and serial ports and NVRAM. For a detailed list of which devices are included in this area and where the devices appear within this area, refer to Section 6.1.5, “I/O Device Mapping.” The next 8 MB are used for the I/O configuration. Refer to Section 6.1.7, “I/O Configuration Space Mapping,” for mappings of the I/O configuration information. The first page after this 8 MB of configuration information is used for SCSI I/O as an implementation option. At the bottom of the System I/O address space is an 8-MB area which is reserved. Within this area is a page for the memory parity address and a second page for the interrupt vector. Refer to Section 6.1.8, “Additional System I/O Mappings,” for mappings of this area.

The I/O Memory area is used for graphics memory, System ROM, and other I/O Memory-based devices. The bottom 16 MB of the I/O Memory area is reserved for fixed address ISA devices and adaptors. The potential for video memory with a 256-MB memory space is shown in the figure. The size of this space will have to be traded off against other I/O Memory use in different configurations. The System ROM and space for system registers is allocated to the top 16 MB of the I/O Memory area address space. System ROM is implemented using Flash ROM. The System ROM contains the code for power-on self tests (POST), code for establishing the initial configuration of the system, code for booting, and system-specific information.

As shown on the right side of the figure, the I/O master has two modes of addressing. The I/O master doing memory address mode transfers will see System Memory as having addresses from 2 GB to 4 GB. The I/O Memory is addressed in the range of 0 to 1 GB. This maintains compatibility with fixed-address ISA devices and adaptors. The upper 16 MB of this space is reserved for the System ROM and system registers and is not addressable by the I/O master. In the memory addressing mode, the PCI master cannot address the space from 1 GB to 2 GB. The I/O master doing I/O addressing mode transfers will see the System I/O memory space as having addresses from 0 to 1 GB.

6.1.1 Processor View of the Memory Map

Figure 20 shows the view of memory from the PowerPC processor. Processor addresses in the 0 through 2 GB - 1 range access System Memory. These memory cycles will not be passed to the I/O bus. Addresses in the range 2 GB through 2 GB + 64 KB - 1 are for ISA-standard I/O. Addresses from 2 GB + 64 KB through 2 GB + 8 MB - 1 are reserved and used in an alternate ISA-standard I/O addressing mode (see below). Addresses between 2 GB + 8 MB through 3 GB - 1 are used by the software for I/O. Within this range, the space 2 GB + 8 MB through 2 GB + 16 MB - 1 is used for I/O configuration and 3 GB - 8 MB through 3 GB - 1 is used for the memory parity address register and the interrupt vector register. Processor memory cycles in the 2 GB through 3 GB - 1 range will be run by the memory controller as an I/O cycle with the most significant bit of the address set to zero. This translation puts the address in the 0 through 1 GB - 1 range for the I/O addresses. Processor memory cycles in the 3 GB through 4 GB - 16 MB - 1 range are used to address I/O Memory. The memory controller will run these addresses as I/O Memory cycles with the two most significant bits set to zero. As an implementation, only the first 16 MB of this space is forwarded to the ISA memory space. Addresses in the range 4 GB - 16 MB through 4 GB - 1 are used for the system ROM and system registers. The memory controller will access the ROM using I/O cycles.

This version of the memory map is called the *contiguous memory map* because the 64 KB of ISA I/O is contained in 64 contiguous KBs. There is no protection mechanism implemented for this area other than the page protection inherent in any memory space. Some operating systems such as Windows NT will use this I/O addressing scheme. Device drivers performing I/O in this area risk interfering with each other. Within the kernel, this risk is limited through testing. For device drivers contained within applications, the risk will be avoided by avoiding applications having device drivers which may corrupt other device drivers.

Figure 21 shows an alternate approach for handling the ISA I/O addresses. This approach is called the *discontiguous memory map*. The only difference between this map and the contiguous memory map is within the 8 MB address space beginning at 2 GB. Within this space, the objective is to have a single device on a memory page. This Reference Implementation translates this 64 KB of I/O addressing to allow for

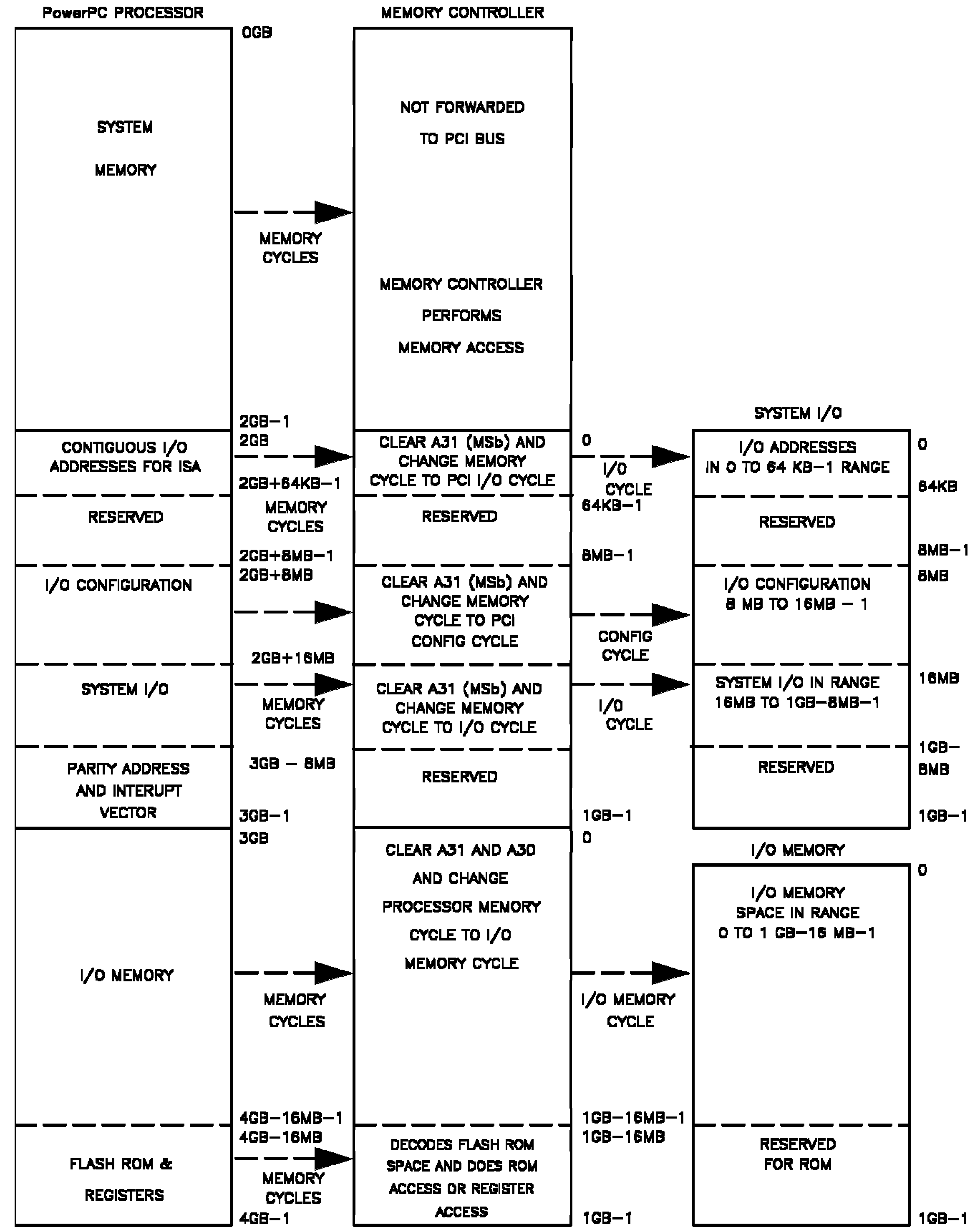
PowerPC PROCESSOR
MEMORY CONTROLLER
SYSTEM I/O
I/O MEMORY
SYSTEM MEMORY
0GB
MEMORY CYCLES
NOT FORWARDED TO PCI BUS
MEMORY CONTROLLER PERFORMS MEMORY ACCESS
2GB−1
2GB
CONTIGUOUS I/O ADDRESSES FOR ISA
2GB+64KB−1
MEMORY CYCLES
CLEAR A31 (MSb) AND CHANGE MEMORY CYCLE TO PCI I/O CYCLE
0
I/O CYCLE
64KB−1
I/O ADDRESSES IN 0 TO 64 KB−1 RANGE
0
64KB
RESERVED
RESERVED
RESERVED
2GB+8MB−1
2GB+8MB
8MB−1
8MB−1
8MB
I/O CONFIGURATION
CLEAR A31 (MSb) AND CHANGE MEMORY CYCLE TO PCI CONFIG CYCLE
CONFIG CYCLE
I/O CONFIGURATION 8 MB TO 16MB − 1
2GB+16MB
SYSTEM I/O
MEMORY CYCLES
CLEAR A31 (MSb) AND CHANGE MEMORY CYCLE TO I/O CYCLE
I/O CYCLE
SYSTEM I/O IN RANGE 16MB TO 1GB−8MB−1
16MB
PARITY ADDRESS AND INTERUPT VECTOR
3GB − 8MB
RESERVED
RESERVED
1GB− 8MB
3GB−1
1GB−1
1GB−1
3GB
0
I/O MEMORY
CLEAR A31 AND A30 AND CHANGE PROCESSOR MEMORY CYCLE TO I/O MEMORY CYCLE
MEMORY CYCLES
I/O MEMORY CYCLE
0
I/O MEMORY SPACE IN RANGE 0 TO 1 GB−16 MB−1
4GB−16MB−1
4GB−16MB
1GB−16MB−1
1GB−16MB
FLASH ROM & REGISTERS
MEMORY CYCLES
DECODES FLASH ROM SPACE AND DOES ROM ACCESS OR REGISTER ACCESS
RESERVED FOR ROM
4GB−1
1GB−1
1GB−1

Figure 20. Reference Implementation Contiguous Memory Map

2048 devices, each having 32 bytes. In this address space, each device has the first 32 bytes of a 4096-byte page.

When in the discontiguous mode, the memory controller will translate the address and issue a PCI I/O cycle to that address. For addresses in the range of 2 GB through 2 GB + 8 MB - 1, the memory controller will

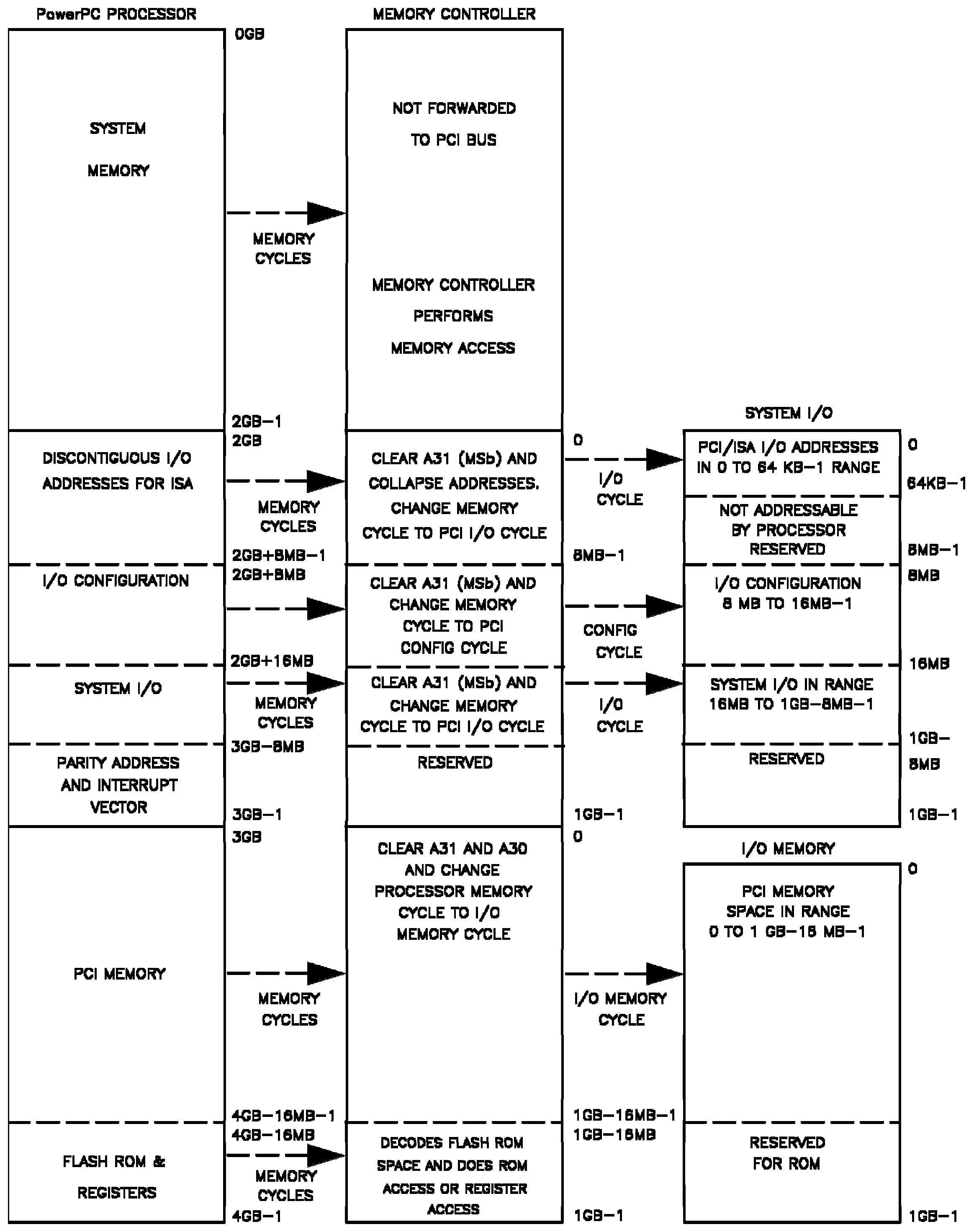

Figure 21. Reference Implementation Discontiguous Memory Map

clear the most significant bit of the address and will convert it to the 0 through 64 KB - 1 range. Then the memory controller will issue an I/O cycle to this converted address. This address translation by the memory controller makes the low-order 5 bits of the I/O address equivalent to the low-order 5 bits of the processor I/O address. The next 7 bits of the processor address are ignored. The processor address bits 9 through 19 are placed in the next-higher-order 11 bits of the I/O address. The remaining processor high-order address bits indicate that this address is in the ISA address space. They must be zero, except the most significant bit, which is set to one to indicate System I/O or I/O Memory. **Notice that the processor numbers bits with zero as the most significant bit (MSb)**. Table 13 shows this address translation using a 16-bit ISA address which by convention has the LSb labeled 0.

Table 13. Translation of ISA I/O Addresses

Processor Address		**ISA I/O Address**
31 LSb	is assigned to	0 LSb
30	is assigned to	1
29	is assigned to	2
28	is assigned to	3
27	is assigned to	4
26 - 20	ignore	
19	is assigned to	5
18	is assigned to	6
17	is assigned to	7
16	is assigned to	8
15	is assigned to	9
14	is assigned to	10
13	is assigned to	11
12	is assigned to	12
11	is assigned to	13
10	is assigned to	14
9	is assigned to	15 MSb
8 - 1	must be 0	
0 MSb	must be 1	

The switch between the contiguous memory map and the discontiguous memory map will be controlled by software setting a configuration register. The design of the system should not preclude switching modes as required for the software environment. For instance, an operating system which is designed to use the discontiguous mode may switch to the contiguous mode when emulating other operating environments for applications which have drivers using the contiguous mode.

With the device mapping which is shown in the column for the discontiguous memory map in Table 14, most devices are contained within a page. Using memory page protection, device drivers can be protected from each other. Operating systems such as OS/2 built on the Workplace environment will use this discontiguous memory map.

6.1.2 I/O Master View of the I/O Map

Figure 22 shows an I/O master view of the memory map when in the I/O mode. The memory controller does not perform any operations for I/O cycles generated by I/O masters or the I/O bus bridge.

6.1.3 I/O Master View of the Memory Map

Figure 23 shows an I/O master view of the memory map when the I/O master is performing memory transfers. The memory controller ignores any memory operation for I/O master memory cycles which fall in the 16 MB through 2 GB - 1 range of I/O Memory. The memory controller performs a System Memory access for I/O master memory cycles which fall into the 2 GB through 4 GB - 1 range of I/O Memory. For accesses in this range, the memory controller complements the most significant bit of the memory address from the I/O master. This modified address, which is in the range 0 through 2 GB - 1, is used for System Memory access and cache snoop.

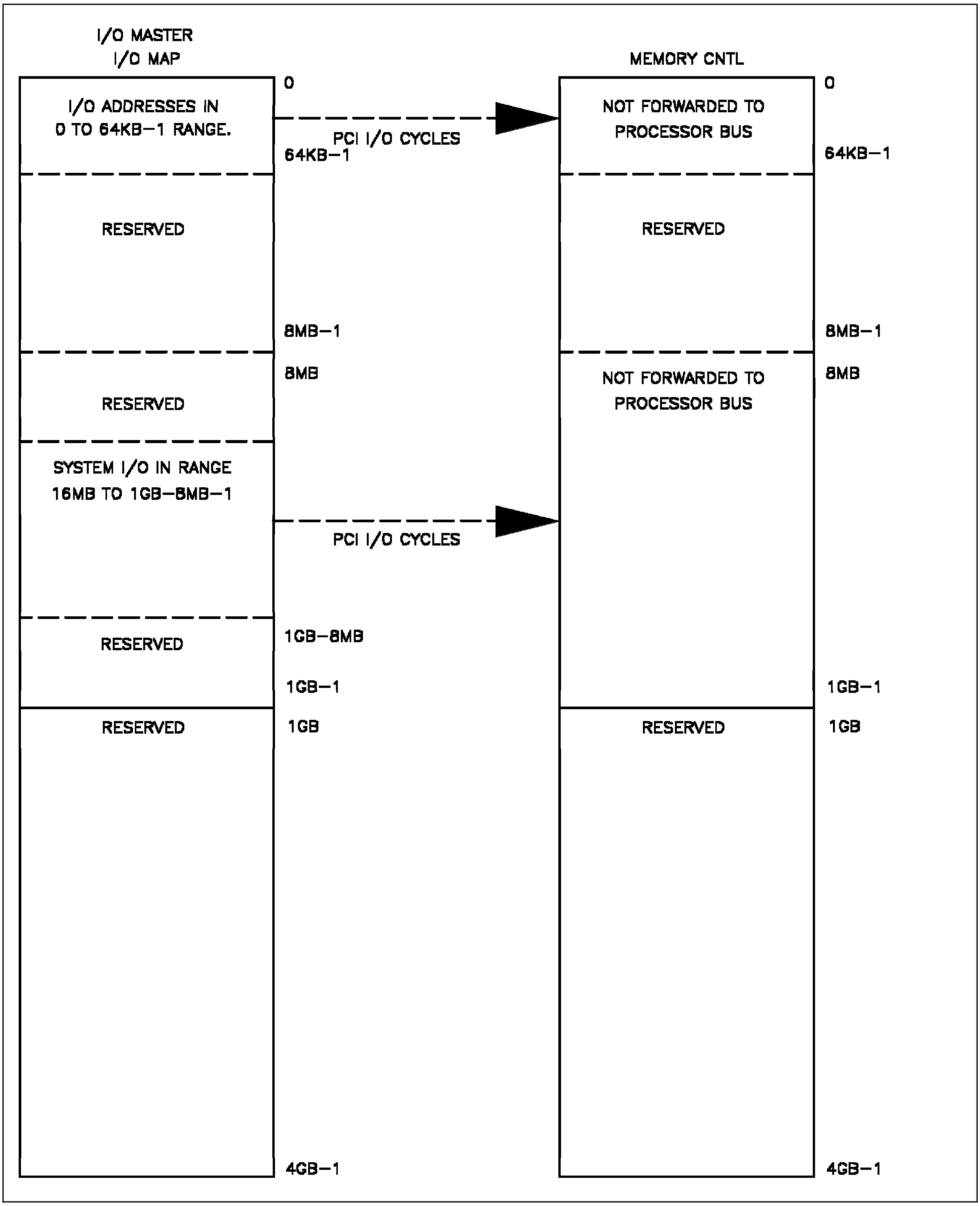

Figure 22. I/O Master View of I/O Map

/ In addition to these memory transfers, ISA masters may store to System Memory in the range of 0 to 16 MB. Within this 16-MB range, holes are defined by ISA usage conventions. I/O Memory cycles on behalf of ISA masters in the range 0 to 16 MB will be forwarded to System Memory in the range of 0 to 16 MB.
/ If a system has an ISA bus, then it is recommended that PCI devices not use I/O Memory in the range of 0
/ to 16 MB, and that PCI devices in the I/O Memory range of 2 GB to 2 GB + 16 MB avoid conflicts with

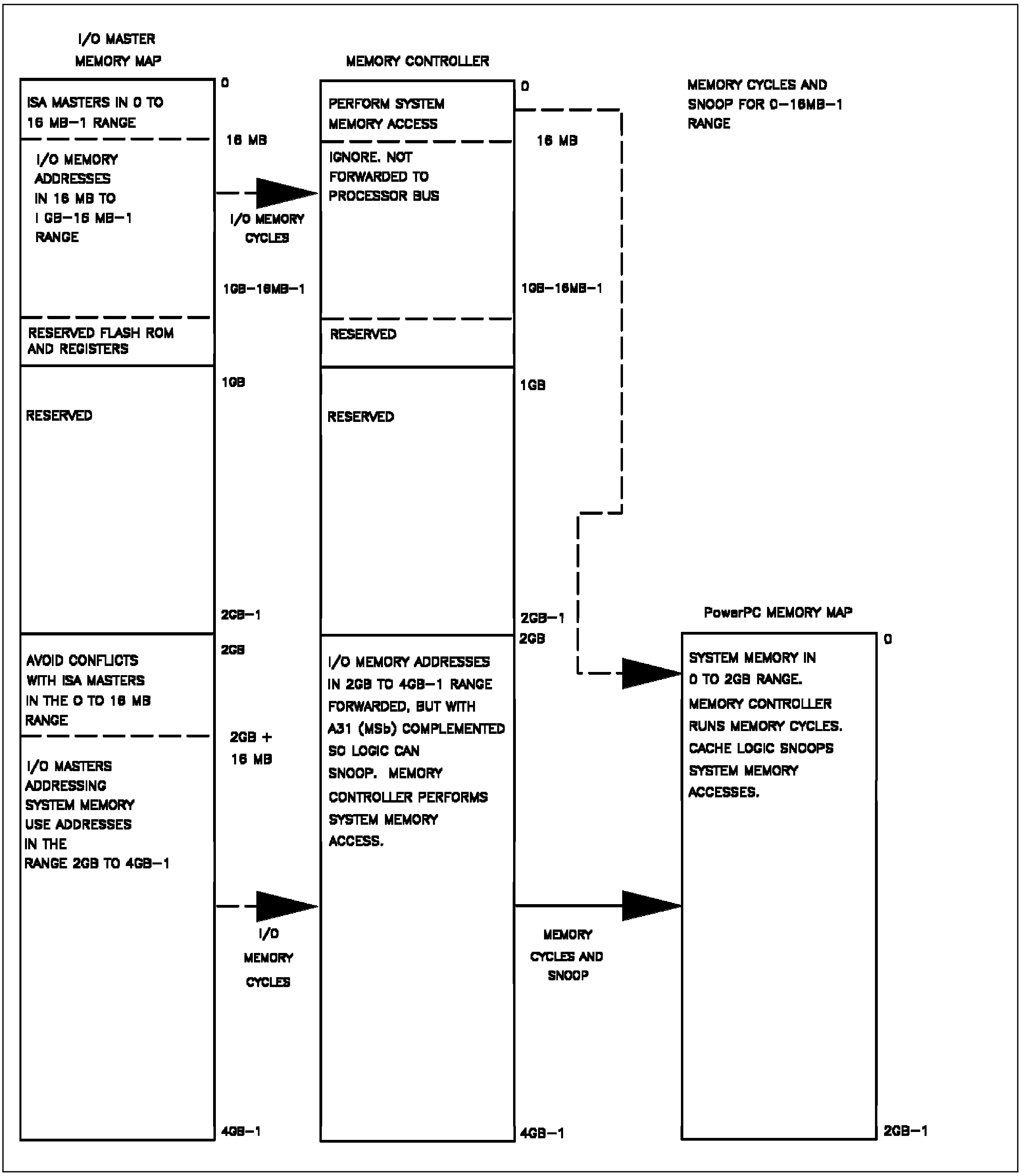

Figure 23. I/O Master View of Memory Map

/ the ISA devices in the System Memory range of 0 to 16 MB. In addition, low System Memory space reserved for interrupt vectors and used by some operating systems must be avoided by both PCI and ISA masters.

6.1.4 Rationale for Memory Model

The reason this design was chosen was to avoid a hole in the System Memory space and to allow the System Memory space and I/O space to expand without interference in each other's areas. In the current PC/AT-type memory map, I/O Memory located on I/O devices resides at an address between X'A0000' through X'DFFFF'. Without the memory remapping shown in this design, there would be a hole in System Memory. This hole would be used for the I/O devices. In addition, a second storage area for video adaptors and System ROM would have to be assigned either at the top of System Memory or as a continuation of the I/O area. If I/O space was required to be increased beyond the current 64 KB identified by the Intel specification, then the hole in System Memory would grow and software would have to change to accommodate this change. If video and ROM storage were part of this area, they would contribute to the likelihood of this area growing. If video and ROM storage were above the System Memory and the System Memory grew because more than 32 bits of addressing are available, software would have to change to accommodate this growth. The decision was made to use the most significant bit in the address to distinguish I/O space from System Memory space. The hardware makes the translation based on this bit. Supporting this decision is the information which tells the start and length of each area. This information makes the areas relocatable and provides flexibility for future growth in the address space.

6.1.5 I/O Device Mapping

Table 14 gives the ISA addresses, corresponding Reference Implementation system addresses for both the contiguous I/O map and the discontiguous I/O map, and the functions at each address. Refer to device and chip manufacturer literature for documentation of the utilization of these ISA addresses. The contiguous I/O map has the 64 KB of I/O space mapped into a contiguous 64 KB of space starting at 2 GB. The discontiguous I/O map has this 64 KB of I/O space mapped into the first 8 MB above 2 GB. These two I/O maps are described in Section 6.1, "Memory and I/O Map." Within Table 14, several I/O address ranges are marked as "Do not use" for the discontiguous I/O map. These spaces would be in the same 32 bytes as another device, and if both devices were used, the protection that is enforced by the mapping of one device per page would not be possible.

Table 14 (Page 1 of 4). System I/O Address Map

Address in Hex			
	PowerPC Processor Map		
ISA Space	**Contiguous**	**Discontiguous**	**Function**
0000-000F	8000 0000-000F	8000 0000-000F	DMA 1 Registers and Control
0020-0021	8000 0020-0021	8000 1000-1001	Interrupt 1 Control and Mask
0040-0043	8000 0040-0043	8000 2000-2003	Timer 1
0060	8000 0060	8000 3000	Reset UBus IRQ 12 and Keyboard Chip Select
0061	8000 0061	8000 3001	NMI Status and Control
0062	8000 0062	8000 3002	Keyboard Reserved
0064	8000 0064	8000 3004	Keyboard Chip Select
0066-0067	8000 0066-0067	8000 3006-3007	Keyboard Reserved
0070	8000 0070	8000 3010	RTC Address and NMI Enable
0071	8000 0071	8000 3011	RTC Read/Write
0074	8000 0074	8000 3014	NVRAM Address STB 0

Table 14 (Page 2 of 4). System I/O Address Map

Address in Hex			
	PowerPC Processor Map		
ISA Space	Contiguous	Discontiguous	Function
0075	8000 0075	8000 3015	NVRAM Address STB 1
0076	8000 0076	8000 3016	Reserved for NVRAM
0077	8000 0077	8000 3017	NVRAM Data Port
0078-007C	8000 0078-007C	8000 3018-301C	Reserved
0080-008F	8000 0080-008F	8000 4000-400F	DMA Page Registers 0-7
0090	8000 0090	8000 4010	DMA Page Register Reserved
0092	8000 0092	8000 4012	Port 92 Register (Used for LE mode and Soft Reset)
0094-0096	8000 0094-0096	8000 4014-4016	DMA Page Register Reserved
0098	8000 0098	8000 4018	DMA Page Register Reserved
009C-009E	8000 009C-009E	8000 401C-401E	DMA Page Register Reserved
009F	8000 009F	8000 401F	DMA Low Page Register Refresh
00A0-00A1	00A0-00A1	8000 5000-5001	Interrupt 2 Control and Mask
00C0-00CF	8000 00C0-00CF	8000 6000-600F	DMA 2 Address Registers
00D0-00DF	8000 00D0-00DF	8000 6010-601F	DMA 2 Control Registers
00F0	8000 00F0	8000 7010	Coprocessor Error Register Reserved
0170-0177	8000 0170-0177	8000 B010-B017	Secondary Disk IDE
01F0-01F7	8000 01F0-01F7	8000 F010-F017	Primary Disk IDE
0220-0227	8000 0220-0227	8001 1000-1007*	Serial Port 3 (Secondary)
0228-022F	8000 0228-022F	8001 1008-100F*	Serial Port 4 (Secondary)
0238-023F	8000 0238-023F	8001 1018-101F	Serial Port 4
0278-027A	8000 0278-027A	8001 3018-301A	Parallel Port 3
027A-027F	8000 027A-027F	8001 301A-301F	Reserved Parallel Port 3
02E0-02E7	8000 02E0-02E7	8001 7000-7007*	Serial Port 4 (Tertiary)
02E8-02EF	8000 02E8-02EF	8001 7008-700F*	Serial Port 3 (Tertiary) or 4 (fourth choice)
02F8-02FF	8000 02F8-02FF	8001 7018-701F	Serial Port 2
0338-033F	8000 0338-033F	8001 9018-901F	Serial Port 3
0370-0371	8000 0370-0371	8001 B010-B011	Diskette Drive Control (Secondary) Reserved
0372	8000 0372	8001 B012	Diskette Drive Control (Secondary)
0373-0377	8000 0373-0377	8001 B013-B017	Diskette Drive Control (Secondary) Reserved and Secondary Disk IDE (376-7)
0378-037A	8000 0378-037A	Do not use	Parallel Port 2
037B-037F	8000 037B-037F	Do not use	Reserved Parallel Port 2

Table 14 (Page 3 of 4). System I/O Address Map

	Address in Hex			
		PowerPC Processor Map		
	ISA Space	Contiguous	Discontiguous	Function
	0398	8000 0398	8001 C018	Super I/O Index Address
	0399	8000 0399	8001 C019	Super I/O Data Address
/	03BC-03BE	8000 03BC-03BE	8001 D01C-D01E	Parallel Port 1
/	03BF	8000 03BF	8001 D01F	Reserved Parallel Port 1
	03E0-03E3	8000 03E0-03E3	8001 F000-F003	PCMCIA Carrier Card Setup
/	03E8-03EF	8000 03E8-03EF	Do not use	Serial Port 3 (fourth choice)
	03F0-03F1	8000 03F0-03F1	Do not use	Diskette Drive Control (Prime) Reserved
	03F2	8000 03F2	Do not use	Diskette Drive Control (Prime)
	03F3-03F7	8000 03F3-03F7	Do not use	Diskette Drive Control (Prime) Reserved and Primary Disk IDE (3F6-7)
	03F8-03FF	8000 03F8-03FF	8001 F018-F01F	Serial Port 1
	040B	8000 040B	8002 000B	DMA 1 Extended Mode Register
	0410-041F	8000 0410-041F	8002 0010-001F	DMA Scatter/Gather Command/Status
	0420-042F	8000 0420-042F	8002 1000-100F	DMA Scatter/Gather Descriptor (Ch 0-3)
	0434-043F	8000 0434-043F	8002 1014-101F	DMA Scatter/Gather Descriptor (Ch 5-7)
	0481-0483	8000 0481-0483	8002 4001-4003	DMA High Page Registers
	0487	8000 0487	8002 4007	DMA High Page Registers
	0489	8000 0489	8002 4009	DMA High Page Registers
	048A-048B	8000 048A-048B	8002 400A-400B	DMA High Page Registers
	04D6	8000 04D6	8002 6016	DMA 2 Extended Mode Register
	0800-0802	8000 0800-0802	8004 0000-0002	Reserved
	0803	8000 0803	8004 0003	SIMM ID (32/8 MB)
	0804	8000 0804	8004 0004	SIMM Presence
	0805-0807	8000 0805-0807	8004 0005-0007	Reserved
	0808	8000 0808	8004 0008	Hardfile Light Register
	0809-080B	8000 0809-080B	8004 0009-000B	Reserved
	080C	8000 080C	8004 000C	Equipment Present
	080D-080F	8000 080D-080F	8004 000D-000F	Reserved
	0810	8000 0810	8004 0010	Password Protect 1 Register
	0811	8000 0811	8004 0011	Reserved
	0812	8000 0812	8004 0012	Password Protect 2 Register
	0813	8000 0813	8004 0013	Reserved
	0814	8000 0814	8004 0014	L2 Invalidate

Table 14 (Page 4 of 4). System I/O Address Map

Address in Hex			
	PowerPC Processor Map		
ISA Space	Contiguous	Discontiguous	Function
0815-0817	8000 0815-0817	8004 0015-0017	Reserved
0818	8000 0818	8004 0018	Key Lock Position Register
0819-081B	8000 0819-081B	8004 0019-001B	Reserved
081C	8000 081C	8004 001C	System Control
081D-081F	8000 081D-081F	8004 001D-001F	Reserved
0820	8000 0820	8004 1000	Memory Controller Size Programming Register
0821	8000 0821	8004 1001	Memory Controller Timing Programming Register
0830	8000 0830	8004 1010	Audio Index Register
0831	8000 0831	8004 1011	Audio Indexed Data Register
0832	8000 0832	8004 1012	Audio Status Register
0833	8000 0833	8004 1013	Audio PIO Data Register
0840	8000 0840	8004 2000	Read Memory Parity Error
0842	8000 0842	8004 2002	Read Processor DPE Error
0843	8000 0843	8004 2003	Clear Processor DPE Error
0844	8000 0844	8004 2004	Read Illegal Transfer Error
0850	8000 0850	8004 2010	ISA I/O Map Type
0852	8000 0852	8004 2012	System Board Identification
1378-137D	8000 1378-137D	8009 B018-B01D	Parallel Port 4
15E8-15EA	8000 15E8-15EA	800A F008-F00A	Reserved
4100-4101	8000 4100-4101	8020 8000-8001	Reserved

Legend:

*	Do not use if conflicts with another used serial port

Software Development Note: The definition of some of these registers, particularly registers that are marked as reserved or those defined in the X'8xx' range, may change for other reference implementations. The address and content should be referenced abstractly by software systems.

Some system I/O addresses which are defined for this implementation are described in the following subsections. **For the registers defined in this subsection, the bit significance of these registers is given in Big-Endian (BE) form (the processor's viewpoint).** The naming convention used in the schematics and the vendor data books is Little-Endian (LE) Mode. Figure 24 shows the naming convention used in these subsections.

Within this section some bits are marked as reserved. Indeterminate data will be in these bits when the byte is read.

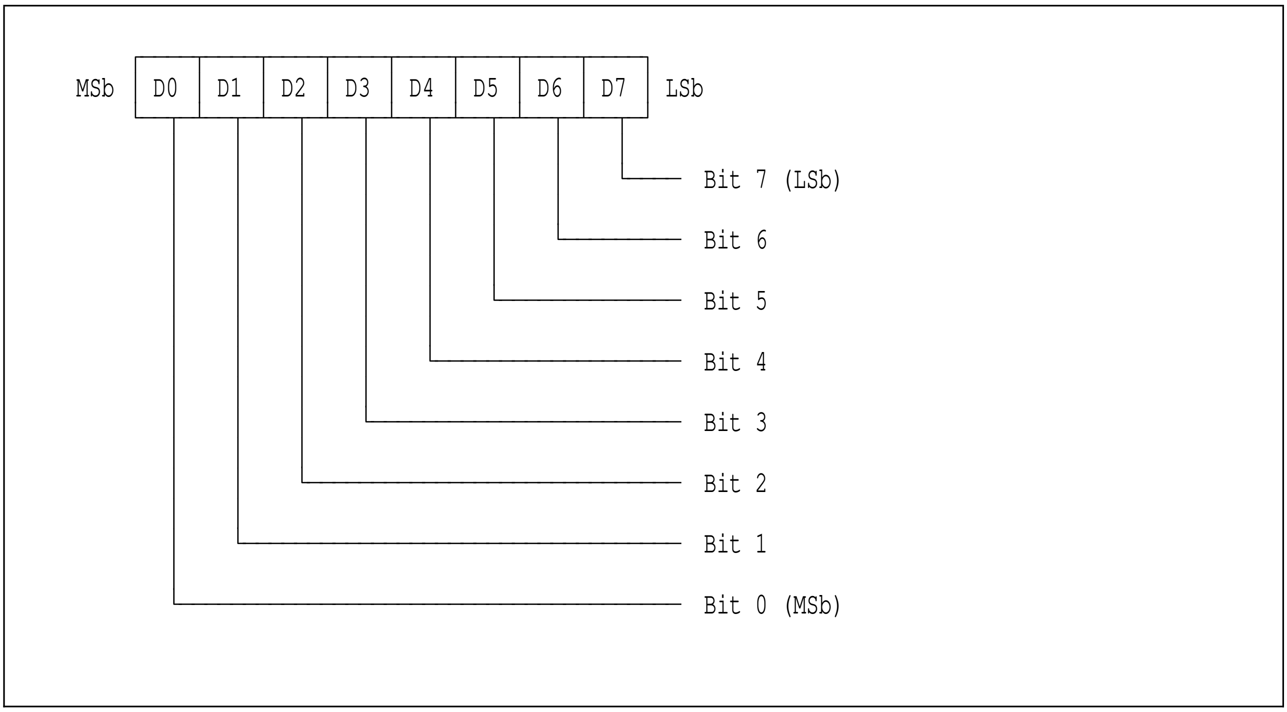

Figure 24. Register Bit Numbering

6.1.5.1 PORT 0092 -- Special Port 92 (Read/Write)

Bit 7: Soft Reset. Changing this bit from 0 to 1 will cause a system soft reset to occur. This bit must be returned to 0 before another soft reset can be issued.

Bit 6: LE Mode. Writing a 1 to this register sets the system board into LE Mode. Writing a 0 changes the system board into BE Mode. A store instruction can be used to change the Endian mode of the system. The PCI Bridge and Memory Controller (e.g. IBM 27 82650) will synchronize the completion of this mode switching before acknowledging the instruction. Software must insure that the switch is complete before information may reliably be transferred into the processor. The example in Figure 26 shows one way of confirming the completion of this switch.

Bits (0:5): Reserved.

6.1.5.2 PORT 0808 -- Hardfile Light Register (Write Only)

Bit 7: Hardfile Activity Light. This bit may be used to control the light that indicates hardfile activity.

0 = Light off.

1 = Light on.

Bits (0:6): Reserved.

This register’s initial state after reset is B'xxxx xxx0'. It is reset when port X'4D' in the SIO is utilized to reset the native I/O and ISA slots.

The Reference Implementation system board decodes the following range of addresses for this port: X'0808' - X'080B'.

6.1.5.3 PORT 080C -- Equipment Present Register (Read Only)

Bit 7: L2 Cache Present. This bit indicates whether or not an L2 cache is installed. The bit is 0 when a card in the upgrade slot grounds the correspondingly named signal at the upgrade connector; otherwise, it is 1.

0 = L2 cache installed.

1 = L2 cache absent.

Bit 6: Upgrade Present. This bit indicates whether or not an upgrade processor is installed. The bit is 0 when a card in the upgrade slot grounds the correspondingly named signal at the upgrade connector; otherwise, it is 1.

0 = Upgrade processor installed.

1 = Upgrade processor absent.

Bit 5: L2 256 KB. This bit indicates whether or not a 256-KB L2 cache is installed. The bit is 0 when a card in the upgrade slot grounds the correspondingly named signal at the upgrade connector; otherwise, it is 1.

0 = L2 cache is not 256 KB.

1 = L2 cache is 256 KB or is absent.

Bit 4: L2 is Copy-Back. This bit indicates whether or not a copy-back L2 cache is installed. The bit is 0 when a card in the upgrade slot grounds the correspondingly named signal at the upgrade connector; otherwise, it is 1.

0 = L2 Cache is write-through.

1 = L2 Cache is copy-back or is absent.

Bit 3: PCI Slot 1 Occupied. This bit indicates whether or not there is a card installed in PCI Slot No. 1. The bit is 0 when a card in the slot grounds the correspondingly named signal at the PCI connector; otherwise, it is 1.

0 = Card installed.

1 = Card absent.

Bit 2: PCI Slot 2 Occupied. This bit indicates whether or not there is a card installed in PCI Slot No. 2. The bit is 0 when a card in the slot grounds the correspondingly named signal at the PCI connector; otherwise, it is 1.

0 = Card installed.

1 = Card absent.

Bit 1: SCSI Fuse. This bit indicates the status of the SCSI fuse.

0 = Fuse bad.

1 = Fuse OK.

Bit 0: Reserved.

This register's initial state after reset is "As Populated."

The Reference Implementation system board decodes the following range of addresses for this port: X'080C' - X'080F'.

All of these bits are intended for static reporting. Dynamically changing the corresponding signals at the connectors may have undesirable effects on the operation of the system.

6.1.5.4 PORT 0810 -- Password Protect 1 Register (Write Only)

Bits (0:7): Any Value. Writing any value to this port sets a flip-flop which prevents any subsequent access to addresses X'20' - X'2F' of the Time Of Day Clock NVRAM address space. This flip-flop is cleared only on reset. A read has no effect.

This register's initial state after reset is in unprotected mode. It is reset when port X'4D' in the SIO is utilized to reset the native I/O and ISA slots.

The Reference Implementation system board decodes the following range of addresses for this port: X'0810' - X'0811'.

6.1.5.5 PORT 0812 -- Password Protect 2 Register (Write Only)

Bits (0:7): Any Value. Writing any value to this port sets a flip-flop which prevents any subsequent access to addresses X'30' - X'3F' of the Time Of Day NVRAM Clock address space. This flip-flop is cleared only on power-on reset. A read has no effect.

This register's initial state after reset is unprotected mode. It is reset when port X'4D' in the SIO is utilized to reset the native I/O and ISA slots.

The Reference Implementation system board decodes the following range of addresses for this port: X'0812' - X'0813'.

6.1.5.6 PORT 0814 -- L2 Invalidate Register (Write Only)

Bits (0:7): Any Value. Writing any value to these bits will cause the L2 cache to invalidate all its contents and immediately be in a state to begin caching again. The hardware creates a pulse on the L2_TAG_CLR# line going to the upgrade connector when this port is written. A read has no effect.

Programming Note: To guarantee correct operation of this function, there must be no L2 cache operation immediately following a write to this port. The correct sequence is to disable the cache with a write to Port 081C, invalidate the cache with this port, and enable the cache with a write to Port 081C. *Sync* instructions may be required to assure correct ordering of I/O cycles dealing with L2 cache control.

The Reference Implementation system board decodes the following range of addresses for this port: X'0814' - X'0817'.

6.1.5.7 PORT 0818 -- Reserved for Keylock (Read Only)

This register is reserved for systems which use the keylock function. The register is functional and will indicate the status of any signal connected to the keylock position of the front panel connector.

Bit 7: This bit gives real-time status of the signal wired to the keylock position of the front panel connector.

1 = Signal high.

0 = Signal low.

Bits (0:6): Reserved.

6.1.5.8 PORT 081C -- System Control Register (Read/Write)

Bits (4:7): Reserved.

Bit 3: Floppy drive motor inhibit. This bit will inhibit the start of the floppy drive motor and is used to eliminate pulsing the motor while doing a device select for media sensing.

0 = Motor signal allowed.

1 = Motor inhibited.

WARNING: D3 must be set to 0 for correct operation of the floppy drive. It defaults to 0.

Bit 2: Mask Transfer Error. This bit will mask any Transfer Error Acknowledge (TEA) signal going to the processor.

WARNING: Operating with TEA masked defeats all error checking in the system and could lead to lockup on incorrect code or other types of system faults.

0 = TEA masked.

1 = TEA not masked.

Bit 1: L2 Update Inhibit#. Setting this bit to 0 will disable the cache but will NOT invalidate the data tags. Also, no snoop operations are performed and no data updates are made while this bit is a 0. Setting it to a 1 will allow normal caching to resume. See bit 0.

Eieio instructions may have to be utilized to insure correct ordering of operations dealing with L2 cache control.

0 = L2 cache updating is disabled.

1 = L2 cache updating is enabled.

Bit 0: L2 Cache Miss Inhibit#. Setting this bit to 0 allows L2 cache misses to bypass the cache and not update the cache contents. This is useful for data movements that do not require cache updates when old instructions or data should be retained. When this bit is a 1, caching operations are normal. See bit 1.

Eieio instructions may have to be utilized to insure correct ordering of operations dealing with L2 cache control.

0 = Allows bypass.

1 = All L2 misses are updated.

This register's initial state after reset is B'000x xxxx'. It is reset when port X'4D' in the SIO is utilized to reset the native I/O and ISA slots.

The Reference Implementation system board decodes the following range of addresses for this port: X'081C' - X'081F'.

6.1.5.9 PORT 0850 -- I/O Map Type Register (Read/Write)

Bit 7: I/O Map. This bit controls the I/O address mode. Setting this bit to 1 allows a contiguously mapped mode wherein 601 cycles with addresses of 2 GB to 2 GB + 64 KB are contiguously mapped to PCI addresses 0 to 64 KB. Setting this bit to 0 causes discontiguous I/O mode as described in Section 6.1.1, "Processor View of the Memory Map."

0 = Contiguous I/O map mode (601 addresses ISA at 2 GB to 2 GB + 64 KB).

1 = Non-contiguous I/O map mode (601 addresses ISA I/O at 2 GB to 2 GB + 8 MB).

Bits (0:6): Reserved.

The Reference Implementation system board decodes the following range of addresses for this port: X'0850' - X'0853'.

This register's initial state after reset is B'xxxx xxx1' (contiguous mode). The register is reset when port X'4D' in the SIO is utilized to reset the native I/O and ISA slots.

6.1.6 System Interrupt Assignments

Table 15 gives the interrupt assignments for the system interrupts. The table gives the IRQ number, the priority, and the connections. The implementation uses an interrupt controller compatible with two 8259s cascaded through IRQ2. The controller is configured at boot time for interrupt 15 to be level sensitive and contain all the PCI interrupts.

Table 15. System Interrupt Assignments

IRQ	Priority	Connection
0 Master	1	Timer 1 counter 0 (internal to SIO chip)
1	2	Keyboard
2	3-10	Cascade from controller 2
3	11	Com 2, ISA pin B25
4	12	Com 1, ISA pin B24
5	13	ISA pin B23
6	14	Floppy, ISA pin B22
7	15	Parallel LPT 1, ISA pin B21
8 Slave	3	RTC
9	4	ISA pin B04
10	5	Audio, ISA pin D03
11	6	ISA pin D04
12	7	Mouse, ISA pin D05
13	8	SCSI
14	9	ISA pin D07
15	10	PCI interrupts (optional), ISA pin D06

6.1.7 I/O Configuration Space Mapping

Table 16 gives the PCI I/O configuration space mapping. Each I/O device may have up to 256 bytes assigned to configuration registers. The layout of these registers includes more address space than 256 bytes, but enables a simple hardware mapping. An address bit is directly connected to the PCI configuration line, ID SEL. Addresses beyond the limit of 256 bytes will cause more than one device to be selected. Within this table, address lines (i.e. A/D) are numbered in Little-Endian fashion (e.g. 0 is the LSb).

Table 16. I/O Configuration Registers

Device	ID SEL Line	Processor Address	PCI Address
SIO	A/D 11	8080 0800h-08FF	0080 0800h-08FF
SCSI	A/D 12	8080 1000h-10FF	0080 1000h-10FF
PCI expansion slot 1	A/D 13	8080 2000h-20FF	0080 2000h-20FF
PCI expansion slot 2	A/D 14	8080 4000h-40FF	0080 4000h-40FF
PCI expansion slot 3	A/D 15	8080 8000h-80FF	0080 8000h-80FF
PCI expansion slot 4	A/D 16	8081 0000h-00FF	0081 0000h-00FF
PCI expansion slot 5	A/D 17	8082 0000h-00FF	0082 0000h-00FF
PCI expansion slot 6	A/D 18	8084 0000h-00FF	0084 0000h-00FF
PCI expansion slot 7	A/D 19	8088 0000h-00FF	0088 0000h-00FF

6.1.8 Additional System I/O Mappings

Table 17 gives the Reference Implementation processor addresses and the functions of some registers in System I/O space. Four bytes of space are allocated for the interrupt acknowledgement. However, the current implementation need return only 1 byte.

Table 17. Registers in System I/O Space

Processor Address	Function
BFFF EFF0h-EFF3	Memory parity error address
BFFF FFF0h-FFF3	Interrupt vector register

The Memory Parity Addressing will return enough of the parity error address to identify the page in which the parity error exists.

6.1.9 I/O Memory Mapping

Table 18 gives the PCI addresses, corresponding processor addresses and the functions of some registers in I/O Memory.

Table 18. Register Map in I/O Memory

I/O Memory Address	Processor Address	Function
3FF0 0100	FFF0 0100	Starting address after hard reset
3FFF FFF0	FFFF FFF0	Flash write address and data
3FFF FFF1	FFFF FFF1	Flash lock out write

6.1.9.1 601-To-ROM Cache Fill Read Cycles (Special Burst)

At power-on time, the 601 comes up in Big-Endian Mode with its cache enabled and begins burst mode fetching at address X'FFF0 0100'. The system board logic, at this time, defaults to Big-Endian Mode and the bring-up code is read.

The ROM is located on the PCI bus physically, but not logically. This requires the PCI Bridge and Memory Controller (PCIB/MC) to decode ROM addresses, run eight cycles to the PCI bus, accumulate the 8 single bytes of read data into an 8-byte buffer line, and control responses to the 601 during the burst cycles.

This cycle is a highly specialized cycle that is not a true burst even though the 601 is in burst mode. It obeys the following rules:

- At the beginning of each cycle, the A/D (2:0) bits are set to zero regardless of the states of the 601_A (29:31) bits.
- Logic in the PCIB/MC causes ROM cycles to run at approximately 150 ns - 200 ns each to accumulate 8 bytes of data. A/D (2:0) are incremented on each cycle. Thus one 8-byte beat takes approximately 1.6 µsec. The timing is 16 CPU clocks on the first beat and 13 CPU clocks on each subsequent beat, with a one- or two-clock overhead after the last beat.
- After the first 8 bytes are accumulated, they are transmitted to the 601 as the first beat of the burst. The same data is repeated on the next three beats in rapid order without cycling the ROM. The reason for this is to prohibit cycles on the PowerPC processor bus which are long enough to interfere with refresh or cause extreme latency.

  Software Develpment Note: The initial stages of bringup must be programmed to account for this pattern of repetition. When enough useful instructions have been executed, the cache may be turned off so that it is not necessary to code the entire Flash in this manner.

- BE/LE data multiplexing follows the same rules as for normal memory, but in this case the low three address bits have no effect on the result. The bytes in Flash 0, 1, 2 ... are passed to 601 byte lanes 0, 1, 2... in BE mode, and they are swapped on byte lanes 7, 6, 5... in LE mode.

6.1.9.2 601-To-ROM Non-Burst Read Cycles

These cycles work very similarly to the special burst cycles described above. They obey basically the same rules. The difference is that they are naturally single-beat cycles, so data is not repeated.

- Any 601-supported-size transfer with any alignment may be used, and the cycle will complete as described in the previous section. Note that 8 bytes of data are always read from ROM.
- BE/LE data multiplexing follows the same rules as for normal memory, but in this case the low three address bits have no effect on the result. The bytes in ROM 0, 1, 2 ... are passed to 601 byte lanes 0, 1, 2 ... in BE mode, and they are swapped on byte lanes 7, 6, 5 ... in LE mode.
- The cycle time is approximately 1.6 µsec to 2 µsec whether 1 or up to 8 bytes are retrieved.

6.1.9.3 601-To-ROM Write Cycles (Flash-Based Implementation)

Writing to Flash ROM is another very specialized cycle. Only one address X'FFFF FFF0' can be used for writing all data to Flash. The Flash ROM address and data are both encoded into the 4 bytes of data written by this store operation. Only a word (4-byte) write cycle is supported (although only 1 byte is written at a time). The first 3 bytes contain the Flash address in order of low-order byte to high-order byte. The fourth byte (most significant byte in the word) contains the byte of data to be placed in the Flash ROM.

Flash ROM protection must be implemented within software. Port X'FFFF FFF1' can be used to lock out all Flash writes. Writing any data to this port address locks out all Flash ROM writes until the power is turned off and back on. In addition, the Flash itself has means to lock out updates to certain sectors by writing control sequences. Consult the Flash chip manufacturer’s specification for details.

6.1.9.4 PCI Masters To ROM

This implementation provides no mechanism for the PCI devices to access the ROM.

6.1.10 DMA Assignments

Table 19 gives the DMA request and grant lines. Details of the DMA controller implementation are provided in Section 6.3, "I/O Complex Components."

Table 19. System Interrupt Assignments

DMA Channel	DRQ	Connection
0	0	I/O connectors
1	1	I/O connectors
2	2	Super I/O, floppy diskette controller
3	3	I/O connectors
4	4	Cascade in SIO
5	5	I/O connectors
6	6	Audio connectors
7	7	Audio connectors

6.2 Processor Complex Components

This section describes components which could be used to implement the processor complex for a PowerPC Reference Platform-compliant system. Refer to Figure 18 for a diagram of the processor complex within the system configuration.

6.2.1 System Board

The system board contains most of the electronics for this Reference Implementation. Major I/O subsystems are connected to main memory through the PCI bus. The graphics subsystem, System I/O and SCSI are located on this bus. The video is attached to the PCI bus through a PCI socket, allowing for future upgradability and a fast interface. Flash ROM accesses share the PCI A/D lines with the control signalling provided by the memory controller.

The board is designed to an industry-standard LBX 9" by 13". It requires +5 volts to power most of the components. Plus 12 and -12 volts are also required to support some of the peripheral features. Components that require +3.6 volts (601 processor and PCI Bridge/Memory Controller) are supported using a system-board-mounted regulator to convert +5 to +3.6 volts. Negative 5 volts is supplied to the ISA connectors via the riser.

The system board will be available as a component from the IBM Microelectronics Division.

6.2.2 PowerPC Processor

The 601 processor (IBM PPC601 or Motorola MPC601) is a PowerPC processor available in operating frequencies of 50, 66 or 80 MHz. There are 32 address bits and 64 data bits. It contains a 32-KB eight-way set associated internal cache. The 601 is packaged in a 304-pin quad-flat-pack (QFP) package and requires +3.6-volt power.

6.2.3 PCI Bridge and Memory Controller (PCIB/MC)

This component, IBM 27 82650 (abbreviated as '650) is actually composed of two modules which act together. A buffer chip, IBM 27 82653, makes the connections between the processor, memory and PCI buses under the control of a control chip set, IBM 27 82654. IBM 27 82654 is implemented in a 160-pin chip. IBM 27 82653 is implemented in 304-pin Gate Array QFP chips.

The attributes of the PCI Bridge and Memory Controller are as follows:

- Supports 604 and 603 processors (except the 603 operation in 1:1 clock mode, which is not supported)
- SIMM memory controller
- Bridge from the PowerPC processor to the PCI bus
- Burst or single-beat access from CPU
- Burst or single-beat access from PCI
- PCI and PowerPC processor bus arbitration
- Error reporting to the PowerPC processor
- Supports only non-interleaved memory access operation (paging is supported)
- Memory on the PCI bus is not executable, nor cachable by the processor
- Memory on the ISA bus is not executable, nor cachable by the processor
- Supports Bi-Endian mode switching by performing the appropriate address modification and data multiplexing

The PCIB/MC chip set is available from IBM Microelectronics. Other chip vendors are working on similar solutions.

6.2.4 Memory Subsystem

The Reference Platform's memory subsystem can support up to 256 MB of system memory on eight SIMM sockets. Each SIMM socket can support an 8-MB or 32-MB parity-checking SIMM memory with access speeds less than or equal to 70 ns. These SIMMs are in an industry-standard 168-pin package. These SIMMs are an approved JEDEC standard for 8-byte SIMMs. The DRAM subsystem is 72 bits wide, which includes 64 data bits and 8 parity bits. One parity bit is generated for each byte of data written. During a read operation, one parity bit is checked for each byte of data.

The features and functions of the system memory are as follows:

- Supports parity checking
- Supports 8-MB and 32-MB SIMMs
- Supports mixed use of both types of SIMMs (8 MB and 32 MB)
- Supports empty SIMM sockets at any position

6.2.5 Clock Generation

The primary clock generation function is accomplished with an IBM clock chip. Sixty-six-MHz systems will use the IBM25JP-CLK01 clock chip. Eighty-MHz systems will use the IBM25JP-CLK03 clock chip. The chips are pin compatible and use a seed oscillator to generate the 2X, processor clocks, and 601 bus clocks needed by the system. One of the bus clock outputs feeds a Phase Lock Loop (PLL) clock generator which generates the PCI clocks. In the normal mode of operation the PLL divides the 601 bus clock by 2.

A second clock chip is used to generate clock signals for the PCI bus. The features and functions of the AMCC S4403 BiCMOS PLL Clock Generator are as follows:

- Generates eight clock outputs for PCI devices
- Forty-four-pin PLCC package

- Maintains +/- 1 ns synchronization to the 601 bus clock and any output clock
- Can be connected for divide-by-1 or divide-by-2 operation

6.2.6 Upgrade Slot

| A 2x101 Micro Channel-style card connector socket is provided on the system board in which either an L2 Cache card or a processor upgrade card may be installed. The 601 bus signals and various control signals such as presence detect signals and cache control signals are wired to the socket. Section 6.7, “Upgrade Slot Definition,” gives the list of pins and their functions on this connector. Support is included on the system board for:

- Write-through or copy-back cache cards
- Two cache-type ID bits in a system board register
- Various cache controls such as flush and disable
- Powering down the processor installed on the system board when an upgrade processor is detected. (This slot does not support multiprocessing.)
- Arbitration for the upgrade processor

The upgrade processor card must conform to the PowerPC (logical) bus definition. The primary upgrade target is the 604 processor. The upgrade processor card must provide a 3.6- or 3.3-volt regulator as required. Plus 5 volts is available at the socket.

6.2.6.1 L2 Cache

An L2 cache is part of the Reference Implementation. The cache is organized with a 72-bit data/parity path. The cache card will not check parity, but will store and retrieve parity generated elsewhere. The L2 cache for the Reference Implementation will have the following attributes:

- 256 KB or 512 KB
- Pluggable
- Direct-mapped
- 3-1-1-1 burst read hits
- Supports cycle times of 15 ns or greater
- 32-byte line
- Updates cache on bursts and single-beat write hits of exactly 8 bytes
- Invalidates line if single-beat write hit is of less than 8 bytes
- Qualifies cache updates with PowerPC Processor Cache Inhibit signal
- Will provide a minimum of 8 bytes of data on ANY cache read hit
- External L2_CACHE_INHibit Input from System Control Register
- External L2_CACHE_DISable Input from System Control Register
- External L2_CACHE_FLUSH Input from System Control Register
  - This will cause the L2 to invalidate all cache lines; no data will be written back to memory
- Supports storage of parity on data lines
- Caches only lower 2 GB address space (System Memory space)
- No address pipelining is supported

6.2.7 Flash ROM or EPROM

This component of the Reference Implementation contains the bring-up code. In addition, it is recommended that machine model-specific data such as the processor and system board bus speeds, native I/O complement, and memory map boundaries be programmed into this device. There is no other source of this information built into the system board. Programming this data into Flash ROM before or during the manufacturing process is possible.

After power on, the processor fetches the initial code from this device. A maximum of 16 MB ROM space is architected, but the system board supports 512 KB.

A jumper (J20) is provided on the system board which allows use of an EPROM (AMD 27C040-150JC) in place of the Flash ROM (AMD 29F040-120) for systems that do not wish to bear the cost of Flash ROM. Alternatively, a 256-KB EPROM may be utilized.

6.3 I/O Complex Components

This section describes components used to implement the I/O complex for the PowerPC Reference Implementation. Refer to Figure 18 for a diagram of the I/O complex within the system configuration.

6.3.1 Graphics Subsystem

The graphics subsystem consists of a PCI-attached Weitek 9000 and 1 MB of standard video RAM. The VRAM may be expanded to 2 MB. A Brooktree Bt_485 RAMDAC is used to convert digital red-green-blue (RGB) signals to analog video signals.

An alternate video system is the S3 86C928PCI attached to the PCI bus. It contains the same VRAM configuration and RAMDAC as the implementation using the Weitek.

6.3.2 SCSI Subsystem

The SCSI subsystem functions are performed by an NCR 53C810 chip. This component attaches directly to the PCI bus on the system board and is described in detail below:

- Eight-bit SCSI-2 interface
- Supports variable block size and scatter/gather data transfers
- Supports 32-bit word data bursts with variable burst lengths
- Full 32-bit PCI bus master
- Sixty-four-byte FIFO buffer

6.3.3 I/O Control Subsystem

The I/O Control Subsystem is implemented in two chips which are described in the next two subsections.

6.3.3.1 System I/O Bridge

The system I/O bridge function is provided by an Intel 82378ZB. This chip provides a PCI-to-ISA bus bridge where the native I/O and the ISA slots reside. It also provides system services such as DMA timers and interrupt control. Its major functions are as follows:

- Bridge between PCI and ISA
  - Eight- or 16-bit ISA devices
  - Twenty-four-bit addressing on ISA
  - Partially decodes Native I/O addresses
  - Passes unclaimed PCI memory addresses below 16 MB to ISA
  - Passes unclaimed PCI I/O addresses below 64 KB to ISA
  - Generates ISA clock, programmable divide by 3 and 4 ratios
  - Allows ISA mastering and has programmable decodes which map ISA memory cycles to the PCI bus
  - Has 32-bit posted memory write data buffer, no I/O buffering
- Seven-channel DMA between ISA devices and memory

  - Eight- or 16-bit devices on ISA (and optionally PCMCIA) bus only
  - Thirty-two-bit addressing of DMA
  - Function of two 83C37's
  - Eight-byte bidirectional buffer for DMA data
  - Supports Scatter-Gather DMA operations
  - Supports Guaranteed Access Time Mode for DMA and ISA masters
- Timer block -- function of 82C54
- Interrupt controller -- function of two 8259's
- Functions as PCI slave during programming and ISA slave cycles, as bus master during DMA (or ISA master cycles)

6.3.3.2 Native I/O Controller

This component contains the floppy disk controller, two serial ports, the IDE controller, and one parallel port. Control is provided via the ISA bus. This function is provided by a National PC87312 SUPER I/O chip.

Additional hardware and software may be used to facilitate the floppy disk media sense function. The National Super I/O controller chip will normally enable disk rotation anytime the media select line is activated. This pulse to the motor circuitry could lead to component failure and noise when software frequently (e.g. one hundred times a second) polls the floppy media. To eliminate the motor circuit pulse, software that intends to poll the floppy must set the motor inhibit bit in register 81C and hardware must inhibit this signal to the floppy motor circuit. A second approach that does not use hardware is possible for systems that have only one drive. In this approach, software which wishes to sense the media places the controller in four floppy mode and issues a media select for drive zero, but does not issue a motor select for drive zero. To avoid lighting the floppy access lamp, software floppy device drivers should set the drive select bits to B'11' when the floppy is not being used.

6.3.4 Audio Subsystem

Business audio is provided through the Crystal Semiconductor CS4231 stereo audio integrated circuit, which is connected to the ISA bus. The IC provides simultaneous capture and playback with two independently controllable DMA channels, each with its own 16-sample FIFO buffer. In addition, the audio chip provides compression and decompression and Big-Endian and Little-Endian sample modes. The audio subsystem is processor driven and does not include a digital signal processor (DSP). As such, it can play MIDI files, but is not a full-function MIDI system. One audio output is to a single speaker mounted in the cabinet. Also supported are four rear-mounted 3.5 mm jacks for:

- Stereo earphones (the speaker is muted when an earphone plug is inserted into the connector)
- Microphone input
- Stereo line in
- Stereo line out

There is also a system board-mounted connector with cable for direct playback from a CD-ROM.

6.3.4.1 Timer 2 Audio Support

The conventional PC Timer 2 signal is summed with the outputs of the business audio chip at the operational amplifier which drives the speaker. This provides the capability of supporting standard audio on configurations using this system board. Thus the software may elect to directly drive Timer 2 audio or to emulate Timer 2 audio through the business audio chip. Note that the use of the Timer 2 for audio is not recommended.

6.3.5 601 Processor Internal Real-Time Clock

The 601 processor requires a 7.8125-MHz clock input to support its internal Real-Time Clock. All other PowerPC processors contain a Time Base which is driven off of other clocks within the system.

6.3.6 Real-Time Clock (RTC)

Real-Time Clock and calendar functions are performed by a Dallas Semiconductor DS1385S chip. A description of this chip follows:

- RTC Function (PC compatible)
- Four KB Non-volatile RAM
- Connected to the X bus, which is a buffered 8-bit subset of the ISA bus
- Separate replaceable battery
- Sixty-four bytes of CMOS RAM

Figure 25 shows the map of information contained in the CMOS RAM contained on this chip.

6.3.7 Keyboard/Mouse -- Intel 8042AH

An Intel 8042AH chip provides keyboard and mouse controls and is connected to the X bus.

6.3.8 I/O Decoder

This component resides on the X bus. It receives partial decode signals from the SIO chip and further decodes these to produce chip selects for various components. It also contains most of the system registers and implements password protection.

6.4 Endian Switching Process

The process outlined in this section will switch the system from Big-Endian to Little-Endian mode when used in this Reference Implementation with a 601 processor. Because this process must execute during transition periods in which the processor and system components are in different Endian modes, care must be taken to assure that interrupts are not taken or that data and instructions do not remain in cache in the wrong Endian order. The instructions must not cross a page boundary. The process is outlined below:

a) Enable address translation
b) Flush all system caches
c) Disable interrupts
d) Enter supervisor state
e) Set the processor and system board state to Little-Endian (see Figure 26)

Note: Processor is now in Little-Endian mode. All instructions must be in Little-Endian order.

f) Put interrupt handlers and processor data structures in Little-Endian format
g) Enable interrupts
h) Start the Little-Endian operating system initialization

```
/* Structure map for CMOS on PowerPC Reference Platform */
/* CMOS is the 64 bytes of RAM in the DS1385 chip */
/* The CRC's are computed with x**16+x**12+x**5 + 1 polynomial */
/* The clock is kept in 24 hour BCD mode and should be set to UT(GMT) */

#ifndef _CMOS_
#define _CMOS_

struct _CMOS_MAP {
  unsigned char DateAndTime[14]; /* 00 = Seconds
                                    01 = Seconds Alarm
                                    02 = Minutes
                                    03 = Minutes Alarm
                                    04 = Hours
                                    05 = Hours Alarm
                                    06 = Day of Week
                                    07 = Day of Month
                                    08 = Month
                                    09 = Year (two digits)
                                    0A = Status Register A
                                    0B = Status Register B - Alarm
                                    0C = Status Register C - Flags
                                    0D = Status Register D - Battery */
  unsigned char SystemDependentArea1[2];
  unsigned char SystemDependentArea2[8];
  unsigned char FeatureByte0[1];
  unsigned char FeatureByte1[1]; /* 19 = PW Flag;
                                    attribute = write protect */
  unsigned char Century[1]; /* century byte in BCD, e.g. 0x19 currently */
  unsigned char FeatureByte3[1];
  unsigned char FeatureByte4[1];
  unsigned char FeatureByte5[1];
  unsigned char FeatureByte6[1];
  unsigned char FeatureByte7[1]; /* 1F = Alternate PW Flag;
                                    attribute = write protect */
  unsigned char BootPW[14]; /* Power-on password needed to boot system;
                               reset value = 0x00000000000000005a5a5a5a5a5a);
                               attribute = lock */
  unsigned char BootCrc[2]; /* CRC on BootPW */
  unsigned char ConfigPW[14]; /* Configuration Password needed to
                                 change configuration of system;
                                 reset value = 0x00000000000000005a5a5a5a5a5a);
                                 attribute = lock */
  unsigned char ConfigCrc[2]; /* CRC on ConfigPW */
  } CMOS_MAP;

#endif  //ndef _CMOS_
```

Figure 25. Map of CMOS on DS1385S

```
    x00  mfspr     R2,1008    ;Load the HID0 register
    x04  ori  R2,R2,LE_BIT    ;Set the Little-Endian bit in R2
    x08  sync
    x0C  sync
    x10  sync
    x14  mtspr     1008,R2    ;Set the processor into Little-Endian mode
;At least 3 sync instructions must precede and follow the above
;instruction because of processor design and pipelines.
    x18  sync
    x1C  sync
    x20  sync
    x24  stb  R5,0(R29) ;Set Endian mode of system board
;Register R5 has the data and R29 has the address for the Endian control port
;Endian control port at X'8000 0092' must be addressed at
;X'8000 0095' because processor is modifying addresses now
; To this point all instructions are in Big Endian Format
;Include a string of palindromic instructions to pass time until the system
;completes the switch.  Twenty five are suggested based on a 66 MHz processor
    x2C  addi R0,R1,0x138     ;Instructions which work LE or BE
                    ;This instruction generates 0x38010138
    x30  addi R0,R1,0x138
    x34  addi R0,R1,0x138
    ...
    xyy  addi R0,R1,0x138
;Start of Little Endian instructions
```

Figure 26. Instruction Stream to Switch Endian Modes

This switching process is only applicable for 601 processors in a design with a Bi-Endian memory as shown in Figure 56.

If the design uses the Big-Endian memory approach as shown in Figure 57, then this process must be modified. The Little-Endian portion of the operating system loader and any of the rest of the operating system which was brought into memory before the system was switched to Little-Endian mode does not reside in memory in the correct format. This information was brought into memory in the true Little-Endian format as it existed on the media. It must exist in memory in the “PowerPC Little-Endian” form. This form has the bytes reversed within each doubleword. After the system is in Little-Endian mode, further input from media of Little-Endian information will arrive in system memory in the correct form. Examples of approaches to accomplish this byte reversal are as follows:

- Before the processor switches to Little-Endian mode, load and store each Little-Endian doubleword doing a byte reversal in the process.
- Swap out the Little-Endian material while in the Big-Endian mode and swap it back in after the system is in Little-Endian mode and before the processor has been switched to Little-Endian mode. The instructions to perform the swap-in must have been byte reversed.
- During the install process, byte reverse each doubleword of Little-Endian information on the media. This format cannot be moved to an implementation with Bi-Endian memory.

6.5 Devices and Subsystems Used

The list of devices described below is intended to reflect the typical devices used in the PowerPC Reference Platform Implementation. No endorsement of the product or vendor is stated by this list. This list is a minimum set of devices requiring operating system support. These devices, adaptors, and subsystems help the Reference Implementation compete with other PCs and workstations. These system solutions establish the recommended implementation. Alternative devices may be used in compliant implementations, but any software-visible difference will have to be accounted for by the vendor in the abstraction software.

6.5.1 Storage Subsystems

Hard Disk Drives (SCSI-2)

Vendor	Description
Quantum	270-MB Hard Disk Drive
Conner	340-MB Hard Disk Drive
Conner or Maxtor	540-MB Hard Disk Drive
IBM SSD	1-GB Hard Disk Drive
IBM SSD	2-GB Hard Disk Drive

Floppy Disk Drives

Vendor	Description
Alps	1.44-MB 3.5" FDD
Alps	2.88-MB 3.5" FDD
Canon	1.22-MB 5.25" FDD

CD-ROM Drives (SCSI-2)

Vendor	Description
Toshiba	CD-ROM XA DS

Optical Disk Drives

Vendor	Description
IBM SSD	M-O Optical drive 3.5" 128 MB 40 ms

Tape Drives (SCSI)

Vendor	Description
HP	35470A DAT
HP	35480 DAT
WangDAT	3200
Wangtek	5525ES (QIC)
Wangtek	51000ES (QIC)
Tandberg	3820
Tandberg	4120

Disk Arrays (SCSI)

Vendor	Description
Microarray	RAID5 Disk Array
OASIS	RAID5 Disk Array

6.5.2 ISA Bus Subsystems

Communications

Vendor	**Description**
IBM	Token Ring Network 16/4 Adaptor
IBM	PS/VP 10BASE-T Ethernet Adaptor
IBM	3278/79 Emulation Adaptor
IBM	Enhanced 5250 Emulation Adaptor Kit
IBM	X.25 Adaptor
3COM	3Com EtherLink** ISA Adaptor
Intel	EtherExpress** 16 (ISA)
Novell	NE2000/3200 (ISA)
Ungermann-Bass	NIUPS
Standard MicroSystems	EtherCard** (ISA)

MultiMedia Adaptors

Vendor	**Description**
IBM	Mwave Adaptor
Creative Labs	Sound Blaster** Pro/16

Fax/Data Modems

Vendor	**Description**
Hayes	Modem 2400-14400
Intel	SatisFAXtion** Modem
Megahertz	Modems
Practical Peripherals	Modems
US Robotics	Modems

Printer/Plotter Devices

The following types of printers/plotters should be supported:

- EPSON Dot Matrix
- HP (PCL)
- HP (Plotter)
- PostScript** compatible

Scanners

Vendor	**Description**
HP	ScanJet** IIp/c
(Any)	Scanners with TWAIN-standard interfaces

6.5.3 Human Interface Subsystems

Display Options

The following types of displays are supported:

- CRTs capable of 1024x768 and 1280x1024
- LCDs capable of 640x480 and 1024x768

PS/2-Compatible Mouse

Vendor	Description
IBM	3 button
IBM	2 button

6.5.4 PCI Subsystems

Vendor	Description
S3	Graphics Adaptor
Weitek	Graphics Adaptor
IBM GTX	Graphics Adaptor

6.6 Base Configuration and Capacities

This section contains lists of components for three example desktop configurations as well as a list of components common to all three. These lists are given to provide a range of system configurations or models possible with the Reference Implementation.

- Common to all models
  - Power Supply
  - 1.44-MB Diskette Drive
  - 5.25" CD-ROM
  - System Board
  - Keyboard
  - Mouse
- Model 1
  - 8-MB SIMM
  - Hard drive -- 240 MB
  - Base Video (1-MB VRAM, PCI attached, 1024x768x8 bit)
- Model 2
  - 2 x 8-MB SIMM (16 MB)
  - Hard Drive -- 340 MB
  - Cache Card -- 256 KB
  - Base Video (1 MB, 1024x768x8 bit)
- Model 3
  - 32-MB SIMM
  - Hard Drive -- 540 MB
  - Cache Card -- 256 KB
  - Extended Video (2 MB, 1280x1024x8 bit)

6.7 Upgrade Slot Definition

The upgrade slot consists of a 2x101 Micro Channel-style card edge connector. The slot is designed to support a write-through L2 cache, a copy-back L2 cache, or an upgrade processor (currently targeted for a 604). If both an upgrade card and an L2 card are required, it is possible to design a short riser to accept both; however, the electrical performance of the local bus degrades due to loading and reflections if this is done. The Reference Implementation currently does not support both.

This section defines the detail for the upgrade slot of the Reference Implementation. The definition of the signals provided on the upgrade slot is enumerated. The protocol for both write-through (WT) and copy-back (CB) secondary caches is defined.

6.7.1 Upgrade Slot Signal Descriptions

The following table describes the functions of the signals included at the upgrade slot connector:

Table 20 (Page 1 of 5). Upgrade Connector Signal Definitions

Signal Name	Pin Type	Description
PROCESSOR BUS SIGNALS: The pin type is stated from the point of view of the option card for CB or WT cache cards. Upgrade cards drive all signals listed in this group.	--	See PowerPC processor-specific user's manuals
60X_A(0:31)	Bi	CPU address bus, 0=most significant bit The '650 bridge drives the 60X_Address lines during an I/O snoop operation. (CPU address parity is not supported)
60X_D(0:63)	Bi	CPU data bus, 0=most significant bit Note that the '650 bridge does not drive data on the 60X_Data lines during an I/O snoop operation.
60X_D_PARITY(0:7)	Bi	CPU data parity, 0=most significant bit Parity is present on these lines when the PowerPC processor provides data and when the memory provides data to the PowerPC processor. Cache devices must store and forward the parity or run the system with all error checking disabled. (Mask_TEA bit set in System Control Register, '081C'.) Note that the '650 bridge does not drive data on the 60X_Parity lines during an I/O snoop operation.
TT(0:4)	Bi	CPU transfer type During an I/O snoop the '650 bridge drives these lines to '01010'b for read and '00010'b for write.
TSIZ(0:2)	Bi	CPU transfer size During an I/O snoop the '650 bridge drives these lines to '100'b.
TC(0:1)	In	CPU transfer code During an I/O snoop these lines float.
CI#	In	CPU cache inhibit During an I/O snoop this line floats.
GBL#	In	CPU global This signal is pulled to ground through a 1-K resistor on the system board.

Table 20 (Page 2 of 5). Upgrade Connector Signal Definitions

Signal Name	Pin Type	Description
WT#	In	CPU write-through During an I/O snoop this line floats.
SHD#	Bi	CPU shared This signal is pulled high with a 10-K resistor on the system board.
TBST#	Bi	CPU transfer burst During an I/O snoop the ’650 Bridge drives this line to '1'b.
TS#	Bi	CPU transfer start The ’650 bridge drives this line active for one clock period to initiate an I/O snoop.
XATS#	Bi	CPU extended transfer start (PIO) (Not supported on system board.)
AACK#	Bi	CPU address acknowledge Cache cards who claim a cycle must pace the transfer according to the protocol in this document. If the card drives this signal active (low), it must drive it high for one clock before tri-stating the signal. The ’650 bridge drives this signal low for one clock period to terminate an I/O snoop cycle.
ARTRY#	Bi	CPU address retry Cache cards drive this signal low on an I/O snoop hit. See Section 6.7.2, “Protocol for Copy-Back and Write-Through Secondary Caches,” for details.
DRTRY#	Bi	CPU data retry This line is driven low whenever the ’650 bridge drives the TEA signal. During an I/O snoop the ’650 bridge does not drive this line.
TA#	Bi	CPU transfer acknowledge Cache devices who claim a cycle must pace the transfer according to the protocol in this document. If the card drives this signal active (low), it must drive it high for one clock before tri-stating the signal.
TEA#	In	CPU transfer error acknowledge If the ’650 detects a DPE parity error signal from the processor following a sequence in which an L2 cache provides data to the processor, it will activate the error protocol and drive “TEA.”
BR_60X#	In	CPU bus request

Table 20 (Page 3 of 5). Upgrade Connector Signal Definitions

Signal Name	Pin Type	Description
BG_60X#	In	CPU bus grant Cache cards may monitor this signal to determine if a cycle originates with the CPU or with the ’650 Bridge.
DPE#	In	CPU data parity error
SYSTEM BOARD INTERFACE SIGNALS: The pin types in this section are stated from the point of view of the option card.		
HRESET#	In	Hard Reset Output from system board; active low during power-on reset or when reset button is pressed. This signal should be wired to the HRESET# of upgrade processors.
BUS_CLK(0:2)	In	Bus clock Three copies of the bus clock signal. These run at 66 MHz in the Reference Implementation. The rising edge of each is within 0-800 ps after the rising edge of 2X_Clk.
FULL_SPEED	Out	Full speed Strapping this signal to ground will cause the system board to operate at half speed on the local bus. This may be used for debug purposes or to force the local bus to operate at one-half of the processor frequency. It may be necessary to change the configuration of the PCI clock generator if this signal is grounded. It has a pullup on the system board.
BR_L2#	Out	Bus request L2 This signal is wired to the ’650 Bridge to support copy-back L2 cache cards. Its priority is after the PowerPC processor. See Section 6.7.2.6, “Copy-Back L2 (CB L2) Protocol,” for details.
BG_L2#	In	Bus grant L2 This signal is wired to the ’650 Bridge to support copy-back L2 cache cards. See Section 6.7.2.6, “Copy-Back L2 (CB L2) Protocol,” for details.
L2_CACHE_PRESENT#	Out	L2 cache present This signal should be strapped to ground on any L2 cache card. Its state is reported in the equipment register '080C'. It also modifies the behavior of the ’650 Bridge chip set when it is low. It is not intended to be dynamically changed.

Table 20 (Page 4 of 5). Upgrade Connector Signal Definitions

Signal Name	Pin Type	Description
L2_CACHE_CB	Out	L2 cache copy-back This signal should be left unconnected for copy-back and strapped to ground for L2 cache cards that are write-through. Its state is reported in the equipment register '080C' for diagnostic or other software purposes. Its state does not modify the behavior of the ’650 Bridge chip set. It is not intended to be dynamically changed.
L2_CACHE_256K	Out	L2 cache 256 KB This signal is strapped to ground for L2 cache cards that are not 256 KB. Its state is reported in the equipment register '080C' for diagnostic or other software purposes. It does not modify the behavior of the ’650 Bridge chip set when it is low. It is not intended to be dynamically changed.
L2_CLAIM#	Out	L2 cache claim This signal is driven by an L2 cache card to indicate that it will provide or accept data. The ’650 Bridge defers the memory cycle to the L2 when this signal is activated according to the protocol set out in this document. It must be synchronous to the BUS_CLKs at the connector. The state of L2_CLAIM# while TS# is active and in the next clock cycle is immaterial. It is sampled in the second clock interval after the clock interval in which TS# is active and thereafter during the same bus transaction. Its state (high or low) must be maintained until AACK# is asserted by the option card or by the system processor.
L2_TAG_CLR#	In	L2 tag clear This signal is output from the register at port '0814'h. As long as this signal is low, the L2 cache should reset all tags to invalid and stay in a reset condition.
L2_UPDATE_INH#	In	L2 update inhibit This signal is output from bit 1 of the register at port '081C'h. As long as this signal is low, the L2 cache should disable all cache functions including snooping, but it should not invalidate any tags.
L2_MISS_INH#	In	L2 miss inhibit This signal is output from bit 0 of the register at port '081C'h. As long as this signal is low, the L2 cache should inhibit updating the cache contents on a miss.

Table 20 (Page 5 of 5). Upgrade Connector Signal Definitions

Signal Name	**Pin Type**	**Description**
UPGRADE_PROC_PRES#	Out	Upgrade processor present This signal should be strapped to ground on cards having an upgrade processor. A low on this signal causes the 601 processor to tri-state all outputs and enter a low-power mode. The system expects this to be a static signal and it may not be changed dynamically. The state of the bit is reported in the Equipment register at port '080C'.
BUS_CLK_SPEED(1:0)	Out	Bus clock speed These signals are used by the system board to pass information about the CPU bus speed to the upgrade card so that the upgrade card may select an appropriate clock multiplier. For example, if the bus speed indicated is 66 MHz and an upgrade processor is capable of 99-MHz operation, the card may select a 1.5x multiplier. If the processor were capable of 132 MHz, it would select a 2.0x multiplier. The coding is as follows: • 00 = 66 MHz • 01 = Not presently defined • 10 = Not presently defined • 11 = Not presently defined The Reference Implementation connects these pins to ground (00b). Cards which do not require bus clock speed information may tie these pins to ground.
INT_60X#	In	CPU Interrupt This signal originates at the ’650 bridge. It is included for upgrade processor support.
SRESET#	In	CPU soft reset This signal should be wired to the SRESET# pin of upgrade processors.
GROUND	--	There are 28 ground pins.
+5 VOLTS	--	There are 20 +5-volt pins. Note that devices requiring other supply voltages will require regulators. The current capacity at 0.5 amp per pin is approximately 10 amps. (The system power budget may be less.)
RESERVED	--	There are seven reserved signals at the connector. These are no-connects on the system board.

Figure 27 defines the timing relationships for the upgrade slot. The characters enclosed in brackets, < > , refer to the table values found in Table 21. The figure does not represent any particular operation.

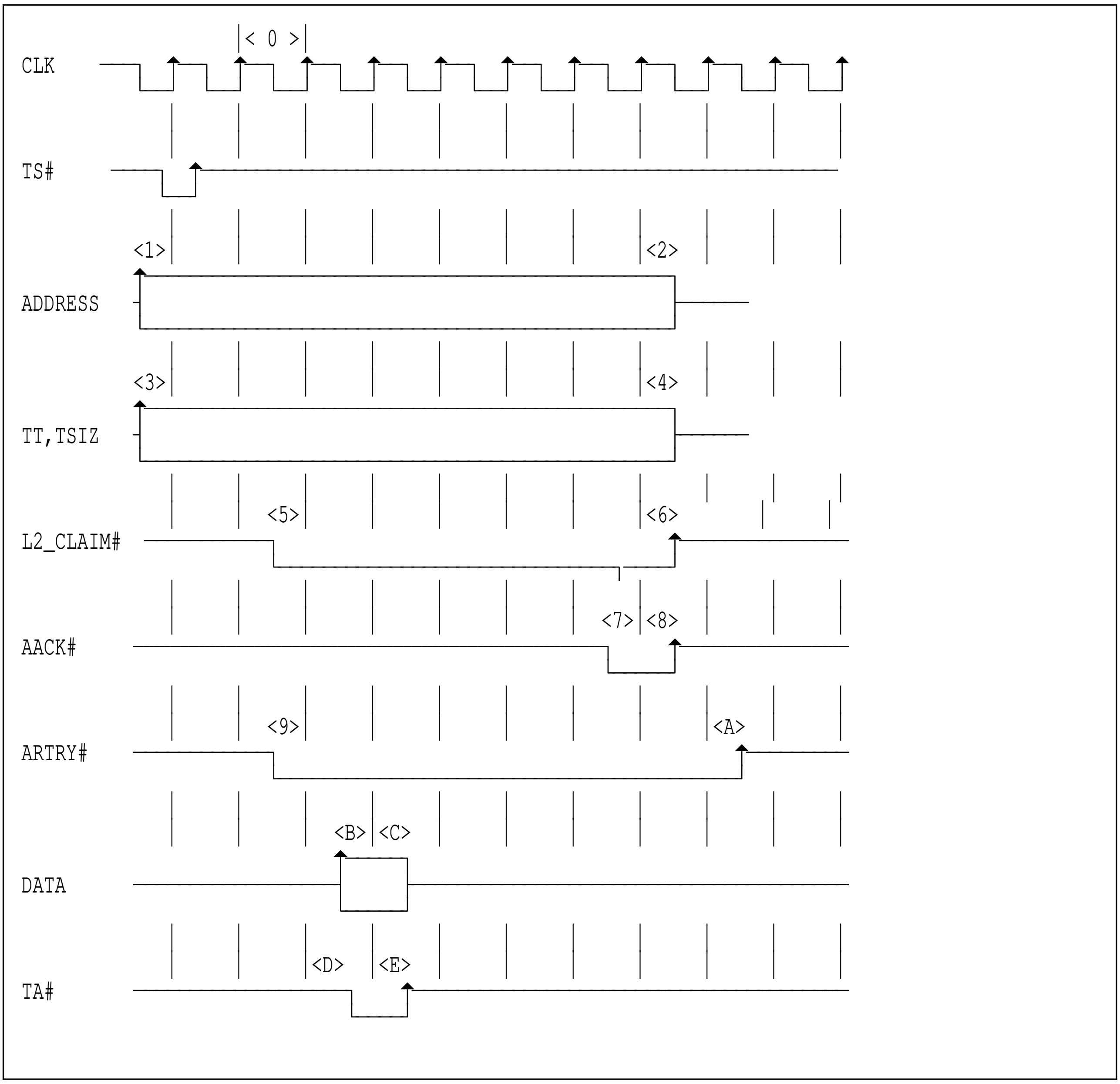

Figure 27. Upgrade Slot Synchronous Signal Timings

Table 21. Upgrade Slot Synchronous Signal Timing and Load Parameters

< >	SIGNAL	PARAMETER	RECEIVE (ns)	DRIVE (ns)	MAX LOAD (pf)	MAX LENGTH (inch)
0	BUS_CLK	CYCLE TIMING	n/a	15	13	1
0		DUTY CYCLE %	n/a	40/50%	--	--
1	60X_A(0:31)	SETUP	2	6	50/10	2
2		HOLD	0	0	--	--
3	TT,TSIZ	SETUP	2	6	50/10	2
4		HOLD	0	0	--	--
5	L2_CLAIM#	SETUP	n/a	6	n/a	n/a
6		HOLD	n/a	0	--	--
7	AACK#	SETUP	2	6	50/10	2
8		HOLD	0	0	n/a	--
9	ARTRY#	SETUP	2	6	50/10	2
A		HOLD	0	0	--	--
B	60X_D(0:63)	SETUP	2.5	6	50/10	2
B	60X_D_PARITY(0:7)	SETUP	2.5	6	50/10	2
C		HOLD	0	0	--	--
D	TA#	SETUP	2	6	50/10	2
E		HOLD	0	0	--	--

Note:

- In the Maximum Load column, the first number is load that is driven by the option card; the second number is maximum load allowed on the option card including wire and pins.
- All signals are synchronous with respect to the BUS_CLK.
- The figures in the Drive column indicate the setup and hold times that the option card must meet at the connector interface.
- The figures in the Receive column indicate the set up and hold times that the system board provides with the specified load.

Table 22 shows the timing and load requirements for the asynchronous L2 cache control signals at the upgrade slot. These signals are asynchronous with respect to the BUS_CLK. However, they are stable at least 40 ns prior to any TS#. In the Reference Implementation, they transition approximately 50 ns prior to the end of a memory-mapped 601-to-PCI I/O cycle. L2_TAG_CLR# is a pulse of approximately 500 ns length which returns to the high state approximately 40 ns before any possible TS#.

Table 22. L2 Control Signal Timing and Load Parameters

SIGNAL	SETUP (RE:TS#)	MIN PULSE	NOM PULSE	MAX LOAD	MAX LENGTH
L2_UPDATE_INH#	40 ns	S/W	S/W	50 pf	8"
L2_TAG_CLR#	40 ns	120 ns	500 ns	50 pf	8"
L2_MISS_INH#	40 ns	S/W	S/W	50 pf	8"

Note:

- MIN refers to the minimum pulse width allowable by the L2 design.
- S/W indicates that the actual pulse width is controlled by software because the signals are outputs of registers contained on the system board.
- The loads and line lengths are the maximum allowed for a device in the slot.

The Reference Implementation L2 cache design specifies that software must activate L2_UPDATE_INH# by writing to its control register, then activate L2_TAG_CLR# (which produces a pulse), and then deactivate L2_UPDATE_INH#. This gives approximately 500 ns minimum timing between edges as shown in Figure 28; however, to accommodate future enhancements, it is recommended that designers assume 120 ns minimum.

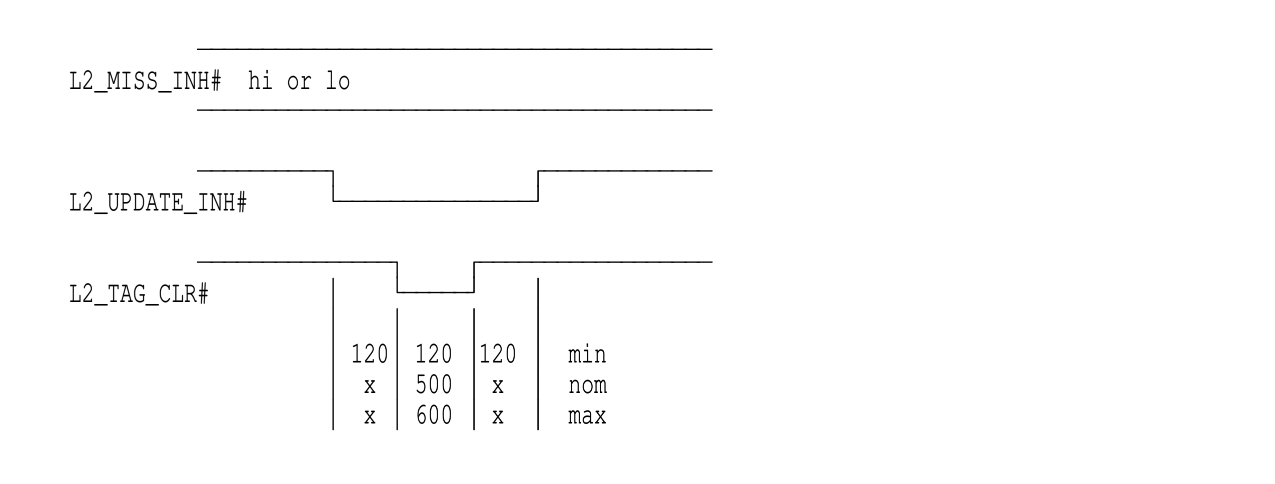

Figure 28. L2 Control Signal Timings

The following outputs from the upgrade card may be strapped to ground or not connected.

- L2_CACHE_PRES#
- L2_CACHE_CB
- L2_CACHE_256K
- L2_PROC_PRES#

6.7.2 Protocol for Copy-Back and Write-Through Secondary Caches

This section will describe the required protocol for any L2 designed to work in the upgrade slot of this system. The bridge chips used are the IBM27-82650 bridge chip set.

6.7.2.1 Bridge Chip Functions

The IBM 27 82650 Bridge chip set ('650) is responsible for system memory cycles as well as generation of I/O snoop cycles to maintain cache coherency in the L1 (PowerPC processor) and L2. The signal at the slot called L2_CACHE_PRESENT# must be continuously asserted (grounded) if the optional L2 is installed.

For PowerPC processor-to-memory cycles, the '650 will sample the L2_CLAIM# input the second clock after TS# is asserted by the processor. If L2_CLAIM# is not asserted and L2_CACHE_PRESENT# is asserted, the '650 Bridge will pace both the data and address tenures to the PowerPC processor. This means that the '650 Bridge will drive all handshaking signals (AACK#, ARTRY#, TA#, etc.) as well as provide the data from memory. The '650 Bridge will assert AACK# on the last data transfer cycle (be it a single-beat cycle or 4-beat burst), coincident with the last TA#. See Figure 29.

Note: The '650 Bridge will sample L2_CLAIM# only if L2_CACHE_PRESENT# is asserted.

On PCI-to-memory cycles, the '650 Bridge will master the PowerPC processor bus to drive the required snoop cycles to the processor and L2. Contrary to I/O PowerPC processor-mastered cycles, the '650 Bridge will not sample the L2_CLAIM# line. The '650 Bridge will drive AACK# active the clock after TS# is asserted, and sample ARTRY# the third clock. Also, it should be noted that the '650 does not drive data on to the 60X_Data lines during an I/O snoop.

The '650 Bridge will sample ARTRY# asserted the second clock after TS# to determine if there was a snoop hit in either the L1 or the L2. If there is a hit, the PCI device is signalled to retry and the PCI bus grant is removed.

Note: The '650 Bridge will NOT drive TA on to the PowerPC processor bus on I/O snoop cycles. It drives TBST# high. The transfer type encoding output during an I/O snoop will be:

PCI READ from Memory	TT(0:4) = 01010
PCI WRITE to Memory	TT(0:4) = 00010
	The '650 Bridge does not drive or sample TT4. This is pulled down on the system board.

If the '650 Bridge samples ARTRY# active on the second clock after TS#, it will drive ARTRY# inactive the second clock after AACK# is inactive, and then tri-state its buffer. This is required, since neither the L2 nor the PowerPC processor can restore this signal -- they both may be driving it, and one is a 3.x-volt part while the other may be a 5-volt part. Note that the '650 Bridge must be configured during the set-up process to enable this function.

The '650 Bridge will sample L2_BR# and 60X_BR# the clock after ARTRY# was asserted (third clock after TS#) by either one or both the PowerPC processor and L2. If only one is active, the '650 Bridge will grant the bus to that requester before granting the bus to another master (with the possible exception of refresh).

If, on the clock after ARTRY# was asserted to the '650 Bridge, both the PowerPC processor and L2 bus request lines are active, the '650 Bridge will grant the bus to the processor. If after the end of the cycle the L2_BR# is still active, then the '650 Bridge will grant the bus to the L2. However, the copy-back L2 will see a "write with kill" cycle when the processor pushes its data to the memory, and the CB L2 cache will invalidate or update its line and drop its bus request. The data from the processor is always more current than the L2 data.

The '650 Bridge will sample DPE# from the PowerPC processor two clocks after each TA# asserted by the L2 (i.e. L2_CLAIM# was asserted) to determine if the L2 is functioning correctly. If the '650 Bridge samples DPE# asserted, then TEA# will be asserted if the cycle is still in progress on the bus. Because of this function, all L2 caches must store and forward parity or the system must be operated with all error checking disabled.

6.7.2.2 General L2 Controller Requirements

L2 caches must manage coherency on a 32-byte line basis.

This protocol requires that L2_CLAIM# be valid the second clock after TS. This means that any cache designed for use in this system must be able to decode a cache hit within this time and assert this line. L2_CLAIM# may be asserted before the second clock cycle after TS, but must be valid on this clock cycle. It must be held active from the second clock cycle after TS# through the clock cycle in which AACK# is asserted.

If L2_CLAIM# is driven active as mentioned above, the L2 cache has claimed the cycle. This means it is responsible for pacing both the address phase (with AACK#) and the data phase (with TA#).

An L2 controller that must drive ARTRY# must be able to drive this signal on the third clock after TS# and must leave it asserted until the clock after AACK# is de-asserted.

The following are required of any L2 used in the system:

- Assert L2_CACHE_PRESENT# (tie low, static).
- If DISABLE_L2_CACHE# is asserted, the cache is essentially turned off.

  No snoops or updates are performed. Only two operations will be performed if the cache is in the disable state:

  a) Power-On Reset
  b) Reset Tags
- If L2_MISS_INH# is asserted, the cache performs I/O snoops only.

  No replacements are done in the cache on a miss. Cache responds to hits only by invalidation, or by providing the data.
- If L2_TAG_CLR# is asserted, the cache will reset itself and clear all data valid bits. No copy-back of dirty data is required.
- Determine if the cycle is a PowerPC processor-mastered cycle, or an I/O snoop cycle generated by the ’650 Bridge.
- Assert L2_CLAIM# the second clock after TS# is asserted on PowerPC processor-mastered cycles.
- Assert ARTRY# (if required) the second clock after TS# is asserted.
- Maintain coherency on 32-byte cache lines (sectors) granularity.
- Store data parity.

6.7.2.3 Determining I/O (PCI) Snoop Cycles

The CB L2 must be able to distinguish between a PowerPC processor cycle to or from memory and an I/O snoop cycle because no data is present on the processor data lines on an I/O snoop cycle. The CB L2 can determine the master of the current data tenure by sampling the BG_60X# on the clock before TS# is sampled active. This is required, since the arbiter can remove the PowerPC processor bus grant on the clock in which TS# is active.

6.7.2.4 Write-Through L2 (WT L2) Protocol

A WT L2 is used to provide the PowerPC processor faster read access to memory when any reads miss the L1. The L2 does not provide any performance improvement on cache cast-outs (PowerPC processor- to-memory writes).

Coherency is enforced by the I protocol. That is, data in the L2 can only be in one of two states:

a) Valid
b) Invalid

The data can only be changed from invalid to valid on burst reads or writes from the PowerPC processor. The WT L2 will sample the CI# output of the PowerPC processor. If CI# is asserted (CACHE_INHIBIT), no caching will occur on misses. If the data is already present in the WT L2, the state of the CI# input is ignored.

Data is invalidated for any of the following reasons:

- Any write of less than 8 bytes to a cached sector
- L2_TAG_CLR# is asserted

Since data is never stale or dirty in the WT L2, any hit in the cache to a valid sector will simply overwrite the previously stored data on a write cycle, and produce the data on a read cycle. This means that if the data is written by the PowerPC processor as a write with kill, the data can still be stored in the WT L2, but will be invalidated on an I/O snoop of a PCI write.

The WT L2 snoops all transactions on the PowerPC processor bus to maintain coherency, as well as to determine whether or not it can provide the data or should store the data.

The WT L2 ignores all address-only cycles (TT3=0). Note that ignoring address-only cycles does not violate the PowerPC architecture because in this implementation there is no cachable I/O memory.

On WT L2 cache hits (read only), the L2 will always assert L2_CLAIM# on the second clock cycle after TS# is sampled low. On reads, it will assert TA# the same clock as L2_CLAIM#, and on burst accesses, will assert a TA# on the next three clocks. See Figure 30, Figure 31, and Figure 32 for details.