FUJITSU

SPARC™ Memory Management Unit

MB86920

FEATURES:

- Compatible with the SPARC Reference MMU
- 32-bit virtual address, 36-bit physical address
- Fixed 4K-byte page size
- Support for sparse address spaces with 3-level map
- Support for large linear mappings (4K, 256K, 16M, 4G)
- Support for multiple contexts
- Page level protections
- Selective flushing and probing
- Hardware miss-processing
- Fully associative 64-entry page descriptor cache
- LRU replacement algorithm
- Translation cache

The Fujitsu SPARC MMU (MB86920) has been implemented as a single CMOS device, but due to its architecture, it is capable of accepting higher levels of integration or being embedded with the CPU in other fast technologies.

The MB86920 uses three levels of page tables to store translation information. Page table entries from main memory are cached inside the MB86920 to provide quick translations. The MB86920 provides system designers with a common architecture for memory management across a wide variety of systems. Support for large linear mappings with a single Page Table Entry, support for multiple contexts and 36-bit physical addresses provide improvements over typical MMU designs that will lead to significant performance advantages in many computer systems.

SPARC is a trademark of Sun Microsystems, Inc.

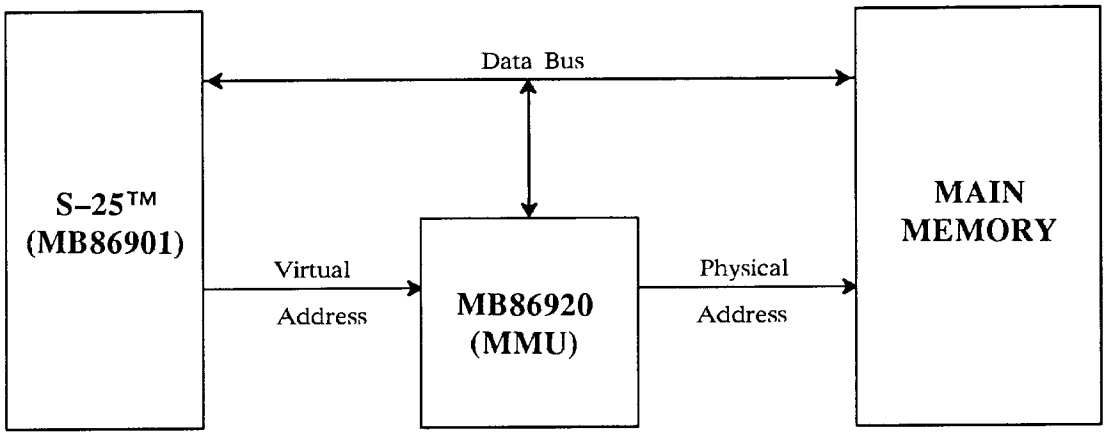


FIGURE 2.1 MB86920 IN SIMPLIFIED TYPICAL SYSTEM

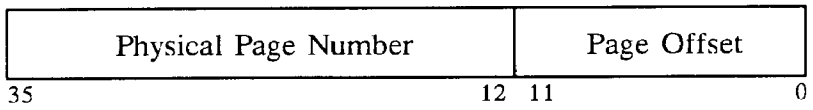


FIGURE 2.2 PHYSICAL ADDRESS FORMAT

1. Introduction

The Fujitsu SPARC MMU (MB86920) has been implemented as a single CMOS device, but due to its architecture, it is capable of accepting higher levels of integration or being embedded with the CPU in other fast technologies.

The MB86920 uses three levels of page tables to store translation information. Page table entries from main memory are cached inside the MB86920 to provide quick translations. The MB86920 provides system designers with a common architecture for memory management across a wide variety of systems. Support for large linear mappings with a single Page Table Entry, support for multiple contexts and 36-bit physical addresses provide improvements over typical MMU designs that will lead to significant performance advantages in many computer systems.

2. Software Architecture

The MB86920 provides three primary functions. First, the MB86920 provides address translation from the virtual addresses of each running process to physical addresses in main memory. This mapping is done in units of 4K-byte pages so that, for example, an 8-megabyte process would not need to be located in a contiguous section of main memory. Any virtual page can be mapped into any available physical page. Second, the MB86920 provides memory protection so that one process cannot accidentally read or write the address space of another process. This is necessary with most operating systems to allow multiple processes to safely run simultaneously. Third, the MB86920 is used to implement virtual memory. The MB86920 assists by determining which pages are already in main memory, and signaling a page fault if a memory reference occurs to a page not currently resident in physical memory.

S–25 is a trademark of Fujitsu Microelectronics, Inc.

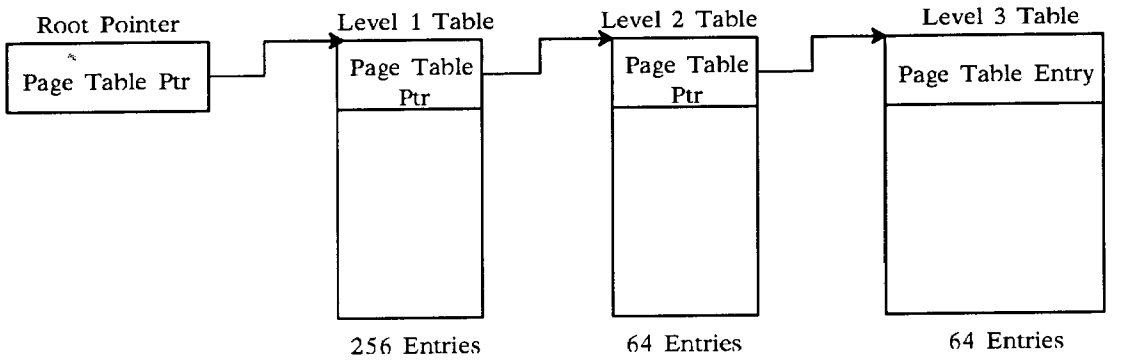


FIGURE 2.3 THREE LEVEL TABLE STRUCTURE

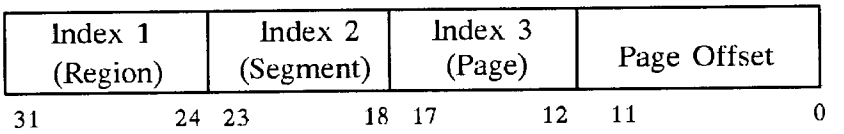


FIGURE 2.4 VIRTUAL ADDRESS FORMAT

The MB86920 translates virtual addresses from the CPU into physical addresses, as shown in Figure 2.1. A 32-bit virtual address is translated into a 36-bit physical address, providing for a 64-gigabyte physical address space. This large physical address space is intended to support large physical memories, and also to allow mapping of 32-bit buses (e.g. VME, MultiBus II, etc.). As shown in Figure 2.2, the physical address is logically composed of an offset in the 4K-byte page and a Physical Page Number.

Pages are always aligned on 4K-byte boundaries, hence the low order 12 bits of a physical address are always the same as the low order 12 bits of the virtual address, and do not require translation. For every virtual page there is a corresponding Page Table Entry that contains the physical page number for that virtual page. Translating from a virtual address to a physical address consists primarily of replacing the virtual page number with the physical page number.

All the address translation information required by the MB86920 is resident in physically addressed data structures in memory. The MB86920 fetches translations from these data structures, as required, by accessing main memory directly. Mapping of the virtual address space is accomplished by three levels of page tables in order to support sparse addressing efficiently. The first and second levels of these tables typically contain pointers to the next level and are called Page Table Pointers. Ultimately a Page Table Pointer will point to a Page Table Entry. A representation of the full three levels of mapping is shown in Figure 2.3.

The virtual address is divided into fields as shown in Figure 2.4.

Each of the three index fields is used to provide the offset into the corresponding level of page table. A full set of tables is rarely required, as the virtual address space is often sparsely populated. In some cases, the full three levels of page tables are not required to obtain the Page Table Entry. This happens when a 256K, 16M or 4G-byte section of linear memory is mapped with a single Page Table Entry. See the description of the Page Table Entry that follows for details.

CPU memory references would be too slow if each one required following the three levels of page tables in main memory in order to translate a virtual address to a physical address. Fortunately the Page Table Entries can be cached inside the MB86920 in what is commonly called an Address Translation

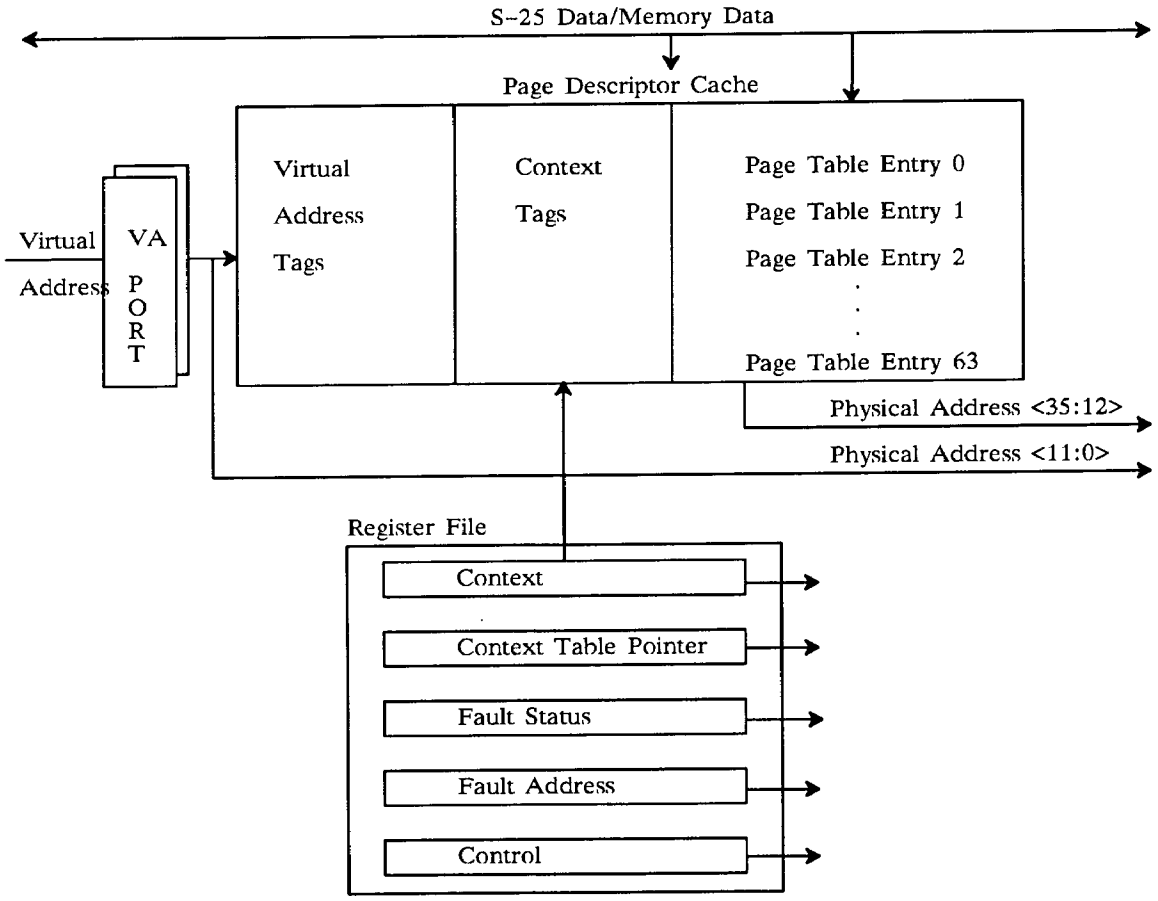


FIGURE 2.5 MB86920 CONCEPTUAL BLOCK DIAGRAM

Buffer, Translation Lookaside Buffer or Page Descriptor Cache (PDC). These cached entries provide quick translations without having to fetch the translation information from main memory.

Figure 2.5 shows a conceptual block diagram of the major components of the MB86920 implementation. Virtual addresses come in and are latched in internal registers. The virtual address is then compared against the set of virtual address tags stored on-chip in the PDC. A match against one of the tags indicates that the correct Page Table Entry is already stored on-chip, and the physical address is generated directly. If no match is made, then "miss-processing" occurs. Miss-processing means that the MB86920 automatically requests and takes over the physical address and data buses to fetch Page Table Pointers until it reaches the needed Page Table Entry; this process is also known as tablewalk. The Page Table Entry is then cached on-chip, translation occurs, and the original memory request continues from the latched address. Memory access permissions are also performed for every translation, and a fault will be generated if the permissions are incorrect.

2.1. Contexts

The MB86920 can retain translations for a number of active process address spaces at the same time. This is done to speed up context switches between different processes. Each active address space is identified by a "context" number. The context number may also be used by the system to maintain several active processes in a virtual cache. The management of the multiple contexts, including the assignment of contexts to processes, the reclamation of unused contexts and the reassignment of contexts, is the responsibility of the memory management software. Context numbers are used to

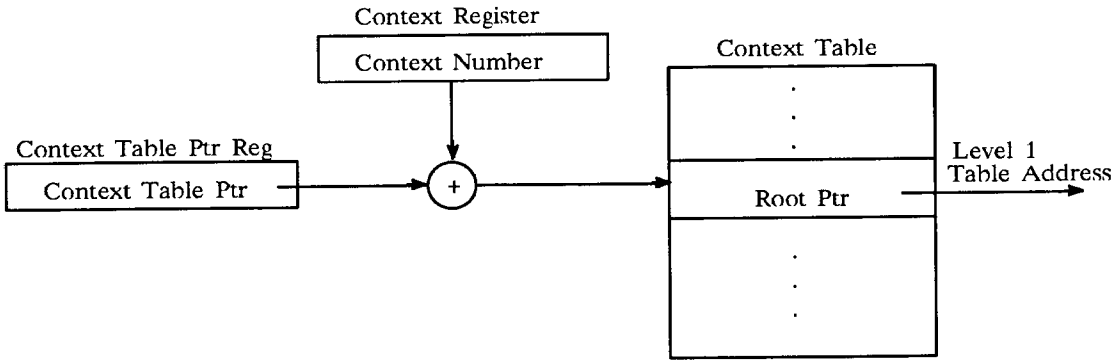


FIGURE 2.1.1 CONTEXT ADDRESS SPACE SELECTION

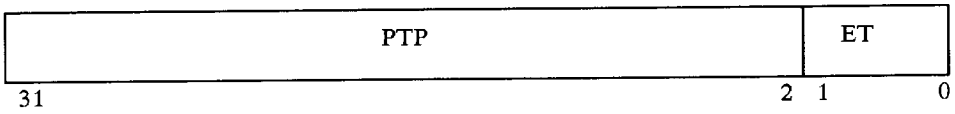


FIGURE 2.2.1 PAGE TABLE POINTER ENTRY FORMAT

identify the root of the page table hierarchy for a given process (see Figure 2.1.1).

At any one time only one of these address spaces is active. The current active address space is identified by its context number. This provides the offset in the context table used to retrieve the pointer to the page tables for the corresponding address space. The MB86920 supports 256 contexts. See the discussion of the Context Register.

2.2. Page Table Pointer

A Page Table Pointer (PTP) entry contains the physical address of a page table. It may be found in the Context Table or in the Level 1 or Level 2 page tables. A PTP is defined as shown in Figure 2.2.1.

Field Definitions:

PTP Page Table Pointer—Physical pointer to the base of the next level of translation tables. The PTP appears on bits 35 through 6 on the physical address bus during miss-processing. The page tables pointed to by PTP must be aligned on a boundary equal to the size of the page table. The sizes of the page tables of the different levels are summarized in Table 2.2.1.

ET Entry Type—This field must contain the value 1 (see Table 2.3.2).

Level	Size (bytes)
1	1024
2	256
3	256

TABLE 2.2.1 PAGE TABLE SIZES

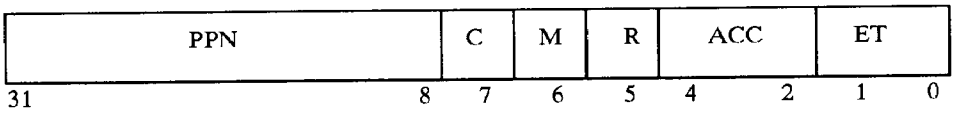


FIGURE 2.3.1 PAGE TABLE ENTRY FORMAT

2.3. Page Table Entry

A Page Table Entry (PTE) specifies both the physical address of a page and its access permissions. A PTE is defined as shown in Figure 2.3.1.

Field Definitions:

PPN Physical Page Number—The translated physical address of the page. The PPN appears on bits 35 to 12 of the physical address bus.

C Cacheable—If this bit is one, the page is cacheable by an instruction and/or data cache. Note that caching of instructions and/or data is not a function of the MB86920.

M Modified—This bit is set to one by the MB86920 when the page is accessed for writing.

R Referenced—This bit is set to one by the MB86920 when the page is accessed.

ACC Access Permission—These bits indicate whether access to this page is allowed for the transaction being performed. The Address Space identifier determines whether a given access is a data access or an instruction access, and whether the access is being done by the user or the supervisor. The field interpretations are shown in Table 2.3.1.

ET Entry Type—This field must contain a value of 0 or 2 (see Table 2.3.2).

ACC	Permissions	
	User	Supervisor
0	Read Only	Read Only
1	Read/Write	Read/Write
2	Read/Execute	Read/Execute
3	Read/Write/Execute	Read/Write/Execute
4	Execute Only	Execute Only
5	Read Only	Read/Write
6	No Access	Read/Execute
7	No Access	Read/Write/Execute

TABLE 2.3.1 ACCESS FIELD INTERPRETATIONS

ET	Entry Type
0	Invalid
1	Page Table Pointer
2	Page Table Entry
3	Reserved

TABLE 2.3.2 TABLE ENTRY TYPES

If a PTE is found in the Context Table or a level 1 or 2 page table, the address translation process is stopped and that PTE is used. Virtual addresses, starting from the first virtual address which references the PTE through the last virtual address which references the PTE, will be mapped linearly to physical addresses as specified by PPN. The physical address specified by PPN must be aligned on a boundary equal to the size of the area mapped by the PTE.

The sizes of the areas mapped by different levels in the page tables are summarized in Table 2.3.3.

Level	Mapping Size
3	4 Kilobytes
2	256 Kilobytes
1	16 Megabytes
Root	4 Gigabytes

TABLE 2.3.3 PAGE TABLE LEVELS MAPPING SIZE

Note: A Page Table Entry with an ACC code of 6 or 7 is considered a supervisor page. A Page Descriptor Cache of PTEs ignores the context number used to fetch the PTE when matching address tags for supervisor pages.

Note: Any or all PDC entries may be for a non-level 3 PTE. This provides large linear mappings for buses, coprocessors and kernels without requiring many translation cache entries because the MB86920 enters into the PDC the table's level as well. It then uses the level information to mask the lower portions of the virtual address not required for matching. The software should be careful with lower level PTEs since an address corresponding to a level 0 (Root) PTE mapping potentially could also match other non-level 0 PDC entries.

2.4. MB86920 Flush/Probe Model

The means of flushing an entry from the MB86920 Page Descriptor Cache as well as executing an MB86920 "probe" is by the use of the alternate address spaces provided by the FUJITSU SPARC CPU.

The flush operation purges the object specified in the Type field from the PDC, along with all objects it points to, except that flushing of the entire current context does not purge PTEs accessible only to the supervisor (PTE[ACC] codes 6 or 7). When flushing entries which are only accessible to the supervisor, the context in which the entry was fetched is ignored. A flush operation is accomplished by executing a Store operation (STA instruction) to the appropriate address (see Figure 2.4.1 and Table 2.4.1). The data used in the Store is ignored.

A probe operation is accomplished by executing a Load operation (LDA instruction) to the appropriate address (see Figure 2.4.1 and Table 2.4.1). The data returned is the object specified by the Type field. The availability of data is signaled by the MB86920 to the processor through /MDS, as in register loads. If a probe fails due to an error or the required object does not exist, a zero is returned and the Fault Status Register(see section 3.5) is set accordingly to indicate the type of error.

In an alternate address space used for flushing/probing, an address is composed as shown in Figure 2.4.1.

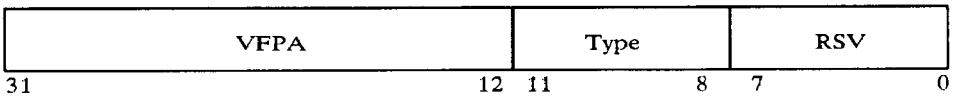


FIGURE 2.4.1 VIRTUAL ADDRESS FORMAT FOR FLUSHES AND PROBES

Type	Probe Action	Flush Action	Flush Compare Criterion
0	Probe Level 3	Flush Level 3 PTE(s) [Page]	(S+CTX)*VA[31:12]
1	Probe Level 2	Flush Level 2 & 3 PTEs [Segment]	(S+CTX)*VA[31:18]
2	Probe Level 1	Flush Level 1, 2 & 3 PTEs [Region]	(S+CTX)*VA[31:24]
3	Probe Context Table	Flush Level 0, 1, 2 & 3 PTEs [Context]	U*CTX
4	Probe PTE	Flush All PTEs	None
5-F	None	None	None

TABLE 2.4.1 TYPE FIELD INTERPRETATION

Field Definitions:

VFPA Virtual flush or probe address.

Type The Type field indicates what kind of flush operation on the PDC is desired, a specific page table entry (Level 3), some or all of the intermediate Levels table entries, an entire context, or the whole PDC. The Type field is encoded as shown in Table 2.4.1.

RSV The field is reserved and must be zero. It is ignored by the MB86920.

A flush operation requires that it meets a compare criterion (as shown above), where S (for supervisor), is equivalent to ACC codes of 6 or 7, U (for user), is equivalent to ACC codes of 0 through 5, and CTX is a comparison based on the context register. The MB86920 flushes only the entries specified by a flush operation.

Probe types three to zero cause the MB86920 to access the system tables where it then obtains and returns the PTE, the level 1/0 entry or the root pointer. An invalid Entry Type (ET) field corresponding to a searched entry does not cause a memory exception; it is simply read out and passed on to the CPU. If an illegal or reserved ET field is obtained before getting to the searched entry or a physical bus error exception (/PEXC) is received during the probe transaction, no memory exception is reported but the error is recorded in the Fault Status register. This type of failure causes a zero value to be returned. A type four probe returns a valid (ET=2) PTE entry from whichever level page table (0, 1, 2, or 3) that contains the translation for the VFPA field and the context register. A zero is returned if no valid PTE (ET=2) can be found in the tables or if there is an external bus error.

Virtual Address	Description	R/W
000000xx	Control Register	R/W
000001xx	Context Table Pointer Register	R/W
000002xx	Context Register	R/W
000003xx	Fault Status Register	R
000004xx	Fault Address Register	R
000005xx to 00000Fxx	Reserved	–
000010xx to FFFFFFxx	Unassigned	–

TABLE 3.1.1 REGISTER ADDRESS MAP

3. Hardware Architecture

3.1. Accessing MB86920 Registers

The MB86920 only requires five internal 32-bit registers. The Control Register contains general MB86920 control and status flags. The current process identifier is stored in the Context Register, and a pointer to the base of the context table in memory is stored in the Context Table Pointer Register. If a fault occurs, the address causing the fault is placed in the Fault Address Register and the cause of the fault can be determined from bits in the Fault Status Register. All the internal MB86920 registers can be accessed directly through peripheral accesses. The peripheral address map for the MB86920 is as shown in Table 3.1.1.

Note: The MB86920 is mapped via the CPU's Alternate Address Space Identifiers (ASIs) as described in the ASIs' pinout description.

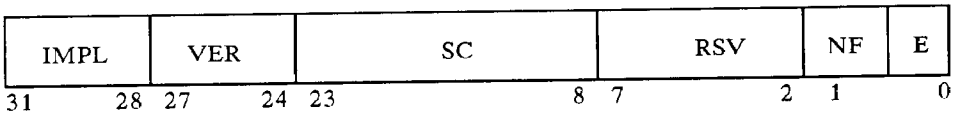


FIGURE 3.2.1 CONTROL REGISTER

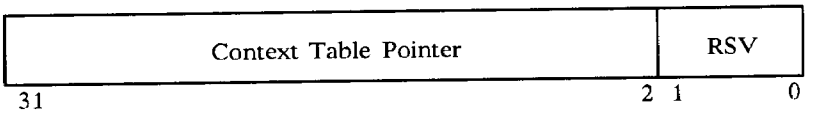


FIGURE 3.3.1 CONTEXT TABLE POINTER

3.2. Control Register

The MB86920 control register's format is shown in Figure 3.2.1.

IMPL This field identifies the implementation number of the MB86920. This field is hardwired, therefore, read-only and reads as 1.

VER This field identifies a particular Fujitsu MMU version. It is hardwired, therefore, read-only and reads as 0.

SC The MB86920 does not implement the System Control bits. The SC field reads as zero and writes to it are ignored.

RSV This field is reserved and must be written as zero.

NF The No Fault bit can be set to prevent supervisor data accesses from signaling a data fault to the CPU. If not set, all normal MB86920 processing continues with errors being signaled by setting the Fault Status Register (see section 3.5), if any.

E The E bit is used to turn on (enable) or turn off the MB86920 and is defined as:

E bit	MMU State
1	Enabled/On
0	Disabled/Off

When the MB86920 is disabled, all virtual addresses pass through the MB86920 untranslated and appear as physical addresses. In this case, the upper four bits of the physical address are zero. On MB86920 reset, the MB86920 is disabled.

3.3. Context Table Pointer Register

The Context Table Pointer Register format is defined as shown in Figure 3.3.1.

The Context Table Pointer points to a table in physical memory which contains level 1 Page Table Pointers. The context table is indexed by the contents of the Context register (see section 3.4). The Context Table Pointer appears on bits 35 through 6 of the physical address bus. The table pointed to by Context Table Pointer must be aligned on a boundary equal to the size of the table. The RSV field is reserved and must be zero.

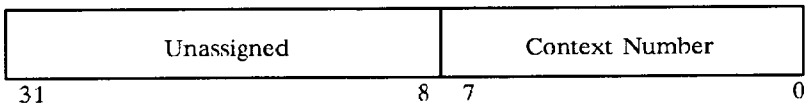


FIGURE 3.4.1 CONTEXT REGISTER

3.4. Context Register

The simple Context Register's format is shown in Figure 3.4.1.

The Context Register defines which of the possible process virtual address spaces is considered the "current" address space. Subsequent accesses to memory through the MB86920 are translated for the current address space. This continues until the Context Register is changed. When the Context register is written, the appropriate Root Pointer is not immediately retrieved from the Context Table; instead, the Root Pointer is retrieved during that context's first translation access. The unassigned field should be zero.

This MMU implementation allows for up to 256 contexts.

3.5. Fault Status Register

The Fault Status Register provides information on exceptions (faults) the MB86920 gives to the CPU. Since the CPU is pipelined, several faults may occur before a trap is taken. The faults are grouped into three classes: instruction access faults, data access faults and translation table access faults. If another instruction access fault occurs before the fault status of a previous instruction access fault is read by the CPU, the MB86920 sets a bit in the Fault Status Register to indicate that it has been overwritten.

The MB86920 and CPU ensure that if multiple data access faults occur, only the status of the one taken by the CPU is latched into the Fault Status Register. If data fault status overwrites previous instruction fault status the overwrite bit is cleared, since the fault status is represented exactly. Translation table access faults occur if MB86920 page table access causes external system errors.

An instruction access fault does not overwrite a data access fault. If a translation table access fault overwrites previous instruction or data access faults the overwrite bit is cleared. An instruction or data access fault does not overwrite a translation table access fault. The overwriting of multiple faults is summarized in Table 3.5.1, the value of OW represents the overwrite-bit content after the overwriting has taken place.

The MB86920's Fault Status Register is defined as shown in Figure 3.5.1.

RSV This field is reserved and must be zero.

EBE1 External Bus Error Field 1: This field tracks the EBE[3:0] inputs by accumulating the pin values, high order to low order respectively, from the last read to the Fault Status Register. The reading of the Fault Status Register clears this field. (See Example Table 3.5.7.)

EBE2 External Bus Error Field 2: This field tracks the EBE[3:0] inputs by accumulating the pin values, high order to low order respectively, from the last write and the next-to-last write to the Fault Status Register. The reading of the Fault Status Register clears this field. (See Example Table 3.5.7.)

L In the case of translation errors or invalid address errors, the Level field is set to the page talbe level of the entry which caused the fault. If an external bus error is encountered while fetching a PTE or PTP, the Level field records the page table level for the entry. The field is defined as shown in Table 3.5.2.

	FAULTS		
OVERWRITES	Trans	Data	Instr
Trans Faults	Y (OW=1)	N	N
Data Faults	Y (OW=0)	Y (OW=1)	N
Instr Faults	Y (OW=0)	Y (OW=0)	Y (OW=1)

TABLE 3.5.1 OVERWRITING OF MULTIPLE FAULTS

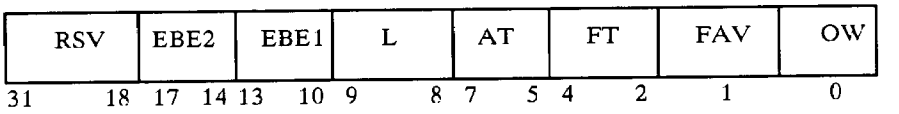


FIGURE 3.5.1 FAULT STATUS REGISTER

AT The Access Type field defines the type of access which caused the fault. It is defined as shown in Table 3.5.3. (Loads and stores to user/supervisor instruction space can be caused by load/store alternate instructions with ASI = 8-0xB.)

FT The Fault Type field defines the type of the current fault. It is defined in Table 3.5.4.

Invalid address errors, protection errors and privilege violations are a function of the Access Type and the ACC field of the corresponding PTE. The errors are set as shown in Table 3.5.5.

A translation error is indicated when an external bus error occurs while fetching an entry from a page table: a PTP is found in a level 3 page table, or a PTE or PTP contains unacceptable field values. The **L** field records the page table level at which the error occurred. The access bus error code is set when an external bus error occurs during memory access (that is not a page table walk access).

If a single access causes multiple errors, the fault type is recognized following the priority structure shown in Table 3.5.6 (from highest to lowest).

The highest priority fault is recorded in the Fault Type field. Reading the Fault Status Register clears it. Writes to the Fault Status Register are ignored.

FAV Fault Address Valid bit is set to one if the contents of the Fault Address Register are valid. The Fault Address Register may not be valid for instruction faults. The Fault Address Register is valid for data faults and translation errors.

OW The Overwrite bit is set to one if the Fault Status Register has been written more than once by faults of the same class since the last time it was read. If an instruction fault occurs and the OW bit is set, system software must determine the cause by probing the MB86920 and/or memory.

L	Level
0	Root pointer
1	Level 1 entry
2	Level 2 entry
3	Level 3 entry

TABLE 3.5.2 LEVEL FIELD WITHIN FAULT STATUS REGISTER

AT	Access Type
0	Load from User Data Space
1	Load from Supervisor Data Space
2	Load/Execute from User Instruction Space
3	Load/Execute from Supervisor Instruction Space
4	Store to User Data Space
5	Store to Supervisor Data Space
6	Store to User Instruction Space
7	Store to Supervisor Instruction Space

TABLE 3.5.3 ACCESS TYPE FIELD WITHIN FAULT STATUS REGISTER

FT	Fault Type
0	None
1	Invalid address error
2	Protection error
3	Privilege violation error
4	Translation error
5	Access bus error
6	Internal error
7	Reserved

TABLE 3.5.4 FAULT TYPE FIELD WITHIN FAULT STATUS REGISTER

AT	FT Code								
	PTE [ET] = 0	PTE [ET] = 3 *	PTE [ET] = 2, PTE [ACC] =						
			0	1	2	3	4	5	6	7
0	1	4	–	–	–	–	2	–	3	3
1	1	4	–	–	–	–	2	–	–	–
2	1	4	2	2	–	–	–	2	3	3
3	1	4	2	2	–	–	–	2	–	–
4	1	4	2	–	2	–	2	2	3	3
5	1	4	2	–	2	–	2	–	2	–
6	1	4	2	2	2	–	2	2	3	3
7	1	4	2	2	2	–	2	2	2	–

* Note: If a PTP (PTE[ET]=1) is found in level 3, a FT code of 4 is reported.

TABLE 3.5.5 FAULT TYPE CODES

Priority	Error
1	Internal error
2	Translation error
3	Invalid address error
4	Privilege violation
5	Protection error
6	Access bus error

TABLE 3.5.6 PRIORITY OF MULTIPLE ERRORS

EBE Pins				CLK	EBE Fields							
0	1	2	3	R.E.	EBE1				EBE2			
0	0	0	0	n	0	0	0	0	0	0	0	0
0	1	0	1	n+1	0	1	0	1	0	1	0	1
0	1	1	1	n+2	1	1	1	1	0	1	1	1
x	x	x	x		0	0	0	0	0	0	0	0

TABLE 3.5.7 EBE1/EBE2 EXAMPLE

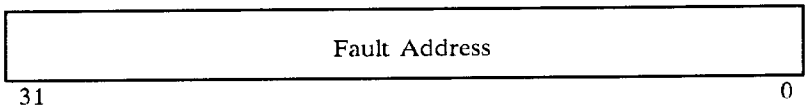


FIGURE 3.6.1 FAULT ADDRESS REGISTER

3.6. Fault Address Register

The Fault Address Register is defined as shown in Figure 3.6.1.

The Fault Address Register contains the virtual memory address of the fault recorded in the Fault Status Register. Fault addresses are overwritten using the same priority as for the Fault Status Register. Writes to the Fault Address Register are ignored.

Note: The MB86920 latches the full address.

4. Referenced and Modified Bit Update

A successful translation, of any kind, results in the Referenced bit in the Page Table Entry being checked. If the Referenced bit (R) is zero, the MB86920 sets the Referenced bit of the Page Table Entry in memory to one. The MB86920 does not keep an R bit on-chip.

A successful translation of a write operation results in the Modified bit being checked. If the Modified bit (M) is zero, the MB86920 sets it to one in both the cached PTE and the PTE in system memory.

Note: The MB86920 provides signals such that updating the Referenced or Modified bit is atomic with respect to other system accesses to the page tables. In addition, it is synchronous to the access which caused the update.

ASI*	TRANSACTION TYPE
03	MMU Flush/Probe
04	MMU Register Access
05-07	Reserved[1]
08	User Instruction Space
09	Supervisor Instruction Space
0A	User Data Space
0B	Supervisor Data Space
2x	ASI Passthrough

**Note: With any values for ASIs, the MMU simply passes VA onto PA with the upper 4 bits of PA equal to zero. Also TV gets asserted.*

TABLE 5.1 MEMORY TRANSACTIONS RECOGNIZED by the MMU

5. MB86920 Transactions with the S-25

The MB86920 uses ASI[7-0] to classify transactions (refer to Table 5.1) as MB86920 peripheral transactions, other peripheral transactions, or memory transactions. Memory transactions are further classified during the ASI as supervisor/user and instruction/data. The MB86920 uses RD to distinguish store operations from load and fetch operations. The signal NULLCYC causes the transaction to be ignored.

Note(s): 1. Accesses with an ASI of five, six or seven result in a virtual passthrough cycle (refer to chapter 9).

MB86920 peripheral reads are handled much like cache misses. Load transactions (MB86920 to CPU), have one hold cycle. In this cycle, the MB86920 asserts **M**emory **HOLD** (/MHOLD), **M**emory **D**ata **S**trobe (/MDS), and supplies the data. Store transactions are just like memory stores unless the context register is written; then /MHOLD is asserted until the MB86920 can fetch the new Root Pointer indexed by the new context register value.

If the MB86920 is off (E bit equals zero), the MB86920 passes the virtual address through to the physical address pins, with the upper physical address pins (PA[35:32]) equal to zero; the protection information becomes meaningless, high impedance signals come out in a 3-state condition, and control/handshake signals come out inactive.

If the MB86920 is turned on and the ASIs indicate a memory transaction, the MB86920 translates the address regardless of the level of /MMUCE. The MB86920 always translates memory references; however, if the MB86920 is disabled (/MMUCE negated), access checking is not performed and the tablewalk inhibited if a PDC miss occurs. Tablewalk and access checking are only performed when the MB86920 is enabled (/MMUCE asserted); however, if a PDC hit occurs, the physical address gets driven onto the PA bus validated through the assertion of TV. Early assertion of NULLCYC also prevents violation checking and table accesses.

On an address translation, the MB86920 samples the virtual address on the rising edge of the clock, the address is then presented to the **P**age **D**escriptor **C**ache (PDC). By the next rising edge the MB86920 asserts the translation address line (TV) if a hit occurs in the PDC. If a hit occurs, the proper physical address is driven onto the physical address bus. Also, for caches which perform their own access checking, the access privilege (ACC[2-0]), modified (MOD), and cacheable (C) bits from the page table entry are available. All of these bits are valid on the rising edge of the clock one cycle after the sampling of the virtual address.

If an access mode or privilege violation occurs, the MB86920 asserts /MHOLD, /MEXC, and /MDS on the cycle following the translation.

If the MB86920 does not have the page table entry for the virtual address in its cache, TV is not asserted. Instead, the MB86920 asserts /MHOLD, arbitrates for the physical bus and when granted, acquires the page table entry from memory. When the entry has been received, it is placed into the PDC. TV is then asserted, the PA bus driven with the translated address, the protection information (PI) made available, and the bus released. On the following cycle, /MHOLD gets negated and the CPU resumes its activities. During the fetching of the PTPs, TV does not get asserted.

6. MB86920 Transactions with Memory

The MB86920 requires access to memory to fetch or update page table entries. The access is obtained by a simple synchronous protocol.

Upon detecting the need for memory access, the MB86920 will assert **P**hysical **B**us **REQ**uest (PBREQ). This signal will be driven out of a register on the rising edge of the clock and will remain asserted as a lock on memory access until completion of the page table transaction sequence which may take as many as three reads and one write.

When the bus interface unit (BIU) determines that the memory access may proceed, it asserts **P**hysical **B**us **GRANT** (PBGRANT). This signal is sampled on every rising edge, begining on the rising edge following the assertion of PBREQ. PBGRANT needs to be maintained for only one cycle. The bus must be reserved for the MB86920 until PBREQ is unasserted.

The MB86920 first asserts **P**hysical **A**ddress **S**trobe (/PAS) on the cycle following assertion of PBGRANT. This strobe indicates a page table transaction on the Physical Address Bus is under way. /PAS is asserted only for the first cycle of each transaction. The address is valid during the first cycle.

The direction of data transfer is determined by **P**hysical **R**ead/**W**rite (PR/WN) which is valid upon assertion of /PAS.

The MB86920 then waits until **P**hysical **D**ata **S**trobe (/PDS) is asserted by the BIU. This is asserted when MB86920 bound data is valid on the Data Bus or when memory bound data need no longer be driven onto the Data Bus. On loads, the MB86920 is prepared to receive data on the cycle following /PAS. The data is sampled on the rising edge of the clock. On stores, the MB86920 begins driving the data bus on the cycle following /PAS. The data should be sampled on the rising edge of the clock.

Errors in the memory system may be reported by assertion of **P**hysical **EXC**eption (/PEXC). The signal is sampled when /PDS is asserted. Any non-zero value will cause the MB86920 to report a memory exception.

The MB86920 may require several memory accesses to complete the page table transaction. When it does, it continues to assert PBREQ to maintain ownership of the path to memory. Each access will be initiated by assertion of /PAS for one cycle. The physical address and data direction should be sampled during this cycle. Each access is terminated by assertion of /PDS.

7. MB86920 System Configurations

The MB86920 is designed to work in systems with virtual caches; however, due to the fast translation capability, it is ideal for systems utilizing physical caches. In addition, the versatility of the MB86920 allows it to operate on-line or off-line thus, tailored for most system environments.

7.1. MB86920 In Virtual Cache Systems

Figure 7.1.1 illustrates the MB86920 in a virtual cache system where the MB86920 is on-line i.e, the MB86920 and the cache receive the virtual address at the exact same time. This type of system is used when the cache controller leaves the access checking to the MB86920. In such designs, the MB86920 may halt the transaction even if the cache contains the fetched data but the PDC misses. Although this is an uncommon event, some system designers prefer to duplicate the permission information in the cache so that access rights be checked by the cache, and involve the MMU only on cache misses rather than on every access. In this type of system the MMU is considered off-line, as shown in Figure 7.1.2.

However, an off-line MMU needs means of providing the cache with the access rights when loading the leaf table entry after a PDC miss and cache miss. The MB86920 provides that information after having loaded the PTE during the time where the physical address is valid on the bus. Essentially, the MB86920 provides the cacheable bit, the modify bit, and the access permission bits to the cache every time the PA lines indicate translation. Note that when the MB86920 is off-line, it sees a delayed virtual address since it comes from the Cache Address Register (CAR), and not from the CPU.

7.2. MB86920 In Physical Cache Systems

Designers might choose to avoid virtual cache overheads associated with synonym detection by implementing instead a physical cache. It is conceivable to slightly slow down the system's clock rate to allow for slower SRAMs and yet build high performance, affordable physical cache systems. Figure 7.2 shows the MB86920 in a physical cache system.

8. MB86920 Architectural Description

The MB86920 is a 32-bit fully associative memory management machine implemented using Fujitsu's high speed 1.2 micron CMOS process technology. A simplified architectural block diagram is shown in Figure 8.1.

VA PORT: Incoming virtual addresses are latched within this port, then driven and directed to the CAM, Translation Cache, and Physical Address Generation blocks.

PDC: A specialized content addressable memory (CAM), 64 entries deep, determines whether the virtual address in the VA port matches one of 64 VA CAM-Tags. If so, a RAM containing permissions, protections and physical page numbers is enabled. The width of the mapping accommodates various address spaces: 4K, 256K, 16M and 4G bytes. The mapping however, does not stop at the address level; a context field in the CAM must also match the value in the context register. An internal state engine automatically loads CAM and RAM entries on misses.

TRANSLATION CACHE: Simultaneous to the CAM matching process, a two-entry cache containing partial translations is also accessed. Since address spaces are logically divided into instructions and data, and each space will mostly be in contiguous areas, two entries were designed-in. Most of the time, one entry contains instruction partial translations, while the other entry contains data partial translations. Partial translations get automatically loaded when a miss occurs in both the CAM and the translation cache (refer to Figure 8.2); as the PTPs are fetched and used to obtain the next table entry, they are also loaded into the translation cache. The least recently used entry gets replaced.

PHYSICAL ADDRESS GENERATION: This unit really has two purposes. One is address routing, the other is address generation. The unit routes and drives the physical address coming from the RAM during a PDC hit; as part of the operation, it concatenates the 12 bits of page offset with the 24 PPN bits. Similar routing, concatenation and driving process occurs when the address comes from the translation cache. This sub-block also routes the address in the VA port during passthrough mode.

The real address generation unit only merges address fields. Its sources are the table entry from the data port (PTPs and PTEs), the virtual address (indices), the context, and the context table pointer.

REGISTER FILE: The register file is composed of five 32-bit registers: control register, context table pointer register, context register, fault status register, and fault address register. The registers are used for three purposes:

1. By the processor to identify the type of MMU and to turn the MB86920 on or off at its convenience.

2. By the processor to inform the MB86920 the location in physical memory of the context table.

3. By the MB86920 to inform the processor of its status.

All five registers are accessed in user mode by setting ASIs to 04 Hex, and if the upper addresses are zero, as shown in section 3.1, VA[11:8] perform the register selection. Writes to read-only registers or accesses to unimplemented register address spaces, result in a null cycle i.e., a regular register access cycle as seen from the bus takes place (zero is driven out on reads and writes are ignored) with no memory exceptions reported.

ARBITER: The arbiter is intended to interface with an external bus controller capable of resolving simultaneous bus request conflicts e.g., from the cache controller and the MB86920. The arbiter requests the bus only when it does not have the physical bus, a PDC miss has taken place, and the part is enabled (through the E bit and /MMUCE). The MB86920 is ready to request the bus the cycle following the PDC miss and on the same clock's rising edge of the cycle in which the part requests the bus, it begins sampling the /PGRANT line. On the same cycle in which the grant is detected, the MB86920 drives the address for the table entry and places an address-strobe (for more detail refer to the PDC-miss-processing timings).

CONTROL: This sub-block controls the machine based on events occurring in the other units. On a PDC hit for example, it compares the ASIs with the RAM access bits (ACC) and determines if a violation is taking place or not. It then sends control signals to the appropriate units to either release the physical address and continue with regular handshaking or to enter a memory exception handshaking process. It also initiates the tablewalking sequence and monitors external events. This section also controls the internal bus traffic.

DATA PORT: The data port drives information into and out of the MB86920, into and out of the register file, into the translation cache, into the physical address unit, and into the PDC RAM. Note that the data port is used for both the physical data bus, and the logical data bus.

LRU LOGIC: Least recently used is the replacement algorithm by which valid entries in the PDC get overwritten by new entries when the PDC is fully loaded. Flush operations invalidate PDC entries but do not clear LRU history; therefore, invalid detected entries override the LRU replacement policy. Simulations have demonstrated that the LRU replacement algorithm combined with a 64-entry PDC results in a hit rate of over 99.9% within typical UNIX® environments.

UNIX is a registered trademark of AT&T

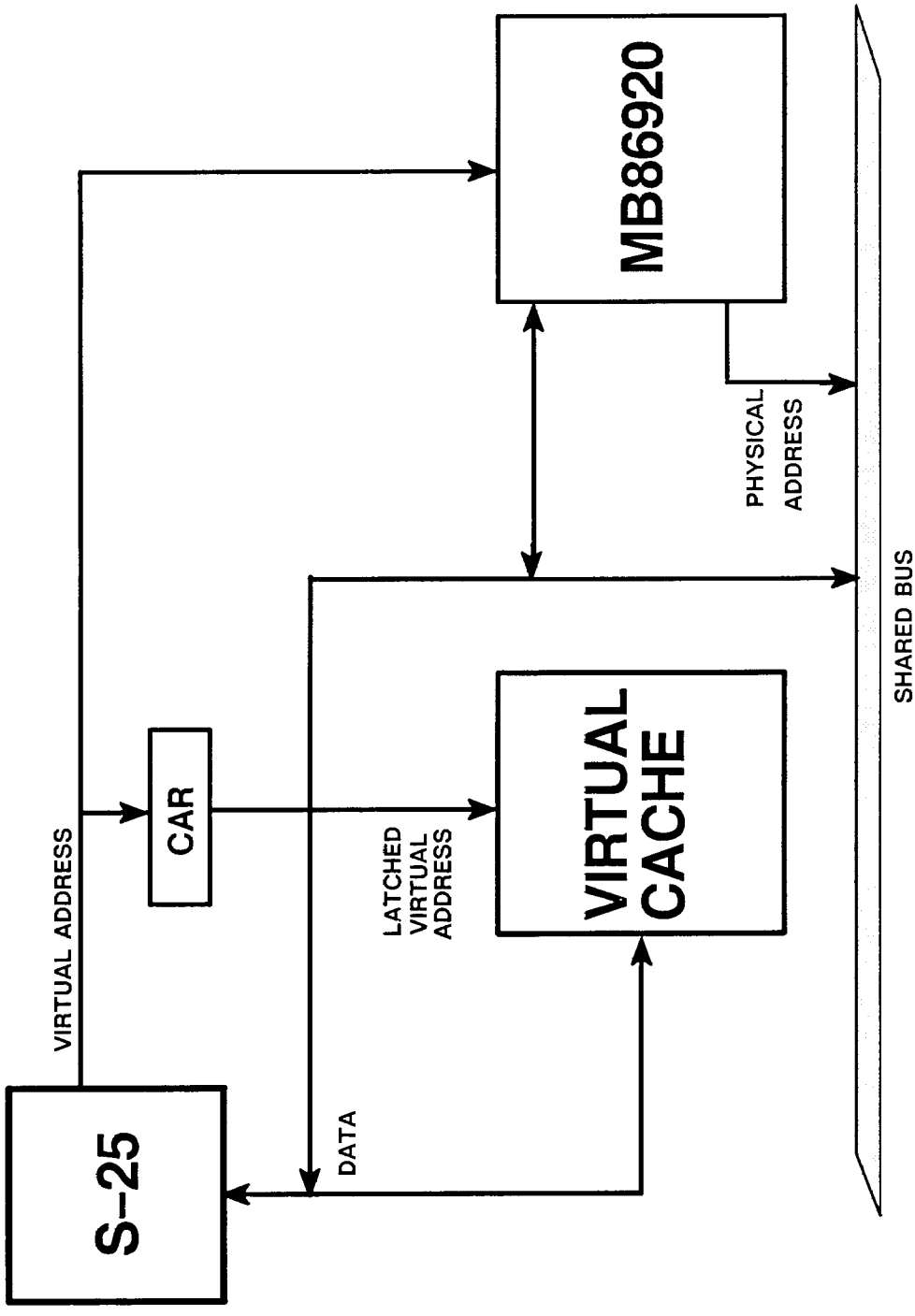


FIGURE 7.1.1 ON-LINE VIRTUAL SYSTEM

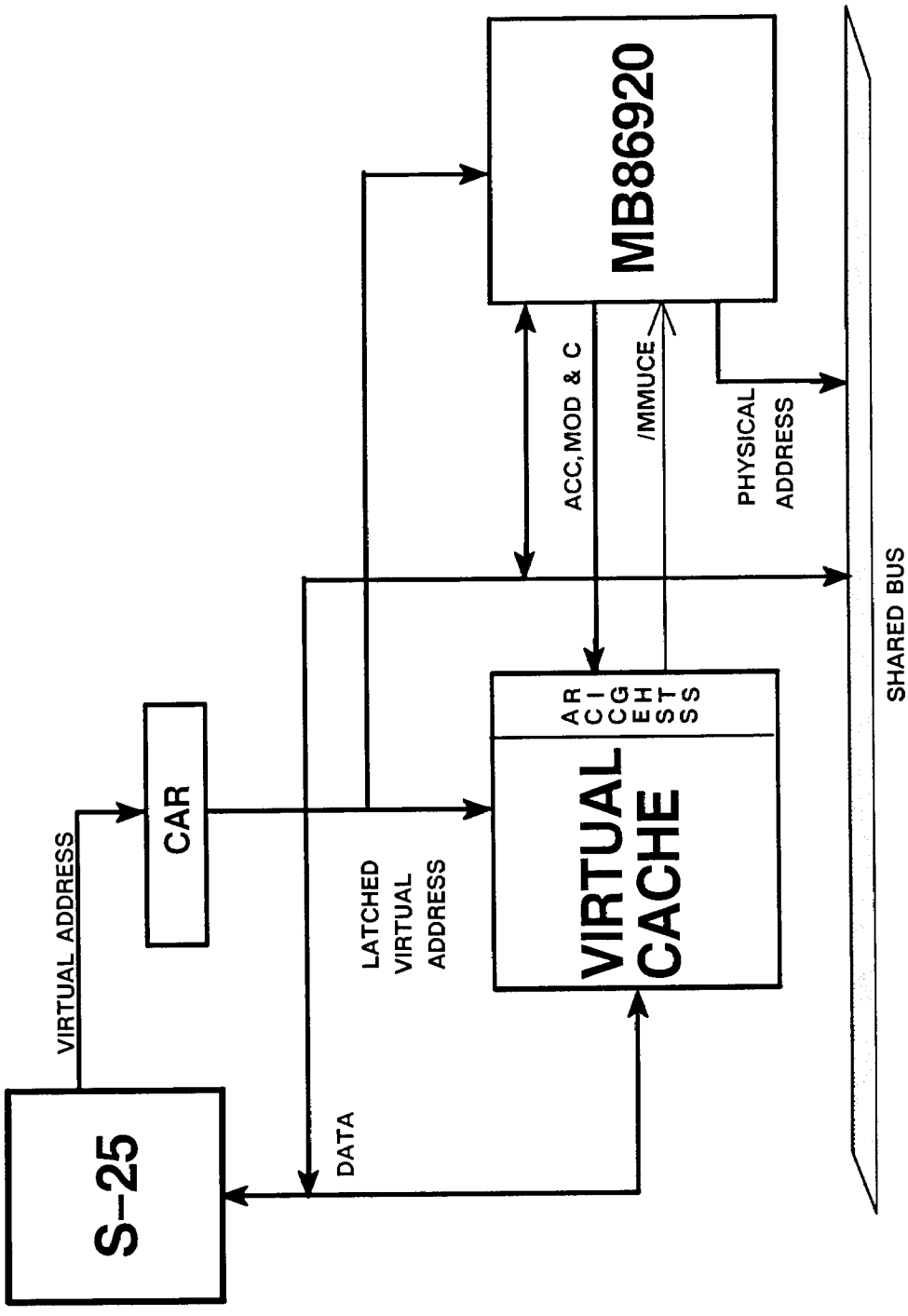


FIGURE 7.1.2 OFF-LINE VIRTUAL SYSTEM

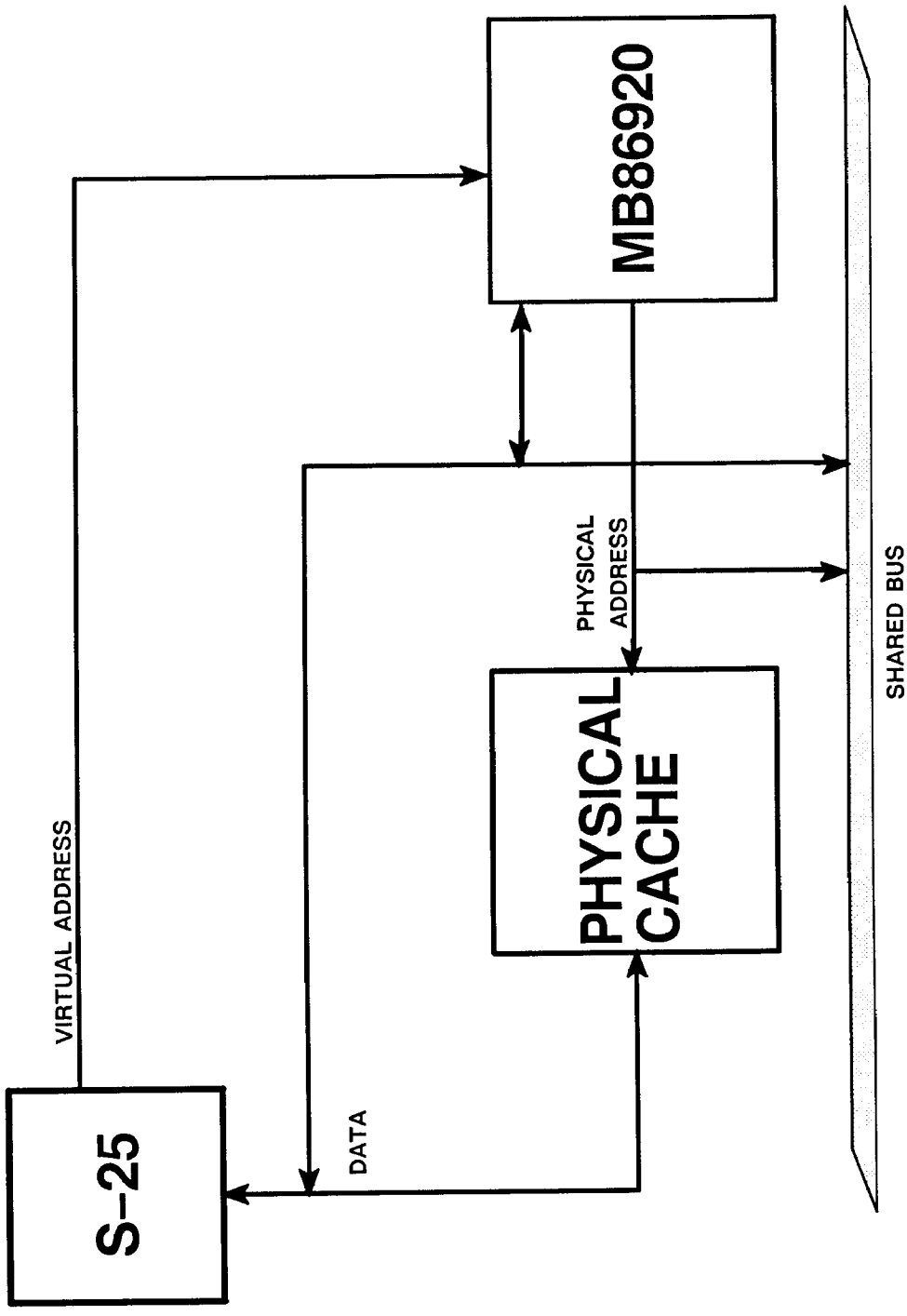


FIGURE 7.2 PHYSICAL CACHE SYSTEM

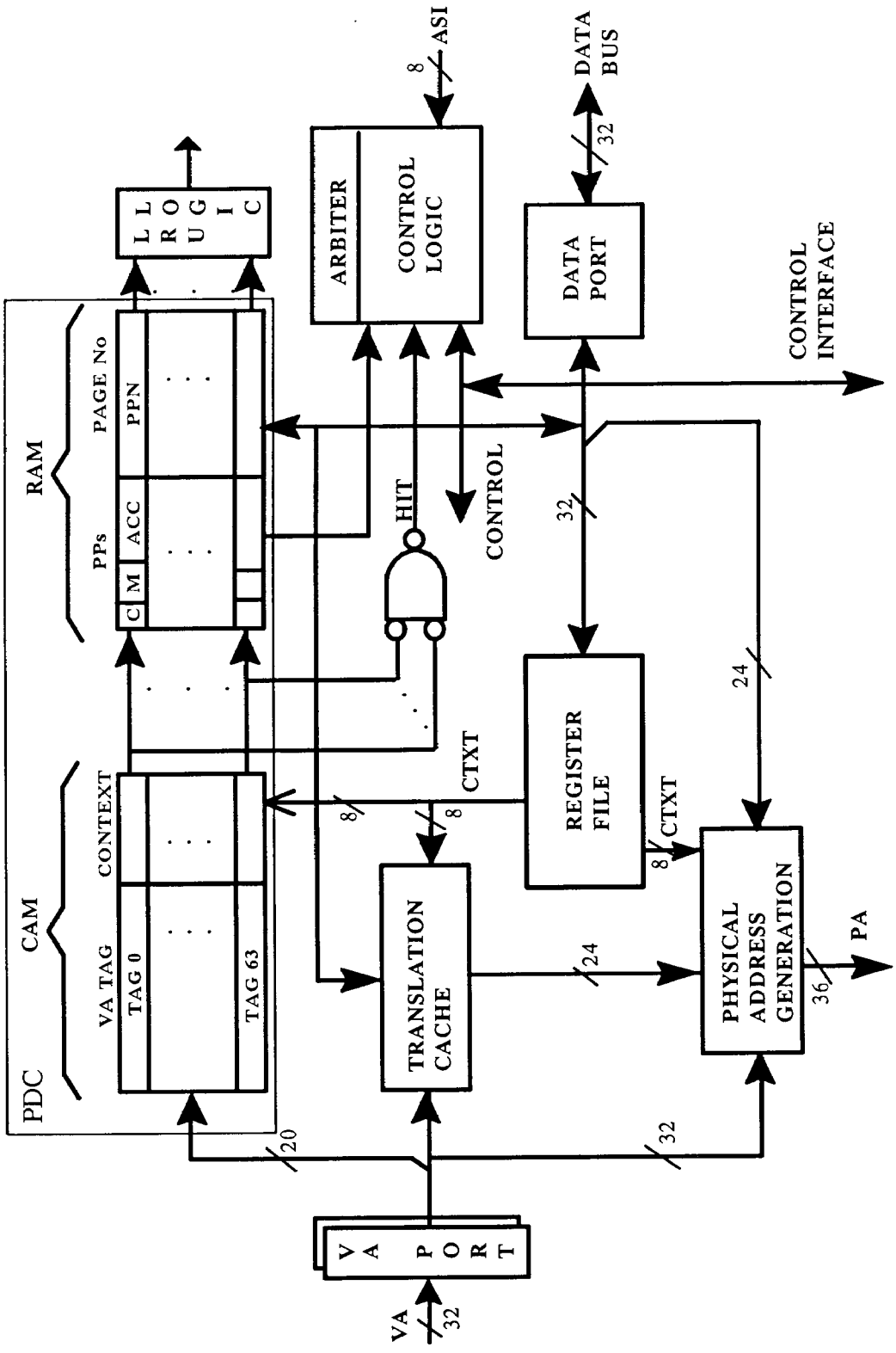


FIGURE 8.1 ARCHITECTURAL BLOCK DIAGRAM

9. Transaction States

Once powered, the MB86920 is always involved in one of six possible transaction states:

VIRTUAL PASSTHROUGH: To fall into this state when the MB86920 is off, the ASIs must be different than 04 and 2x Hex. This state is automatically obtained after power-up, after reset, and after clearing E in the control register. This state is characterized by making the physical address equal to the virtual address with the upper four physical address lines equal to zero. The state is also entered when setting E and switching the ASIs to anything but: 03, 04, 08, 09, 0A, 0B and 2x (Hex).

PHYSICAL PASSTHROUGH: This state is characterized by making the physical address equal to the virtual address with the upper four physical address lines equal to ASI[3:0]. This state is entered whether or not the MB86920 is on as long as ASIs equal 2x Hex. This state is esentially beneficial before initializing the MB86920 in order to access physical addresses.

REGISTER: The register access mode can be entered regardless of whether the part is on or off, as long as the ASIs are equal to 04 Hex. Refer to timings.

FLUSH/PROBE: To enter the flush or probe states, the MB86920 does not need to be turned on (E bit set to 1), chip enabled or both, only the ASIs need to equal 03 Hex. Refer to section 2.4.

TRANSLATION: Translation state is entered when the MB86920 is turned on and the ASIs are equal to 08, 09, 0A or 0B Hex. However, there are two substates:

(a) *Full Translation*: A full translation requires that the physical address be generated, that access permisions be checked and violations reported. For this to happen, the MB86920 must not only be turned on, it must also be enabled by asserting /MMUCE.

In on-line systems, and physical cache systems, the line must be tied low; in off-line systems, the line can be activated only during cache misses.

(b) *Partial Translation*: A partial translation consists of the physical address being generated without any access checking. This occurs when the MB86920 is not enabled (/MMUCE negated) or NULLCYC is asserted.

TABLEWALK: Tablewalk is the state entered by the MB86920 when it is enabled, the ASIs are equal to 08, 09, 0A or 0B Hex, and the PDC reports a miss or as a result of a probe operation. During this state, the MB86920 requests the physical bus, becomes the bus master, and accesses the system tables to ultimately obtain the PTE associated with the virtual address that missed on the PDC. On Type[4:0] Probes, the MB86920 accesses the page tables too, unless the probe is type [3] and the Root Pointer is already internally stored.

10. Initialization

After reset, the E bit, within the control register is set to zero. On power-up, the registers come up in an unknown state, and after reset, they retain the content they had prior to being reset. E set to zero implies that the MB86920 is not enabled and therefore, not performing translations (VA[31:0] = PA[31:0]). All addresses are assumed physical, and PA[35:32] are set to zero unless the ASIs are equal to 2x Hex, in which case ASI[3:0] are passed directly to PA[35:32].

The registers have to be set in the following order:

1. Context Table Pointer Register
2. Context Register
3. Control register (E=1)

The Fault Status Register should then be read in order to clear its contents. Note that enabling the MB86920 results in switching from passthrough mode to translation mode.

11. Pinout

S-25 Interface

Signal	Type	Pin(s)	Description
VA[31-0]	I	L14, A14, A13, M1, O1, P2-P14, 015, N15, M15, L15, K15, J15, H15, G15, F15, E15, D15, C15, B15	Virtual Address: Instruction or data address which is to to be checked or translated.
ASI[7-0]	I	D1, E1, F1, G1, H1, J1, K1, L1	Address Space Identifier: These eight lines identify the address space referenced by VA during instruction and data accesses. The MB86920 recognizes the following encodings: 00000011 MB86920 Probe/Flush 00000100 MB86920 Register Access 00001000 User Instruction Space 00001001 Supervisor Instruction Space 00001010 User Data Space 00001011 Supervisor Data Space 0010xxxx Physical Passthrough
/HAL	I	C1	Hold Address Latch (Active Low): When asserted low, it inhibits clocking the virtual address registers.
NULLCYC	I	B1	Null Cycle (Active High): When asserted high it inhibits checking or updating the page descriptor cache.
LDST	I	A2	Load and Store (Active High): When asserted high, it indicates that an Atomic Load-Store operation is in progress. It is used by the MB86920 to detect early exception processing i.e., check for write permission as the read is performed.
RD	I	A3	Read Data: Active high during all load or instruction fetch operations and active low for store operations.
/MDS	O	E5	Memory Data Strobe (Active Low; open drain): Signals devices on the processor data bus when data from the MB86920 is valid.
/MHOLD	O	07	MHold (Active Low): Asserted if the MB86920 is engaged in a page table access.
/MEXC	O	L12	Memory Exception (Active Low): Asserted if a virtual address translation results in an access violation or if a page table access cannot be completed.

Data Bus

Signal	Type	Pin(s)	Description
D[31-0]	I/O	B14, C10, N12, L5, L11, D4, E4, M4, M5, M12, D12, C12, C11, M3, N3, N7, N13, M13, L13, D13,C13, B13, B12, N2, O2, O3, O4, O13, O14, N14, M14, C14	Data (3-State): A 32-bit bus which is used to access the MB86920 register or to access page table data in memory.
DM	O	O6	Data Master (Active High): Asserted during the last cycle of a Probe operation (returning data), during register load operations, and throughout the tablewalk process.

System Control Signals

/RESET	I	D2	Reset (Active Low): Initializes the MB86920 and leaves it in passthrough mode.
CLK1	I	N5	Clock 1: Same as S-25's system clock 1 i.e., Clock 1 is a 75% duty cycle clock used by the MB86920 to obtain internal and external synchronization. It must be high during the first three quarters of the duty cycle and low during the last.

Cache Control And Off-line Support Signals

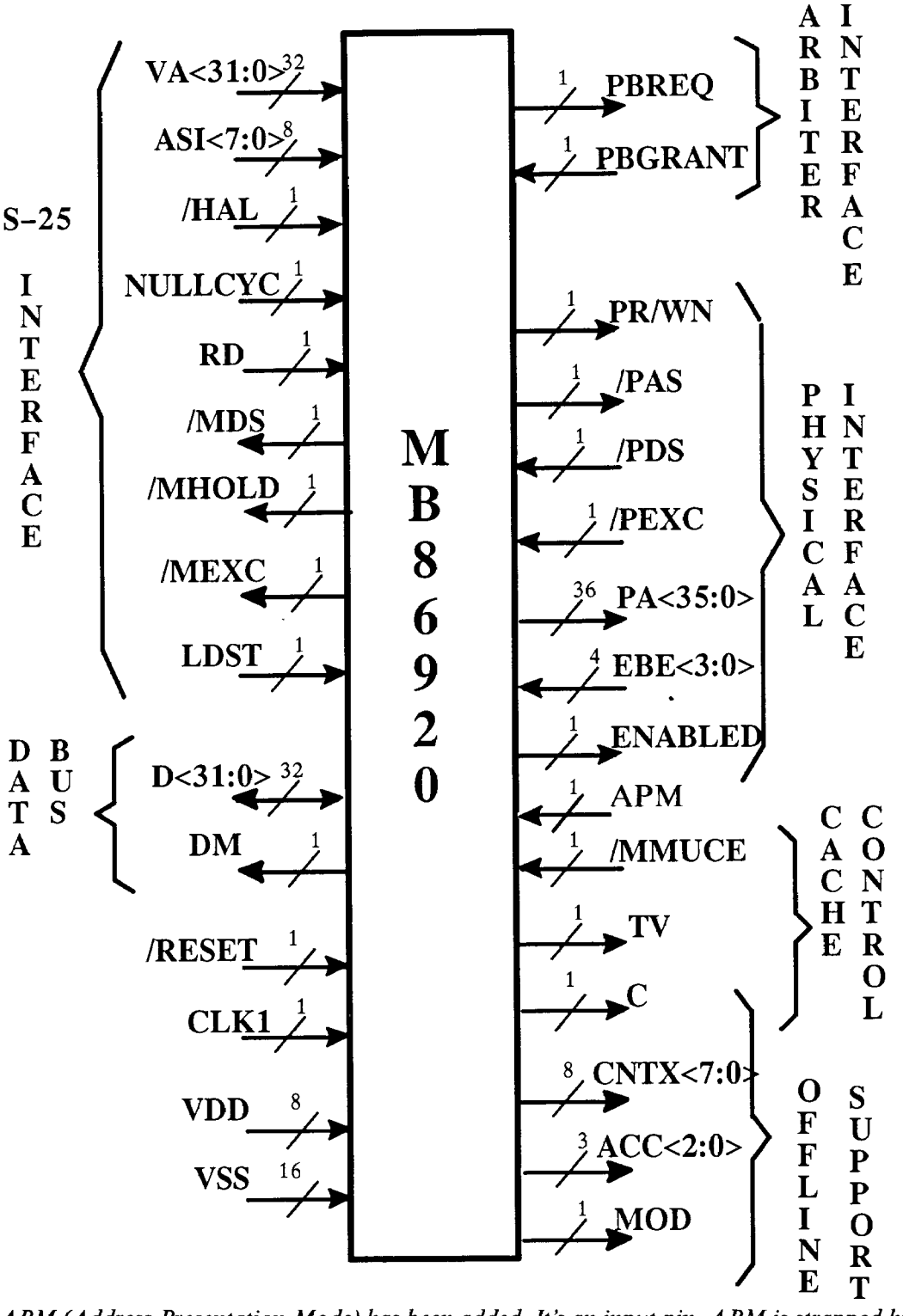
CNTX[7-0]	O	N6, B2, C2, O5, C4, C3, D3, E3	Context: Reflects the value of the MB86920 context register.
ACC[2-0]	O	C5, L3, N9	Access: Defines the access privileges of the word whose virtual address is being translated by the MB86920.
MOD	O	K13	Modified (Active High): Indicates that the page containing the word whose virtual address is being translated by the MB86920 has been written into.
C	O	N10	Cacheable (Active High): Indicates that the data at the virtual address being translated by the MB86920 is cacheable.
/MMUCE	I	A4	MMU Chip Exnable (Active Low): This signal informs the MB86920 to perform the translation by either releasing the physical address or beginning the tablewalk process. If /MMUCE is high the LRU information is not updated.
TV	O	N11	Translation Valid (Active High): Indicates that the physical address bus contains a valid translation of the address on the virtual bus. TV is valid whenever PA is a result of a translation or PA is equal to Virtual Address. Note: TV is not affected by /MMUCE.

Bus Arbitration

Signal	Type	Pin(s)	Description
PBREQ	O	N8	Physical Bus Request (Active High): Indicates the MB86920 needs the physical bus.
PBGRANT	I	A5	Physical Bus Grant (Active High): Indicates that the MB86920 has possesion of the physical bus.

Physical Bus Interface

PA[35-0]	O	J13, H13, G13, F13, E13, B10, B9, B8, B6, B5, B4, E2, F2, G2, H2, J2, K2, L2, M2, O10, O11, O12, E14, D14, C9, C8, C7, C6, G3, H3, J3, K3, K14, J14, G14, F14	Physical Address : This is a 36-bit bus which is the translation of a virtual address or the address of a page table entry.
PR/WN	O	B11	Physical Read/Write (3-State): If high, indicates the MB86920 is reading a word from memory. If low, the MB86920 is writing a word to memory.
/PAS	O	D11	Physical Address Strobe (Active Low; 3-State): Indicates that the MB86920 is writing a word to memory.
/PDS	I	A6	Physical Data Strobe: Asserted when the external bus has acquired outbound data or when inbound data is valid.
/PEXC	I	A7	Physical Exception (Active Low;): When low, an error has occurred in a memory access. It must be asserted simultaneously with /PDS.
ENABLED	O	N4	MMU Enabled (Active High): This signal reflects the status of the Control Register's E bit. When active, the MMU is on, when negated the MMU is off.
EBE[3:0]	I	A11-A8	External Bus Error: These inputs are system dependent and they can be used to detect various types of memory exceptions. They are sampled on every rising edge of CLK1 and reported through the EBE field within the Fault Status Register.
APM	I	N1	Address Presentation Mode: Determines whether certain cache control and physical bus interface outputs are registered before being driven. The signals are registered for APM = 1, not registered for APM = 0. The affected signals are PA[35:0], ACC[2:0], MOD, C, TV, PR/WN, /PAS. Note: PR/WN and /PAS are not affected on the prototype.



An APM (Address Presentation Mode) has been added. It's an input pin. APM is strapped high or low. When an APM = 1, PA is delayed by one cycle. When APM = 0, there is no delay.

FIGURE 11.1 MMU SIGNALS

ELECTRICAL CHARACTERISTICS

ABSOLUTE MAXIMUM RATINGS*

Rating		Symbol	Min.	Max.	Unit
Supply Voltage		V_{DD}	V_{SS}^{**} - 0.5	6.0	V
Input Voltage		V_I	V_{SS}^{**} - 0.5	V_{DD}+ 0.5	V
Output Voltage		V_O	V_{SS}^{**} - 0.5	V_{DD}+ 0.5	V
Temperature under Bias	Ceramic Plastic	T_{bias}	-40 -25	+125 +85	°C
Storage Temperature	Ceramic Plastic	T_{stg}	-65 -40	+150 +125	°C
Power Dissipation		P_D	0.5	*TBD*	W

Note : * Permanent device damage may occur if ABSOLUTE MAXIMUM RATINGS are exceeded. Functional operation should be restricted to the conditions as detailed in the operational sections of the data sheet. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

** V_{SS} = 0 Volt.

RECOMMENDED OPERATING CONDITIONS

Parameter	Symbol	Min.	Typ.	Max.	Unit
Supply Voltage	V_{DD}	4.75	5.0	5.25	V
Input High Voltage	V_{IH}	2.2			V
Input Low Voltage	V_{IL}			0.8	V
Operating Temperature	T_A	0		70	°C

CAPACITANCE (T_A= 25°C , V_{DD}= V_I = 0 Volt , F_o = 1 MHz)

Parameter	Symbol	Min.	Typ.	Max.	Unit
Input Pin Capacitance	C_{IN}			16	pF
Output Pin Capacitance	C_{OUT}			16	pF
I/O Pin Capacitance	$C_{I/O}$			16	pF

DC CHARACTERISTICS

(RECOMMENDED OPERATING CONDITIONS unless otherwise noted)

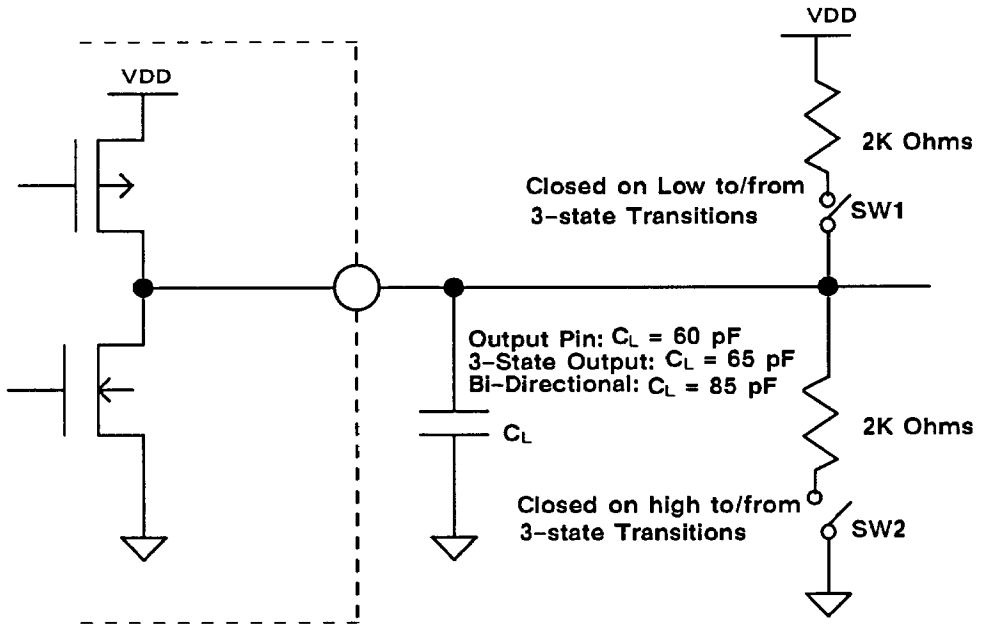
Parameter	Symbol	Condition	Min.	Typ.	Max.	Unit
Power Supply Current	I_{DDS}	Steady State*			0.2	mA
Output High Voltage	V_{OH}	I_{OH}=-0.2 mA	4.2		V_{DD}	V
Output Low Voltage	V_{OL}	I_{OL}= 3.2 mA	V_{SS}		0.4	V
Output Low [1] Voltage for Driver Output	V_{OL}	I_{OL}= 12.0 mA	V_{SS}		0.4	V
Output High [2] Voltage for Driver Output	V_{OH}	I_{OH}= -4 mA	4.2		V_{DD}	V
Input High Voltage	V_{IH}		2.2			V
Input Low Voltage	V_{IL}				0.8	V
Input Leakage Current	I_{LI}	V_I = 0 to V_{DD}	-10		10	u A
3-State Leakage Current	I_{LZ}	V_I = 0 to V_{DD}	-40		40	u A
Input Pull-Up/ Down Resistor	RP	V_{IH} = V_{DD} V_{IL} = V_{SS}	25	50	100	KΩ

* Note: $V_{IH} = V_{DD}$, $V_{IL} = V_{SS}$

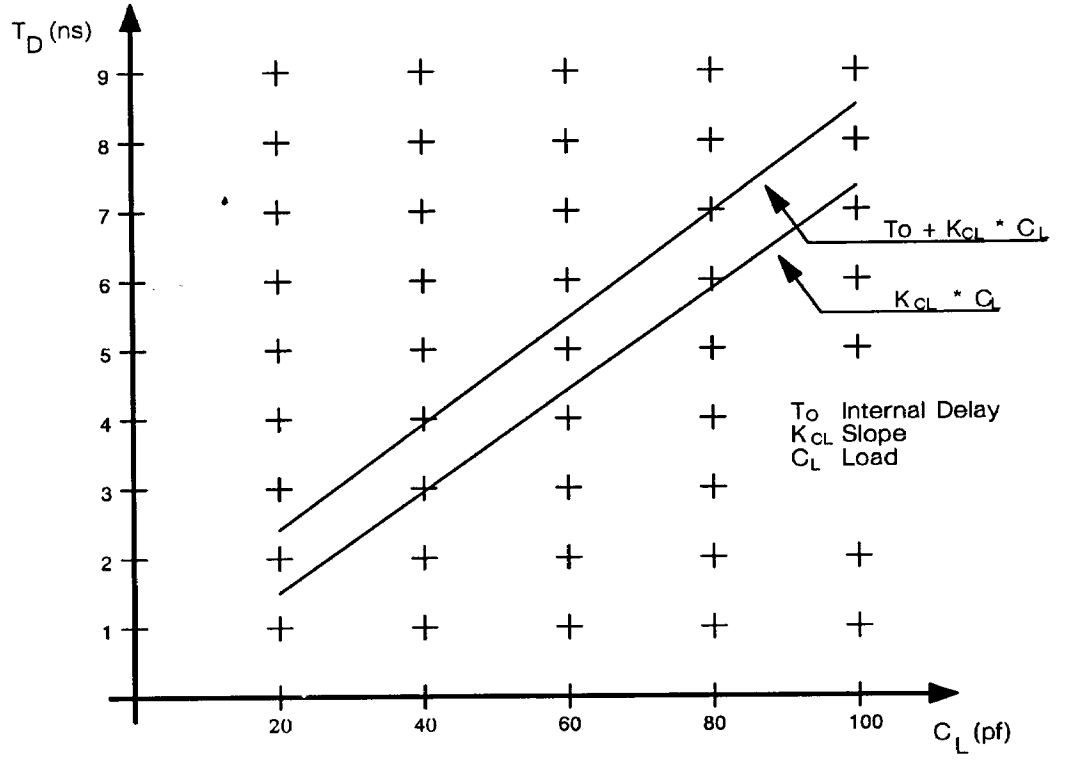
[1] Applies to: [D31:0], Enabled, /PDS, PR/WN, /PAS, PA[11:0], DM, /MHOLD, /MEXC, BBREQ, CNTX[7:0]

[2] Applies to: PA[35:12], ACC[2:0], MOD, C, TV.

SIGNAL OUTPUT TEST LOAD



MAXIMUM OUTPUT DELAY VS. CAPACITANCE LOADING



AC CHARACTERISTICS

$V_{DD} = 5V \pm 5\%$ $T_A = 0$ to $+70°C$

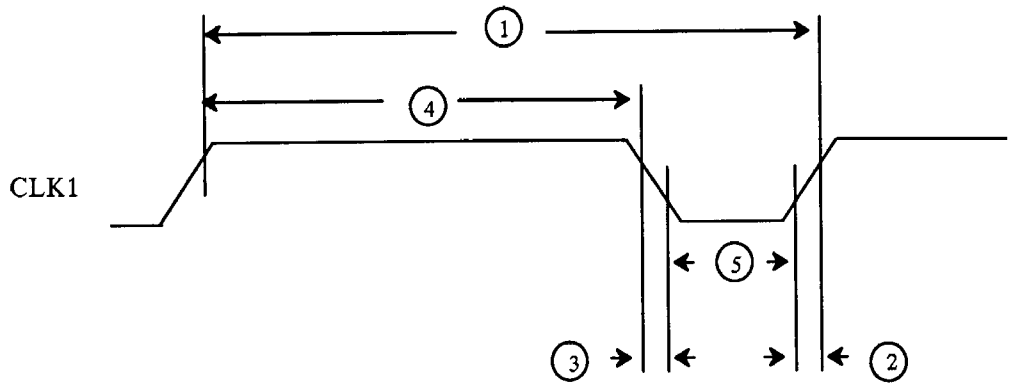
Capacitance: Output pin = 60 pF
3-state = 65pF
Bi-directional = 85 pF

Symbol	Parameter	Min	Typ	Max	Units	Test Condition
1	System Clock Cycle Time (T)	40			ns	
2	System Clock Rise Time			2	ns	
3	System Clock Fall Time			2	ns	
4	System Clock (Clk1) High Time	28	30		ns	
5	System Clock (Clk1) Low Time	8	10		ns	
6	$\overline{\text{RESET}}$ Active Time	10			T	
7	DM Asserted From CLK1 Rising Edge			23	ns	
8	DM Negated From CLK1 Rising Edge	0		28	ns	
9	Load Data Valid From CLK1 Rising Edge			36	ns	
10	Load Data Hold From CLK1 Rising Edge	4			ns	
11	Load Data 3-State From CLK1 Rising Edge			45	ns	
12	$\overline{\text{MHOLD}}$ Valid From CLK1 Rising Edge			27	ns	
13	$\overline{\text{MHOLD}}$ Negated From CLK1 Rising Edge	0		21	ns	
14	$\overline{\text{MDS}}$ Asserted From CLK1 Rising Edge			20	ns	
15	$\overline{\text{MDS}}$ Negated From CLK1 Rising Edge	1			ns	
16	Store Data Setup To CLK1 Rising Edge	14			ns	
17	Store Data Hold From CLK1 Rising Edge	5			ns	
18	TV Asserted From CLK1 Rising Edge			23	ns	

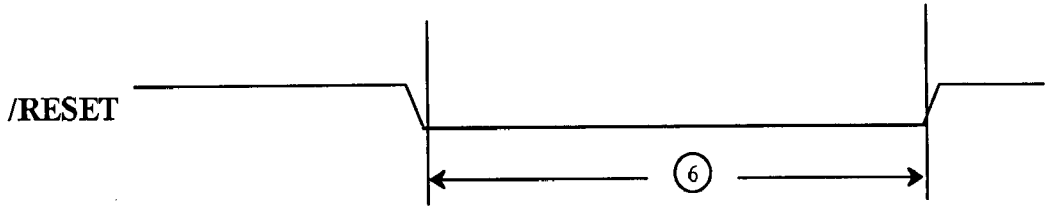
Symbol	Parameter	Min	Typ	Max	Units	Test Condition
19	TV Negated from CLK1 Rising Edge	0		24	ns	
20	PA Valid From CLK1 Rising Edge		28	38	ns	
21	PA Invalid From CLK1 Rising Edge	0			ns	
22	No Parameter Defined					
23	Protection Information (PI) From CLK1 Rising Edge			25	ns	
24	PI Invalid From CLK1 Rising Edge	0		TBD	ns	
25	VA Setup to CLK1 Rising Edge	6			ns	
26	VA Hold From CLK1 Rising Edge	4			ns	
27	PBREG Valid From CLK1 Rising Edge			23	ns	
28	PBREQ Invalid From CLK1 Rising Edge	0		30	ns	
29	PBGRANT Setup To CLK1 Rising Edge	3			ns	
30	PBGRANT Hold From CLK1 Rising Edge	4			ns	
31	Table Address Valid From CLK1 Rising Edge			36	ns	
32	Table Address Hold From CLK1 Rising Edge	0			ns	
33	See 43				ns	
34	See 44				ns	
35	$\overline{\text{PAS}}$ Asserted From CLK1 Rising Edge			34	ns	
36	$\overline{\text{PAS}}$ Negated From CLK1 Rising Edge	0		26	ns	

Symbol	Parameter	Min	Typ	Max	Units	Test Condition
37	$\overline{\text{PDS}}$ Setup to CLK1 Rising Edge	11			ns	
38	$\overline{\text{PDS}}$ Hold From CLK1 Rising Edge	7			ns	
39	PR/WN Asserted from CLK1 Rising Edge			34	ns	
40	PR/WN Negated from CLK1 Rising Edge			26	ns	
41	Read Data Setup To CLK1 Rising Edge	5			ns	
42	Read Data Hold From CLK1 Rising Edge	11			ns	
43	Write Data Valid From CLK1 Rising Edge			35	ns	
44	Write Data Invalid from CLK1 Rising Edge			35	ns	
45	$\overline{\text{HAL}}$ Setup to CLK1 Rising Edge	10			ns	
46	$\overline{\text{HAL}}$ Hold From CLK1 Rising Edge	4			ns	
47	$\overline{\text{MMUCE}}$ Setup from CLK1 Rising Edge	11			ns	
48	$\overline{\text{MMUCE}}$ Hold from CLK 1 Rising Edge	7			ns	
49	$\overline{\text{PEXC}}$ Setup To CLK1 Rising Edge	11			ns	
50	$\overline{\text{PEXC}}$ Hold to CLK1 Rising Edge	7			ns	
51	$\overline{\text{MEXC}}$ Asserted From CLK1 Rising Edge			20	ns	
52	$\overline{\text{MEXC}}$ Hold From CLK1 Rising Edge	1			ns	
53	$\overline{\text{MDS}}$ Asserted From CLK1 Rising Edge			20	ns	
54	$\overline{\text{MDS}}$ Asserted From CLK1 Rising Edge	1			ns	

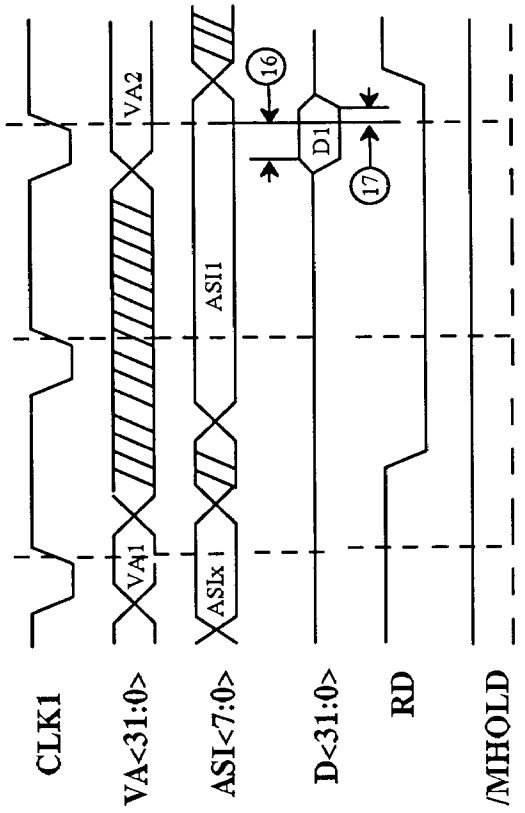
NOTE: PA, ACC, MOD, C and TV are affected by APM. All timings are provided with APM = 1.



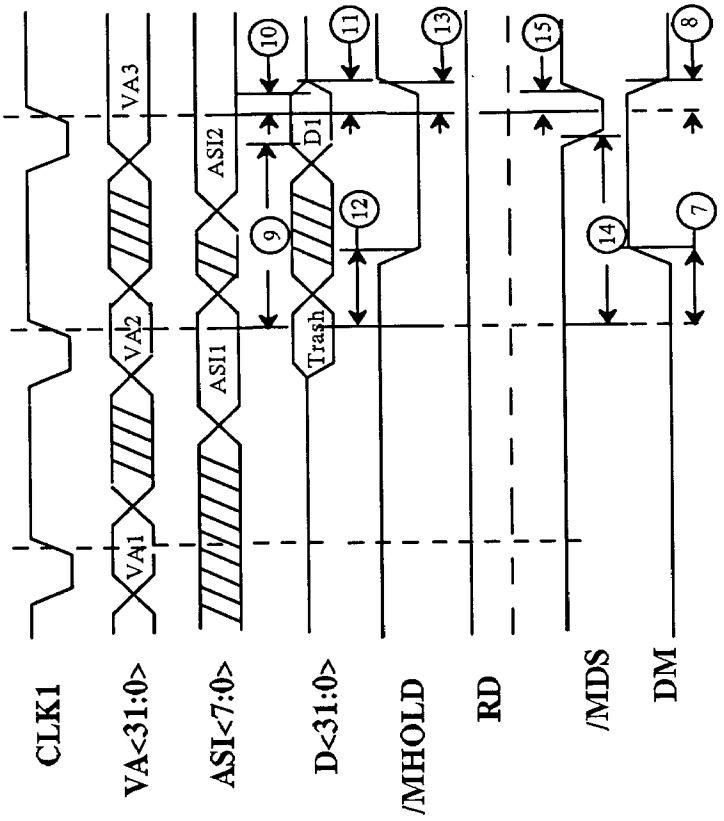
TIMING 1 SYSTEM CLOCK TIMING



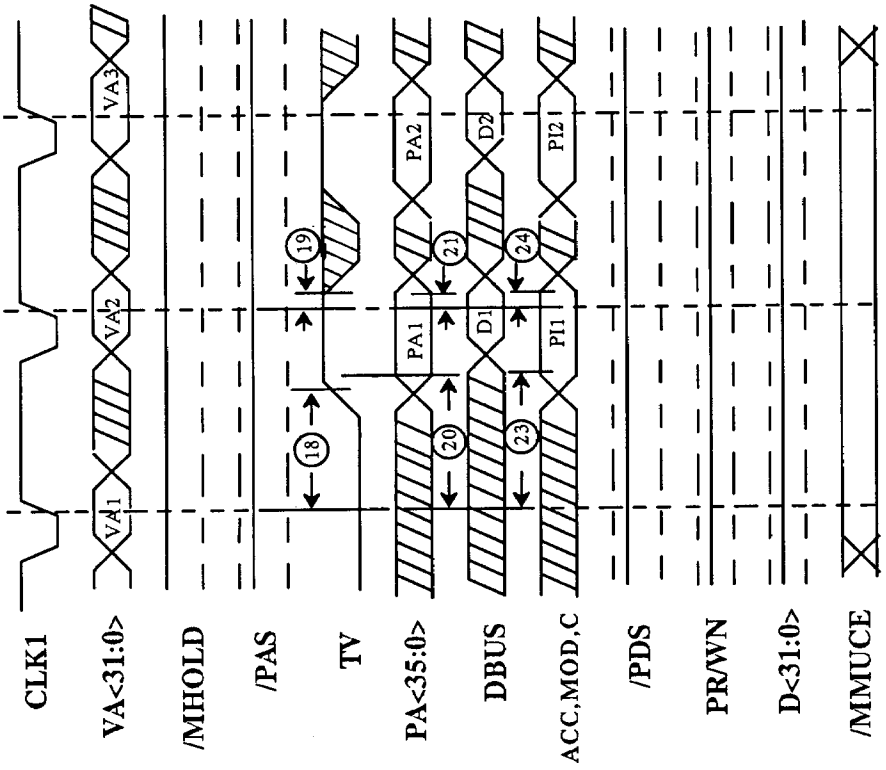
TIMING 2 RESET TIMING



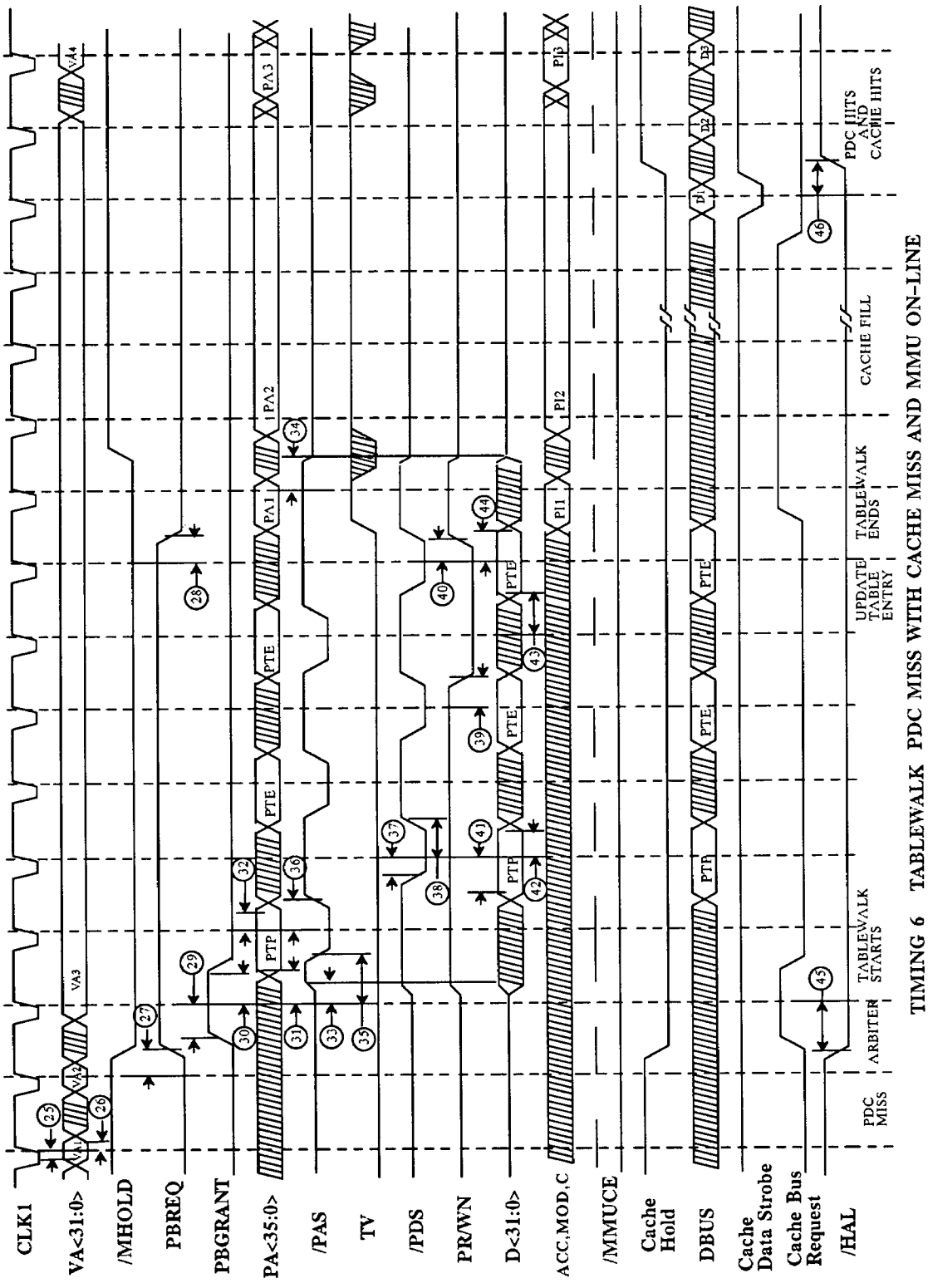
TIMING 3 REGISTER STORE



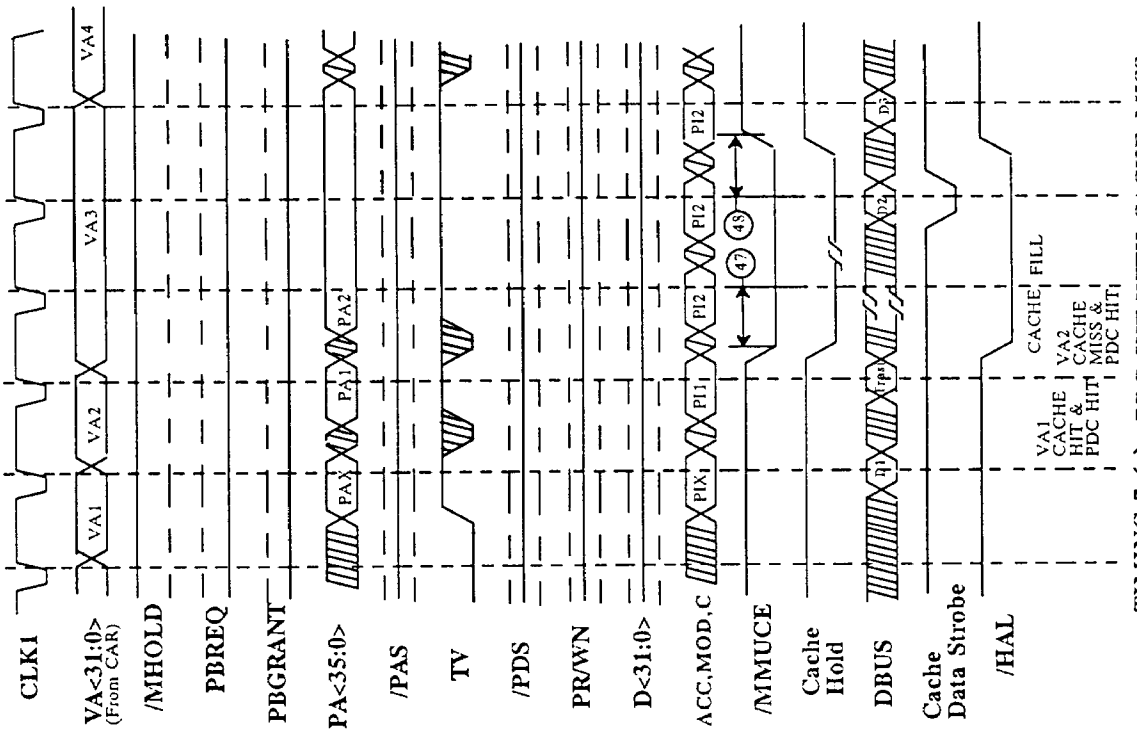
TIMING 4 REGISTER LOAD



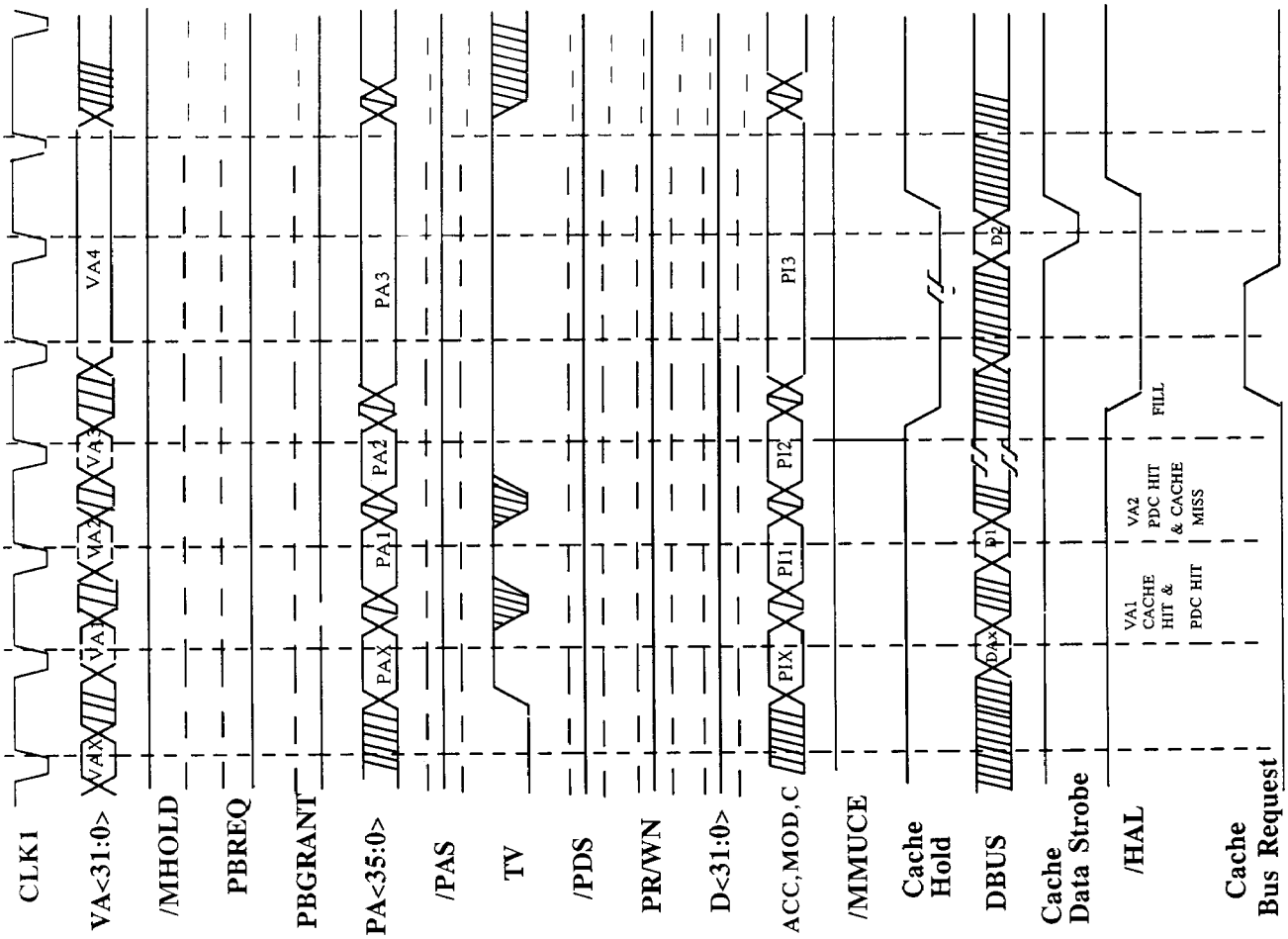
TIMING 5 PDC-HITS WITH CACHE-HITS (MMU-ON-LINE)



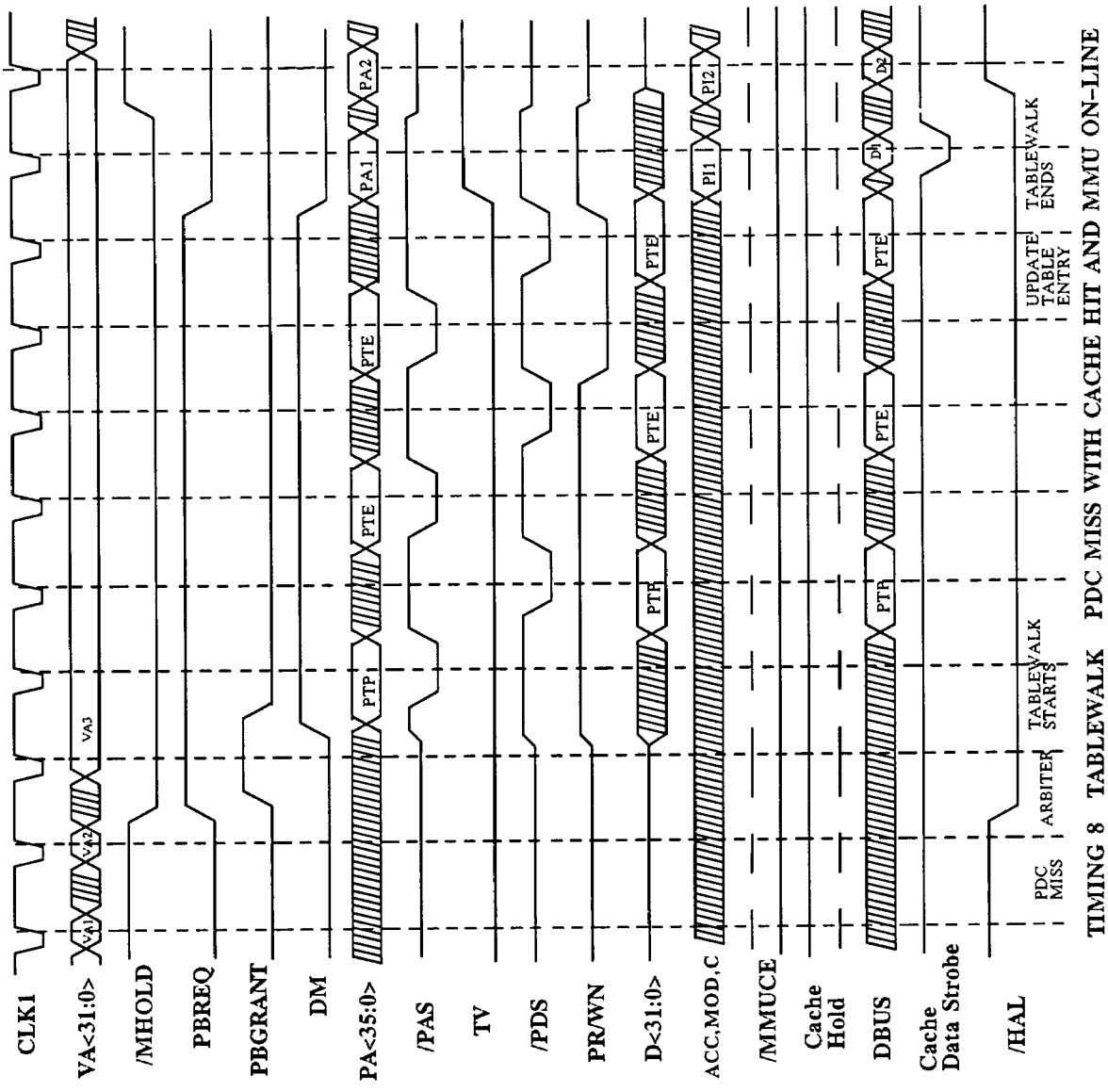
TIMING 6 TABLEWALK PDC MISS WITH CACHE MISS AND MMU ON-LINE



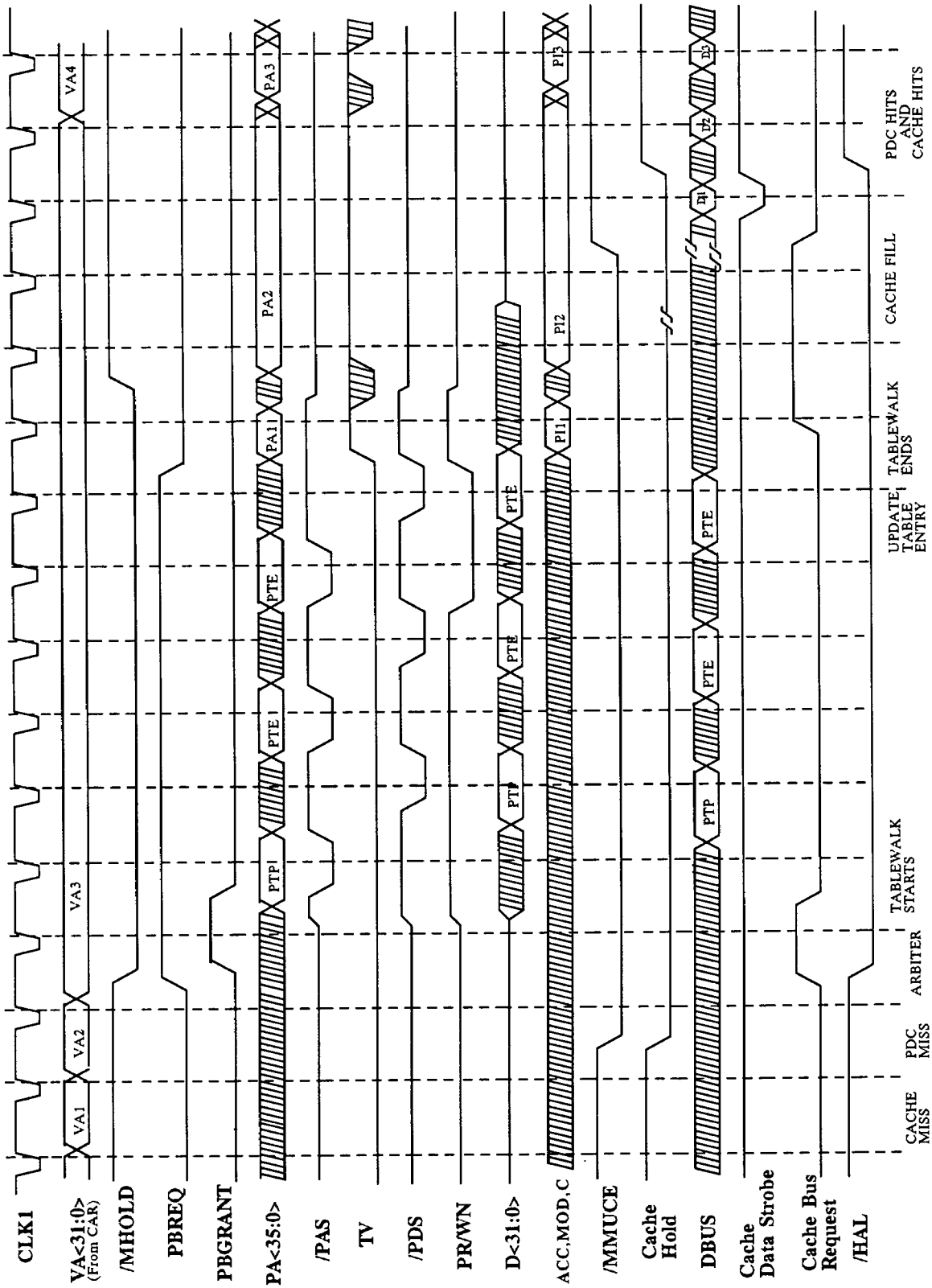
TIMING 7 (a) PDC-HIT WITH CACHE-MISS
(MMU OFF-LINE)



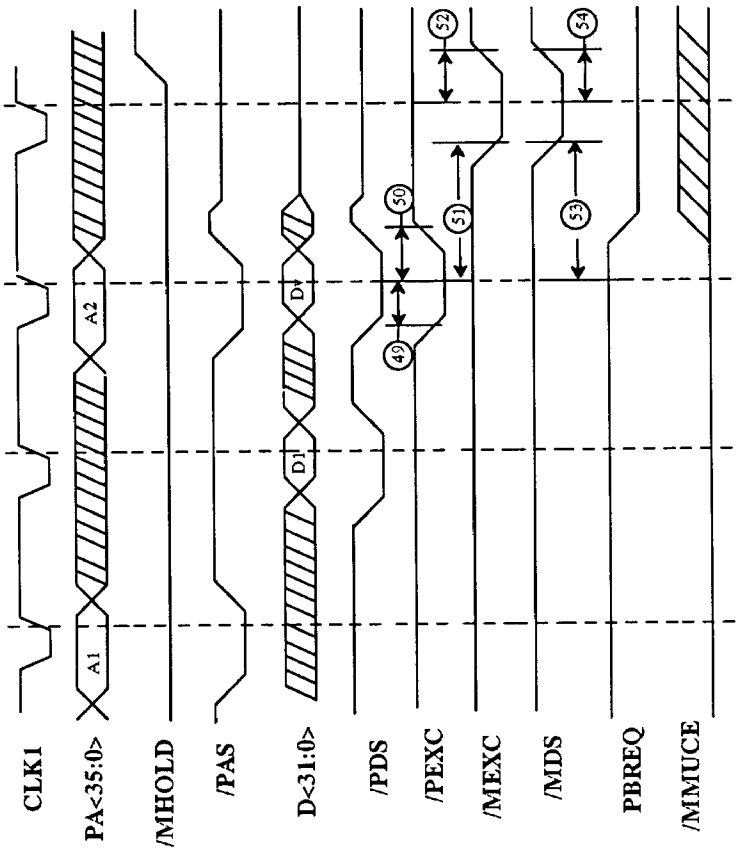
TIMING 7 (b) PDC-HIT WITH CACHE-MISS (MMU ON-LINE)



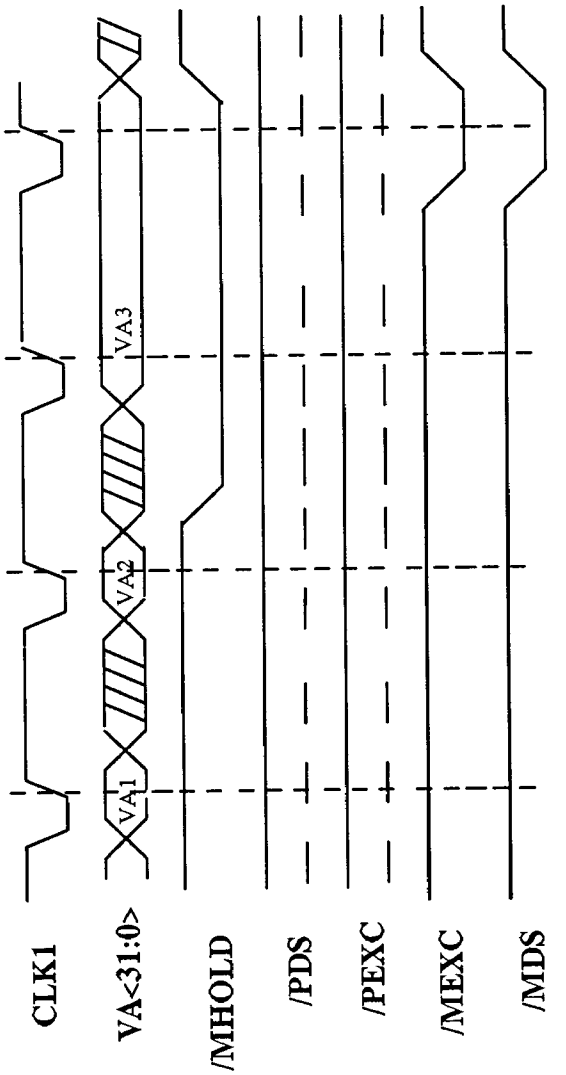
TIMING 8 TABLEWALK PDC MISS WITH CACHE HIT AND MMU ON-LINE



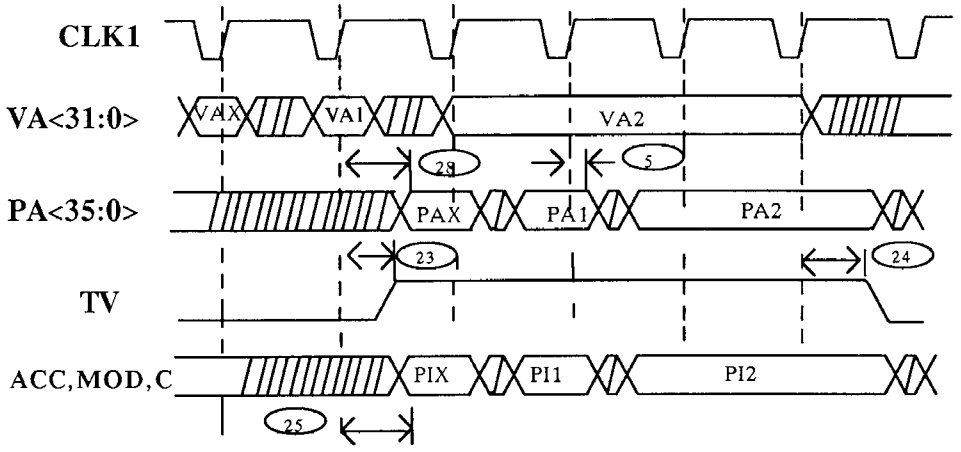
TIMING 9 TABLEWALK; PDC MISS WITH CACHE MISS AND MMU OFF-LINE



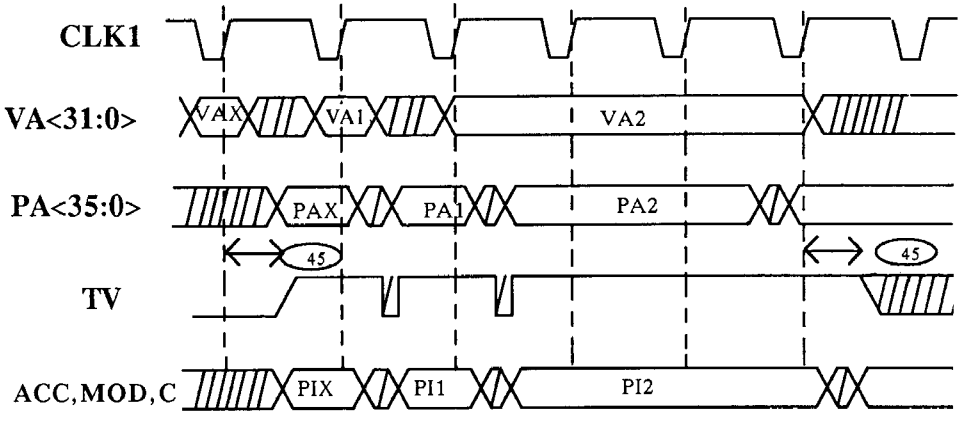
TIMING 10 MEMORY EXCEPTION DETECTED DURING TRANSLATION



TIMING 11 MEMORY EXCEPTION DETECTED BY PDC

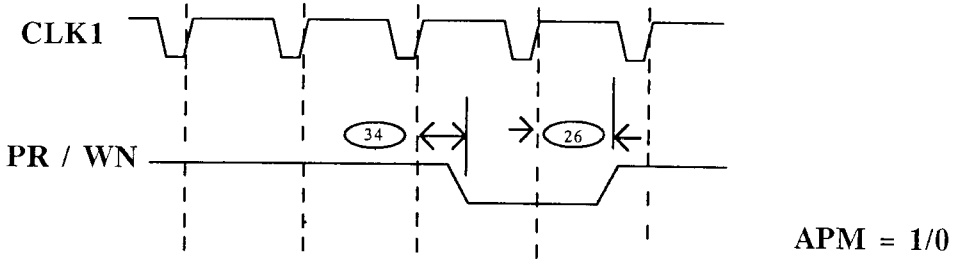


12(a) Signal Affected when APM = 1



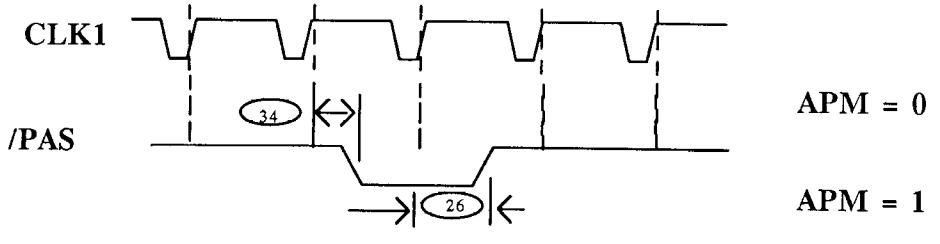
12(b) Signal Affected when APM = 0

TIMING 12 Signals Affected when APM = 0 and APM = 1



TIMING 12(c)

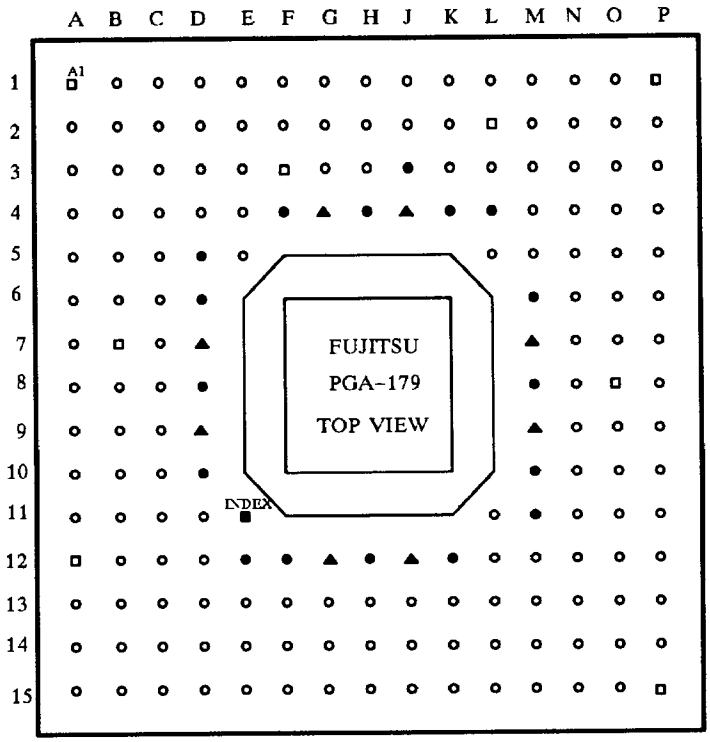
Note: In the current Rev the /PAS and PR/WN signals are not affected whether APM = 0 or APM = 1.



TIMING 12(d)

TIMING 12 (Continued) Signals Affected when APM = 0 and APM = 1

MB86920 PIN ASSIGNMENT



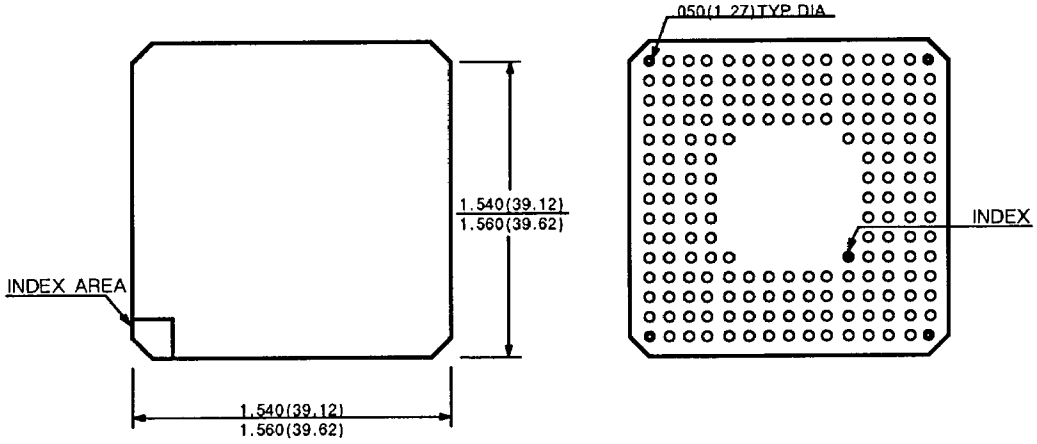
Pin Group	Symbol	Pinout
VDD	▲	D7, D9, G4, G12, J4, J12, M7, M9
VSS	●	D5, D6, D8, D10, E12, F4, F12, H4, H12, K4, K12, L4 M6,M8, M10, M11
INDEX	■	E11
No Connect (NC)	□	A1, A12, B7, F3, B3, O8, O9, P1, P15

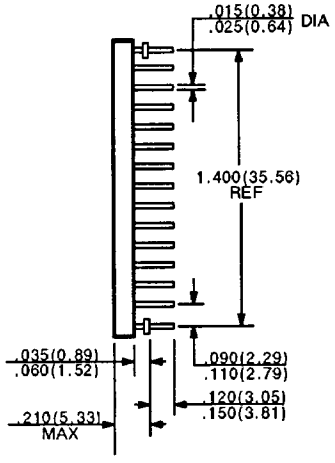
Pin Assignment

PIN	I/O	NAME	PIN	I/O	NAME	PIN	I/0	NAME
A1	-	NC	E1	I	ASI6	M1	O	VA27
A2	I	LDST	E2	O	PA24	M2	O	PA17
A3	I	RD	E3	O	CNTX0	M3	I/O	D18
A4	I	/MMUCE	E4	I/O	D25	M4	I/O	D24
A5	I	PBGRANT	E5	O	/MDS	M5	I/O	D23
A6	I	/PDS	E12	I	VSS	M6	I	VSS
A7	I	/PEXC	E13	O	PA31	M7	I	VDD
A8	I	EBE0	E14	O	PA13	M8	I	VSS
A9	I	EBE1	E15	I	VA3	M9	I	VDD
A10	I	EBE2	F1	I	ASI5	M10	I	VSS
A11	I	EBE3	F2	O	PA23	M11	I	VSS
A12	-	NC	F3	-	NC	M12	I/O	D22
A13	I	VA28	F4	I	VSS	M13	I/O	D14
A14	I	VA29	F12	I	VSS	M14	I/O	D1
A15	-	NC	F13	O	PA32	M15	I	VA10
						N1	I	APM
B1	I	NULLCYC	F14	O	PA0	N2	I/O	D8
B2	O	CNTX6	F15	I	VA4	N3	I/O	D17
B3	O	NC	G1	I	ASI4	N4	O	ENABLED
B4	O	PA25	G2	O	PA22	N5	I	CLK1
B5	O	PA26	G3	O	PA7	N6	O	CNTX7
B6	O	PA27	G4	I	VDD	N7	I/O	D16
B7	-	NC	G12	I	VDD	N8	O	PBREQ
B8	O	PA28	G13	O	PA33	N9	O	ACC0
B9	O	PA29	G14	O	PA1	N10	O	C
B10	O	PA30	G15	I	VA5	N11	O	TV
B11	O	PR/WN	H1	I	ASI3	N12	I/O	D29
B12	I/O	D9	H2	O	PA21	N13	I/O	D15
B13	I/O	D10	H3	O	PA6	N14	I/O	D2
B14	I/O	D31	H4	I	VSS	N15	I	VA11
B15	I	VA0	H12	I	VSS	O1	I	VA 26
C1	I	/HAL	H13	O	PA34	O2	I/O	D7
C2	O	CNTX5	H14	I	VA30	O3	I/O	D6
C3	O	CNTX2	H15	I	VA6	O4	I/O	D5
C4	O	CNTX3	J1	I	ASI2	O5	O	CNTX4
C5	O	ACC2	J2	O	PA20	O6	O	DM
C6	O	PA8	J3	O	PA5	O7	O	/MHOLD
C7	O	PA9	J4	I	VDD	O8	-	NC
C8	O	PA10	J12	I	VDD	O9	-	PA18
C9	O	PA11	J13	O	PA35	O10	O	PA16
C10	I/O	D30	J14	O	PA2	O11	O	PA15
C11	I/O	D19	J15	I	VA7	O12	O	PA14
C12	I/O	D20	K1	I	ASI1	O13	I/O	D4
C13	I/O	D11	K2	O	PA19	O14	I/O	D3
C14	I/O	D0	K3	O	PA4	O15	I	VA12
C15	I	VA1	K4	I	VSS	P1	-	NC
D1	I	ASI7	K12	I	VSS	P2	I	VA25
D2	I	/RESET	K13	O	MOD	P3	I	VA24
D3	O	CNTX1	K14	O	PA3	P4	I	VA23
D4	I/O	D26	K15	I	VA8	P5	I	VA22
D5	I	VSS	L1	I	ASI0	P6	I	VA21
D6	I	VSS	L2	O	NC	P7	I	VA20
D7	I	VDD	L3	O	ACC1	P8	I	VA19
D8	I	VSS	L4	I	VSS	P9	I	VA18
D9	I	VDD	L5	I/O	D28	P10	I	VA17
D10	I	VSS	L11	I/O	D27	P11	I	VA16
D11	O	/PAS	L12	I	/MEXC	P12	I	VA15
D12	I/O	D21	L13	I/O	D13	P13	I	VA14
D13	I/O	D12	L14	O	VA31	P14	I	VA13
D14	O	PA12	L15	I	VA9	P15	-	NC
D15	I	VA2						

MB86920 PACKAGE DESCRIPTION

179-LEAD CERAMIC/PLASTIC
PIN GRID ARRAY PACKAGE
(CASE No.: PGA-179P-MO1)





Dimensions in
inches (millimeters)