DEC 05 1991

ADVANCE INFORMATION

82485
SECOND LEVEL CACHE CONTROLLER FOR THE i486™ MICROPROCESSOR

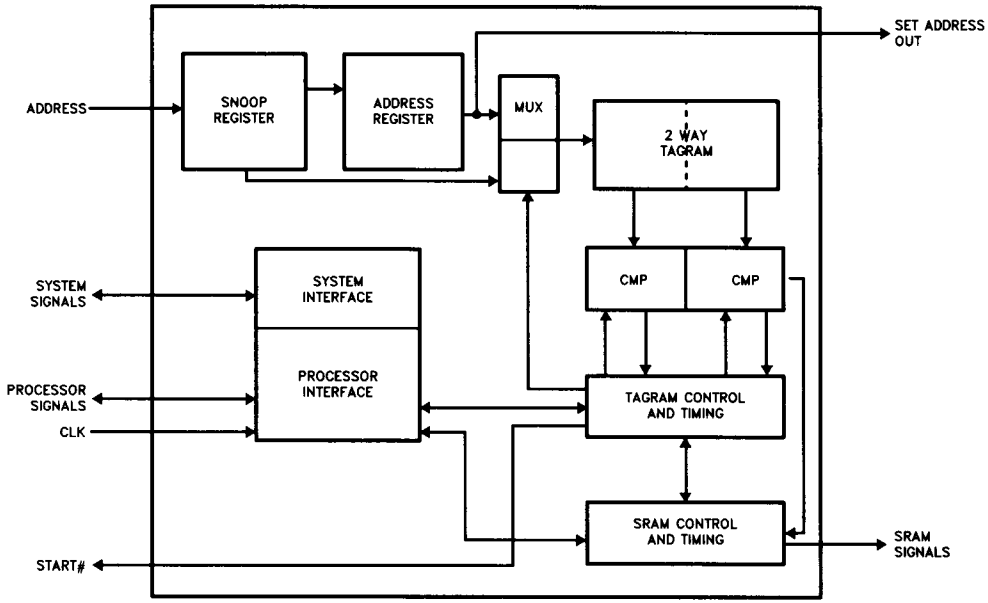
- **High Performance**
  - — Zero Wait State Access on Cache Hit
  - — One Clock Bursting
  - — Two-Way Set Associative
  - — Write Protect Attribute Per Tag
  - — Start Memory Cycles in Parallel
- **Easy to Use**
  - — Matches i486™ Microprocessor Bus Timing
  - — Supports Invalidation Cycles
  - — Maintains Memory on Writes
- **High Integration**
  - — Single Chip Tag RAM and Controller
  - — No Logic Needed for CPU and Cache Connection
  - — Maps Full 4 Gigabyte Address Space
- **Flexible System Configurations**
  - — Supports 64K or 128K Cache Memory Per Controller
  - — Allows Multiple Controllers for Larger Cache Size
  - — Supports Non-Cacheable Memory Areas

The 82485 is a second-level cache controller designed to improve the performance of i486™ Microprocessor systems. One 82485 cache controller supports 64K or 128K bytes of second level cache memory that maps to the entire 4 Gigabytes of the i486 microprocessor address space. The controller is completely software transparent. Several controllers may be cascaded to provide larger cache sizes. One controller plus SRAMs provides a 64K or a 128K cache. External EPROM can be cached yet remain write protected. The 82485 is fully compatible with the i486 microprocessor. All i486 CPU bus cycles and timings are supported.

A complete, optional second level cache controller using the 82485 is available as the 485Turbocache Module from Intel (data sheet order number 240722).

i486 is a trademark of Intel Corporation.

82485 Internal Block Diagram



240831-1

Intel Corporation assumes no responsibility for the use of any circuitry other than circuitry embodied in an Intel product. No other circuit patent licenses are implied. Information contained herein supersedes previously published specifications on these devices from Intel. **January 1991**
© INTEL CORPORATION, 1991 **Order Number: 240831-001**

82485
Second Level Cache Controller for the i486 Microprocessor

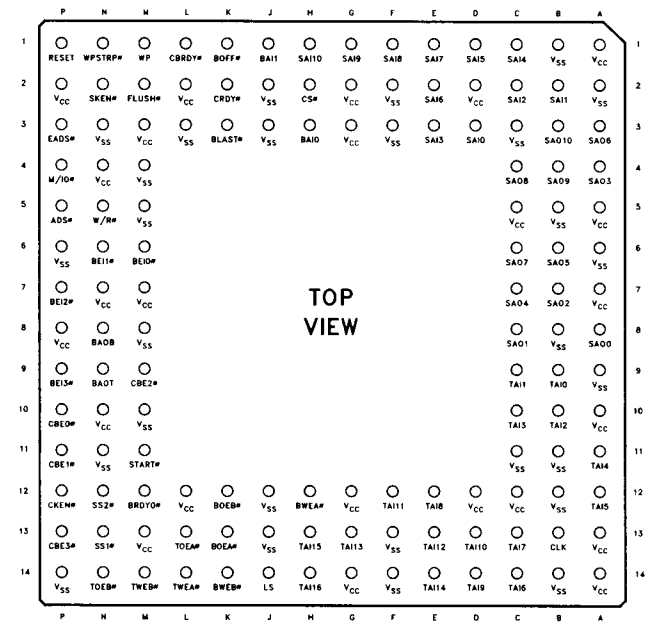
CONTENTS

CONTENTS PAGE

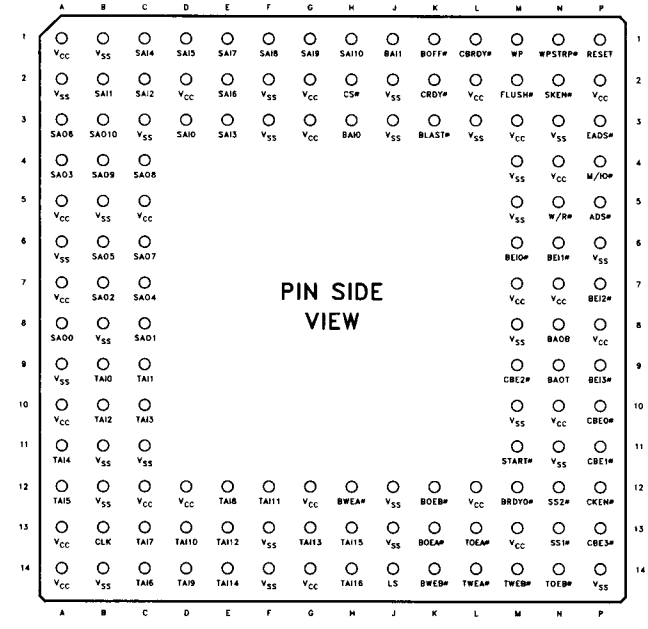
CONTENTS PAGE

1.0 PINOUT

This chapter provides the pinout diagram and pin numbering scheme for the 82485 cache controller. A brief description of each pin is also provided.



240831-2



240831-3

Figure 1.1 82485 Pinout

1.1 Pin Cross Reference (sorted by pin name)

Pin Name	Location
ADS#	P5
BAI0	H3
BAI1	J1
BAOT	N9
BAOB	N8
BEI0#	M6
BEI1#	N6
BEI2#	P7
BEI3#	P9
BLAST#	K3
BOEA#	K13
BOEB#	K12
BOFF#	K1
BRDY0#	M12
BWEA#	H12
BWEB#	K14
CBE0#	P10
CBE1#	P11
CBE2#	M9
CBE3#	P13
CKEN#	P12
CLK	B13
CBRDY#	L1
CRDY#	K2
CS#	H2

Pin Name	Location
EADS#	P3
FLUSH#	M2
LS	J14
M/IO#	P4
RESET	P1
SAI0	D3
SAI1	B2
SAI2	C2
SAI3	E3
SAI4	C1
SAI5	D1
SAI6	E2
SAI7	E1
SAI8	F1
SAI9	G1
SAI10	H1
SAO0	A8
SAO1	C8
SAO2	B7
SAO3	A4
SAO4	C7
SAO5	B6
SAO6	A3
SAO7	C6
SAO8	C4

Pin Name	Location
SAO9	B4
SAO10	B3
SKEN#	N2
SS1#	N13
SS2#	N12
START#	M11
TAI0	B9
TAI1	C9
TAI2	B10
TAI3	C10
TAI4	A11
TAI5	A12
TAI6	C14
TAI7	C13
TAI8	E12
TAI9	D14
TAI10	D13
TAI11	F12
TAI12	E13
TAI13	G13
TAI14	E14
TAI15	H13
TAI16	H14
TOEA#	L13
TOEB#	N14

Pin Name	Location
TWEA#	L14
TWEB#	M14
WP	M1
WPSTRP#	N1
W/R#	N5
VSS	A2, A6, A9, B1, B5, B8, B11, B12, B14, C3, C11, F2, F3, F13, F14, J2, J3, J12, J13, L3, M4, M5, M8, M10, N3, N11, P6, P14
V_{CC}	A1, A5, A7, A10, A13, A14, C5, C12, D2, D12, G2, G3, G12, G14, L2, L12, M3, M7, M13, N4, N7, N10, P2, P8

1.2 Pin Description Overview

Pin Name	Type	Active	Description
Control Signals			
CLK	I	High	CLOCK is the timing reference from which the 82485 monitors and generates events. CLK must be the same as the i486 processor clock.
RESET	I	High	RESET forces the 82485 into a known state before execution can begin. It causes the entire cache to be flushed. Setup and hold times t24 and t25 must be met to guarantee recognition. The 82485 requires 4 inactive clock periods after RESET is deasserted.
ADS#	I	Low	ADDRESS STROBE is generated by the i486 processor. It indicates the start of an external cycle. It is used by the 82485 to latch valid addresses and begin a cache cycle. Setup and hold times t8 and t9 must be met for proper operation.

1.2 Pin Description Overview (Continued)

Pin Name	Type	Active	Description
Control Signals (Continued)			
M/IO#	I	—	MEMORY/IO is generated by the i486 processor to indicate a memory (M/IO# high) or an I/O (M/IO# low) access. The 82485 only responds to memory accesses. Setup and hold times t8 and t9 must be met for proper operation.
W/R#	I	—	WRITE/READ is generated by the i486 processor to indicate a write (W/R# high) or a read (W/R# low) access. Setup and hold times t8 and t9 must be met for proper operation.
START#	O	Low	MEMORY START indicates that a cache read miss or a write has occurred and that the current access must be serviced by the memory system. START# is not activated for I/O cycles and if CS# is not asserted.
BRDYO#	O	Low	BURST READY OUT is a burst ready signal driven by the 82485 to the processor. It is activated when a read hit occurs to the 82485 and should be a term in the BRDY# signal to the processor.
CBRDY#	I	Low	CACHE BURST READY IN is the burst ready input from the memory system. It indicates that data from a burst transfer is available to be written into the cache. CBRDY# is ignored in T1 and idle cycles, and is applied to both the 82485 and the processor in parallel. Setup and hold times t12 and t13 must be met for proper operation.
CRDY#	I	Low	CACHE READY IN is the non-burst ready input from the memory system. It indicates that data from a non-burst transfer is available to be written into the cache. Like CBRDY#, CRDY# is ignored in T1 and idle cycles, and is applied to the 82485 and processor in parallel. Setup and hold times t12 and t13 must be met for proper operation.
BLAST#	I	Low	BURST LAST is output by the processor to indicate the last cycle of a transfer. Setup and hold times t10 and t11 must be met for proper operation.
BOFF#	I	Low	BACKOFF is a processor input sampled by the 82485 that terminates the cycle in process immediately. The 82485 will drive all the write and output enables inactive in the clock after BOFF# is asserted. All cycles except invalidation cycles will be ignored when BOFF# is active. Setup and hold times t18 and t19 must be met for proper operation.
CS#	I	Low	CHIP SELECT is used to allow multiple 82485s to be mapped to different physical memory locations by decoding address bits to select the appropriate controller. Chip Select is sampled in T1 and with EADS#. Setup and hold times t30 and t31 must be met for proper operation.
Address Signals			
SAI0–SAI10	I	—	SET ADDRESS IN pins are latched internally and decoded by the 82485 to select the appropriate location within the tag RAM. Setup and hold times t6 and t7 must be met for proper operation.
TAI0–TAI16	I	—	TAG ADDRESS IN pins are compared to the internal TAG addresses to determine if the access is a "hit" or "miss". Setup and hold times t6 and t7 must be met for proper operation.

1.2 Pin Description Overview (Continued)

Pin Name	Type	Active	Description
Address Signals (Continued)			
BAI0-BAI1	I	—	BURST ADDRESS IN selects the burst start address used during read cycles and the proper dword during write cycles. Setup and hold times t6 and t7 must be met for proper operation.
LS	I	—	LINE SELECT defines which line of the tag will be accessed with the 128K sectored cache configuration. This pin must be strapped low in the 64K configuration. Setup and hold times t6 and t7 must be met for proper operation. This output is not defined when the bus is idle.
BE0# -BE3#	I	Low	BYTE ENABLES define which byte(s) are valid for partial write cycles when there is a cache hit. The 82485 forces these inputs low during read hit cycles and cache line fills. Setup and hold times t6 and t7 must be met for proper operation.
Cacheability Signals			
CKEN#	O	Low	CACHE ENABLE TO CPU is the KEN# term generated by the 82485 to the processor. CKEN# is active in T1 and inactive in the first T2. It will remain inactive for the remainder of the cycle if it is a read miss cycle. If the cycle is a read hit cycle, CKEN# will become active in the second T2 and remain active for the duration of the cycle. CKEN# is active on an idle bus.
SKEN#	I	Low	SYSTEM CACHE ENABLE is an input from the main memory sytem to indicate whether the current cycle is cacheable in the 82485. To fill an 82485 cache line, SKEN# must be asserted both the clock before the 1st BRDY# or RDY# and the clock before the last BRDY# or RDY# is returned to the processor. Setup and hold times t14 and t15 must be met for proper operation.
FLUSH#	I	Low	FLUSH causes the 82485 to invalidate the entire contents of the cache. Set up and hold times t24 and t25 must be met to guarantee recognition in any specific clock. Flush# may be asserted asynchronously according to pulse width time t26.
WP	I	High	WRITE PROTECT defines a line as write protected. Any writes to a write protected line will not update the cache RAM. Setup and hold times t16 and t17 must be met at each clock edge regardless of the value of the pin.
WPSTRP#	I	Low	WRITE PROTECT STRAPPING OPTION changes the behavior of CKEN# when a read hit occurs on a write protected line. If WPSTRP# is high, CKEN# will validate a WP line fill to the processor. If WPSTRP# is low, write protected lines will not be validated by CKEN#. WPSTRP# is not a programmable option.
EADS#	I	Low	EXTERNAL ADDRESS STROBE indicates that an invalidation address is present on the address bus. The 82485 will invalidate this address if it is present in the cache and CS# is active. The 82485 can accept an EADS# every other clock. Setup and hold times t20 and t21 must be met for proper operation.

1.2 Pin Description Overview (Continued)

Pin Name	Type	Active	Description
RAM Interface Pins			
BAOT/BAOB	O	—	BURST ADDRESS OUT (TOP and BOTTOM). During the first transfer of a cache line read or fill, the state of these pins will be identical to BAI1. This allows both top and bottom bank SRAMs to be selected in anticipation that there will be a cache hit in one of them. The 82485 will manipulate these pins during subsequent transfers of the line access. These outputs are not defined when the bus is idle.
SAO0–SAO10	O	—	SET ADDRESS OUT pins drive the set address to the external SRAM. The set address value is the value latched at the SAI pins during the first transfer of any cache access. These outputs are not defined when the bus is idle.
SS1#–SS2#	O	Low	SECTOR SELECT pins provide the extra decoding to select the appropriate line in the external data RAM when the 82485 is configured to support a 128K cache. SS2# should not be connected in the 64K configuration. These outputs are not defined when the bus is idle.
CBE0#–CBE3#	O	Low	CACHE BYTE ENABLE signals control which bytes are written to the external RAM. During all read and dword write cycles all 4 byte enables are active. During partial writes, 1 or 2 of the byte enables are active. These outputs are not defined when the bus is idle.
TOEA#–TOEB# BOEA#–BOEB#	O	Low	TOP and BOTTOM OUTPUT ENABLES select which RAM bank will be enabled onto the data bus during a read cycle. TOEA# and TOEB# enable the top RAM bank and BOEA# and BOEB# enable the bottom RAM bank.
TWEA#–TWEB# BWEA#–BWEB#	O	Low	TOP and BOTTOM WRITE ENABLES cause the contents of the write data latch (internal to the SRAM) to be written into its data array. TWEA# and TWEB# enable the top RAM bank. BWEA# and BWEB# enable the bottom RAM bank.

2.0 FUNCTIONAL DESCRIPTION

2.1 Introduction

The 82485 is a single ported, two-way set associative cache controller designed specifically to interface with the i486 microprocessor. The controller supports either a sectored configuration (two lines per tag) or a non- sectored configuration (one line per tag). A single 82485 will directly support a non-sectored 64K data cache or a 128K sectored data cache. Both the 64K and 128K configurations are able to map the entire 4 gigabytes of the i486 microprocessor address space. Multiple 82485 cache controllers may be used to increase cache size. The 82485 interfaces directly to the i486 microprocessor. All i486 CPU bus cycles and timings are supported.

The 82485 also supports 0 wait state processor operation when there is a cache hit and has provisions to support invalidation cycles, BOFF# cycles, and premature BLAST# terminations. The controller is look aside (monitors bus activity in parallel to the processor) and write through (all writes propagate to the system bus), so it supports the same cache consistency mechanisms as the i486 CPU. The controller also provides a safe method to cache ROM BIOS through the use of a write protect pin and a write protect strapping option.

The data cache (Static RAM) resides external to the 82485. The 82485 provides all controls for the SRAMs. No external latches or tranceivers are required. The 82485 output buffers support up to eight SRAMs. A 64K cache can be designed with only five components; nine components for a 128K cache. Two-way set associativity is provided by dual banked SRAMs. Data parity is supported.

The 82485 can be used to design a custom second level cache configuration. For an easier system design and higher integration, the 82485M Turbocache can be used (see data sheet order number 240722). This module is a complete second level cache in one package. It consists of a single 82485 cache controller and SRAM to provide a complete 64K or 128K second level i486 microprocessor second level cache.

2.2 Base Architecture

The 82485 is a look-aside, write-through, 64K or 128K, two-way set-associative cache controller that maps the entire 4 gigabyte address space of the i486 microprocessor. It connects directly to the i486 microprocessor. It has 4096 (4K) tag locations supporting one 16-byte line per tag in a 64K configuration, or two lines per tag in a 128K configuration. Each tag has a write protection bit, each line has a separate valid bit. This section explains the features of the the 82485 architecture.

2.2.1 LOOK-ASIDE IMPLEMENTATION

The 82485 supports a look-aside cache configuration. This means that the signals from the processor are input directly to both the cache controller and memory subsystem in parallel (see Figure 2.1). With this implementation, there is no penalty in the case of a cache miss if memory starts a read in parallel with the cache access. The signals are provided to the memory subsystem as soon as they are available so memory can begin the cycle immediately. Read data is not delayed while the cache controller determines a "hit" or a "miss". If there is a hit in the cache, the memory cycle may be aborted.

A system using the 82485 cache controller works in the following way. When ADS# is asserted by the processor to indicate the start of an external bus cycle, the cache begins a cycle. The memory may begin a cycle in parallel or wait for the 82485 to signal a miss. The 82485 determines if the data being accessed is in the cache. If it is not (a cache miss) the 82485 asserts the START# signal to the memory subsystem. This indicates that the memory subsystem must complete (or begin) the cycle. If the data being accessed is in the cache (a cache hit) then the 82485 does not assert the START# signal and the read data is supplied by the cache memory. See section 4.1.5 for a description of the START# signal.

The look-aside implementation allows both the cache and the memory subsystem to start a cycle simultaneously. This eliminates delays that result from a look-through implementation that forces the cache to determine a hit or a miss before the memory cycle can begin (see Figure 2.2). In the look through implementation, the cache receives the processor control signals first and determines a hit or miss. If there is a cache miss at least one wait state is incurred before the memory subsystem starts a cycle.

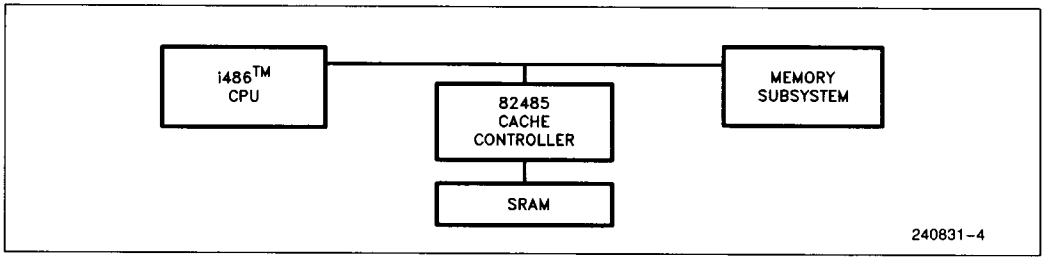


Figure 2.1 Look-Aside Implementation

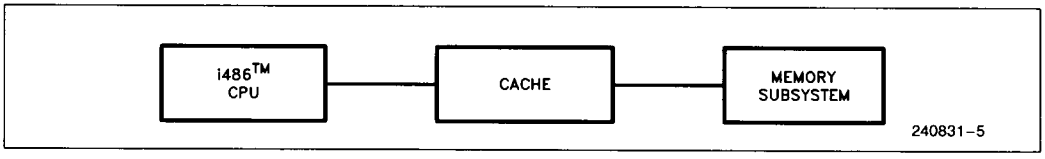


Figure 2.2 Look-Through Implementation

2.2.2 WRITE-THROUGH ARCHITECTURE

The 82485 is a write-through cache controller. This means that every write to the cache is also written out to main memory. For this reason, cache consistency is easily maintained. (Cache consistency is the concept of keeping all copies of a memory location the same. See section 2.3.5 Invalidation Cycles.) Note that the 82485 does not affect bus utilization for write cycles.

2.2.3 INTERNAL ARCHITECTURE

The 82485 cache controller supports a 64K or a 128K cache. In either configuration, the cache is 2-way set-associative with a 16-byte line size. Figure 2.3 shows the 82485 conceptually.

The 82485 contains 2K SETs. Each SET has 2 WAYs. Each WAY in each SET has its own TAG address. In other words, each WAY has 2K TAGs. In the 64K configuration, each tag references one 16-byte line. In the 128K configuration, each tag references two 16-byte lines. It is called SECTORING when one tag references multiple lines. See section 2.2.3.2 for an explaination of sectoring.

2.2.3.1 2-Way Set-Associativity

The concept of two-way set-associativity means that each cacheable address in main memory can be in one of two locations (WAYs) in the cache determined by the SET address. For example, first, the SET address is determined by 11 (2^{11} = 2K) address lines. Once the SET is determined, the two TAG addresses in that SET are compared with the accessed address to determine if there is a match.

2.2.3.2 Sectoring

In the 64K configuration, each TAG references one 16-byte line. This is a non-sectored configuration. In this mode, 17 address lines are used as the TAG address. This enables the entire 4 gigabytes of i486 microprocessor address space to be mapped into the cache (2^{11} set addresses * 2^{17}tag addresses * 16-bytes/line = 4 Gbytes).

In the 128K configuration, each TAG references two consecutive 16-byte lines. This is a sectored configuration since each TAG references more than one line.

Figure 2.4 is an example of one tag in a sectored cache. If this tag points to address 2500H, then the adjacent line is reserved for address 2510H (A4 high). If for example, address 2510H had been written first, the tag would still contain 25H and only address 2500H could be placed in the first line.

In the 128K mode 16 address lines are used as the TAG address. However in this configuration it is necessary to determine which line at the TAG address is being referenced. Address A4 is used for this purpose. Address bit A4 is the line select (LS) input to the 82485 when the controller is configured for a 128K cache. In the 64K configuration, the LS input should be tied low. The 128K configuration also maps the entire 4 Gigabytes of the i486 microprocessor address space (2^{11} set addresses * 2^{16} tag addresses * 2 lines/tag * 16-bytes/line = 4 Gbytes).

The i486 microprocessor address lines used for set, tag, and line select inputs for both the 64K and 128K configurations are shown in Figure 4.2 and explained further in section 4.1.1.1.

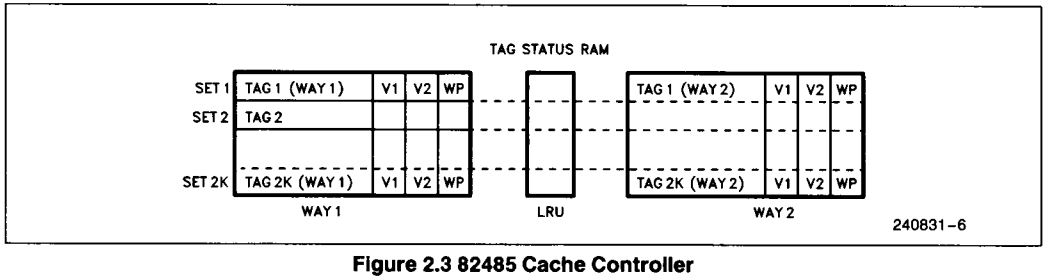


Figure 2.3 82485 Cache Controller

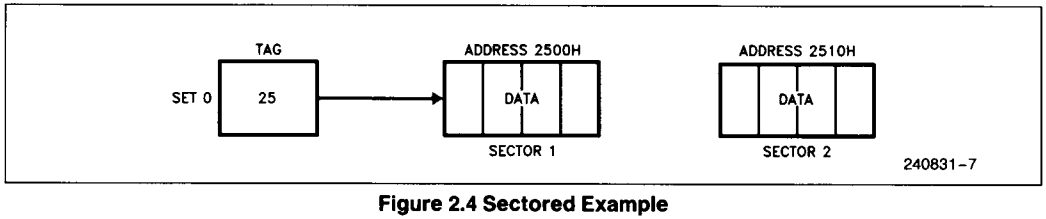


Figure 2.4 Sectored Example

2.2.3.3 LRU Algorithm

The 82485 uses a modified Least Recently Used (LRU) Algorithm to determine which tag in the set should be replaced. This helps keep the most often accessed information in the cache. However, if there is an empty tag, that tag will be filled before another tag is invalidated and replaced. A tag is considered empty when all lines in the tag are marked invalid.

Note that each line in a tag has a valid bit (see Figure 2.3). In the 64K configuration, each tag references one line so there is only one valid bit. In the 128K configuration, each tag references two lines, so there are 2 valid bits per tag.

In summary, when a line is brought into the cache, it gets placed in a WAY based on one of two determining factors (in order of precidence):

1) In the 128K configuration only: The tag is already present
2) According to the LRU bit

2.3 Controller Operation

This section discusses how the 82485 functions. Write cycles, cache read hit cycles, cache read miss cycles, invalidate and BOFF# cycles are all discussed.

Note that the 82485 does not have a BS8# or BS16# input. All transfers are assumed to be 32-bit transfers with valid data on all 32 data lines.

2.3.1 READ HIT CYCLES

In read hit cycles, the 82485 will provide a line of data to the i486 CPU in 5 clocks (see Figure 2.5). In T1 the 82485 begins the tag lookup to see if the read cycle is a hit. Also during T1 the 82485 will assert CKEN# to the processor to indicate to the CPU that this is a cacheable transfer. In each of the next four T2 clocks, the cache will provide 32 bits of data to the CPU. The value of CKEN# in the third T2 indicates whether the data can be cached in the i486 processor. The state of CKEN# in the third T2 is determined by the WPSTRP# pin and whether the line is write protected or not. See section 2.3.4 for an explaination of write protection.

Note that data from the 82485 to the processor is always bursted and is always a 32-bit transfer.

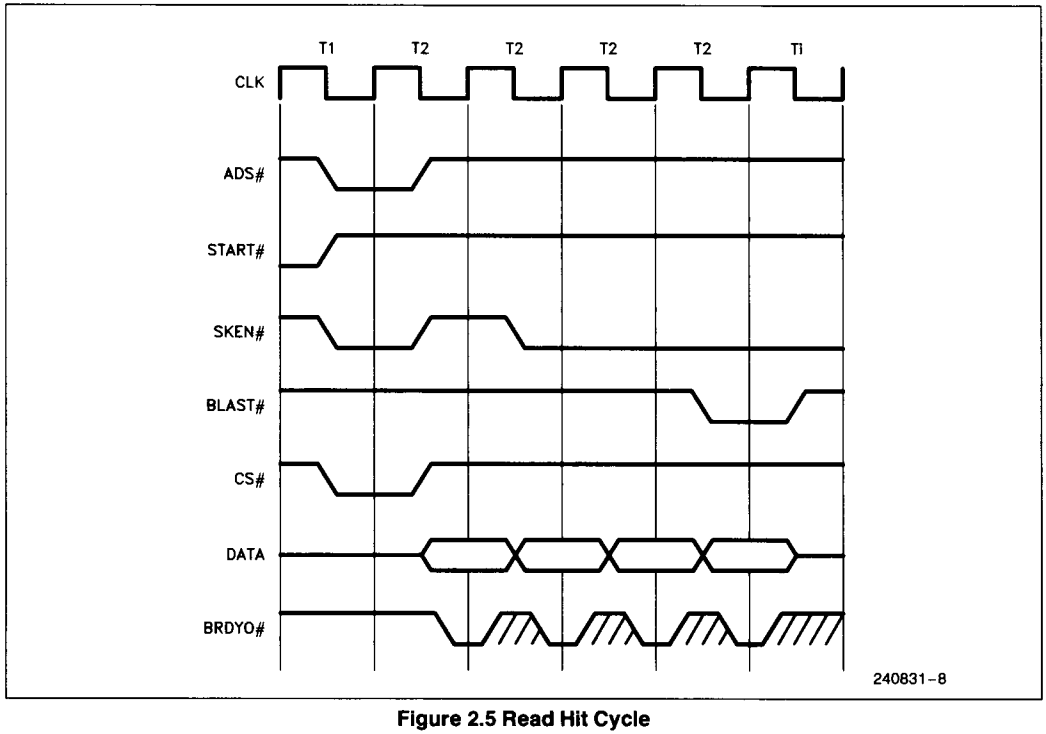


Figure 2.5 Read Hit Cycle

2.3.2 READ MISS CYCLES

In T1, the 82485 begins its tag lookup to determine if the read cycle is a hit. If the address is not present in the cache (a miss), START# is issued to indicate to the memory system that it must service the current cycle. The 82485 is then idle until SKEN#, the 82485's KEN# input is seen active. If SKEN# is inactive the clock before the first transfer from memory begins, the line is non-cacheable and ignored by the 82485.

However, if SKEN# is asserted, the 82485 chooses a free line in the cache or uses the LRU algorithm to invalidate a line in the cache in preparation for a line fill. The data is then returned to the cache and processor simultaneously. If SKEN# is asserted in the clock before the last CRDY# or CBRDY# of the transfer, the line is cached and the valid bit updated. If SKEN# is not asserted for the second time, the line is left invalid. The line is also left invalid if BOFF# or an early BLAST# aborts the cycle.

The Write Protect (WP) bit is sampled during the third transfer of the line fill to determine if the line is write protected. See section 2.3.4 for an explaination of write protection.

During a read miss cycle, the 82485 can not accept data from memory in zero wait states. The earliest data may be returned is the clock after START# is sampled active.

Note that the 82485 is capable of accepting both burst and non-burst line fills from memory (see Figures 2.6 and 2.7). However for performance reasons, data will always be bursted from the cache to the processor. Also note that the 82485 will only perform a line fill if 32-bits of data are transferred at a time. The 82485 does not recognize 8-bit or 16-bit transfers.

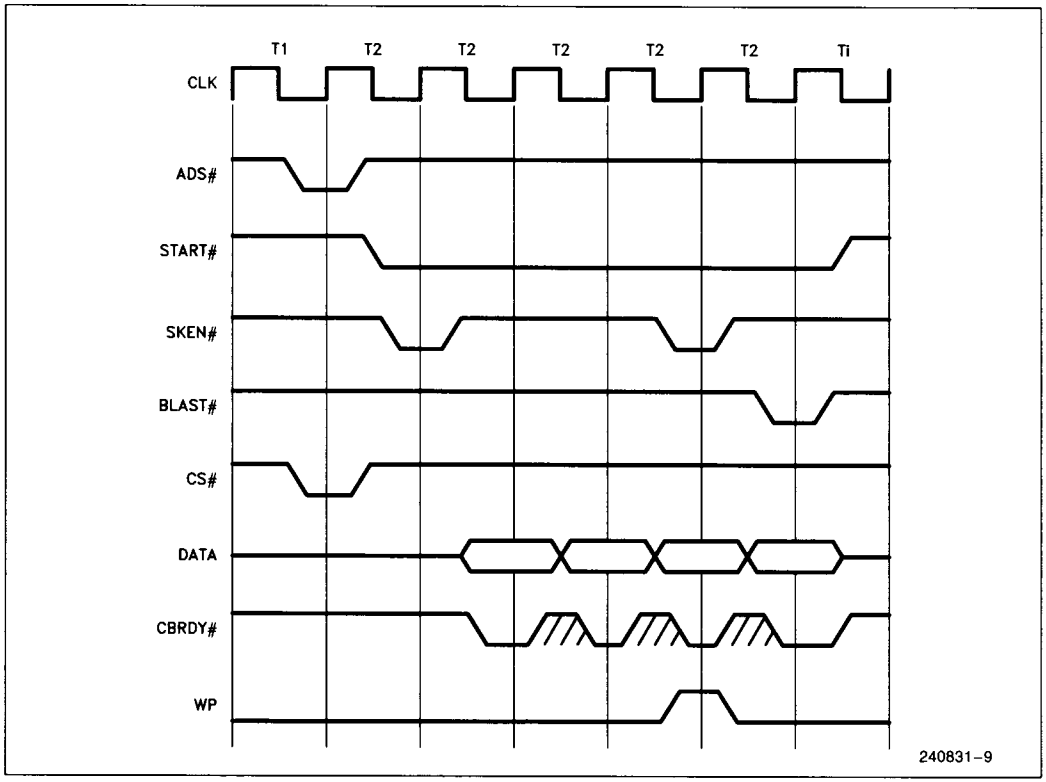


Figure 2.6 Write Protected Read Miss Cycle (Burst)

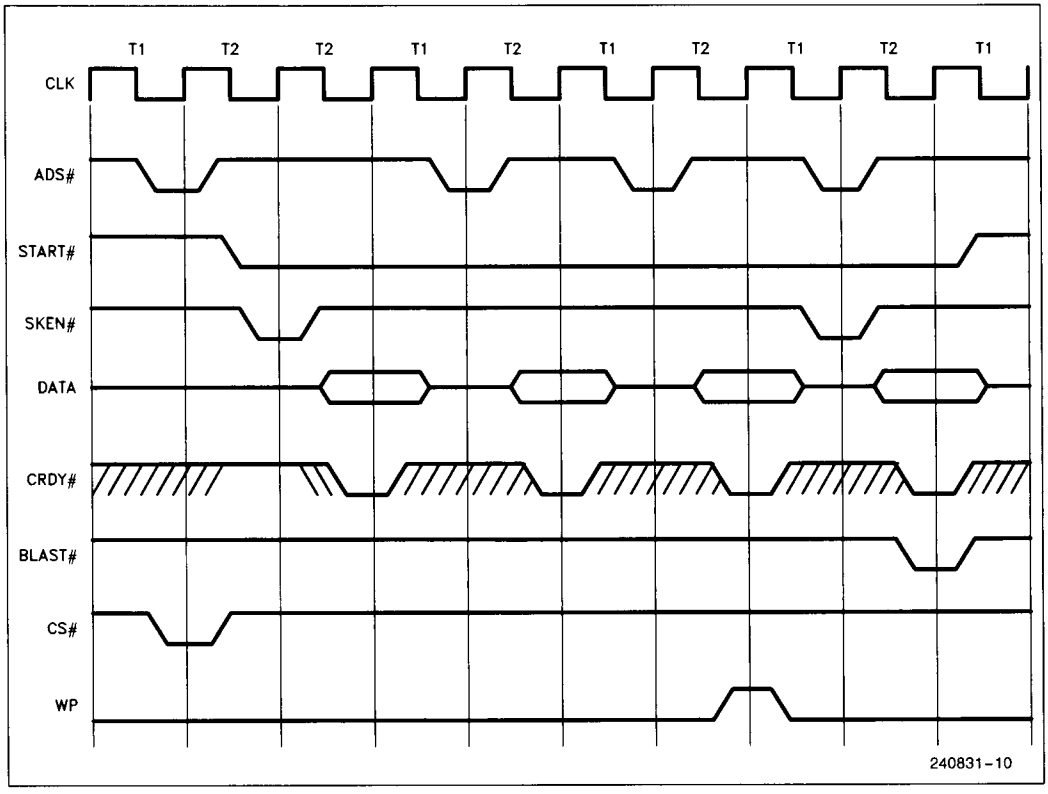


Figure 2.7 Write Protected Read Miss Cycle (Non-Burst)

2.3.3 WRITE CYCLES

Since the 82485 is a write through cache controller all write cycles whether a hit or a miss in the cache are written by the i486 microprocessor to main memory. Writes that are a hit in the cache are updated in the cache as well. As with the read cycles, the 82485 does a tag lookup in T1 to determine a hit or a miss. If there is a cache hit the cache is updated in T2. Write misses do not affect cache contents. Writes to write protected lines do not affect cache contents.

2.3.4 WRITE PROTECTION

The 82485 provides two inputs for write protection: the Write Protect (WP) pin and the Write Protect Strapping Option (WPSTRP#). These pins are used together to allow the safe cacheing of Read-Only-Memory (ROM).

The WP input to the 82485 is sampled with the third data transfer. The value of the pin is stored in the tag for the line being filled. Any line that has its WP bit set is write protected; that is, any writes to that line will be ignored by the 82485. Write protected lines should not be cached in the i486 microprocessor. If WPSTRP# is low, the 82485 will not allow write protected lines to be cached in the i486 processor. If any of these write protected lines are read into the i486 microprocessor internal cache, consistency must be maintained by the system.

The WPSTRP# option when active (low), prevents write protected lines from being cached in the i486 microprocessor internal cache. When the WP and WPSTRP# inputs are used together it is possible to cache read-only memory with no concern that it will be altered by a subsequent write.

There is only one WP bit per tag. This means that each line in the 64K configuration can be write protected. In the 128K configuration, there is one WP bit for two lines.

2.3.5 INVALIDATION CYCLES

An invalidation cycle should be issued whenever a line in the cache needs to be removed from the cache to maintain consistency. For example, an invalidation cycle should be run when a bus master other than the i486 microprocessor writes to a main memory address that is also present in the cache. This prevents the cache from having a different copy of the data than main memory. An invalidation cycle causes the valid bit in the tag of the invalidated line to be cleared.

An invalidation cycle is generated by asserting EADS# to the 82485. When EADS# is asserted, the 82485 will invalidate the address currently on the address pins, regardless of what the 82485 is doing. Chip Select (CS#) must be asserted for an invalidation to occur.

Normally when the i486 microprocessor performs an invalidation cycle, AHOLD, HOLD or BOFF# is asserted to allow another bus master to drive the address of the line to be invalidated. However, it is not necessary to tristate the address bus. An invalidation cycle that is performed on the current address driven by the processor is called a self-invalidation and is supported by the 82485.

The 82485 can support an invalidation cycle every other clock regardless of what the controller is doing. There may be a performance penalty, however, if EADS# is asserted at a time when the tag memory is in use. Since the 82485 tags are single ported, only one tag access per clock is allowed.

Figure 2.8 shows a read miss cycle with an invalidation occurring in the third transfer of a line fill. Under normal conditions, the 82485 would, on the next clock, validate the current line that is being filled. However, since EADS# occurs here, the tagram is occupied on the next clock with a tag lookup to see if the invalidate is a hit so the current line is not validated. If it is a hit, the following clock is used to perform the actual invalidation. The next clock is spent validating the current line fill. Should the i486 microprocessor begin a cycle immediately, the 82485 is not able to perform its tag lookup until the tag memory is free (delayed by two clocks). This causes START# to be delayed, and subsequently a memory cycle to be delayed. For greatest performance, EADS# should not be issued in the second, third or fourth transfer of a cache line fill.

Note that if a self-invalidation occurs in T1, ADS# and EADS# are sampled at the same time and the 82485 will invalidate the line and assert START# as in a normal read miss cycle. If EADS# is asserted at any other time during a read hit, START# is not asserted.

2.3.6 BOFF# CYCLES

When BOFF# is asserted to the i486 microprocessor, the CPU will float its address, data and status pins in the the next clock. Any bus cycle in progress is aborted. The bus remains in the high impedance state until BOFF# is negated. When BOFF# is negated, the i486 processor restarts the aborted bus cycle by re-driving the address and status and asserting ADS#. If BOFF# is asserted during a burst cycle to the processor, the processor will ignore data returned only on the clock that BOFF# was asserted. Data from previous cycles will still be valid.

When BOFF# is asserted to the 82485, the controller, like the processor, will relinquish the bus in the next clock. While BOFF# is asserted, as any other time, the 82485 monitors EADS# to perform any invalidation cycles.

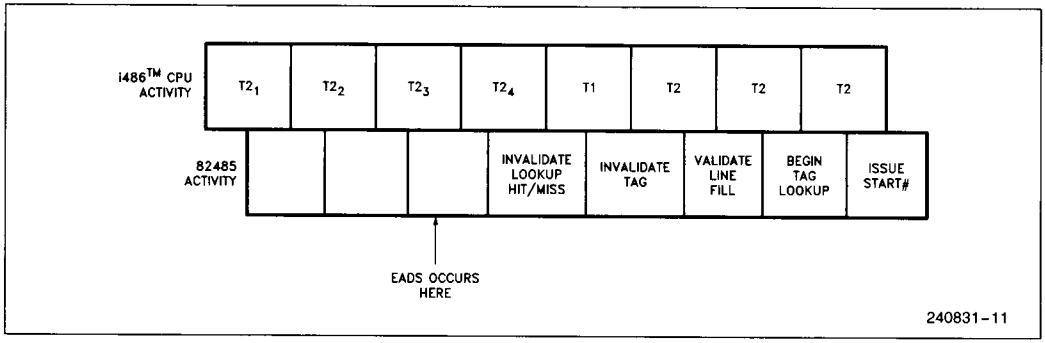


Figure 2.8 Invalidation During Read Miss

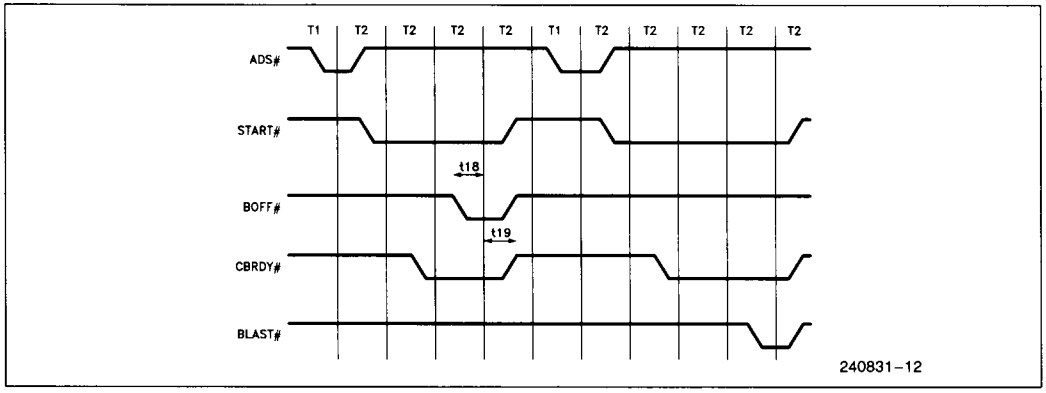


Figure 2.9 Aborted Line Fill

If BOFF# is asserted during a cache read hit (data is being transferred from the cache to the CPU), the 82485 invalidates the line being transferred. Once BOFF# has been released and the cycle resumes, the 82485 sees this as a cache miss and the memory system must supply the remaining data.

If BOFF# is asserted during a cache read miss (memory is transferring data to cache and CPU), the 82485 will treat the cycle like an aborted fill and the line will remain invalid. Once BOFF# is released and the cycle restarted, the remainder of the line fill will be seen as a partial line fill (BLAST# will be asserted before four transfers complete) and will remain invalid.

Figure 2.9 is an example of an aborted line fill. Since the line transfer is interrupted before the transfer completes, it stays invalidated. Once the transfer resumes, the 82485 sees a new cycle begin with ADS#, but it completes with BLAST# after three transfers. It treats this as an aborted line fill cycle, and the cycle is never validated.

Asserting BOFF# in the same clock as ADS# will cause the i486 CPU to float its bus in the next clock and leave ADS# floating low. Since ADS# is floating low, a peripheral device may think that a new bus cycle has begun even though the cycle was aborted. The 82485 handles this circumstance in most cases since an active ADS# in the clock BOFF# is deasserted is ignored. The only circumstance that must be handled by the system is as follows:

BOFF# is asserted in T1, and before BOFF# is deasserted, HOLD is asserted and remains asserted after BOFF# is deasserted (see Figure 2.10). In this circumstance it is necessary for the system to assure that ADS# is either driven to a valid level or pulled high in the clock after BOFF# is deasserted (meeting the 82485 ADS# setup time).

There are several ways to avoid this system restriction:

1) Do not assert BOFF# in T1.
2) Use a "two clock" backoff: in the 1st clock AHOLD is asserted and in the 2nd clock BOFF# is asserted. This guarantees that ADS# will not be floating low.
3) Do not assert HOLD when BOFF# is asserted.

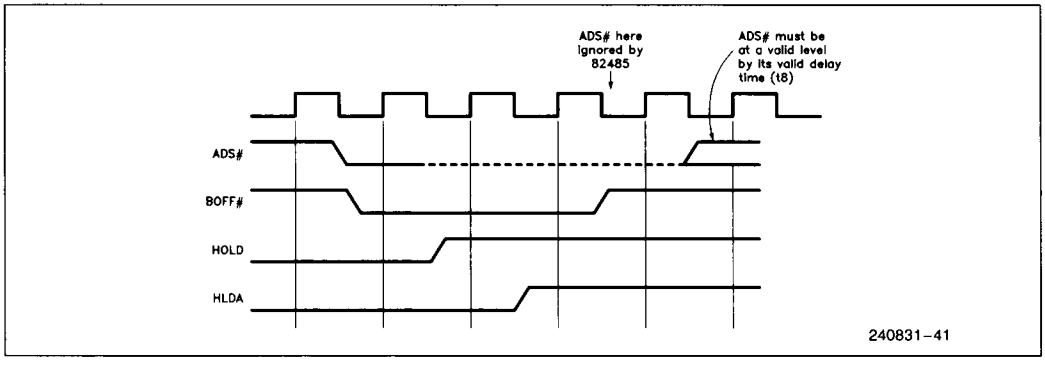


Figure 2.10 BOFF# Asserted in T1

2.4 Incompatibilities

Below is a list of some special considerations that the 82485 requires to be designed into an i486 CPU system. They have been summarized to point out any possible inconsistencies between the i486 CPU specification and the 82485 specification:

1) Invalidation cycles may not be performed every clock. Unlike the i486 microprocessor, the 82485 only allows EADS# assertion every other clock at most.
2) The minimum clock high voltage is slightly higher than the i486 microprocessor specification. It is still within TTL levels.
3) During i486 microprocessor non-cacheable, non-burst, code prefetch cycles the processor can issue a HLDA. The 82485 does not receive the HLDA signal and may not realize that another bus master is driving the bus. If a bus master other than the i486 CPU is allowed to drive the ADS# input to the 82485, the HLDA signal from the CPU must be gated into the BOFF# input of the 82485 (see section 4.1.1.1).

3.0 SYSTEM INTERFACE

The following section describes the basic connection of the 82485 in an i486 microprocessor system. The microprocessor bus connection, the memory bus connections and the 82485 to SRAM connections are highlighted.

A typical interface is shown in Figure 3.1. All of the signal the i486 processor generates "feed around" the 82485 cache. That is, they go to both the 82485 and the memory controller in parallel. In turn most of the memory generated signals feed-around the 82485 back to the processor. This is what makes the 82485 a look- aside cache controller. All of the signals the 82485 encounters are described below.

3.1 Control Signals

This section describes each of the 82485 control signals. ADS#, M/IO#, W/R#, RESET, BOFF#, and BLAST# connect directly to the corresponding processor signals and have the same function as the processor pins. BRDYO# and START# are outputs of the 82485, while CS# is an input to the 82485.

3.1.1 ADS#, W/R#, M/IO#

These processor control signals are used by the 82485 to indicate the start of a new cycle, and identify the type of cycle. ADS# assertion indicates a T1 cycle and initiates the tag lookup process in the 82485. I/O cycles are ignored by the 82485.

ADS# is the primary signal that activates the 82485 cache controller. When ADS# goes low, the 82485 begins the hit/miss tag lookup regardless of the state of Chip Select (CS#). For this reason, any bus master that controls the ADS# input to the 82485 must meet the 82485 address bus setup and hold times, regardless of the state of CS#. Chip Select, when inactive, disables the 82485 outputs only. Note that Chip Select must be asserted for invalidation cycles.

3.1.2 RESET

RESET is an asynchronous input that causes the 82485 to reset its internal machines to a known state: its entire cache contents invalidated, and expecting the start of a new bus cycle. RESET must be high for at least 10 clocks on a warm reset for the 82485 to reset properly. On power-up, RESET must be high for at least 100 clocks. There must be no bus activity for at least 4 clocks after the falling edge of RESET.

3.1.3 BOFF#

Once BOFF# is sampled by the 82485, it relinquishes control of the data bus by the next clock. If a read hit line transfer was in progress, that line is marked invalid and the transfer would NOT continue once BOFF# was released. Once BOFF# was released and the cycle resumes, the 82485 sees this as a cache miss and the memory system must supply the remaining data. If a read miss transfer was interrupted by BOFF# the 82485 would leave the line invalid even if the transfer continues one BOFF# has been released. The 82485 will recognize invalidations during BOFF#, but will only do so if CS# is active. See section 2.3.6 for further explaination.

3.1.4 START#

START# is a signal asserted by the 82485 to indicate that the memory subsystem must process the current cycle. START# is always driven and is asserted for all read miss cycles and memory write cycles. START# is not activated for I/O cycles so M/IO# may need to be included in the logic that receives START#. START# is also not activated if CS# is sampled inactive along with ADS#. If locked cycles do not generate CS#, LOCK# needs to be included in the logic that receives START#.

START# is normally active in the first T2, but may be delayed if an invalidation cycle forced the previous cycle to be extended (see Invalidation Cycles).

3.1.5 BRDYO#

The 82485 generates BRDYO# when it is bursting data back to the i486 microprocessor during read hit cycles. BRDYO# is always driven (not an open collector output) and should be used by external logic to create the BRDY# input signal to the i486 processor. Since the 82485 is a zero wait state, single clock burst cache controller, BRDYO# is activated in the first until the fourth T2 unless the cycle is interrupted or delayed.

3.1.6 CRDY#, CBRDY#

CRDY# and CBRDY# are the ready and burst ready inputs to the 82485. They should behave exactly like the i486 RDY# and BRDY# inputs. CBRDY# should be used in conjunction with BRDYO# to generate the i486 microprocessor's BRDY# input. Likewise, CRDY# should be used to form the i486 processor RDY# input.

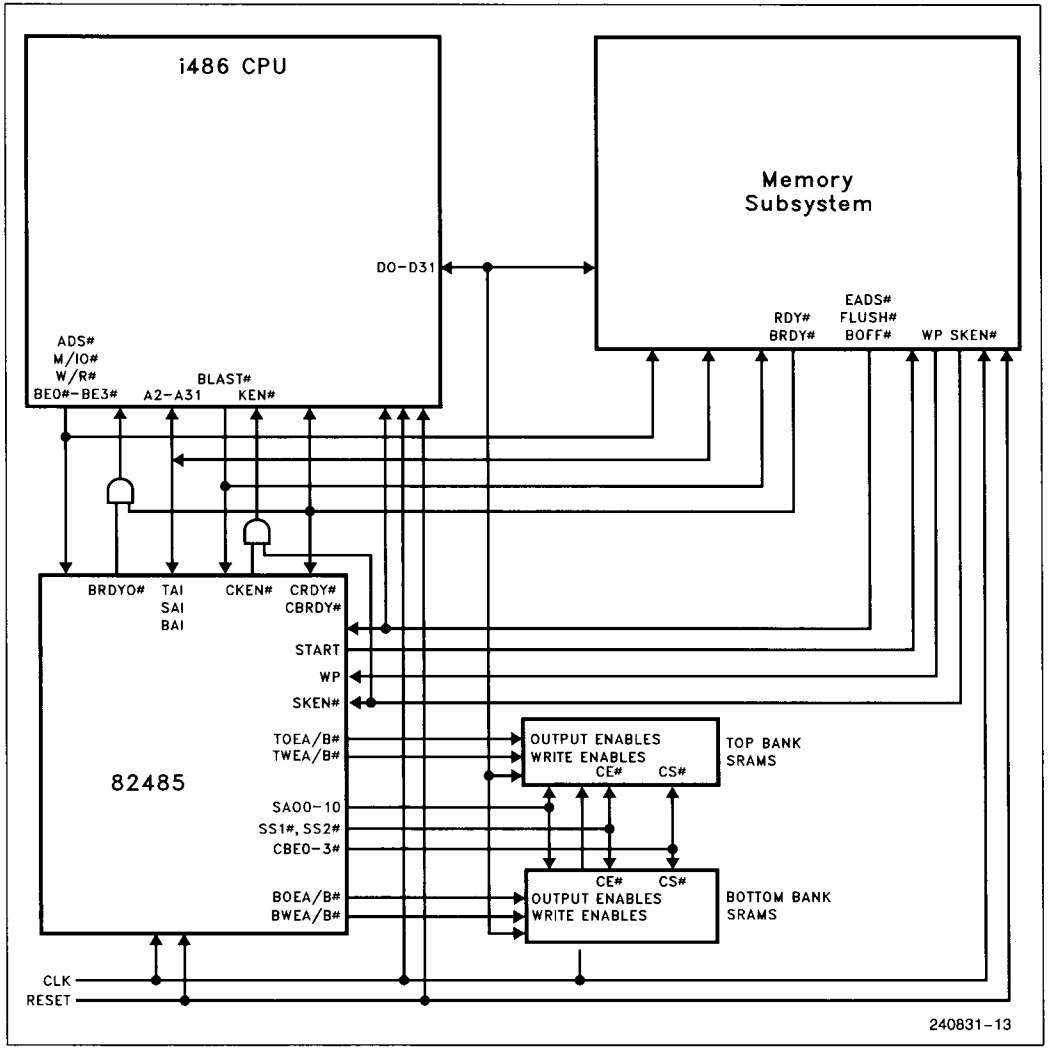


Figure 3.1 82485 Typical Configuration

The 82485 does not sample the CBRDY# or CRDY# inputs during read hits, so it is not possible to artificially add wait states to the 82485's burst transfer to the processor. The CBRDY# and CRDY# inputs must follow 82485 setup and hold times even outside the sampling window.

3.1.7 BLAST#

BLAST# is used by the 82485 to indicate the end of a transfer. It is sampled at the end of each T2 with CRDY# or CBRDY#. If BLAST# is asserted with CRDY# or CBRDY# before the end of a cache line fill from a read miss, that transfer is left invalid by the 82485. If BLAST# is asserted with BRDY0# before the end of a read hit cycle, a partial line will be transferred, and no invalidation will take place.

3.1.8 CHIP SELECT CS#

Chip Select is used to select the proper 82485 controller if multiple controllers are used. Otherwise, if only one controller is used, CS# may be grounded. CS# is generated by decoding the lowest order tag addresses input to the 82485. For example, two 82485s configured for a 128K cache would decode A16 for their chip selects. A16 high would select controller 1, A16 low would select controller 2. The following table summarizes the addresses used for decoding:

Size	# of 82485s	Address Bit(s) to Decode
64K	2	A15
64K	4	A15, A16
128K	2	A16
128K	4	A16, A17

If desired, A16 and A17 may be decoded for 64K modules. Performance may be increased because of increased granularity, however, if A15 and A16 are used.

With CS# inactive, invalidation cycles are ignored, START# is inactive, and CKEN# is inactive. CKEN# does however, always activate in T1 because it is not possible for the 82485 to recognize CS# before then.

If required, the LOCK# signal may be used as a term in the creation of CS#. If locked cycles do not generate CS#, START# must be generated externally so memory may handle the cycle.

3.2 Cacheability Signals

This section describes the signals that determine which address locations can be cached in the 82485 and the i486 microprocessor. FLUSH# and EADS# connect directly to the processor signals and have the same function as the processor pins. Note that the 82485 does not decode the software generateed flush condition and generate the FLUSH# signal to the 82485.

3.2.1 CPU CACHE ENABLE CKEN#

CKEN# is generated by the 82485 to indicate that its current transfer, during a read hit cycle, is cacheable to the i486 microprocessor. It is always driven (not an open collector output) and must be used as one of the terms that generates KEN# to the i486 microprocessor. CKEN# is always active in T1, but then goes inactive and remains inactive unless the cycle is a read hit cycle.

For read miss and write cycles, CKEN# goes inactive in T2 and remains inactive until the next T1. It is the responsibility of the system to generate the KEN# signal to the processor in these cases.

In a read hit cycle, CKEN# goes active again in the second T2 and remains active throughout the cycle. This forces external KEN# logic to activate KEN# and make the cycle cacheable to the i486 microprocessor. However, if the line being transferred is write-protected, **and** the WPSTRP# pin is tied low, CKEN# stays inactive in the second T2 and remains inactive throughout the cycle. This allows write protected lines in the 82485 to be cacheable only in the 82485.

3.2.2 SYSTEM CACHE ENABLE SKEN#

The SKEN# input to the 82485 is similar to the KEN# input to the i486 microprocessor. It is sampled just like KEN#, the clock before the first and last transfers of a line fill, to indicate whether the line is cacheable. If the KEN# input to the i486 microprocessor is connected to the SKEN# input of the 82485, the CPU internal cache and the 82485 cache will cache the same items. It is possible to control KEN# and SKEN# separately so the 82485 cache and the i486 microprocessor cache different areas of memory (see section 4.1.4).

When SKEN# is asserted, the 82485 chooses a free line in the cache or uses the modified LRU algorithm to invalidate a line in the cache in preparation for a line fill. Therefore if SKEN# is asserted before the first transfer, but not before the last transfer, a line may have been unnecessarily invalidated.

SKEN# is a synchronous input and must meet setup and hold times regardless of whether it is being sampled or not.

3.2.3 FLUSH#

The 82485 FLUSH# input behaves exactly like the i486 microprocessor input. Once asserted, FLUSH# will invalidate the entire contents of its cache memory regardless of the state of CS#. While FLUSH# is asserted, the 82485 continues to track CPU bus cycles and treats all accesses as cache misses, activating START# appropriately.

FLUSH# may be used asynchronously with both the i486 processor and the 82485. If the proper pulse widths are given, FLUSH# will be recognized, but it is possible that the FLUSH# will be recognized on different clock edges for each device. This may happen if FLUSH# assertion or deassertion is near its setup and hold times when one device may recognize it and the other may not.

3.2.4 WRITE PROTECT WP

The write protect input is an active high input that indicates to the 82485 that the current line transfer is write-protected. It is sampled with the third CRDY# or CBRDY# of the transfer. The 82485 saves this information as a single bit in each tag location. In the 128K configuration where there is a single tag for 2 consecutive lines, the write protect bit is valid for both lines. If a location has been write-protected, writes to that location will be ignored.

WP is a synchronous input and must meet setup and hold times regardless of whether it is being sampled or not.

3.2.5 WRITE PROTECT STRAP WPSTRP#

It indicates whether write protected items in the 82485 should be cacheable in the i486 microprocessor internal cache. If WPSTRP# is high, CKEN# will go active in the second T2 during all read hit cycles to indicate that they are cacheable. If WPSTRP# is low, CKEN# will be inactive in all T2 clocks for read hit cycles to locations that are write-protected. This allows write protected lines to be cached by the 82485 cache and not by the i486 microprocessor.

3.2.6 EADS#

EADS# assertion causes the 82485 to invalidate the address present on the address bus if CS# is seen active. AHOLD need not be asserted, nor is it even used as an input to the 82485. EADS# may be asserted at most once every other clock as that is the fastest 82485 invalidation rate. The section titled "Invalidation Cycles" describes where EADS# may be asserted for maximum performance.

3.3 Address Signals

This section describes how the address signals from the i486 microprocessor interface to the 82485.

3.3.1 BURST ADDRESS IN BAI0–AI1

Burst Address In Zero and One. These signal are direct connections to A2 and A3 of the i486 microprocessor address bus. They are used to determine the starting burst address, and in which SRAM bank the accessed address is located.

3.3.2 SET ADDRESS IN SAI0–SAI10

Set Address In Zero through Ten. These are the 11 set address inputs to the 82485. In the 64K configuration, these pins are direct connections to A4–A14 of the i486 microprocessor address bus. In the 128K configuration these pins are direct connections to A5–A15 of the i486 microprocessor address bus. The address on these pins determines in which set the requested data could be located. The set address is latched by the 82485 cache controller.

3.3.3 TAG ADDRESS IN TAI0–AI16

Tag Address in Zero through Sixteen. These are the 17 tag inputs to the 82485. In the 64K configuration, these pins are direct connections to A15–A31 of the i486 microprocessor address bus. In the 128K configuration, A16–A31 of the i486 microprocessor address bus are direct connections to TAI0–TAI15. TAI16 is not used and must be tied high or low. The address on these inputs is compared with the 82485 tagram to determine if there is a hit or miss in the cache. The tag address is latched by the 82485 cache controller.

3.3.4 LINE SELECT LS

The line select bit is used in the 128K sectored configuration only. It is used to determine which of the two lines in the accessed tag is the requested line. In the 128K configuration, LS is a direct connection to A4 of the i486 microprocessor address bus. LS should be tied low in the 64K configuration.

3.3.5 BE0#–BE3#

The Byte Enable inputs are direct connections to BE0# through BE3# of the i486 microprocessor address bus. The byte enables are used to complete partial byte or word writes to the 82485 on cache write hit cycles. They are forced low during read hit cycles and during line fills on a read miss. All transfers from main memory to the 82485 and from the 82485 to the processor are dword (32-bit) transfers.

3.4 SRAM Interface Signal

This section describes the interface signals between the 82485 and the cache RAM (SRAM). The data bus and the data parity bits are not used by the 82485 and are direct connections from the i486 microprocessor to the cache SRAMs. All signals described here are outputs from the 82485 cache controller.

3.4.1 BURST ADDRESS OUT BAOT/BAOB

In a cache design with the 82485 there must be two banks of SRAM, a top bank and a bottom bank (see chapter 4). The Burst Address Out Top (BAOT) output of the 82485 is the A0 input to the top bank of SRAMS. The Burst Address Out Bottom (BAOB) output of the 82485 is the A0 input to the bottom bank of SRAMs. During the first transfer of a cache line read or fill, BAOT and BAOB are both identical to BAI1 (A3 from the processor) so the bank in which there is a hit can be determined. After the first transfer, the 82485 controls the value of BAOT and BAOB depending on where the first cache access was a hit.

3.4.2 SET ADDRESS OUT SAO0-SAO10

Set Address Out Zero through Ten. These pins drive the SET address to the external cache RAM (pins A1-A11 of the SRAM). The set address out value is the value latched at the SAI inputs to the 82485.

3.4.3 SECTOR SELECT SS1#-SS2#

The Sector Select outputs are used to support the sectored 128K configuration. In the 128K configuration, they provide the decoding to the cache RAM to determine which line in the tag is being accessed and activate the appropriate SRAM chip enables. In the 64K configuration, SS1# is used for the chip enable to all SRAMs. SS2# is not used in the 64K configuration and should not be connected.

3.4.4 CACHE BYTE ENABLE CBE0#-CBE3#

The cache byte enable signals to the cache RAM control which bytes are written for partial dword write cycles when there is a cache write hit. These pins reflect the BE0-BE3# input values during write cycles. The byte enables are all active for read hit and cache line fill cycles. All transfers from main memory to the 82485 to the processor are dword (32-bit) transfers.

3.4.5 TOP AND BOTTOM OUTPUT ENABLES TOEA#-TOEB, BOEA#- BOEB#

These signals are direct connections to the output enables of each bank (top and bottom) of the cache RAM and each bank (A and B) within each SRAM. They select which RAM bank will be enabled onto the data bus during a read hit cycle. These signals are required to support the i486 microprocessor burst mode as well as the two-way set-associative architecture of the 82485. The top and bottom banks of SRAM support the i486 microprocessor burst mode. The A and B banks within the SRAM support the controllers two-way set- associative architecture. See section 4.3.1 for further explanation.

3.4.6 TOP AND BOTTOM WRITE ENABLE TWEA#-TWEB#, BWEA#-BWEB#

These signals are direct connections to the write enables of each bank (top and bottom) of the cache RAM and each bank (A and B) within each SRAM. These enables cause the contents of the write data latch (internal to the SRAM) to be written to its data array during write hit cycles and read miss cycles. These signals are also required to support the i486 microprocessor burst mode and the two-way set-associative architecture of the 82485. See section 4.3.2 for further discussion.

4.0 SYSTEM CONFIGURATIONS

This section will discuss the design considerations that should be taken into account when designing an i486 microprocessor system with an 82485 cache controller. First, the 82485 interface to the processor will be discussed for both single and multiple 82485 configurations. Also the 82485 interface to the cache RAM will be examined in detail.

4.1 Single 82485 Interface to the i486 Microprocessor

In a single 82485 configuration, little or no extra logic is required to include the cache. Most of the 82485/CPU interface signals are a direct connection from one device to the other. The cache SRAM data pins normally connect dirctly to the CPU data bus.

4.1.1 i486 MICROPROCESSOR SIGNALS

The following signals are simply connected from the i486 microprocessor to the 82485: ADS#, W/R#, M/IO#, BE0#-BE3#, BLAST#, RESET, CLK, FLUSH#, and EADS#. Depending on the system configuration, BOFF# may be included in the above list (see section 4.1.7)

4.1.1.1 Address A2–A31

The address bus of the i486 microprocessor is also directly connected to the 82485. However, the connections may not be intuitive (see Figure 4.1). Address bits A2 and A3 from the processor should be directly connected to the Burst Address In (BAI) inputs of the 82485. Address A2 is connected to BAI0, A3 is connected to BAI1. The remaining connections depend on what size cache is implemented.

In the 64K configuration, A4–A14 of the processor address bus are the Set Address In (SAI0–10) inputs to the 82485. Address A15–A31 are the Tag Address In (TAI0–16) inputs to the 82485. Since sectoring is not used in the 64K configuration, LS is tied low.

In the 128K configuration, A4 is used as the line select pin and should be directly connected to the LS input of the 82485. Addresses A5–A15 of the processor address bus are used as the SAI0-SAI10 inputs. The remaining address pins A16–A31 are the TAI0–TAI15 inputs to the 82485. TAI16 is not used in the 128K configuration and must be tied high or low.

4.1.2 CRDY#, CBRDY#

The memory ready signals, CRDY# and CBRDY# are connected directly from the memory system to the 82485, but are combined with the other system ready signals to form the i486 microprocessor RDY# and BRDY#. The 82485 BRDYO# output must be logically ANDed with CBRDY# and other burst ready signals to form BRDY# to the processor.

4.1.3 WP, WPSTRP#

The memory system must generate the WP input. If write protection is not needed, WP may be tied to VSS and WPSTRP# tied to VCC. If the system needs to prevent write protected lines in the 82485 from being cached in the i486 microprocessor, WPSTRP# must be tied to VSS.

4.1.4 KEN#, SKEN# GENERATION

For performance reasons, it is desirable to allow the contents of the second level cache to be cacheable in the processor on-chip cache. Since the 82485 activates CKEN# only during a read hit cycles, the CKEN output may be ANDed with the system cache enable signal to form KEN# to the i486 microprocessor (see Figure 4.2). This will result in all (WPSTRP# high) cache hits to the 82485 being cached in the i486 microprocessor internal cache. If the 82485 and the processor will cache the same areas of memory, the KEN# input to the processor may be tied to the SKEN# input of the 82485.

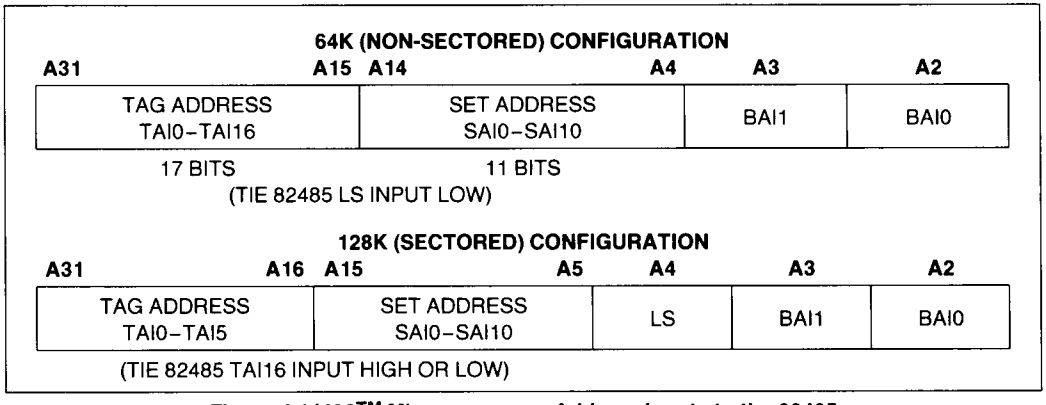


Figure 4.1 i486™ Microprocessor Address Inputs to the 82485

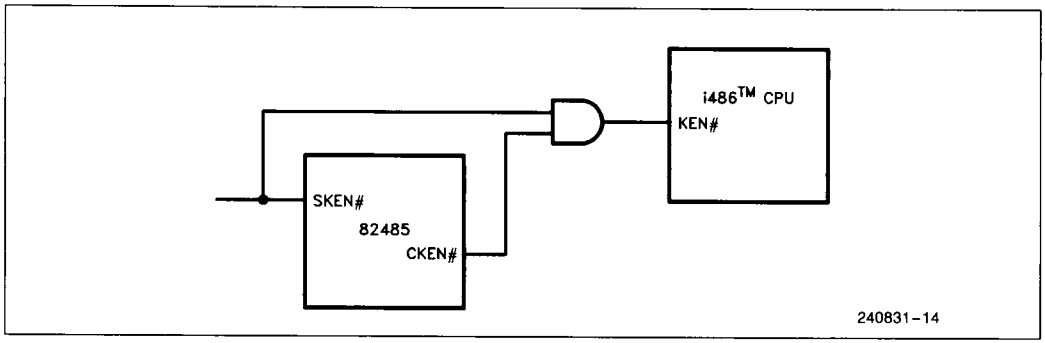


Figure 4.2 CPU and 82485 Cache Same Memory Locations

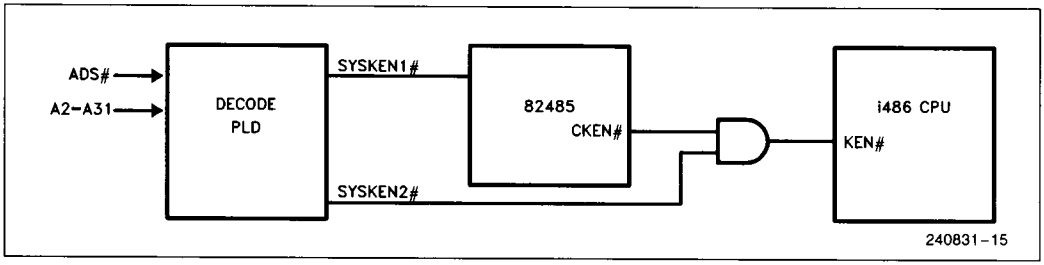


Figure 4.3 CPU and 82485 Cache Different Memory Locations

If the 82485 and the processor cache different areas of memory, the memory system can generate two cache enable signals: one that is ANDed with CKEN# to produce KEN# to the processsor, and another for the SKEN# input to the 82485 (see Figure 4.3).

4.1.5 START# GENERATION

START# goes low to indicate that the memory system must complete the current cycle. This is true for all memory writes and read misses. It is the system's responsibility to recognize I/O cycles and begin an I/O access without waiting for START#. START# is not generated if CS# is inactive. If CS# is inhibited by any signal, that signal should be encoded into the START# logic.

START# is asserted in the first T2 but may be delayed if there was an invalidation in the previous cycle. Because the assertion of START# maybe somewhat unpredictable, it is recommended that START# be used to either begin a DRAM RAS cycle, or enable DRAM output buffers.

Figure 4.4 shows how START# may be the indication to DRAM control to begin a cycle. Once START# is sampled active, a RAS and CAS cycle begin. This will incur an extra wait state to cache read misses since the earliest a memory cycle will begin is the first T2.

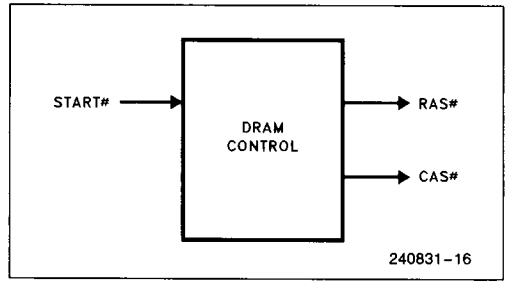


Figure 4.4 START# to Begin a DRAM Cycle

Figure 4.5 shows that START# may enable DRAM data buffers. The actual DRAM cycle begins once ADS# and M/IO# are sampled low, but will not complete until the buffers have been gated allowing data to be written to the i486 microprocessor data bus. If the cycle is a read hit, the buffers are never enabled. Since the 82485 takes 5 clocks to complete the burst transfer, RAS precharge time can be easily absorbed.

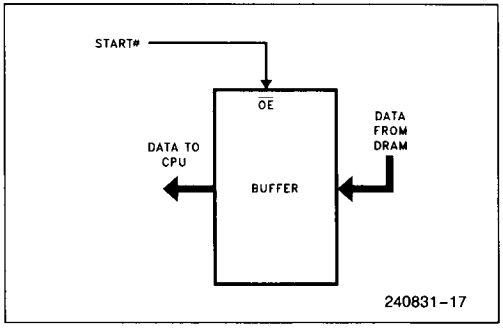


Figure 4.5 START# to Enable Data Buffers

See "START# Predictability" under "Performance Considerations" for detailed information on how START# may be asserted in a predictable manner.

4.1.6 CS# GENERATION

In a single 82485 system the CS# input can be tied low if the memory is owned by the processor bus. If memory is owned by the peripheral bus, locked cycles should not be cached. In this case LOCK# would have to be encoded into the chip select and START# logic.

4.1.7 BOFF# INPUT TO THE 82485

In most systems, the BOFF# signal to the processor can be connected directly to the 82485. This is always true if the i486 microprocessor is the only device driving the ADS# signal. However, if a bus master other than the i486 CPU is allowed to drive the ADS# input to the 82485, it may be necessary to gate the HLDA output of the processor with the system BOFF# into the 82485 BOFF# input (see Figure 4.6). This is only necessary in the circumstance when the i486 microprocessor gets interrupted in the middle of a transfer by a HOLD/HLDA pair and another bus master is allowed to drive the ADS# input to the 82485. The 82485 does not otherwise receive the HLDA signal and does not realize the i486 has relinquished its data bus. If another bus master takes over the bus, the 82485 may assume the i486 microprocessor is still driving the bus and continue to cache data. Since an i486 microprocessor transfer can only be interrupted by a HOLD/HLDA pair in non-cacheable, non-burst code prefetch cycles, this is not likely to affect the 82485. The 82485 is not affected if 1) the i486 CPU is the only device that drives ADS# to the 82485 OR 2).

The 82485 only caches burst transfers (CBRDY# is always returned, not CBRDY#) OR 3) The 82485 only caches data that is cacheable in the i486 microprocessor internal cache. If the 82485 is not affected, there is no need to gate HLDA into the controller and BOFF# can be connected directly from the processor to the 82485.

4.2 Multiple 82485 Interface to the i486 Microprocessor

The multiple 82485 scheme is similar to the single 82485 cache scheme. All i486 microprocessor connections remain the same. The only difference is that now they are connected to the processor, memory system and ALL 82485s in parallel.

The same ready and burst ready outputs from the memory system connect to the CRDY# and CBRDY# inputs to all 82485 controllers. The CBRDY# input is then ANDed with the BRDYO# outputs from all 82485s to form the BRDY# signal to the i486 microprocessor (see Figure 4.7).

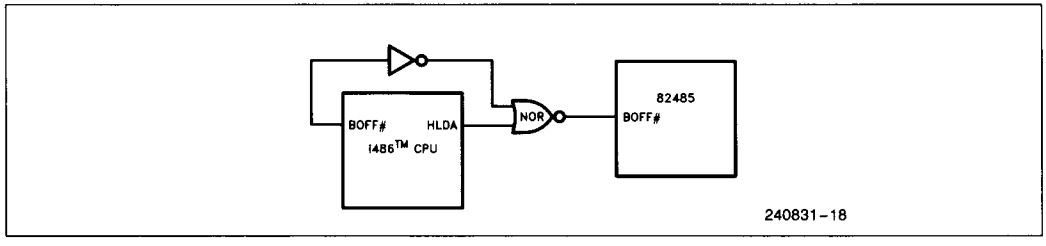


Figure 4.6 82485 BOFF# Generation

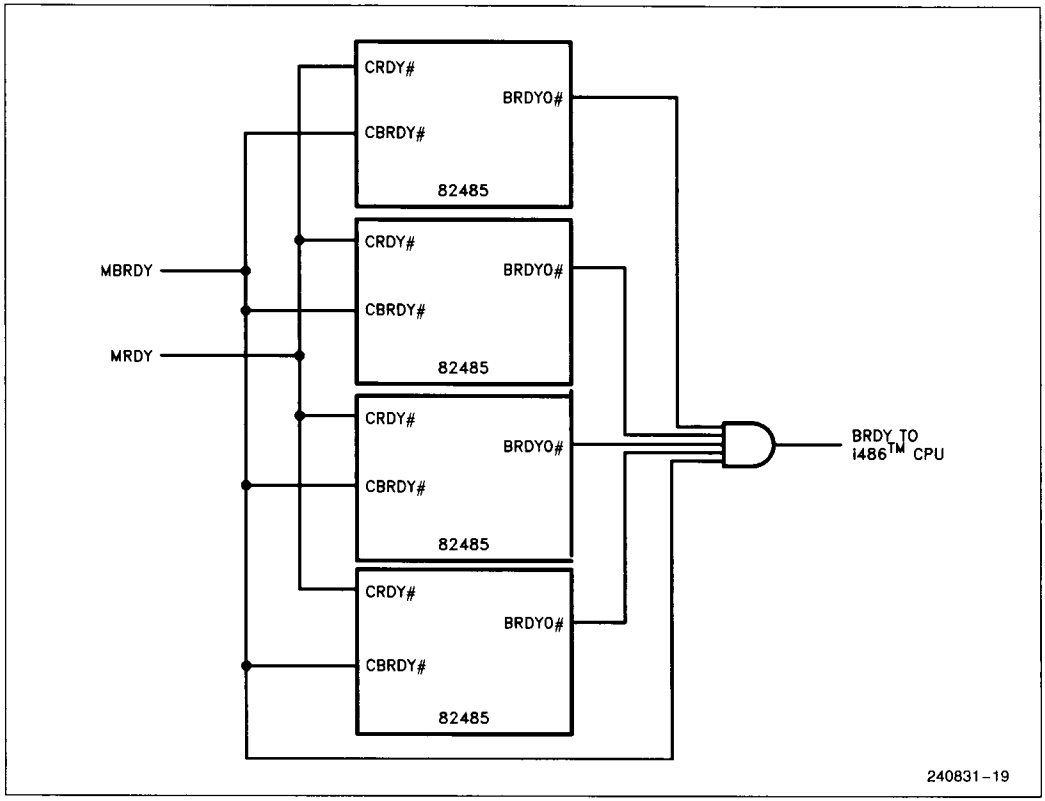


Figure 4.7 Multiple 82485 CRDY#, CBRDY#, BRDYO# Generation

4.2.1 START# GENERATION

START# is activated by a single 82485 at a time because CS# is active for a single 82485 at a time. START#, therefore, may be ANDed with all other START# signals to form the system START# indication. See section 4.1.5 for details on how START# may be used. Figure 4.8 shows an example of a multiple cache configuration.

4.2.2 KEN#

Like START#, CKEN# is only activated for chip selected modules. Therefore, all CKEN# outputs may be ANDed together to form the i486 microprocessor KEN# signal. A system cache enable signal must also be included in the AND terms since it is the system's responsibility to generate KEN# during read miss cycles. Figure 4.8 illustrates this.

4.2.3 SKEN#

Since SKEN# is used during read miss cycles and ignored otherwise, the system cache enable signal can be connected to all 82485 SKEN# inputs. If multiple sources can create the KEN# signal to the processor, the processors KEN# input may be fed into all 82485s. If the processor caches different memory locations that the 82485, SKEN# must be generated separately and then connected to all 82485 SKEN# inputs.

4.2.4 CS#

Chip Select is used to identify which 82485 is being addressed. It is the result of decoding the lowest order tag address bits. Figure 4.8 shows how a PLD chooses one of four 82845 controllers. Anytime an address is present on the address bus, including invalidation cycles, one of the 82485 controllers is selected. For example, in a two controller scheme, the table in section 3.1.8 shows that A15 should be used to decode between the two controllers. The chip selectt logic required to do this consists of an inverter so that A15 selects one controller and the compliment of A15 selects the second.

Chip select must be valid in T1 (with ADS#) and during invalidation cycles (with EADS#). See Figure 7.3 for a diagram depicting when CS# needs to be active.

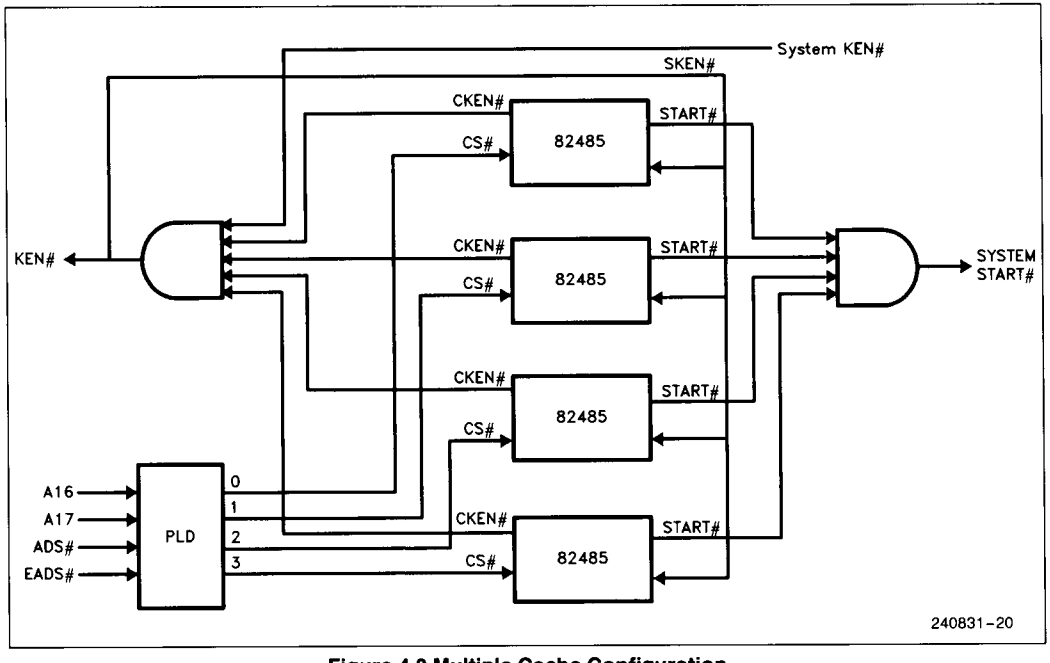


Figure 4.8 Multiple Cache Configuration

4.2.5 DESIGN CONSIDERATIONS

When adding more than one 82485 to a system, care should be taken to ensure that the address mappings of each controller do not overlap. The 82485 does not have provisions to communicate with other controllers. See section 4.1.7 for an additional design consideration that is applicable to both single and multiple 82485 designs.

4.3 82485 to Data Cache RAM (SRAM) Interface

A synchronous cache data Static RAM configured as a dual 4K X 18 SRAM with self timed writes and an address latch is needed to work with the 82485 cache controller. Each SRAM must have 2 internal banks: Bank A and Bank B to support the two-way set-associative configuration of the 82485. The SRAMs are used in pairs to extend the data width from 16-bits to 32-bits (36-bits with parity) (see Figure 4.9). The MT562818 SRAMs from Micron Technologies meet all of the SRAM requirements of the 82485 cache controller.

Four SRAMs are used in a 64K cache configuration. One SRAM pair comprises the Top Bank, the other SRAM pair comprises the Bottom Bank. Eight SRAMs are used in the 128K configuration (see Figure 4.10). Both a top and bottom bank of SRAMs are needed to support the burst mode of the i486 microprocessor and the 82485 (see section 4.3.1).

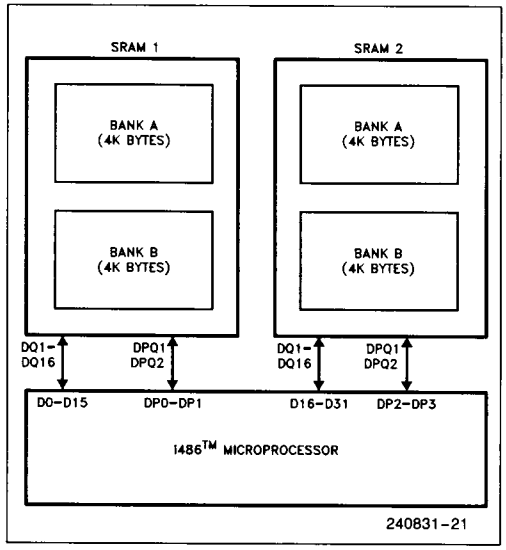
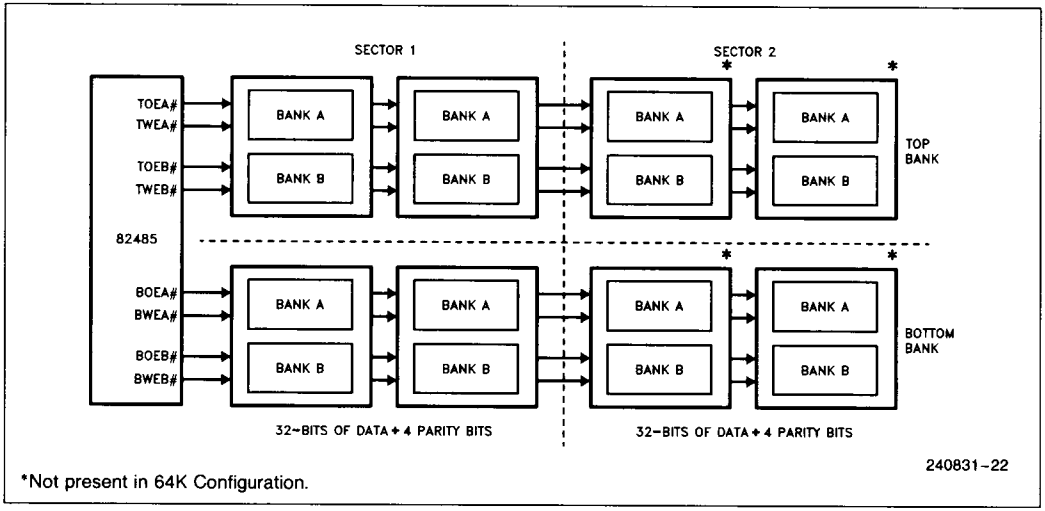


Figure 4.9 One (Top or Bottom) SRAM Bank Interface to the i486 CPU



*Not present in 64K Configuration.

Figure 4.10 64K and 128K SRAM Configuration

This section will explain in detail the SRAM configuration and the interface from the 82485 to the SRAMs.

4.3.1 BURST ORDER

The 82485 uses the i486 microprocessor burst order to fill the cache and to transfer data from the cache to the processor. The burst order is as follows:

First address	Second address	Third address	Fourth address
0	4	8	C
4	0	C	8
8	C	0	4
C	8	4	0

An important thing to note is that if the first address of the first dword is a 0 or 4, the address of the next dword will be a 4 or 0. Similary if the address for the first dword is an 8 or C, the address of the next dword will be a C or 8. This allows the 82485 to organize the data locations in the external data RAM so the first two accesses are from the same SRAM.

Each of the data RAMs has its own internal address latch. In order to make the first dword access as soon as possible, the first dword address is latched and input to all SRAMs (both banks A and B in both the top and Bottom banks) at once.

Figure 4.11 illustrates the external memory organization of the SRAMs. Recall that in a 2-way, set- associative architecture, once the set is determined, the access can be a hit in one of two WAYs: the left way or the right way. Note that in the memory organization of the SRAMs, the first two accesses will always be in the same SRAM chip. If the first address is a 0 or 4 and it is a hit in the left way, the first two accesses will be in the top bank. If the first address is a 0 or 4 and it is a hit in the right way, the first two accesses will be in the bottom bank. Likewise, the first two accesses will be a hit in the same SRAM if the first address is an 8 or C. This allows the first two and the last two accesses to come from the same SRAM. Also, because the data is latched into both the top and bottom banks at once, no time is lost in the data access after a hit or miss is determined. While the first two data accesses are being made from one SRAM, BAOT and BAOB are internally controlled and the internal RAM address latch of the other SRAM is restrobed. Figure 4.12 shows an example of a read hit cycle at address C and a hit in the right way.

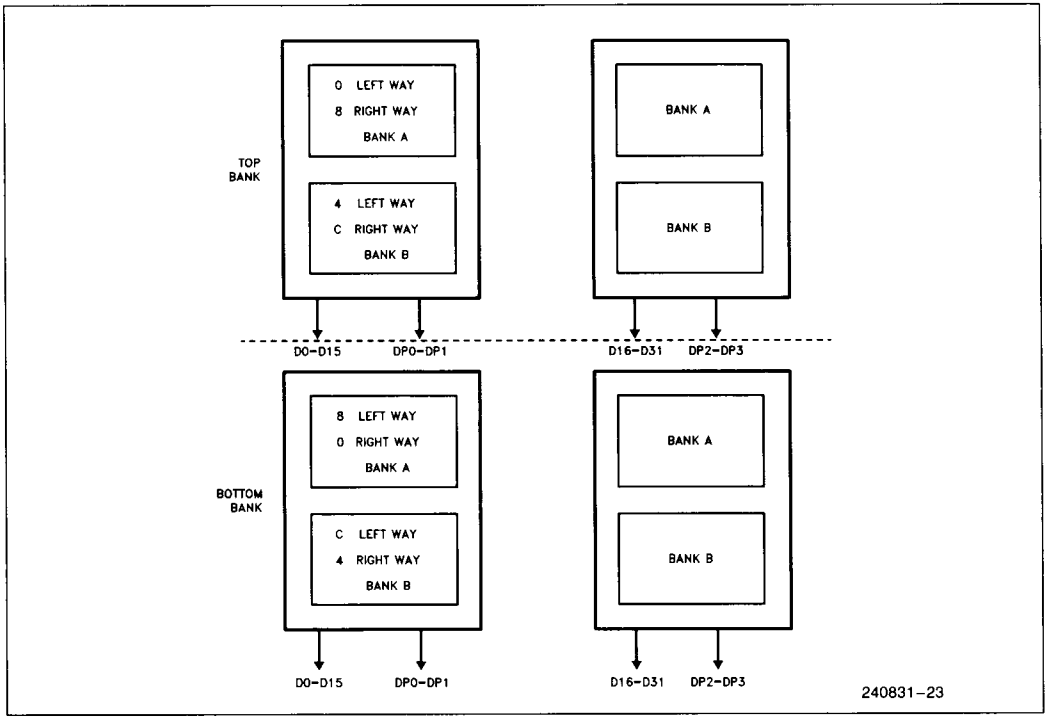


Figure 4.11 External Memory Organization of SRAMs

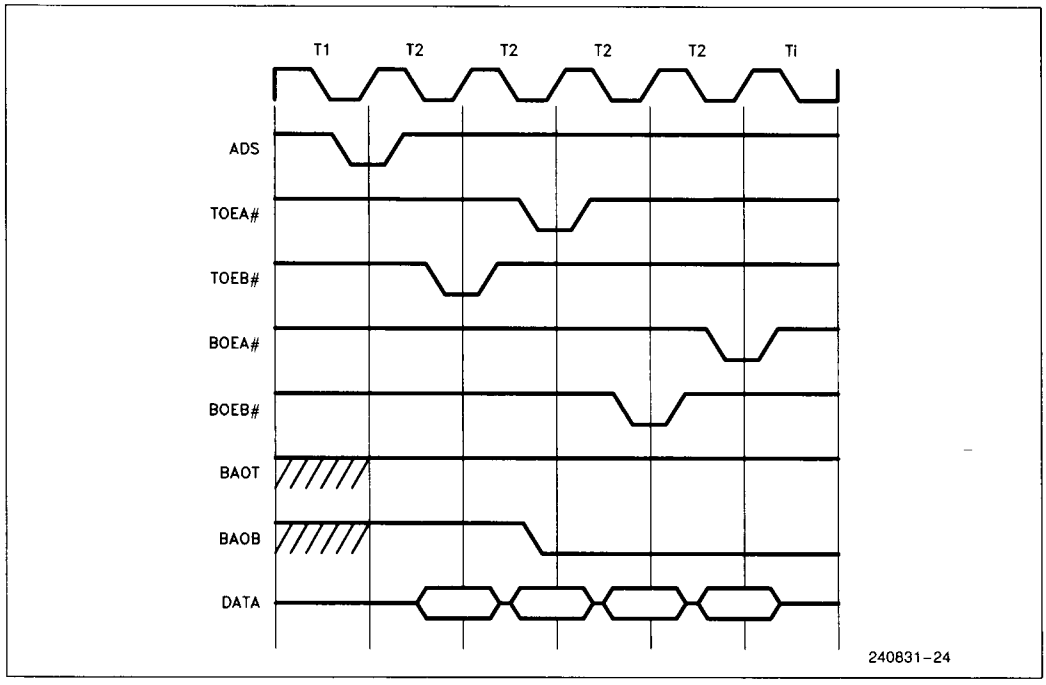


Figure 4.12 Read Hit in Right Way At Address 0CH

4.3.2 PIN INTERFACE

Much of the 82485 to SRAM interface is direct pin to pin connections. The following output signals are direct connections:TOEA#, TOEB#, BOEA#, BOEB#, TWEA#, TWEB#, BWEA#, BWEB#, BAOT, BAOB, and SAO0–SA010. See Figure 4-13 for an example using the Micron MT562818 SRAM.

4.3.2.1 Sector Select

The sector selects are Chip enables (CE#) to the SRAM. They provide the extra decoding necessary to determine which line in the tag is accessed in the sectored 128K configuration. In the 128K mode, SS1# is connected to CE# of all SRAMs in Sector 1 (see Figure 4.10). Sector select SS2# is connected to the CE# of all SRAMs in sector 2.

In the 64K mode, SS1# is connected to the CE# of all SRAMs and SS2# is not connected.

4.3.2.2 Cache Byte Enable (CBE0#–CBE3#)

Byte enable out pins are connected to the Chip Select (CS0#, CS1#) inputs of the SRAM. The SRAM chip selects choose which byte of data is enabled onto the data bus. When CS0# is low, DQ1– DQ8 and DQP1 are enabled. When CS1# is low, DQ9–DQ16 and DQP2 are enabled. Recall that the SRAM chips are used in pairs to obtain the full 32-bit data bus (see Figure 4.9). Since SRAM 1 outputs data bits D0–D15, the CSO# and CS1# pins are connected to CBE0# and CBE1# of the 82485. The CS0# and CS1# pins in SRAM 2 are connected to CBE2# and CBE3# of the 82485.

4.3.2.3 Other Connections

As shown in Figure 4.9 the data and parity outputs of the SRAM are directly connected to the i486 microprocessor data bus.

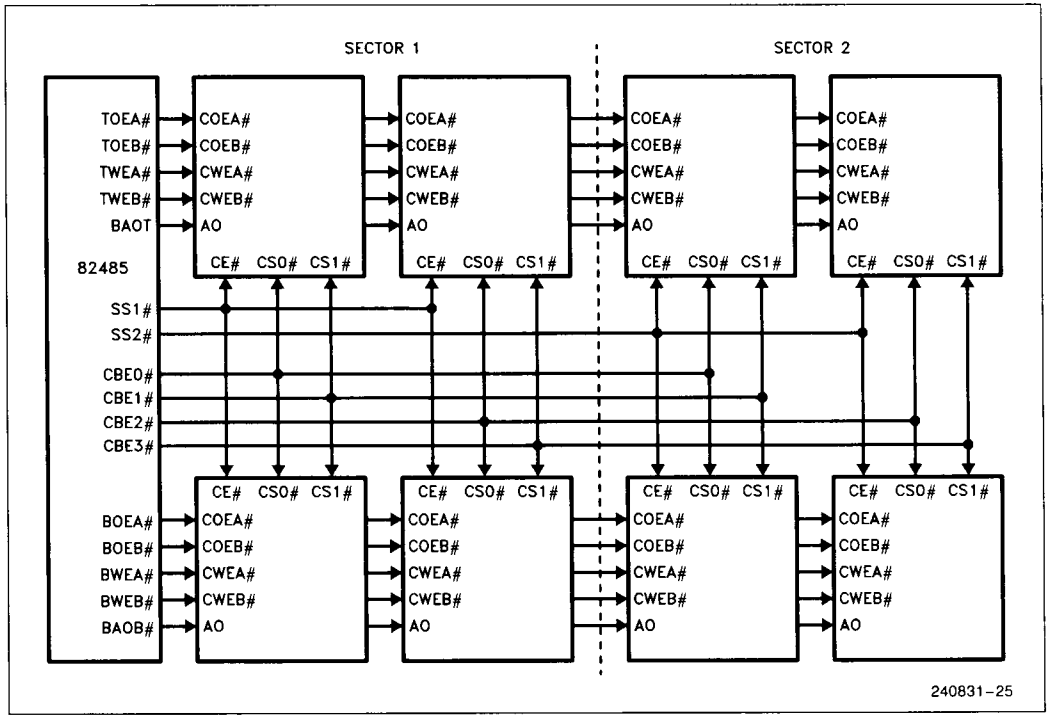


Figure 4.13 82485 to Micron SRAM Interface

4.3.3 SRAM INTERFACE TIMINGS

This section will provide timing diagrams for the output signal of the 82485 that connect to the SRAM. The signal include TOEA#, TOEB#, BOEA#, BOEB#, TWEA#, TWEB#, BWEA#, BWEB#, BAOT, BAOB, SAO0–10, SS1#, SS2#, and CBE0–3#.

Figure 4.14 shows a read hit cycle. ADS# is issued from the processor. This indicates that there is a valid address on the bus and and external cycle is ready to begin. With a valid address, the following 82485 inputs are valid: TAI0–16, SAI–10, LS, BAI0–1 and BE0–3#. The address is latched internally by the 82485. By the end of T1 or the beginning of the first T2 (max valid delay t32, t33), the following 82485 outputs are valid: SAO0–10, CBE0- -3#, BAOT, BAOB, and the sector selects SS1#, and SS2#. At this time BAOT and BAOB are at the same logic level as BAI1 (a direct feed through of A3 from the i486 microprocessor).

In each of the four T2 clocks in a read hit cycle, data is enabled onto the bus by one of the four output enables TOEA#, TOEB#, BOEA#, BOEB#. The output enables meet the valid delay time t34 in section 7.2.1. The first two accesses will come from a single SRAM bank (top or bottom), the next two accesses will come from the other SRAM bank. Whether the OE from bank A or bank B of the SRAM is enabled depends on the WAY in which there is a hit (see section 4.3.1). During the second transfer, the burst address is toggled by either BAOT or BAOB changing state. If there was a hit in the top bank, BAOB will toggle, if there was a hit in the bottom bank, BAOT will toggle. BAOT or BAOB will toggle during the second T2 with a valid delay corresponding to t32a.

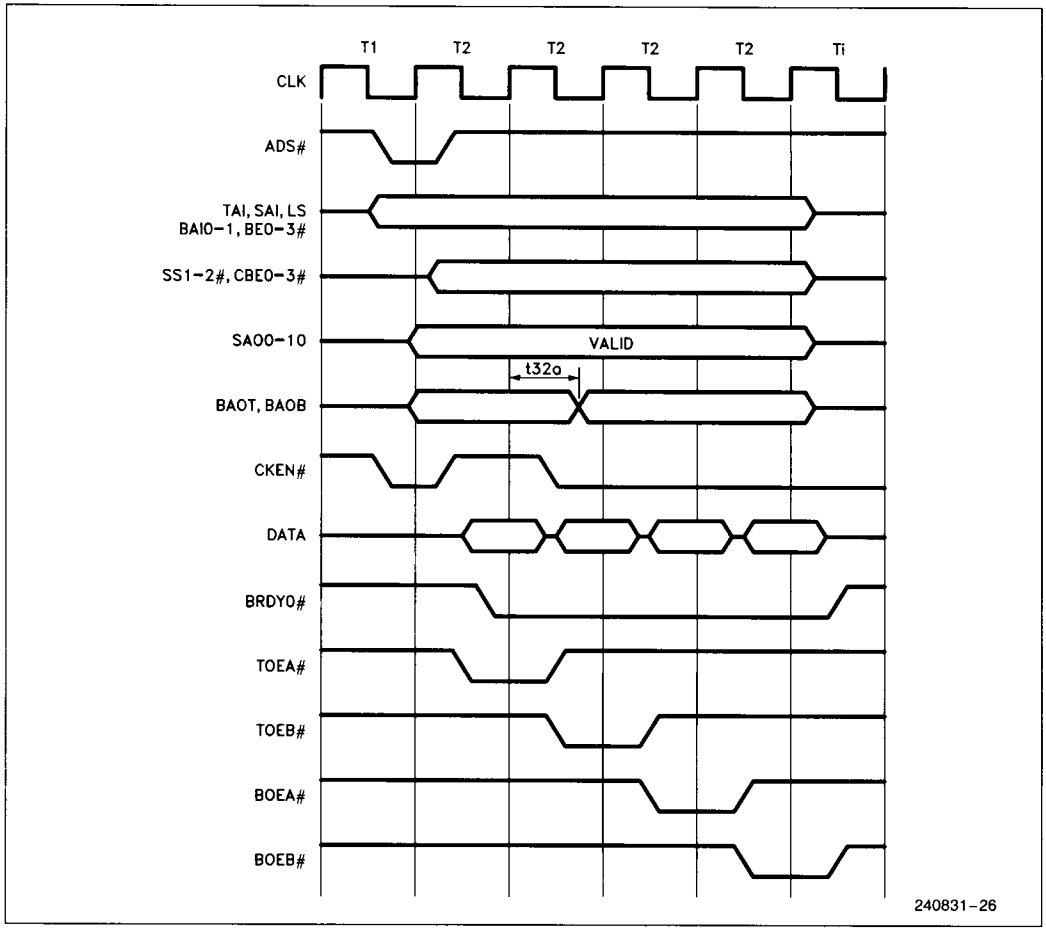


Figure 4.14 Read Hit Cycle

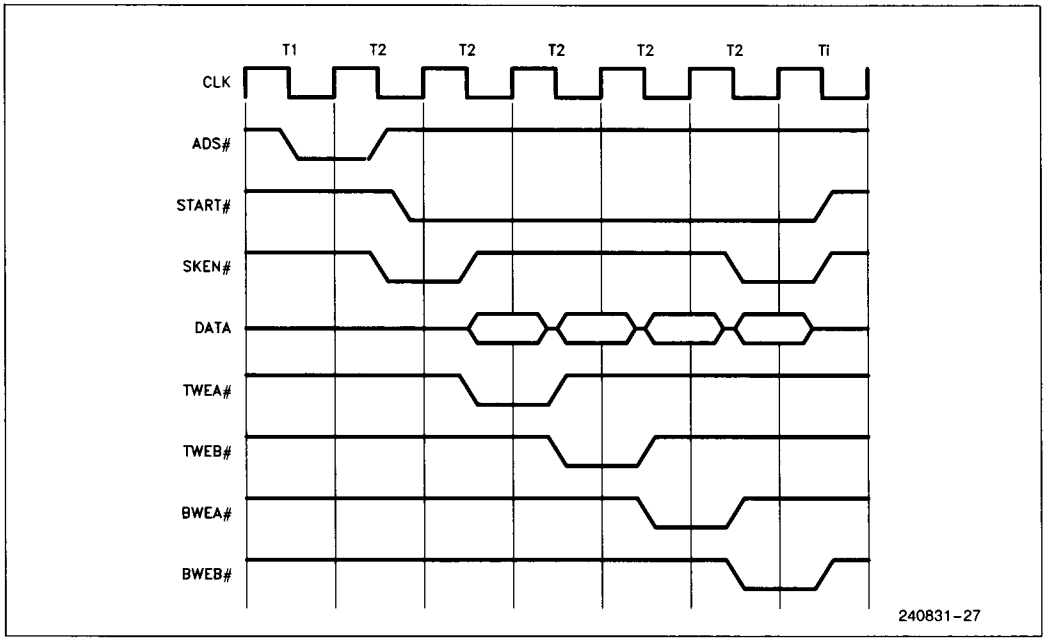


Figure 4.15 Cache Line Fill

Figure 4.15 shows a cache line fill resulting from a read miss. ADS# is issued by the processor. The 82485 performs a tag lookup to determine a hit or a miss. If the access is a cache miss, START# will be asserted by the 82485. If SKEN# is driven active to the 82485 the clock before the first CRDY# or CBRDY#, a cache line fill will take place. The line will be validated only if SKEN# is driven active again the clock before the last CRDY# or CBRDY#. As the data is returned, the appropriate write enable (TWEA#, TWEB#, BWEA#, BWEB#) is driven to allow the data to be written to the 82485. The appropriate write enables are determined by the starting address and the WAY in which there is a hit. The write enables meet the valid delay times t35 in section 7.2.1.

Figure 4-16 shows a direct processor write. As in the previous cycles, ADS# is issued by the processor and the SAO0-10, BAOT, BOAB, SS1-2#, and CBE0-3# outputs are valid by the end of T1 or the beginning of the first T2 (t32, t33). If the address is a cache hit, the cache be will updated with the data from the processor. The data is written into the SRAM when the appropriate write enable from the 82485 is activated as shown. If the write address is a cache miss, the 82485 does not pulse the write enables and the data is not written into the cache.

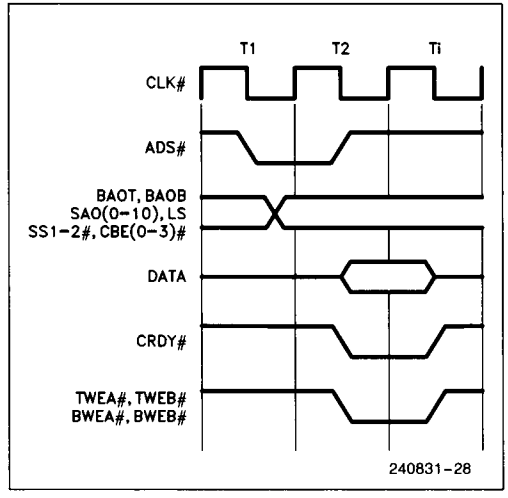


Figure 4.16 Write Cycle

4.3.3.1 82485/SRAM Timing Equations

The following equations will help determine the critical SRAM timings necessary for an SRAM to work with the 82485 cache controller. The Micron MT56C2818 SRAM meets the necessarry specifications.

READ CYCLE TIMINGS

SRAM Address Access Time:

2CLK - SAO(0–10) valid(t32) - Address Access Time - i486 CPU data setup ≥ 0

@33 MHz: 60 - 30 - Address Access Time - 5 ≥ 0

@33 MHz: Address Access Time ≤ 25 ns

@25 MHz: 80 - 40 - Address Access Time - 5 ≥ 0

@25 MHz: Address Access Time ≤ 35 ns

CHIP SELECT/CHIP ENABLE ACCESS TIME

2 CLK-CBE0-3#/SS1-2 valid (t33)-Access Time-i486 CPU data setup ≥ 0

@33 MHz: 60 - 31 - Access Time - 5 ≥ 0

@33 MHz: Chip Select Access Time ≤ 24 ns

@33 MHz: Chip Enable Access Time ≤ 24 ns

@25 MHz: 80 - 40 - Access Time - 5 ≥ 0

@25 MHz: Chip Select Access Time ≤ 35 ns

@25 MHz: Chip Enable Access Time ≤ 35 ns

Output Enable Access Time:

1CLK - OE Valid Delay(t34) - Output Enable Access - i486 CPU Data Setup ≤ 0

@33 MHz: 30 - 17 - Output Enable Access - 5 ≤ 0

@33 MHz: Output Enable Access Time ≤ 8 ns

@25 MHz: 40 - 25 - Output Enable Access - 5 ≤ 0

@25 MHz: Output Enable Access Time ≤ 10 ns

WRITE CYCLE TIMINGS

SRAM compatible with the 82485 must be able to perform self time writes in one clock.

Write Enable Setup Time (to write strobe):

1CLK - WE Valid Delay(t34) - Write Enable Setup ≥ 0

@33 MHz: 30 - 19 - Write Enable Setup ≥ 0

@33 MHz: Write Enable Setup Time ≤ 11 ns

@25 MHz: 40 - 28 - Write Enable Setup ≥ 0

@25 MHz: Write Enable Setup Time ≤ 12 ns

Write Enable Hold Time (after write strobe) must be less than or equal to 3 ns for both 25MHz and 33Hz systems.

Address Setup Time (to write strobe):

2CLK - SAO(0-10) Valid Delay (t32) - Address Setup time ≥ 0

@33 MHz: 60 - 30 - Address Setup Time ≥ 0

@33 MHz: Address Setup Time ≤ 30 ns

@25 MHz: 80 - 40 - Address Setup Time ≥ 0

@25 MHz: Address Setup Time ≤ 40 ns

CHIP SELECT/CHIP ENABLE SETUP TIME

2 CLK-SS1-2#/CBE0-3# valid (t33)-Setup Time ≥ 0

@33 MHz: 60 - 31 - Setup Time ≥ 0

@33 MHz: Chip Setup Time ≤ 29 ns

@33 MHz: Chip Enable Setup Time ≤ 29 ns

@25 MHz: 80 - 40 - Setup Time ≥ 0

@25 MHz: Chip Select Setup Time ≤ 40 ns

@25 MHz: Chip Enable Setup Time ≤ 40 ns

Data Setup Time (to write strobe):

1CLK - i486 CPU Data Valid Delay(t10) - Data Setup Time ≥ 0

@33 MHz: 30 - 18 - Data Setup Time ≥ 0

@33 MHz: Data Setup Time ≤ 12 ns

@25 MHz: 40 - 22 - Data Setup Time ≥ 0

@25 MHz: Data Setup Time ≤ 18 ns

Data Hold Time (after write strobe) must be less than 3ns for both 25MHz and 33MHz systems.

5.0 PERFORMANCE CONSIDERATIONS

The following section offers a few special consideration that will increase cache performance or ease hardware design. These considerations are simply design notes and are not deviations from the 82485 or i486 microprocessor specifications.

5.1 SKEN# Assertion

SKEN# is an input to the 82485 to indicate the cacheability of a line in the 82485 during a read miss cycle. It is sampled exactly like KEN# to the processor, one clock before the first dword transfer of a line fill and one clock before the last dword.

During a line fill the 82485 loads the dwords of the line directly into the appropriate slot in cache memory. This means that once SKEN# has been sampled active by the 82485 it must commit a line and invalidate a location to prepare for the incoming line. Once a line fill completes with a proper SKEN#, the line can be validated.

A potential performance loss exists if a system designer chooses during noncacheable cycles to keep SKEN# active, but inactivate SKEN# the clock before the last transfer (see Figure 5.1). Once the 82485 sees SKEN# low in T1 it commits a line in the cache by invalidating an entry despite the fact that SKEN# was later deasserted. The performance loss can be avoided if SKEN# was held inactive until cacheability could be determined.

5.2 Invalidation Window

When an invalidation is requested with the assertion of EADS#, the 82485 must immediately invalidate the address present on the address bus. If the tag portion of the 82485 is in use, the invalidation takes priority and will suspend the other action. This may decrease performance. To avoid this, EADS# should not be issued in the second, third or fourth transfer of a cache read miss cycle. Section 2.3.5 explains this in detail.

5.3 BOFF# Assertion

If BOFF# is asserted and the 82485 is in the middle of a cacheable read miss cycle, the 82485 treats the current line as non-cacheable. Once BOFF# is release and the cycle continues, the 82485 will treat the rest of the cycle as a non-cacheable cycle.

In most systems BOFF# is a rare occurance, so performance loss is negligible. If, however, BOFF# is regular and predictable, system performance can be increased by timing BOFF# so that the four dword transfers of a line fill are never interrupted. Section 2.6 explains aborted cycles in more detail.

5.4 START# Predictability

START# is asserted in the first T2 of a read miss cycle unless an invalidation occured in the previous cycle. The section titled Invalidation Cycle (2.3.5) explains why START# may be delayed. If START# must be a predictable signal to the system, and invalidation cycles can not be timed to occur before the second transfer of a read miss cycle, there is a way to ensure the predictability of START# if self invalidations ar not allowed in the last transfer of a line fill.

When EADS# is asserted towards the end of a read miss cycle, there are 3 tag accesses that need to be made before START# can be issued: invalidate lookup, the actual invalidation (if a hit), and the validation of the current line fill (if cacheable). Since there is no way to predict the hit/miss possibility of an invalidation request, it is assumed that 2 tag accesses will be required to service it. One tag access can be saved however by making the current line fill non-cacheable.

To do this, SKEN# to the 82485 may be deasserted if AHOLD is detected. If SKEN# is deasserted the clock before the last CBRDY#, the line is non-cacheable. Figure 5.2a shows how the assertion of EADS# in the third transfer of a line fill icurs a 1 clock delay in START#. Figure 5.2b shows EADS# assertion in the fourth clock, but since AHOLD will cause the processor to delay ADS# at least one extra clock (self invalidations are not allowed in the last transfer), START# is delayed only one clock as well. Assertion of EADS# in the second transfer of a burst causes a 1 clock delay in START# without deasserting SKEN# (see Figure 5.2b), so there is no advantage to dropping SKEN# for EADS# assertion then.

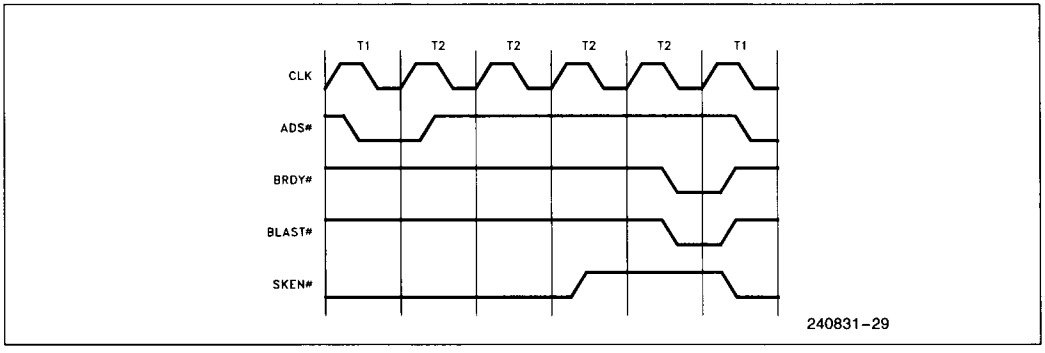
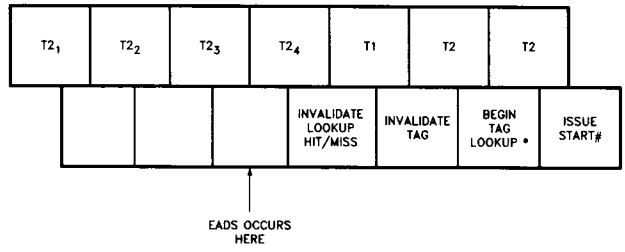
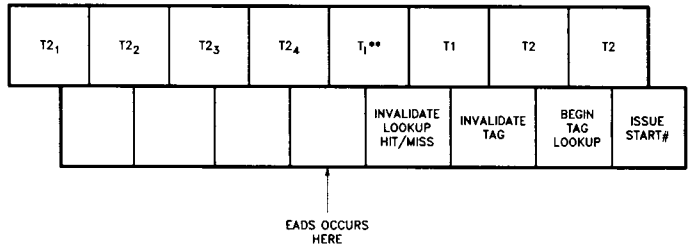


Figure 5.1 Method of SKEN# Generation NOT Recommended



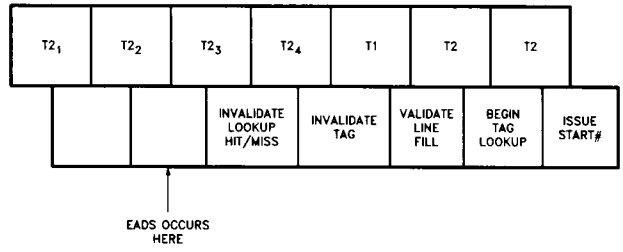
240831-30

A



240831-31

B



240831-32

C

NOTE:
* Tag validation is not done since SKEN# was deasserted.
**T1 occurs because AHOLD assertion and deassertion causes ADS# to be delayed.

Figure 5.2 Predictable START# Delay

In summary if SKEN# is deasserted in response to AHOLD during the third or fourth transfer of a line fill, START# will be delayed at most one clock. This makes START# predictable: it will always be valid in the second T2 of a read miss cycle. Note that if START# was not delayed, its value is retained in the second T2.

6.0 TESTING AND DATA INTEGRITY

Power up self test programs can test cache memory. The following algorithm will test any number of 82485 cache controllers and SRAM up to 512K of cache memory.

1) Flush or Reset the cache.
2) Write "1" to every bit of a 512K block of memory.
3) Read the 512K block. This fills the cache.
4) Disable CS# and write "0" to the 512K block. This fills memory.
5) Re-enable CS#
6) Read the 512K block back.
   - Repetitive assertions of START# indicate the cache boundry (size of the cache).
   - Data not equal to "1" indicates bad 82485 or SRAM.
7) Repeat with "0" in the cache and "1" in memory.

7.0 ELECTRICAL DATA

NOTICE: This data sheet contains information on products in the sampling and initial production phases of development. The specifications are subject to change without notice.

7.1 Maximum Ratings

Case Temperature Under Bias . . . −65°C to +110°C

Storage Temperature . . . . . −65°C to +150°C

Voltage on any pin with respect to ground −0.5V to V_{CC} +0.5V

Power Dissipation 1.32W

Supply Voltage with respect to V_{SS} . . −0.5V to 6.5V

**WARNING: Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only. Operation beyond the "Operating Conditions" is not recommended and extended exposure beyond the "Operating Conditions" may affect device reliability.*

7.2 D.C. Specifications (Vcc = 5V ±5%, Tcase = 0°C to 85°C)

Symbol	Parameters	Min	Max	Unit	Notes
Vil	Input Low Voltage	−0.3	+0.8	V	Not Including Clock
Vih	Input High Voltage	2.0	Vcc+0.3	V	Not Including Clock
Vol	Output Low Voltage		+0.45	V	Note 3
Voh	Output High Voltage	2.4		V	Note 4
Vcil	CLK Input Low Voltage	−0.3	+0.8	V	
Vcih	CLK Input High Voltage	2.0	Vcc+0.3	V	
Icc	Supply Current		250	mA	
Ili	Input Leakage Current		± 15	uA	Note 1
Iil	Input Leakage Current		−400	uA	Note 2
Cin	Input Capacitance		15	pF	

ADVANCE INFORMATION SEE INTEL FOR DESIGN-IN INFORMATION

Notes:

1) This parameter is for inputs without pullups or pulldowns (all inputs except TAI15, TAI16 and WPSTRP#) and 0≤ Vin ≤Vcc

2) This parameter is for inputs with pullups (TAI15, TAI16 and WPSTRP#) and Vil = 0.45

3) Measured at 4.0 mA for SAO0-10, SS1-2#, TWEA/B#, BWEA/B#, TOEA/B#, BOEA/B#, BAOT/B, CBE0-3#. Measured at 5.0 mA for START#, BRDYO#, CKEN#.

4) Measured at 0.9 mA for SAO0-10, SS1-2#, TWEA/B#, BWEA/B#, TOEA/B#, BOEA/B#, BAOT/B, CBE0-3#. Measured at −1.0 mA for START# BRDYO#, CKEN#.

7.3 A.C Specifications (Vcc = 5V ± 5%, Tcase = 0°C to 85°C,Ci = 50pF)

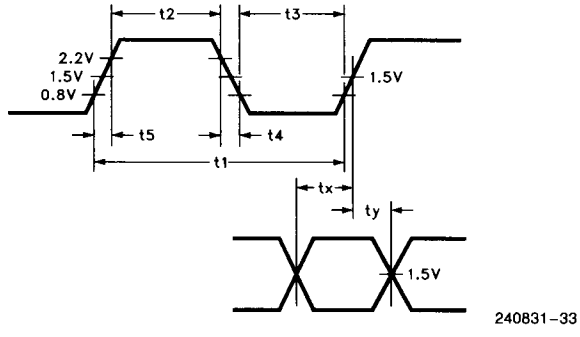
Symbol	Parameter	25 MHz		33 MHz		Units	Figure	Notes
		Min	Max	Min	Max			
	Frequency	24	25	24	33	MHz	7.1	
t1	CLK period	40	42	30	42	ns	7.1	
t1a	CLK period stability		0.1%		0.1%	delta		adjacent clocks
t2	CLK high time	14		11		ns	7.1	@ 2V
t3	CLK low time	14		11		ns	7.1	@ 0.8V
t4	CLK fall time		4		3	ns	7.1	
t5	CLK rise time		4		3	ns	7.1	
t6	TAI0-16,SAI0-10,BAI0-1 BE0-3#, LS setup time	17		13		ns	7.2	
t7	TAI0-16,SAI0-10,BAI0-1 BE0-3#, LS hold time	3		3		ns	7.2	
t8	ADS#, M/IO#, W/R# setup time	17		13		ns	7.2	
t9	ADS#, M/IO#, W/R# hold time	3		3		ns	7.2	
t10	BLAST# setup time	10		8		ns	7.2	
t11	BLAST# hold time	3		3		ns	7.2	
t12	CRDY#, CBRDY# setup time	8		5		ns	7.2,7.4	
t13	CRDY#, CBRDY# hold time	3		3		ns	7.2,7.4	
t14	SKEN# setup	8		5		ns	7.2	
t15	SKEN# hold	3		3		ns	7.2	
t16	WP setup	8		8		ns	7.2	
t17	WP hold	3		3		ns	7.2	
t18	BOFF# setup	10		8		ns	2.9	
t19	BOFF# hold	3		3		ns	2.9	
t20	EADS# setup	8		5		ns	7.3	
t21	EADS# hold	3		3		ns	7.3	
t22	TAI0-16, SAI0-10,BAI0-1, BE0-3#, LS setup for invalidations	5		5		ns	7.3	
t23	TAI0-16, SAI0-10,BAI0-1, BE0-3#, LS hold for invalidations	3		3		ns	7.3	
t24	RESET,FLUSH# setup	10		5		ns	7.5	
t25	RESET, FLUSH# hold	3		3		ns	7.5	

ADVANCE INFORMATION
SEE INTEL FOR DESIGN-IN INFORMATION

7.3 A.C Specifications (Vcc = 5V ± 5%, Tcase = 0°C to 85°C,Ci = 50pF) (Continued)

Symbol	Parameter	25 MHz		33 MHz		Units	Figure	Notes
		Min	Max	Min	Max			
t26	FLUSH# pulse width	60		45		ns	7.5	For asynchronous use
t27	BRDYO# valid delay	5	22	5	16	ns	7.3	
t28	CKEN# valid delay	3	18	3	15	ns	7.3	
t29	START# valid delay	5	22	5	16	ns	7.2	
t30	CS# setup	6		6		ns	7.2,7.3	
t31	CS# hold	3		3		ns	7.2,7.3	
t32	BAOT,BAOB, SAO1-10, valid delay	3	40	3	30	ns	7.2	Measured from the T1 clock Note 1
t32a	BAOT,BAOB valid delay	3	40	3	30	ns	7.2,4.14	Measured from T2 clock see fig
t33	SS1-2#, CBE0-3# valid delay	3	40	3	31	ns	7.2	Measured from T1
t34	TOEA-B#, BOEA-B# valid delay	5	25	5	17	ns	7.3	
t35	TWEA-B#, BWEA-B# valid delay	3	28	3	19	ns	7.2	

NOTE:
1. SAO timings assume a 100 pF load.



tx = input setup times
ty = input hold times, output float, valid and hold times

Figure 7.1 CLK Waveforms

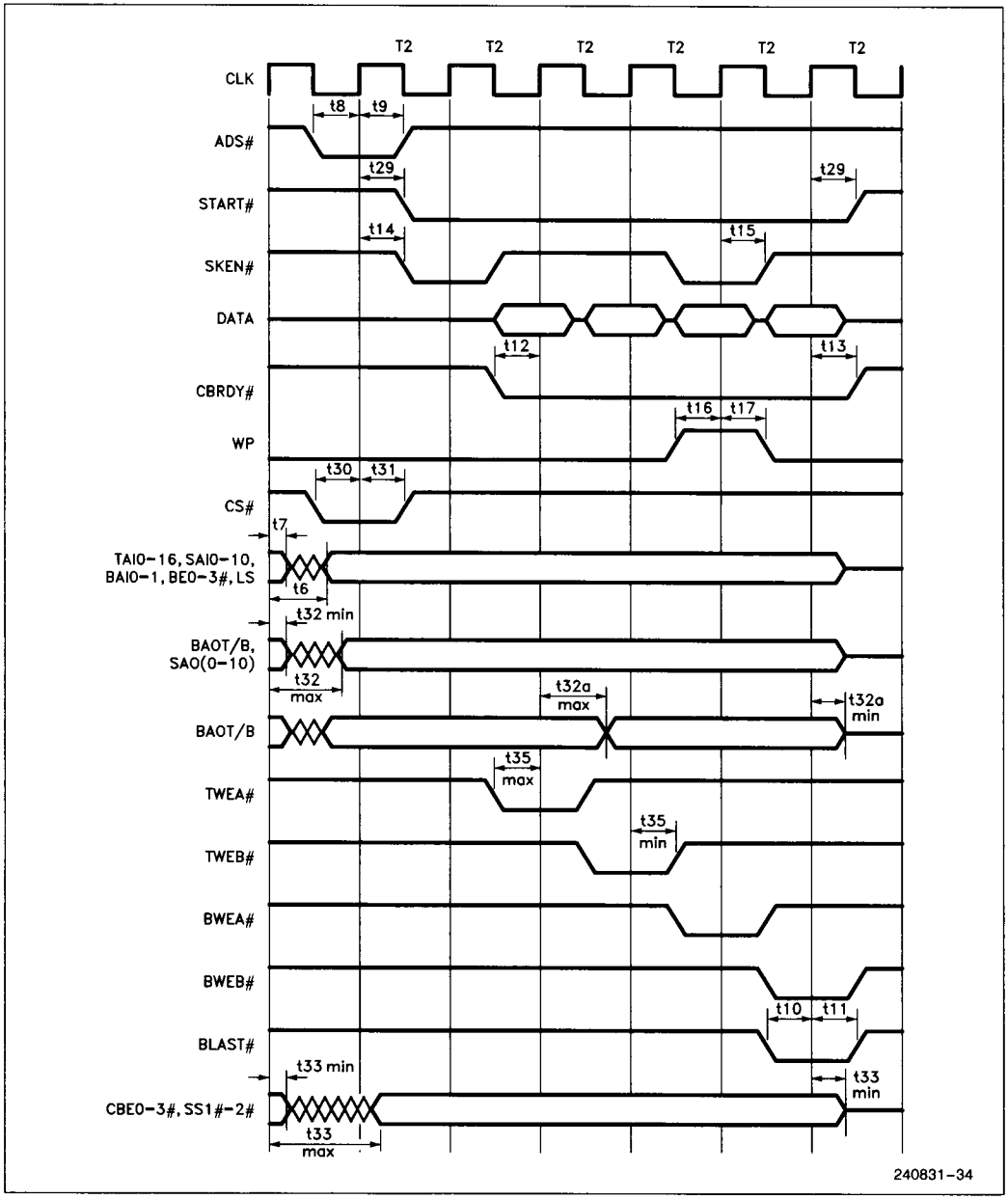


Figure 7.2 Read Miss with Write Protection

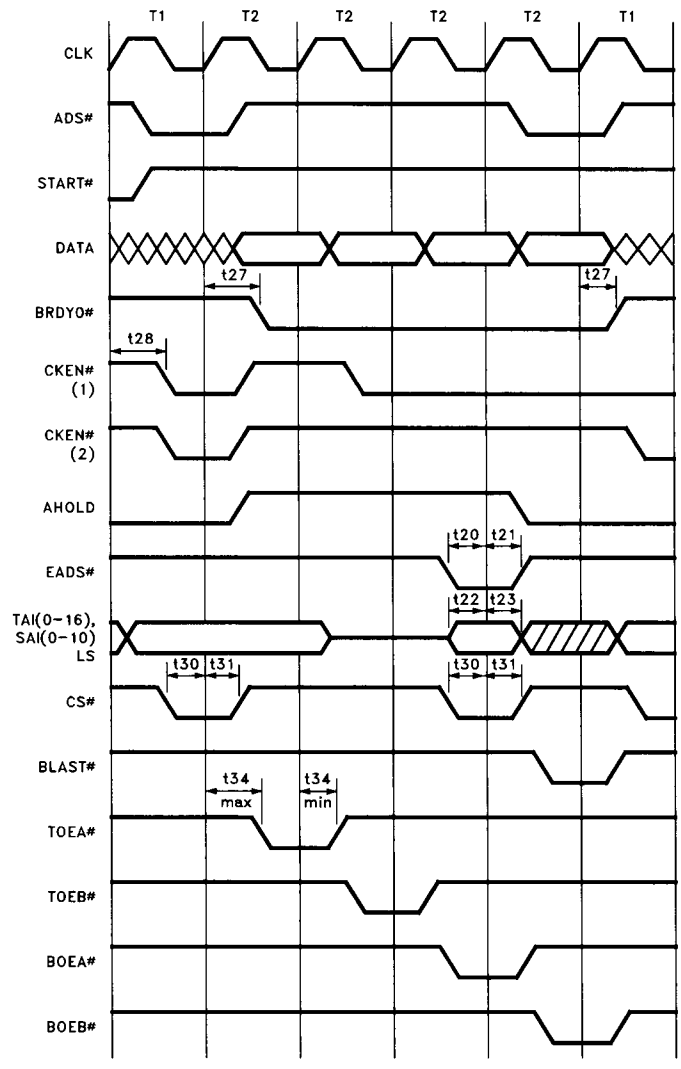


Figure 7.3 Read Hit Cycle with Invalidation

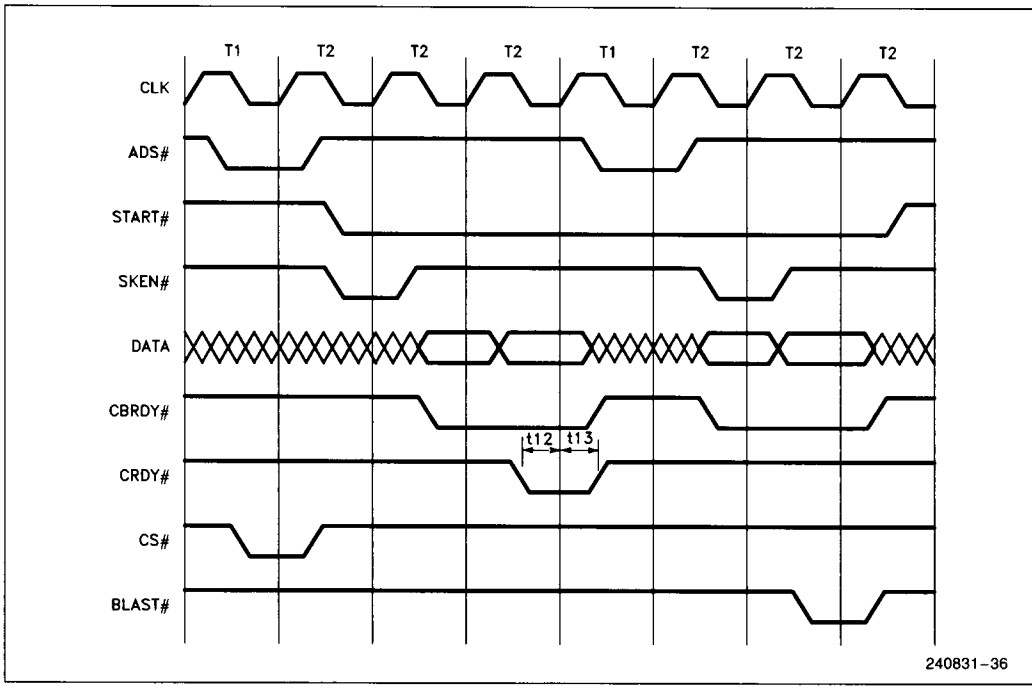


Figure 7.4 Multiple Cycle Line Fill

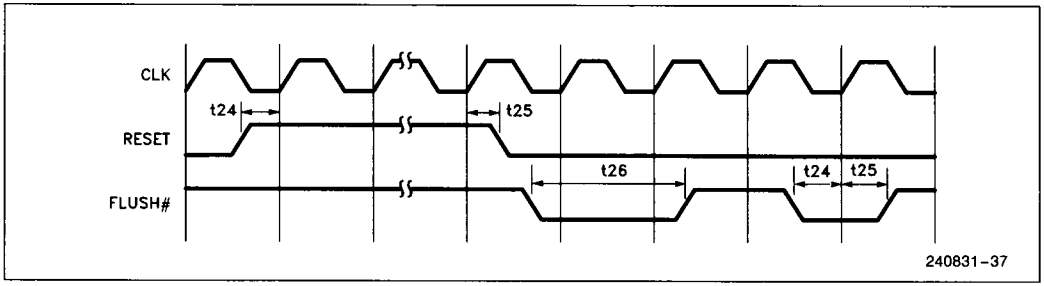


Figure 7.5 RESET and FLUSH#

7.4 Typical Output Valid Delay vs Load Capacitance under Worst Case Conditions

Figure 7.6 depicts the typical output valid delay vs load capacitance under worst case conditions for the address outputs (BAOT/B, CBE0–3#, SS1–2#, and SAO0–10).

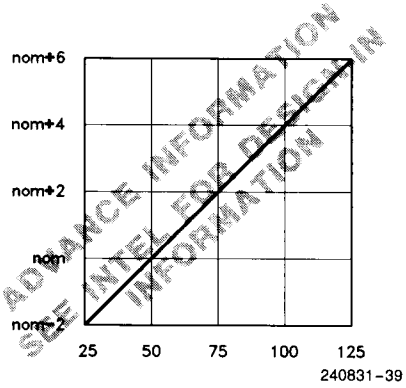


Figure 7.6

Figure 7.7 depicts the typical output valid delay vs load capacitance under worst case conditions for the control signals going active (low). The control outputs include TOEA/B#, BOEA/B#, TWEA/B#, BWEA/B#, BRDYO# and CKEN#.

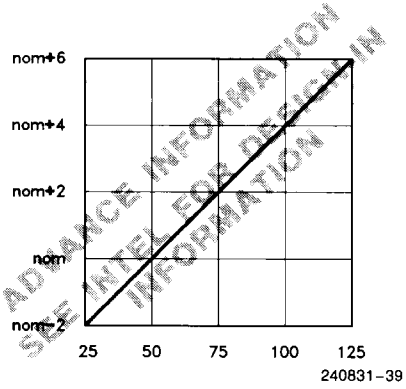


Figure 7.7

Figure 7.8 depicts the typical output valid delay vs load capacitance under worst case conditions for the control signals going inactive (high).

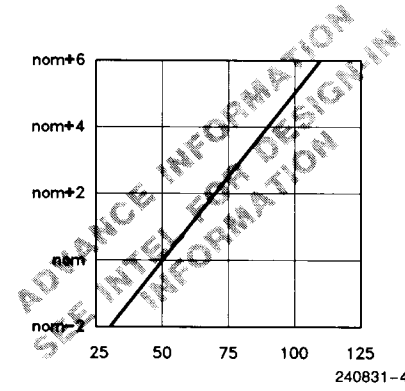


Figure 7.8

NOTE FOR ALL GRAPHS:
1. The graphs will not be linear outside of the capacitive load range shown.
2. nom = nominal value given in A.C. Characteristics Table.

8.0 MECHANICAL DATA

This section gives package dimensions (see Figure 8.1). Package thermal specifications are to be determined.

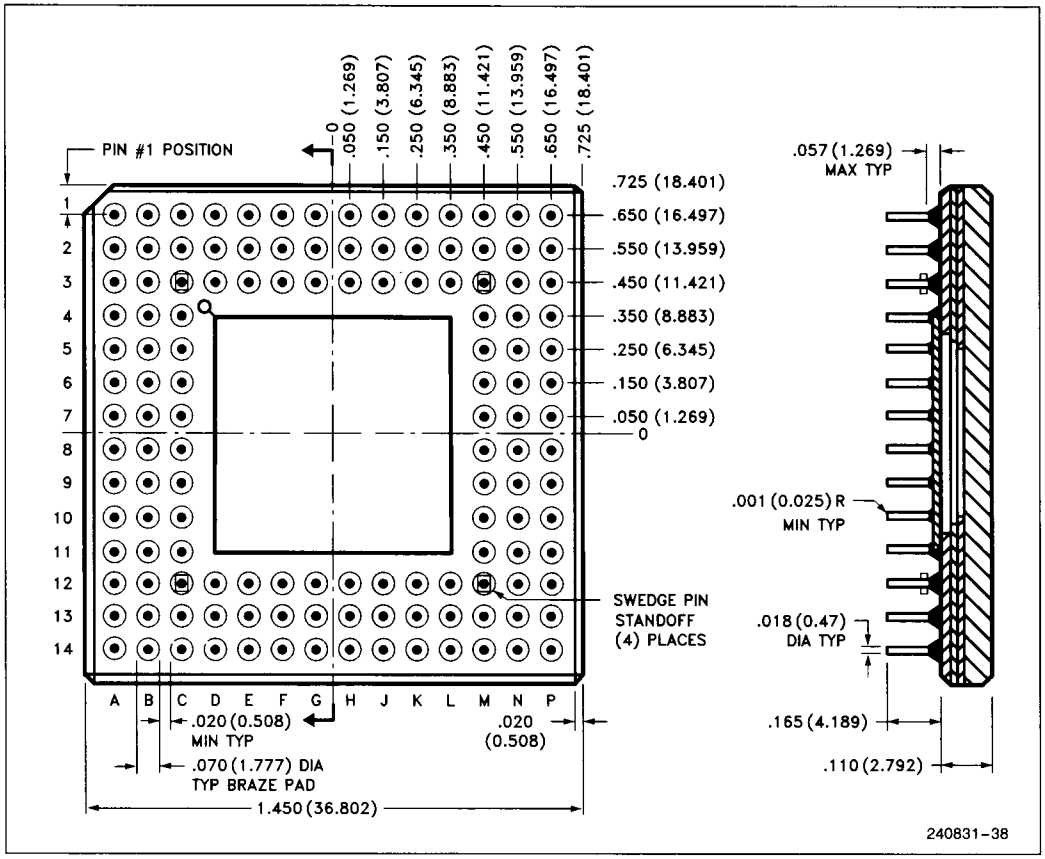


Figure 8.1 132-Pin Ceramic PGA Package Dimensions