Welcome to E-XFL.COM

Understanding Embedded - Microprocessors

Embedded microprocessors are specialized computing chips designed to perform specific tasks within an embedded system. Unlike general-purpose microprocessors found in personal computers, embedded microprocessors are tailored for dedicated functions within larger systems, offering optimized performance, efficiency, and reliability. These microprocessors are integral to the operation of countless electronic devices, providing the computational power necessary for controlling processes, handling data, and managing communications.

Applications of Embedded - Microprocessors

Embedded microprocessors are utilized across a broad spectrum of applications, making them indispensable in

Details	
Product Status	Active
Core Processor	68040
Number of Cores/Bus Width	1 Core, 32-Bit
Speed	25MHz
Co-Processors/DSP	-
RAM Controllers	-
Graphics Acceleration	No
Display & Interface Controllers	-
Ethernet	-
SATA	-
USB	-
Voltage - I/O	5.0V
Operating Temperature	0°C ~ 70°C (TA)
Security Features	-
Package / Case	182-BEPGA
Supplier Device Package	182-PGA (47.24x47.24)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mc68040rc25v

TABLE OF CONTENTS (Continued)

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

SECTION 3
MEMORY MANAGEMENT UNIT
(EXCEPT MC68EC040 AND MC68EC040V)

NOTE

This section does not apply to the MC68EC040 and MC68EC040V. Refer to **Appendix B MC68EC040** for details. All references to M68040 in this section only, refer to the MC68040, MC68040V, and MC68LC040.

The M68040 supports a demand-paged virtual memory environment. Demand means that programs request memory accesses through logical addresses, and paged means that memory is divided into blocks of equal size, called page frames. Each page frame is divided into pages of the same size. The operating system assigns pages to page frames as they are required to meet the needs of the program.

The M68040 memory management includes the following features:

- Independent Instruction and Data Memory Management Units (MMUs)
- 32-Bit Logical Address Translation to 32-Bit Physical Address
- User-Defined 2-Bit Physical Address Extension
- Addresses Translated in Parallel with Indexing into Data or Instruction Cache
- 64-Entry Four-Way Set-Associative Address Translation Cache (ATC) for Each MMU (128 Total Entries)
- Global Bit Allowing Flushes of All Nonglobal Entries from ATCs
- Selectable 4K or 8K Page Size
- Separate Supervisor and User Translation tables
- Two Independent Blocks for Each MMU Can Be Defined as Transparent (Untranslated)
- Three-Level Translation Tables with Optional Indirection
- Supervisor and Write Protections
- History Bits Automatically Maintained in Descriptors
- External Translation Disable Input Signal (`MDIS`) for Emulator Support
- Caching Mode Selected on Page Basis

The MMUs completely overlap address translation time with other processing activities when the translation is resident in one of the ATCs. ATC accesses operate in parallel with

For More Information On This Product,
Go to: www.freescale.com
Freescale Semiconductor, Inc.

instruction, and the stacked FA points to the first longword in the missing page. When an ATC access error occurs while prefetching the next instruction on the non-existant page after a change of flow instruction, the exception should be cleared by execution of the new instruction flow. Either avoid this scenario, or have a dummy resident page following the exceptional instruction.

Figure 3-22 illustrates a general flowchart for address translation. The top branch of the flowchart applies to transparent translation. The bottom three branches apply to ATC translation.

3.6 MMU EFFECT ON RSTI AND MDIS

The following paragraphs describe MMU effects on the RSTI and MDIS pins.

3.6.1 Effect of RSTI on the MMUs

When the M68040 is reset by the assertion of the reset input signal, the E-bits of the TCR and TTRs are cleared, disabling address translation. This reset causes logical addresses to be passed through as physical addresses, allowing an operating system to set up the translation tables and MMU registers as required. After the translation tables and registers are initialized, the E-bit of the TCR can be set, enabling paged address translation. While address translation is disabled, the attribute bits for an access that an ATC entry or a TTR normally supplies are zero, selecting write-through cachable mode, no write protection, and user page attribute bits cleared. RSTI does not affect the P-bit of the TCR.

A reset of the processor does not invalidate any entries in the ATCs or alter the page size. A PFLUSH instruction must be executed to flush all existing valid entries from the ATCs after a reset operation and before translation is enabled. PFLUSH can be executed even if the E-bit is cleared.

3.6.2 Effect of MDIS on Address Translation

The assertion of MDIS prevents the MMUs from performing ATC searches and the execution unit from performing table searches. With address translation disabled, logical addresses are used as physical addresses. MDIS disables the MMUs on the next internal access boundary when asserted and enables the MMUs on the next boundary after the signal is negated. The assertion of this signal does not affect the operation of the transparent translation registers or execution of the PFLUSH or PTEST instructions.

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

SECTION 4
INSTRUCTION AND DATA CACHES

NOTE

Ignore all references to the memory management unit (MMU) when reading for the MC68EC040 and MC68EC040V. The functionality of the MC68040 transparent translation registers has been changed in the MC68EC040 and MC68EC040V to the access control registers. Refer to **Appendix B MC68EC040** for details.

The M68040 contains two independent, 4-Kbyte, on-chip caches located in the physical address space. Accessing instruction words and data simultaneously through separate caches increases instruction throughput. The M68040 caches improve system performance by providing cached data to the on-chip execution unit with very low latency. Systems with an alternate bus master receive increased bus availability.

Figure 4-1 illustrates the instruction and data caches contained in the instruction and data memory units. The appropriate memory unit independently services instruction prefetch and data requests from the integer unit (IU). The memory units translate the logical address in parallel with indexing into the cache. If the translated address matches one of the cache entries, the access hits in the cache. For a read operation, the memory unit supplies the data to the IU, and for a write operation, the memory unit updates the cache. If the access does not match one of the cache entries (misses in the cache) or a write access must be written through to memory, the memory unit sends an external bus request to the bus controller. The bus controller then reads or writes the required data.

Cache coherency in the M68040 is optimized for multimaster applications in which the M68040 is the caching master sharing memory with one or more noncaching masters (such as DMA controllers). The M68040 implements a bus snooper that maintains cache coherency by monitoring an alternate bus master's access and performing cache maintenance operations as requested by the alternate bus master. Matching cache entries can be invalidated during the alternate bus master's access to memory, or memory can be inhibited to allow the M68040 to respond to the access as a slave. For an external write operation, the processor can intervene in the access and update its internal caches (sink data). For an external read operation, the processor supplies cached data to the alternate bus muster (source data). This prevents the M68040 caches from accumulating old or invalid copies of data (stale data). Alternate bus masters are allowed access to locally modified data within the caches that is no longer consistent with external memory (dirty data). Allowing memory pages to be specified as write-through instead of copyback also supports cache coherency. When a processor writes to write-through pages, external

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

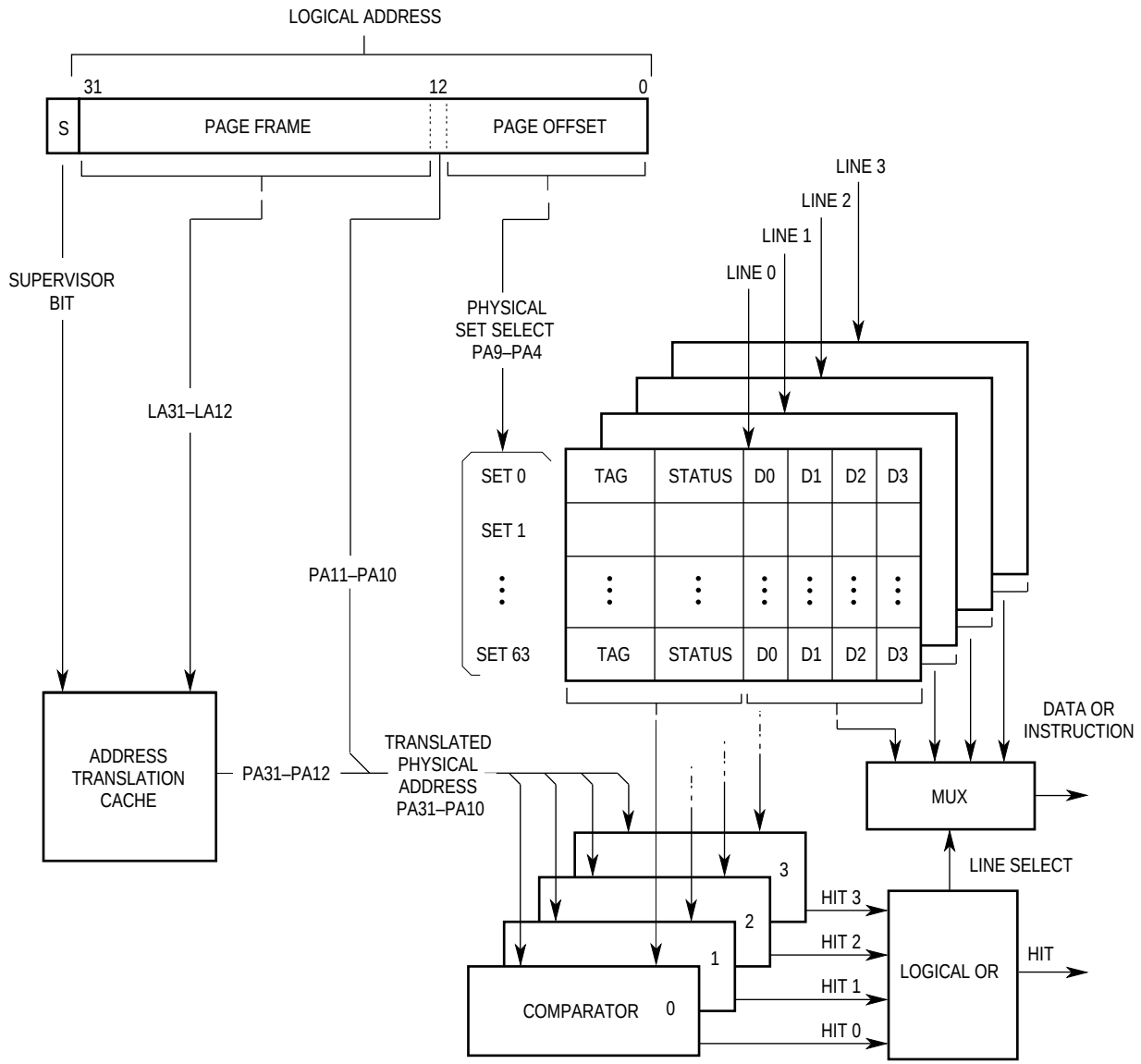


Figure 4-3. Caching Operation

Both caches contain circuitry to automatically determine which cache line in a set to use for a new line. The cache controller locates the first invalid line and uses it; if no invalid lines exist, then a pseudo-random replacement algorithm is used to select a valid line, replacing it with the new line. Each cache contains a 2-bit counter, which is incremented for each access to the cache. The instruction cache counter is incremented for each half-line accessed in the instruction cache. The data cache counter is incremented for each half-line accessed during reads, for each full line accessed during writes in copyback mode, and for each bus transfer resulting from a write in write-through mode. When a miss occurs and all four lines in the set are valid, the line pointed to by the current counter value is replaced, after which the counter is incremented.

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

4.3.2 Cache-Inhibited Accesses

Address space regions containing targets such as I/O devices and shared data structures in multiprocessing systems can be designated cache inhibited. If a page descriptor's CM field indicates nonserialized or serialized, then the access is cache inhibited. The caching operation is identical for both cache-inhibited modes. If the CM field of a matching address indicates either nonserialized or serialized modes, the cache controller bypasses the cache and performs an external bus transfer. The data associated with the access is not cached internally, and the cache inhibited out (CIOUT) signal is asserted during the bus transfer to indicate to external memory that the access should not be cached. If the data cache line is already resident in an internal cache, then the data cache line is pushed from the cache if it is dirty or the data cache line is invalidated if it is valid.

If the CM field indicates serialized, then the sequence of read and write accesses to the page is guaranteed to match the sequence of the instruction order. Without serialization, the IU pipeline allows read accesses to occur before completion of a write-back for a previous instruction. Serialization forces operand read accesses for an instruction to occur only once by preventing the instruction from being interrupted after the operand fetch stage. Otherwise, the instruction is aborted, and the operand is accessed when the instruction is restarted. These guarantees apply only when the CM field indicates the serialized mode and the accesses are aligned. Regardless of the selected cache mode, locked accesses are implicitly serialized. The TAS, CAS, and CAS2 instructions use locked accesses for operands in memory and for updating translation table entries during table search operations.

4.3.3 Special Accesses

Several other processor operations result in accesses that have special caching characteristics besides those with an implied cache-inhibited access in the serialized mode. Exception stack accesses, exception vector fetches, and table searches that miss in the cache do not allocate cache lines in the data cache, preventing replacement of a cache line. Cache hits by these accesses are handled in the normal manner according to the caching mode specified for the accessed address.

Accesses by the MOVE16 instruction also do not allocate cache lines in the data cache for either read or write misses. Read hits on either valid or dirty cache lines are read from the cache. Write hits invalidate a matching line and perform an external access. Interacting with the cache in this manner prevents a large block move or block initialization implemented with a MOVE16 from being cached, since the data may not be needed immediately.

If the data cache is re-enabled after a locked access has hit and the data cache was disabled, the next non-locked access that results in a data cache miss will not be cached.

4.4 CACHE PROTOCOL

The cache protocol for processor and snooped accesses is described in the following paragraphs. In all cases, an external bus transfer will cause a cache line state to change

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

6.2 INSTRUCTION SHIFT REGISTER

The M68040 IEEE standard 1149.1A implementation includes a 3-bit instruction shift register without parity. The register shifts one of eight instructions, which can either select the test to be performed or access a test data register, or both. Data is transferred from the instruction shift register to latched decoded outputs during the update-IR state. The instruction shift register is reset to all ones in the TAP controller test-logic-reset state, which is equivalent to selecting the BYPASS instruction. During the capture-IR state, the binary value 001 is loaded into the parallel inputs of the instruction shift register.

The M68040 IEEE standard 1149.1A implementation includes three mandatory public instructions (BYPASS, SAMPLE/PRELOAD, and EXTEST) and four manufacturer's public instructions. The four manufacturer's public instructions provide the capability to disable all device output drivers, operate the device in a BYPASS configuration without a system clocking requirement, and select one of two output drive capabilities on a pin-by-pin basis. The M68040 implementation does not support the optional standard public instructions. Table 6-1 lists the three bits used in the instruction shift register to decode the instructions and their related encodings. Note that the least significant bit of the instruction (bit 0) is the first bit to be shifted into the instruction shift register.

Table 6-1. IEEE Standard 1149.1A Instructions

Bit 2	Bit 1	Bit 0	Instruction Selected	Test Data Register Accessed
0	0	0	EXTEST	Boundary Scan
0	0	1	HIGHZ	Bypass
0	1	0	SAMPLE/PRELOAD	Boundary Scan
0	1	1	DRVCTL.T	Boundary Scan
1	0	0	SHUTDOWN	Bypass
1	0	1	PRIVATE	Bypass
1	1	0	DRVCTL.S	Boundary Scan
1	1	1	BYPASS	Bypass

EXTEST, HIGHZ, DRVCTL.T, SHUTDOWN, and PRIVATE have a PCLK and BCLK restriction. Failure to comply with this restriction results in potential internal damage to the device (see **6.4 Restrictions**). Once the restriction is complied with, SHUTDOWN, EXTEST, HIGHZ, and DRVCTL.T can be entered regardless of order. The system clocks (PCLK and BCLK) must be kept running while in the SAMPLE/PRELOAD, DRVCLT.S, and BYPASS instructions. Failure to do so could result in potential internal damage to the device.

6.2.1 EXTEST

The external test instruction (EXTEST) selects the 184-bit boundary scan register. This instruction also activates two internal functions that are intended to protect the device from potential damage while performing boundary scan operations.

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

The combination of operand size and alignment determines the number of bus cycles required to perform a particular memory access. Table 7-3 lists the number of bus cycles required for different operand sizes with all possible alignment conditions for read and write cycles. The table confirms that alignment significantly affects bus cycle throughput for noncachable accesses. For example, in Figure 7-5 the misaligned long-word operand took three bus cycles because the byte offset = $1. If the byte offset = $0, then it would have taken one bus cycle. The M68040 system designer and programmer should account for these effects, particularly in time-critical applications.

Table 7-3. Memory Alignment Influence on Noncachable and Write-Through Bus Cycles

Transfer Size	Number of Bus Cycles			
	$0*	$1*	$2*	$3*
Instruction	1	N/A	N/A	N/A
Byte Operand	1	1	1	1
Word Operand	1	2	1	2
Long-Word Operand	1	3	2	3

*Where the byte offset (A1 and A0) equals this encoding.

The processor always prefetches instructions by reading a long word from a half-line address (A2–A0 = $0), regardless of alignment. When the required instruction begins at the second long word, the processor attempts to fetch the entire half-line (two long words) although the second long word contains the required instruction.

7.4 PROCESSOR DATA TRANSFERS

The transfer of data between the processor and other devices involves the address bus, data bus, and control signals. The address and data buses are normally parallel, nonmultiplexed buses, supporting byte, word, long-word, and line (16-byte) bus cycles. Line transfers are normally performed using an efficient burst transfer, which provides an initial address and time-multiplexes the data bus to transfer four long words of information to or from the slave device. Slave devices that do not support bursting can burst-inhibit the first long word of a line transfer, forcing the bus master to complete the access using three additional long-word bus cycles. All bus input and output signals are synchronous to the rising edge of the BCLK signal. The M68040 moves data on the bus by issuing control signals and using a handshake protocol to ensure correct data movement. The following paragraphs describe the bus cycles for byte, word, long-word, and line read, write, and read-modify-write transfers.

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

7.4.1 Byte, Word, and Long-Word Read Transfers

During a read transfer, the processor receives data from a memory or peripheral device. Since the data read for a byte, word, or long-word access is not placed in either of the internal caches by definition, the processor ignores the level on the transfer cache inhibit ($\overline{TCI}$) signal when latching the data. The bus controller performs byte, word, and long-word read transfers for the following cases:

- Accesses to a disabled cache.
- Accesses to a memory page that is specified noncachable.
- Accesses that are implicitly noncachable (read-modify-write accesses and accesses to an alternate logical address space via the MOVES instruction).
- Accesses that do not allocate in the data cache on a read miss (table searches, exception vector fetches, and exception stack deallocation for an RTE instruction).
- The first transfer of a line read is terminated with transfer burst inhibit ($\overline{TBI}$), forcing completion of the line access using three additional long-word read transfers.

Figure 7-8 is a flowchart for byte, word, and long-word read transfers. Bus operations are similar for each case and vary only with the size indicated and the portion of the data bus used for the transfer. Figure 7-9 is a functional timing diagram for byte, word, and long-word read transfers.

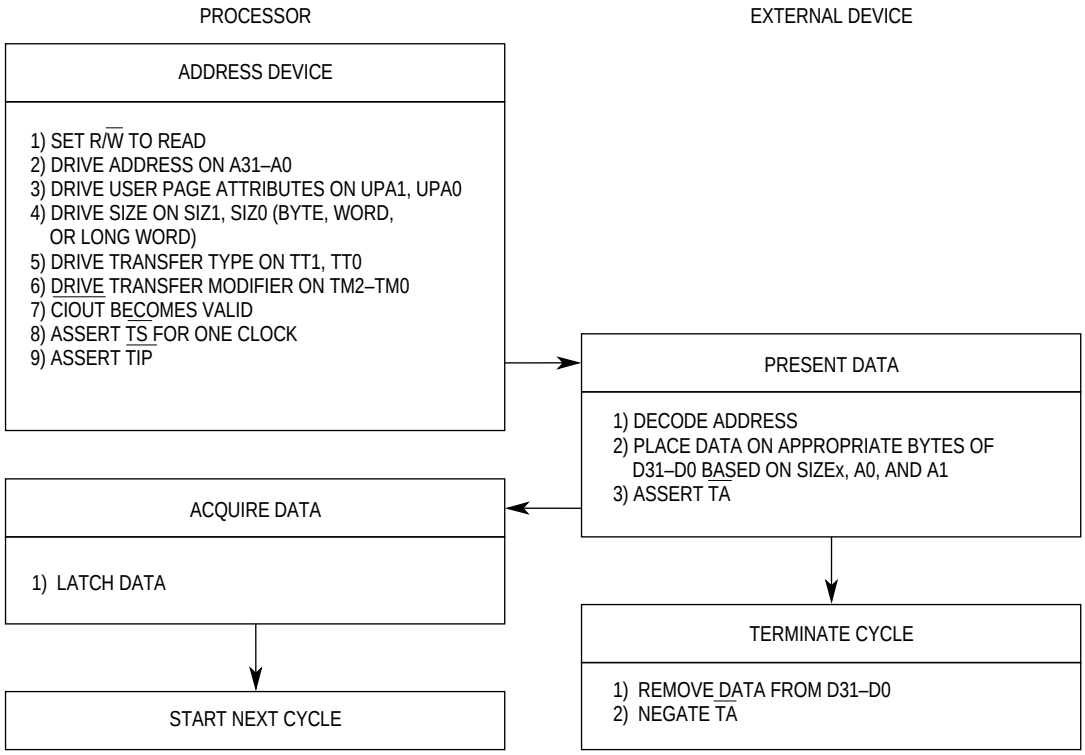


Figure 7-8. Byte, Word, and Long-Word Read Transfer Flowchart

For More Information On This Product,
Go to: www.freescale.com
Freescale Semiconductor, Inc.

7.9.1 Snoop-Inhibited Cycle

For alternate bus master accesses in which the SCx signal encodings indicate that snooping is inhibited (SCx = $0), the M68040 immediately negates $\overline{MI}$ and allows memory to respond to the access. Snoop-inhibited alternate bus master accesses do not affect performance of the processor since no cache lookups are required. Figure 7-40 illustrates an example of a snoop-inhibited operation in which an alternate bus master is granted the bus for an access. No matter what the values are on the SCx and TTx signals, $\overline{MI}$ is asserted between bus cycles. Because $\overline{MI}$ is asserted while a cache lookup is performed, snooping inherently degrades system performance.

$\overline{MI}$ is asserted from the last $\overline{TA}$ of the current bus cycle if the M68040 owns the bus and loses it (see Figure 7-40). If an alternate bus master has the bus and loses it, there are two different resulting cases. Usually, an idle clock occurs between the alternate bus master's cycle and the MC68040's cycle. If so, $\overline{MI}$ is asserted during the idle clock and negated from the same edge that the M68040 asserts the $\overline{TS}$ signal (see Figure 7-40). If there is no idle clock, $\overline{MI}$ is not asserted. $\overline{MI}$ is asserted during and after reset until the first bus cycle of the M68040. Even though snoop is inhibited, all $\overline{TA}$ or $\overline{TEA}$ assertions while $\overline{MI}$ is asserted are ignored. If a line snoop is started, the M68040 still requires four $\overline{TA}$ assertions.

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

illustrated, which results in a memory access having the equivalent of two wait states. Variations in the timing required by snooping logic to access the caches can delay the negation of $\overline{MI}$ by up to two additional clocks. External logic must ensure that the termination signals negate at all rising BCLK edges in which $\overline{MI}$ is asserted. Otherwise, if one of the termination signals is asserted, either the M68040 ignores all termination signals, reading them as negated, or the M68040 exhibits improper operation.

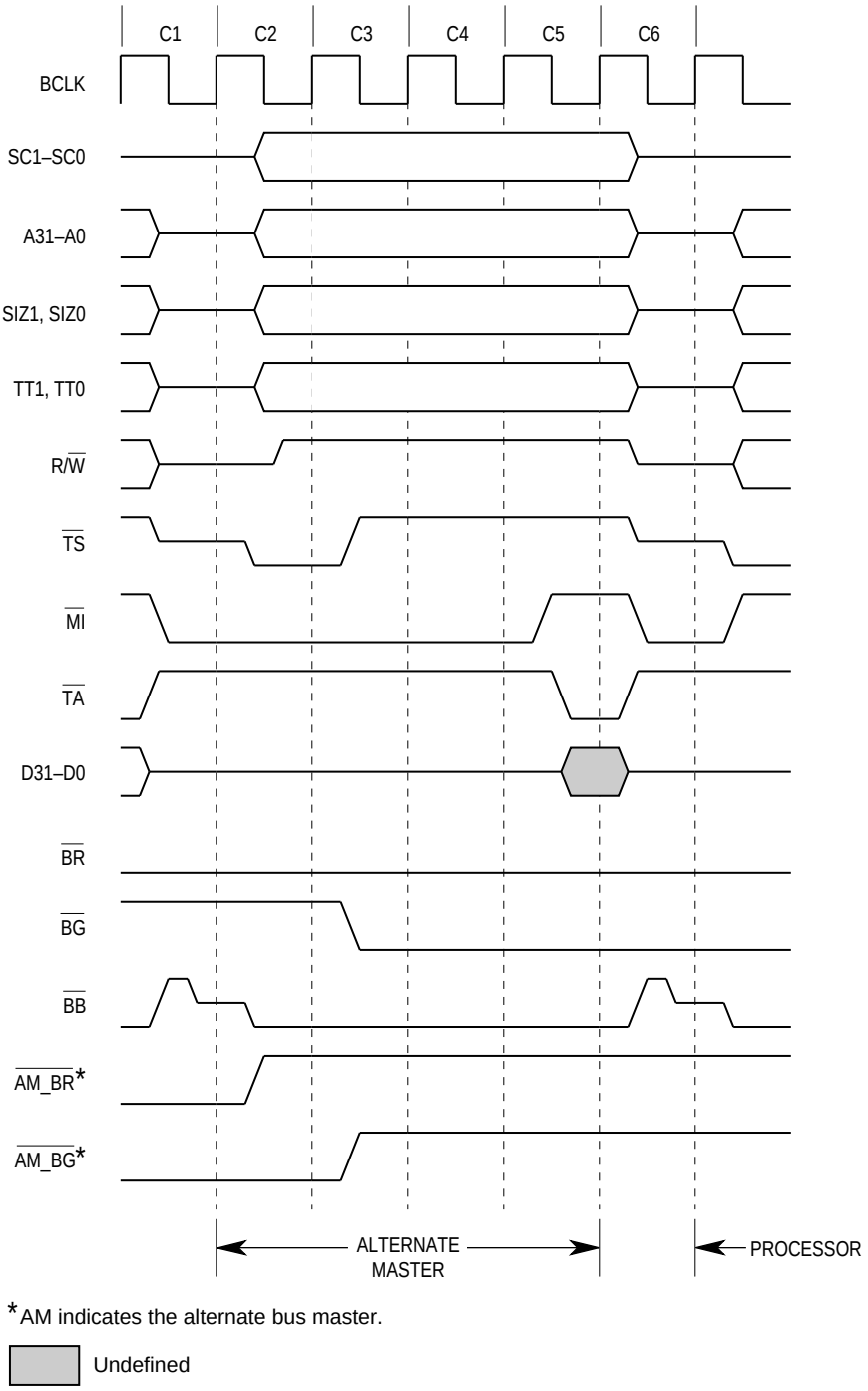


Figure 7-41. Snoop Access with Memory Response

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

3. Instruction access faults occur when the PC section is deadlocked because of the faulted data or another prefetch is required, the copyback stage is empty, and the data cache and bus controller are idle. Since instruction access faults are reset, they can be ignored.
4. An internal access fault also occurs when the data or instruction MMU detects that a successful address translation is not possible because the page is write protected, supervisor only, or nonresident. Furthermore, when an address translation cache (ATC) miss occurs, the processor searches the translation tables in memory for the mapping, and then retries the access. If a valid translation for the logical address is not available due to a problem encountered during the table search, an internal access fault occurs when the aborted access is retried. The problem encountered could be either an invalid descriptor or the assertion of the $\overline{\text{TEA}}$ signal during a bus cycle used to access the translation tables. A miss in the ATC causes the processor to automatically initiate a table search but does not cause an internal access fault unless one of the three previous conditions is encountered. However, this is not true if the memory management unit (MMU) is disabled.

When an exception is detected, all parts of the execution unit either remain or are forced to idle, at which time the highest priority exception is taken. Restarting the instruction or a user-defined supervisor cleanup exception handler routine regenerates lower priority exceptions on the return from exception handling. Internal access faults and bus errors are reported after all other pending integer instructions complete execution. If an exception is generated during completion of the earlier instructions, the pending instruction fault is cleared, and the new exception is serviced first. The processor restarts the pending prefetch after completing exception handling for the earlier instructions and takes a bus error exception if the access faults again. For data access faults, the processor aborts current instruction execution. If a data access fault is detected, the processor waits for the current instruction prefetch bus cycle to complete, then begins exception processing immediately.

As illustrated in Figure 8-1, the processor begins exception processing for an access fault by making an internal copy of the current SR. The processor then enters the supervisor mode and clears T1 and T0. The processor generates exception vector number 2 for the access fault vector. It saves the vector offset, PC, and internal copy of the SR on the stack. The saved PC value is the logical address of the instruction executing at the time the fault was detected. This instruction is not necessarily the one that initiated the bus cycle since the processor overlaps execution of instructions. It also saves information to allow continuation after a fault during a MOVEM instruction and to support other pending exceptions. The faulted address and pending write-back information is saved. The information saved on the stack is sufficient to identify the cause of the bus error, complete pending write-backs, and recover from the error. The exception handler must complete the pending write-backs. Up to three write-backs can be pending for push errors and data access errors.

If a bus error occurs during the exception processing for an access fault, address error, or reset or while the processor is loading internal state information from the stack during the execution of an RTE instruction, a double bus fault occurs, and the processor enters the halted state as indicated by the PST3–PST0 encoding $5. In this case, the processor

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

Table 8-6. Access Error Stack Frame Combinations

Main Case	SSW_RW	SSW_PUSH	WB1S		WB2S		WB3S	Easy Cleanup Data Written	Hard Cleanup Action
			1V	1M16	2V	2M16	3V		
All Read Access Errors	1[a]	No	0	X	0	X	0	None	(Note b)
	1[a]	No	0	X	0	X	1	WB3D	
	All other read cases are not possible.								
Cache Push Physical Bus Error[c]	0	Yes	0	X	0	X	0	PD3–0	(Note b)
	0	Yes	0	X	0	X	1	PD3–0, WB3D	
	0	Yes	0	X	1	0	0	PD3–0, WB2D	
	0	Yes	0	X	1	0	1	PD3–0, WB2D, WB3D	
	0	Yes	0	X	1	1	0	PD3–0, ~WB2D[d]	
Normal Write Physical bus Error	0	No	1	0	0	X	0	WB1D	(Note b)
	0	No	1	0	0	X	1	WB1D, WB3D	
	0	No	1	0	1	0	0	WB1D, WB2D	
	0	No	1	0	1	0	1	WB1D, WB2D, WB3D	
	0	No	1	0	1	1	0	WB1D, ~WB2D[d]	
MOVE16 Write Physical Bus Error	0	No	1	1	0	X	1	PD3–0, WB3D	(Note b)
	0	No	1	1	0	X	0	PD3–0	
	0	No	1	1	1	0	0	PD3–0, WB2D	
	0	No	1	1	1	0	1	PD3–0, WB2D, WB3D	
	0	No	1	1	1	1	0	PD3–0, ~WB2D[d]	
Write Page Fault	0	No	0	X	1	0	0	WB2D	Write PD3–0 and skip[e].
	0	No	0	X	1	0	1	WB2D, WB3D	
	0	No	0	X	1	1	0	~WB2D[d]	
Impossible Write Cases	0	Yes	1	X	X	X	X	(Note f)	—
	0	Don't Care	X	X	X	1	1	(Note g)	

NOTES:

a. The data memory unit stage is tied up until the bus controller passes the read back through the data memory unit and to the execution stage in the integer unit. Therefore, no pending write is possible in WB1 or WB2. WB3 could hold a pending write that was deferred due to operand read or was generated after the read.

b. If any kind of access error is reported and if a MOVE16 write is pending in the WB2 stage, then that MOVE16 read must hit in the cache so the MOVE16 can be safely restarted since it has not caused bus cycles that could retouch peripherals.

c. A cache push physical bus error is normally considered a fatal error. For these cases, the FA field is a physical address, not a logical address as in the other cases.

d. Indicates that the data should not be written even though the V-bit for it is set (WB2 corresponds to a MOVE16 write).

e. The exception handler must alter the stacked PC to point past the MOVE16 and predecrement and postincrement address registers.

f. 1V must be 0 for push exceptions.

g. The execution stage does not post a write until the MOVE16 is in the integer unit.

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

MC68040 produces the same results as any other device that conforms to the IEEE 754 standard but does not support extended precision. The results are the same when performing the same operation in extended precision and storing the results in single- or double-precision format.

The FPU performs all floating-point internal operations in extended precision. It supports mixed-mode arithmetic by converting single- and double-precision operands to extended-precision values before performing the specified operation. The FPU converts all memory data formats to extended-precision before using it in a floating-point operation or loading it in a floating-point data register. The FPU also converts extended-precision data formats in a floating-point data register to any data format and either stores it in a memory destination or in an integer data register.

If the external operand is a denormalized number, the number is normalized before an operation is performed. However, an external denormalized number moved into a floating-point data register is stored as a denormalized number.

If an external operand is an unnormalized number, the number is normalized before it is used in an arithmetic operation. If the external operand is an unnormalized zero (i.e., with a mantissa of all zeros), the number is converted to a normalized zero before the specified operation is performed. The regular use of unnormalized inputs not only defeats the purpose of the IEEE 754 standard, but also can produce gross inaccuracies in the results.

9.4.1 Intermediate Result

Figure 9-7 illustrates the intermediate result format. The intermediate result's exponent for some dyadic operations (i.e., multiply and divide) can easily overflow or underflow the 15-bit exponent of the destination floating-point register. To simplify the overflow and underflow detection, intermediate results in the FPU maintain a 16-bit, twos-complement integer exponent. Detection of an overflow or underflow intermediate result always converts the 16-bit exponent into a 15-bit biased exponent before being stored in a floating-point data register. The FPU internally maintains the 67-bit mantissa for rounding purposes. The mantissa is always rounded to 64 bits (or less, depending on the selected rounding precision) before it is stored in a floating-point data register.

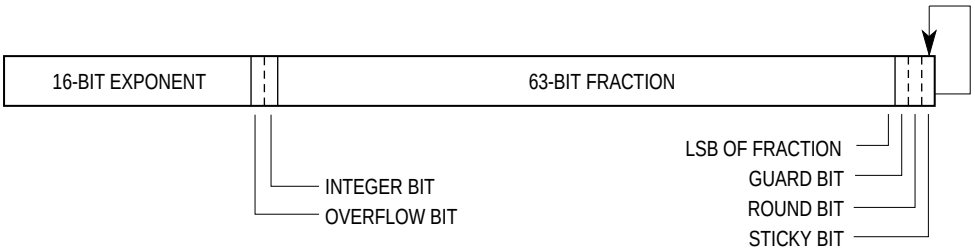


Figure 9-7. Intermediate Result Format

If the destination is a floating-point data register, the result is in the extended-precision format and is rounded to the precision specified by the FPSR PREC bits before being stored. All mantissa bits beyond the selected precision are zero. If the single- or double-

For More Information On This Product,
Go to: www.freescale.com
Freescale Semiconductor, Inc.

The CMDREG1B field of the floating-point state frame can be used to determine the instruction that caused of the OPERR exception. Note that CMDREG1B could be any of the instructions listed in Table 9-11. If the destination is a floating-point data register, this exception handler needs to supply the contents. If the destination is memory, the effective address is supplied in the format $3 stack frame. If the destination is an integer data register, the FPIAR points to the F-line instruction word that contains the integer data register number. To exit the user OPERR exception handler, the saved floating-point frame need not be restored and can be discarded prior to execution of the RTE instruction.

9.7.4 Overflow

An overflow exception is detected for arithmetic operations in which the destination is a floating-point data register or memory when the intermediate result's exponent is greater than or equal to the maximum exponent value of the selected rounding precision. Overflow can only occur when the destination is in the single-, double-, or extended-precision format; all other data format overflows are handled as operand errors. At the end of any operation that could potentially overflow, the intermediate result is checked for underflow, rounded, and then checked for overflow before it is stored to the destination. If overflow occurs, the OVFL bit is set in the FPSR EXC byte.

Even if the intermediate result is small enough to be represented as an extended-precision number, an overflow can occur. The intermediate result is rounded to the selected precision, and the rounded result is stored in the extended-precision format. If the magnitude of the intermediate result exceeds the range of the selected rounding precision format, an overflow occurs.

9.7.4.1 MASKABLE EXCEPTION CONDITIONS. There are no conditions.

9.7.4.2 NONMASKABLE EXCEPTION CONDITIONS. When the OVFL bit is set in the FPSR EXC byte as a result of a floating-point instruction, the processor always takes a nonmaskable overflow exception. If the destination is a floating-point data register, then the register is not affected, and either a pre-instruction or a post-instruction exception is reported. If the destination is a memory or integer data register, an undefined result is stored, and a post-instruction exception is taken immediately. Execution begins at the M68040FPSP OVFL exception handler.

The values defined in Table 9-12 are stored in the destination based on the rounding mode defined in the FPCR MODE byte. The M68040FPSP OVFL exception handler rounds the result according to the rounding precision defined in the FPCR MODE byte if the destination is a floating-point data register. If the destination is in memory or an integer data register, then the rounding precision in the FPCR MODE byte is ignored, and the given destination format defines the rounding precision. If the instruction has a forced rounding precision (e.g., FSADD, FDMUL), the instruction defines the rounding precision. The M68040FPSP OVFL exception handler then checks to see if the user OVFL exception handler is enabled.

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

11.7 INPUT AC TIMING SPECIFICATIONS (see Figures 11-3 to 11-7)

Num	Characteristic	25 MHz		33 MHz		40 MHz		Unit
		Min	Max	Min	Max	Min.	Max.	
15	Data-In Valid to BCLK (Setup)	5	—	4	—	3	—	ns
16	BCLK to Data-In Invalid (Hold)	4	—	4	—	3	—	ns
17	BCLK to Data-In High Impedance (Read Followed by Write)	—	49	—	36.5	—	30.25	ns
22a	$\overline{TA}$ Valid to BCLK (Setup)	10	—	10	—	8	—	ns
22b	$\overline{TEA}$ Valid to BCLK (Setup)	10	—	10	—	9	—	ns
22c	$\overline{TCI}$ Valid to BCLK (Setup)	10	—	10	—	9	—	ns
22d	$\overline{TBI}$ Valid to BCLK (Setup)	11	—	10	—	9	—	ns
23	BCLK to $\overline{TA}$, $\overline{TEA}$, $\overline{TCI}$, $\overline{TBI}$ Invalid (Hold)	2	—	2	—	2	—	ns
24	$\overline{AVEC}$ Valid to BCLK (Setup)	5	—	5	—	5	—	ns
25	BCLK to $\overline{AVEC}$ Invalid (Hold)	2	—	2	—	2	—	ns
31	DLE Width High	8	—	8	—	8	—	ns
32	Data-In Valid to DLE (Setup)	2	—	2	—	2	—	ns
33	DLE to Data-In Invalid (Hold)	8	—	8	—	8	—	ns
34	BCLK to DLE Hold	3	—	3	—	3	—	ns
35	DLE High to BCLK	16	—	12	—	12	—	ns
36	Data-In Valid to BCLK (DLE Mode Setup)	5	—	5	—	5	—	ns
37	BCLK to Data-In Invalid (DLE Mode Hold)	4	—	4	—	4	—	ns
41a	$\overline{BB}$ Valid to BCLK (Setup)	7	—	7	—	7	—	ns
41b	$\overline{BG}$ Valid to BCLK (Setup)	8	—	7	—	7	—	ns
41c	$\overline{CDIS}$, $\overline{MDIS}$ Valid to BCLK (Setup)	10	—	8	—	8	—	ns
41d	$\overline{IPLx}$ Valid to BCLK (Setup)	4	—	3	—	3	—	ns
42	BCLK to $\overline{BB}$, $\overline{BG}$, $\overline{CDIS}$, $\overline{IPLx}$, $\overline{MDIS}$ Invalid (Hold)	2	—	2	—	2	—	ns
44a	Address Valid to BCLK (Setup)	8	—	7	—	7	—	ns
44b	SIZx Valid to BCLK (Setup)	12	—	8	—	8	—	ns
44c	TTx Valid to BCLK (Setup)	6	—	8.5	—	8.5	—	ns
44d	R/$\overline{W}$ Valid to BCLK (Setup)	6	—	5	—	5	—	ns
44e	SCx Valid to BCLK (Setup)	10	—	11	—	8	—	ns
45	BCLK to Address,SIZx, TTx, R/$\overline{W}$, SCx Invalid (Hold)	2	—	2	—	2	—	ns
46	$\overline{TS}$ Valid to BCLK (Setup)	5	—	9	—	7	—	ns
47	BCLK to $\overline{TS}$ Invalid (Hold)	2	—	2	—	2	—	ns
49	BCLK to $\overline{BB}$ High Impedance (MC68040 Assumes Bus Mastership)	—	9	—	9	—	9	ns
51	$\overline{RSTI}$ Valid to BCLK	5	—	4	—	4	—	ns
52	BCLK to $\overline{RSTI}$ Invalid	2	—	2	—	2	—	ns
53	Mode Select Setup to $\overline{RSTI}$ Negated	20	—	20	—	20	—	ns
54	$\overline{RSTI}$ Negated to Mode Selects Invalid	2	—	2	—	2	—	ns

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

B.7.3 DC Electrical Specifications (V_{CC} = 5.0 Vdc ±5 %)

Characteristic	Symbol	Min	Max	Unit
Input High Voltage	V_{IH}	2	V_{CC}	V
Input Low Voltage	V_{IL}	GND	0.8	V
Undershoot	—	—	0.8	V
Input Leakage Current @ 0.5–2.4 V $\overline{AVEC}$, BCLK, $\overline{BG}$, $\overline{CDIS}$, $\overline{IPLA}$, PCLK, $\overline{RSTI}$, SCx, $\overline{TBI}$, TLNx, $\overline{TCI}$, TCK, $\overline{TEA}$	I_{in}	20	20	mA
Hi-Z (Off-State) Leakage Current @ 0.5–2.4 V An, $\overline{BB}$, $\overline{CIOUT}$, Dn, $\overline{LOCK}$, $\overline{LOCKE}$, R/$\overline{W}$, SIZx, $\overline{TA}$, TDO, $\overline{TIP}$, TMx, TLNx, $\overline{TS}$, TTx, UPAx	I_{TSI}	20	20	mA
Signal Low Input Current, V_{IL} = 0.8 V TMS, TDI, $\overline{TRST}$	I_{IL}	–1.1	–0.18	mA
Signal High Input Current, V_{IH} = 2.0 V TMS, TDI, $\overline{TRST}$	I_{IH}	–0.94	–0.16	mA
Output High Voltage, I_{OH} = 5 mA	V_{OH}	2.4	—	V
Output Low Voltage, I_{OL} = 5 mA	V_{OL}	—	0.5	V
Capacitance*, V_{in} = 0 V, f = 1 MHz	C_{in}	—	25	pF

*Capacitance is periodically sampled rather than 100% tested.

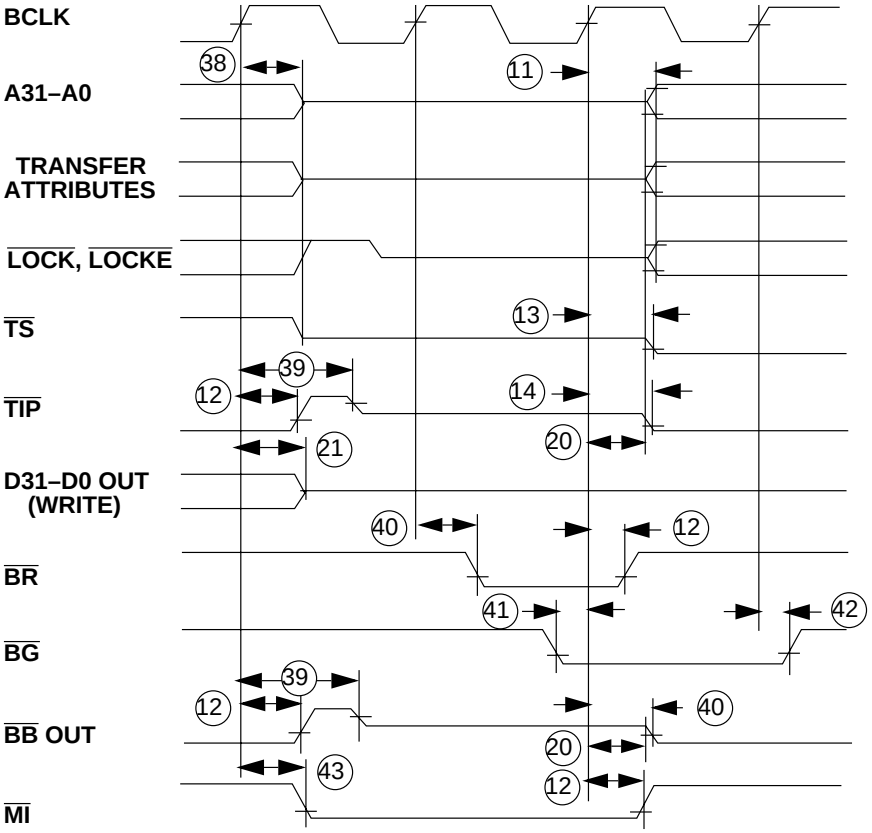
B.7.4 Power Dissipation

Frequency	Watts
Maximum Values (V_{CC} = 5.25 V, T_A = 0°C)	
20 MHz	3.2
25 MHz	3.9
33 MHz	4.9
40 MHz	5.5
Typical Values (V_{CC} = 5 V, T_A = 25°C)*	
20 MHz	2.0
25 MHz	2.4
33 MHz	3.0
40 MHz	3.5

*This information is for system reliability purposes.

For More Information On This Product,
Go to: www.freescale.com

Freescale Semiconductor, Inc.

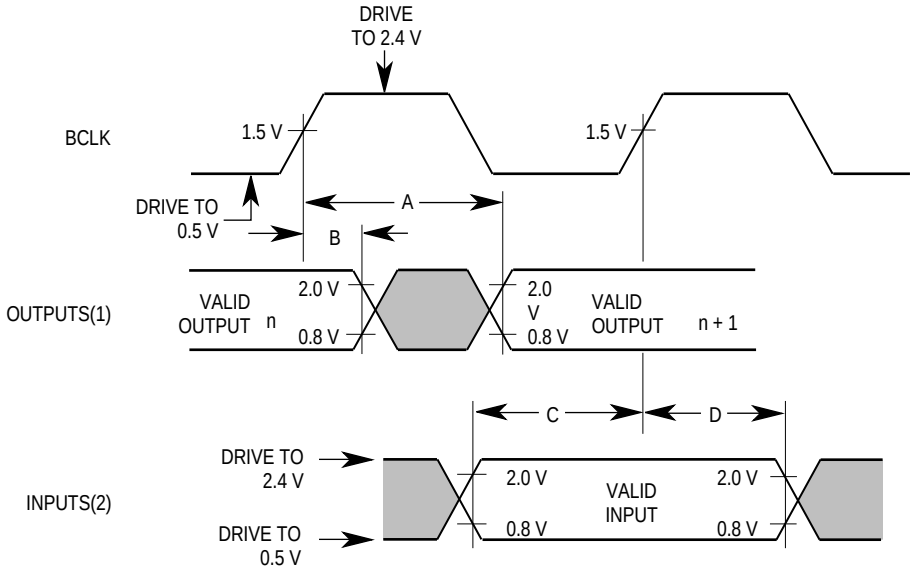


NOTE: Transfer Attribute Signals = UPAx, SIZx, TTx, TMx, TLNx, R/$\overline{W}$, and $\overline{CIOUT}$

Figure B-9. Bus Arbitration Timing

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com



NOTES:

1. This output timing is applicable to all parameters specified relative to the rising edge of the clock.
2. This input timing is applicable to all parameters specified relative to the rising edge of the clock.

LEGEND:

A. Maximum output delay specification.
B. Minimum output hold time.
C. Minimum input setup time specification.
D. Minimum input hold time specification.

Figure C-11. Drive Levels and Test Points for AC Specifications

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

–U–

–V–

–W–

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com