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	20MHz
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	184-BCQFP
Supplier Device Package	184-CQFP (31.3x31.3)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mc68lc040fe20a

uses. The IU identifies a logical address by accessing either the supervisor or user address space, maintaining the differentiation between supervisor and user modes. The MMUs use the indicated privilege mode to control and translate memory accesses, protecting supervisor code, data, and resources from user program accesses. Refer to **Appendix B MC68EC040** for details concerning the MC68EC040 address translation.

Programs access registers based on the indicated mode. User programs can only access registers specific to the user mode; whereas, system software executing in the supervisor mode can access all registers, using the control registers to perform supervisory functions. User programs are thus restricted from accessing privileged information, and the operating system performs management and service tasks for the user programs by coordinating their activities. This difference allows the supervisor mode to protect system resources from uncontrolled accesses.

Most instructions execute in either mode, but some instructions that have important system effects are privileged and can only execute in the supervisor mode. For instance, user programs cannot execute the STOP or RESET instructions. To prevent a user program from entering the supervisor mode, except in a controlled manner, instructions that can alter the S-bit in the SR are privileged. The TRAP instructions provide controlled access to operating system services for user programs.

If the S-bit in the SR is set, the processor executes instructions in the supervisor mode. Because the processor performs all exception processing in the supervisor mode, all bus cycles generated during exception processing are supervisor references, and all stack accesses use the active supervisor stack pointer. If the S-bit of the SR is clear, the processor executes instructions in the user mode. The bus cycles for an instruction executed in the user mode are user references. The values on the transfer modifier pins indicate either supervisor or user accesses.

The processor utilizes the user mode and the user programming model when it is in normal processing. During exception processing, the processor changes from user to supervisor mode. Exception processing saves the current value of the SR on the active supervisor stack and then sets the S-bit, forcing the processor into the supervisor mode. To return to the user mode, a system routine must execute one of the following instructions: MOVE to SR, ANDI to SR, EORI to SR, ORI to SR, or RTE, which execute in the supervisor mode, modifying the S-bit of the SR. After these instructions execute, the instruction pipeline is flushed and is refilled from the appropriate address space.

The MC68040 integrates the functions of the IU, FPU, and MMU. The registers depicted in the programming model (see Figure 1-2) provide operand storage and control for these three units. The registers are partitioned into two levels of privilege modes: user and supervisor. The user programming model is the same as the user programming model of the MC68030, which consists of 16, general-purpose, 32-bit registers and two control registers. The MC68040 user programming model also incorporates the MC68881/MC68882 programming model consisting of eight, 80-bit, floating-point data registers, a floating-point control register, a floating-point status register, and a floating-point instruction address register.

Table 1-3. Notational Conventions (Continued)

Data Format And Type	
+ inf	Positive Infinity
<fmt>	Operand Data Format: Byte (B), Word (W), Long (L), Single (S), Double (D), Extended (X), or Packed (P).
B, W, L	Specifies a signed integer data type (twos complement) of byte, word, or long word.
D	Double-precision real data format (64 bits).
k	A twos complement signed integer (–64 to +17) specifying a number's format to be stored in the packed decimal format.
P	Packed BCD real data format (96 bits, 12 bytes).
S	Single-precision real data format (32 bits).
X	Extended-precision real data format (96 bits, 16 bits unused).
– inf	Negative Infinity
Subfields and Qualifiers	
#<xxx> or #<data>	Immediate data following the instruction word(s).
( )	Identifies an indirect address in a register.
[ ]	Identifies an indirect address in memory.
bd	Base Displacement
ccc	Index into the MC68881/MC68882 Constant ROM
d_n	Displacement Value, n Bits Wide (example: d_{16} is a 16-bit displacement).
LSB	Least Significant Bit
LSW	Least Significant Word
MSB	Most Significant Bit
MSW	Most Significant Word
od	Outer Displacement
SCALE	A scale factor (1, 2, 4, or 8, for no-word, word, long-word, or quad-word scaling, respectively).
SIZE	The index register's size (W for word, L for long word).
{offset:width}	Bit field selection.
Register Names	
CCR	Condition Code Register (lower byte of status register)
DFC	Destination Function Code Register
FPcr	Any Floating-Point System Control Register (FPCR, FPSR, or FPIAR)
FPm, FPn	Any Floating-Point Data Register specified as the source or destination, respectively.
IC, DC, IC/DC	Instruction, Data, or Both Caches
MMUSR	MMU Status Register
PC	Program Counter
Rc	Any Non Floating-Point Control Register
SFC	Source Function Code Register
SR	Status Register

S—Supervisor Protection

This bit is set if the S-bit in the page descriptor is set. Setting this bit does not indicate that a violation has occurred.

CM—Cache Mode

This 2-bit field is copied from the CM bits in the page descriptor.

M—Modified

This bit is set if the M-bit is set in the page descriptor associated with the address.

W—Write Protect

This bit is set if the W-bit is set in any of the descriptors encountered during the table search. Setting this bit does not indicate that a violation has occurred.

T—Transparent Translation Register Hit

If the T-bit is set, then the PTEST address matches an instruction or data TTR, the R-bit is set, and all other bits are zero.

R—Resident

The R-bit is set if the PTEST address matches an instruction or data TTR or if the table search completes by obtaining a valid page descriptor.

3.2 LOGICAL ADDRESS TRANSLATION

The function of the MMUs is to translate logical addresses to physical addresses. The MMUs perform translations according to control information in translation tables. The operating system creates these translation tables and stores them in memory. The processor then fetches a translation table as needed and stores it in an ATC.

3.2.1 Translation Tables

The M68040 uses the ATCs in the instruction and data memory units with translation tables stored in memory to perform the translations from logical to physical addresses. The operating system loads the translation tables for a program into memory. No distinction is made in the translation of instruction accesses versus data accesses because the instruction and data MMUs access the same translation table for a specific privilege mode, either user or supervisor. This lack of distinction results in a merged instruction and data address space.

Figure 3-7 illustrates the three-level tree structure of a general translation table supported by the M68040. The root- and pointer-level tables contain the base addresses of the tables at the next level. The page-level tables contain either the physical address for the translation or a pointer to the memory location containing the physical address. Only a portion of the translation table for the entire logical address space is required to be resident in memory at any time—specifically, only the portion of the table that translates

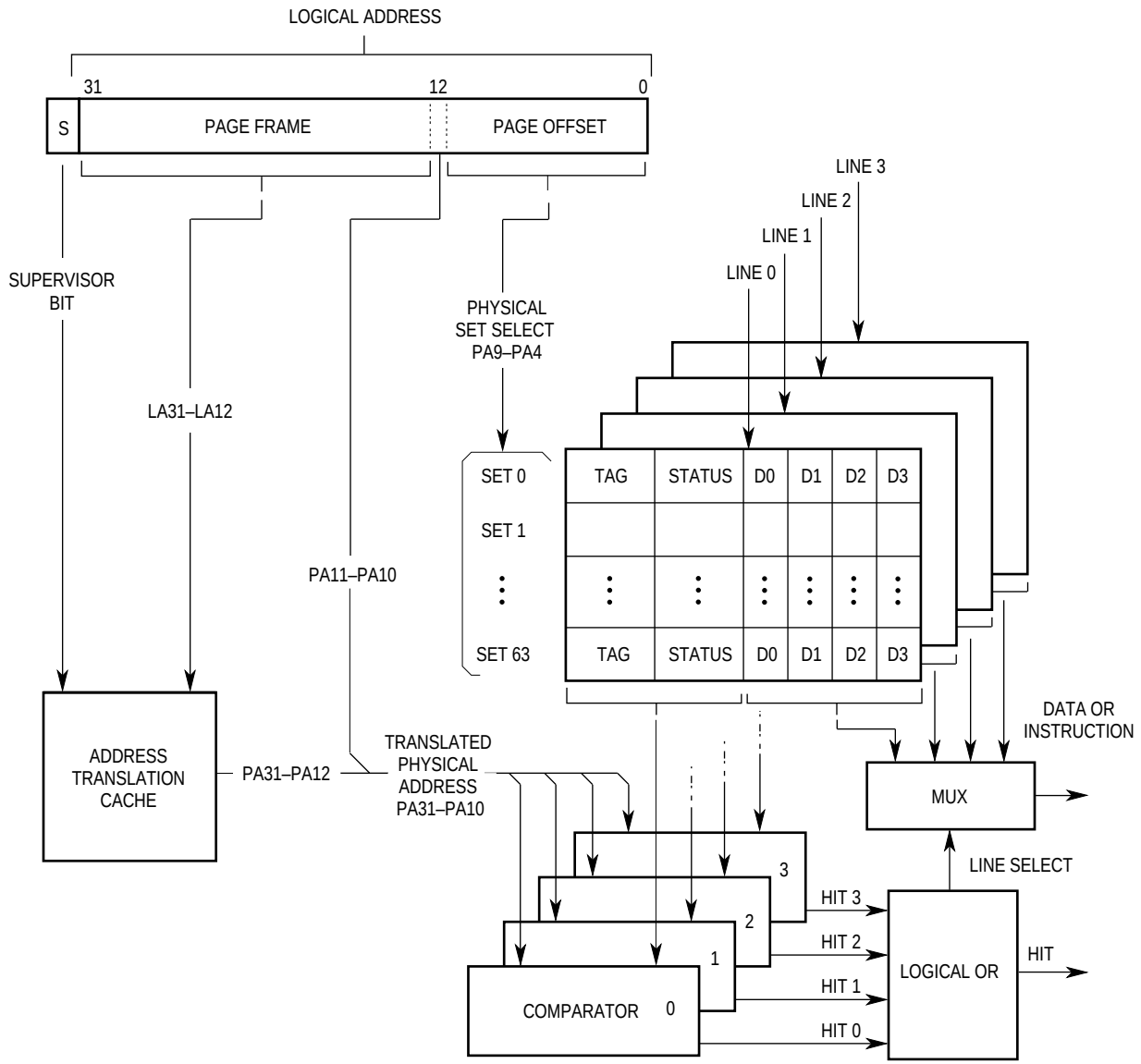


Figure 4-3. Caching Operation

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.

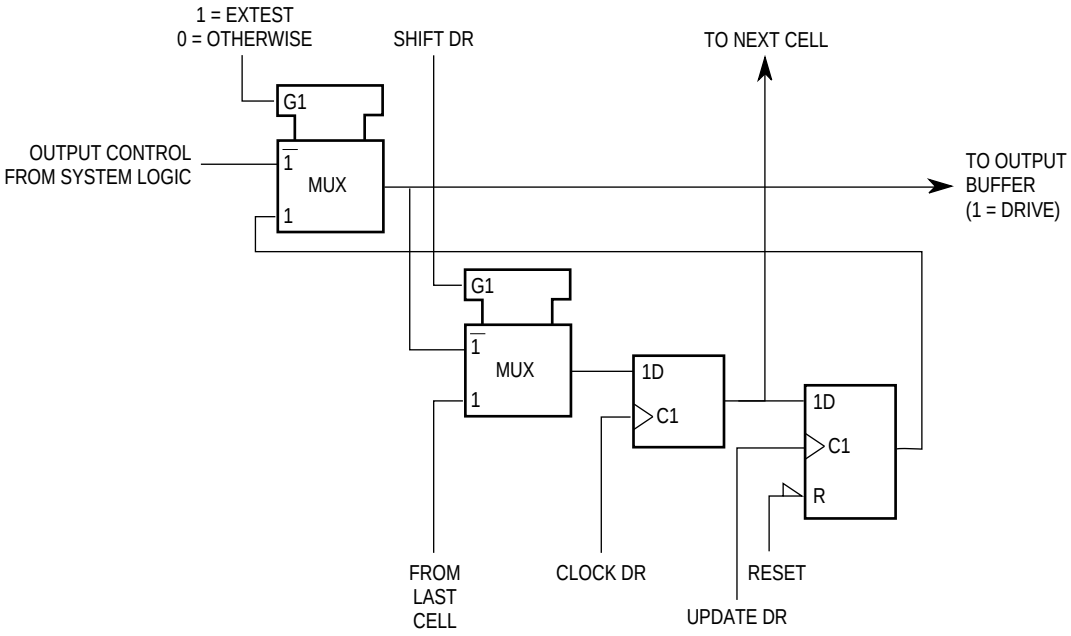


Figure 6-5. Output Control Cells (IO.Ctl)

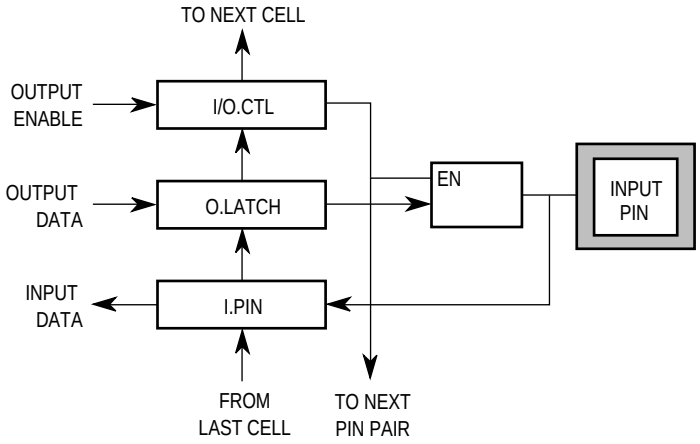


Figure 6-6. General Arrangement of Bidirectional Pins

6.6 MOTOROLA M68040 BSDL DESCRIPTION (VERSION 2.2)

Revision List:

1. LOCK and LOCKE controlled by io.1 vice io.0 (4D98D).
3. No other changes to Version 2.1 BSDL.
2. Instruction opcodes changed for SAMPLE, SHUTDOWN, and BYPASS.
3. New instructions DRVCTL.T, DRVCTL.S and PRIVATE added.
4. New instructions DRVCTL.T and DRVCTL.S renamed to DRVCTL_T and DRVCTL_S for syntax compatibility.
5. Register access specified for DRVCTL_T, DRVCTL_S, and PRIVATE instructions.
6. No other changes to Version 1.0 BSDL.

Package Type: 18 x 18 PGA

This BSDL is for the newer MC68040 mask sets of E26A and after (roughly after the second half of 1992). It does not include the 0.8-µm mask sets D43B, D50D, and D98D. For MC68LC040 and MC68EC040, two pin names have changed. To make the necessary modifications, change all occurrences of DLE to JS0 and `MDIS` to JS1.

```
entity MC68040 is

generic(PHYSICAL_PIN_MAP:string := "PGA_18x18");

port  (TDI:     in          bit;
       TDO:     out         bit;
       TMS:     in          bit;
       TCK:     in          bit;
       TRST:    in          bit;
       RSTO:    buffer      bit;
       IPEND:   buffer      bit;
       CIOUT:   out         bit;
       UPA:     out         bit_vector(0 to 1);
       TT:      inout       bit_vector(0 to 1);
       A:       inout       bit_vector(0 to 31);
       D:       inout       bit_vector(0 to 31);
       LOCKE:   out         bit;
       LOCK:    out         bit;
       R_W:     inout       bit;
       TLN:     out         bit_vector(0 to 1);
       TM:      out         bit_vector(0 to 2);
       SIZ:     inout       bit_vector(0 to 1);
       MI:      buffer      bit;
       BR:      buffer      bit;
       TS:      inout       bit;
       BB:      inout       bit;
       TIP:     out         bit;
       PST:     buffer      bit_vector(0 to 3);
       TA:      inout       bit;
       TEA:     in          bit;
       BG:      in          bit;
       SC:      in          bit_vector(0 to 1);
       TBI:     in          bit;
       AVEC:    in          bit;
       TCI:     in          bit;
```

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.

On the initial cycle of a line transfer, a retry causes the processor to retry the bus cycle as illustrated in Figure 7-29. However, the processor recognizes a retry signaled during the second, third, or fourth cycle of a line as a bus error and causes the processor to abort the line transfer. A burst-inhibited line transfer can only be retried on the initial transfer. A burst-inhibited line transfer aborts if a retry is signaled for any of the three long-word transfers used to complete the line transfer. Negating the bus grant ($\overline{BG}$) signal on the M68040 while asserting both $\overline{TA}$ and $\overline{TEA}$ provides a relinquish and retry operation for any bus cycle that can be retried (see Figure 7-31).

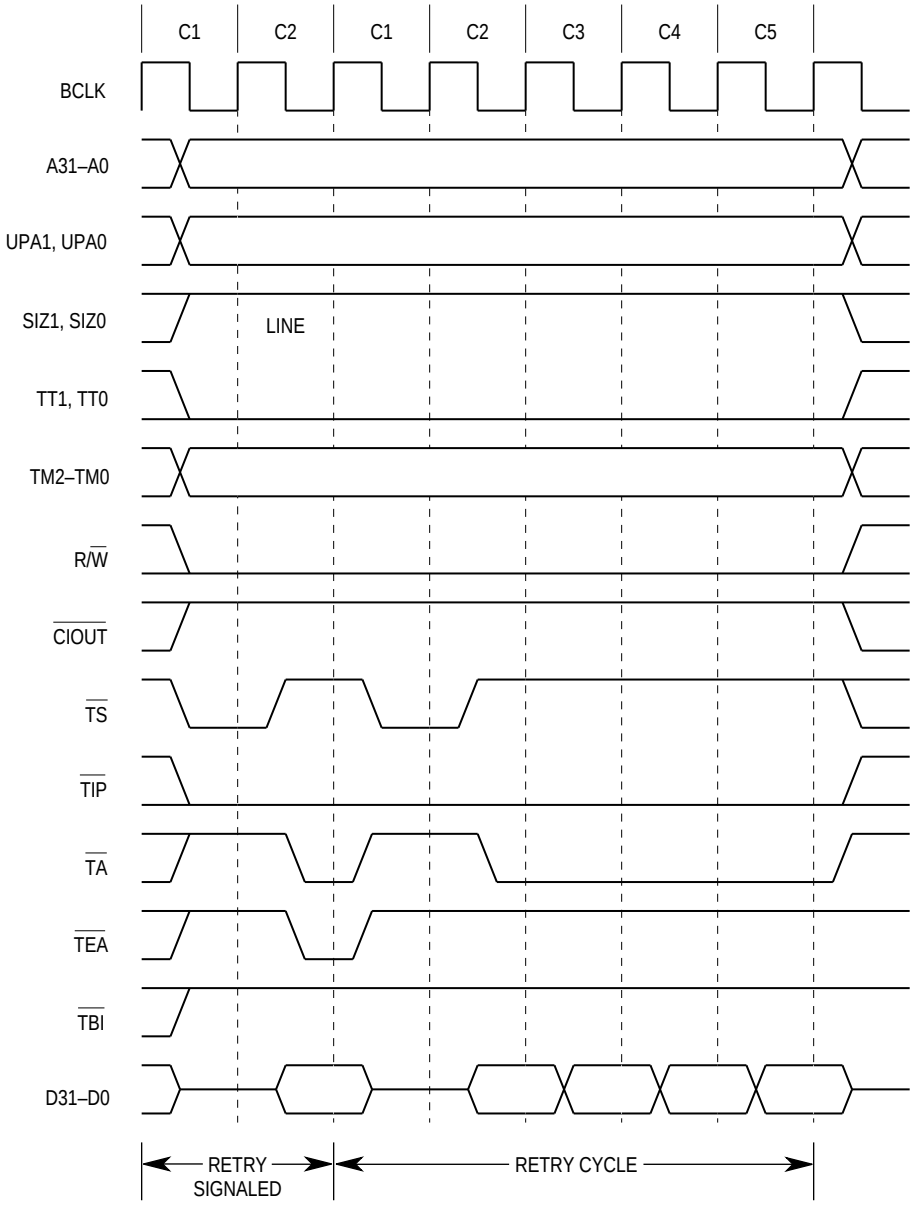
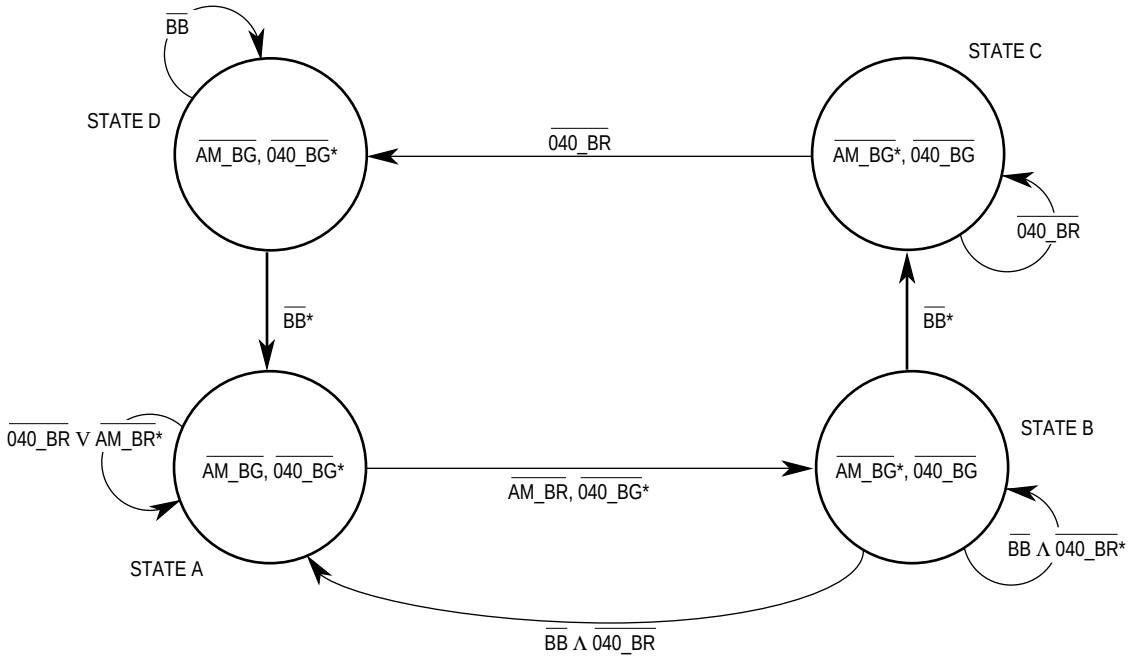
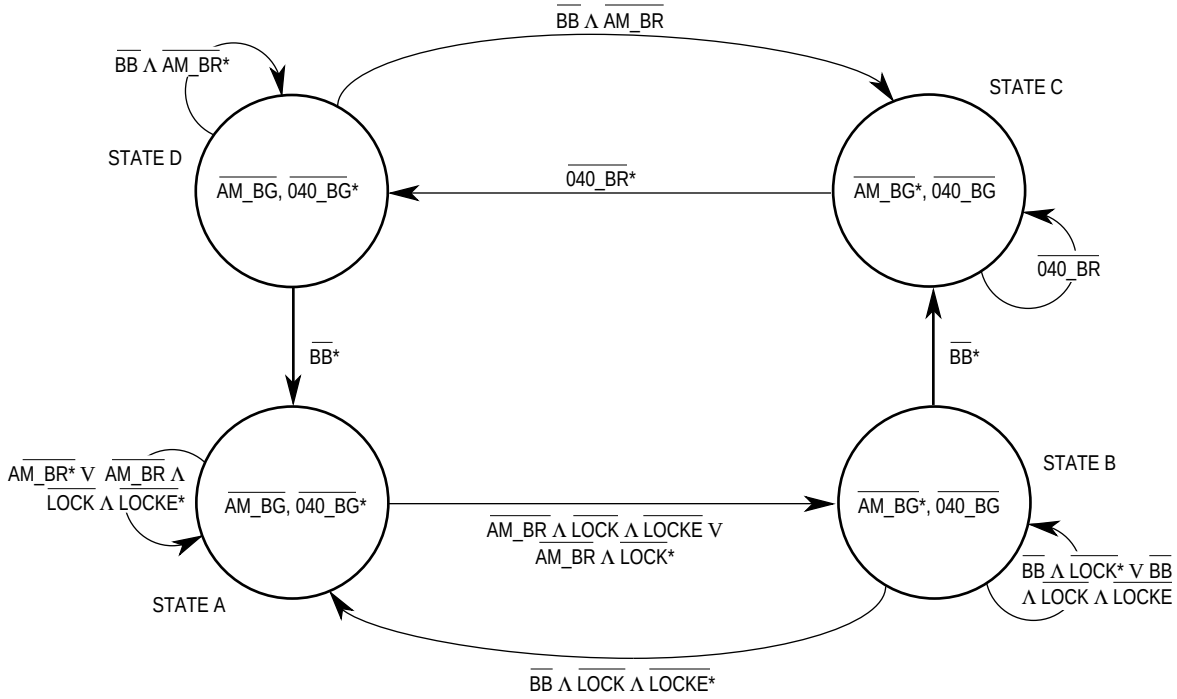


Figure 7-29. Retry Operation on Line Write



(a) MC68040 High Priorty, Default Bus Master



* Indicates the signal is asserted for that device.

(b) MC68040 Low-Priorty, Default Bus Master

Figure 7-37. M68040 Synchronous DMA Arbitration

does not attempt to alter the current state of memory. Only an external reset can restart a processor halted by a double bus fault.

The supervisor stack has special requirements to ensure that exceptions can be stacked. The stack must be resident with correct protection in the direction of growth to ensure that exception stacking never has a bus error or internal access fault. Memory pages allocated to the stack that are higher in memory than the current stack pointer can be nonresident since an RTE or FRESTORE instruction can check for residency and trap before restoring the state.

A special case exists for systems that allow arbitration of the processor bus during locked transfer sequences. If the arbiter can signal a bus error of a locked translation table update due to an improperly broken lock, any pages touched by exception stack operations must have the U-bit set in the corresponding page descriptor to prevent the occurrence of the locked access during translation table searches.

8.2.2 Address Error Exception

An address error exception occurs when the processor attempts to prefetch an instruction from an odd address. This includes the case of a conditional branch instruction with an odd branch offset that is not taken. A prefetch bus cycle is not executed, and the processor begins exception processing after the currently executing instructions have completed. If the completion of these instructions generates another exception, the address error exception is deferred, and the new exception is serviced. After exception processing for the address error exception commences, the sequence is the same as an access fault exception, except that the vector number is 3 and the vector offset in the stack frame refers to the address error vector. The stack frame is generated containing the address of the instruction that caused the address error and the address itself (A0 is cleared). If an address error occurs during the exception processing for a bus error, address error, or reset, a double bus fault occurs.

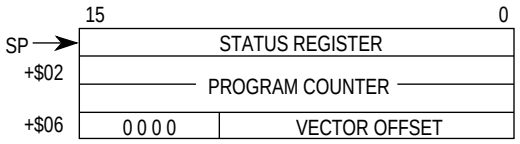
8.2.3 Instruction Trap Exception

Certain instructions are used to explicitly cause trap exceptions. The TRAP#n instruction always forces an exception and is useful for implementing system calls in user programs. The TRAPcc, FTRAPcc, TRAPV, CHK, and CHK2 instructions force exceptions if the user program detects an error, which can be an arithmetic overflow or a subscript value that is out of bounds. The DIVS and DIVU instructions force exceptions if a division operation is attempted with a divisor of zero.

As illustrated in Figure 8-1, when a trap exception occurs, the processor internally copies the SR, enters the supervisor mode, and clears T1 and T0. The processor generates a vector number according to the instruction being executed. Vector 5 is for DIVx, vector 6 is for CHK and CHK2, and vector 7 is for FTRAPcc, TRAPcc, and TRAPV instructions. For the TRAP#n instruction, the vector number is 32 plus n. The stack frame saves the trap vector offset, the PC, and the internal copy of the SR on the supervisor stack. The saved value of the PC is the logical address of the instruction following the instruction that caused the trap. For all instruction traps other than TRAP#n, a pointer to the instruction

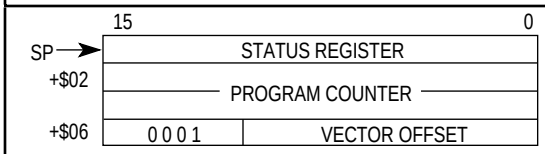
8.4.1 Four-Word Stack Frame (Format $0)

If a four-word stack frame is on the active stack and an RTE instruction is encountered, the processor updates the SR and PC with the data read from the stack, increments the stack pointer by eight, and resumes normal instruction execution.

Stack Frames	Exception Types	Stacked PC Points To
 FOUR-WORD STACK FRAME–FORMAT $0	• Interrupt • Format Error • TRAP #N • Illegal Instruction • A-Line Instruction • F-Line Instruction • Privilege Violation • Floating-Point Pre-Instruction	• Next Instruction • RTE or RESTORE Instruction • Next Instruction • Illegal Instruction • A-Line Instruction • F-Line Instruction • First Word of Instruction Causing Privilege Violation • Floating-Point Pre-Instruction Exception

8.4.2 Four-Word Throwaway Stack Frame (Format $1)

If a four-word throwaway stack frame is on the active stack and an RTE instruction is encountered, the processor increments the active stack pointer by eight, updates the SR with the value read from the stack, and then begins RTE processing again, as illustrated in Figure 8-6. The processor reads a new format word from the stack frame on top of the active stack (which may or may not be the same stack used for the previous operation) and performs the proper operations corresponding to that format. In most cases, the throwaway frame is on the interrupt stack, and when the SR value is read from the stack, the S-bit and M-bit are set. In that case, there is a normal four-word frame on the master stack. However, the second frame can be any format (even another throwaway frame) and can reside on any of the three system stacks.

Stack Frames	Exception Types	Stacked PC Points To
 THROWAWAY FOUR-WORD STACK FRAME–FORMAT $1	• Created on interrupt stack during interrupt exception processing when transition from master state to interrupt state occurs.	• Next Instruction: same as on master stack.

CMDREG1B—This field contains the command word of the exceptional floating-point instruction for an E1 exception, which is an exception detected by the conversion unit (CU) in the floating-point pipeline (see Figure 9-1). For FSQRT, CMDREG1B [6–0] are mapped from $4 for the instruction to $5 in CMDREG1B. All other instructions map directly.

CMDREG3B—This field contains the encoded instruction command word for an E3 exception, which is an exception detected by the write-back unit (WB) in the floating-point pipeline (see Figure 9-1). Figure 9-11 details the bit mapping between CMDREG1B and CMDREG3B. For FSQRT, bits CMDREG1B [6–0] are changed from $4 for the instruction to $5 for CMDREG1B, and therefore map to $21 for CMDREG3B.

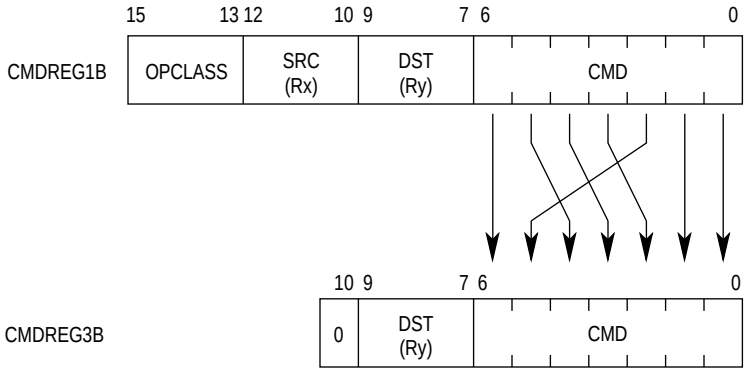


Figure 9-11. Mapping of Command Bits for CMDREG3B Field

CU_SAVEPC—This field contains the PC for the FPU pipeline's conversion unit.

E1—If set, this bit indicates that an exception has been detected by the conversion unit pipeline stage. All exception types are possible. The exception handler first checks for an E3 exception and processes it before checking and processing an E1 exception. The E1 exception is processed if the E1 bit is set. For the unimplemented instruction state frame, the source operand's unsupported data type is packed if the E1 bit is set.

E3—If set, this bit indicates that an exception has been detected by the WB pipeline stage. Only OVFL, UNFL, and INEX2 exceptions on opclass 010 or 000 (register to register and memory to register) for FADD, FSUB, FMUL, FDIV, FSQRT can occur. The exception handler must check for and process an E3 exception first.

ETS, ETE, ETM—Collectively, these fields are referred to as the ETEMP register and normally contain the source operand converted to extended precision. If the instruction specifies a packed decimal real source, bits 63–0 of the operand reside in ETM [63–00], and the ETS and ETE fields are undefined.

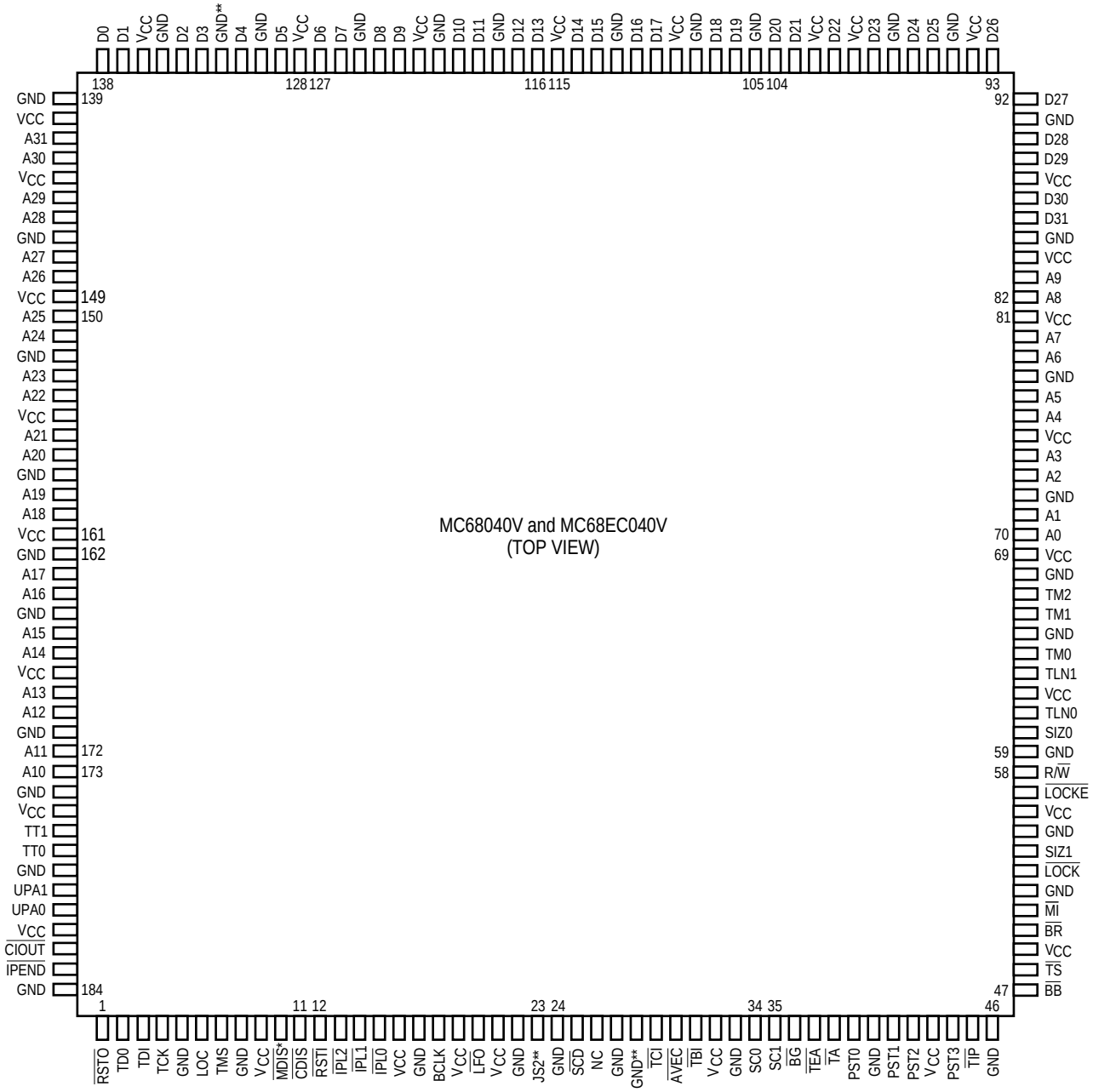
FPIARCU—This field contains the instruction address register for the FPU pipeline's conversion unit.

11.6 OUTPUT AC TIMING SPECIFICATIONS (See Figures 11-3 to 11-7)

Num	Characteristic	25 MHz				33 MHz				40 Mhz				Unit
		Large[1]		Small[2]		Large[1]		Small[2]		Large[1]		Small[2]		
		Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	
11[3]	BCLK to Address $\overline{\text{CIOUT}}$, $\overline{\text{LOCK}}$, $\overline{\text{LOCKE}}$, R/$\overline{\text{W}}$, SIZx, TLN, TMx, TTx, UPAx Valid	9	21	9	30	6.50	18	6.50	25	5.25	16	5.25	24	ns
12	BCLK to Output Invalid (Output Hold)	9	—	9	—	6.50	—	6.50	—	5.25	—	5.25	—	ns
13	BCLK to $\overline{\text{TS}}$ Valid	9	21	9	30	6.50	18	6.50	25	5.25	16	5.25	24	ns
14	BCLK to $\overline{\text{TIP}}$ Valid	9	21	9	30	6.50	18	6.50	25	5.25	17	5.25	24	ns
18[4]	BCLK to Data Out Valid	9	23	9	32	6.50	20	6.50	27	5.25	18	5.25	26	ns
19[4]	BCLK to Data Out Invalid (Output Hold)	9	—	9	—	6.50	—	6.50	—	5.25	—	5.25	—	ns
20[3,4]	BCLK to Output Low Impedance	9	—	9	—	6.50	—	6.50	—	5.25	—	5.25	—	ns
21[5]	BCLK to Data-Out High Impedance	9	20	9	20	6.50	17	6.50	17	5.25	16	5.25	16	ns
26[3]	BCLK to Multiplexed Address Valid	19	31	19	40	14	26	14	33	13	25	13	32	ns
27[3,5]	BCLK to Multiplexed Address Driven	19	—	19	—	14	—	14	—	13	—	13	—	ns
28[3,4,5]	BCLK to Multiplexed Address High Impedance	9	18	9	18	6.50	15	6.50	15	5.25	14	5.25	14	ns
29[4,5]	BCLK to Multiplexed Data Driven	19	—	19	—	14	20	14	20	13	19	13	19	ns
30[4]	BCLK to Multiplexed Data Valid	19	33	19	42	14	28	14	35	13	27	13	34	ns
38[3]	BCLK to Address, $\overline{\text{CIOUT}}$, $\overline{\text{LOCK}}$, $\overline{\text{LOCKE}}$, R/$\overline{\text{W}}$, SIZx, $\overline{\text{TS}}$, TLNx, TMx, TTx, UPAx High Impedance	9	18	9	18	6.50	15	6.50	15	5.25	14	5.25	14	ns
39	BCLK to $\overline{\text{BB}}$, $\overline{\text{TA}}$, $\overline{\text{TIP}}$ High Impedance	19	28	19	28	14	23	14	23	11.5	22	11.5	22	ns
40	BCLK to $\overline{\text{BR}}$, $\overline{\text{BB}}$ Valid	9	21	9	30	6.50	18	6.50	25	5.25	16	5.25	24	ns
43	BCLK to $\overline{\text{MI}}$ Valid	9	21	9	30	6.50	18	6.50	25	5.25	17	5.25	24	ns
48	BCLK to $\overline{\text{TA}}$ Valid	9	21	9	30	6.50	18	6.50	25	5.25	17	5.25	24	ns
50	BCLK to $\overline{\text{IPEND}}$, PSTx, $\overline{\text{RSTO}}$ Valid	9	21	9	30	6.50	18	6.50	25	5.25	17	5.25	24	ns

NOTES:
1. Output timing is specified for a valid signal measured at the pin. Large buffer timing is specified driving a 50 Ω transmission line with a length characterized by a 2.5-ns one-way propagation delay, terminated through 50 Ω to 2.5 V. Large buffer output impedance is 4–12 Ω, resulting in incident wave switching for this environment. All large buffer outputs must be terminated to guarantee operation.
2. Small buffer timing is specified driving an unterminated 30 Ω transmission line with a length characterized by a 2.5 ns one-way propagation delay. Small buffer output impedance is typically 30 Ω; the small buffer specifications include approximately 5 ns for the signal to propagate the length of the transmission line and back.
3. Timing specifications 11, 20, and 38 for address bus output timing apply when normal bus operation is selected. Specifications 26, 27, and 28 should be used when the multiplexed bus mode of operation is enabled.
4. Timing specifications 18 and 19 for data bus output timing apply when normal bus operation is selected. Specifications 28 and 29 should be used when the multiplexed bus mode of operation is enabled.
5. Timing specifications 21, 27, 28, and 29 are measured from BCLK edges. By design, the MC68040 cannot drive address and data simultaneously during multiplexed operations.

12.2.7 MC68040V and MC68EC040V Quad Flat Pack



NOTES:
* On MC68EC040V this pin is called JS1.
** All these pins are in the JTAG scan chain. On an MC68040 design JS2 = GND; on an MC68060 design JS2 = CLK.

A.6.6 A.6.7 Output AC Timing Specifications (see Figures A-5* to A-9)

Num	Characteristic	20 MHz		25 MHz		33 MHz		40 MHz		Unit
		Min	Max	Min	Max	Min	Max	Min	Max	
11	BCLK to Address, $\overline{\text{CIOUT}}$, $\overline{\text{LOCK}}$, $\overline{\text{LOCKE}}$, PSTx, R/$\overline{\text{W}}$, SIZx, TLNx,TMx, TTx, UPAx Valid	11.5	35	9	30	6.5	25	5.25	24	nS
12	BCLK to Output Invalid (Output Hold)	11.5	—	9	—	6.5	—	5.25	—	nS
13	BCLK to $\overline{\text{TS}}$ Valid	11.5	35	9	30	6.5	25	5.25	24	nS
14	BCLK to $\overline{\text{TIP}}$ Valid	11.5	35	9	30	6.5	25	5.25	24	nS
18	BCLK to Data-Out Valid	11.5	37	9	32	6.5	27	5.25	26	nS
19	BCLK to Data-Out Invalid (Output Hold)	11.5	—	9	—	6.5	—	5.25	—	nS
20	BCLK to Output Low Impedance	11.5	—	9	—	6.5	—	5.25	—	nS
21	BCLK to Data-Out High Impedance	11.5	25	9	20	6.5	17	5.25	16	nS
38	BCLK to Address, $\overline{\text{CIOUT}}$, $\overline{\text{LOCK}}$, $\overline{\text{LOCKE}}$, R/$\overline{\text{W}}$, SIZx, $\overline{\text{TS}}$, TLNx, TMx, TTx, UPAx High Impedance	11.5	23	9	18	6.5	15	5.25	14	nS
39	BCLK to $\overline{\text{BB}}$, $\overline{\text{TA}}$, $\overline{\text{TIP}}$ High Impedance	23	33	19	28	14	25	11.5	22	nS
40	BCLK to $\overline{\text{BR}}$, $\overline{\text{BB}}$ Valid	11.5	35	9	30	6.5	23	5.25	14	nS
43	BCLK to $\overline{\text{MI}}$ Valid	11.5	35	9	30	6.5	25	5.25	24	nS
48	BCLK to $\overline{\text{TA}}$ Valid	11.5	35	9	30	6.5	25	5.25	24	nS
50	BCLK to $\overline{\text{IPEND}}$, PSTx, $\overline{\text{RSTO}}$ Valid	11.5	35	9	30	6.5	25	5.25	24	nS

*Output timing is specified for a valid signal measured at the pin. Timing is specified driving an unterminated 30-Ω transmission line with a length characterized by a 2.5-nS one-way propagation delay. Buffer output impedance is typically 30 Ω; the buffer specifications include approximately 5 nS for the signal to propagate the length of the transmission line and back.

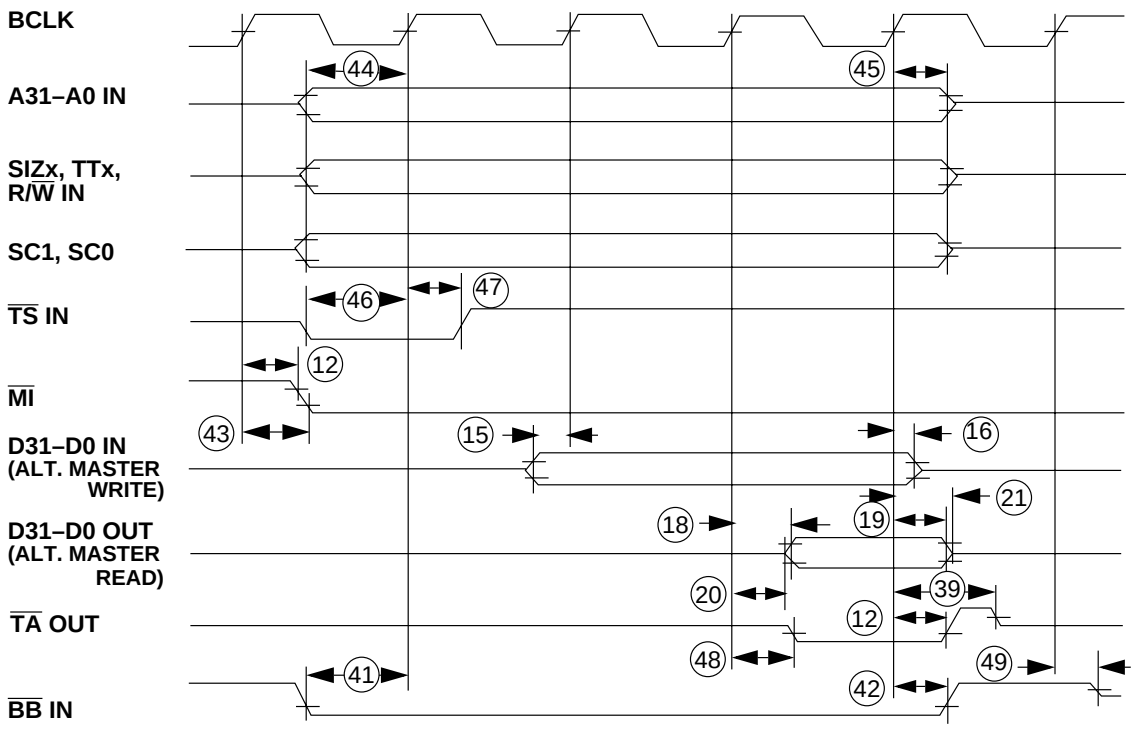
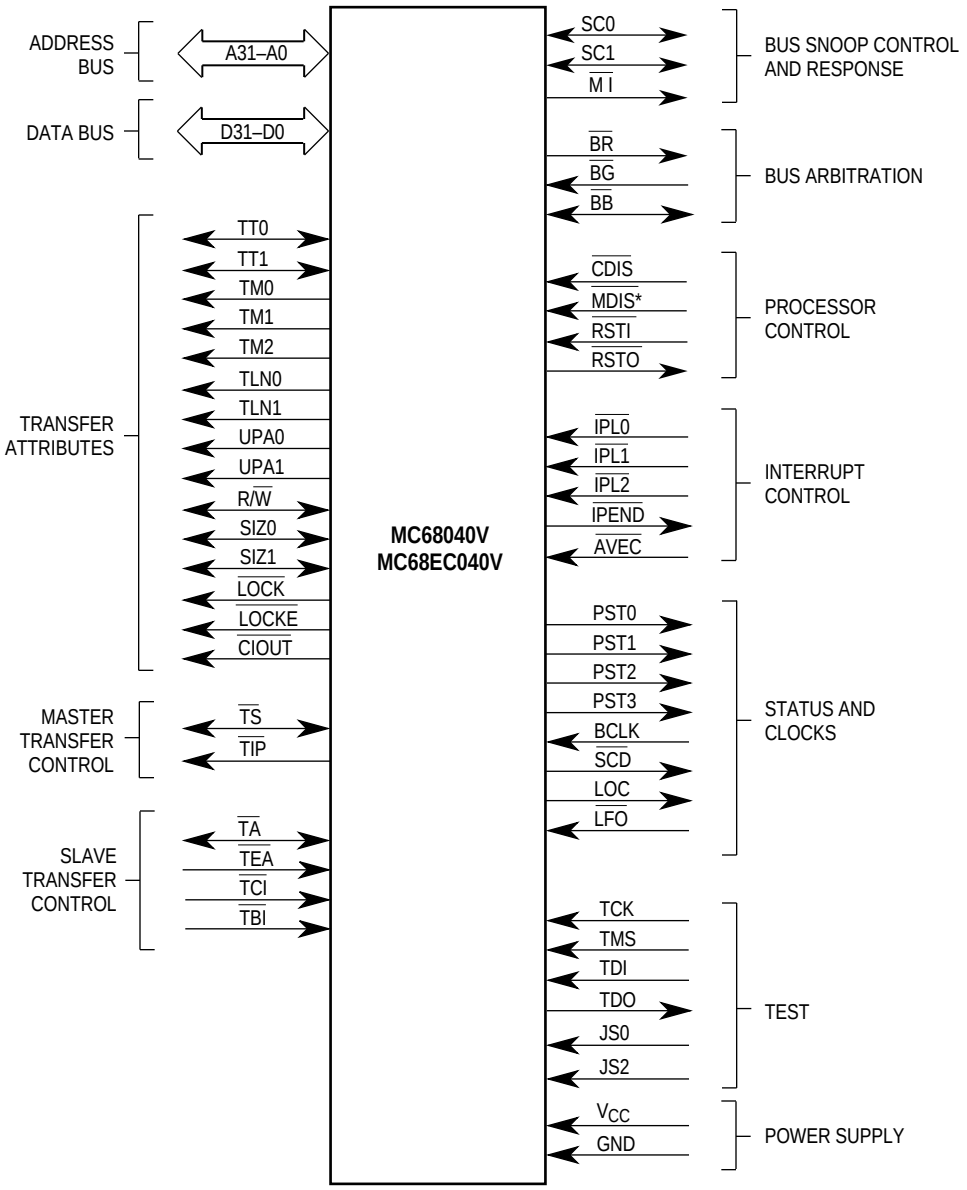


Figure B-10. Snoop Hit Timing



NOTE: *This signal is JS1 on the MC68EC040V.

Figure C-1. MC68040V and MC68EC040V Functional Signal Groups

C.2 LOW-POWER STOP MODE

The low-power stop mode is a reduced power mode of operation, that causes the MC68040V and MC68EC040V to remain quiescent until either a reset or non-masked interrupt occurs. This mode of operation has four phases of operation and is triggered by the low-power stop (LPSTOP) instruction:

1. Perform a LPSTOP broadcast cycle.
2. End integer unit (IU) instruction pipeline sequencing, which is similar to the STOP instruction sequence (IMM data ➧ SR), at termination of the LPSTOP broadcast cycle.

C.6 MC68040V AND MC68EC040V JTAG (PRELIMINARY)

The MC68040V and MC68EC040V include dedicated user-accessible test logic that is fully compatible with the IEEE standard 1149.1A *Standard Test Access Port and Boundary Scan Architecture*. Problems associated with testing high-density circuit boards have led to the standard's development under the sponsorship of the IEEE Test Technology Committee and the Joint Test Action Group (JTAG).

The following paragraphs are to be used in conjunction with the supporting IEEE document and includes those chip-specific items that the IEEE standard requires to be defined and additional information specific to the MC68040V and MC68EC040V implementations. For example, the IEEE standard 1149.1A test access port (TAP) controller states are referenced in this section but are not described. For these details and application information regarding the standard, refer to the IEEE standard 1149.1A document.

The MC68040V and MC68EC040V implementations support circuit board test strategies based on the standard. The test logic utilizes static logic design and is system logic independent of the device. The MC68040V and MC68EC040V implementations provide capabilities to:

a. Perform boundary scan operations to test circuit board electrical continuity,
b. Bypass the MC68040V and MC68EC040V by reducing the shift register path to a single cell,
c. Sample the MC68040V and MC68EC040V system pins during operation and transparently shift out the result,
d. Disable the output drive to output-only pins during circuit board testing.

NOTE

The IEEE standard 1149.1A test logic cannot be considered completely benign to those planning not to use this capability. Certain precautions must be observed to ensure that this logic does not interfere with system operation. Refer to **C.6.4 Disabling The IEEE Standard 1149.1A Operation**.

Figure C-4 illustrates a block diagram of the MC68040V and MC68EC040V implementations of IEEE standard 1149.1A. The test logic includes a 16-state dedicated TAP controller. These 16 controller states are defined in detail in the IEEE standard 1149.1A, but only 8 are included in this section.

Test-Logic-Reset	Run-Test/Idle
Capture-IR	Capture-DR
Update-IR	Update-DR
Shift-IR	Shift-DR

Four dedicated signal pins provides access to the TAP controller:

TCK—A test clock input that synchronizes the test logic.

TMS—A test mode select input with an internal pullup resistor sampled on the rising edge of TCK to sequence the TAP controller.

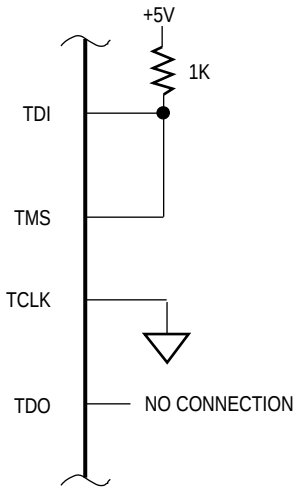


Figure C-10. Circuit Disabling IEEE Standard 1149.1A

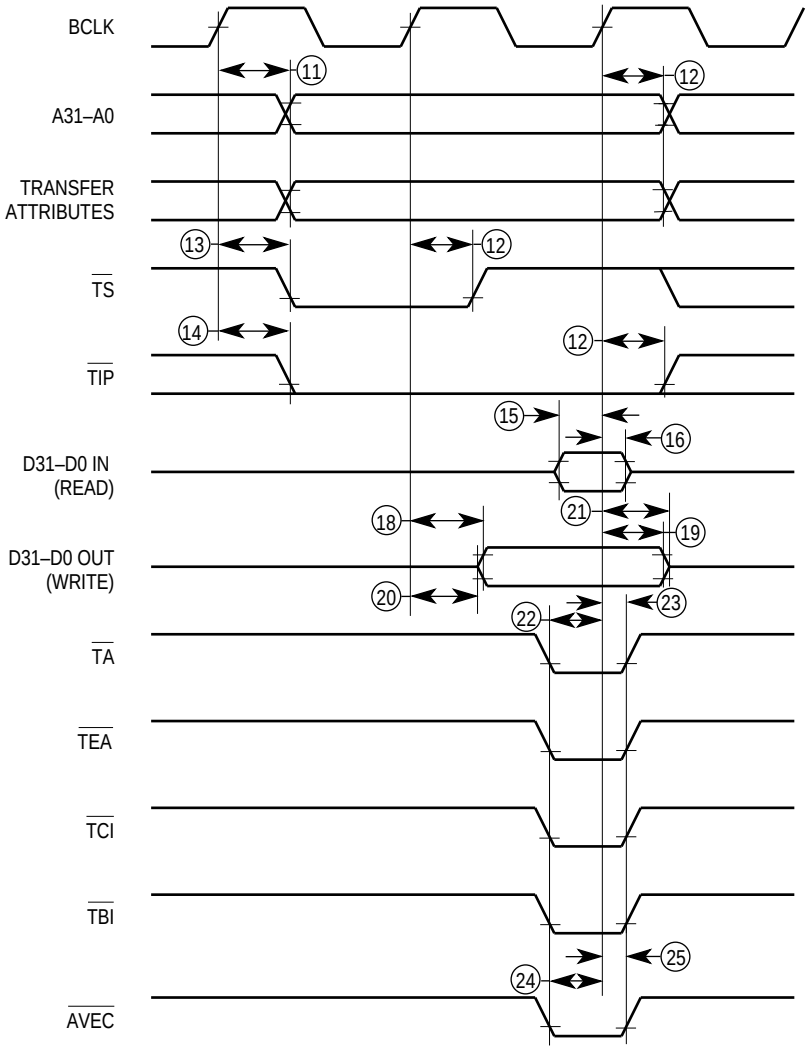
C.6.5 MC68040V and MC68EC040V JTAG Electrical Characteristics

The following paragraphs provide information on JTAG electrical and timing specifications This section is subject to change. For the most recent specifications, contact a Motorola sales office or complete the registration card at the beginning of this manual.

JTAG DC Electrical Specifications—PRELIMINARY

Characteristic	Symbol	Min	Max	Unit
Input High Voltage	V_{IH}	2	5.5	V
Input Low Voltage	V_{IL}	GND	0.8	V
Overshoot	—	—	TBD	V
TCK Input Leakage Current @ 0.5–2.4 V	I_{in}	TBD	TBD	µA
TDO Hi-Z (Off-State) Leakage Current @ 0.5–2.4 V	I_{TST}	TBD	TBD	µA
Signal Low Input Current, V_{IL} = 0.8 V TMS, TDI	I_L	TBD	TBD	mA
Signal High Input Current, V_{IH} = 2.0 V TMS, TDI	I_H	TBD	TBD	mA
TDO Output High Voltage I_{OH} = 5ma	V_{OH}	2.4	—	V
TDO Output Low Voltage I_{OL} = 5ma	V_{OL}	—	0.5	V
Capacitance*, V_{in} = 0 V, f = 1 MHz	C_{in}	—	TBD	pF

*Capacitance is periodically sampled rather than 100% tested.



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

Figure C-13. Read/Write Timing