Freescale Semiconductor - MC68340CAG25E Datasheet

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	CPU32
Number of Cores/Bus Width	1 Core, 32-Bit
Speed	25MHz
Co-Processors/DSP	-
RAM Controllers	DRAM
Graphics Acceleration	No
Display & Interface Controllers	-
Ethernet	-
SATA	-
USB	-
Voltage - I/O	5.0V
Operating Temperature	-40°C ~ 85°C (TA)
Security Features	-
Package / Case	144-LQFP
Supplier Device Package	144-LQFP (20x20)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mc68340cag25e

TABLE OF CONTENTS (Continued)

Paragraph Number — Title — Page Number

LIST OF TABLES (Continued)

SECTION 4
SYSTEM INTEGRATION MODULE

The MC68340 system integration module (SIM40) consists of several functions that control the system start-up, initialization, configuration, and the external bus with a minimum of external devices. It also provides the IEEE 1149.1 boundary scan capabilities. The SIM40 includes the following functions:

- System Configuration and Protection
- Clock Synthesizer
- Chip Selects and Wait States
- External Bus Interface
- Bus Arbitration
- Dynamic Bus Sizing
- IEEE 1149.1 Test Access Port

4.1 MODULE OVERVIEW

The SIM40 is essentially identical to the SIM implemented in the MC68330. The SIM40 has similar features to the SIM in the MC68331, MC68332, and MC68333. The periodic interrupt timer, double bus fault monitor, software watchdog, internal bus monitor, and spurious interrupt monitor are identical. However, many of the other features in the SIM's differ in their use and details.

The system configuration and protection function controls system configuration and provides various monitors and timers, including the internal bus monitor, double bus fault monitor, spurious interrupt monitor, software watchdog timer, and the periodic interrupt timer.

The clock synthesizer generates the clock signals used by the SIM40 and the other on-chip modules, as well as CLKOUT used by external devices.

The programmable chip select function provides four chip select signals that can enable external memory and peripheral circuits, providing all handshaking and timing signals. Each chip select signal has an associated base address register and an address mask register that contain the programmable characteristics of that chip select. Up to three wait states can be programmed by setting bits in the address mask register.

4.2.5.2 PORT B. Port B pins can be independently programmed to function as chip selects, IRQ≈ and MODCK pins, or discrete I/O pins. These pins are multiplexed as shown in Figure 4-7. Selection of a pin function is accomplished by a combination of the port B pin assignment register (PPARB) and the FIRQ bit of the MCR. See Table 4-5 for port B combinations. By changing the value of the FIRQ bit and the corresponding bits in the PPARB for a particular signal, the port B pins can be configured for different pin functions. Upon reset, port B is configured as MODCK, IRQ7, IRQ6, IRQ5, IRQ3, and CS3–CS0.

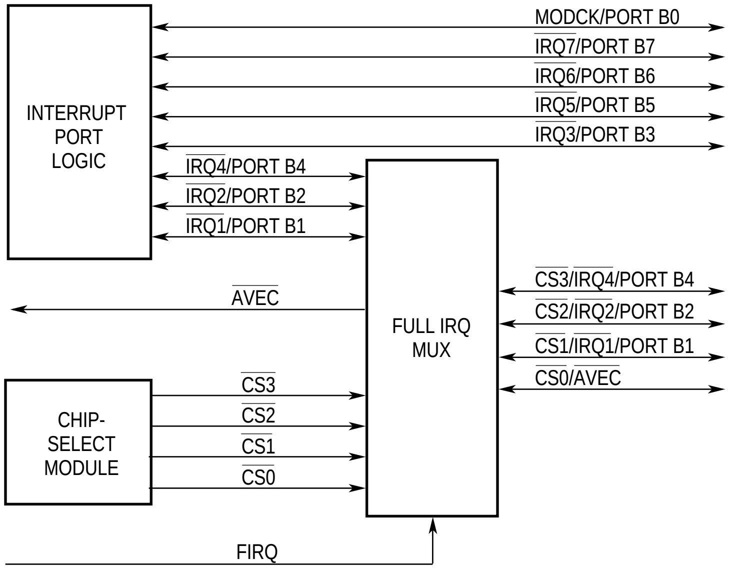


Figure 4-7. Full Interrupt Request Multiplexer

Table 4-5. Port B Pin Assignment Register

Signal	Pin Function			

	FIRQ = 0
	PPARB = 0
		FIRQ = 0
		PPARB = 1
			FIRQ = 1
			PPARB = 0
				FIRQ = 1
				PPARB = 1

IRQ7	PORTB7	IRQ7	PORTB7	IRQ7
IRQ6	PORTB6	IRQ6	PORTB6	IRQ6
IRQ5	PORTB5	IRQ5	PORTB5	IRQ5
IRQ3	PORTB3	IRQ3	PORTB3	IRQ3
CS3	CS3	CS3	PORTB4	IRQ4
CS2	CS2	CS2	PORTB2	IRQ2
CS1	CS1	CS1	PORTB1	IRQ1
CS0	CS0	CS0	AVEC	AVEC
MODCK	PORTB0	MODCK	PORTB0	MODCK

NOTE: MODCK has no function after reset.

PIV7–PIV0—Periodic Interrupt Vector Bits 7–0

These bits contain the value of the vector generated during an IACK cycle in response to an interrupt from the periodic timer. When the SIM40 responds to the IACK cycle, the periodic interrupt vector from the PICR is placed on the bus. This vector number is multiplied by four to form the vector offset, which is added to the vector base register to obtain the address of the vector.

4.3.2.7 PERIODIC INTERRUPT TIMER REGISTER (PITR). The PITR contains control for prescaling the software watchdog and periodic timer as well as the count value for the periodic timer. This register can be read or written at any time. Bits 15–10 are not implemented and always return zero when read. A write does not affect these bits.

PITR $024

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	SWP	PTP	PITR7	PITR6	PITR5	PITR4	PITR3	PITR2	PITR1	PITR0
RESET:															
0	0	0	0	0	0	MODCK	MODCK	0	0	0	0	0	0	0	0

Supervisor Only

Bits 15–10—Reserved

SWP—Software Watchdog Prescale

This bit controls the software watchdog clock source as shown in **4.3.2.5 System Protection Control Register (SYPCR)**.

1 = Software watchdog clock prescaled by a value of 512.
0 = Software watchdog clock not prescaled.

The SWP reset value is the inverse of the MODCK bit state on the rising edge of reset.

PTP—Periodic Timer Prescaler Control

This bit contains the prescaler control for the periodic timer.

1 = Periodic timer clock prescaled by a value of 512.
0 = Periodic timer clock not prescaled.

The PTP reset value is the inverse of the MODCK bit state on the rising edge of reset.

PITR7–PITR0—Periodic Interrupt Timer Register Bits 7–0

The remaining bits of the PITR contain the count value for the periodic timer. A zero value turns off the periodic timer.

4.3.2.8 SOFTWARE SERVICE REGISTER (SWSR). The SWSR is the location to which the software watchdog servicing sequence is written. The software watchdog can be enabled or disabled by the SWE bit in the SYPCR. SWSR can be written at any time, but returns all zeros when read.

SWSR $027

7	6	5	4	3	2	1	0
SWSR7	SWSR6	SWSR5	SWSR4	SWSR3	SWSR2	SWSR1	SWSR0
RESET:							
0	0	0	0	0	0	0	0

Supervisor Only

4.3.3 Clock Synthesizer Control Register (SYNCR)

The SYNCR can be read or written only in supervisor mode. The reset state of SYNCR produces an operating frequency of 8.39 MHz when the PLL is referenced to a 32.768-kHz reference signal. The system frequency is controlled by the frequency control bits in the upper byte of the SYNCR as follows:

$$F_{SYSTEM} = F_{CRYSTAL}\ [2^{(2+2W+X)}] \times (Y+1)$$

SYNCR $004

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
W	X	Y5	Y4	Y3	Y2	Y1	Y0	RSVD	0	0	SLIMP	SLOCK	RSTEN	STSIM	STEXT
RESET:															
0	0	1	1	1	1	1	1	0	0	0	U	U	0	0	0

U = Unaffected by reset

Supervisor Only

W—Frequency Control Bit

This bit controls the prescaler tap in the synthesizer feedback loop. Setting the bit increases the VCO speed by a factor of 4, requiring a time delay for the VCO to relock (see equation for determining system frequency).

X—Frequency Control Bit

This bit controls a divide-by-two prescaler, which is not in the synthesizer feedback loop. Setting the bit doubles the system clock speed without changing the VCO speed, as specified in the equation for determining system frequency; therefore, no delay is incurred to relock the VCO.

Y5–Y0—Frequency Control Bits

The Y-bits, with a value from 0–63, control the modulus downcounter in the synthesizer feedback loop, causing it to divide by the value of Y+1 (see the equation for determining system frequency). Changing these bits requires a time delay for the VCO to relock.

Bits 7–5—Reserved

Bit 7 is reserved for factory testing.

4.3.4.1 BASE ADDRESS REGISTERS. There are four 32-bit base address registers in the chip select function, one for each chip select signal.

Base Address 1 $044, $04C, $054, $05C

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BA31	BA30	BA29	BA28	BA27	BA26	BA25	BA24	BA23	BA22	BA21	BA20	BA19	BA18	BA17	BA16
RESET:	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U

Supervisor Only

Base Address 2 $046, $04E, $056, $05E

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BA15	BA14	BA13	BA12	BA11	BA10	BA9	BA8	BFC3	BFC2	BFC1	BFC0	WP	FTE	NCS	V
RESET:	U	U	U	U	U	U	U	U	U	U	U	U	U	U	0	0

U = Unaffected by reset Supervisor Only

BA31–BA8—Base Address Bits 31–8

The base address field, the upper 24 bits of each base address register, selects the starting address for the chip select. The specified base address must be on a multiple of the selected block size. The corresponding bits, AM31–AM8, in the address mask register define the size of the block for the chip select. The base address field (and the base function code field) is compared to the address on the address bus to determine if a chip select should be generated.

BFC3–BFC0—Base Function Code Bits 3–0

The value programmed into this field causes a chip select to be asserted for a certain address space type. There are nine function code address spaces (see **Section 3 Bus Operation**) specified as either user or supervisor, program or data, CPU, and DMA. These bits should be used to allow access to one type of address space. If access to more than one type of address space is desired, the FCMx bits should be used in addition to the BFCx bits. To prevent access to CPU space, set the NCS bit.

WP—Write Protect

This bit can restrict write accesses to the address range in a base address register. An attempt to write to the range of addresses specified in a base address register that has this bit set returns BERR.

1 = Only read accesses are allowed.
0 = Either read or write accesses are allowed.

FTE—Fast-Termination Enable

This bit causes the cycle to terminate early with an internal DSACK≈, giving a fast two-clock external access. When clear, all external cycles are at least three clocks. If fast termination is enabled, the DD bits of the corresponding address mask register are overridden (see **Section 3 Bus Operation**).

1 = Fast termination cycle enabled (termination determined by PS bits).
0 = Fast termination cycle disabled (termination determined by DD and PS bits).

5.2.2 Registers

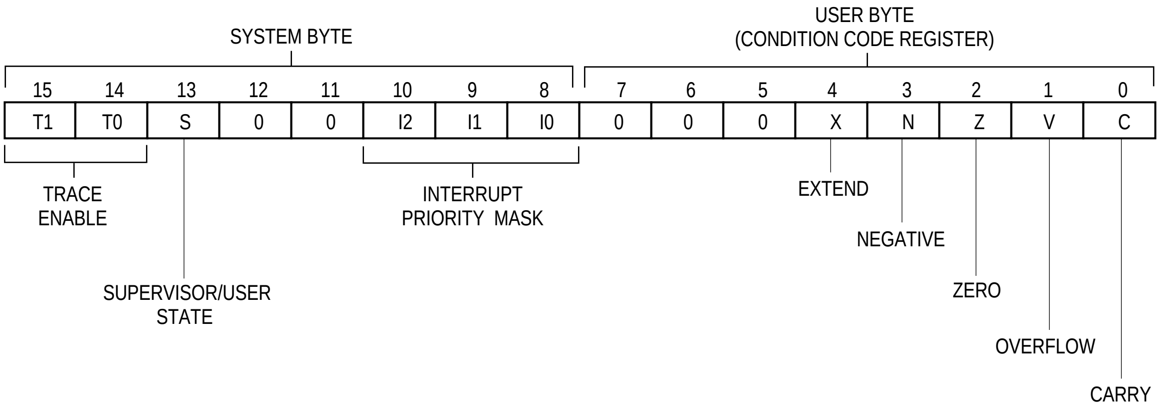
Registers D7–D0 are used as data registers for bit, byte (8-bit), word (16-bit), long-word (32-bit), and quad-word (64-bit) operations. Registers A6 to A0 and the USP and SSP are address registers that may be used as software SPs or base address registers. Register A7 (shown as A7 and A7' in Figures 5-3 and 5-4) is a register designation that applies to the USP in the user privilege level and to the SSP in the supervisor privilege level. In addition, address registers may be used for word and long-word operations. All of the 16 general-purpose registers (D7–D0, A7–A0) may be used as index registers.

The PC contains the address of the next instruction to be executed by the CPU32. During instruction execution and exception processing, the processor automatically increments the contents of the PC or places a new value in the PC, as appropriate.

The SR (see Figure 5-5) contains condition codes, an interrupt priority mask (three bits), and three control bits. Condition codes reflect the results of a previous operation. The codes are contained in the low byte (CCR) of the SR. The interrupt priority mask determines the level of priority an interrupt must have to be acknowledged. The control bits determine trace mode and privilege level. At user privilege level, only the CCR is available. At supervisor privilege level, software can access the full SR.

The VBR contains the base address of the exception vector table in memory. The displacement of an exception vector is added to the value in this register to access the vector table.

Alternate source and destination function code registers (SFC and DFC) contain 3-bit function codes. The CPU32 generates a function code each time it accesses an address. Specific codes are assigned to each type of access. The codes can be used to select eight dedicated 4-Gbyte address spaces. The MOVEC instruction can use registers SFC and DFC to specify the function code of a memory address.



15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
T1	T0	S	0	0	I2	I1	I0	0	0	0	X	N	Z	V	C

Figure 5-5. Status Register

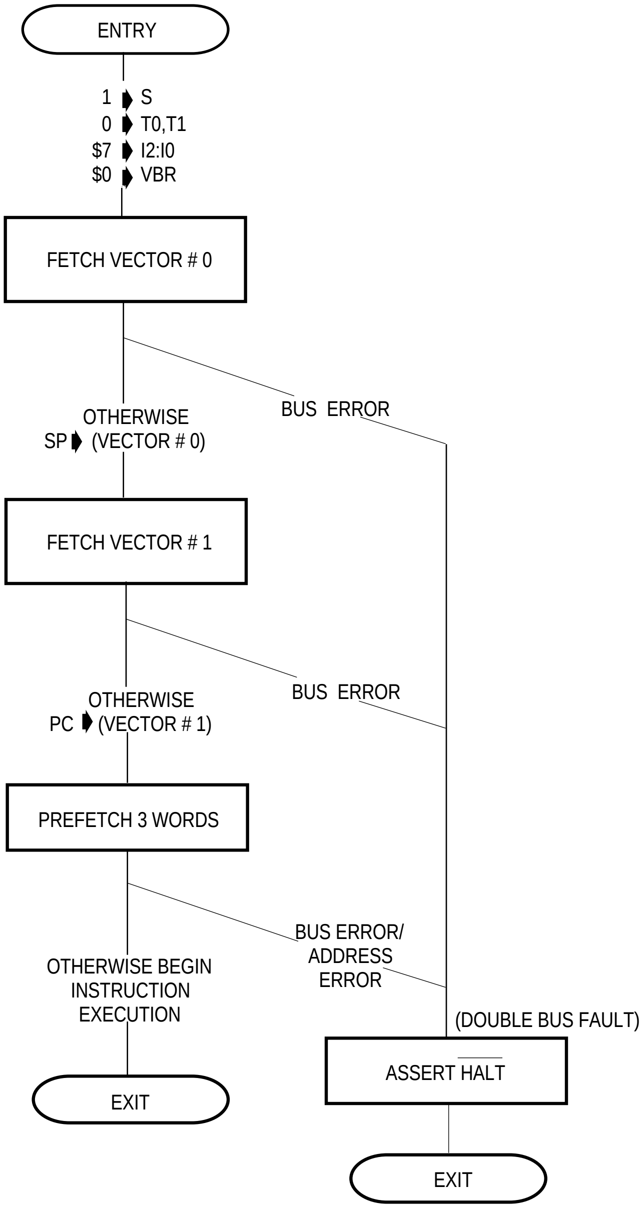


Figure 5-11. Reset Operation Flowchart

	15				0
SP ⇒	STATUS REGISTER				
+$02	RETURN PROGRAM COUNTER HIGH				
	RETURN PROGRAM COUNTER LOW				
+$06	1	1	0	0	VECTOR OFFSET
+$08	FAULTED ADDRESS HIGH				
	FAULTED ADDRESS LOW				
+$0C	DBUF HIGH				
	DBUF LOW				
+$10	CURRENT INSTRUCTION PROGRAM COUNTER HIGH				
	CURRENT INSTRUCTION PROGRAM COUNTER LOW				
+$14	INTERNAL TRANSFER COUNT REGISTER				
+$16	0	0	SPECIAL STATUS WORD		

Figure 5-15. Format $C—BERR Stack for Prefetches and Operands

	15				0
SP ⇒	STATUS REGISTER				
+$02	RETURN PROGRAM COUNTER HIGH				
	RETURN PROGRAM COUNTER LOW				
+$06	1	1	0	0	VECTOR OFFSET
+$08	FAULTED ADDRESS HIGH				
	FAULTED ADDRESS LOW				
+$0C	DBUF HIGH				
	DBUF LOW				
+$10	CURRENT INSTRUCTION PROGRAM COUNTER HIGH				
	CURRENT INSTRUCTION PROGRAM COUNTER LOW				
+$14	INTERNAL TRANSFER COUNT REGISTER				
+$16	0	1	SPECIAL STATUS WORD		

Figure 5-16. Format $C—BERR Stack on MOVEM Operand

5.6.1.1 BACKGROUND DEBUG MODE (BDM) OVERVIEW. Microprocessor systems generally provide a debugger, implemented in software, for system analysis at the lowest level. The BDM on the CPU32 is unique because the debugger is implemented in CPU microcode.

BDM incorporates a full set of debug options—registers can be viewed and/or altered, memory can be read or written, and test features can be invoked.

A resident debugger simplifies implementation of an in-circuit emulator. In a common setup (see Figure 5-18), emulator hardware replaces the target system processor. A complex, expensive pod-and-cable interface provides a communication path between target system and emulator.

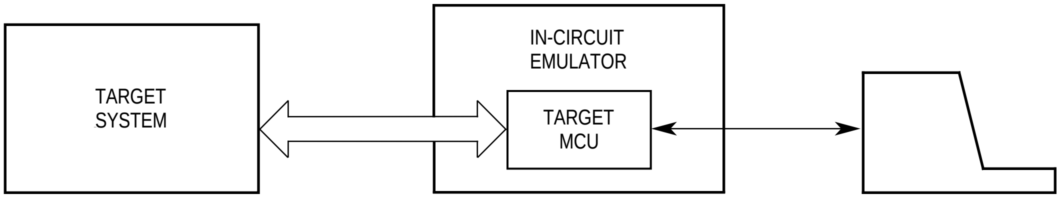


Figure 5-18. In-Circuit Emulator Configuration

By contrast, an integrated debugger supports use of a bus state analyzer (BSA) for in-circuit emulation. The processor remains in the target system (see Figure 5-19), and the interface is simplified. The BSA monitors target processor operation and the on-chip debugger controls the operating environment. Emulation is much closer to target hardware; thus, many interfacing problems (i.e., limitations on high-frequency operation, AC and DC parametric mismatches, and restrictions on cable length) are minimized.

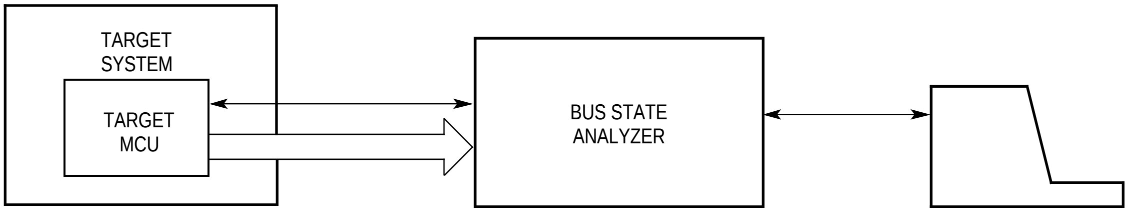
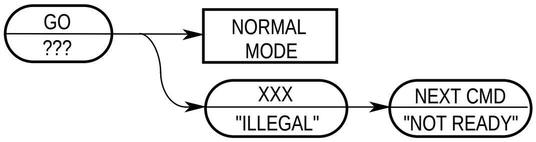


Figure 5-19. Bus State Analyzer Configuration

5.6.1.2 DETERMINISTIC OPCODE TRACKING OVERVIEW. CPU32 function code outputs are augmented by two supplementary signals that monitor the instruction pipeline. The IFETCH output signal identifies bus cycles in which data is loaded into the pipeline and signals pipeline flushes. The IPIPE output signal indicates when each mid-instruction pipeline advance occurs and when instruction execution begins. These signals allow a BSA to synchronize with instruction stream activity. Refer to **5.6.3 Deterministic Opcode Tracking** for complete information.

5.6.1.3 ON-CHIP HARDWARE BREAKPOINT OVERVIEW. An external breakpoint input and an on-chip hardware breakpoint capability permit breakpoint trap on any

Command Sequence:



Operand Data:

None

Result Data:

None

5.6.2.8.13 Call User Code (CALL). This instruction provides a convenient way to patch user code. The return PC is stacked at the location pointed to by the current SP. The stacked PC serves as a return address to be restored by the RTS command that terminates the patch routine. After stacking is complete, the 32-bit operand data is loaded into the PC. The pipeline is flushed and refilled from the location pointed to by the new PC, BDM is exited, and normal mode instruction execution begins.

NOTE

If a bus error or address error occurs during return address stacking, the CPU returns an error status via the serial interface and remains in BDM.

If a bus error or address error occurs on the first instruction prefetch from the new PC, the processor exits BDM and the error is trapped as a normal mode exception. The stacked value of the current PC may not be valid in this case, depending on the state of the machine prior to entering BDM. For address error, the PC does not reflect the true return PC. Instead, the stacked fault address is the (odd) return PC.

Command Format:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0

5.7.3.10 BIT MANIPULATION INSTRUCTIONS. The bit manipulation instruction table indicates the number of clock periods needed for the processor to perform the specified operation on the given addressing mode. The total number of clock cycles is outside the parentheses. The numbers inside parentheses (r/p/w) are included in the total clock cycle number. All timing data assumes two-clock reads and writes.

Instruction	Head	Tail	Cycles
BCHG #, Dn	2	0	6(0/2/0)∗
BCHG Dn, Dm	4	0	6(0/1/0)
BCHG #, 〈FEA〉	1	2	8(0/2/1)∗
BCHG Dn, 〈FEA〉	2	2	8(0/1/1)
BCLR #, Dn	2	0	6(0/2/0)∗
BCLR Dn, Dm	4	0	6(0/1/0)
BCLR #, 〈FEA〉	1	2	8(0/2/1)∗
BCLR Dn, 〈FEA〉	2	2	8(0/1/1)
BSET #, Dn	2	0	6(0/2/0)∗
BSET Dn, Dm	4	0	6(0/1/0)
BSET #, 〈FEA〉	1	2	8(0/2/1)∗
BSET Dn, 〈FEA〉	2	2	8(0/1/1)
BTST #, Dn	2	0	4(0/2/0)∗
BTST Dn, Dm	2	0	4(0/1/0)
BTST #, 〈FEA〉	1	0	4(0/2/0)∗
BTST Dn, 〈FEA〉	2	0	8(0/1/0)

∗ = An # fetch EA time must be added for this instruction: 〈FEA〉+〈FEA〉+〈OPER〉

Therefore, if a peripheral generates it asynchronously, it must be at least two clock periods long.

The DMA channel responds to cycle steal requests the same as all other requests. However, if subsequent DREQ≈ pulses are generated before DACK≈ is asserted in response to each request, they are ignored. If DREQ≈ is asserted after the DMA channel asserts DACK≈ for the previous request but before DACK≈ is negated, then the new request is serviced before bus ownership is released. If a new request is not generated by the time DACK≈ is negated, the bus is released.

6.3.2.3 EXTERNAL REQUEST WITH OTHER MODULES. The DMA controller can be externally connected to the serial module and used in conjunction with the serial module to send or receive data. The DMA takes the place of a separate service routine for accessing or storing data that is sent or received by the serial module. Using the DMA also lowers the CPU32 overhead required to handle the data transferred by the serial module. Figure 6-4 shows the external connections required for using the DMA with the serial module.

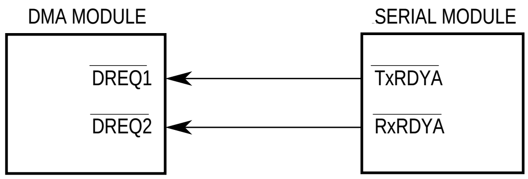


Figure 6-4. DMA External Connections to Serial Module

For serial receive, the DMA reads data from the serial receive buffer (RB) register (when the serial module has filled the buffer on input) and writes data to memory. For serial transmit, the DMA reads data from memory and writes data to the serial transmit buffer (TB) register. Only dual-address mode can be used with the serial module. The MC68340 on-chip peripherals do not support single-address transfers.

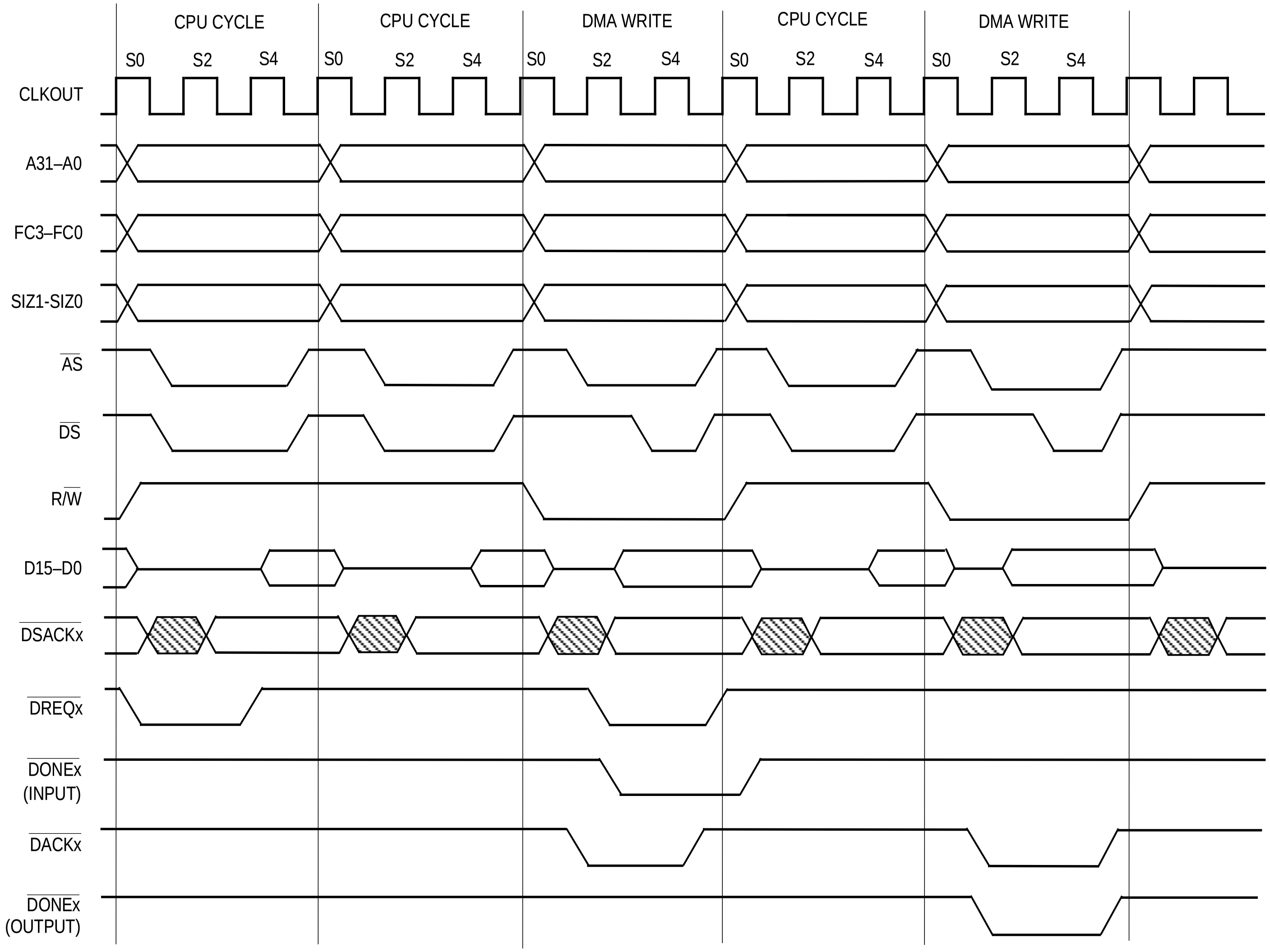
The timer modules can be used with the DMA in a similar manner. By connecting TOUTx to DREQ≈, the timer can request a DMA transfer.

6.4 DATA TRANSFER MODES

The DMA channel supports single- and dual-address transfers. The single-address transfer mode consists of one DMA bus cycle, which allows either a read or a write cycle to occur. The dual-address transfer mode consists of a source operand read and a destination operand write. Two DMA bus cycles are executed for the dual-address mode: a DMA read cycle and a DMA write cycle.

6.4.1 Single-Address Mode

The single-address DMA bus cycle allows data to be transferred directly between a device and memory without going through the DMA. In this mode, the operand transfer takes



NOTE:
1. DREQx must be active for two consecutive clocks for a DMA request to be recognized.
2. To cause another DMA transfer, DREQx is asserted after DACKx is asserted and before DACKx is negated.
3. DACKx and DONEx (DMA control signals) are asserted in the destination (write) DMA cycle.

Figure 6-8. Single-Address Write Timing (Cycle Steal)

7.2.9.1 RTSB. When used for this function, this signal can be programmed to be automatically negated and asserted by either the receiver or transmitter. When connected to the CTS≈ input of a transmitter, this signal can be used to control serial data flow.

7.2.9.2 OP1. When used for this function, this output is controlled by bit 1 in the OP.

7.2.10 Channel A Clear-To-Send (CTSA)

This active-low input is the channel A clear-to-send.

7.2.11 Channel B Clear-To-Send (CTSB)

This active-low input is the channel B clear-to-send.

7.2.12 Channel A Transmitter Ready (T≈RDYA)

This active-low output signal is programmable as the channel A transmitter ready or as a dedicated parallel output, and cannot be masked by the interrupt enable register (IER).

7.2.12.1 T≈RDYA. When used for this function, this signal reflects the complement of the status of bit 2 of the channel A status register (SRA). This signal can be used to control parallel data flow by acting as an interrupt to indicate when the transmitter contains a character.

7.2.12.2 OP6. When used for this function, this output is controlled by bit 6 in the OP.

7.2.13 Channel A Receiver Ready (R≈RDYA)

This active-low output signal is programmable as the channel A receiver ready, channel A FIFO full indicator, or a dedicated parallel output, and cannot be masked by the IER.

7.2.13.1 R≈RDYA. When used for this function, this signal reflects the complement of the status of bit 1 of the ISR. This signal can be used to control parallel data flow by acting as an interrupt to indicate when the receiver contains a character.

7.2.13.2 FFULLA. When used for this function, this signal reflects the complement of the status of bit 1 of the ISR. This signal can be used to control parallel data flow by acting as an interrupt to indicate when the receiver FIFO is full.

7.2.13.3 OP4. When used for this function, this output is controlled by bit 4 in the OP.

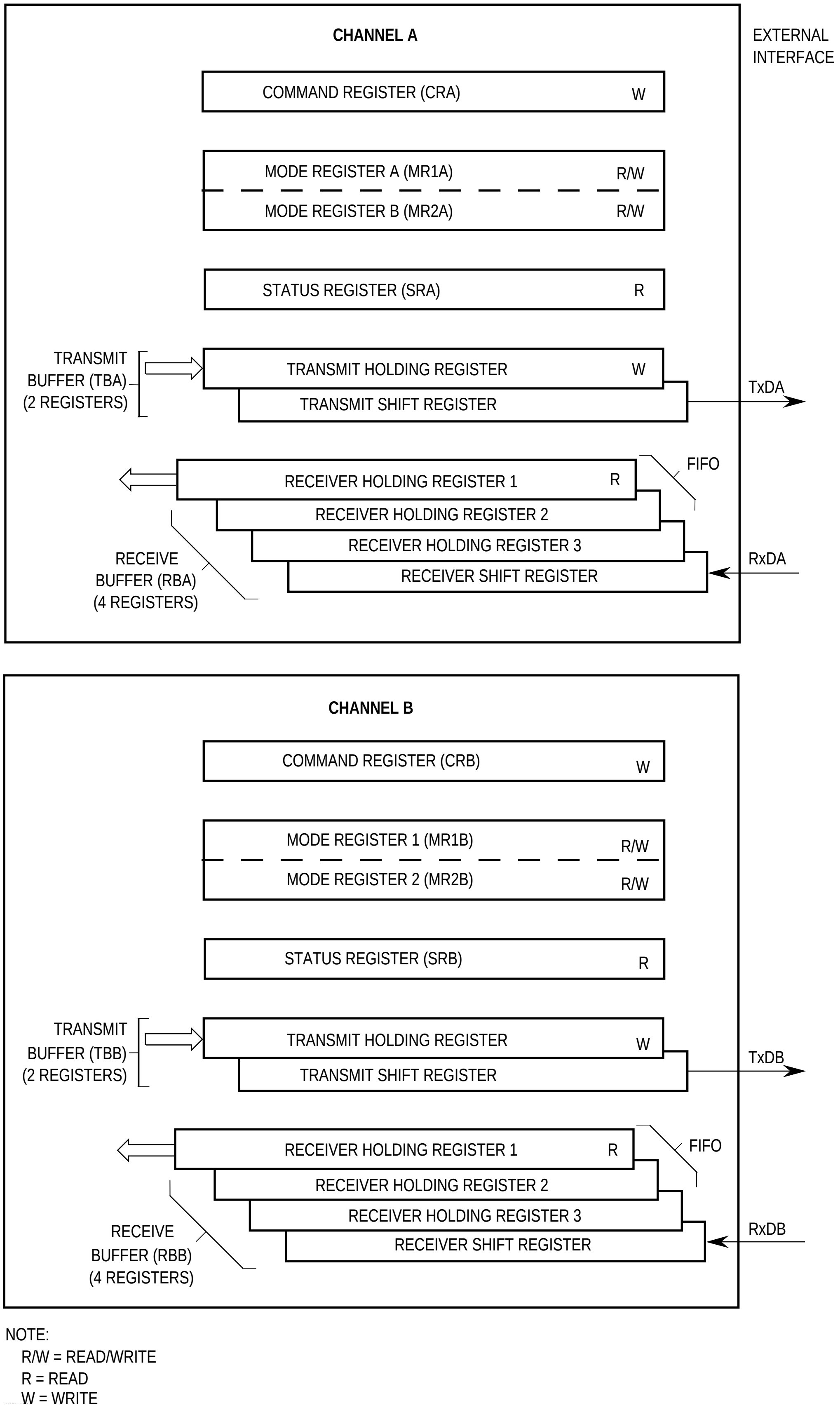

CHANNEL A
EXTERNAL INTERFACE
COMMAND REGISTER (CRA) W
MODE REGISTER A (MR1A) R/W
MODE REGISTER B (MR2A) R/W
STATUS REGISTER (SRA) R
TRANSMIT BUFFER (TBA) (2 REGISTERS)
TRANSMIT HOLDING REGISTER W
TRANSMIT SHIFT REGISTER
TxDA
FIFO
RECEIVER HOLDING REGISTER 1 R
RECEIVER HOLDING REGISTER 2
RECEIVER HOLDING REGISTER 3
RECEIVER SHIFT REGISTER
RxDA
RECEIVE BUFFER (RBA) (4 REGISTERS)
CHANNEL B
COMMAND REGISTER (CRB) W
MODE REGISTER 1 (MR1B) R/W
MODE REGISTER 2 (MR2B) R/W
STATUS REGISTER (SRB) R
TRANSMIT BUFFER (TBB) (2 REGISTERS)
TRANSMIT HOLDING REGISTER W
TRANSMIT SHIFT REGISTER
TxDB
FIFO
RECEIVER HOLDING REGISTER 1 R
RECEIVER HOLDING REGISTER 2
RECEIVER HOLDING REGISTER 3
RECEIVER SHIFT REGISTER
RxDB
RECEIVE BUFFER (RBB) (4 REGISTERS)
NOTE:
R/W = READ/WRITE
R = READ
W = WRITE

Figure 7-4. Transmitter and Receiver Functional Diagram

1 = This bit is set when the counter output equals the value in the COM.

0 = This bit is cleared when a timeout occurs, the COM register is accessed (read or write), the timer is reset with the SWR bit, or the RESET signal is asserted on the IMB. This bit is cleared regardless of the state of the TC bit.

This bit can be used to indicate when a write to the PREL1 or PREL2 registers will not cause a problem during a counter reload at timeout. To ensure that the write to the PREL register is recognized at timeout, the latency between the read of the COM bit and the write to the PREL register must be considered.

PO7–PO0—Prescaler Output

These bits show the levels on each of the eight output taps of the prescaler. These values are updated every time that the system clock goes high and a read cycle of this byte in the SR is not in progress.

8.4.5 Counter Register (CNTR)

The CNTR reflects the value of the counter. This value can be reliably read at any time since it is updated on every rising edge of the system clock (except in the input capture/output compare mode) when a read of the register is not in progress. This read-only register can be read when the timer module is enabled (i.e. the STP bit in the MCR is cleared).

CNTR $60A, $64A

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CNT15	CNT14	CNT13	CNT12	CNT11	CNT10	CNT9	CNT8	CNT7	CNT6	CNT5	CNT4	CNT3	CNT2	CNT1	CNT0
RESET:																
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Supervisor/User

All 24 bits of the prescaler and the counter may be obtained by one long-word read at the address of the SR, since the CNTR is contiguous to it. Any changes in the prescaler value due to the two cycles necessary to perform a long-word read should be considered. If this latency presents a problem, the TGATE≈ signal may be used to disable the decrement function while the reads are occurring.

8.4.6 Preload 1 Register (PREL1)

The PREL1 stores a value that is loaded into the counter in some modes of operation. This value is loaded into the counter on the first falling edge of the counter clock after the counter is enabled. This register can be be read and written when the timer module is enabled (i.e. the STP bit in the MCR is cleared). However, a write to this register must be completed before timeout for the new value to be reliably loaded into the counter.

11.8 DMA MODULE AC ELECTRICAL SPECIFICATIONS (See notes (a), (b), (c), and (d) corresponding to part operation, GND = 0 Vdc, TA = 0 to 70°C; see Figure 11-12)

		3.3 V		3.3 V or 5.0 V		5.0 V		
		8.39 MHz		16.78 MHz		25.16 MHz		
Num.	Characteristic	Min	Max	Min	Max	Min	Max	Unit
1	CLKOUT Low to AS, DACK, DONE Asserted	—	60	—	30	—	20	ns
2	CLKOUT Low to AS, DACK Negated	—	60	—	30	—	20	ns
3	DREQ≈ Asserted to AS Asserted (for DMA Bus Cycle)	$3t_{cyc} + t_{AIST} + t_{CLSA}$						ns
4[1]	Asynchronous Input Setup Time to CLKOUT Low	15	—	8, 5	—	5	—	ns
5	Asynchronous Input Hold Time from CLKOUT Low	30	—	15	—	10	—	ns
6	AS to DACK Assertion Skew	-30	30	–15	15	–10	10	ns
7	DACK to DONE Assertion Skew	-30	30	–15	15	–8	8	ns
8	AS, DACK, DONE Width Asserted	200	—	100	—	70	—	ns
8A	AS, DACK, DONE Width Asserted (Fast Termination Cycle)	80	—	40	—	28	—	ns

NOTES:
(a) The electrical specifications in this document for both the 8.39 and 16.78 MHz @ 3.3 V ±0.3 V are preliminary and apply only to the appropriate MC68340V low voltage part.
(b) The 16.78-MHz specifications apply to the MC68340 @ 5.0 V ±5% operation.
(c) The 25.16 MHz @ 5.0 V ±5% electrical specifications are preliminary.
(d) For extended temperature parts T_A = –40 to +85°C. These specifications are preliminary.
1. Specification #4 for 16.78 MHz @ 3.3 V ±0.3 V will be 8 ns.

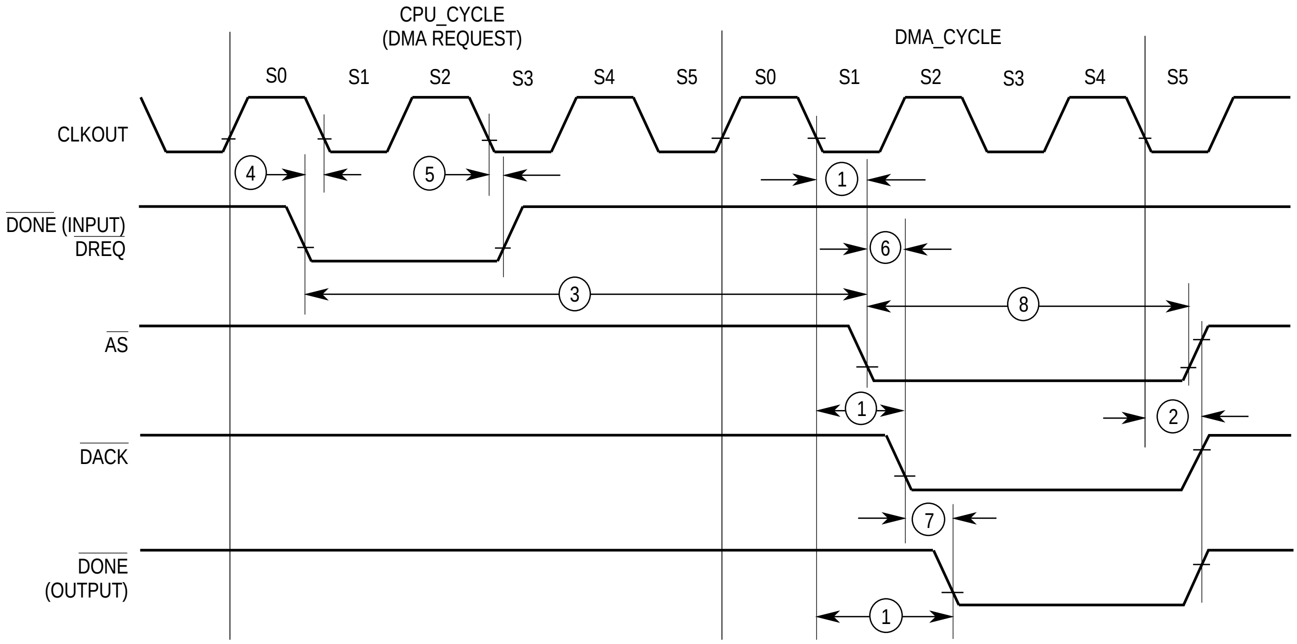


Figure 11-12. DMA Signal Timing Diagram