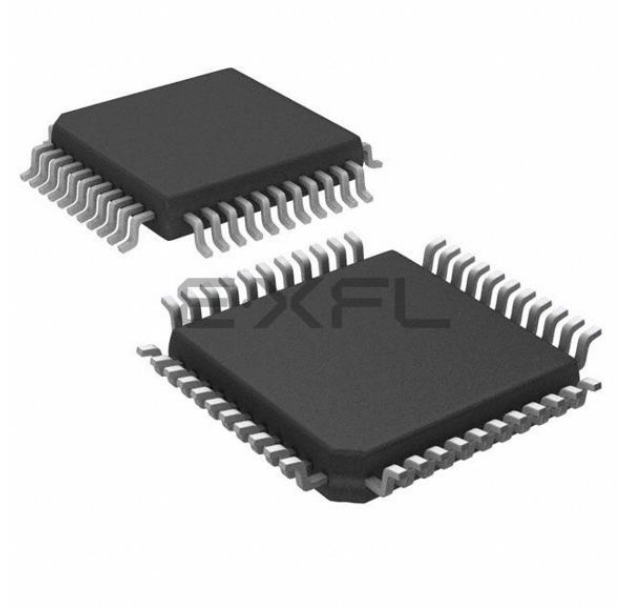


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	HC05
Core Size	8-Bit
Speed	2.1MHz
Connectivity	SCI, SPI
Peripherals	POR, WDT
Number of I/O	24
Program Memory Size	16KB (16K x 8)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	352 x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 5.5V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-QFP
Supplier Device Package	44-QFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc68hc705c9acfbe

MC68HC705C9A

Advance Information Data Sheet

To provide the most up-to-date information, the revision of our documents on the World Wide Web will be the most current. Your printed copy may be an earlier revision. To verify you have the latest information available, refer to:

http://www.freescale.com/

The following revision history table summarizes changes contained in this document. For your convenience, the page number designators have been linked to the appropriate location.

Revision History

Date	Revision Level	Description	Page Number(s)
October, 2001	3.0	Format update to current publication standards	N/A
		Figure 12-10. SPI Slave Timing Diagram — Corrected labels for MISO and MOSI and subtitle for part b.	145
February, 2002	4.0	Figure 8-3. Timer Status Register (TSR) — Corrected address designator from $0012 to $0013.	78
September, 2005	4.1	Updated to meet Freescale identity guidelines.	Throughout

- The port D data direction register ($0007) is disabled and the seven port D pins become input only.
- SPI output signals (MOSI, MISO, and SCK) do not require the data direction register control for output capability.
- The port D wire-OR mode control bit (bit 5 of SPCR $000A) is disabled, preventing open-drain configuration of port D.
- The $\overline{\text{RESET}}$ pin becomes input only.

1.4 Mask Options

The following two mask option registers are used to select features controlled by mask changes on the MC68HC05C9A and the MC68HC05C12A:

- Port B mask option register (PBMOR)
- C12 mask option register (C12MOR)

The mask option registers are EPROM locations which must be programmed prior to operation of the microcontroller.

1.4.1 Port B Mask Option Register (PBMOR)

The PBMOR register, shown in Figure 1-2, contains eight programmable bits which determine whether each port B bit (when in input mode) has the pullup and interrupt enabled. The port B interrupts share the vector and edge/edge-level sensitivity with the $\overline{\text{IRQ}}$ pin. For more details, (see 4.3 External Interrupt (IRQ or Port B)).

$3FF0	Bit 7	6	5	4	3	2	1	Bit 0
	PBPU7	PBPU6	PBPU5	PBPU4	PBPU3	PBPU2	PBPU1	PBPU0

Figure 1-2. Port B Mask Option Register

PBPU7–PBPU0 — Port B Pullup/Interrupt Enable Bits
1 = Pullup and CPU interrupt enabled
0 = Pullup and CPU interrupt disabled

NOTE

The current capability of the port B pullup devices is equivalent to the MC68HC05C9A, which is less than the MC68HC05C12A.

1.4.2 C12 Mask Option Register (C12MOR)

The C12MOR register, shown in Figure 1-3, controls the following options:

- Select between MC68HC05C9A/C12A configuration
- Enable/disable stop mode (C12A mode only)
- Enable/disable COP (C12A mode only)
- Edge-triggered only or edge- and level-triggered external interrupt pin (IRQ pin) (C12A mode only).

1.5 Software-Programmable Options (MC68HC05C9A Mode Only)

The C9A option register (OR), shown in Figure 1-4, is enabled only if configured in C9A mode. This register contains the programmable bits for the following options:

- Map two different areas of memory between RAM and EPROM, one of 48 bytes and one of 128 bytes
- Edge-triggered only or edge- and level-triggered external interrupt ($\overline{\text{IRQ}}$ pin and any port B pin configured for interrupt)

This register must be written to by user software during operation of the microcontroller.

$3FDF	Bit 7	6	5	4	3	2	1	Bit 0
Read: / Write:	RAM0	RAM1					IRQ	
Reset:	0	0	0	0	0	0	1	0

= Unimplemented

Figure 1-4. C9A Option Register

RAM0 — Random Access Memory Control Bit 0

This read/write bit selects between RAM or EPROM in location $0020 to $004F. This bit can be read or written at any time.

1 = RAM selected
0 = EPROM selected

RAM1— Random Access Memory Control Bit 1

This read/write bit selects between RAM or EPROM in location $0100 to $017F. This bit can be read or written at any time.

1 = RAM selected
0 = EPROM selected

IRQ — Interrupt Request Bit

This bit selects between an edge-triggered only or edge- and level- triggered external interrupt pin. This bit is set by reset, but can be cleared by software. This bit can be written only once.

1 = Edge and level interrupt option selected
0 = Edge-only interrupt option selected

Addr	Register Name
$0000	Port A Data Register
$0001	Port B Data Register
$0002	Port C Data Register
$0003	Port D Data Register
$0004	Port A Data Direction Register
$0005	Port B Data Direction Register
$0006	Port C Data Direction Register
$0007	Port D Data Direction Register (C9A Only)
$0008	Unused
$0009	Unused
$000A	Serial Peripheral Control Register
$000B	Serial Peripheral Status Register
$000C	Serial Peripheral Data Register
$000D	Baud Rate Register
$000E	Serial Communications Control Register 1
$000F	Serial Communications Control Register 2
$0010	Serial Communications Status Register
$0011	Serial Communications Data Register
$0012	Timer Control Register
$0013	Timer Status Register
$0014	Input Capture Register High
$0015	Input Capture Register Low
$0016	Output Compare Register High
$0017	Output Compare Register Low
$0018	Timer Register High
$0019	Timer Register Low
$001A	Alternate Timer Register High
$001B	Alternate Timer Register Low
$001C	EPROM Programming Register
$001D	C9A COP Reset Register
$001E	C9A COP Control Register
$001F	Reserved

Figure 2-3. I/O Register Summary

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$000D	SCI Baud Rate Register BAUD See page 69.	Read:			SCP1	SCP0		SCR2	SCR1	SCR0
		Write:								
		Reset:	—	—	0	0	—	U	U	U
$000E	SCI Control Register 1 (SCCR1) See page 65.	Read:	R8	T8		M	WAKE			
		Write:								
		Reset:	U	U	0	U	U	0	0	0
$000F	SCI Control Register 2 (SCCR2) See page 66.	Read:	TIE	TCIE	RIE	ILIE	TE	RE	RWU	SBK
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$0010	SCI Status Register (SCSR) See page 68.	Read:	TDRE	TC	RDRF	IDLE	OR	NF	FE	
		Write:								
		Reset:	1	1	0	0	0	0	0	—
$0011	SCI Data Register (SCDR) See page 65.	Read:	SCD7	SDC6	SCD5	SCD4	SCD3	SCD2	SCD1	SCD0
		Write:								
		Reset:	Unaffected by reset							
$0012	Timer Control Register (TCR) See page 53.	Read:	ICIE	OCIE	TOIE	0	0	0	IEDG	OLVL
		Write:								
		Reset:	0	0	0	0	0	0	U	0
$0013	Timer Status Register (TSR) See page 54.	Read:	ICF	OCF	TOF	0	0	0	0	0
		Write:								
		Reset:	U	U	U	0	0	0	0	0
$0014	Input Capture Register High (ICRH) See page 56.	Read:	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
		Write:								
		Reset:	Unaffected by reset							
$0015	Input Capture Register Low (ICRL) See page 56.	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	Unaffected by reset							
$0016	Output Compare Register High (OCRH) See page 56.	Read:	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
		Write:								
		Reset:	Unaffected by reset							
$0017	Output Compare Register Low (OCRL) See page 56.	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	Unaffected by reset							
$0018	Timer Register High (TRH) See page 55.	Read:	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
		Write:								
		Reset:	1	1	1	1	1	1	1	1

= Unimplemented R = Reserved U = Unaffected

Figure 2-4. Input/Output Registers (Sheet 2 of 3)

Chapter 4
Interrupts

4.1 Introduction

The MCU can be interrupted by five different sources, four maskable hardware interrupts, and one non-maskable software interrupt:

- External signal on the $\overline{\text{IRQ}}$ pin or port B pins
- 16-bit programmable timer
- Serial communications interface
- Serial peripheral interface
- Software interrupt instruction (SWI)

Interrupts cause the processor to save register contents on the stack and to set the interrupt mask (I bit) to prevent additional interrupts. The RTI instruction causes the register contents to be recovered from the stack and normal processing to resume.

Unlike reset, hardware interrupts do not cause the current instruction execution to be halted, but are considered pending until the current instruction is complete.

NOTE

The current instruction is the one already fetched and being operated on.

When the current instruction is complete, the processor checks all pending hardware interrupts. If interrupts are not masked (CCR I bit clear) and if the corresponding interrupt enable bit is set, the processor proceeds with interrupt processing; otherwise, the next instruction is fetched and executed.

If an external interrupt and a timer, SCI, or SPI interrupt are pending at the end of an instruction execution, the external interrupt is serviced first. The SWI is executed the same as any other instruction, regardless of the I-bit state.

Table 4-1 shows the relative priority of all the possible interrupt sources. Figure 4-1 shows the interrupt processing flow.

4.2 Non-Maskable Software Interrupt (SWI)

The SWI is an executable instruction and a non-maskable interrupt: It is executed regardless of the state of the I bit in the CCR. If the I bit is zero (interrupts enabled), SWI executes after interrupts which were pending when the SWI was fetched, but before interrupts generated after the SWI was fetched. The interrupt service routine address is specified by the contents of memory locations $3FFC and $3FFD.

5.5 MC68HC05C9A Compatible COP

This COP is controlled with two registers; one to reset the COP timer and the other to enable and control COP and clock monitor functions. Figure 5-3 shows a block diagram of the MC68HC05C9A COP.

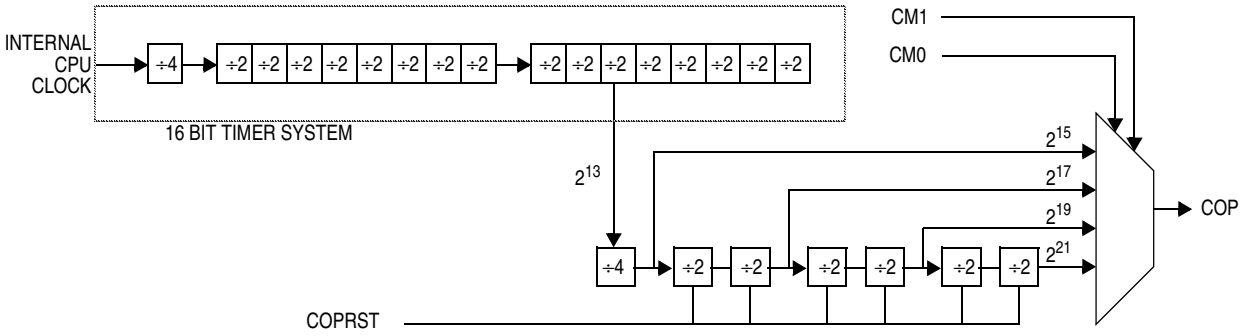


Figure 5-3. C9A COP Block Diagram

5.5.1 C9A COP Reset Register

This write-only register, shown in Figure 5-4, is used to reset the COP.

$001D	Bit 7	6	5	4	3	2	1	Bit 0
Read:								
Write:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 5-4. COP Reset Register (COPRST)

The sequence required to reset the COP timer is:

- Write $55 to the COP reset register
- Write $AA to the COP reset register

Both write operations must occur in the order listed, but any number of instructions may be executed between the two write operations provided that the COP does not time out between the two writes. The elapsed time between software resets must not be greater than the COP timeout period. If the COP should time out, a system reset will occur and the device will be re-initialized in the same fashion as a power-on reset or reset.

Reading this register does not return valid data.

5.5.2 C9A COP Control Register

The COP control register, shown in Figure 5-5, performs these functions:

- Enables clock monitor function
- Enables MC68HC05C9A compatible COP function
- Selects timeout duration of COP timer

and flags the following conditions:

- A COP timeout
- Clock monitor reset

Chapter 6
Low-Power Modes

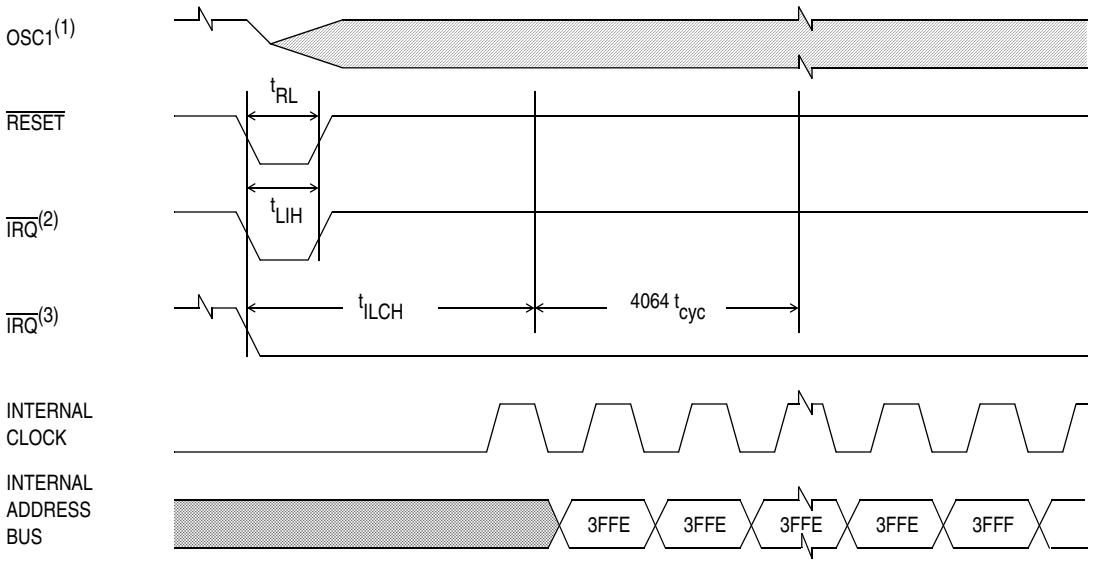
6.1 Introduction

This section describes the low-power modes.

6.2 Stop Mode

The STOP instruction places the MCU in its lowest-power consumption mode. In stop mode, the internal oscillator is turned off, halting all internal processing, including timer operation.

During the stop mode, the TCR bits are altered to remove any pending timer interrupt request and to disable any further timer interrupts. The timer prescaler is cleared. The I bit in the CCR is cleared to enable external interrupts. All other registers and memory remain unaltered. All input/output lines remain unchanged. The processor can be brought out of the stop mode only by an external interrupt or reset. See Figure 6-1.



Notes:
1. Represents the internal gating of the OSC1 pin
2. $\overline{\text{IRQ}}$ pin edge-sensitive mask option
3. $\overline{\text{IRQ}}$ pin level and edge-sensitive mask option

Figure 6-1. Stop Recovery Timing Diagram

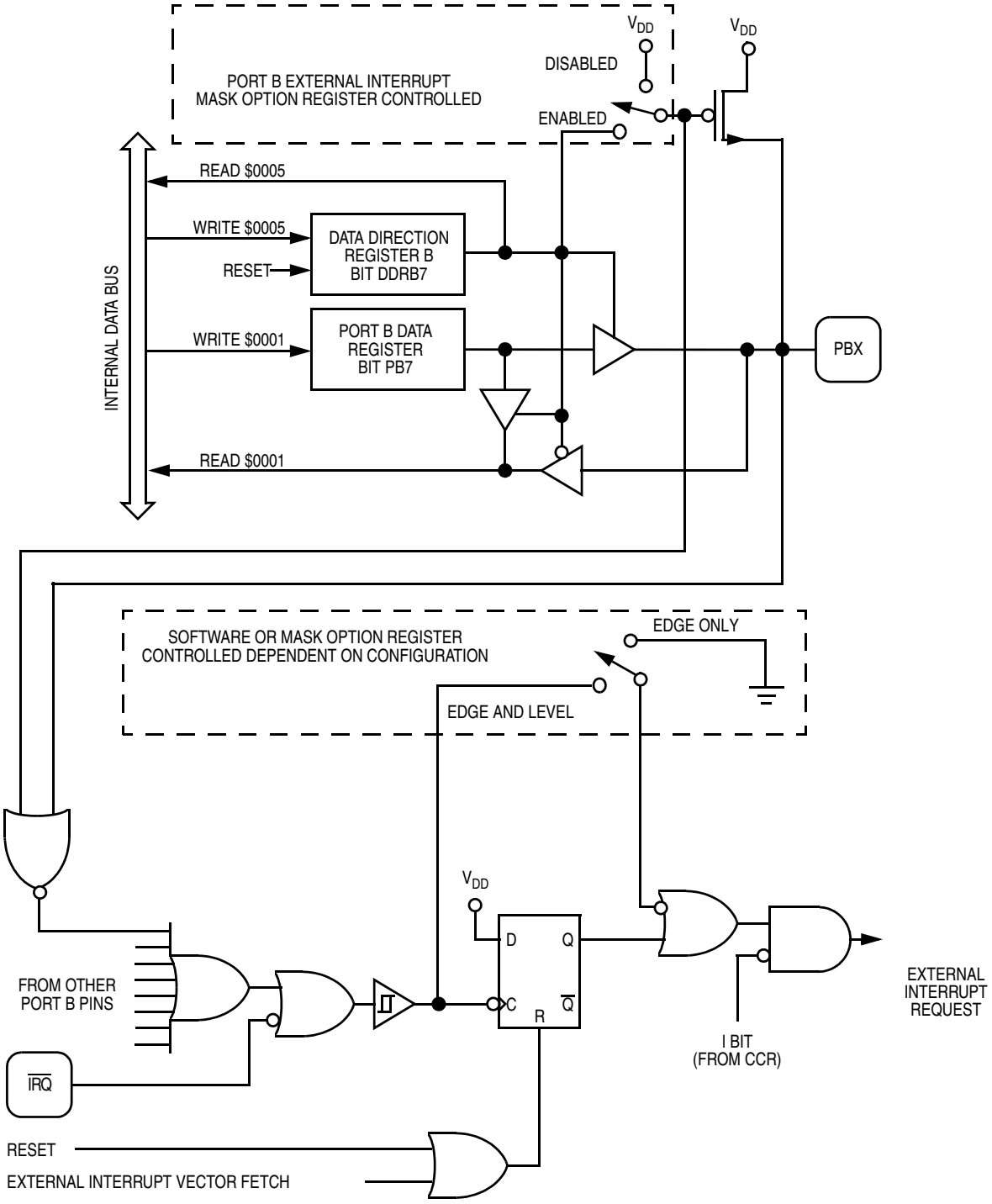


Figure 7-2. Port B I/O Logic

8.2 Timer Operation

The core of the capture/compare timer is a 16-bit free-running counter. The counter provides the timing reference for the input capture and output compare functions. The input capture and output compare functions provide a means to latch the times at which external events occur, to measure input waveforms, and to generate output waveforms and timing delays. Software can read the value in the 16-bit free-running counter at any time without affecting the counter sequence.

Because of the 16-bit timer architecture, the I/O registers for the input capture and output compare functions are pairs of 8-bit registers.

Because the counter is 16 bits long and preceded by a fixed divide-by-4 prescaler, the counter rolls over every 262,144 internal clock cycles. Timer resolution with a 4-MHz crystal is 2 μs.

8.2.1 Input Capture

The input capture function is a means to record the time at which an external event occurs. When the input capture circuitry detects an active edge on the TCAP pin, it latches the contents of the timer registers into the input capture registers. The polarity of the active edge is programmable.

Latching values into the input capture registers at successive edges of the same polarity measures the period of the input signal on the TCAP pin. Latching values into the input capture registers at successive edges of opposite polarity measures the pulse width of the signal.

8.2.2 Output Compare

The output compare function is a means of generating an output signal when the 16-bit counter reaches a selected value. Software writes the selected value into the output compare registers. On every fourth internal clock cycle the output compare circuitry compares the value of the counter to the value written in the output compare registers. When a match occurs, the timer transfers the programmable output level bit (OLVL) from the timer control register to the TCMP pin.

The programmer can use the output compare register to measure time periods, to generate timing delays, or to generate a pulse of specific duration or a pulse train of specific frequency and duty cycle on the TCMP pin.

8.3 Timer I/O Registers

The following I/O registers control and monitor timer operation:

- Timer control register (TCR)
- Timer status register (TSR)
- Timer registers (TRH and TRL)
- Alternate timer registers (ATRH and ATRL)
- Input capture registers (ICRH and ICRL)
- Output compare registers (OCRH and OCRL)

8.3.2 Timer Status Register

The timer status register (TSR), shown in Figure 8-3, contains flags to signal the following conditions:

- An active signal on the TCAP pin, transferring the contents of the timer registers to the input capture registers
- A match between the 16-bit counter and the output compare registers, transferring the OLVL bit to the TCMP pin
- A timer roll over from $FFFF to $0000

$0013	Bit 7	6	5	4	3	2	1	Bit 0
Read:	ICF	OCF	TOF	0	0	0	0	0
Write:								
Reset:	U	U	U	0	0	0	0	0

= Unimplemented U = Undetermined

Figure 8-3. Timer Status Register (TSR)

ICF — Input Capture Flag

The ICF bit is set automatically when an edge of the selected polarity occurs on the TCAP pin. Clear the ICF bit by reading the timer status register with ICF set and then reading the low byte ($0015) of the input capture registers. Resets have no effect on ICF.

OCF — Output Compare Flag

The OCF bit is set automatically when the value of the timer registers matches the contents of the output compare registers. Clear the OCF bit by reading the timer status register with OCF set and then reading the low byte ($0017) of the output compare registers. Resets have no effect on OCF.

TOF — Timer Overflow Flag

The TOF bit is set automatically when the 16-bit counter rolls over from $FFFF to $0000. Clear the TOF bit by reading the timer status register with TOF set, and then reading the low byte ($0019) of the timer registers. Resets have no effect on TOF.

9.4 SCI Transmitter Features

Features of the SCI transmitter include:

- Transmit data register empty flag
- Transmit complete flag
- Send break

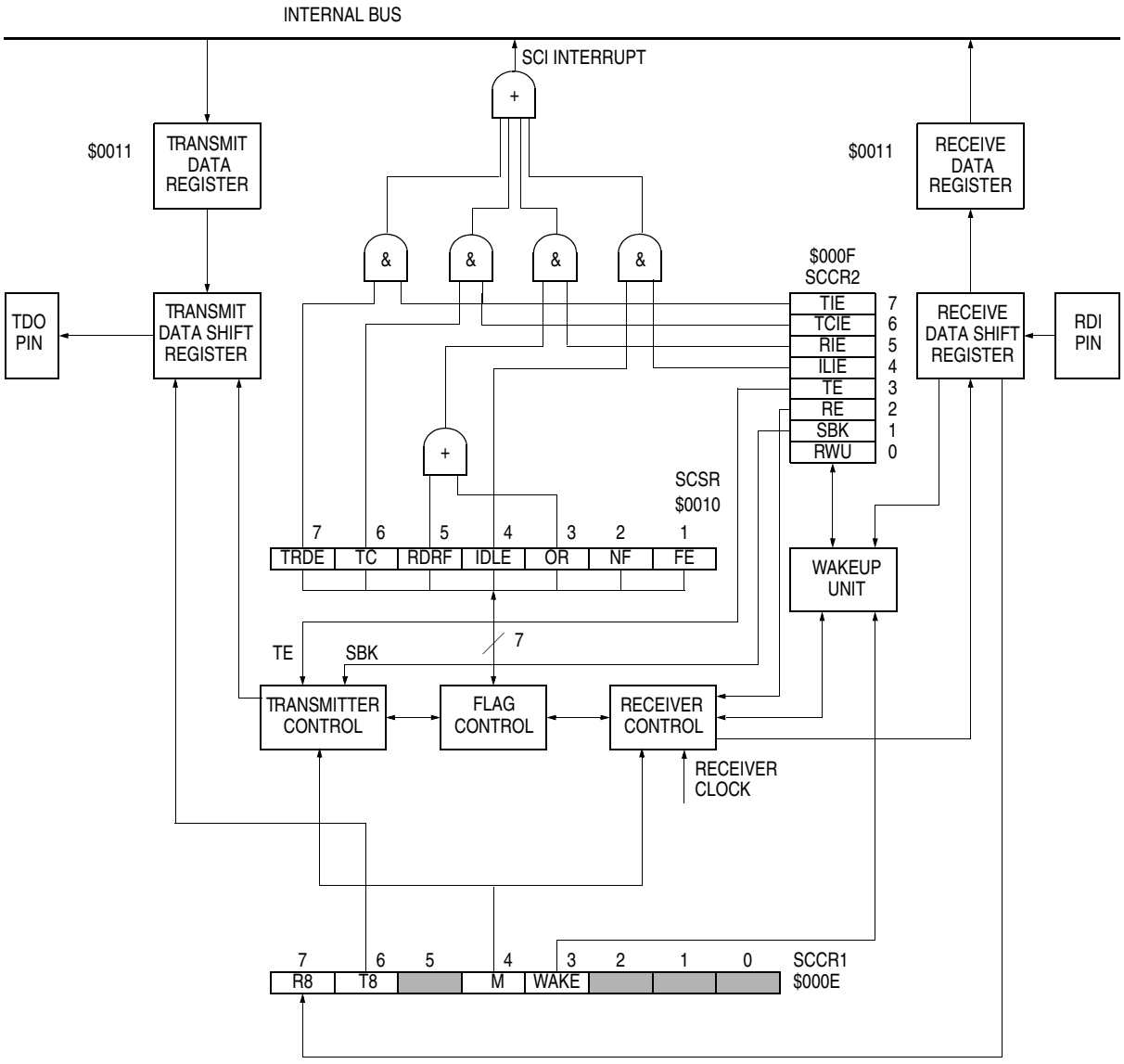


Figure 9-1. Serial Communications Interface Block Diagram

NOTE

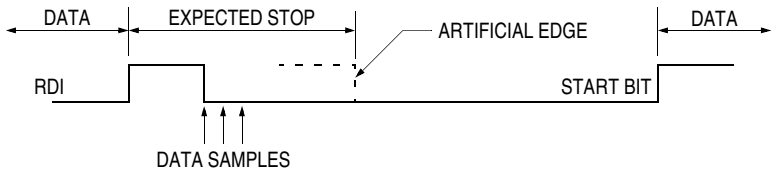
The serial communications data register (SCI SCDR) is controlled by the internal R/W signal. It is the transmit data register when written to and the receive data register when read.

9.11 Start Bit Detection

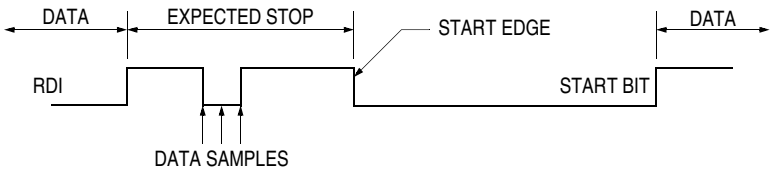
When the input (idle) line is detected low, it is tested for three more sample times (referred to as the start edge verification samples in Figure 9-4). If at least two of these three verification samples detect a logic 0, a valid start bit has been detected; otherwise, the line is assumed to be idle. A noise flag is set if all three verification samples do not detect a logic 0. Thus, a valid start bit could be assumed with a set noise flag present.

If a framing error has occurred without detection of a break (10 0s for 8-bit format or 11 0s for 9-bit format), the circuit continues to operate as if there actually was a stop bit, and the start edge will be placed artificially. The last bit received in the data shift register is inverted to a logic 1, and the three logic 1 start qualifiers (shown in Figure 9-4) are forced into the sample shift register during the interval when detection of a start bit is anticipated (see Figure 9-6); therefore, the start bit will be accepted no sooner than it is anticipated.

If the receiver detects that a break (RDRF = 1, FE = 1, receiver data register = $003B) produced the framing error, the start bit will not be artificially induced and the receiver must actually detect a logic 1 before the start bit can be recognized (see Figure 9-7).



a) Case 1: Receive line low during artificial edge



b) Case 2: Receive line high during expected start edge

Figure 9-6. SCI Artificial Start Following a Frame Error

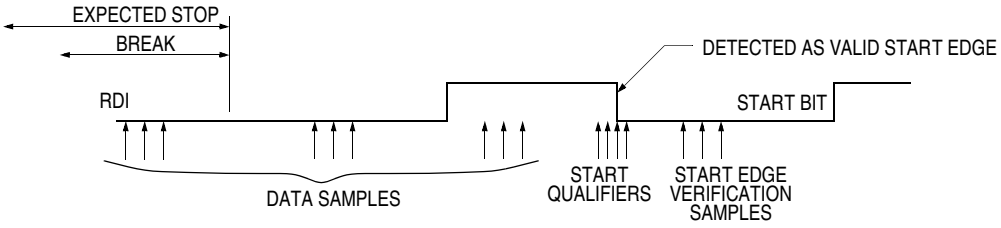


Figure 9-7. SCI Start Bit Following a Break

9.12 Transmit Data Out (TDO)

Transmit data is the serial data from the internal data bus that is applied through the SCI to the output line. Data format is as discussed in 9.6 Data Format and shown in Figure 9-3. The transmitter generates a bit time by using a derivative of the RT clock, thus producing a transmission rate equal to 1/16th that of the receiver sample clock.

9.13 SCI I/O Registers

The following I/O registers control and monitor SCI operation:

- SCI data register (SCDR)
- SCI control register 1 (SCCR1)
- SCI control register 2 (SCCR2)
- SCI status register (SCSR)

9.13.1 SCI Data Register

The SCI data register (SCDR), shown in Figure 9-8, is the buffer for characters received and for characters transmitted.

$0011	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	BIT7	BIT6	BIT55	BIT4	BIT3	BIT2	BIT1	BIT0
Reset:	Unaffected by reset							

Figure 9-8. SCI Data Register (SCDR)

9.13.2 SCI Control Register 1

The SCI control register 1 (SCCR1), shown in Figure 9-9, has these functions:

- Stores ninth SCI data bit received and ninth SCI data bit transmitted
- Controls SCI character length
- Controls SCI wakeup method

$000E	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	R8	T8		M	WAKE			
Reset:	U	U	0	U	U	0	0	0

= Unimplemented U = Undetermined

Figure 9-9. SCI Control Register 1 (SCCR1)

R8 — Bit 8 (Received)

When the SCI is receiving 9-bit characters, R8 is the ninth bit of the received character. R8 receives the ninth bit at the same time that the SCDR receives the other eight bits. Resets have no effect on the R8 bit.

11.3.3 Jump/Branch Instructions

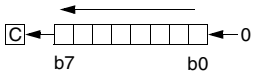
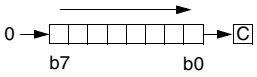
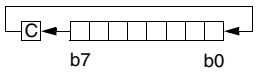
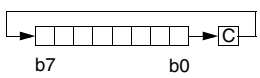
Jump instructions allow the CPU to interrupt the normal sequence of the program counter. The unconditional jump instruction (JMP) and the jump-to-subroutine instruction (JSR) have no register operand. Branch instructions allow the CPU to interrupt the normal sequence of the program counter when a test condition is met. If the test condition is not met, the branch is not performed.

The BRCLR and BRSET instructions cause a branch based on the state of any readable bit in the first 256 memory locations. These 3-byte instructions use a combination of direct addressing and relative addressing. The direct address of the byte to be tested is in the byte following the opcode. The third byte is the signed offset byte. The CPU finds the effective branch destination by adding the third byte to the program counter if the specified bit tests true. The bit to be tested and its condition (set or clear) is part of the opcode. The span of branching is from –128 to +127 from the address of the next location after the branch instruction. The CPU also transfers the tested bit to the carry/borrow bit of the condition code register.

Table 11-3. Jump and Branch Instructions

Instruction	Mnemonic
Branch if Carry Bit Clear	BCC
Branch if Carry Bit Set	BCS
Branch if Equal	BEQ
Branch if Half-Carry Bit Clear	BHCC
Branch if Half-Carry Bit Set	BHCS
Branch if Higher	BHI
Branch if Higher or Same	BHS
Branch if $\overline{\text{IRQ}}$ Pin High	BIH
Branch if $\overline{\text{IRQ}}$ Pin Low	BIL
Branch if Lower	BLO
Branch if Lower or Same	BLS
Branch if Interrupt Mask Clear	BMC
Branch if Minus	BMI
Branch if Interrupt Mask Set	BMS
Branch if Not Equal	BNE
Branch if Plus	BPL
Branch Always	BRA
Branch if Bit Clear	BRCLR
Branch Never	BRN
Branch if Bit Set	BRSET
Branch to Subroutine	BSR
Unconditional Jump	JMP
Jump to Subroutine	JSR

Table 11-6. Instruction Set Summary (Sheet 4 of 6)

Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			H	I	N	Z	C				
LDA #*opr* LDA *opr* LDA *opr* LDA *opr*,X LDA *opr*,X LDA ,X	Load Accumulator with Memory Byte	A ← (M)	—	—	↕	↕	—	IMM DIR EXT IX2 IX1 IX	A6 B6 C6 D6 E6 F6	ii dd hh ll ee ff ff	2 3 4 5 4 3
LDX #*opr* LDX *opr* LDX *opr* LDX *opr*,X LDX *opr*,X LDX ,X	Load Index Register with Memory Byte	X ← (M)	—	—	↕	↕	—	IMM DIR EXT IX2 IX1 IX	AE BE CE DE EE FE	ii dd hh ll ee ff ff	2 3 4 5 4 3
LSL *opr* LSLA LSLX LSL *opr*,X LSL ,X	Logical Shift Left (Same as ASL)		—	—	↕	↕	↕	DIR INH INH IX1 IX	38 48 58 68 78	dd ff 	5 3 3 6 5
LSR *opr* LSRA LSRX LSR *opr*,X LSR ,X	Logical Shift Right		—	—	0	↕	↕	DIR INH INH IX1 IX	34 44 54 64 74	dd ff 	5 3 3 6 5
MUL	Unsigned Multiply	X : A ← (X) × (A)	0	—	—	—	0	INH	42		1 1
NEG *opr* NEGA NEGX NEG *opr*,X NEG ,X	Negate Byte (Two's Complement)	M ← –(M) = $00 – (M) A ← –(A) = $00 – (A) X ← –(X) = $00 – (X) M ← –(M) = $00 – (M) M ← –(M) = $00 – (M)	—	—	↕	↕	↕	DIR INH INH IX1 IX	30 40 50 60 70	dd ff 	5 3 3 6 5
NOP	No Operation		—	—	—	—	—	INH	9D		2
ORA #*opr* ORA *opr* ORA *opr* ORA *opr*,X ORA *opr*,X ORA ,X	Logical OR Accumulator with Memory	A ← (A) ∨ (M)	—	—	↕	↕	—	IMM DIR EXT IX2 IX1 IX	AA BA CA DA EA FA	ii dd hh ll ee ff ff	2 3 4 5 4 3
ROL *opr* ROLA ROLX ROL *opr*,X ROL ,X	Rotate Byte Left through Carry Bit		—	—	↕	↕	↕	DIR INH INH IX1 IX	39 49 59 69 79	dd ff 	5 3 3 6 5
ROR *opr* RORA RORX ROR *opr*,X ROR ,X	Rotate Byte Right through Carry Bit		—	—	↕	↕	↕	DIR INH INH IX1 IX	36 46 56 66 76	dd ff 	5 3 3 6 5
RSP	Reset Stack Pointer	SP ← $00FF	—	—	—	—	—	INH	9C		2

12.4 Power Considerations

The average chip-junction temperature, T_J, in °C, can be obtained from:

$$T_J = T_A + (P_D \times \theta_{JA}) \quad (1)$$

where:

T_A = Ambient temperature, °C
θ_{JA} = Package thermal resistance, junction to ambient, °C/W
$P_D = P_{INT} + P_{I/O}$
$P_{INT} = I_{DD} \times V_{DD}$ watts (chip internal power)
$P_{I/O}$ = Power dissipation on input and output pins (user determined)

For most applications $P_{I/O} \ll P_{INT}$ and can be neglected.

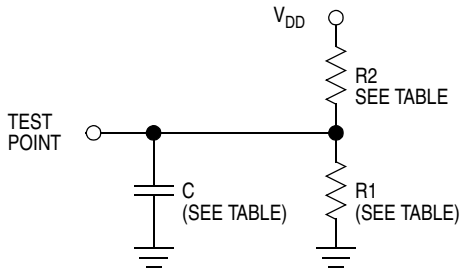
The following is an approximate relationship between P_D and T_J (neglecting P_J):

$$P_D = K \div (T_J + 273\ °C) \quad (2)$$

Solving equations (1) and (2) for K gives:

$$K = P_D \times (T_A + 273\ °C) + \theta_{JA} \times (P_D)^2 \quad (3)$$

where K is a constant pertaining to the particular part. K can be determined from equation (3) by measuring P_D (at equilibrium) for a known T_A. Using this value of K, the values of P_D and T_J can be obtained by solving equations (1) and (2) iteratively for any value of T_A.



V_{DD} = 4.5 V

Pins	R1	R2	C
PA7–PA0 PB7–PB0 PC7–PC0 PD5–PD0, PD7	3.26 Ω	2.38 Ω	50 pF

V_{DD} = 3.0 V

Pins	R1	R2	C
PA7–PA0 PB7–PB0 PC7–PC0 PD5–PD0, PD7	10.91 Ω	6.32 Ω	50 pF

Figure 12-1. Test Load

12.9 5.0-Vdc Serial Peripheral Interface Timing

No.	Characteristic[1]	Symbol	Min	Max	Unit
	Operating frequency Master Slave	 $f_{OP(M)}$ $f_{OP(S)}$	 dc dc	 0.5 2.1	f_{OP} MHz
1	Cycle time Master Slave	 $t_{CYC(M)}$ $t_{CYC(S)}$	 2.0 480	 — —	t_{CYC} ns
2	Enable lead time Master Slave	 $t_{LEAD(M)}$ $t_{LEAD(S)}$	 (2) 240	 — —	ns
3	Enable lag time Master Slave	 $t_{LAG(M)}$ $t_{LAG(S)}$	 (2) 720	 — —	ns
4	Clock (SCK) high time Master Slave	 $t_{W(SCKH)M}$ $t_{W(SCKH)S}$	 340 190	 — —	ns
5	Clock (SCK) low time Master Slave	 $t_{W(SCKL)M}$ $t_{W(SCKL)S}$	 340 190	 — —	ns
6	Data setup time (inputs) Master Slave	 $t_{SU(M)}$ $t_{SU(S)}$	 100 100	 — —	ns
7	Data hold time (inputs) Master Slave	 $t_{H(M)}$ $t_{H(S)}$	 100 100	 — —	ns
8	Slave access time (time to data active from high-impedance state)	t_A	0	120	ns
9	Slave disable time (hold time to high-impedance state)	t_{DIS}	—	240	ns
10	Data valid Master (before capture edge) Slave (after enable edge)[3]	 $t_{V(M)}$ $t_{V(S)}$	 0.25 —	 — 240	 $t_{CYC(M)}$ ns
11	Data hold time (outputs) Master (after capture edge) slave (After Enable Edge)	 $t_{HO(M)}$ $t_{HO(S)}$	 0.25 0	 — —	 $t_{CYC(M)}$ ns
12	Rise time (20% V_{DD} to 70% V_{DD}, C_L = 200 pF) SPI outputs (SCK, MOSI, and MISO) SPI inputs (SCK, MOSI, MISO, and $\overline{SS}$)	 t_{RM} t_{RS}	 — —	 100 2.0	 ns µs
13	Fall time (70% V_{DD} to 20% V_{DD}, C_L = 200 pF) SPI outputs (SCK, MOSI, and MISO) SPI inputs (SCK, MOSI, MISO, and $\overline{SS}$)	 t_{FM} t_{FS}	 — —	 100 2.0	 ns µs

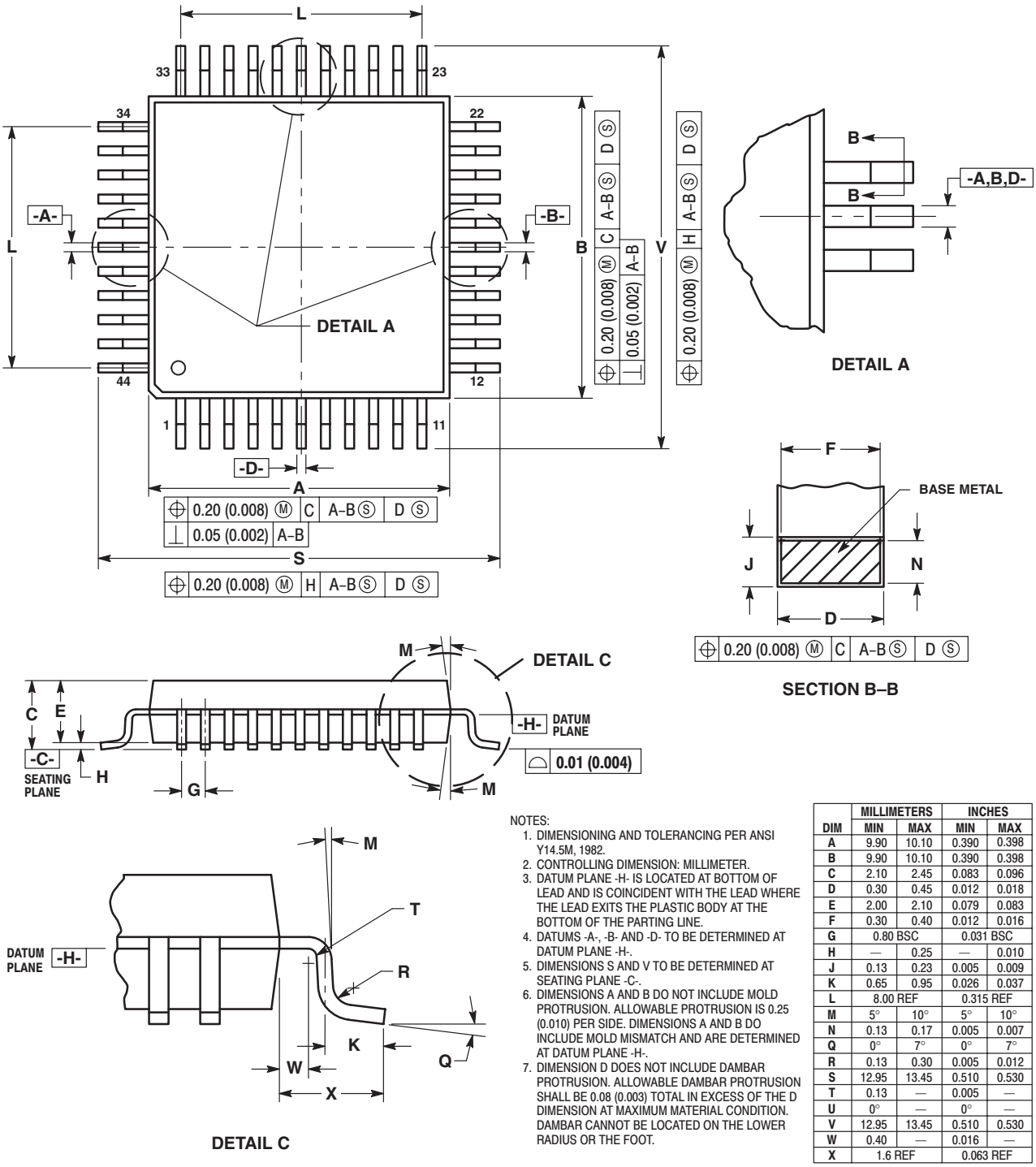
1. V_{DD} = 5.0 Vdc ± 10%; V_{SS} = 0 Vdc, T_A = –40 to +85°C, unless otherwise noted. Refer to Figure 12-9 and Figure 12-10.
2. Signal production depends on software.
3. Assumes 200 pF load on all SPI pins.

12.10 3.3- Vdc Serial Peirpheral Interface Timing

No.	Characteristic[1]	Symbol	Min	Max	Unit
	Operating frequency Master Slave	 $f_{OP(M)}$ $f_{OP(S)}$	 dc dc	 0.5 1.0	f_{OP} MHz
1	Cycle time Master Slave	 $t_{CYC(M)}$ $t_{CYC(S)}$	 2.0 1.0	 — —	t_{CYC} µs
2	Enable lead time Master Slave	 $t_{LEAD(M)}$ $t_{LEAD(S)}$	 (2) 500	 — —	ns
3	Enable lag time Master Slave	 $t_{LAG(M)}$ $t_{LAG(S)}$	 (2) 1.5	 — —	ns µs
4	Clock (SCK) high time Master Slave	 $t_{W(SCKH)M}$ $t_{W(SCKH)S}$	 720 400	 — —	ns
5	Clock (SCK) low time Master Slave	 $t_{W(SCKL)M}$ $t_{W(SCKL)S}$	 720 400	 — —	ns
6	Data setup time (inputs) Master Slave	 $t_{SU(M)}$ $t_{SU(S)}$	 200 200	 — —	ns
7	Data hold time (inputs) Master Slave	 $t_{H(M)}$ $t_{H(S)}$	 200 200	 — —	ns
8	Slave access time (time to data active from high-impedance state)	t_A	0	250	ns
9	Slave disable time (hold time to high-impedance state)	t_{DIS}	—	500	ns
10	Data valid Master (before capture edge) Slave (after enable edge)[3]	 $t_{V(M)}$ $t_{V(S)}$	 0.25 —	 — 500	 $t_{CYC(M)}$ ns
11	Data hold time (outputs) Master (after capture edge) Slave (after enable edge)	 $t_{HO(M)}$ $t_{HO(S)}$	 0.25 0	 — —	 $t_{CYC(M)}$ ns
12	Rise time (20% V_{DD} to 70% V_{DD}, C_L = 200 pF) SPI outputs (SCK, MOSI, and MISO) SPI inputs (SCK, MOSI, MISO, and $\overline{SS}$)	 t_{RM} t_{RS}	 — —	 200 2.0	 ns µs
13	Fall time (70% V_{DD} to 20% V_{DD}, C_L = 200 pF) SPI outputs (SCK, MOSI, and MISO) SPI inputs (SCK, MOSI, MISO, and $\overline{SS}$)	 t_{FM} t_{FS}	 — —	 200 2.0	 ns µs

1. V_{DD} = 3.3 Vdc ± 0.3 Vdc; V_{SS} = 0 Vdc, T_A = –40 to +85 °C, unless otherwise noted. Refer to Figure 12-9 and Figure 12-10.
2. Signal production depends on software.
3. Assumes 200 pF load on all SPI pins.

13.5 44-Lead Quad Flat Pack (QFP) (Case 824A-01)



NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: MILLIMETER.
3. DATUM PLANE -H- IS LOCATED AT BOTTOM OF LEAD AND IS COINCIDENT WITH THE LEAD WHERE THE LEAD EXITS THE PLASTIC BODY AT THE BOTTOM OF THE PARTING LINE.
4. DATUMS -A-, -B- AND -D- TO BE DETERMINED AT DATUM PLANE -H-.
5. DIMENSIONS S AND V TO BE DETERMINED AT SEATING PLANE -C-.
6. DIMENSIONS A AND B DO NOT INCLUDE MOLD PROTRUSION. ALLOWABLE PROTRUSION IS 0.25 (0.010) PER SIDE. DIMENSIONS A AND B DO INCLUDE MOLD MISMATCH AND ARE DETERMINED AT DATUM PLANE -H-.
7. DIMENSION D DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL BE 0.08 (0.003) TOTAL IN EXCESS OF THE D DIMENSION AT MAXIMUM MATERIAL CONDITION. DAMBAR CANNOT BE LOCATED ON THE LOWER RADIUS OR THE FOOT.

	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	9.90	10.10	0.390	0.398
B	9.90	10.10	0.390	0.398
C	2.10	2.45	0.083	0.096
D	0.30	0.45	0.012	0.018
E	2.00	2.10	0.079	0.083
F	0.30	0.40	0.012	0.016
G	0.80 BSC		0.031 BSC	
H	—	0.25	—	0.010
J	0.13	0.23	0.005	0.009
K	0.65	0.95	0.026	0.037
L	8.00 REF		0.315 REF	
M	5°	10°	5°	10°
N	0.13	0.17	0.005	0.007
Q	0°	7°	0°	7°
R	0.13	0.30	0.005	0.012
S	12.95	13.45	0.510	0.530
T	0.13	—	0.005	—
U	0°	—	0°	—
V	12.95	13.45	0.510	0.530
W	0.40	—	0.016	—
X	1.6 REF		0.063 REF	

Figure 13-4. 44-Lead QFP (Case 824A-01)