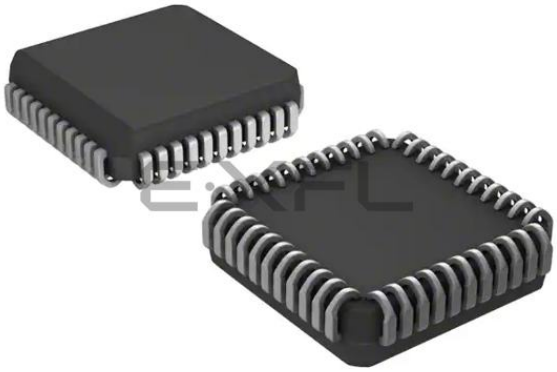


Welcome to E-XFL.COM

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	HC11
Core Size	8-Bit
Speed	3MHz
Connectivity	SCI, SPI
Peripherals	POR, WDT
Number of I/O	26
Program Memory Size	-
Program Memory Type	ROMless
EEPROM Size	-
RAM Size	192 x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-LCC (J-Lead)
Supplier Device Package	44-PLCC (16.59x16.59)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc68hc11d0cfne3

If this mode is entered out of reset, the EPROM is located at $7000–$7FFF and vector accesses are from external memory. To be in expanded-multiplexed mode with EPROM located at $F000–$FFFF, it is necessary to start in single-chip mode, executing out of EPROM, and then set the MDA bit of the HPRIO register to switch mode.

NOTE

R/$\overline{W}$, AS, and the high-order address bus (port B) are inputs in single-chip mode. These inputs may need to be pulled up so that off-chip accesses cannot occur while the MCU is in single-chip mode.

2.2.3 Special Bootstrap Mode (BOOT)

This special mode is similar to single-chip mode. The resident bootloader program contains a 256-byte program in a special on-chip read-only memory (ROM). The user downloads a small program into on-board RAM using the SCI port. Program control is passed to RAM when an idle line of at least four characters occurs. In this mode, all interrupt vectors are mapped to RAM (see Table 2-2), so that the user can set up a jump table, if desired.

Bootstrap mode (BOOT) is entered out of reset if the voltage level on both MODA and MODB is low. The programming aspect of bootstrap mode, used to program the one-time programmable ROM (OTPROM) through the MCU, is entered automatically if $\overline{IRQ}$ is low and programming voltage is available on the V_{PP} pin. $\overline{IRQ}$ should be pulled up while in reset with MODA and MODB configured for bootstrap mode to prevent unintentional programming of the EPROM.

This versatile mode (BOOT) can be used for test and diagnostic functions on completed modules and for programming the on-board PROM. The serial receive logic is initialized by software in the bootloader ROM, which provides program control for the SCI baud rate and word format. Mode switching to other modes can occur under program control by writing to the SMOD and MDA bits of the HPRIO register. Two special bootloader functions allow either an immediate jump-to-RAM at memory address $0000 or an immediate jump-to-EPROM at $F000.

Table 2-2. Bootstrap Mode Jump Vectors

Address	Vector
00C4	SCI
00C7	SPI
00CA	Pulse accumulator input edge
00CD	Pulse accumulator overflow
00D0	Timer overflow
00D3	Timer output compare 5/input capture 4
00D6	Timer output compare 4
00D9	Timer output compare 3
00DC	Timer output compare 2
00DF	Timer output compare 1
00E3	Timer input capture 3
00E5	Timer input capture 2
00E8	Timer input capture 1

At the end of the interrupt service routine, a return-from interrupt (RTI) instruction is executed. The RTI instruction causes the saved registers to be pulled off the stack in reverse order. Program execution resumes at the return address.

Certain instructions push and pull the A and B accumulators and the X and Y index registers and are often used to preserve program context. For example, pushing accumulator A onto the stack when entering a subroutine that uses accumulator A and then pulling accumulator A off the stack just before leaving the subroutine ensures that the contents of a register will be the same after returning from the subroutine as it was before starting the subroutine.

3.2.5 Program Counter (PC)

The program counter, a 16-bit register, contains the address of the next instruction to be executed. After reset, the program counter is initialized from one of six possible vectors, depending on operating mode and the cause of reset.
See Table 3-1.

Table 3-1. Reset Vector Comparison

Mode	POR or $\overline{\text{RESET}}$ Pin	Clock Monitor	COP Watchdog
Normal	$FFFE, $FFFF	$FFFC, $FFFD	$FFFA, $FFFB
Test or boot	$BFFE, $BFFF	$BFFC, $FFFD	$BFFA, $FFFB

3.2.6 Condition Code Register (CCR)

This 8-bit register contains:

- Five condition code indicators (C, V, Z, N, and H)
- Two interrupt masking bits ($\overline{\text{IRQ}}$ and $\overline{\text{XIRQ}}$)
- One stop disable bit (S)

In the M68HC11 CPU, condition codes are updated automatically by most instructions. For example, load accumulator A (LDAA) and store accumulator A (STAA) instructions automatically set or clear the N, Z, and V condition code flags. Pushes, pulls, add B to X (ABX), add B to Y (ABY), and transfer/exchange instructions do not affect the condition codes. Refer to Table 3-2, which shows what condition codes are affected by a particular instruction.

3.2.6.1 Carry/Borrow (C)

The C bit is set if the arithmetic logic unit (ALU) performs a carry or borrow during an arithmetic operation. The C bit also acts as an error flag for multiply and divide operations. Shift and rotate instructions operate with and through the carry bit to facilitate multiple-word shift operations.

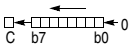
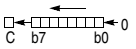
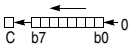
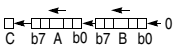
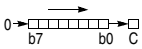
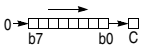
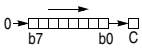
3.2.6.2 Overflow (V)

The overflow bit is set if an operation causes an arithmetic overflow. Otherwise, the V bit is cleared.

3.2.6.3 Zero (Z)

The Z bit is set if the result of an arithmetic, logic, or data manipulation operation is 0. Otherwise, the Z bit is cleared. Compare instructions do an internal implied subtraction and the condition codes, including Z, reflect the results of that subtraction. A few operations (INX, DEX, INY, and DEY) affect the Z bit and no other condition flags. For these operations, only = and ≠ conditions can be determined.

Table 3-2. Instruction Set (Sheet 5 of 8)

Mnemonic	Operation	Description	Addressing Mode	Instruction			Condition Codes							
				Opcode	Operand	Cycles	S	X	H	I	N	Z	V	C
INX	Increment Index Register X	IX + 1 ⇒ IX	INH	08	—	3	—	—	—	—	—	Δ	—	—
INY	Increment Index Register Y	IY + 1 ⇒ IY	INH	18 08	—	4	—	—	—	—	—	Δ	—	—
JMP (opr)	Jump	See Figure 3-2	EXT IND,X IND,Y	7E 6E 18 6E	hh ll ff ff	3 3 4	—	—	—	—	—	—	—	—
JSR (opr)	Jump to Subroutine	See Figure 3-2	DIR EXT IND,X IND,Y	9D BD AD 18 AD	dd hh ll ff ff	5 6 6 7	—	—	—	—	—	—	—	—
LDAA (opr)	Load Accumulator A	M ⇒ A	A IMM A DIR A EXT A IND,X A IND,Y	86 96 B6 A6 18 A6	ii dd hh ll ff ff	2 3 4 4 5	—	—	—	—	Δ	Δ	0	—
LDAB (opr)	Load Accumulator B	M ⇒ B	B IMM B DIR B EXT B IND,X B IND,Y	C6 D6 F6 E6 18 E6	ii dd hh ll ff ff	2 3 4 4 5	—	—	—	—	Δ	Δ	0	—
LDD (opr)	Load Double Accumulator D	M ⇒ A,M + 1 ⇒ B	IMM DIR EXT IND,X IND,Y	CC DC FC EC 18 EC	jj kk dd hh ll ff ff	3 4 5 5 6	—	—	—	—	Δ	Δ	0	—
LDS (opr)	Load Stack Pointer	M : M + 1 ⇒ SP	IMM DIR EXT IND,X IND,Y	8E 9E BE AE 18 AE	jj kk dd hh ll ff ff	3 4 5 5 6	—	—	—	—	Δ	Δ	0	—
LDX (opr)	Load Index Register X	M : M + 1 ⇒ IX	IMM DIR EXT IND,X IND,Y	CE DE FE EE CD EE	jj kk dd hh ll ff ff	3 4 5 5 6	—	—	—	—	Δ	Δ	0	—
LDY (opr)	Load Index Register Y	M : M + 1 ⇒ IY	IMM DIR EXT IND,X IND,Y	18 CE 18 DE 18 FE 1A EE 18 EE	jj kk dd hh ll ff ff	4 5 6 6 6	—	—	—	—	Δ	Δ	0	—
LSL (opr)	Logical Shift Left		EXT IND,X IND,Y	78 68 18 68	hh ll ff ff	6 6 7	—	—	—	—	Δ	Δ	Δ	Δ
LSLA	Logical Shift Left A		A INH	48	—	2	—	—	—	—	Δ	Δ	Δ	Δ
LSLB	Logical Shift Left B		B INH	58	—	2	—	—	—	—	Δ	Δ	Δ	Δ
LSLD	Logical Shift Left Double		INH	05	—	3	—	—	—	—	Δ	Δ	Δ	Δ
LSR (opr)	Logical Shift Right		EXT IND,X IND,Y	74 64 18 64	hh ll ff ff	6 6 7	—	—	—	—	0	Δ	Δ	Δ
LSRA	Logical Shift Right A		A INH	44	—	2	—	—	—	—	0	Δ	Δ	Δ
LSRB	Logical Shift Right B		B INH	54	—	2	—	—	—	—	0	Δ	Δ	Δ

Chapter 4
Resets, Interrupts, and Low-Power Modes

4.1 Introduction

This section describes the internal and external resets and interrupts of the MC68HC711D3 and its two low power-consumption modes.

4.2 Resets

The microcontroller unit (MCU) can be reset in any of these four ways:

1. An active-low input to the $\overline{\text{RESET}}$ pin
2. A power-on reset (POR) function
3. A clock monitor failure
4. A computer operating properly (COP) watchdog timer timeout

The $\overline{\text{RESET}}$ input consists mainly of a Schmitt trigger that senses the $\overline{\text{RESET}}$ line logic level.

4.2.1 $\overline{\text{RESET}}$ Pin

To request an external reset, the $\overline{\text{RESET}}$ pin must be held low for at least eight E-clock cycles, or for one E-clock cycle if no distinction is needed between internal and external resets.

4.2.2 Power-On Reset (POR)

Power-on reset occurs when a positive transition is detected on V_{DD}. This reset is used strictly for power turn on conditions and should not be used to detect any drop in the power supply voltage. If the external $\overline{\text{RESET}}$ pin is low at the end of the power-on delay time, the processor remains in the reset condition until $\overline{\text{RESET}}$ goes high.

4.2.3 Computer Operating Properly (COP) Reset

The MCU contains a watchdog timer that automatically times out unless it is serviced within a specific time by a program reset sequence. If the COP watchdog timer is allowed to timeout, a reset is generated, which drives the $\overline{\text{RESET}}$ pin low to reset the MCU and the external system.

In the MC68HC711D3, the COP reset function is enabled out of reset in normal modes. If the user does not want the COP enabled, he must write a 1 to the NOCOP bit of the configuration control register (CONFIG) after reset. This bit is writable only once after reset in normal modes (see 2.3.3 Configuration Control Register for more information). Protected control bits (CR1 and CR0) in the configuration options register (OPTION) allow the user to select one of the four COP timeout rates. Table 4-1 shows the relationship between CR1 and CR0 and the COP timeout period for various system clock frequencies.

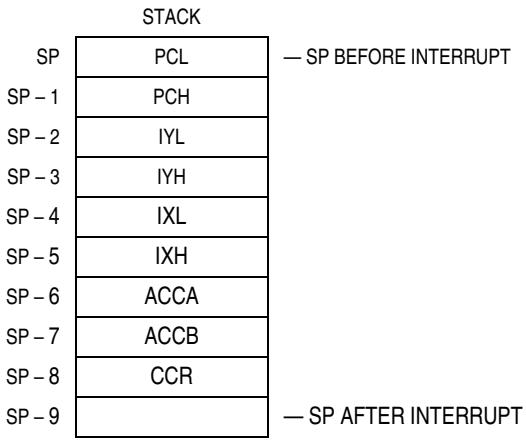


Figure 4-3. Interrupt Stacking Order

4.3.1 Software Interrupt (SWI)

The SWI is executed the same as any other instruction and takes precedence over interrupts only if the other interrupts are masked (with I and X bits in the CCR set). SWI execution is similar to that of the maskable interrupts in that it sets the I bit, stacks the central processor unit (CPU) registers, etc.

NOTE

The SWI instruction cannot be executed as long as another interrupt is pending. However, once the SWI instruction has begun, no other interrupt can be honored until the first instruction in the SWI service routine is completed.

4.3.2 Illegal Opcode Trap

Since not all possible opcodes or opcode sequences are defined, an illegal opcode detection circuit has been included in the MCU. When an illegal opcode is detected, an interrupt is required to the illegal opcode vector. The illegal opcode vector should never be left uninitialized.

4.3.3 Real-Time Interrupt (RTI)

The real-time interrupt (RTI) provides a programmable periodic interrupt. This interrupt is maskable by either the I bit in the CCR or the RTI enable (RTII) bit of the timer interrupt mask register 2 (TMSK2). The rate is based on the MCU E clock and is software selectable to the $E \div 2^{13}$, $E \div 2^{14}$, $E \div 2^{15}$, or $E \div 2^{16}$. See PACTL, TMSK2, and TFLG2 register descriptions in Chapter 8 Programmable Timer for control and status bit information.

4.3.4 Interrupt Mask Bits in the CCR

Upon reset, both the X bit and I bit of the CCR are set to inhibit all maskable interrupts and XIRQ. After minimum system initialization, software may clear the X bit by a TAP instruction, thus enabling XIRQ interrupts. Thereafter software cannot set the X bit. So, an XIRQ interrupt is effectively a non-maskable interrupt. Since the operation of the I bit related interrupt structure has no effect on the X bit, the internal $\overline{\text{XIRQ}}$ pin remains effectively non-masked. In the interrupt priority logic, the XIRQ interrupt is a higher priority than any source that is maskable by the I bit. All I bit related interrupts operate normally with their own priority relationship.

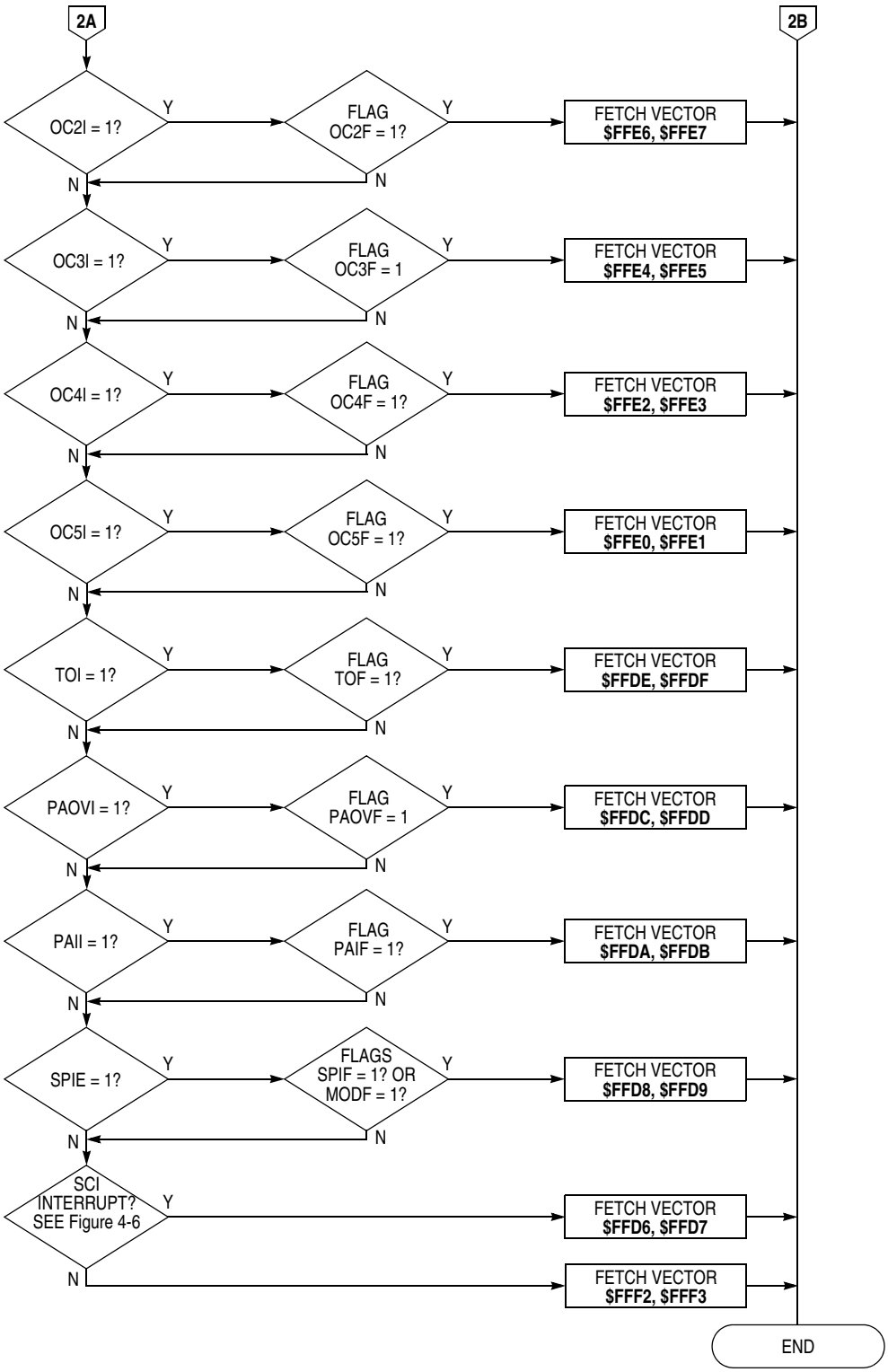


Figure 4-5. Interrupt Priority Resolution (Sheet 2 of 2)

5.5 Port D

Port D is an 8-bit, general-purpose I/O port with a data register (PORTD) and a data direction register (DDRD). The eight port D bits (D7–D0) can be used for general-purpose I/O, for the serial communications interface (SCI) and serial peripheral interface (SPI) subsystems, or for bus data direction control

5.5.1 Port D Data Register

Address: $0008

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
Reset:	0	0	0	0	0	0	0	0

Figure 5-7. Port D Data Register (PORTD)

PORTD can be read at any time and inputs return the sensed levels at the pin; whereas, outputs return the input level of the port D pin drivers. If PORTD is written, the data is stored in an internal latch, and can be driven only if port D is configured as general-purpose output. This port shares functions with the on-chip SCI and SPI subsystems, while bits 6 and 7 control the direction of data flow on the bus in expanded and special test modes.

5.5.2 Port D Data Direction Register

Address: $0009

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	DDD7	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0
Reset:	0	0	0	0	0	0	0	0

Figure 5-8. Data Direction Register for Port D (DDRD)

DDD7–DDD0 — Data Direction for Port D

When port D is a general-purpose I/O port, the DDRD register controls the direction of the I/O pins as follows:

0 = Configures the corresponding port D pin for input only
1 = Configures the corresponding port D pin for output

In expanded and test modes, bits 6 and 7 are dedicated AS and R/$\overline{W}$.

When port D is functioning with the SPI system enabled, bit 5 is dedicated as the slave select ($\overline{SS}$) input. In SPI slave mode, DDD5 has no meaning or effect. In SPI master mode, DDD5 affects port D bit 5 as follows:

0 = Port D bit 5 is an error-detect input to the SPI.
1 = Port D bit 5 is configured as a general-purpose output line.

If the SPI is enabled and expects port D bits 2, 3, and 4 (MISO, MOSI, and SCK) to be inputs, then they are inputs, regardless of the state of DDRD bits 2, 3, and 4. If the SPI expects port D bits 2, 3, and 4 to be outputs, they are outputs only if DDRD bits 2, 3, and 4 are set.

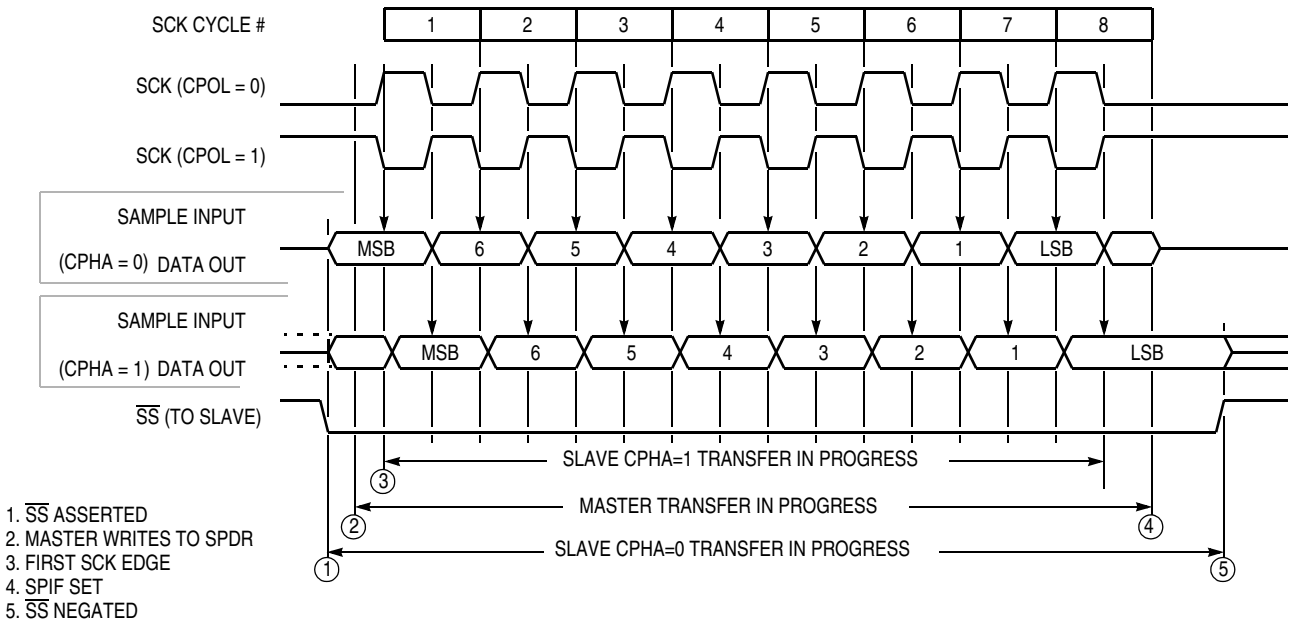


Figure 7-2. SPI Transfer Format

7.4 Clock Phase and Polarity Controls

Software can select one of four combinations of serial clock phase and polarity using two bits in the SPI control register (SPCR). The clock polarity is specified by the CPOL control bit, which selects an active high or active low clock, and has no significant effect on the transfer format. The clock phase (CPHA) control bit selects one of two different transfer formats. The clock phase and polarity should be identical for the master SPI device and the communicating slave device. In some cases, the phase and polarity are changed between transfers to allow a master device to communicate with peripheral slaves having different requirements.

When CPHA equals 0, the slave select ($\overline{SS}$) line must be negated and reasserted between each successive serial byte. Also, if the slave writes data to the SPI data register (SPDR) while $\overline{SS}$ is active low, a write collision error results.

When CPHA equals 1, the $\overline{SS}$ line can remain low between successive transfers.

7.5 SPI Signals

This subsection contains description of the four SPI signals:

- Master in/slave out (MISO)
- Master out/slave in (MOSI)
- Serial clock (SCK)
- Slave select ($\overline{SS}$)

7.5.1 Master In/Slave Out (MISO)

MISO is one of two unidirectional serial data signals. It is an input to a master device and an output from a slave device. The MISO line of a slave device is placed in the high-impedance state if the slave device is not selected.

Table 8-1. Timer Summary

Control Bits	XTAL Frequencies			
	4.0 MHz	**8.0 MHz**	**12.0 MHz**	**Other Rates**
	1.0 MHz	**2.0 MHz**	**3.0 MHz**	**(E)**
	1000 ns	**500 ns**	**333 ns**	**(1/E)**
PR1 and PR0	**Main Timer Count Rates**			
0 0 1 count — overflow —	1.0 µs 65.536 ms	500 ns 32.768 ms	333 ns 21.845 ms	(E/1) ($E/2^{16}$)
0 1 1 count — overflow —	4.0 µs 262.14 ms	2.0 µs 131.07 ms	1.333 µs 87.381 ms	(E/4) ($E/2^{18}$)
1 0 1 count — overflow —	8.0 µs 524.29 ms	4.0 µs 262.14 ms	2.667 µs 174.76 ms	(E/8) ($E/2^{19}$)
1 1 1 count — overflow —	16.0 µs 1.049 s	8.0 µs 524.29 ms	5.333 µs 349.52 ms	(E/16) ($E/2^{20}$)

8.2 Timer Structure

Figure 8-2 shows the capture/compare system block diagram. The port A pin control block includes logic for timer functions and for general-purpose input/output (I/O). For pins PA2, PA1, and PA0, this block contains both the edge-detection logic and the control logic that enables the selection of which edge triggers an input capture. The digital level on PA2–PA0 can be read at any time (read PORTA register), even if the pin is being used for the input capture function. Pins PA6–PA4 are used for either general-purpose output or as output compare pins. Pin PA3 can be used for general-purpose I/O, input capture 4, output compare 5, or output compare 1. When one of these pins is being used for an output compare function, it cannot be written directly as if it were a general-purpose output. Each of the output compare functions (OC5–OC2) is related to one of the port A output pins. Output compare 1 (OC1) has extra control logic, allowing it optional control of any combination of the PA7–PA3 pins. The PA7 pin can be used as a general-purpose I/O pin, as an input to the pulse accumulator, or as an OC1 output pin.

8.3 Input Capture

The input capture function records the time an external event occurs by latching the value of the free-running counter when a selected edge is detected at the associated timer input pin. Software can store latched values and use them to compute the periodicity and duration of events. For example, by storing the times of successive edges of an incoming signal, software can determine the period and pulse width of a signal. To measure period, two successive edges of the same polarity are captured. To measure pulse width, two alternate polarity edges are captured.

In most cases, input capture edges are asynchronous to the internal timer counter, which is clocked relative to the PH2 clock. These asynchronous capture requests are synchronized to PH2 so that the latching occurs on the opposite half cycle of PH2 from when the timer counter is being incremented. This synchronization process introduces a delay from when the edge occurs to when the counter value is detected. Because these delays offset each other when the time between two edges is being measured, the delay can be ignored. When an input capture is being used with an output compare, there is a similar delay between the actual compare point and when the output pin changes state.

8.5 Real-Time Interrupt

The real-time interrupt feature, used to generate hardware interrupts at a fixed periodic rate, is controlled and configured by two bits (RTR1 and RTR0) in the pulse accumulator control (PACTL) register. The RTII bit in the TMSK2 register enables the interrupt capability. The four different rates available are a product of the MCU oscillator frequency and the value of bits RTR1 and RTR0. Refer to Table 8-5 for the periodic real-time interrupt rates.

Table 8-5. Periodic Real-Time Interrupt Rates

RTR1 and RTR0	E = 1 MHz	E = 2 MHz	E = 3 MHz	E = X MHz
0 0	2.731 ms	4.096 ms	8.192 ms	$(E/2^{13})$
0 1	5.461 ms	8.192 ms	16.384 ms	$(E/2^{14})$
1 0	10.923 ms	16.384 ms	32.768 ms	$(E/2^{15})$
1 1	21.845 ms	32.768 ms	65.536 ms	$(E/2^{16})$

The clock source for the RTI function is a free-running clock that cannot be stopped or interrupted except by reset. This clock causes the time between successive RTI timeouts to be a constant that is independent of the software latencies associated with flag clearing and service. For this reason, an RTI period starts from the previous timeout, not from when RTIF is cleared.

Every timeout causes the RTIF bit in TFLG2 to be set, and if RTII is set, an interrupt request is generated. After reset, one entire real-time interrupt period elapses before the RTIF flag is set for the first time. Refer to the TMSK2, TFLG2, and PACTL registers.

8.5.1 Timer Interrupt Mask 2 Register

The timer interrupt mask 2 register (TMSK2) contains the real-time interrupt enable bits.

Address: $0024

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	TOI	RTII	PAOVI	PAII	0	0	PR1	PR0
Reset:	0	0	0	0	0	0	0	0

Figure 8-16. Timer Interrupt Mask 2 Register (TMSK2)

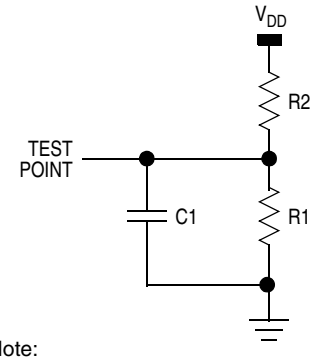
TOI — Timer Overflow Interrupt Enable Bit
Refer to 8.4 Output Compare (OC).

RTII — Real-Time Interrupt Enable Bit
0 = RTIF interrupts disabled
1 = Interrupt requested

PAOVI — Pulse Accumulator Overflow Interrupt Enable Bit
Refer to 8.7 Pulse Accumulator.

PAII — Pulse Accumulator Input Edge Bit
Refer to 8.7 Pulse Accumulator.

Bits 3–2 — Unimplemented
Always read 0.

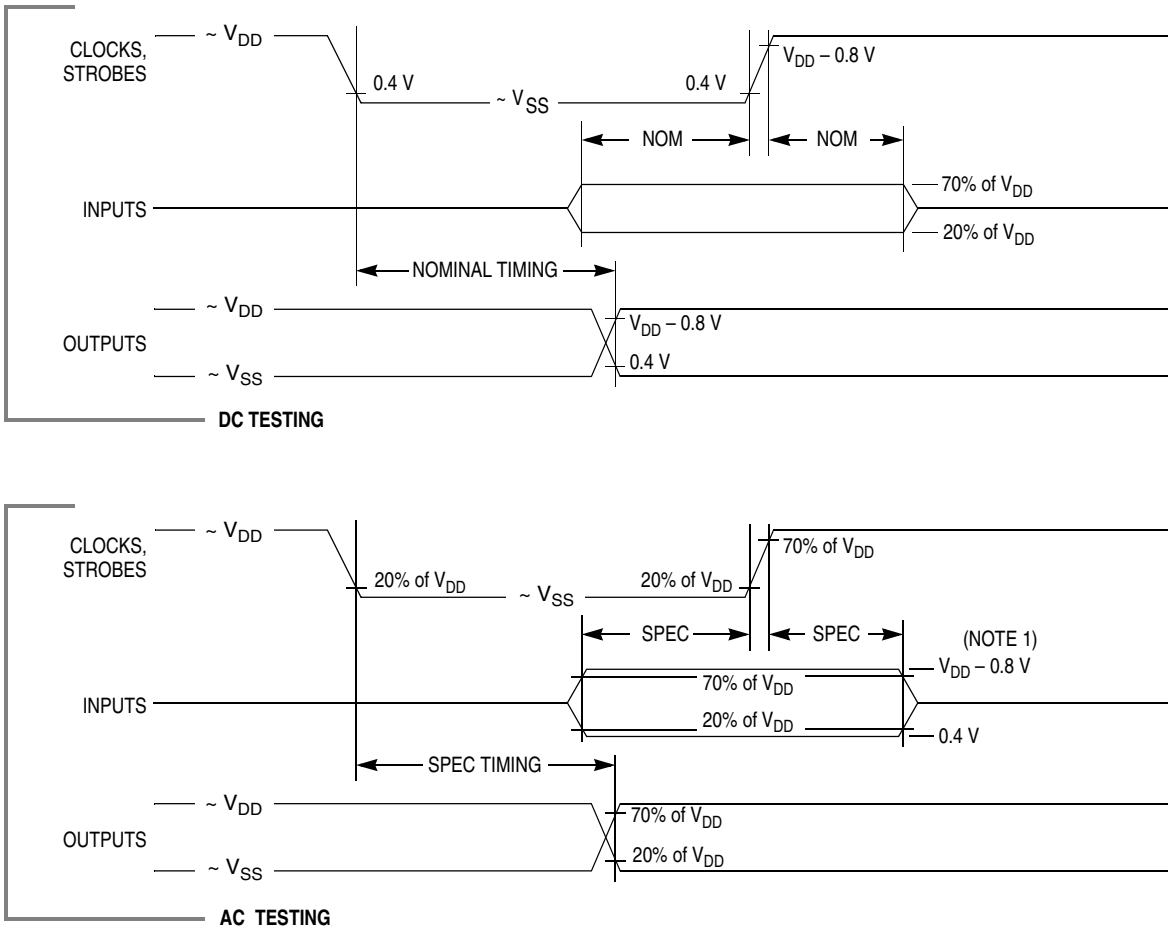


EQUIVALENT TEST LOAD(1)

Pins	R1	R2	C1
PA3–PA7 PB0–PB7 PC0–PC7 PD0, PD5–PD7 E	3.26 K	2.38 K	90 pF
PD1—PD4	3.26 K	2.38 K	200 pF

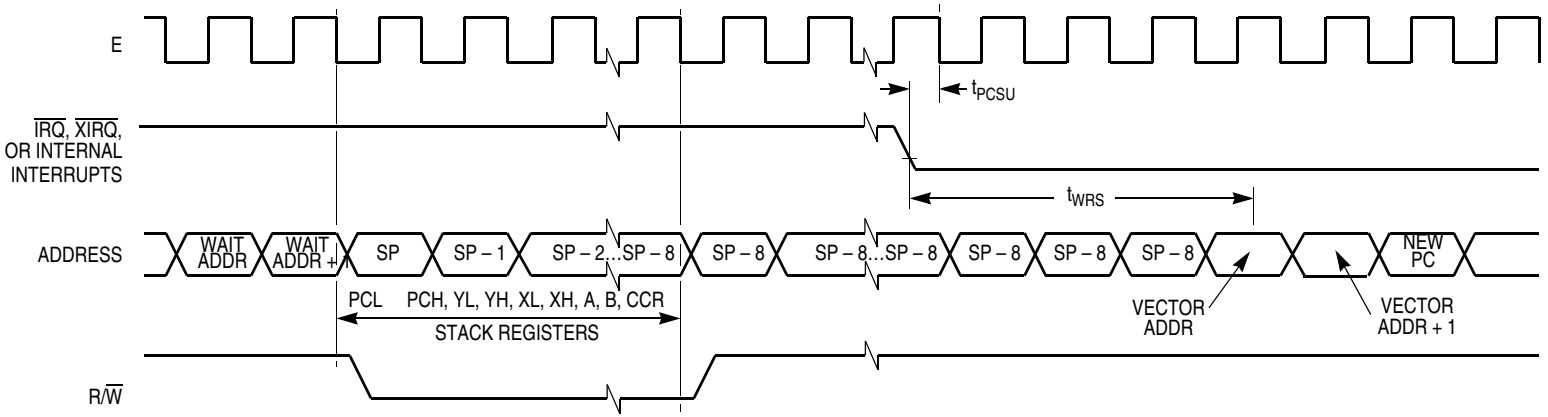
Note:
1. Full test loads are applied during all ac electrical timing measurements.

Figure 9-1. Equivalent Test Load



Note:
1. During ac timing measurements, inputs are driven to 0.4 volts and V_{DD} – 0.8 volts while timing measurements are taken at the 20% and 70% of V_{DD} points.

Figure 9-2. Test Methods



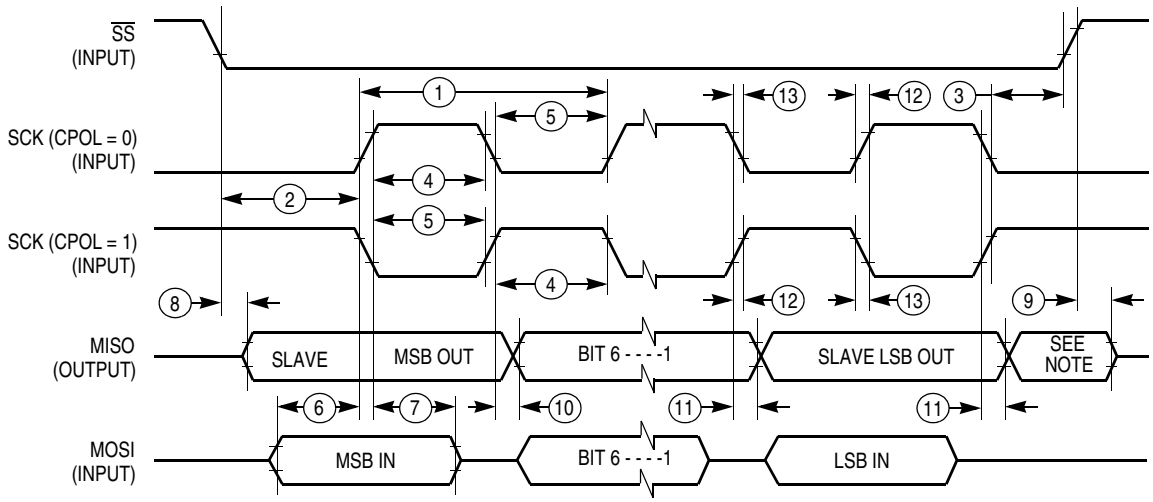
Note: $\overline{\text{RESET}}$ also causes recovery from WAIT.

Figure 9-6. WAIT Recovery Timing Diagram

9.9 Serial Peripheral Interface Timing

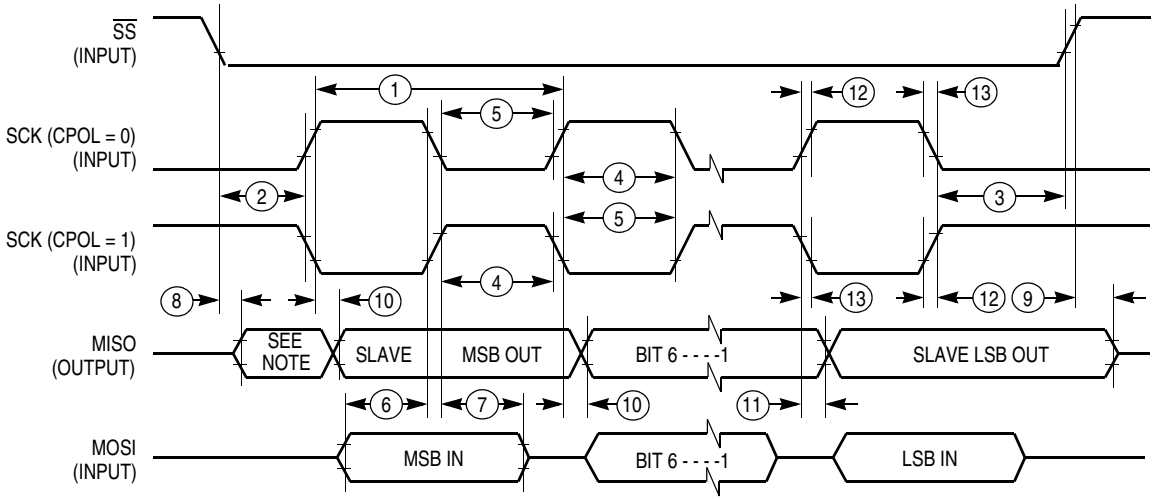
Num	Characteristic[1]	Symbol	2.0 MHz		3.0 MHz		Unit
			Min	Max	Min	Max	
	Operating frequency Master Slave	 $f_{op(m)}$ $f_{op(s)}$	 dc dc	 0.5 2.0	 dc dc	 0.5 3.0	f_{op} MHz
1	Cycle time Master Slave	 $t_{cyc(m)}$ $t_{CYC(s)}$	 2.0 500	 — —	 2.0 333	 — —	t_{cyc} ns
2	Enable lead time Master[2] Slave	 $t_{lead(m)}$ $t_{lead(s)}$	 — 250	 — —	 — 240	 — —	ns
3	Enable lag time Master[2] Slave	 $t_{lag(m)}$ $t_{lag(s)}$	 — 250	 — —	 — 240	 — —	ns
4	Clock (SCK) high time Master Slave	 $t_{w(SCKH)m}$ $t_{w(SCKH)s}$	 340 190	 — —	 227 127	 — —	ns
5	Clock (SCK) low time Master Slave	 $t_{w(SCKL)m}$ $t_{w(SCKL)s}$	 340 190	 — —	 227 127	 — —	ns
6	Data setup time (inputs) Master Slave	 $t_{su(m)}$ $t_{su(s)}$	 100 100	 — —	 100 100	 — —	ns
7	Data hold time (inputs) Master Slave	 $t_{h(m)}$ $t_{h(s)}$	 100 100	 — —	 100 100	 — —	ns
8	Access time (time to data active from high-impedance state) Slave	t_a	0	120	0	120	ns
9	Disable time (hold time to high-impedance state) Slave	t_{dis}	—	240	—	167	ns
10	Data valid (after enable edge)[3]	$t_{v(s)}$	—	240	—	167	ns
11	Data hold time (outputs) (after enable edge)	t_{ho}	0	—	0	—	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	 — —	 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	 — —	 100 2.0	 ns µs

1. V_{DD} = 5.0 Vdc ± 10%, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H. All timing is shown with respect to 20% V_{DD} and 70% V_{DD}, unless otherwise noted.
2. Signal production depends on software.
3. Assumes 200 pF load on all SPI pins.



Note: Not defined but normally MSB of character just received

Figure 9-13. SPI Slave Timing (CPHA = 0)



Note: Not defined but normally LSB of character previously transmitted

Figure 9-14. SPI Slave Timing (CPHA = 1)

A.2 Block Diagram

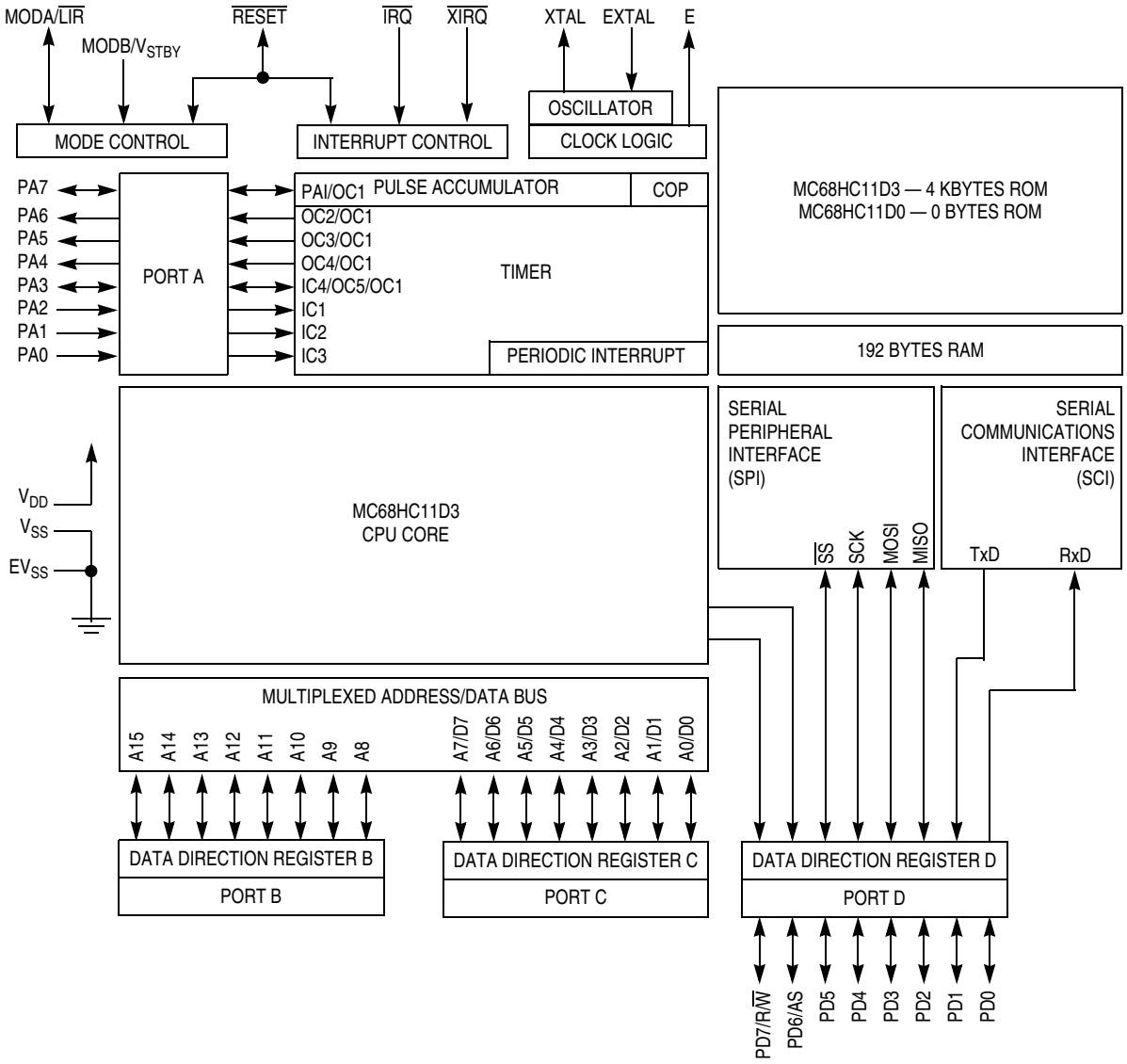


Figure A-1. MC68HC11D3 Block Diagram

A.4 Memory Map

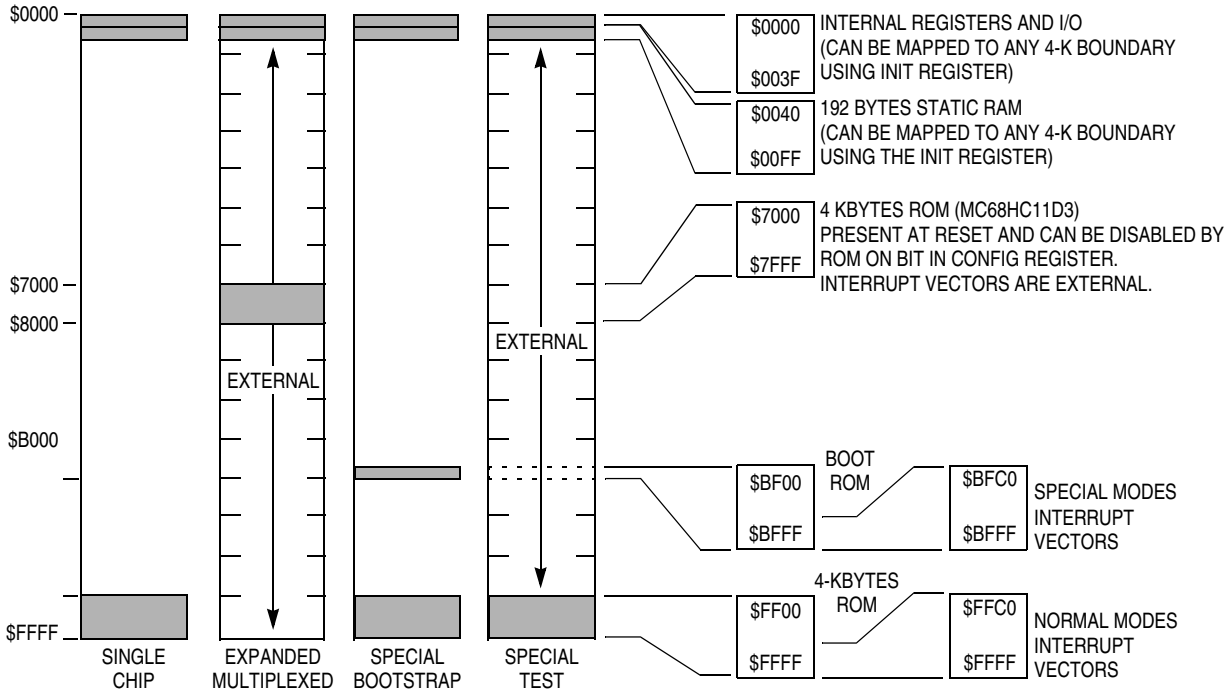


Figure A-4. MC68HC11Dx(1) Memory Map

A.5 MC68HC11D3 and MC68HC11D0 Electrical Characteristics

The parameters given in Chapter 9 Electrical Characteristics apply to the MC68HC11D3 and MC68HC11D0 with the exceptions given here.

A.5.1 Functional Operating Temperature Range

Rating	Symbol	Value	Unit
Operating temperature range MC68HC11D0C	T_A	T_L to T_H –40 to +85	°C

A.5.2 Thermal Characteristics

Characteristic	Symbol	Value	Unit
Package thermal resistance (junction-to-ambient) 44-pin plastic leaded chip carrier (PLCC) 44-pin plastic quad flat pack (QFP	Θ_{JA}	 50 85	°C/W

1. MC68HC11D0 only operates in expanded multiplexed mode and bootstrap mode.

Characteristic(1)	Symbol	Min	Max	Unit
Input high voltage All inputs except $\overline{RESET}$ $\overline{RESET}$	V_{IH}	0.7 x V_{DD} 0.8 x V_{DD}	V_{DD} + 0.3 V_{DD} + 0.3	V
Input low voltage All inputs	V_{IL}	V_{SS} – 0.3	0.2 x V_{DD}	V
I/O ports, three-state leakage PA7, PA3, PC7–PC0, V_{In} = V_{IH} or V_{IL} PD7–PD0, MODA/$\overline{LIR}$, $\overline{RESET}$	I_{OZ}	—	±10	μA
Input leakage current V_{In} = V_{DD} or V_{SS} PA2–PA0, $\overline{IRQ}$, $\overline{XIRQ}$ V_{In} = V_{DD} or V_{SS} MODB/V_{STBY}	I_{In}	 — —	 ±1 ±10	μA
RAM standby voltagePower down	V_{SB}	2.0	V_{DD}	V
RAM standby currentPower down	I_{SB}	—	10	μA
Input capacitancePA2–PA0, $\overline{IRQ}$, $\overline{XIRQ}$, EXTAL PA3, PA7, PC7–PC0, PD7–PD0, MODA/$\overline{LIR}$, $\overline{RESET}$	C_{In}	— —	8 12	pF
Output load capacitanceAll outputs except PD4–PD1 PD4–PD1	C_L	— —	90 100	pF
Total supply current(3) RUN: Single-chip mode V_{DD} = 5.5 V V_{DD} = 3.0 V Expanded multiplexed mode V_{DD} = 5.5 V V_{DD} = 3.0 V WAIT — All peripheral functions shut down: Single-chip mode V_{DD} = 5.5 V V_{DD} = 3.0 V Expanded multiplexed mode V_{DD} = 5.5 V V_{DD} = 3.0 V STOP — No clocks, single-chip mode: V_{DD} = 5.5 V V_{DD} = 3.0 V	I_{DD} WI_{DD} SI_{DD}	 8 4 14 7 3 1.5 5 2.5 50 25	 15 8 27 14 6 3 10 5 50 25	mA mA μA
Power dissipation Single-chip mode V_{DD} = 5.5 V V_{DD} = 3.0 V Expanded multiplexed mode V_{DD} = 5.5 V V_{DD} = 3.0 V	P_D	 44 12 77 21	 85 24 150 42	mW

1. V_{DD} = 3.0 Vdc to 5.5 Vdc, V_{SS} = 0 Vdc, T_A = T_L to T_H, unless otherwise noted.
2. V_{OH} specification for $\overline{RESET}$ and MODA is not applicable because they are open-drain pins. V_{OH} specification is not applicable to ports C and D in wired-OR mode.
3. EXTAL is driven with a square wave, and
   t_{cyc} = 1000 ns for 1 MHz rating;
   t_{cyc} = 500 ns for 2 MHz rating;
   $V_{IL} \leq 0.2$ V;
   $V_{IH} \geq V_{DD} - 0.2$ V;
   No dc loads

B.2.3 Control Timing

Characteristic[1]	Symbol	1.0 MHz		2.0 MHz		Unit
		Min	Max	Min	Max	
Frequency of operation	f_O	dc	1.0	dc	2.0	MHz
E-clock period	t_{cyc}	1000	—	500	—	ns
Crystal frequency	f_{XTAL}	—	4.0	—	8.0	MHz
External oscillator frequency	4 f_O	dc	4.0	dc	8.0	MHz
Processor control setup time t_{PCSU} = 1/4 t_{cyc} + 50 ns	t_{PCSU}	325	—	200	—	ns
Reset input pulse width[2] To guarantee external reset vector Minimum input time can be preempted by internal reset	PW_{RSTL}	8 1	— —	8 1	— —	t_{cyc}
Interrupt pulse width, PW_{IRQ} = t_{cyc} + 20 ns $\overline{IRQ}$ edge-sensitive mode	PW_{IRQ}	1020	—	520	—	ns
Wait recovery startup time	t_{WRS}	—	4	—	4	t_{cyc}
Timer pulse width PW_{TIM} = t_{cyc} + 20 ns Input capture pulse accumulator input	PW_{TIM}	1020	—	520	—	ns

1. V_{DD} = 3.0 Vdc to 5.5 Vdc, V_{SS} = 0 Vdc, T_A = T_L to T_H. All timing is shown with respect to 20% V_{DD} and 70% V_{DD}, unless otherwise noted.
2. Reset is recognized during the first clock cycle it is held low. Internal circuitry then drives the pin low for four clock cycles, releases the pin, and samples the pin level two cycles later to determine the source of the interrupt. Refer to Chapter 4 Resets, Interrupts, and Low-Power Modes for further details.

B.2.4 Peripheral Port Timing

Characteristic[1]	Symbol	1.0 MHz		2.0 MHz		Unit
		Min	Max	Min	Max	
Frequency of operation (E-clock frequency)	f_O	dc	1.0	dc	2.0	MHz
E-clock period	t_{cyc}	1000	—	500	—	ns
Peripheral data setup time[2] MCU read of ports A, B, C, and D	t_{PDSU}	100	—	100	—	ns
Peripheral data hold time[2] MCU read of ports A, B, C, and D	t_{PDH}	50	—	50	—	ns
Delay time, peripheral data write MCU write to port A MCU writes to ports B, C, and D t_{PWD} = 1/4 t_{cyc} + 150 ns	t_{PWD}	— —	200 350	— —	200 225	ns

1. V_{DD} = 3.0 Vd to 5.5 Vdc, V_{SS} = 0 Vdc, T_A = T_L to T_H. All timing is shown with respect to 20% V_{DD} and 70% V_{DD}, unless otherwise noted.
2. Port C and D timing is valid for active drive (CWOM and DWOM bits not set in PIOC and SPCR registers respectively).