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	2MHz
Connectivity	SCI, SPI
Peripherals	POR, WDT
Number of I/O	30
Program Memory Size	-
Program Memory Type	ROMless
EEPROM Size	512 x 8
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	4.75V ~ 5.25V
Data Converters	A/D 8x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	68-LCC (J-Lead)
Supplier Device Package	68-PLCC (24.21x24.21)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mchc11f1cfne2

TABLE OF CONTENTS
(Continued)

2.5 Four Times E-Clock Frequency Output (4XOUT)

Although the circuit shown in **Figure 2-6** will work for any M68HC11 MCU, the MC68HC11F1 has an additional clock output that is four times the E-clock frequency. This output (4XOUT) can be used to directly drive the EXTAL input of another M68HC11 MCU. Refer to **Figure 2-7**. The 4XOUT output is enabled after reset and can be disabled by clearing the CLK4X bit in the OPT2 register.

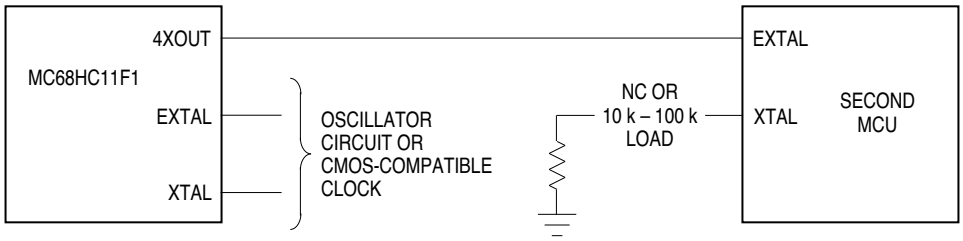

Figure 2-7 4XOUT Signal Driving a Second MCU

2.6 Interrupt Request ($\overline{\text{IRQ}}$)

The $\overline{\text{IRQ}}$ input provides a means of generating asynchronous interrupt requests for the CPU. Either falling-edge triggering or low-level triggering is selected by the IRQE bit in the OPTION register. $\overline{\text{IRQ}}$ is always configured for level-sensitive triggering at reset. Connect an external pull-up resistor, typically 4.7 kΩ, to V_{DD} when $\overline{\text{IRQ}}$ is used in a level-sensitive wired-OR configuration. Refer to **SECTION 5 RESETS AND INTERRUPTS**.

2.7 Non-Maskable Interrupt ($\overline{\text{XIRQ}}$)

The $\overline{\text{XIRQ}}$ input provides a means of requesting a non-maskable interrupt after reset initialization. During reset, the X bit in the condition code register (CCR) is set and any interrupt is masked until MCU software enables it. Because the $\overline{\text{XIRQ}}$ input is level sensitive, it can be connected to a multiple-source wired-OR network with an external pull-up resistor to V_{DD}. $\overline{\text{XIRQ}}$ is often used as a power loss detect interrupt.

Whenever $\overline{\text{XIRQ}}$ or $\overline{\text{IRQ}}$ are used with multiple interrupt sources ($\overline{\text{IRQ}}$ must be configured for level-sensitive operation if there is more than one source of $\overline{\text{IRQ}}$ interrupt), each source must drive the interrupt input with an open-drain type of driver to avoid contention between outputs. There should be a single pull-up resistor near the MCU interrupt input pin (typically 4.7 kΩ). There must also be an interlock mechanism at each interrupt source so that the source holds the interrupt line low until the MCU recognizes and acknowledges the interrupt request. If one or more interrupt sources are still pending after the MCU services a request, the interrupt line will still be held low and the MCU will be interrupted again as soon as the interrupt mask bit in the condition code register (CCR) is cleared (normally upon return from an interrupt). Refer to **SECTION 5 RESETS AND INTERRUPTS**.

2.8 MODA and MODB (MODA/$\overline{LIR}$ and MODB/V_{STBY})

During reset, MODA and MODB select one of the four operating modes. Refer to **SECTION 4 OPERATING MODES AND ON-CHIP MEMORY**.

After the operating mode has been selected, the $\overline{LIR}$ pin provides an open-drain output to indicate that execution of an instruction has begun. The LIR pin is configured for wired-OR operation (only pulls low). A series of E-clock cycles occurs during execution of each instruction. The $\overline{LIR}$ signal is asserted (drives low) during the first E-clock cycle of each instruction (opcode fetch). This output is provided for assistance in program debugging.

The V_{STBY} pin is used to input RAM standby power. The MCU is powered from the V_{DD} signal unless the difference between the level of V_{STBY} and V_{dd} is greater than one MOS threshold (about 0.7 volts). When these voltages differ by more than 0.7 volts, the internal 768-byte RAM and part of the reset logic are powered from V_{STBY} rather than V_{DD}. This allows RAM contents to be retained without V_{DD} power applied to the MCU. Reset must be driven low before V_{DD} is removed and must remain low until V_{DD} has been restored to a valid level.

2.9 V_{RH} and V_{RL}

These pins provide the reference voltage for the analog-to-digital converter. Bypass capacitors should be used to minimize noise on these signals. Any noise on V_{RH} and V_{RL} will directly affect A/D accuracy.

2.10 R/$\overline{W}$

In expanded and test modes, R/$\overline{W}$ indicates the direction of transfers on the external data bus. A logic level one on this pin indicates that a read cycle is in progress. A logic zero on this pin indicates that a write cycle is in progress and that no external device should drive the data bus.

The E-clock can be used to enable external devices to drive data onto the data bus during the second half of a read bus cycle (E clock high). R/$\overline{W}$ can then be used to control the direction of data transfers. R/$\overline{W}$ drives low when data is being written to the external data bus. R/$\overline{W}$ will remain low during consecutive data bus write cycles, such as when a double-byte store occurs.

2.11 Port Signals

For the MC68HC11F1, 54 pins are arranged into six 8-bit ports: A, B, C, E, F, and G, and one 6-bit port (D). Each of these seven ports serves a purpose other than I/O, depending on the operating mode or peripheral functions selected. Note that ports B, C, and F are available for I/O functions only in single-chip and bootstrap modes. The pins of ports A, C, D, and G are fully bidirectional. Ports B and F are output-only ports. Port E is an input-only port. Refer to **Table 2-1** for details about the 54 port signals' functions within different operating modes.

4.3.2 Initialization

Because bits in the following registers control the basic configuration of the MCU, an accidental change of their values could cause serious system problems. The protection mechanism, overridden in special operating modes, requires a write to the protected bits only within the first 64 bus cycles after any reset, or only once after each reset. **Table 4-2** summarizes the write access limited registers.

4.3.2.1 CONFIG Register

CONFIG controls the presence and position of the EEPROM in the memory map. CONFIG also enables the COP watchdog timer.

CONFIG — System Configuration Register **$103F**

	Bit 7	6	5	4	3	2	1	Bit 0	
	EE3	EE2	EE1	EE0	—	NOCOP	—	EEON	
RESET:	1	1	1	1	1	P	1	1	Single Chip
	1	1	1	1	1	P(L)	1	1	Bootstrap
	P	P	P	P	1	P	1	P	Expanded
	P	P	P	P	1	P(L)	1	0	Special Test

P indicates a previously programmed bit. P(L) indicates that the bit resets to the logic level held in the latch prior to reset, but the function of COP is controlled by DISR bit in TEST1 register.

The CONFIG register consists of an EEPROM byte and static latches that control the start-up configuration of the MCU. The contents of the EEPROM byte are transferred into static working latches during reset sequences. The operation of the MCU is controlled directly by these latches and not by CONFIG itself. In normal modes, changes to CONFIG do not affect operation of the MCU until after the next reset sequence. When programming, the CONFIG register itself is accessed. When the CONFIG register is read, the static latches are accessed.

These bits can be read at any time. The value read is the one latched into the register from the EEPROM cells during the last reset sequence. A new value programmed into this register cannot be read until after a subsequent reset sequence. Unused bits always read as ones.

In special test mode, the static latches can be written directly at any time. In all modes, CONFIG bits can only be programmed using the EEPROM programming sequence, and are neither readable nor active until latched via the next reset. Refer to **4.4.3 CONFIG Register Programming**.

EE[3:0] — EEPROM Mapping Control

EE[3:0] select the upper four bits of the EEPROM base address. In single-chip and bootstrap modes, EEPROM is forced to $FE00–$FFFF regardless of the value of EE[3:0].

Table 4-4 EEPROM Mapping

EE[3:0]	EEPROM Position
0 0 0 0	$0E00 – $0FFF
0 0 0 1	$1E00 – $1FFF
0 0 1 0	$2E00 – $2FFF
0 0 1 1	$3E00 – $3FFF
0 1 0 0	$4E00 – $4FFF
0 1 0 1	$5E00 – $5FFF
0 1 1 0	$6E00 – $6FFF
0 1 1 1	$7E00 – $7FFF
1 0 0 0	$8E00 – $8FFF
1 0 0 1	$9E00 – $9FFF
1 0 1 0	$AE00 – $AFFF
1 0 1 1	$BE00 – $BFFF
1 1 0 0	$CE00 – $CFFF
1 1 0 1	$DE00 – $DFFF
1 1 1 0	$EE00 – $EFFF
1 1 1 1	$FE00 – $FFFF

Bit 3 — Not implemented
Always reads one

NOCOP — COP System Disable
0 = COP system enabled (forces reset on time-out)
1 = COP system disabled

Bit 1 — Not implemented
Always reads one

EEON — EEPROM Enable
In single-chip modes EEON is forced to one (EEPROM enabled).
0 = 512 bytes of EEPROM is disabled from the memory map
1 = 512 bytes of EEPROM is present in the memory map

4.3.2.2 INIT Register

The internal registers used to control the operation of the MCU can be relocated on 4-Kbyte boundaries within the memory space with the use of INIT. This 8-bit special-purpose register can change the default locations of the RAM and control registers within the MCU memory map. It can be written only once within the first 64 E-clock cycles after a reset. It then becomes a read-only register.

INIT — RAM and I/O Mapping Register **$103D**

	Bit 7	6	5	4	3	2	1	Bit 0
	RAM3	RAM2	RAM1	RAM0	REG3	REG2	REG1	REG0
RESET:	0	0	0	0	0	0	0	1

Table 4-10 General-Purpose Chip Select Starting Address

Size (Bytes)	Valid Starting Address Bits
0 K (Disabled)	None
1 K	GA[15:10]
2 K	GA[15:11]
4 K	GA[15:12]
8 K	GA[15:13]
16 K	GA[15:14]
32 K	GA15
64 K	None

CSGSIZ — General-Purpose Chip Select Size Control **$105F**

	Bit 7	6	5	4	3	2	1	Bit 0
	IO1AV	IO2AV	—	GNPOL	GAVLD	GSIZA	GSIZB	GSIZC
RESET:	0	0	0	0	0	1	1	1

IO1AV — I/O Chip Select 1 Address Valid
0 = I/O chip select 1 is active during E-clock valid time (E-clock high)
1 = I/O chip select 1 is active during address valid time

IO2AV — I/O Chip Select 2 Address Valid
0 = I/O chip select 1 is active during E-clock valid time (E-clock high)
1 = I/O chip select 1 is active during address valid time

GNPOL — General-Purpose Chip Select Polarity Select
0 = CSGEN is active low
1 = CSGEN is active high

GAVLD — General-Purpose Chip Select Address Valid Select
0 = CSGEN is valid during E-clock valid time (E-clock high)
1 = CSGEN is valid during address valid time

G1SZA–G1SZC — General-Purpose Chip Select Size
Refer to **Table 4-11**.

Table 4-11 General-Purpose Chip Select Size Control

GSIZA	GSIZB	GSIZC	Size (Bytes)
0	0	0	64 K
0	0	1	32 K
0	1	0	16 K
0	1	1	8 K
1	0	0	4 K
1	0	1	2 K
1	1	0	1 K
1	1	1	0 K (Disabled)

The illegal opcode trap mechanism works for all unimplemented opcodes on all four opcode map pages. The address stacked as the return address for the illegal opcode interrupt is the address of the first byte of the illegal opcode. Otherwise, it would be almost impossible to determine whether the illegal opcode had been one or two bytes. The stacked return address can be used as a pointer to the illegal opcode so the illegal opcode service routine can evaluate the offending opcode.

5.4.4 Software Interrupt

SWI is an instruction, and thus cannot be interrupted until complete. SWI is not inhibited by the global mask bits in the CCR. Because execution of SWI sets the I mask bit, once an SWI interrupt begins, other interrupts are inhibited until SWI is complete, or until user software clears the I bit in the CCR.

5.4.5 Maskable Interrupts

The maskable interrupt structure of the MCU can be extended to include additional external interrupt sources through the $\overline{\text{IRQ}}$ pin. The default configuration of this pin is a low-level sensitive wired-OR network. When an event triggers an interrupt, a software accessible interrupt flag is set. When enabled, this flag causes a constant request for interrupt service. After the flag is cleared, the service request is released.

5.4.6 Reset and Interrupt Processing

Figure 5-1 and **Figure 5-3** illustrate the reset and interrupt process. **Figure 5-1** illustrates how the CPU begins from a reset and how interrupt detection relates to normal opcode fetches. **Figure 5-3** is an expansion of a block in **Figure 5-1** and illustrates interrupt priorities. **Figure 5-5** shows the resolution of interrupt sources within the SCI subsystem.

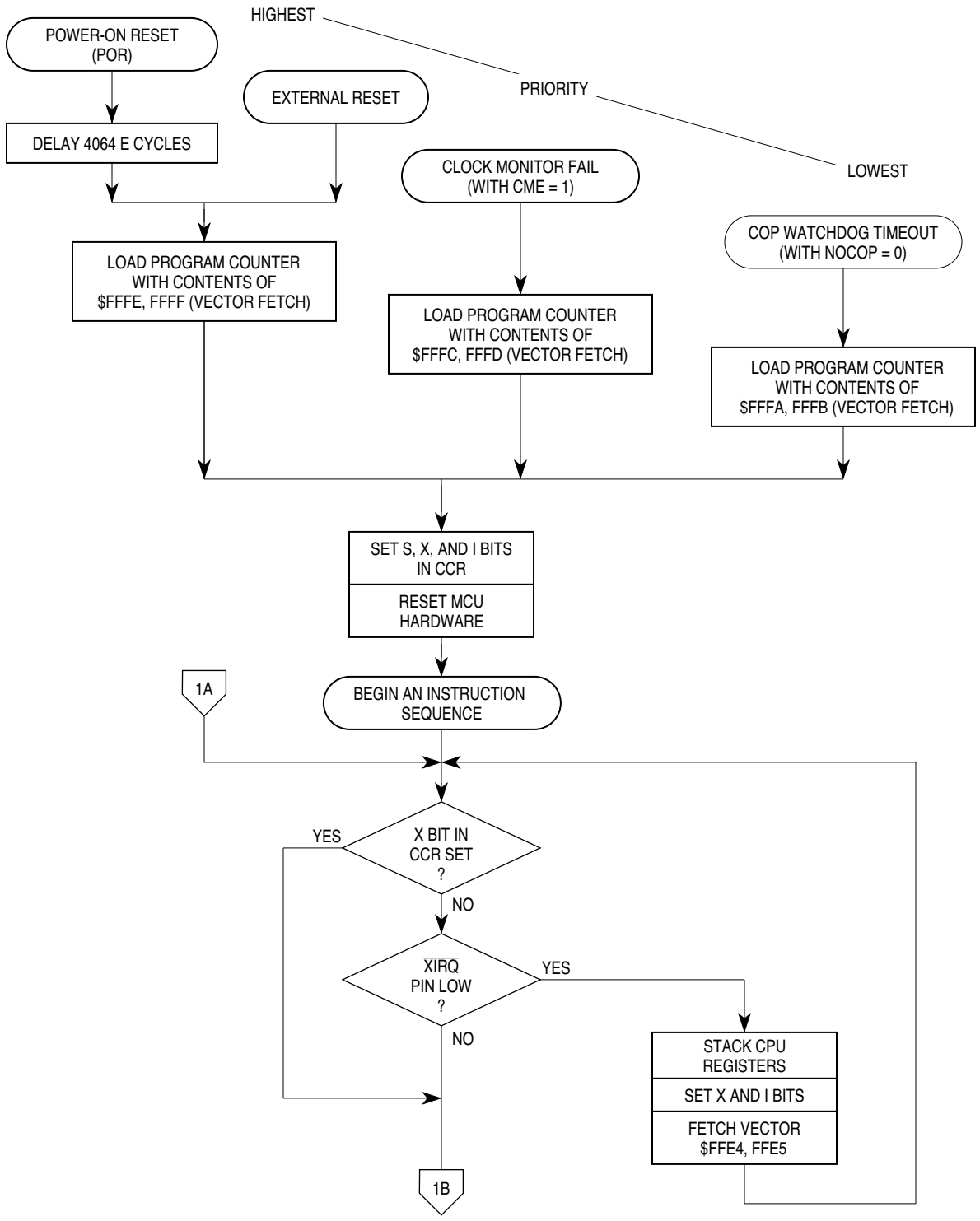

Figure 5-1 Processing Flow Out of Reset (1 of 2)

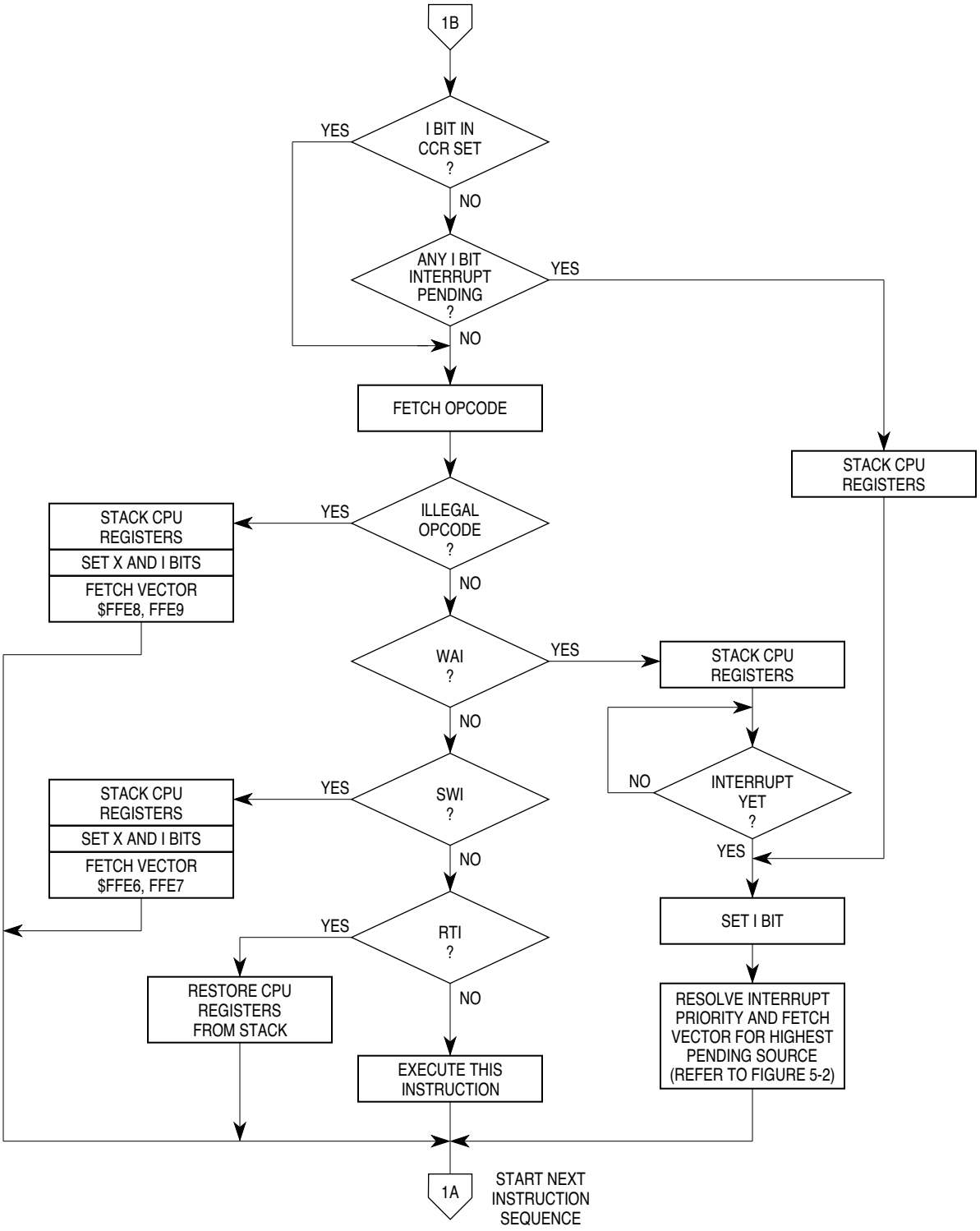

Figure 5-2 Processing Flow Out of Reset (2 of 2)

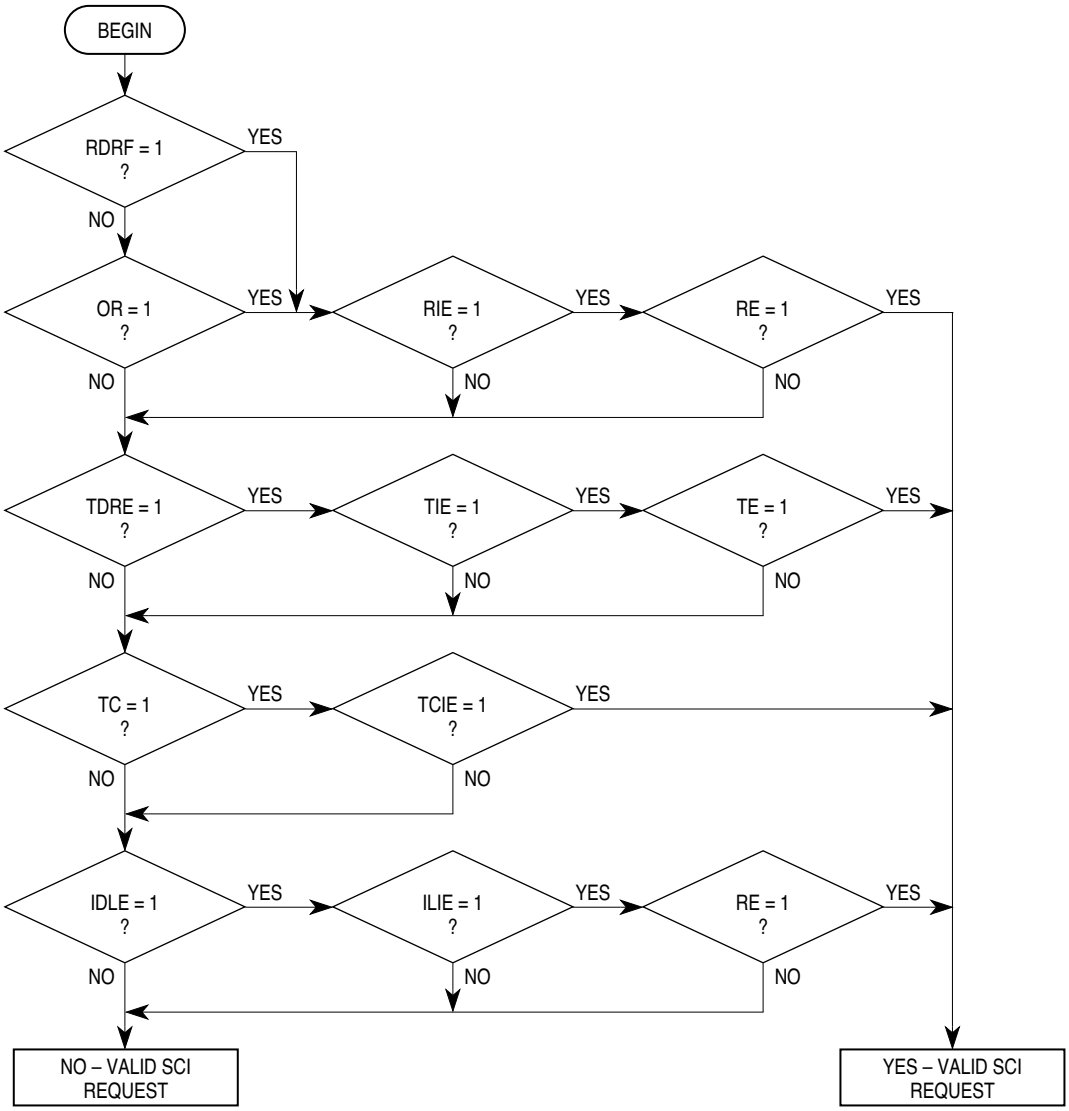

Figure 5-5 Interrupt Source Resolution Within SCI

5.5 Low Power Operation

Both STOP and WAIT suspend CPU operation until a reset or interrupt occurs. The WAIT condition suspends processing and reduces power consumption to an intermediate level. The STOP condition turns off all on-chip clocks and reduces power consumption to an absolute minimum while retaining the contents of all 1024 bytes of RAM.

impedance state, as neither the N-channel nor the P-channel devices are active. It is customary to have an external pull-up resistor on lines that are driven by open-drain devices. Port C can only be configured for wired-OR operation when the MCU is in single-chip or bootstrap modes.

PORTC — Port C Data **$1006**

	Bit 7	6	5	4	3	2	1	Bit 0
	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
S. Chip or Boot:	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
RESET:	I	I	I	I	I	I	I	I
Expan. or Test:	DATA7	DATA6	DATA5	DATA4	DATA3	DATA2	DATA1	DATA0

DDRC — Data Direction Register for Port C **$1007**

	Bit 7	6	5	4	3	2	1	Bit 0
	DDC7	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0
RESET:	0	0	0	0	0	0	0	0

DDC[7:0] — Data Direction for Port C
0 = Input
1 = Output

6.4 Port D

In all modes, port D bits [5:0] can be used either for general-purpose I/O, or with the SCI and SPI subsystems. During reset, port D pins are configured as high impedance inputs (DDRD bits cleared).

The DWOM control bit in the SPCR register disables port D's P-channel output drivers. Because the N-channel driver is not affected by DWOM, setting DWOM causes port D to become an open-drain-type output port suitable for wired-OR operation. In wired-OR mode, (PORTD bits are at logic level zero), pins are actively driven low by the N-channel driver. When a port D bit is at logic level one, the associated pin is in a high-impedance state, as neither the N-channel nor the P-channel devices are active. It is customary to have an external pull-up resistor on lines that are driven by open-drain devices. Port D can be configured for wired-OR operation in any operating mode.

PORTD — Port D Data **$1008**

	Bit 7	6	5	4	3	2	1	Bit 0
	—	—	PD5	PD4	PD3	PD2	PD1	PD0
RESET:	0	0	I	I	I	I	I	I
Alt. Pin Func.:	—	—	$\overline{SS}$	SCK	MOSI	MISO	TxD	RxD

R8 — Receive Data Bit 8
If M bit is set, R8 stores the ninth bit in the receive data character.

T8 — Transmit Data Bit 8
If M bit is set, T8 stores the ninth bit in the transmit data character.

M — Mode (Select Character Format)
0 = Start bit, 8 data bits, 1 stop bit
1 = Start bit, 9 data bits, 1 stop bit

WAKE — Wakeup by Address Mark/Idle
0 = Wakeup by IDLE line recognition
1 = Wakeup by address mark (most significant data bit set)

7.6.3 Serial Communications Control Register 2

The SCCR2 register provides the control bits that enable or disable individual SCI functions.

SCCR2 — SCI Control Register 2 **$102D**

	Bit 7	6	5	4	3	2	1	Bit 0
	TIE	TCIE	RIE	ILIE	TE	RE	RWU	SBK
RESET:	0	0	0	0	0	0	0	0

TIE — Transmit Interrupt Enable
0 = TDRE interrupts disabled
1 = SCI interrupt requested when TDRE status flag is set

TCIE — Transmit Complete Interrupt Enable
0 = TC interrupts disabled
1 = SCI interrupt requested when TC status flag is set

RIE — Receiver Interrupt Enable
0 = RDRF and OR interrupts disabled
1 = SCI interrupt requested when RDRF flag or the OR status flag is set

ILIE — Idle-Line Interrupt Enable
0 = IDLE interrupts disabled
1 = SCI interrupt requested when IDLE status flag is set

TE — Transmitter Enable
When TE goes from zero to one, one unit of idle character time (logic one) is queued as a preamble.
0 = Transmitter disabled
1 = Transmitter enabled

RE — Receiver Enable
0 = Receiver disabled
1 = Receiver enabled

8.3.1 Master In Slave Out

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.

8.3.2 Master Out Slave In

The MOSI line is the second of the two unidirectional serial data signals. It is an output from a master device and an input to a slave device. The master device places data on the MOSI line a half-cycle before the clock edge that the slave device uses to latch the data.

8.3.3 Serial Clock

SCK, an input to a slave device, is generated by the master device and synchronizes data movement in and out of the device through the MOSI and MISO lines. Master and slave devices are capable of exchanging a byte of information during a sequence of eight clock cycles.

There are four possible timing relationships that can be chosen by using control bits CPOL and CPHA in the serial peripheral control register (SPCR). Both master and slave devices must operate with the same timing. The SPI clock rate select bits, SPR[1:0], in the SPCR of the master device, select the clock rate. In a slave device, SPR[1:0] have no effect on the operation of the SPI.

8.3.4 Slave Select

The slave select ($\overline{SS}$) input of a slave device must be externally asserted before a master device can exchange data with the slave device. $\overline{SS}$ must be low before data transactions and must stay low for the duration of the transaction.

The $\overline{SS}$ line of the master must be held high. If it goes low, a mode fault error flag (MODF) is set in the serial peripheral status register (SPSR). To disable the mode fault circuit, write a one in bit 5 of the port D data direction register. This sets the $\overline{SS}$ pin to act as a general-purpose output rather than the dedicated input to the slave select circuit, thus inhibiting the mode fault flag. The other three lines are dedicated to the SPI whenever the serial peripheral interface is on.

The state of the master and slave CPHA bits affects the operation of $\overline{SS}$. CPHA settings should be identical for master and slave. When CPHA = 0, the shift clock is the OR of $\overline{SS}$ with SCK. In this clock phase mode, $\overline{SS}$ must go high between successive characters in an SPI message. When CPHA = 1, $\overline{SS}$ can be left low between successive SPI characters. In cases where there is only one SPI slave MCU, its $\overline{SS}$ line can be tied to V_{SS} as long as only CPHA = 1 clock mode is used.

8.4 SPI System Errors

Two system errors can be detected by the SPI system. The first type of error arises in a multiple-master system when more than one SPI device simultaneously tries to be a master. This error is called a mode fault. The second type of error, write collision, indicates that an attempt was made to write data to the SPDR while a transfer was in progress.

Table 9-1 Timer Summary

Control Bits PR1, PR0	XTAL Frequencies				
	4.0 MHz	**8.0 MHz**	**12.0 MHz**	**16.0 MHz**	**Other Rates**
	1.0 MHz	**2.0 MHz**	**3.0 MHz**	**4.0 MHz**	**(E)**
	1000 ns	**500 ns**	**333 ns**	**250 ns**	**(1/E)**
	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	250 ns 16.384 ms	(1/E) (2^{16}/E)
0 1 1 count — overflow —	4.0 µs 262.14 ms	2.0 µs 131.07 ms	1.333 µs 87.381 ms	1.0 µs 65.536 ms	(4/E) (2^{18}/E)
1 0 1 count — overflow —	8.0 µs 524.29 ms	4.0 µs 262.14 ms	2.667 µs 174.76 ms	2.0 µs 131.07 ms	(8/E) (2^{19}/E)
1 1 1 count — overflow —	16.0 µs 1.049 s	8.0 µs 524.29 ms	5.333 µs 349.52 ms	4.0 µs 262.14 ms	(16/E) (2^{20}/E)

9.1 Timer Structure

Figure 9-2 shows the capture/compare system block diagram. The port A pin control block includes logic for timer functions and for general-purpose I/O. For pins PA3, 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 PA[3:0] can be read at any time (read PORTA register), even if the pin is being used for the input capture function. Pins PA[6:3] are used for either general-purpose I/O, or as output compare pins. 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 (OC[5:2]) is related to one of the port A output pins. Output compare one (OC1) has extra control logic, allowing it optional control of any combination of the PA[7:3] 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.

Table 9-4 RTI Rate Selection

RTR[1:0]	E = 1 MHz	E = 2 MHz	E = 3 MHz	E = 4 MHz	E = X MHz
0 0	8.192 ms	4.096 ms	2.731 ms	2.048 ms	$(2^{13}/E)$
0 1	16.384 ms	8.192 ms	5.461 ms	4.096 ms	$(2^{14}/E)$
1 0	32.768 ms	16.384 ms	10.923 ms	8.192 ms	$(2^{15}/E)$
1 1	65.536 ms	32.768 ms	21.845 ms	16.384 ms	$(2^{16}/E)$

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 time-outs 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 time-out, not from when RTIF is cleared.

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

9.4.1 Timer Interrupt Mask Register 2

This register contains the real-time interrupt enable bits.

TMSK2 — Timer Interrupt Mask Register 2 **$1024**

	Bit 7	6	5	4	3	2	1	Bit 0
	TOI	RTII	PAOVI	PAII	—	—	PR1	PR0
RESET:	0	0	0	0	0	0	0	0

TOI — Timer Overflow Interrupt Enable
0 = TOF interrupts disabled
1 = Interrupt requested when TOF is set to one

RTII — Real-Time Interrupt Enable
0 = RTIF interrupts disabled
1 = Interrupt requested when RTIF set to one

PAOVI — Pulse Accumulator Overflow Interrupt Enable
Refer to **9.6 Pulse Accumulator**.

PAII — Pulse Accumulator Input Edge
Refer to **9.6 Pulse Accumulator**.

PR[1:0] — Timer Prescaler Select
Refer to **Table 9-4**.

NOTE

Bits in TMSK2 correspond bit for bit with flag bits in TFLG2. Ones in TMSK2 enable the corresponding interrupt sources.

APPENDIX A ELECTRICAL CHARACTERISTICS

This appendix contains electrical parameters for the MC68HC11F1 microcontroller.

Table A-1 Maximum Ratings

Rating	Symbol	Value	Unit
Supply Voltage	V_{DD}	– 0.3 to + 7.0	V
Input Voltage	V_{in}	– 0.3 to + 7.0	V
Operating Temperature Range MC68HC11F1 MC68HC11F1C MC68HC11F1V MC68HC11F1M	T_A	T_L to T_H 0 to + 70 – 40 to + 85 – 40 to + 105 – 40 to + 125	°C
Storage Temperature Range	T_{stg}	– 55 to + 150	°C
Current Drain per Pin* Excluding V_{DD}, V_{SS}, AV_{DD}, V_{RH}, and V_{RL}	I_D	25	mA

*One pin at a time, observing maximum power dissipation limits.

Internal circuitry protects the inputs against damage caused by high static voltages or electric fields; however, normal precautions are necessary to avoid application of any voltage higher than maximum-rated voltages to this high-impedance circuit. Extended operation at the maximum ratings can adversely affect device reliability. Tying unused inputs to an appropriate logic voltage level (either GND or V_{DD}) enhances reliability of operation.

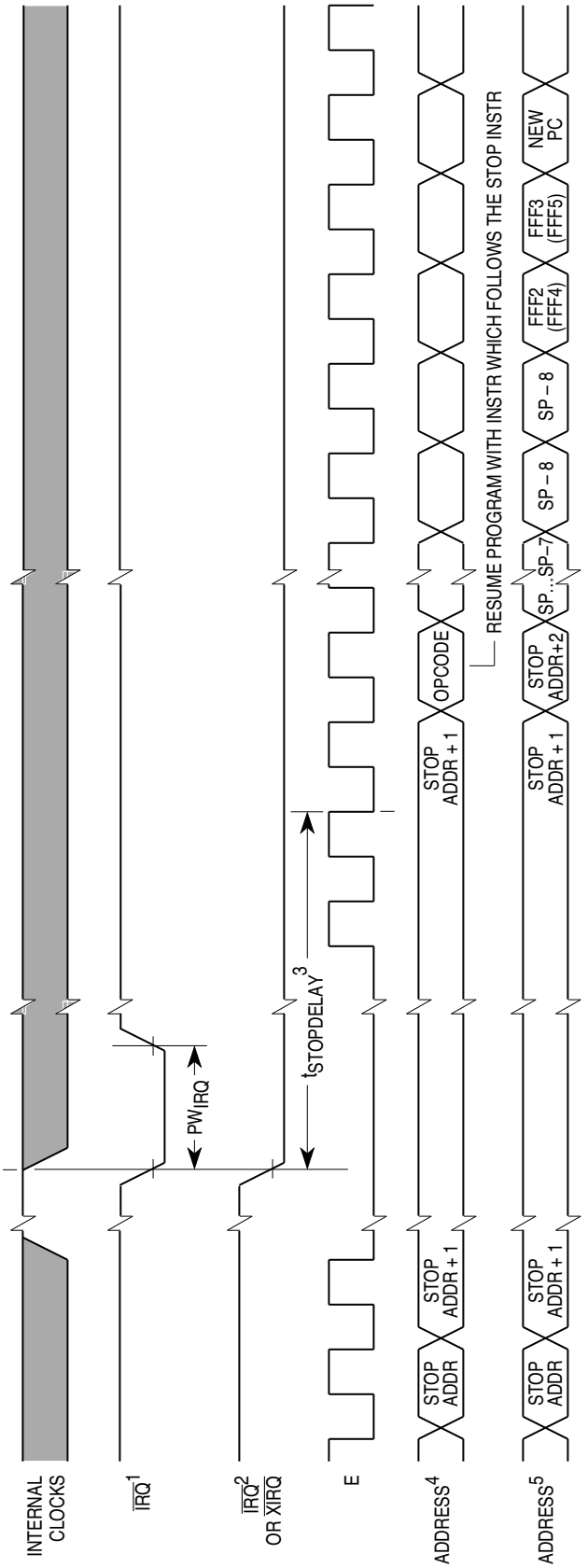

Figure A-4 STOP Recovery Timing Diagram

Table A-6 Analog-To-Digital Converter Characteristics

V_{DD} = 5.0 Vdc ± 5%, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H, 750 kHz ≤ E ≤ 3.0 MHz, unless otherwise noted

Characteristic	Parameter	Min	Absolute	2.0 MHz	3.0 MHz	4.0 MHz	Unit
				Max	Max	Max	
Resolution	Number of Bits Resolved by A/D Converter	—	8	—	—	—	Bits
Non-Linearity	Maximum Deviation from the Ideal A/D Transfer Characteristics	—	—	± 1	± 1	± 1	LSB
Zero Error	Difference Between the Output of an Ideal and an Actual for Zero Input Voltage	—	—	± 1	± 1	± 1	LSB
Full Scale Error	Difference Between the Output of an Ideal and an Actual A/D for Full-Scale Input Voltage	—	—	± 1	± 1	± 1	LSB
Total Unadjusted Error	Maximum Sum of Non-Linearity, Zero Error, and Full-Scale Error	—	—	± 1/2	± 1 1/2	± 1 1/2	LSB
Quantization Error	Uncertainty Because of Converter Resolution	—	—	± 1/2	± 1/2	± 1/2	LSB
Absolute Accuracy	Difference Between the Actual Input Voltage and the Full-Scale Weighted Equivalent of the Binary Output Code, All Error Sources Included	—	—	± 1	± 2	± 2	LSB
Conversion Range	Analog Input Voltage Range	V_{RL}	—	V_{RH}	V_{RH}	V_{RH}	V
V_{RH}	Maximum Analog Reference Voltage (Note 2)	V_{RL}	—	V_{DD} + 0.1	V_{DD} + 0.1	V_{DD} + 0.1	V
V_{RL}	Minimum Analog Reference Voltage (Note 2)	V_{SS} –0.1	—	V_{RH}	V_{RH}	V_{RH}	V
ΔV_R	Minimum Difference between V_{RH} and V_{RL} (Note 2)	3	—	—	—	—	V
Conversion Time	Total Time to Perform a Single Analog-to-Digital Conversion: E Clock Internal RC Oscillator	 — —	 32 —	 — t_{cyc} + 32	 — t_{cyc} + 32	 — t_{cyc} + 32	 t_{cyc} µs
Monotonicity	Conversion Result Never Decreases with an Increase in Input Voltage and has no Missing Codes		Guaranteed				
Zero Input Reading	Conversion Result when $V_{in} = V_{RL}$	00	—	—	—	—	Hex
Full Scale Reading	Conversion Result when $V_{in} = V_{RH}$	—	—	FF	FF	FF	Hex
Sample Acquisition Time	Analog Input Acquisition Sampling Time: E Clock Internal RC Oscillator	 — —	 12 —	 — 12	 — 12	 — 12	 t_{cyc} µs
Sample/Hold Capacitance	Input Capacitance during Sample PE[7:0]	—	20 (Typ)	—	—	—	pF
Input Leakage	Input Leakage on A/D Pins PE[7:0] V_{RL}, V_{RH}	— —	— —	400 1.0	400 1.0	400 1.0	nA µA

NOTES:

1. For f_{op} < 2 MHz, source impedances should equal approximately 10 kΩ. For f_{op} ≥ 2 MHz, source impedances should equal approximately 5 kΩ – 10 kΩ. Source impedances greater than 10 kΩ affect accuracy adversely because of input leakage.
2. Performance verified down to 2.5 V ΔV_R, but accuracy is tested and guaranteed at ΔV_R = 5 V ± 10%

Table A-8 Serial Peripheral Interface Timing

V_{DD} = 5.0 Vdc ± 5%, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H

Num	Characteristic	Symbol	2.0 MHz		3.0 MHz		4.0 MHz		Unit
			Min	Max	Min	Max	Min	Max	
	Operating Frequency Master Slave	 $f_{op(m)}$ $f_{op(s)}$	 dc dc	 1.0 2.0	 dc dc	 1.5 3.0	 dc dc	 2.0 4.0	 MHz MHz
1	Cycle Time Master Slave	 $t_{cyc(m)}$ $t_{cyc(s)}$	 2.0 500	 — —	 2.0 333	 — —	 2.0 250	 — —	 t_{cyc} ns
2	Enable Lead Time Master (Note 2) Slave	 $t_{lead(m)}$ $t_{lead(s)}$	 — 250	 — —	 — 240	 — —	 — 200	 — —	 ns ns
3	Enable Lag Time Master (Note 2) Slave	 $t_{lag(m)}$ $t_{lag(s)}$	 — 250	 — —	 — 240	 — —	 — 200	 — —	 ns ns
4	Clock (SCK) High Time Master Slave	 $t_{w(SCKH)m}$ $t_{w(SCKH)s}$	 340 190	 — —	 227 127	 — —	 130 85	 — —	 ns ns
5	Clock (SCK) Low Time Master Slave	 $t_{w(SCKL)m}$ $t_{w(SCKL)s}$	 340 190	 — —	 227 127	 — —	 130 85	 — —	 ns ns
6	Data Setup Time (Inputs) Master Slave	 $t_{su(m)}$ $t_{su(s)}$	 100 100	 — —	 100 100	 — —	 100 100	 — —	 ns ns
7	Data Hold Time (Inputs) Master Slave	 $t_{h(m)}$ $t_{h(s)}$	 100 100	 — —	 100 100	 — —	 100 100	 — —	 ns ns
8	Access Time (Time to Data Active from High-Impedance State) Slave	 t_a	 0	 120	 0	 120	 0	 120	 ns
9	Disable Time (Hold Time to High-Impedance State) Slave	 t_{dis}	 —	 240	 —	 167	 —	 125	 ns
10	Data Valid (After Enable Edge) (Note 3)	$t_{v(s)}$	—	240	—	167	—	125	ns
11	Data Hold Time (Outputs) (After Enable Edge)	t_{ho}	0	—	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	 — —	 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	 — —	 100 2.0	 ns µs

NOTES:
1. 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.

APPENDIX CDEVELOPMENT SUPPORT

C.1 MC68HC11F1 Development Tools

The following table and text provide a reference to development tools for the MC68HC11F1 microcontrollers. For more complete information refer to the appropriate manual for each system.

Table C-1 MC68HC11F1 Development Tools

Device	Evaluation Systems	Modular Development Systems
MC68HC11F1	M68HC11F1EVS	MMDS11*

* For MC68HC11F1 support, the MMDS11 must be used with an MC68HC11F1 emulator module.

C.2 MC68HC11EVS — Evaluation System

The EVS is an economical tool for designing, debugging, and evaluating target systems based on the MC68HC11F1 MCU. The two printed circuit boards that comprise the EVS are the MC68HC11F1EM emulator module (EM) and the M68HC11PFB platform board (PFB).

- Monitor/debugger firmware
- One-line assembler/disassembler
- Host computer download capability
- Dual memory maps:
  - 64 Kbyte monitor map that includes 16 Kbytes of monitor EPROM
  - MC68HC11F1 user map that includes 64 Kbytes of emulation RAM
- OTPROM, EPROM, and EEPROM MCU programmer
- MCU extension I/O port for single-chip, expanded, and special-test operation modes
- RS-232C terminal and host I/O ports
- Logic analyzer connector

C.3 M68MMDS11 — Modular Development System for M68HC11 Devices

The M68MMDS11 Freescale Modular Development System (MMDS11) is a tool for developing embedded systems based on M68HC11 MCUs. The MMDS11 is an emulator system that provides an on-screen bus state analyzer and real-time memory monitoring windows. An integrated design environment includes an editor, an assembler, the user interface, and source-level debug capability. These features significantly reduce the time necessary to develop and debug an embedded MCU system. The compact unit requires minimum space.

- Real-time, non-intrusive, in-circuit emulation
- Assembly-language source-level debugging