MC68HC11F1/D
REV 2

HC11

MC68HC11F1

TECHNICAL
DATA

MOTOROLA

MC68HC11F1
TECHNICAL DATA
MOTOROLA

MC68HC11F1

HIGH-DENSITY COMPLEMENTARY METAL OXIDE SEMICONDUCTOR (HCMOS) MICROCONTROLLER UNIT

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Concluded)

LIST OF ILLUSTRATIONS

LIST OF TABLES

SECTION 1
INTRODUCTION

This section depicts the general characteristics and special features of the MC68HC11F1 high-density complementary metal oxide semiconductor (HCMOS) microcontroller unit (MCU).

This document contains condensed information on the MC68HC11F1 MCU. For more detailed information, see M68HC11RM/AD, *M68HC11 Reference Manual* available at the local Motorola sales office.

1.1 THE MOTOROLA MC68HC11F1 MCU

The MC68HC11F1 MCU contains highly sophisticated on-chip peripheral functions. This high-speed, low-power MCU has a nonmultiplexed bus with a nominal bus speed of 2 MHz. The fully static design allows operations at frequencies down to dc.

1.2 SPECIAL FEATURES

Refer to Figure 1-1 and the following list for hardware and software features of the MC68HC11F1:

- Expanded 16-Bit Timer System with Four-Stage Programmable Prescaler
- Enhanced Nonreturn-to-Zero (NRZ) Serial Communications Interface (SCI)
- Eight-Channel 8-Bit Analog-to-Digital (A/D) Converter
- Block Protect Mechanism for EEPROM and CONFIG
- Nonmultiplexed Expanded Bus
- 68-Pin Packaging
- Power-Saving STOP and WAIT Modes
- 64K Memory Addressability

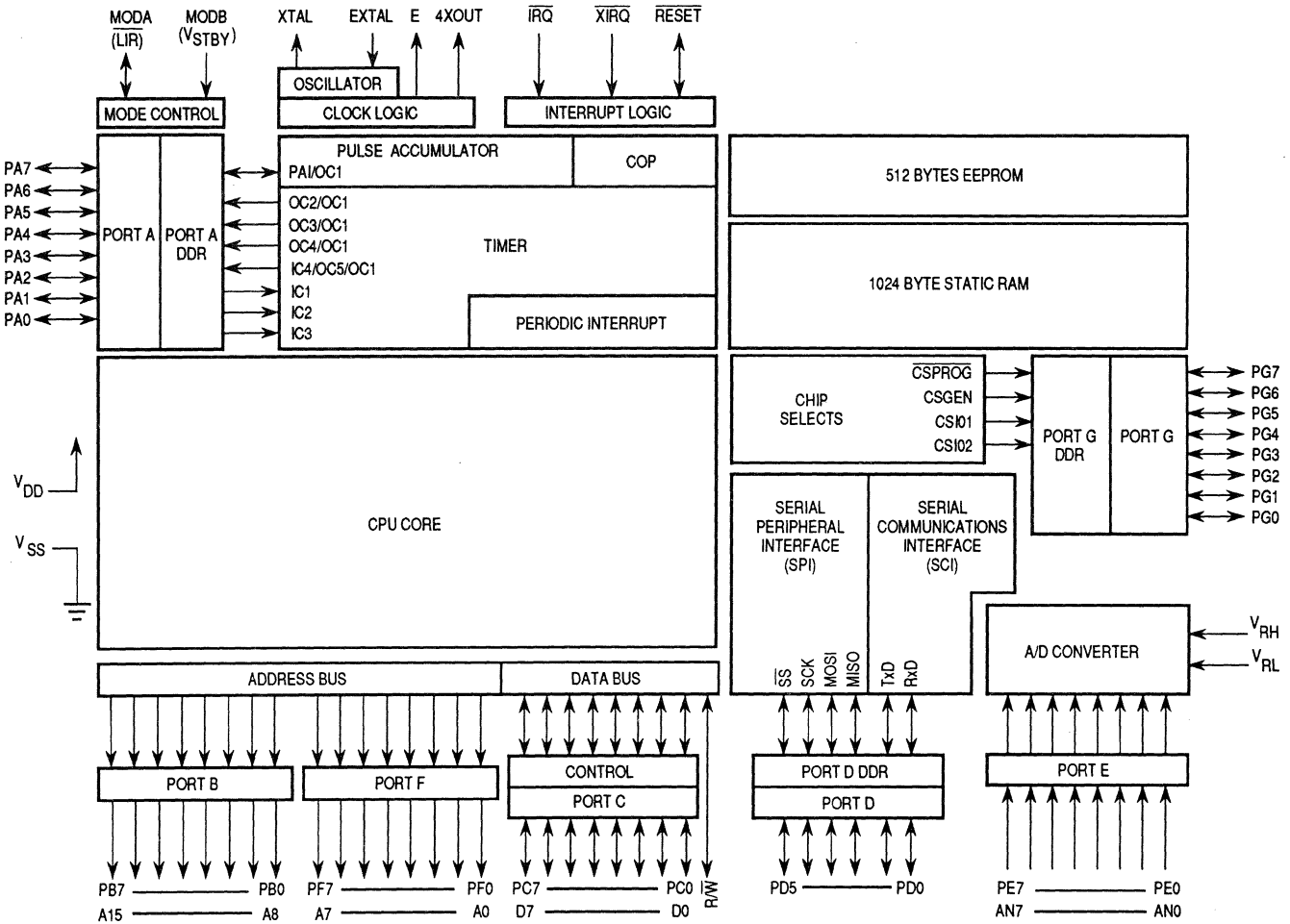


Figure 1-1. Block Diagram

Features (Continued)

- Serial Peripheral Interface (SPI)
- 512 Bytes of EEPROM
- 8-Bit Pulse Accumulator Circuit
- 1024 Bytes of Static RAM (All Saved During Standby)
- Bit Test and Branch Instructions
- Real-Time Interrupt Circuit
- Four Programmable Chip Selects
- Computer Operating Properly (COP) Watchdog System

SECTION 2
OPERATING MODES AND SIGNAL DESCRIPTIONS

This section describes the operating modes and signals of the MC68HC11F1 MCU.

2.1 OPERATING MODES

Although it is intended to operate principally in expanded mode, the MC68HC11F1 uses two dedicated pins, MODA and MODB, to select one of two normal operating modes or one of two special operating modes. The normal operating modes are the single-chip and expanded-nonmultiplexed modes. The special operating modes are the bootstrap and test modes. Mode selection according to the values encoded on the mode select pins (MODA and MODB) is shown in the following table:

MODA	MODB	Mode Selected
0	1	Single Chip
1	1	Expanded Nonmultiplexed
0	0	Special Bootstrap
1	0	Special Test

2.1.1 Single-Chip Mode

In the single-chip mode, the MCU functions as a self-contained microcontroller and has no external address or data bus. The 512-byte EEPROM would contain all program code and is forced to $FE00–$FFFF. This mode provides maximum use of the pins for on-chip peripheral functions, and all the address and data activity occurs within the MCU.

2.1.2 Expanded-Nonmultiplexed Mode

In the expanded-nonmultiplexed mode, the MCU can address up to 64K bytes of address space. High-order address bits are output on the port B pins, and low-order address bits are output on port F. The bidirectional data bus appears on port C. The read/write (R/$\overline{W}$) pin is used to control the direction of

data transfer on the port C bus. Programmable chip selects are available on port G pins, PG7–PG4.

2.1.3 Bootstrap Mode

This special mode is similar to single-chip mode. The resident bootloader program allows a variable length program to be loaded into on-chip RAM through 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 2-1) so that the user can set up a jump table, if desired.

Table 2-1. 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
00E2	Timer Input Capture 3
00E5	Timer Input Capture 2
00E8	Timer Input Capture 1
00EB	Real-Time Interrupt
00EE	$\overline{\text{IRQ}}$
00F1	$\overline{\text{XIRQ}}$
00F4	SWI
00F7	Illegal Opcode
00FA	COP Fail
00FD	Clock Monitor
BF00 (Boot)	Reset

This versatile mode can be used for test and diagnostic functions on completed modules and for programming the on-chip EEPROM. 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 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 EEPROM at $FE00.

2.1.4 Test Mode

This special expanded mode is primarily intended for production testing. However, it can be used to program calibration or personality data into the internal EEPROM. The 512-byte EEPROM is initially turned off in this mode. The user can access a number of special test control bits. Reset and interrupt vectors are fetched externally from locations $BFC0–$BFFF. A switch can be made from this mode to other modes under program control.

2.2 SIGNAL DESCRIPTION

The following paragraphs describe the signals necessary to the various functions of the MCU.

2.2.1 V_{DD} and V_{SS}

Power is supplied to the MCU using these two pins. V_{DD} is power (+5 V ±10%), and V_{SS} is ground (0 V).

2.2.2 $\overline{\text{RESET}}$

This active-low bidirectional control pin is used as an input to initialize the MCU to a known startup state. It is also used as an open-drain output to indicate that an internal failure has been detected in either the clock monitor or in the COP circuit.

2.2.3 XTAL and EXTAL

These pins provide the interface for either a crystal or a CMOS-compatible clock to control the internal clock generator circuitry. The frequency applied must be four times higher than the desired clock rate. Refer to Figure 2-1 for crystal and clock connections.

2.2.4 E Clock

This pin provides an output for the internally generated E clock, which can be used for timing reference. The frequency of the E-clock output is one-fourth that of the input frequency at the XTAL and EXTAL pins.

2.2.5 4XOUT

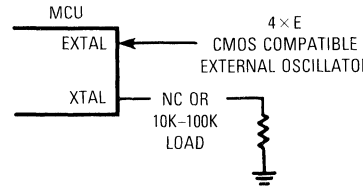
This pin, which provides an output for the 4X buffered clock (4 times the E-clock frequency), can be used to drive the clock input of another processor. This output is enabled out of reset and can be disabled by clearing the CLK4X bit of the OPT2 register. CLK4X is writable one time out of each reset.

2.2.6 $\overline{\text{IRQ}}$

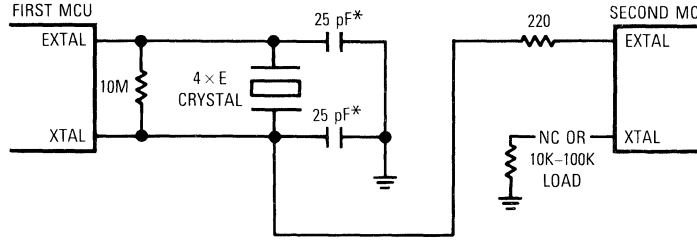
The $\overline{\text{IRQ}}$ pin provides the capability for asynchronously applying interrupts to the MCU. Either negative edge-sensitive triggering or level-sensitive triggering is program selectable by using the IRQE bit of the OPTION register. This pin is configured as level sensitive during reset. An external resistor connected to V_{DD} is typically required on $\overline{\text{IRQ}}$.

2.2.7 $\overline{\text{XIRQ}}$

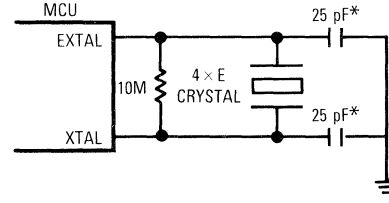
The $\overline{\text{XIRQ}}$ pin provides the capability for asynchronously applying non-maskable interrupts to the MCU. During reset, the X bit in the condition code register is set, masking any interrupt until enabled by software. This level-sensitive input typically requires an external pullup resistor to V_{DD}.



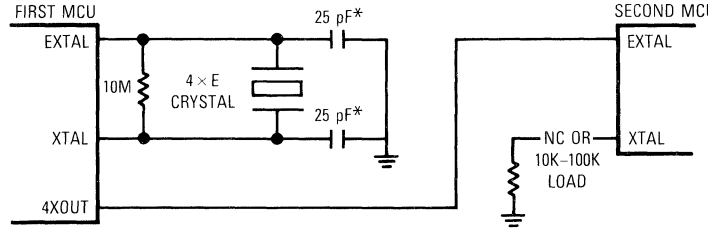
(a) External Oscillator Connections



(b) One Crystal Driving Two MCUs



(c) Common Crystal Connections



(d) 4XOUT Driving Second MCU

*Values include all stray capacitances

Figure 2-1. Oscillator Connections

2.2.8 MODA/$\overline{LIR}$ and MODB/V_{STBY}

During reset, these pins are used to implement the two normal or two special modes of operation. The $\overline{LIR}$ output can be used as an aid to debugging once reset is completed. The open-drain $\overline{LIR}$ pin goes to an active low during the first E-clock cycle of each instruction and remains low for the duration of that cycle. The V_{STBY} input is used to retain RAM contents during powerdown.

2.2.9 V_{RL} and V_{RH}

These pins provide the reference voltage for the A/D converter.

2.2.10 R/$\overline{W}$

The R/$\overline{W}$ output is used to control the direction of transfers on the external data bus in expanded-nonmultiplexed mode. A low on this pin indicates that data is being written to the external data bus. A high on this pin indicates that a read cycle is in progress. R/$\overline{W}$ stays high during single-chip and bootstrap modes.

2.2.11 INPUT/OUTPUT LINES (PA7–PA0, PB7–PB0, PC7–PC0, PD5–PD0, PE7–PE0, PF7–PF0, PG7–PG0)

The 54 input/output (I/O) lines are arranged into six 8-bit ports (ports A, B, C, E, F, and G) and one 6-bit port (port D). Most of these ports serve more than one purpose, depending on the operating mode or peripheral functions selected. Table 2-2 shows the functions of each port and line as the operating mode changes.

Table 2-2. Port Signal Functions (Sheet 1 of 2)

Port	Bit	Single-Chip and Bootstrap Modes	Expanded-Nonmultiplexed and Special Test Modes
A	0	PA0/IC3	PA0/IC3
A	1	PA1/IC2	PA1/IC2
A	2	PA2/IC1	PA2/IC1
A	3	PA3/IC4/OC5 (and/or OC1)	PA3/IC4/OC5 (and/or OC1)
A	4	PA4/OC4 (and/or OC1)	PA4/OC4 (and/or OC1)
A	5	PA5/OC3 (and/or OC1)	PA5/OC3 (and/or OC1)
A	6	PA6/OC2 (and/or OC1)	PA6/OC2 (and/or OC1)
A	7	PA7/PAI (and/or OC1)	PA7/PAI (and/or OC1)

Table 2-2. Port Signal Functions (Sheet 2 of 2)

Port	Bit	Single-Chip and Bootstrap Modes	Expanded-Nonmultiplexed and Special Test Modes
B	0	PB0	A8
B	1	PB1	A9
B	2	PB2	A10
B	3	PB3	A11
B	4	PB4	A12
B	5	PB5	A13
B	6	PB6	A14
B	7	PB7	A15
C	0	PC0	D0
C	1	PC1	D1
C	2	PC2	D2
C	3	PC3	D3
C	4	PC4	D4
C	5	PC5	D5
C	6	PC6	D6
C	7	PC7	D7
D	0	PD0/RxD	PD0/RxD
D	1	PD1/TxD	PD1/TxD
D	2	PD2/MISO	PD2/MISO
D	3	PD3/MOSI	PD3/MOSI
D	4	PD4/$\overline{\text{SCK}}$	PD4/$\overline{\text{SCK}}$
D	5	PD5/$\overline{\text{SS}}$	PD5/$\overline{\text{SS}}$
E	0	PE0/AN0	PE0/AN0
E	1	PE1/AN1	PE1/AN1
E	2	PE2/AN2	PE2/AN2
E	3	PE3/AN3	PE3/AN3
E	4	PE4/AN4	PE4/AN4
E	5	PE5/AN5	PE5/AN5
E	6	PE6/AN6	PE6/AN6
E	7	PE7/AN7	PE7/AN7
F	0	PF0	A0
F	1	PF1	A1
F	2	PF2	A2
F	3	PF3	A3
F	4	PF4	A4
F	5	PF5	A5
F	6	PF6	A6
F	7	PF7	A7
G	0	PG0	PG0
G	1	PG1	PG1
G	2	PG2	PG2
G	3	PG3	PG3
G	4	PG4	PG4/CSIO2
G	5	PG5	PG5/CSIO1
G	6	PG6	PG6/$\overline{\text{CSGEN}}$
G	7	PG7	PG7/$\overline{\text{CSPROG}}$

SECTION 3
MEMORY AND CONTROL AND STATUS REGISTERS

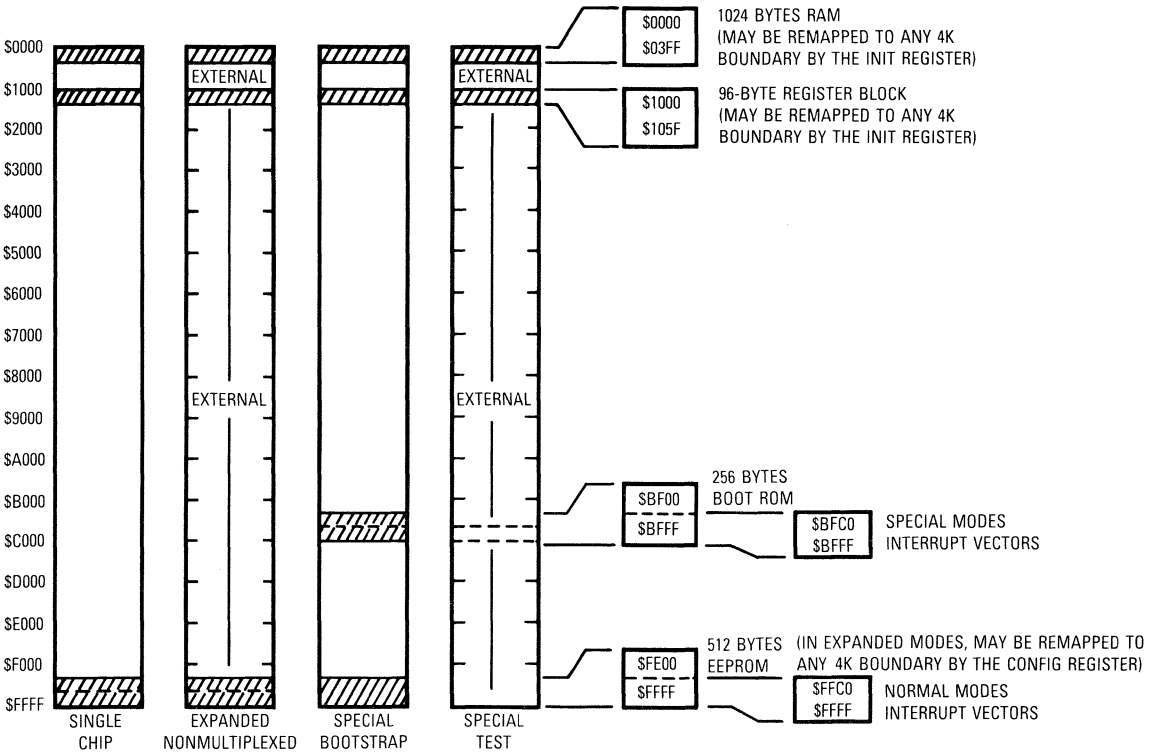
This section describes the memory, memory subsystems mapping, and the mapping of the control and status registers of the MC68HC11F1 MCU.

3.1 MEMORY

Figure 3-1 illustrates the memory map for all four modes of operation: single chip, expanded nonmultiplexed, special bootstrap, and special test. Since this chip is intended to operate principally in expanded mode, there is no internal ROM and the bus is nonmultiplexed. Memory consists mainly of 64K of external-memory-addressing capability available to the user. On-chip, there are 1K of static RAM, 512 bytes of EEPROM, and 96 bytes of status and control registers, all of which are mappable to any 4K boundary in memory. In addition, 256 bytes of bootloader ROM are present only in special bootstrap mode.

3.2 MEMORY SUBSYSTEMS MAPPING

Using the INIT register, the 96-byte control and status register block and the 1K of static RAM are mappable to any 4K boundary in memory. However, reset locates the RAM from $0000–$03FF and register space from $**1**000–$**1**05F, where **1** represents the decoded value of the four low-order bits of the INIT register. The EEPROM is enabled by the EEON bit of the CONFIG register. In expanded-nonmultiplexed and special-test modes, it is located from $xE00–$xFFF (where x represents the value of the four high-order bits of the CONFIG register). In single chip and bootstrap modes, the EEPROM is located from $FE00–$FFFF. Bootstrap ROM is mapped to location $BF00–$BFFF upon transition to bootstrap mode. Should mapping conflicts arise, the register block takes priority over RAM, and the bootstrap ROM has priority over EEPROM.



NOTE: The EEPROM can be enabled in special test mode by writing a one to the EEON bit of the CONFIG register after reset.

Figure 3-1. Memory Map

3.3 CONTROL AND STATUS REGISTERS

There are 96 bytes of status and control registers that are used to control the operation of the MCU. The registers can be relocated to any 4K boundary in memory, but default to location $1000-$105F after reset. To indicate this remappability and default, the address of any remappable register begins with a bold 1. Figure 3-2 is a complete listing of the registers and reserved locations that comprise the control and status register block in memory.

3.4 RAM AND I/O MAPPING REGISTER (INIT)

The INIT register is a special-purpose 8-bit register that is used during initialization to change the default locations of RAM and control registers within the MCU memory map. It can be written to only once within the first 64 E-clock cycles after a reset in normal modes. Thereafter, it becomes a read-only register.

	7	6	5	4	3	2	1	0	
$103D	RAM3	RAM2	RAM1	RAM0	REG3	REG2	REG1	REG0	INIT
RESET:	0	0	0	0	0	0	0	1	

Since the INIT register is set to $01 by reset, the default starting address for RAM is $0000 and $1000 for the control and status registers. RAM3–RAM0 (bits 7–4) specify the starting address for the 1K of RAM. REG3–REG0 (INIT bits 3–0) specify the starting address for the control and status register block. In each case, the four RAM or REG bits become the four most significant bits of the 16-bit address of the RAM or register being written. The remaining twelve bits of the 16-bit address express the final three characters of the address.

Throughout this document, control and status register addresses are displayed with the high-order digit shown as a bold numeral 1, which indicates that the register block may be relocated to some 4K memory page other than its default position of $1000.

RAM and the control and status registers can be relocated independently. If the control and status registers are relocated in such a way as to conflict with RAM, then the register block takes priority, and the RAM at those locations becomes inaccessible. No harmful conflicts result. Lower priority

$1000	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0	PORTA	I/O Port A
$1001	DDA7	DDA6	DDA5	DDA4	DDA3	DDA2	DDA1	DDA0	DDRA	Data Direction for Port A
$1002	PG7	PG6	PG5	PG4	PG3	PG2	PG1	PG0	PORTG	I/O Port G
$1003	DDG7	DDG6	DDG5	DDG4	DDG3	DDG2	DDG1	DDG0	DDRG	Data Direction for Port G
$1004	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0	PORTB	I/O Port B
$1005	PF7	PF6	PF5	PF4	PF3	PF2	PF1	PF0	PORTF	I/O Port F
$1006	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0	PORTC	I/O Port C
$1007	DDC7	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0	DDRC	Data Direction for Port C
$1008	0	0	PD5	PD4	PD3	PD2	PD1	PD0	PORTD	I/O Port D
$1009	0	0	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	DDRD	Data Direction for Port D
$100A	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0	PORTE	I/O Port E
$100B	FOC1	FOC2	FOC3	FOC4	FOC5	0	0	0	CFORC	Compare Force Register
$100C	OC1M7	OC1M6	OC1M5	OC1M4	OC1M3	0	0	0	OC1M	OC1 Action Mask Register
$100D	OC1D7	OC1D6	OC1D5	OC1D4	OC1D3	0	0	0	OC1D	OC1 Action Data Register
$100E	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	TCNT	Timer Counter Register
$100F	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0		
$1010	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	TIC1	Input Capture 1 Register
$1011	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0		
$1012	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	TIC2	Input Capture 2 Register
$1013	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0		
$1014	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	TIC3	Input Capture 3 Register
$1015	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0		
$1016	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	TOC1	Output Compare 1 Register
$1017	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0		
$1018	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	TOC2	Output Compare 2 Register
$1019	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0		
$101A	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	TOC3	Output Compare 3 Register
$101B	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0		
$101C	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	TOC4	Output Compare 4 Register
$101D	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0		
$101E	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	TI4O5	Input Compare 4/Output
$101F	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0		Capture 5 Register
$1020	OM2	OL2	OM3	OL3	OM4	OL4	OM5	OL5	TCTL1	Timer Control Register 1
$1021	EDG4B	EDG4A	EDG1B	EDG1A	EDG2B	EDG2A	EDG3B	EDG3A	TCTL2	Timer Control Register 2
$1022	OC1I	OC2I	OC3I	OC4I	I4O5I	IC1I	IC2I	IC3I	TMSK1	Timer Interrupt Mask Reg. 1
$1023	OC1F	OC2F	OC3F	OC4F	I4O5F	IC1F	IC2F	IC3F	TFLG1	Timer Interrupt Flag Reg. 1
$1024	TOI	RTII	PAOVI	PAII	0	0	PR1	PR0	TMSK2	Timer Interrupt Mask Reg. 2

Figure 3-2. Control and Status Registers (Sheet 1 of 2)

$1025	TOF	RTIF	PAOVF	PAIF	0	0	0	0	TFLG2	Timer Interrupt Flag Reg. 2
$1026	0	PAEN	PAMOD	PEDGE	0	I4/O5	RTR1	RTR0	PACTL	Pulse Accum. Control Reg.
$1027	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	PACNT	Pulse Accum. Count Reg.
$1028	SPIE	SPE	DWOM	MSTR	CPOL	CPHA	SPR1	SPR0	SPCR	SPI Control Register
$1029	SPIF	WCOL	0	MODF	0	0	0	0	SPSR	SPI Status Register
$102A	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	SPDR	SPI Data Register
$102B	TCLR	0	SCP1	SCP0	RCKB	SCR2	SCR1	SCR0	BAUD	SCI Baud Rate Control
$102C	R8	T8	0	M	WAKE	0	0	0	SCCR1	SCI Control Register 1
$102D	TIE	TCIE	RIE	ILIE	TE	RE	RWU	SBK	SCCR2	SCI Control Register 2
$102E	TDRE	TC	RDRF	IDLE	OR	NF	FE	0	SCSR	SCI Status Register
$102F	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	SCDR	SCI Data (Read RDR, Write TDR)
$1030	CCF	0	SCAN	MULT	CD	CC	CB	CA	ADCTL	A/D Control Register
$1031	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	ADR1	A/D Result Register 1
$1032	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	ADR2	A/D Result Register 2
$1033	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	ADR3	A/D Result Register 3
$1034	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	ADR4	A/D Result Register 4
$1035	0	0	0	PTCON	BPRT3	BPRT2	BPRT1	BPRT0	BPROT	EEPROM Block Protect Reg.
$1036									Reserved	
$1037									Reserved	
$1038	GWOM	CWOM	CLK4X	—	—	—	—	—	OPT2	System Configuration Options 2 Reg.
$1039	ADPU	CSEL	IRQE	DLY	CME	FCME	CR1	CR0	OPTION	System Configuration Options
$103A	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	COPRST	Arm/Reset COP Timer Circuitry
$103B	ODD	EVEN	0	BYTE	ROW	ERASE	EELAT	EEPGM	PPROG	EEPROM Prog.Control Reg.
$103C	RBOOT	SMOD	MDA	IRV	PSEL3	PSEL2	PSEL1	PSEL0	HPRIO	Highest Priority I-Bit Interrupt and Misc.
$103D	RAM3	RAM2	RAM1	RAM0	REG3	REG2	REG1	REG0	INIT	RAM and I/O Mapping Reg.
$103E	TILOP	—	OCCR	CBYP	DISR	FCM	FCOP	—	TEST1	Factory Test Control Register
$103F	EE3	EE2	EE1	EE0	1	NOCOP	1	EEON	CONFIG	COP, ROM, and EEPROM Enables
$1040 TO $105B									Reserved	
$105C	IO1SA	IO1SB	IO2SA	IO2SB	GSTHA	GSTHB	PSTHA	PSTHB	CSSTRH	Chip Select Clock Stretch Reg.
$105D	IO1EN	IO1PL	IO2EN	IO2PL	GCSPR	PCSEN	PSIZA	PSIZB	CSCTL	Chip Select Control Register
$105E	GA15	GA14	GA13	GA12	GA11	GA10	—	—	CSGADR	General-Purpose Chip Select Address Reg.
$105F	IO1AV	IO2AV	—	GNPOL	GAVLD	GSIZA	GSIZB	GSIZC	CSGSIZ	General-Purpose Chip Select Size Reg.

Figure 3-2. Control and Status Registers (Sheet 2 of 2)

resources simply become inaccessible. Similarly, if an internal resource conflicts with an external device, no harmful conflict results. Data from the external device is not applied to the internal data bus, thus it cannot interfere with the internal read.

NOTE

There are unused register locations in the 96-byte control and status register block. Reads of these unused registers return data from the undriven internal data bus, not from another source that happens to be located at the same address.

SECTION 4
INPUT/OUTPUT PORTS

The MC68HC11F1 is equipped with six 8-bit I/O ports (A, B, C, E, F, and G) and one 6-bit I/O port (D). Ports B, C, F, and G I/O functions are controlled by the particular mode of operation selected. In the single-chip and bootstrap modes, they are configured as parallel I/O data ports. In expanded-nonmultiplexed and test modes, ports B, C, F, G, and pin R/$\overline{W}$ are configured as a memory expansion bus, with ports B and F as the address bus, port C as the data bus, the R/$\overline{W}$ pin as data bus direction control, and the upper four bits of port G as external chip selects.

The remaining ports are unaffected by mode changes. Ports A, D, and G can be used as general-purpose I/O ports, though each has an alternate function. Port E can be used for general-purpose static inputs and/or A/D converter channel inputs. Port A bits control the timer functions. Port D handles the SPI and SCI functions.

While exercising their general-purpose I/O function, ports A, C, D, and G are under the control of data direction registers (DDR) and port data registers (PORT) A, C, D, and G, respectively.

4.1 PORT A

Port A is an 8-bit general-purpose I/O port with both a data register (PORTA) and a data direction register (DDRA). In addition, port A can be configured for timer input capture (IC) functions, timer output compare (OC) functions, or the pulse accumulator function.

4.1.1 Port A Data Register (PORTA)

	7	6	5	4	3	2	1	0	
$1000	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0	PORTA
RESET:	0	0	0	0	0	0	0	0	
Alternate Pin Function:	PAI	OC2	OC3	OC4	OC5/IC4	IC1	IC2	IC3	
and/or:	OC1	OC1	OC1	OC1	OC1	—	—	—	

PORTA can be read any time. Inputs return the pin level, while outputs return the pin driver input level. If written, PORTA stores the data in an internal latch. It drives the pins only if they are configured as outputs. Writes to PORTA do not change the pin state when the pins are configured for timer output compares.

4.1.2 Port A Data Direction Register (DDRA)

	7	6	5	4	3	2	1	0	
$1001	DDA7	DDA6	DDA5	DDA4	DDA3	DDA2	DDA1	DDA0	DDRA
RESET:	0	0	0	0	0	0	0	0	

1 = Corresponding port A I/O pin is configured as output.
0 = Corresponding port A I/O pin is configured for input only.

The timer forces the I/O state to be an output for each port A bit associated with an enabled output compare. In such a case, the DDRA bits are not changed, but lose immediate control of the associated port A bit. The DDRA regains control of the I/O state of the pin once the associated timer output compare is disabled. Enabling an input capture function does not force the direction of the associated port A pin.

4.2 PORT B (PORTB)

In the single-chip mode, all port B pins are general-purpose output pins (PB7–PB0). In the expanded-nonmultiplexed mode, all of the port B pins act as high-order address bits (A15–A8) of the address bus, and accesses to port B are treated as external accesses.

	7	6	5	4	3	2	1	0	
$1004	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0	PORTB
RESET:	0	0	0	0	0	0	0	0	
Alternate Pin Function:	A15	A14	A13	A12	A11	A10	A9	A8	

While the MCU is operating in single chip or bootstrap mode, a read of port B returns the sensed levels at the inputs of port B pin drivers, while a write causes data to be stored in an internal latch which in turn drives the port B pins.

4.3 PORT C

Port C is an 8-bit, general-purpose I/O port with a data register (PORTC) and a data direction register (DDRC). In the single-chip mode, port C pins are general-purpose I/O pins (PC7–PC0). In the expanded-nonmultiplexed mode, port C is the data bus (D7–D0), and accesses to port C are treated as external accesses. Bidirectional data pins are controlled by the R/$\overline{W}$ signal in expanded modes. Port C can be configured for wired-OR operation in single-chip mode by setting the CWOM bit of the OPT2 register.

4.3.1 Port C Data Register (PORTC)

	7	6	5	4	3	2	1	0	
$1006	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0	PORTC
RESET:	0	0	0	0	0	0	0	0	
Alternate Pin Function:	D7	D6	D5	D4	D3	D2	D1	D0	

While the MCU is operating in single chip or bootstrap mode, PORTC can be read at any time. Inputs return the sensed level at the pin while outputs return the input level of the port C pin drivers. If PORTC is written, the data is stored in an internal latch, which in turn drives port C pins that are configured as outputs. If a port C pin is changed from an input to an output, the pin is driven to the value last written to the internal port C latch.

4.3.2 Port C Data Direction Register (DDRC)

	7	6	5	4	3	2	1	0	
$1007	DDC7	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0	DDRC
RESET:	0	0	0	0	0	0	0	0	

1 = Corresponding port C I/O pin is configured as output.
0 = Corresponding port C I/O pin is configured for input only.

4.4 PORT D

Port D is a 6-bit, general-purpose I/O port with a data register (PORTD) and a data direction register (DDRD). In all modes, the six port D bits (D5–D0) can be used for general-purpose I/O, or for the SCI and SPI subsystems. Port D can also be configured for wired-OR operation.

4.4.1 Port D Data Register (PORTD)

	7	6	5	4	3	2	1	0	
$1008	0	0	PD5	PD4	PD3	PD2	PD1	PD0	PORTD
RESET:	0	0	0	0	0	0	0	0	
Alternate Pin Function:	—	—	$\overline{SS}$	SCK	MOSI	MISO	TxD	RxD	

PORTD can be read at any time. Inputs return the sensed levels at the pin; while outputs return the input level of the port D pin drivers. Reads to bits 6 and 7 always return zeros. If PORTD is written, the data is stored in an internal latch, which in turn drives port D pins that are configured as outputs. Writes to bits 6 and 7 have no meaning or effect. This port shares functions with the on-chip SCI and SPI subsystems. If a port D pin is changed to a general purpose output by writing the corresponding bit of DDRD, or by disabling the SCI or SPI subsystems, the pin is driven to the value last written to the internal port D latch.

4.4.2 Port D Data Direction Register (DDRD)

	7	6	5	4	3	2	1	0	
$1009	0	0	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	DDRD
RESET:	0	0	0	0	0	0	0	0	

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

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

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:

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

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

4.5 PORT E (PORTE)

Port E is used for general-purpose static inputs (PE7–PE0) and/or analog-to-digital (A/D) channel inputs (AN7–AN0) in all operating modes. Port E should not be read as static inputs while an A/D conversion is occurring. Such a read can disturb a conversion that is in progress if it coincides with the sample portion of the conversion cycle.

	7	6	5	4	3	2	1	0	
$100A	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0	PORTE
RESET	U	U	U	U	U	U	U	U	
Alternate Pin Function:	AN7	AN6	AN5	AN4	AN3	AN2	AN1	AN0	

4.6 PORT F (PORTF)

In the single-chip mode, all port F pins are general-purpose output pins (PF7–PF0). In the expanded-nonmultiplexed mode, all of the port F pins act as the low-order address (A7–A0) of the address bus, and accesses to port F are treated as external accesses.

	7	6	5	4	3	2	1	0	
$1005	PF7	PF6	PF5	PF4	PF3	PF2	PF1	PF0	PORTF
RESET:	0	0	0	0	0	0	0	0	
Alternate Pin Function:	A7	A6	A5	A4	A3	A2	A1	A0	

While the MCU is operating in single chip or bootstrap mode, a read of port F returns the sensed levels at the inputs of port F pin drivers, while a write causes data to be stored in an internal latch which in turn drives the port F pins.

4.7 PORT G

Port G is an 8-bit, general-purpose I/O port with both a data register (PORTG) and a data direction register (DDRG). The upper four bits are optionally usable as chip-select outputs in expanded modes. When any of these lines is not being used for a chip select, it can be used as a general-purpose I/O. Port G has a wired-OR mode to facilitate testing.

4.7.1 Port G Data Register (PORTG)

	7	6	5	4	3	2	1	0	
$1002	PG7	PG6	PG5	PG4	PG3	PG2	PG1	PG0	PORTG
RESET:	0	0	0	0	0	0	0	0	
Alternate Pin Function:	CSPRG	CSGEN	CSIO1	CSIO2	—	—	—	—	

PORTG can be read at any time. Inputs return the sensed levels at the pin while outputs return the input level of the port G pin drivers. If PORTG is written, the data is stored in an internal latch, which in turn drives port G pins that are configured as outputs. Writes to port G do not change the pin state when the associated pin is configured as a chip select. If a port G pin is changed to a general purpose output by writing the corresponding bit of DDRG, or by disabling a chip select, the pin is driven to the value last written to the internal port G latch.

4.7.2 Port G Data Direction Register (DDRG)

	7	6	5	4	3	2	1	0	
$1003	DDG7	DDG6	DDG5	DDG4	DDG3	DDG2	DDG1	DDG0	DDRG
RESET:	0	0	0	0	0	0	0	0	

When port G is configured as general-purpose I/O, the DDRG register controls the direction of the I/O pins as follows:

1 = Corresponding port G I/O pin is configured as output.
0 = Corresponding port G I/O pin is configured for input only.

The chip selects force the I/O state to be an output for each port G line associated with an enabled chip select. In this case, the DDRG bits are not changed and have no effect on these lines. DDRG reverts to controlling the I/O state of a pin when the associated chip-select function is disabled.

4.8 SYSTEM CONFIGURATION OPTIONS 2 REGISTER (OPT2)

	7	6	5	4	3	2	1	0	
$1038	GWOM	CWOM	CLK4X	—	—	—	—	—	OPT2
RESET:	0	0	1	0	0	0	0	0	

GWOM — Port G Wired-OR Mode Option
This bit affects all port G pins together.
1 = Port G outputs act as open-drain outputs.
0 = Port G outputs are normal CMOS outputs.

CWOM — Port C Wired-OR Mode Option
This bit affects all port C pins together.
1 = Port C outputs act as open-drain outputs.
0 = Port C outputs are normal CMOS outputs.

CLK4X—4X Clock Output Enable
(This bit can be written at any time on mask number B77M devices.) This bit can only be written once after reset in normal modes (HPRIO register bit SMOD = 0).
1 = Output of 4XOUT clock is enabled.
0 = Output of 4XOUT clock is disabled.

Bits 4–0 — Not implemented

SECTION 5
CHIP SELECTS

The function of the chip selects is to eliminate the need for additional external components to interface with peripherals in expanded-nonmultiplexed mode. Such factors as polarity, address block size, and clock stretching are controlled using the chip-select registers.

There are four programmable chip selects on the MC68HC11F1: two for external I/O (CSIO1 and CSIO2), one for external program space ($\overline{\text{CSPROG}}$), and one general-purpose chip select (CSGEN). All may be enabled by the user via the chip-select control register (CSCTL), and are designed not to conflict with each other nor with internal memory.

The I/O chip selects are approximately 2K each and follow the internal register 4K block of the memory map. Chip select for external program space starts at the end of memory and continues toward the beginning of memory in selectable power-of-two increments, from 8K–64K. The general-purpose chip select has a programmable starting address, and its size is selectable from 1K–64K in power-of-two increments. The general-purpose and I/O chip selects are selectable to be active during E-clock valid time or during address valid time. Any of the four chip selects can be programmed to cause a clock stretch that will occur only during accesses to addresses within that chip select's address range.

5.1 PROGRAM CHIP SELECT ($\overline{\text{CSPROG}}$)

The external program space chip select ($\overline{\text{CSPROG}}$) is active low and active only during address valid time. $\overline{\text{CSPROG}}$ is enabled by the PCSEN bit of the chip-select control register (CSCTL), and its address block size is selected by the PSIZA and PSIZB bits of CSCTL. Its priority versus that of the general-purpose chip select is controlled by the GCSPR bit of the CSCTL register. Program chip select clock stretching is enabled with the PSTHA and PSTHB bits of the CSSTRH register.

5.2 I/O CHIP SELECTS (CSIO1, CSIO2)

The I/O chip selects (CSIO1 and CSIO2) are the chip selects for external I/O devices. The addresses for these chip selects fill the remainder of the memory map 4K block that contains the status and control registers. CSIO1 is mapped from $**1**060–$**1**7FF, and CSIO2 is mapped from $**1**800–$**1**FFF, where **1** is a character representing the value of the high-order nibble of the register block address. CSIO1 and CSIO2 are enabled, and their polarity is selected by the IO1EN, IO1PL, IO2EN, and IO2PL bits of the CSCTL register. The IO1AV and IO2AV bits of the CSGSIZ register determine whether the chip selects are valid during address or E-clock valid times. Clock stretching for the I/O chip selects is enabled with the IO1SA, IO1SB, IO2SA, and IO2SB bits of the CSSTRH register.

5.3 CHIP-SELECT CONTROL REGISTER (CSCTL)

	7	6	5	4	3	2	1	0	
$105D	IO1EN	IO1PL	IO2EN	IO2PL	GCSPR	PCSEN	PSIZA	PSIZB	CSCTL
RESET:	0	0	0	0	0	*	0	0	

IO1EN — Enable for I/O Chip-Select 1
1 = Chip select is enabled.
0 = Chip select is disabled.

IO1PL — Polarity Select for I/O Chip-Select 1
1 = Chip select is active high.
0 = Chip select is active low.

IO2EN — Enable for I/O Chip-Select 2
1 = Chip select is enabled.
0 = Chip select is disabled.

IO2PL — Polarity Select for I/O Chip-Select 2
1 = Chip select is active high.
0 = Chip select is active low.

GCSPR — General-Purpose Chip-Select Priority
1 = General-purpose chip select has priority.
0 = Program chip select has priority.

PCSEN — Enable for Program Chip Select

1 = Program chip select is enabled. Reset sets PCSEN in expanded-non-multiplexed mode.

0 = Program chip select is disabled. Reset clears PCSEN in single-chip mode.

PSIZA and PSIZB — Program Chip-Select Address Sizes

The value of these bits read together decodes as follows:

PSIZA	PSIZB	Address Size	Addresses
0	0	64K Bytes	$0000–$FFFF
0	1	32K Bytes	$8000–$FFFF
1	0	16K Bytes	$C000–$FFFF
1	1	8K Bytes	$E000–$FFFF

5.4 GENERAL-PURPOSE CHIP SELECT (CSGEN)

The general-purpose chip select is the most flexible of the four chip selects, having the most control bits associated with it. Polarity, address versus E-clock valid, and address block size are determined by the GNPOL, GAVLD, GSIZA, GSIZB, and GSIZC bits of the CSGSIZ register. Starting address is selected with the CSGADR register. CSGEN priority versus that of the program chip select is selected with the GCSPR bit of the CSCTL register. Clock stretching for the general-purpose chip select is enabled by the GSTHA and GSTHB bits of the CSSTRH register. Selection of an address size of zero can be used as a CSGEN disable. Reset produces this state.

5.4.1 General-Purpose Chip-Select Size Register (CSGSIZ)

	7	6	5	4	3	2	1	0	
$$05F	IO1AV	IO2AV	—	GNPOL	GAVLD	GSIZA	GSIZB	GSIZC	CSGSIZ
RESET:	0	0	0	0	0	1	1	1	

IO1AV — I/O Chip-Select 1 Address Valid
1 = IOCS1 is valid during address valid time.
0 = IOCS1 is valid during E-clock valid time (E clock high)

IO2AV — I/O Chip-Select 2 Address Valid
1 = IOCS2 is valid during address valid time.
0 = IOCS2 is valid during E-clock valid time (E clock high).

Bit 5 — Not implemented

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

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

GSIZA, GSIZB, and GSIZC — Address Size for CSGEN
The value of these bits read together decodes as follows:

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

5.4.2 General-Purpose Chip-Select Address Register (CSGADR)

	7	6	5	4	3	2	1	0	
$105E	GA15	GA14	GA13	GA12	GA11	GA10	—	—	CSGADR
RESET:	0	0	0	0	0	0	0	0	

GA15–GA10 — General-Purpose Chip-Select Starting Address
These bits correspond to the high-order address bits A15–A10. They determine the starting address of the CSGEN valid address space. The address size selected determines which of this register's bits are valid as follows:

Address Size Selected	CSGADR Bits Valid
0K Bytes	None
1K Bytes	GA15–GA10
2K Bytes	GA15–GA11
4K Bytes	GA15–GA12
8K Bytes	GA15–GA13
16K Bytes	GA15–GA14
32K Bytes	GA15
64K Bytes	None

Bits 1 and 0 — Not implemented

5.5 CLOCK STRETCHING (CSSTRH)

Each of the four chip selects is associated with two bits in the chip-select clock stretch register (CSSTRH). These bits allow clock stretching from zero to three cycles (full E-clock periods) for interfacing to slow devices. Any of the chip selects can be programmed to cause a clock stretch that occurs only during access to addresses that fall within that particular chip select's address range.

During the stretch period, the E clock is held high, and the bus remains in the state that it would normally be in at the end of E high time. Internally, the clocks continue to run, maintaining the integrity of the timers, baud-rate generators, etc.

CSSTRH contains an "A" and a "B" bit for each of the four chip selects, allowing a choice of four clock stretches for each chip select.

	7	6	5	4	3	2	1	0	
$105C	IO1SA	IO1SB	IO2SA	IO2SB	GSTHA	GSTHB	PSTHA	PSTHB	CSSTRH
RESET:	0	0	0	0	0	0	0	0	

IO1SA and IO1SB — I/O Chip-Select 1 Clock Delay

IO2SA and IO2SB — I/O Chip-Select 2 Clock Delay

GSTHA and GSTHB — General-Purpose Chip-Select Clock Delay

PSTHA and PSTHB — Program Chip-Select Clock Delay

The amount of clock stretching during each chip select is determined by decoding the value of bits A and B for each type of chip select as follows:

Bit A	Bit B	Clock Stretch
0	0	0 Cycles
0	1	1 Cycle
1	0	2 Cycles
1	1	3 Cycles

5.6 CHIP-SELECT PRIORITY

An important characteristic of the four chip selects is that they neither conflict with each other nor with internal memory and registers. The means established to ensure this facility is priority. There are two sets of priorities controlled by the value of the general-purpose chip-select priority (GCSPR) bit of the CSCTL register. The highest-to-lowest priority of on-chip memory, on-chip registers, and chip selects is shown in Table 5-1.

Table 5-1. Chip-Select Priority

GCSPR = 0	GCSPR = 1
On-Chip Registers	On-Chip Registers
On-Chip RAM	On-Chip RAM
Bootloader ROM	Bootloader ROM
On-Chip EEPROM	On-Chip EEPROM
I/O Chip Selects	I/O Chip Selects
Program Chip Select	General-Purpose Chip Select
General-Purpose Chip Select	Program Chip Select

SECTION 6
RESETS, INTERRUPTS, AND LOW POWER MODES

This section describes the internal and external resets and interrupts of the MC68HC11F1 MCU and its two low power-consumption modes.

6.1 RESETS

The MCU can be reset in four ways:

- An active-low input to the $\overline{\text{RESET}}$ pin
- A poweron reset function
- A clock monitor failure
- A computer operating properly (COP) watchdog-timer timeout

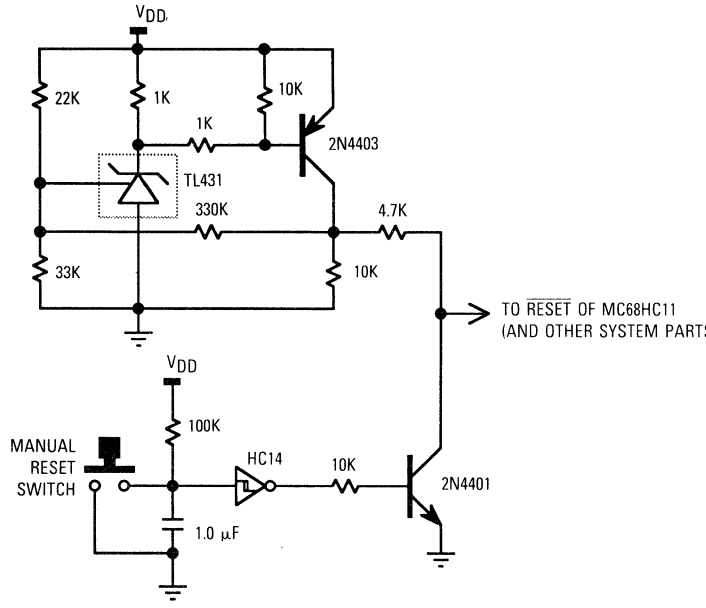
The $\overline{\text{RESET}}$ input consists mainly of a Schmitt trigger that senses the $\overline{\text{RESET}}$ line logic level.

6.1.1 $\overline{\text{RESET}}$ Pin

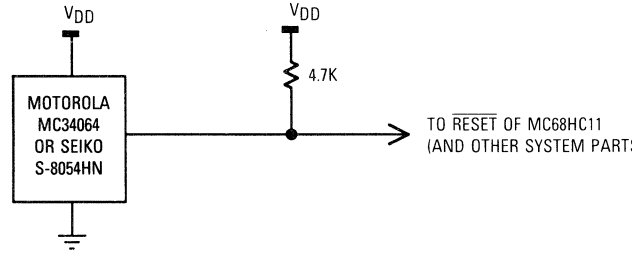
To request an external reset, the $\overline{\text{RESET}}$ pin must be held low for at least eight E-clock cycles, or for one E-clock cycle if no distinction is needed between internal and external resets. To prevent the EEPROM contents from being corrupted during power transitions, the $\overline{\text{RESET}}$ line should be held low while V_{DD} is below minimum operating level. A low voltage inhibit (LVI) circuit is required to protect EEPROM from corruption (see Figure 6-1).

6.1.2 Poweron Reset (POR)

Poweron reset occurs when a positive transition is detected on V_{DD}. This reset is used strictly for power turnon 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 poweron delay time, the processor remains in the reset condition until $\overline{\text{RESET}}$ goes high.



Reset Circuit with LVI and RC Delay



Simple LVI Reset Circuit

Figure 6-1. Typical LVI Reset Circuits

6.1.3 Computer Operating Properly (COP) Reset

The MCU contains a watchdog timer that automatically times out unless it is reset 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.

The COP reset function can be enabled by programming the NOCOP control bit of the system configuration register (CONFIG). Once programmed, this control bit remains set (or cleared), even when no power is applied, and the COP function is enabled or disabled independent of resident software. Protected control bits (CR1 and CR0) in the configuration options register (OPTION) allow the user to select one of four COP timeout rates. Table 6-1 shows the relationship between CR1 and CR0 and the COP timeout period for various system clock frequencies.

Table 6-1. COP Timeout Periods

CR1	CR0	E÷2^{15} Divided By	XTAL=2^{23} Timeout −0/ +15.6 ms	XTAL=8.0 MHz Timeout −0/+16.4 ms	XTAL= 4.9152 MHz Timeout −0/+26.7 ms	XTAL=4.0 MHz Timeout −0/+32.8 ms	XTAL= 3.6864 MHz Timeout −0/+35.6 ms
0	0	1	15.625 ms	16.384 ms	26.667 ms	32.768 ms	35.556 ms
0	1	4	62.5 ms	65.536 ms	106.67 ms	131.07 ms	142.22 ms
1	0	16	250 ms	262.14 ms	426.67 ms	524.29 ms	568.89 ms
1	1	64	1 s	1.049 s	1.707 s	2.1 s	2.276 s
		E=	2.1 MHz	2.0 MHz	1.2288 MHz	1.0 MHz	921.6 kHz

The sequence for resetting the watchdog timer is as follows:

1. write $55 to the COP reset register (COPRST), and
2. write $AA to the COPRST register.

Both writes must occur in this sequence prior to the timeout, but any number of instructions can be executed between the two writes.

6.1.4 Clock Monitor Reset

The MCU contains a clock monitor circuit that measures the E-clock frequency. If the E-clock input rate is above approximately 200 kHz, then the clock monitor does not generate an MCU reset. If the E-clock signal is lost or its frequency falls below 10 kHz, then an MCU reset is generated, and the $\overline{\text{RESET}}$ pin is driven low to reset the external system.

On the B77M mask level devices, the clock monitor can be enabled or disabled by a read/write control bit (CME) in the OPTION register. On masks created after B77M the clock monitor is further qualified by the FCME bit, as indicated in the following register description.

6.1.5 Configuration Options Register (OPTION)

The OPTION register is a special-purpose register with several time-protected bits. OPTION is used during initialization to configure internal system options. This register has an additional bit, FCME, for masks created after the B77M mask number.

Bits 5, 4, 2, 1, and 0 can only be written once during the first 64 E-clock cycles after reset in normal modes (SMOD = 0). In special modes (SMOD = 1), the bits can be written at any time. Bits 7, 6, and 3 can be written at any time.

	7	6	5	4	3	2	1	0	
$1039	ADPU	CSEL	IRQE	DLY	CME	FCME	CR1	CR0	OPTION
RESET:	0	0	0	1	0	0	0	0	

ADPU — Analog-to-Digital Powerup

1 = A/D system is powered up (allow about 100 ms for system stabilization).

0 = A/D system is powered down (reduces supply current).

CSEL — Clock Select

This bit should be set when the E-clock frequency is too slow to program EEPROM (<1 MHz) or to operate the A/D system (<750 kHz).

1 = Internal RC clock source is enabled for the A/D and EEPROM (allow about 10 ms for A/D stabilization).

0 = A/D system and EEPROM use the system E-clock source.

IRQE — $\overline{\text{IRQ}}$ Edge/Level Sensitivity Select

This bit can only be written once during the first 64 E-clock cycles after reset in normal modes.

1 = $\overline{\text{IRQ}}$ is configured to respond only to the falling edges.

0 = $\overline{\text{IRQ}}$ is configured for low-level wired-OR operation.

DLY — STOP Mode Exit Turn-On Delay

This bit is set during reset and can only be written once during the first 64 E-clock cycles after reset in normal modes. If an external clock source rather than a crystal is used, the stabilization delay can be inhibited since the clock source is assumed to be stable.

1 = A stabilization delay of 4064 E-clock cycles is imposed before processing resumes after a STOP mode wakeup.

0 = No stabilization delay is imposed after STOP recovery.

CME — Clock Monitor Enable

If the FCME bit is set then this bit has no effect.

1 = Clock monitor circuit is enabled.

0 = Clock monitor circuit is disabled.

FCME — Force Clock Monitor Enable

This bit is not implemented on B77M mask-level devices. On the B77M mask parts, it always reads zero. This bit is cleared during reset and can only be written once during the first 64 E-clock cycles after reset in normal modes. When this bit is set, the clock monitor feature cannot be disabled until a reset occurs. To use both the clock monitor and STOP, the CME bit should be cleared before the execution of a STOP instruction and set after recovery from STOP.

1 = Clock monitor circuit is enabled until the next reset.

0 = Clock monitor circuit follows the state of the CME bit.

CR1 and CR0 — COP Timer Rate Selects

The COP system is driven by a constant frequency of E divided by 2 to the 15^{th} power. These two bits specify an additional divide-by value to arrive at the COP timeout rate. These bits are cleared during reset and can only be written once during the first 64 E-clock cycles after reset in normal modes. The value of these bits is shown in the following table:

CR1	CR0	$E \div 2^{15}$ Divided By
0	0	1
0	1	4
1	0	16
1	1	64

6.2 INTERRUPTS

Excluding reset-type interrupts, there are 17 hardware interrupts and one software interrupt that can be generated from all the possible sources. These interrupts can be divided into two categories, maskable and nonmaskable. Fifteen of the interrupts can be masked using the I bit of the condition code register (CCR). All the on-chip (hardware) interrupts are individually maskable by local control bits. The software interrupt is nonmaskable. The external input to the $\overline{\text{XIRQ}}$ pin is considered a nonmaskable interrupt because it cannot be masked by software once it is enabled. However, it is masked during reset and upon receipt of an interrupt at the $\overline{\text{XIRQ}}$ pin. Illegal opcode is also a nonmaskable interrupt.

Table 6-2 provides a list of the interrupts with a vector location in memory for each, as well as the actual condition code and control bits that mask each interrupt. Figure 6-2 shows the interrupt stacking order.

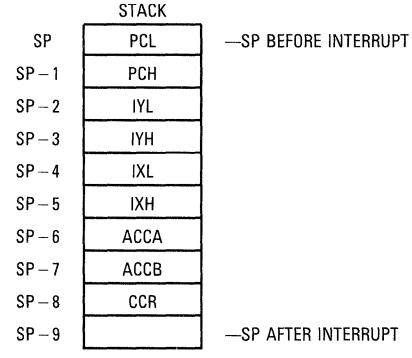


Figure 6-2. Interrupt Stacking Order

Table 6-2. Interrupt Vector Masks and Assignments

Vector Address	Interrupt Source	Condition Code Register Mask	Local Mask
FFC0,FFC1	Reserved	—	—
*	*		
*	*		
FFD4,FFD5	Reserved	—	—
FFD6,FFD7	SCI Serial System	I Bit	—
	Receive Data Register Full	—	RIE
	Receive Overrun	—	RIE
	Idle Line Detect	—	ILIE
	Transmit Data Register Empty	—	TIE
	Transmit Complete	—	TCIE
FFD8,FFD9	SPI Serial Transfer Complete	I Bit	SPIE
FFDA,FFDB	Pulse Accumulator Input Edge	I Bit	PAII
FFDC,FFDD	Pulse Accumulator Overflow	I Bit	PAOVI
FFDE,FFDF	Timer Overflow	I Bit	TOI
FFE0,FFE1	Timer IC4/OC5	I Bit	I4O5I
FFE2,FFE3	Timer Output Compare 4	I Bit	OC4I
FFE4,FFE5	Timer Output Compare 3	I Bit	OC3I
FFE6,FFE7	Timer Output Compare 2	I Bit	OC2I
FFE8,FFE9	Timer Output Compare 1	I Bit	OC1I
FFEA,FFEB	Timer Input Capture 3	I Bit	IC3I
FFEC,FFED	Timer Input Capture 2	I Bit	IC2I
FFEE,FFEF	Timer Input Capture 1	I Bit	IC1I
FFF0,FFF1	Real-Time Interrupt	I Bit	RTII
FFF2,FFF3	$\overline{\text{IRQ}}$ – External Pin	I Bit	None
FFF4,FFF5	$\overline{\text{XIRQ}}$ Pin (Psuedo-Nonmaskable)	X Bit	None
FFF6,FFF7	SWI	None	None
FFF8,FFF9	Illegal Opcode Trap	None	None
FFFA,FFFB	COP Failure (Reset)	None	NOCOP
FFFC,FFFD	Clock Monitor Fail (Reset)	None	CME
FFFE,FFFF	$\overline{\text{RESET}}$	None	None

6.2.1 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 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.

6.2.2 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 requested to the illegal opcode vector. The illegal opcode vector should never be left uninitialized.

6.2.3 Real-Time Interrupt

The real-time interrupt 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 be $E \div 2^{13}$, $E \div 2^{14}$, $E \div 2^{15}$, or $E \div 2^{16}$. See PACTL, TMSK2, and TFLG2 register descriptions in **SECTION 7 PROGRAMMABLE TIMER** for control and status bit information.

6.2.4 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 $\overline{\text{XIRQ}}$. After minimum system initialization, software may clear the X bit by a TAP instruction, thus enabling $\overline{\text{XIRQ}}$ interrupts. Thereafter, software cannot set the X bit. Thus, an $\overline{\text{XIRQ}}$ 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 $\overline{\text{XIRQ}}$ 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.

When an I-bit-related interrupt occurs, the I bit is automatically set by hardware after stacking the CCR byte. The X bit is not affected. When an X-bit-related interrupt occurs, both the X and the I bit are automatically set by hardware after stacking the CCR. A return from interrupt (RTI) instruction restores the X and I bits to their preinterrupt request state.

6.2.5 Interrupt Priority Structure

Interrupts obey a fixed hardware priority circuit to resolve simultaneous requests. However, one I-bit-related interrupt source may be elevated to the highest I bit priority in the resolution circuit.

Six interrupt sources are not masked by the I bit of the CCR and have the fixed priority relationship of the following:

1. Reset
2. Clock Monitor Failure
3. COP Failure
4. $\overline{\text{XIRQ}}$
5. Illegal Opcode
6. SWI

SWI is actually an instruction and has highest priority, other than resets, in that once the SWI opcode is fetched, no other interrupt can be honored until the SWI vector has been fetched.

Each of the previous sources is an input to the priority resolution circuit. The highest I-bit-masked priority input to the resolution circuit is assigned to be connected to any one of the remaining I-bit-related interrupt sources. This assignment is made under the software control of the HPRIO register. To avoid timing races, the HPRIO register can only be written while the I-bit-related interrupts are inhibited (I bit of CCR is logic one). An interrupt that is assigned to this higher priority position is still subject to masking by any associated control bits or by the I bit in the CCR. The interrupt vector address is not affected by assigning a source to the higher priority position.

Figures 6-3, 6-4, and 6-5 illustrate the interrupt process as it relates to normal processing. Figure 6-3 shows how the CPU begins from a reset and how interrupt detection relates to normal opcode fetches. Figure 6-4, an expansion of a block in Figure 6-3, shows how interrupt priority is resolved. Figure 6-5, an expansion of the SCI interrupt block in Figure 6-4, shows the resolution of interrupt sources within the SCI subsystem.

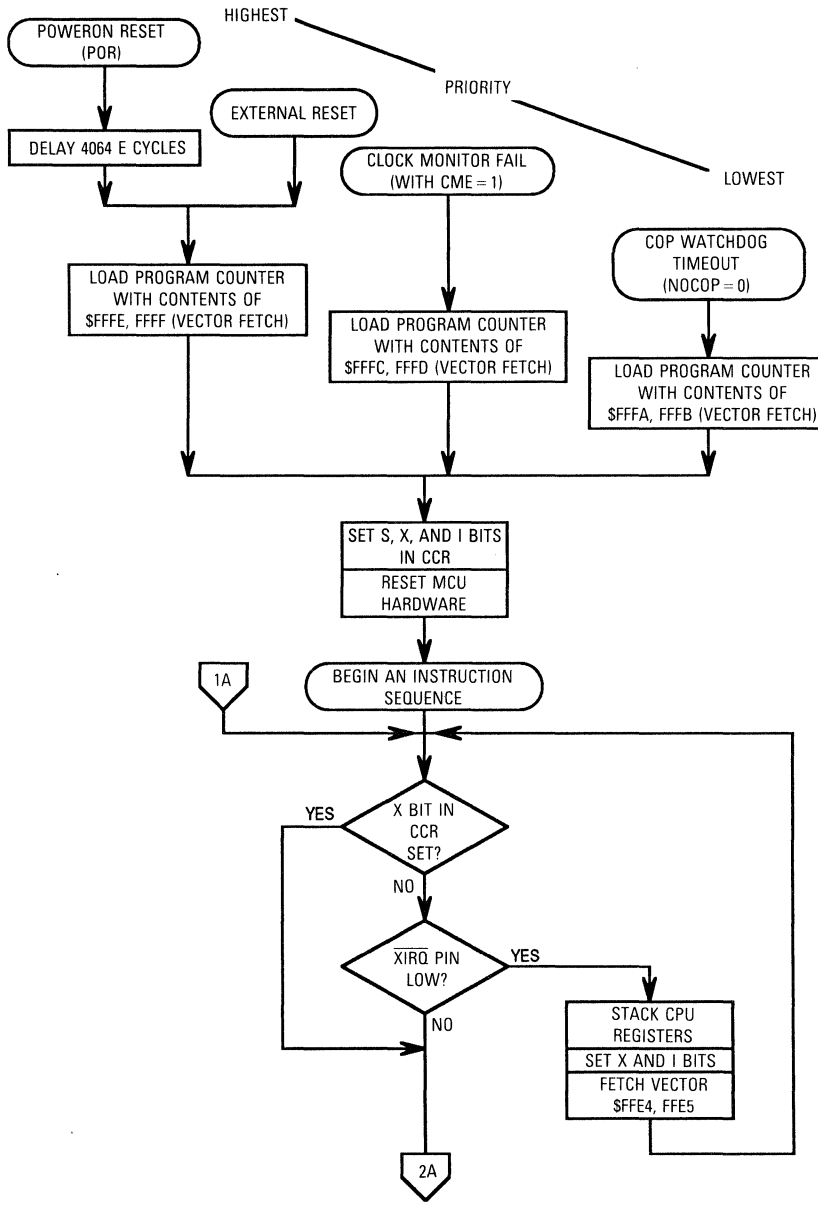


Figure 6-3. Processing Flow Out of Resets (Sheet 1 of 2)

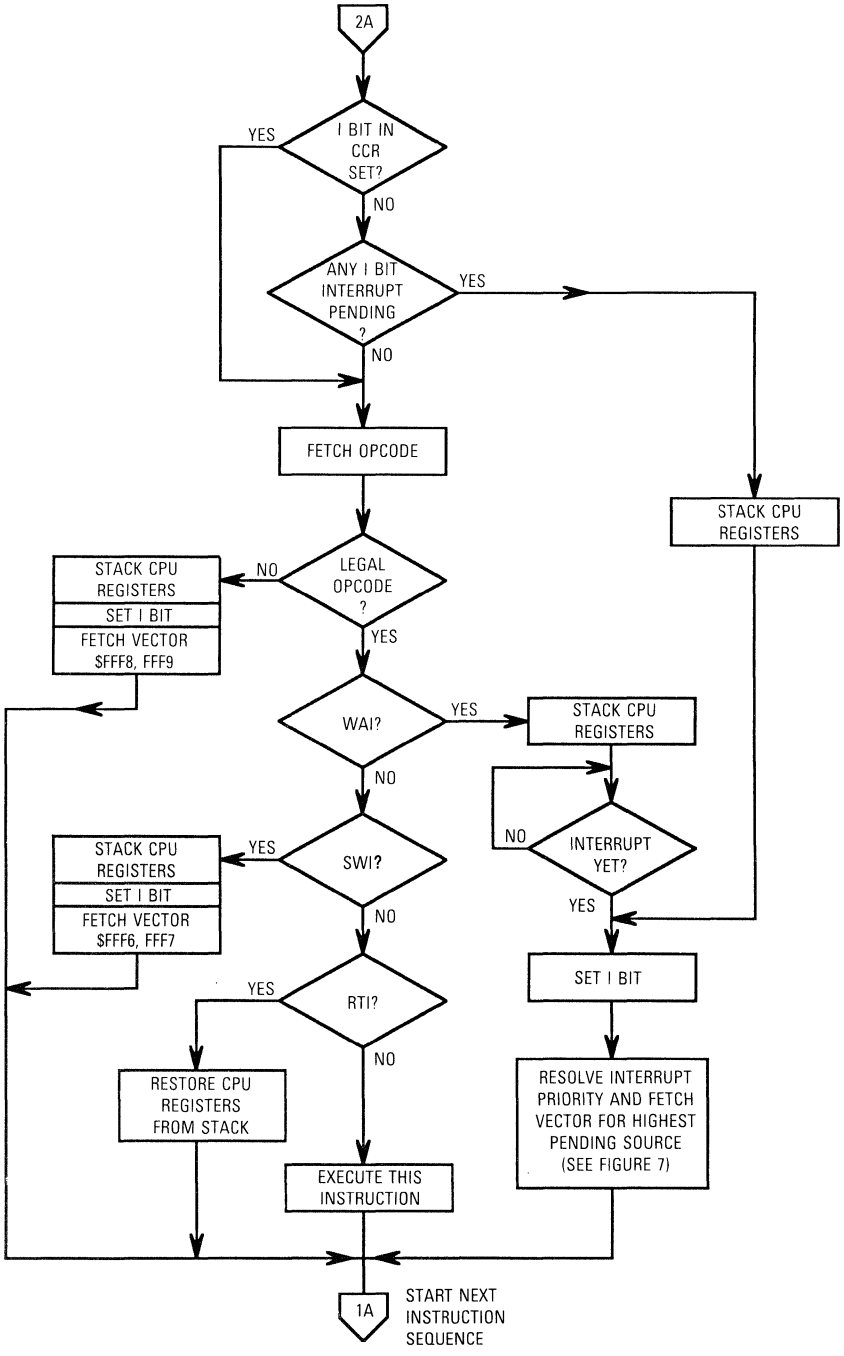


Figure 6-3. Processing Flow Out of Resets (Sheet 2 of 2)

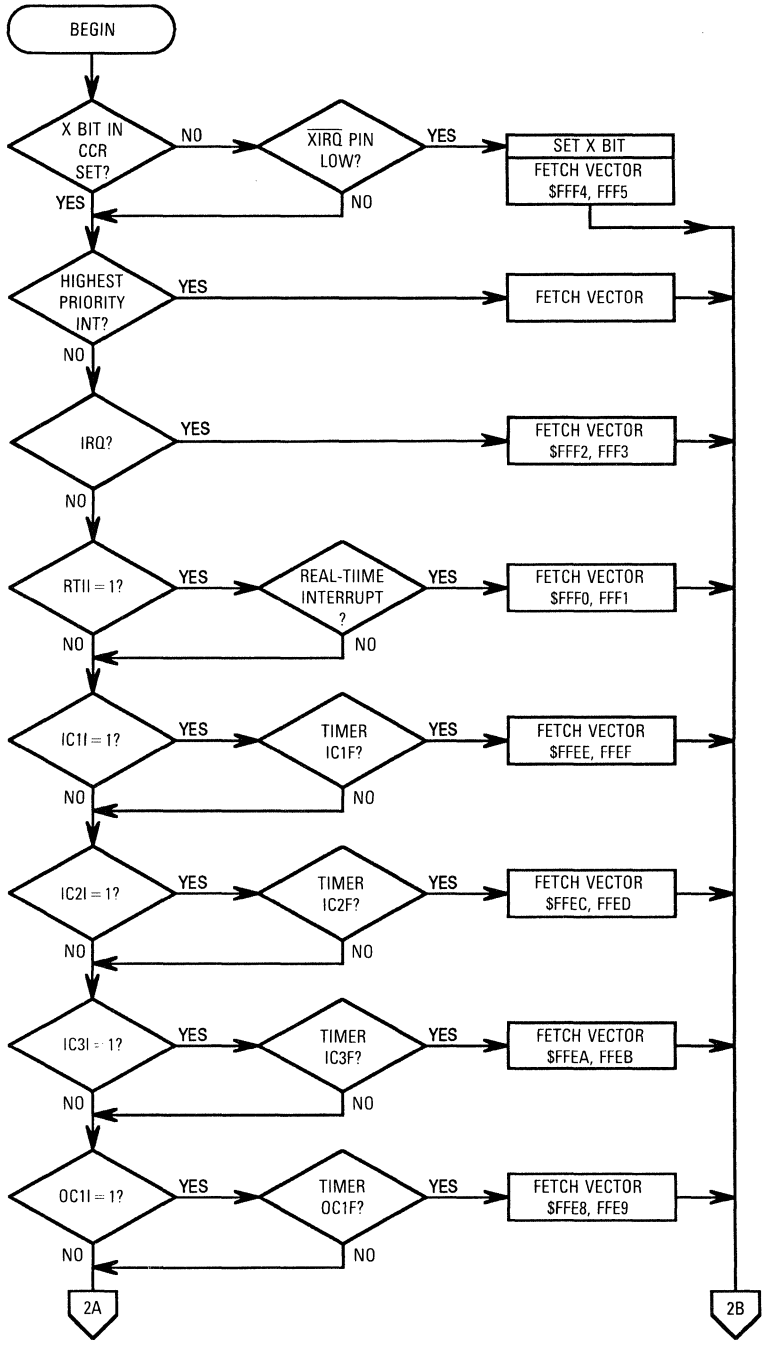


Figure 6-4. Interrupt Priority Resolution (Sheet 1 of 2)

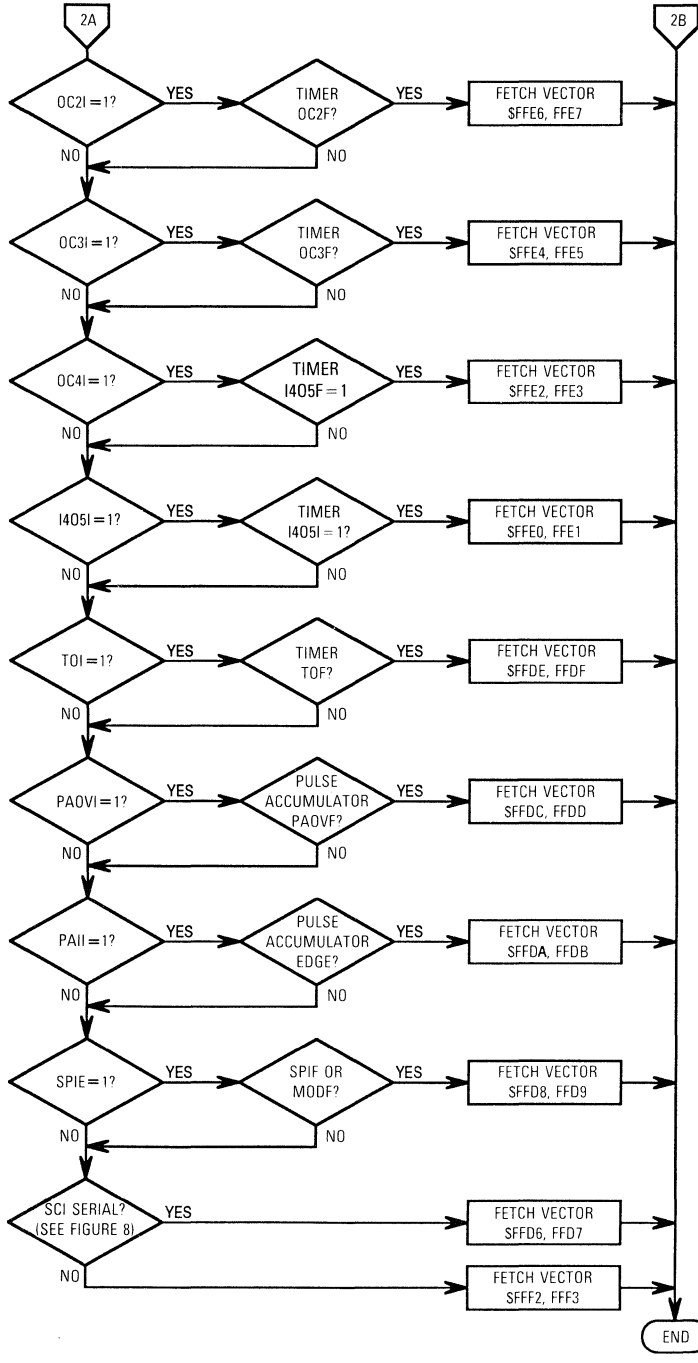


Figure 6-4. Interrupt Priority Resolution (Sheet 2 of 2)

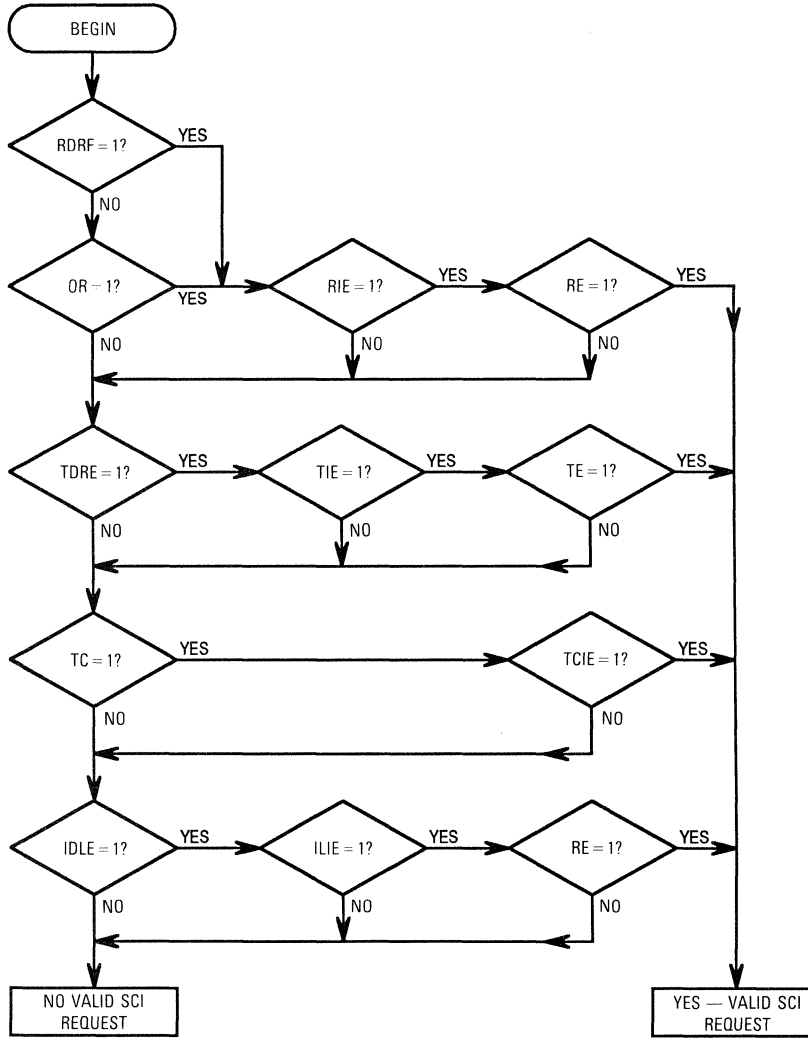


Figure 6-5. Interrupt Source Resolution within SCI

6.2.6 Highest Priority Interrupt and Miscellaneous Register (HPRIO)

Four bits of this register (PSEL3–PSEL0) are used to select one of the I-bit-related interrupt sources and to elevate it to the highest-I-bit masked position of the priority resolution circuit. In addition, four miscellaneous system control bits are included in this register.

	7	6	5	4	3	2	1	0	
$103C	RBOOT	SMOD	MDA	IRV	PSEL3	PSEL2	PSEL1	PSEL0	HPRIO
RESET:	*	*	*	*	0	1	0	1	

* = The reset condition of bits 7, 6, 5, and 4 depends on the mode selected at powerup initialization.

RBOOT — Read Bootstrap ROM

This bit can be read at any time. It can only be written in special modes (SMOD = 1). In special bootstrap mode, it is set during reset. Reset clears it in all other modes.

1 = Bootloader ROM is enabled in the memory map at $BF00–BFFF.
0 = Bootloader ROM is disabled and is not in the memory map.

SMOD and MDA — Special Mode Select and Mode Select A

These two bits can be read at any time. They may only be written in special modes (SMOD = 1). These bits reflect the status of the MODA and MODB input pins at the rising edge of reset. SMOD cannot be written to a one after being cleared without an interim reset. An interpretation of the values of these two bits is shown in the following table:

Input Pins		Mode Description	Latched at Reset	
MODB	MODA		SMOD	MDA
1	0	Single Chip	0	0
1	1	Expanded Nonmultiplexed	0	1
0	0	Special Bootstrap	1	0
0	1	Special Test	1	1

IRV — Internal Read Visibility Enable

This bit may be read at any time. It may be written at any time in special modes (SMOD = 1) and may only be written once in normal modes (SMOD = 0). IRV is set during reset in special modes and cleared by reset in normal modes.

1 = Data from internal reads is driven out on the external data bus in expanded modes.

0 = Data from internal reads is not visible on the external data bus.

NOTE

To prevent bus conflicts, the user must disable all external devices from driving the data bus during any internal access.

PSEL3–PSEL0 — Priority Selects

These four bits are used to specify one I-bit-related interrupt source, which then becomes the highest priority I-bit-related interrupt source. These bits may only be written while the I bit in the CCR is set, inhibiting I-bit-related interrupts. An interpretation of the value of these bits is shown in the following table:

PSEL3	PSEL2	PSEL1	PSEL0	Interrupt Source Promoted
0	0	0	0	Timer Overflow
0	0	0	1	Pulse Accumulator Overflow
0	0	1	0	Pulse Accumulator Input Edge
0	0	1	1	SPI Serial Transfer Complete
0	1	0	0	SCI Serial System
0	1	0	1	Reserved (Default to $\overline{\text{IRQ}}$)
0	1	1	0	$\overline{\text{IRQ}}$ (External Pin)
0	1	1	1	Real-Time Interrupt
1	0	0	0	Timer Input Capture 1
1	0	0	1	Timer Input Capture 2
1	0	1	0	Timer Input Capture 3
1	0	1	1	Timer Output Compare 1
1	1	0	0	Timer Output Compare 2
1	1	0	1	Timer Output Compare 3
1	1	1	0	Timer Output Compare 4
1	1	1	1	Timer IC4/OC5

During reset, PSEL3–PSEL0 are initialized to 0:1:0:1, which corresponds to "Reserved (Default to $\overline{\text{IRQ}}$)". $\overline{\text{IRQ}}$ becomes the highest priority I-bit-related interrupt source.

6.3 LOW POWER MODES

The MC68HC11F1 has two programmable low power-consumption modes, stop and wait. In the wait mode, the on-chip oscillator remains active. In the stop mode, the oscillator is stopped. The following paragraphs describe these two low power-consumption modes.

6.3.1 STOP

The STOP instruction places the MCU in its lowest power-consumption mode, provided the S bit in the CCR is cleared. In this mode, all clocks are stopped, thereby halting all internal processing.

To exit the stop mode, a low level must be applied to either the $\overline{IRQ}$, $\overline{XIRQ}$, or $\overline{RESET}$ pin. An external interrupt used at $\overline{IRQ}$ is only effective if the I bit in the CCR is cleared. An external interrupt applied at the $\overline{XIRQ}$ input is effective, regardless of the setting of the X bit of the CCR. However, the actual recovery sequence differs, depending on the X bit setting. If the X bit is cleared, the MCU starts with the stacking sequence leading to the normal service of the $\overline{XIRQ}$ request. If the X bit is set, the processing always continues with the instruction immediately following the STOP instruction. A low input to the $\overline{RESET}$ pin always results in an exit from the stop mode, and the start of MCU operations is determined by the reset vector.

The CPU will not exit the STOP mode correctly when interrupted by $\overline{IRQ}$ or $\overline{XIRQ}$ if the instruction immediately preceding STOP is a column 4 or 5 accumulator inherent instruction (opcodes $4X and $5X), such as NEGA, NEGB, COMA, COMB, etc. These single byte, two cycle instructions must be followed by a NOP, then the STOP command. If reset is used to exit STOP mode, the CPU will respond properly.

A restart delay is required if the internal oscillator is being used. The delay allows the oscillator to stabilize when exiting the stop mode. If a stable external oscillator is being used, the delay (DLY) bit in the OPTION register can be cleared to bypass the delay. If the DLY bit is clear, the $\overline{RESET}$ pin would not normally be used to exit the stop mode. The reset sequence sets the DLY bit, and the restart delay would be reimposed.

6.3.2 WAIT

The wait (WAI) instruction places the MCU in a low power-consumption mode. The wait mode consumes more power than the stop mode since the oscillator is kept running. Upon execution of the WAI instruction, the machine state is stacked and program execution stops.

The wait state can only be exited by an unmasked interrupt or $\overline{\text{RESET}}$. If the I bit of the CCR is set and the COP is disabled, the timer system is turned off by WAI to further reduce power consumption. The amount of power savings is application dependent. Power savings are also dependent upon the circuitry connected to the MCU pins and upon subsystems, such as the timer, SPI, or SCI, that were or were not active when the wait mode was entered. Clearing the A/D bit (ADPU) of the OPTION register to turn off the A/D subsystem further reduces power consumption in the wait mode.

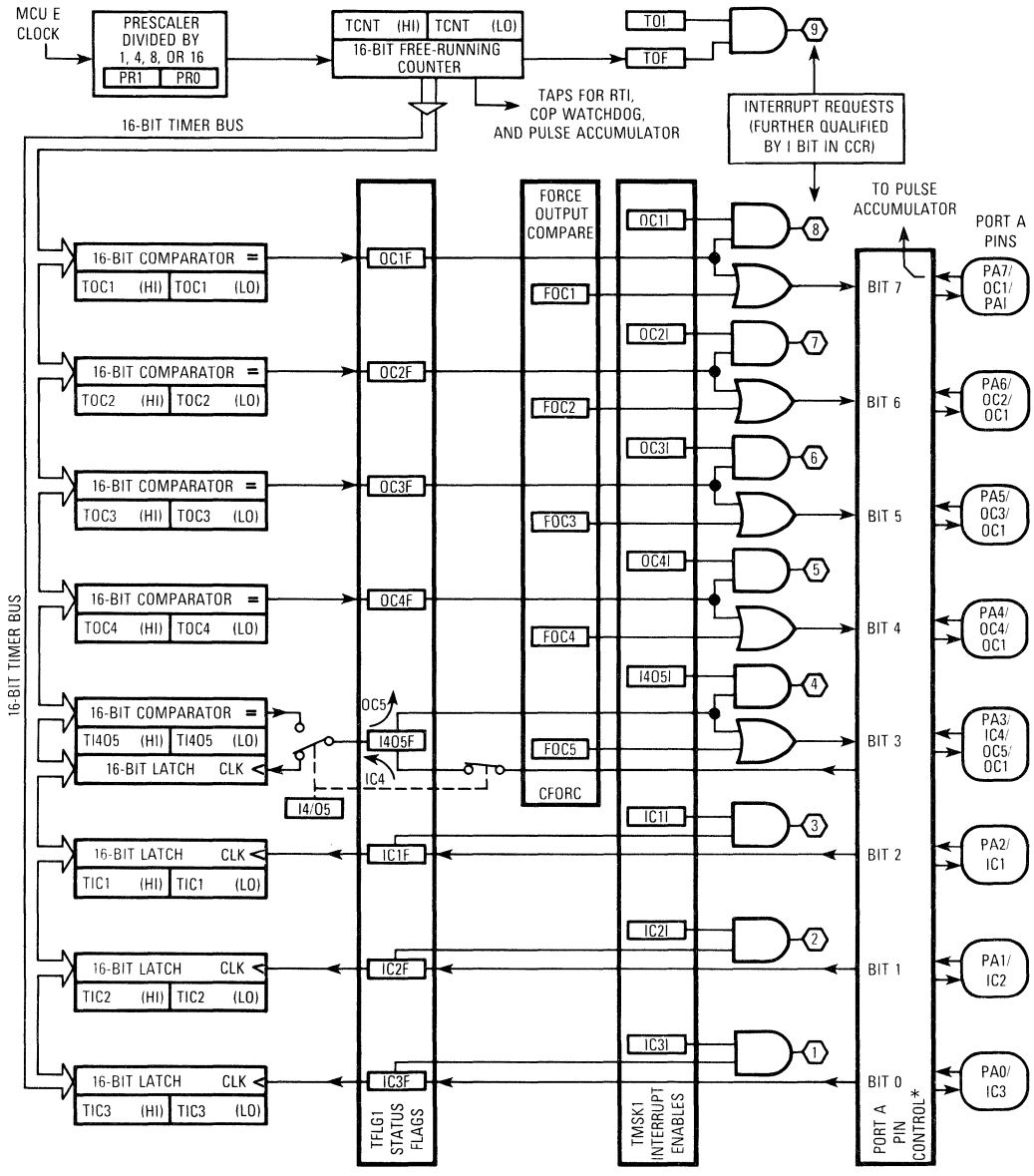
SECTION 7
PROGRAMMABLE TIMER

The timer system uses a time-of-day approach in that all timing functions are related to a single, 16-bit, free-running counter. The free-running counter is clocked by the output of a programmable prescaler. The prescaler is clocked by the E clock and divides this clock by 1, 4, 8, or 16. The prescaler control bits, found in the TMSK2 register, can only be written once during the first 64 E-clock cycles after a reset. The free-running counter (TCNT register) can be read by software at any time without affecting its value since it is clocked and read on the opposite half-cycle of the E clock. The counter is cleared on reset, is a read-only register, and repeats every 65,536 counts. When the count changes from $FFFF to $0000, the timer overflow flag (TOF) bit is set in the timer interrupt flag register 2 (TFLG2). The overflow flag also generates an internal interrupt if the timer overflow interrupt enable (TOI) bit of the timer interrupt mask register 2 (TMSK2) is set. The timer has three input capture and four output compare function registers as well as an additional register that can perform either function under software control.

Figure 7-1 is a block diagram of the timer. The functions and registers of the timer are explained in the following paragraphs.

7.1 INPUT CAPTURE FUNCTION (TCTL2)

There are three, regular, 16-bit, read-only input capture (IC) registers (TIC1, TIC2, and TIC3) and a register that can serve as either the fourth input capture register or the fifth output compare register (TI4O5). These registers are not initialized by reset. Each input capture register is used to latch the value of the free-running counter when a selected transition at an external pin is detected. External devices provide the inputs to IC4–IC1 on the PA3–PA0 pins. An interrupt can be generated when an input capture edge is detected. The time of detection can be read from the appropriate timer register as part of the interrupt routine.



*Port A pin actions controlled by DDRA, OC1M, OC1D, TCTL1, and TCTL2 registers.

Figure 7-1. Timer Block Diagram

The control and status bits to implement the input capture functions are contained in the DDRA, PACTL, TCTL2, TMSK1, and TFLG1 registers. To configure port A bits 0, 1, 2, or 3 as input captures, the user must clear the respective bits in DDRA. These bits are cleared out of reset. Additionally, to enable PA3 as the fourth input capture, the I4/O5 bit in the PACTL register must be set. Otherwise, PA3 is configured as a fifth output compare out of reset, with bit I4/O5 being cleared. If DDRA bit DDA3 is set (configuring PA3 as an output) and IC4 is enabled, then writes to PA3 cause edges on the pin to result in input captures. When the TI4O5 register is acting as IC4, it cannot be written to.

	7	6	5	4	3	2	1	0	
$1021	EDG4B	EDG4A	EDG1B	EDG1A	EDG2B	EDG2A	EDG3B	EDG3A	TCTL2
RESET:	0	0	0	0	0	0	0	0	

EDGxB and EDGxA — Input Capture Edge Control

There are four pairs of these bits. Each pair is cleared to zero by reset and is encoded to configure the input capture edge detector circuit for IC4–IC1. Coding is as follows:

EDGxB	EDGxA	Configuration
0	0	Capture Disabled
0	1	Capture on Rising Edges Only
1	0	Capture on Falling Edges Only
1	1	Capture on Any Rising or Falling Edge

NOTE

IC4 only functions if the I4/O5 bit of the PACTL register is set.

7.2 OUTPUT COMPARE FUNCTION

There are four 16-bit read/write output compare (OC) registers (TOC1, TOC2, TOC3, and TOC4) and a fifth register (TI4O5) that can function under software control as either IC4 or OC5. Each of the output compare registers is set to $FFFF on reset. A value written to an output compare register is compared to the free-running counter value during each E-clock cycle. If a match is found, the particular output compare flag is set in the timer interrupt flag register 1 (TFLG1). An interrupt is then generated if that particular interrupt is enabled in the timer interrupt mask register 1 (TMSK1). In addition to the interrupt, a specified action may be initiated at a timer output pin(s). For OC5–OC2, the pin action is controlled by pairs of bits (OMx and OLx) in the TCTL1 register. The output action is taken on each successful compare, regardless of whether or not the OCxF flag in the TFLG1 register was previously clear.

OC1 is different from the other output compares in that a successful OC1 compare can affect any or all five of the output compare pins. The OC1 output action to be taken when a match is found is controlled by two 5-bit registers, the output compare 1 mask register (OC1M) and the output compare 1 data register (OC1D). OC1M specifies which port A outputs are to be used, and OC1D specifies what data is placed on these port pins.

Upon reset, bit I4/O5 of PACTL is configured as OC5. The OC5 function is then enabled when bits OM5 and OL5 of TCTL1 are appropriately set. Although DDRA configures all port A pins as inputs out of reset, the port A pins assigned to each enabled output compare are forced to be outputs. The DDRA register bits do not change, however. The DDRA bits revert to I/O control once the associated output compare is disabled.

The control and status bits to implement the output compare functions are contained in the PACTL, CFORC, OC1M, OC1D, TCTL1, TMSK1, and TFLG1 registers.

7.2.1 Timer Compare Force Register (CFORC)

This write-only register is used to force early output compare actions. This compare force function is not recommended for use with the output toggle function, because a normal compare occurring immediately before or after the force can result in an undesirable operation.

	7	6	5	4	3	2	1	0	
$100B	FOC1	FOC2	FOC3	FOC4	FOC5	0	0	0	CFORC
RESET:	0	0	0	0	0	0	0	0	

FOC5–FOC1 — Force Output Compare x Action

1 = Causes the action programmed for output compare x to occur; the output compare x flag (OCxF) of TFLG1 is not set.

0 = Not affected.

Bits 2–0 — Not implemented; always read zero.

7.2.2 Output Compare 1 Mask Register (OC1M)

This register is used with OC1 to specify the bits of port A that are affected as the result of a successful OC1 compare. The bits of the OC1M register correspond bit for bit with lines 7–3 of port A.

	7	6	5	4	3	2	1	0	
$100C	OC1M7	OC1M6	OC1M5	OC1M4	OC1M3	0	0	0	OC1M
RESET:	0	0	0	0	0	0	0	0	

OC1M7–OC1M3 — Output Compare Masks

1 = OC1 is enabled to control the corresponding pin of port A.

0 = OC1 is disabled.

Bits 2–0 — Not implemented; always read zero.

7.2.3 Output Compare 1 Data Register (OC1D)

OC1D is used with OC1 to specify the data that is to be stored on the affected pin of port A as the result of a successful OC1 compare. When a successful OC1 compare occurs, for each bit that is set in OC1M, the corresponding data bit in OC1D is stored in the corresponding bit of port A.

	7	6	5	4	3	2	1	0	
$100D	OC1D7	OC1D6	OC1D5	OC1D4	OC1D3	0	0	0	OC1D
RESET:	0	0	0	0	0	0	0	0	

Bits 2–0 — Not implemented; always read zero.

7.2.4 Timer Count Register (TCNT)

	15	14	13	12	11	10	9	8	
$100E									TCNT
$100F									TCNT
	7	6	5	4	3	2	1	0	
RESET:	0 0	0 0	0 0	0 0	0 0	0 0	0 0	0 0	

This 16-bit read-only register contains the prescaled value of the 16-bit timer. A full counter read should first address the most significant byte. A read of this address causes the least significant byte to be latched into a buffer for the next CPU cycle so that a double-byte read will return the full 16-bit state of the counter at the time of the most significant byte read cycle.

7.2.5 Timer Control Register 1 (TCTL1)

The bits of this register are used to specify the action taken as the result of a successful OCx compare.

	7	6	5	4	3	2	1	0	
$1020	OM2	OL2	OM3	OL3	OM4	OL4	OM5	OL5	TCTL1
RESET:	0	0	0	0	0	0	0	0	

OM2–OM5 — Output Mode

OL2–OL5 — Output Level

These control bit pairs are encoded to specify the action taken as the result of a successful OCx compare. OC5 only functions if the I4/O5 bit in the PACTL register is clear. Coding is as follows:

OMx	OLx	Action Taken upon Successful Compare
0	0	Timer Disconnected from Output Pin Logic
0	1	Toggle OCx Output Line
1	0	Clear OCx Output Line to Zero
1	1	Set OCx Output Line to One

7.2.6 Timer Interrupt Mask Register 1 (TMSK1)

This 8-bit register is used to enable or inhibit the timer input capture and output compare interrupts.

	7	6	5	4	3	2	1	0	
$1022	OC1I	OC2I	OC3I	OC4I	I4O5I	IC1I	IC2I	IC3I	TMSK1
RESET:	0	0	0	0	0	0	0	0	

OCxI — Output Compare x Interrupt

1 = Interrupt sequence requested if bit OCxF of TFLG1 is set.

0 = Interrupt inhibited.

ICxI — Input Capture x Interrupt

1 = Interrupt sequence requested if bit ICxF of TFLG1 is set.

0 = Interrupt inhibited.

I4O5I — Input Capture 4 or Output Compare 5 Interrupt

When the I4/O5 bit of PACTL is set, the I4O5I bit acts as the IC4 interrupt bit. When I4/O5 is cleared, the I4O5I bit acts as the OC5 interrupt control bit.

1 = Interrupt sequence requested.

0 = Interrupt inhibited.

7.2.7 Timer Interrupt Flag Register 1 (TFLG1)

This register is used to indicate the occurrence of timer system events. With the TMSK1 register, TFLG1 allows the timer subsystem to operate in a polled or interrupt driven system. Each bit of TFLG1 has a corresponding bit in TMSK1 in the same bit position.

	7	6	5	4	3	2	1	0	
$1023	OC1F	OC2F	OC3F	OC4F	I4O5F	IC1F	IC2F	IC3F	TFLG1
RESET:	0	0	0	0	0	0	0	0	

OCxF — Output Compare x Flag

Set each time the timer counter matches the output compare register x value, the flag bits of TFLG1 are cleared by writing one to the corresponding bit position.

I4O5F — Input Capture 4/Output Compare 5 Flag

This bit functions as the flag for either input capture 4 or output compare 5. The flag is set by hardware as IC4 or OC5, depending on which function was enabled by bit I4/O5 of PACTL at the time of interrupt. This bit is cleared by writing one to bit 3.

ICxF — Input Capture x Flag

Set each time a selected active edge is detected on the ICx input line, these bits are cleared by writing one to the corresponding bit position.

7.2.8 Timer Interrupt Mask Register 2 (TMSK2)

This register is used to control whether or not a hardware interrupt sequence is requested as the result of a status bit being set in the timer interrupt flag register (TFLG1). The two timer prescaler bits are also included in this register.

	7	6	5	4	3	2	1	0	
$1024	TOI	RTII	PAOVI	PAII	0	0	PR1	PR0	TMSK2
RESET:	0	0	0	0	0	0	0	0	

TOI — Timer Overflow Interrupt Enable
1 = Interrupt requested when bit TOF of TFLG2 is set.
0 = Timer overflow interrupt disabled.

RTII — Real-Time Interrupt Enable
1 = Interrupt requested when bit RTIF of TFLG2 is set.
0 = Real-time interrupt disabled.

PAOVI — Pulse Accumulator Overflow Interrupt Enable
1 = Interrupt requested when bit PAOVF of TFLG2 is set.
0 = Pulse accumulator overflow interrupt disabled.

PAII — Pulse Accumulator Interrupt Enable
1 = Interrupt requested when bit PAIF of TFLG2 is set.
0 = Pulse accumulator interrupt disabled.

Bits 3–2 — Not implemented; always read zero.

PR1 and PR0 — Timer Prescaler Select
These bits are used to select the prescaler divide-by ratio. They can be written only during initialization in normal modes. The bits can be written once in the first 64 E-clock cycles following reset in normal modes. The bits are writable at any time in special modes. The value of these bits is decoded as follows:

PR1	PR0	Divide By
0	0	1
0	1	4
1	0	8
1	1	16

7.2.9 Timer Interrupt Flag Register 2 (TFLG2)

Bits of this register indicate the occurrence of timer system events. Coupled with the four high-order bits of TMSK2, the bits of TFLG2 allow the timer subsystem to operate in either a polled or interrupt driven system. Each bit of TFLG2 corresponds to a bit in TMSK2 in the same position.

	7	6	5	4	3	2	1	0	
$1025	TOF	RTIF	PAOVF	PAIF	0	0	0	0	TFLG2
RESET:	0	0	0	0	0	0	0	0	

TOF — Timer Overflow

This bit is set each time the 16-bit free-running counter advances from a value of $FFFF–$0000. It is cleared by a write to TFLG2 with bit 7 set.

RTIF — Real-Time Interrupt Flag

This bit is set at each rising edge of the selected tap point. It is cleared by a write to TFLG2 with bit 6 set.

PAOVF — Pulse Accumulator Overflow Interrupt Flag

This bit is set when the count in the pulse accumulator rolls over from $FF–$00. It is cleared by a write to TFLG2 with bit 5 set.

PAIF — Pulse Accumulator Input-Edge Interrupt Flag

This bit is set when an active edge is detected on the PAI input pin. In the event mode, the event edge triggers PAIF. In gated time accumulator mode, the trailing edge of the gate signal at the PAI input triggers PAIF. It is cleared by a write to TFLG2 with bit 4 set.

Bits 3–0 — Not implemented; always read zero.

7.3 PULSE ACCUMULATOR

The pulse accumulator is an 8-bit counter that can operate in either of two modes, depending on the state of a control bit in the PACTL register. These modes are the event counting mode and the gated time accumulation mode. In the event counting mode, the 8-bit counter is clocked to increasing values by an external pin. The maximum clocking rate for the external event counting mode is the E clock divided by two. In the gated time accumulation mode, a free-running E clock divided by 64 signal drives the 8-bit counter, but only while the external PAI input pin is activated.

The pulse accumulator uses port A bit 7 as the PAI input, but this pin can also be used as general-purpose I/O or as an output compare. Even when port A bit 7 is configured as an output, the pin still drives the input to the pulse accumulator.

7.3.1 Pulse Accumulator Control Register (PACTL)

Three bits of this register control an 8-bit pulse accumulator system. Another bit enables either the output compare 5 function or the input capture 4 function. Two other bits select the rate for the real-time interrupt system.

	7	6	5	4	3	2	1	0	
$1026	0	PAEN	PAMOD	PEDGE	0	I4/O5	RTR1	RTR0	PACTL
RESET:	0	0	0	0	0	0	0	0	

Bits 7 and 3 — Not implemented; always read zero.

PAEN — Pulse Accumulator System Enable
1 = Pulse accumulator on.
0 = Pulse accumulator off (No flags can become set; counter is frozen.)

PAMOD — Pulse Accumulator Mode
1 = Gated time accumulation mode.
0 = Event counter mode.

PEDGE — Pulse Accumulator Edge Control
Depending on the state of the PAMOD bit, this bit has different meanings as shown in the following table:

PAMOD	PEDGE	Action on Clock
0	0	PAI Falling Edge Increments the Counter
0	1	PAI Rising Edge Increments the Counter
1	0	A Zero on PAI Inhibits Counting
1	1	A One on PAI Inhibits Counting

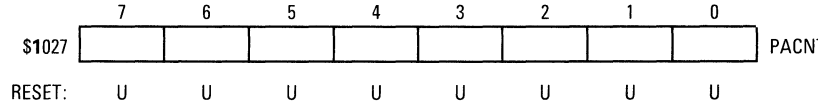
I4/O5 — Configure TI4O5 Register for IC or OC
1 = IC4 function enabled.
0 = OC5 function enabled.

RTR1 and RTR0 — Real-Time Interrupt (RTI) Rate
The decoded value of these pins selects one of four rates for the real-time periodic interrupt circuits. After reset, a full RTI period elapses before the first RTI interrupt. RTI rates are shown in Table 7-1.

Table 7-1. RTI Rate at Various Crystal Frequencies

RTR1	RTR0	Divide E By	XTAL = 2^{23}	XTAL = 8.0 MHz	XTAL = 4.9152 MHz	XTAL = 4.0 MHz	XTAL = 3.6864 MHz
0	0	2^{13}	3.91 ms	4.10 ms	6.67 ms	8.19 ms	8.89 ms
0	1	2^{14}	7.81 ms	8.19 ms	13.33 ms	16.38 ms	17.78 ms
1	0	2^{15}	15.62 ms	16.38 ms	26.67 ms	32.77 ms	35.56 ms
1	1	2^{16}	31.25 ms	32.77 ms	53.33 ms	65.54 ms	71.11 ms
		E =	2.1 MHz	2.0 MHz	1.2288 MHz	1.0 MHz	921.6 kHz

7.3.2 Pulse Accumulator Count Register (PACNT)



This 8-bit read/write register contains the count of external input events at the PAI input or the accumulated count while the PAI input is active in gated time accumulation mode.

SECTION 8
ELECTRICALLY ERASABLE PROGRAMMABLE READ-ONLY MEMORY (EEPROM)

This MCU has 512 bytes of EEPROM that can be mapped to any 4K boundary in memory. The address is $xE00–xFFF, where x represents the value of the four high-order bits of the CONFIG register, and this value is the value of the initial 4K boundary. In single-chip and bootstrap modes, the EEPROM is forced to $FE00–FFFF. In special test mode, the EEPROM is initially disabled out of reset but can be enabled by setting the EEON bit of the CONFIG register.

Programming of the EEPROM is controlled by the EEPROM programming register (PPROG) and block protect register (BPROT). The EEPROM is enabled when the EEON bit of CONFIG is set and disabled when EEON is cleared. EEPROM programming and erasure relies on an internal high-voltage charge pump. At E-clock frequencies below 2 MHz, the efficiency of this charge pump decreases, increasing the time required to program or erase a location. Recommended program and erase time is 10 ms when E clock is 2 MHz and should be increased to as much as 20 ms when E clock is between 1 and 2 MHz. When E clock is below 1 MHz, the clock source for the charge pump should be switched from the system clock to an on-chip RC oscillator clock. Switching is done by setting the CSEL bit in the OPTION register. A 10-ms period should be allowed after setting the CSEL bit to allow the charge pump to stabilize.

An erased EEPROM byte reads as $FF. EEPROM locations can be erased on a byte, row, or bulk basis. Individual bits can be programmed to zero on a byte-wide basis. Programming and erasing the EEPROM is done by using the control bits of the PPROG register, provided that the block protect for the appropriate address block to be programmed is off.

8.1 EEPROM BLOCK PROTECT REGISTER (BPROT)

This 5-bit register protects against inadvertent writes to the CONFIG register and to the EEPROM. To permit the user to separate EEPROM into categories such as temporary or permanent, EEPROM is divided into four individually protected blocks. The CONFIG register is also protected.

In normal operating modes, EEPROM and CONFIG are protected out of reset, and the user has 64 E-clock cycles to unprotect any of the blocks that will require programming or erasing. The BPROT register bits can only be cleared (written to zero) during the first 64 E-clock cycles after reset. Once the bits are cleared, the associated EEPROM section and/or the CONFIG register can be programmed or erased in the normal manner. The EEPROM is visible only if the EEON bit in the CONFIG register is set. In the test or bootstrap modes, bits of the BPROT register can be set or cleared at any time. In either single-chip or expanded mode, BPROT register bits can be written back to one anytime after the first 64 E-clock cycles to protect the EEPROM and/or the CONFIG register. However, these bits can only be cleared again in the test or bootstrap modes.

	7	6	5	4	3	2	1	0	
$1035	0	0	0	PTCON	BPRT3	BPRT2	BPRT1	BPRT0	BPROT
RESET:	0	0	0	1	1	1	1	1	

Bits 7–5 — Not implemented; always read zero.

PTCON — Protect CONFIG Register

1 = Programming/erasure of the CONFIG register disabled.
0 = Programming/erasure of the CONFIG register allowed.

BPRT3–BPRT0 — Block Protect

1 = A set bit protects a block of EEPROM against programming or erasure.
0 = A cleared bit permits programming or erasure of the associated block.

Bit	Block Protected	Block Size
BPRT0	$xE00–xE1F	32 Bytes
BPRT1	$xE20–xE5F	64 Bytes
BPRT2	$xE60–xEDF	128 Bytes
BPRT3	$xEE0–xFFF	288 Bytes

8.2 EEPROM PROGRAMMING CONTROL REGISTER (PPROG)

This register is used to control the programming and erasure of the EEPROM. PPROG is cleared on reset so that the EEPROM is configured for normal read.

	7	6	5	4	3	2	1	0	
$103B	ODD	EVEN	0	BYTE	ROW	ERASE	EELAT	EEPGM	PPROG
RESET:	0	0	0	0	0	0	0	0	

ODD — Program Odd Rows (Test)

EVEN — Program Even Rows (Test)

Bit 5 — Not implemented; always reads zero.

ROW and BYTE — Row Erase Select Bit and Byte Erase Select

The value of these bits read together determines the manner in which EEPROM is erased. The bits encode as follows:

Byte	Row	Action
0	0	Bulk Erase (All 512 Bytes)
0	1	Row Erase (16 Bytes)
1	0	Byte Erase
1	1	Byte Erase

ERASE — Erase Mode Select

1 = Erase mode selected.

0 = Normal read or program mode selected.

EELAT — EEPROM Latch Control

1 = EEPROM address and data bus configured for program or erase mode.

0 = EEPROM address and data bus configured for read mode.

EEPGM — EEPROM Programming Voltage Enable

1 = Programming voltage on.

0 = Programming voltage off.

CAUTION

A strict register access sequence must be followed to allow successful programming and erase operations. The following procedures for modifying the EEPROM and CONFIG register detail the sequence. If an attempt is made to set both the EELAT and EEPGM bits in the same write cycle and this attempt occurs before the required write cycle with the EELAT bit set, then neither is set. If a write to an EEPROM address is performed while the EEPGM bit is set, the write is ignored, and the programming operation in progress is not disturbed. If no EEPROM address is written between when

EELAT is set and EEPGM is set, then no program or erase operation occurs. These safeguards were included to prevent accidental EEPROM changes in cases of program runaway.

8.3 ERASING THE EEPROM

An erased EEPROM byte reads as $FF. Erasure of the EEPROM is controlled by a sequence of bit settings in the PPROG register. The appropriate bits in the BPROT register must also be cleared before the EEPROM can be changed. Programs can be written to perform bulk, row, or byte erase. When a bulk or row erase is used, however, only those bytes of EEPROM not protected by BPROT are erased. Other MCU operations can continue to be performed during EEPROM erase, provided that the operations do not include reads of data from EEPROM.

To erase EEPROM, the user should first ensure that the proper bits of the BPROT register are cleared, then take the following steps using the PPROG register:

1. Write to PPROG with the ERASE, EELAT, and appropriate BYTE and ROW bits set.
2. Write to the appropriate EEPROM address with any data. Row erase ($xE00-$xE0F, $xE10-$xE1F . . . $xFF0-$xFFF) only requires a write to any location in the row. Bulk erase is accomplished by writing to any location in the array.
3. Write to PPROG with ERASE, EELAT, EEPGM, and the appropriate BYTE and ROW bits set.
4. Delay for 10 ms or more as appropriate.
5. Clear the EEPGM bit in PPROG to turn off the high voltage.

8.4 PROGRAMMING THE EEPROM

The programmed state of an EEPROM bit is zero. If the location to be programmed contains zeros in the bits that need to be changed to ones, the zeros must be erased by a separate erase operation before programming. During programming, the ROW and BYTE bits of PPROG are not used. If the E-clock frequency is 1 MHz or less, the CSEL bit of the OPTION register must be set, permitting the operation to be clocked by the on-chip RC oscillator. Other MCU operations can continue to be performed during EEPROM programming, provided the operations do not include reads of data from EEPROM.

To program EEPROM, the user should ensure that the proper bits of the BPROT register are cleared, then take the following steps using the PPROG register:

1. Write to PPROG with the EELAT bit set.
2. Write data to the desired address.
3. Write to PPROG with the EELAT and EEPGM bits set.
4. Delay for 10 ms or more as appropriate.
5. Clear the EEPGM bit in PPROG to turn off the high voltage.

8.5 CONFIGURATION CONTROL REGISTER (CONFIG)

The configuration control register controls the presence of EEPROM in the memory map and enables the COP watchdog system. CONFIG is unique in that it is implemented in EEPROM cells that are latched into a control register at reset. In test mode, CONFIG is directly writable. In normal operating modes, the values that are latched at reset apply until the next reset. Bit PTCON of the BPROT register must first be cleared in order to program or erase CONFIG.

	7	6	5	4	3	2	1	0	
$103F	EE3	EE2	EE1	EE0	1	NOCOP	1	EEON	CONFIG
RESET:	U	U	U	U	1	U	1	U	

EE3–EE0 — EEPROM Map Position

EEPROM is located at $xE00–xFFF, where 'x' is the hexadecimal value represented by these four bits. In single-chip and bootstrap modes, EEPROM is forced to $FE00–FFFF, regardless of the state of these bits.

Bits 3 and 1 — Not implemented; always read one.

NOCOP — Computer Operating Properly System Disable

1 = COP system is disabled; no forced reset on timeout.
0 = COP system is enabled; reset forced on timeout.

EEON — EEPROM Enable

This bit is forced to a one in single-chip and bootstrap modes. In test mode, EEON is forced to a zero out of reset. A simple write to set the bit after reset will be required to enable the EEPROM in test mode. In expanded-nonmultiplexed mode, the values of this bit are as follows:

1 = EEPROM is present in the memory map at the location indicated by bits EE3–EE0.
0 = EEPROM is disabled from the memory map.

8.5.1 Erasing the CONFIG Register

Erasing the CONFIG register follows the same procedures used in erasing the EEPROM, including bulk, byte, and row erase. However, bulk erasure of CONFIG causes erasure of not only CONFIG but also of the entire 512 bytes of EEPROM, as long as the BPROT protection is not activated. The ability to byte erase CONFIG is not present on all members of the M68HC11 Family of MCUs.

The CONFIG register can be programmed or erased while the MCU is operating in any mode, depending on the setting of the PTCON bit of BPROT register. Some members of the MC68HC11 Family of MCUs do not allow changes to CONFIG in normal modes.

8.5.2 Programming the CONFIG Register

Programming the CONFIG register follows the same procedures and restrictions encountered in programming the EEPROM, except that the CONFIG address is used. A read of CONFIG after programming will return the value latched at reset, not the new value.

SECTION 9
SERIAL COMMUNICATIONS INTERFACE

The SCI allows the MCU to be efficiently interfaced with peripheral devices that require an asynchronous serial data format. The SCI uses a standard nonreturn-to-zero (NRZ) format with a variety of baud rates derived from the crystal clock circuit. Interfacing is accomplished using port D pins. PD0 is used for receive data (RxD), and PD1, for transmit data (TxD). The baud-rate generation circuit contains a programmable prescaler and divider clocked by the E clock. Figure 9-1 shows a block diagram of the SCI.

9.1 DATA FORMAT

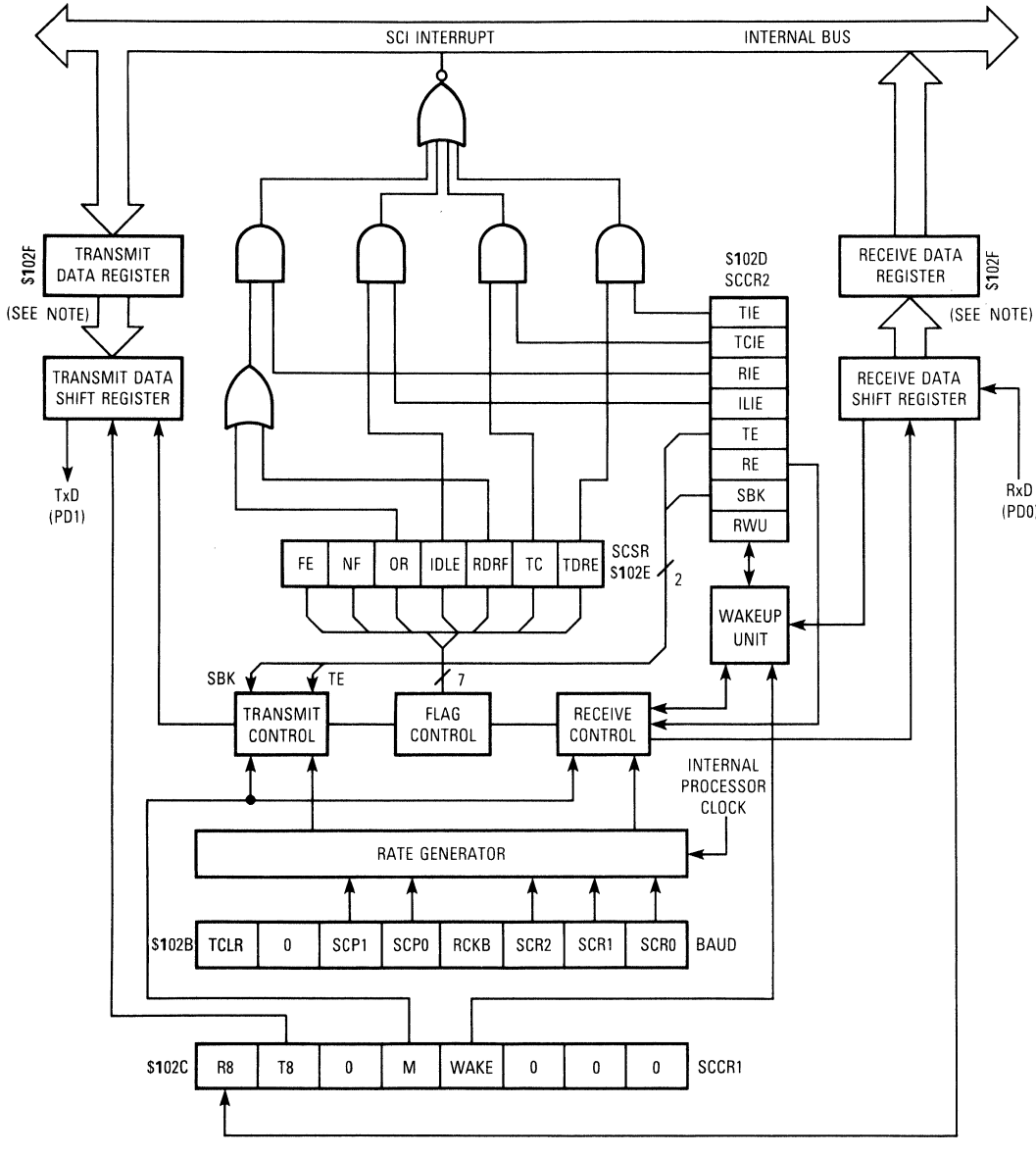
The serial data format requires the following:

1. An idle line in the high state prior to transmission/reception of a message
2. A start bit (logic zero) that is transmitted/received, indicating the start of each character
3. Data that is transmitted and received LSB first
4. A stop bit (logic one) used to indicate the end of a frame (A frame consists of a start bit, a character of eight or nine data bits, and a stop bit.)
5. A break, which is defined as the transmission or reception of a logic zero for some multiple number of frames

Selection of the word length is controlled by the M bit of the SCI control register 1 (SCCR1).

9.2 TRANSMIT OPERATION

The SCI transmitter includes a parallel transmit data register, called the SCDR, and a serial shift register that puts data from the SCDR into serial form. The contents of the serial shift register can only be written through the SCDR. This double-buffered system allows a character to be shifted out serially while another character is waiting in the SCDR to be transferred into the serial shift



NOTE: The serial communications data register (SCDR) is controlled by the internal R/$\overline{W}$ signal. It is the transmit data register when written and received data register when read.

Figure 9-1. SCI Block Diagram

register. The output of the serial shift register is applied to PD1 as long as transmission is in progress or the transmit enable (TE) bit of the serial communication control register (SCCR2) is set.

9.3 RECEIVE OPERATION

In receive operations, the transmit sequence is reversed. Data is received in the serial shift register and is transferred to a parallel receive data register (the SCDR) as a complete word. This double-buffered system allows a character to be shifted in serially while another character is already in the SCDR. An advanced data recovery scheme is used to distinguish valid data from noise in the serial data stream. The data input is selectively sampled to detect receive data, and a majority voting circuit determines the value and integrity of each bit.

9.4 WAKEUP FEATURE

The wakeup feature reduces SCI service overhead in multiple receiver systems. Software for each receiver evaluates the first character(s) of each message. If the message is intended for a different receiver, the SCI can be placed in a sleep mode, stopping the rest of the message from generating requests for service. Whenever a new message begins, logic causes the sleeping receivers to awaken and evaluate the initial character(s) of the new message.

Two methods of wakeup are available, idle-line wakeup or address-mark wakeup. In idle-line wakeup, a sleeping receiver awakens as soon as the RxD line becomes idle. In the address-mark wakeup, a one in the MSB of a character is used to indicate that the message is an address that wakes up all sleeping receivers.

9.5 SCI REGISTERS

The following paragraphs describe the operations of the five addressable registers used in the SCI.

9.5.1 Serial Communications Data Register (SCDR)

The SCDR register is a parallel register that performs two functions. It is the receive data register when it is read and the transmit data register when it

is written. Figure 9-1 shows the SCDR as two separate registers whose single address is $102F.

9.5.2 Serial Communications Control Register 1 (SCCR1)

The SCCR1 register provides the control bits to determine word length and select the method used for the wakeup feature.

	7	6	5	4	3	2	1	0	
$102C	R8	T8	0	M	WAKE	0	0	0	SCCR1
RESET:	U	U	0	0	0	0	0	0	

R8 — Receive Data Bit 8
This bit functions as the ninth serial data bit received when the SCI system is configured for nine data bit operation (the M bit is set).

T8 — Transmit Data Bit 8
This bit functions as the ninth data bit to be transmitted when the SCI system is configured for nine data bit operation (the M bit is set).

Bits 5 and 2–0 — Not implemented; always read zero.

M — SCI Character Length
1 = One start bit, nine data bits, one stop bit — system now configured for nine data bit operation.
0 = One start bit, eight data bits, one stop bit — system now configured for eight data bit operation.

WAKE — Wakeup Method Select
1 = Address Mark.
0 = Idle Line.

9.5.3 Serial Communications Control Register 2 (SCCR2)

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

	7	6	5	4	3	2	1	0	
$102D	TIE	TCIE	RIE	ILIE	TE	RE	RWU	SBK	SCCR2
RESET:	0	0	0	0	0	0	0	0	

TIE — Transmit Interrupt Enable

1 = SCI interrupt requested (if TDRE bit of SCSR is set).
0 = TDRE interrupts disabled.

TCIE — Transmit Complete Interrupt Enable

1 = SCI interrupt requested (if TC bit of SCSR is set).
0 = TC interrupts disabled.

RIE — Receive Interrupt Enable

1 = SCI interrupt requested (if RDRF or OR bit of SCSR is set).
0 = Receive data register full (RDRF) or overrun error (OR) interrupts disabled.

ILIE — Idle-Line Interrupt Enable

1 = SCI interrupt requested (if IDLE bit of SCSR is set).
0 = Idle-line detect (IDLE) interrupts disabled.

TE — Transmit Enable

1 = Transmit shift register output applied to the TxD line.
0 = PD1 pin reverts to general-purpose I/O as soon as current transmitter activity finishes.

RE — Receive Enable

1 = Receiver enabled.
0 = Receiver disabled and RDRF, IDLE, OR, NF, and FE interrupts inhibited.

RWU — Receiver Wakeup

When set by the user's software, this bit puts the receiver to sleep and enables the wakeup function. If the WAKE bit of SCCR1 is logic zero, RWU is cleared by the SCI logic after receiving 10 (SCCR1 M bit = 0) or 11 (SCCR1 M bit = 1) consecutive ones. If WAKE is logic one, RWU is cleared by the SCI logic after receiving a data word whose MSB is set.

SBK — Send Break

If this bit is toggled (set and cleared), the transmitter sends 10 or 11 consecutive zeros, depending on the setting of the M bit of SCCR1, then reverts to idle or sending data. If SBK remains set, the transmitter continually sends whole frames of zeros in groups of 10 or 11 until SBK is cleared.

9.5.4 Serial Communications Status Register (SCSR)

The SCSR provides inputs to the interrupt logic circuits for generation of the SCI system interrupt.

	7	6	5	4	3	2	1	0	
$102E	TDRE	TC	RDRF	IDLE	OR	NF	FE	0	SCSR
RESET:	1	1	0	0	0	0	0	0	

TDRE — Transmit Data Register Empty

This bit is set automatically as the contents of SCDR are transferred to the serial shift register in a transmit operation. New data is not transmitted unless the SCSR register is read before writing to SCDR. TDRE is cleared by a read of SCSR followed by a write to SCDR.

TC — Transmit Complete

This bit is set automatically when all data frame, preamble, or break condition transmissions are complete. When TC is set, the serial line goes idle (continuous mark). TC is cleared by a read of SCSR followed by a write to SCDR.

RDRF — Receive Data Register Full

This bit is set automatically when a character is transferred from the serial shift register to SCDR in a receive operation. RDRF is cleared by a read of SCSR followed by a read of SCDR.

IDLE — Idle-Line Detect

This bit is inhibited while the RWU bit of SCCR2 is set. IDLE is set automatically when the receiver serial input detects an idle line. This bit is cleared by a read of SCSR followed by a read of SCDR.

OR — Overrun Error

This bit is automatically set when a new character is ready to transfer from the receiver shift register to the SCDR and the SCDR is already full (RDRF bit set). Data transfer is inhibited until this bit is cleared. OR is cleared by a read of SCSR followed by a read of SCDR.

NF — Noise Flag

This bit is automatically set if there is noise on any of the received bits, including the start and stop bits. This bit is not set until the RDRF flag is set. It is not set in the case of an overrun. NF is cleared by a read of SCSR followed by a write to SCDR.

FE — Framing Error

This bit is automatically set when no stop bit is detected in the received data character. The FE bit is set at the same time as the RDRF flag is set. If the byte received causes both framing and overrun errors, only the overrun error is recognized. This flag inhibits further transfers until it is cleared. FE is cleared by a read of SCSR followed by a read of SCDR.

Bit 0 — Not implemented; always reads zero.

9.5.5 Baud-Rate Register (BAUD)

This register is used to select different baud rates that can be used as the rate control for the receiver and transmitter.

	7	6	5	4	3	2	1	0	
$102B	TCLR	0	SCP1	SCP0	RCKB	SCR2	SCR1	SCR0	BAUD
RESET:	0	0	0	0	0	U	U	U	

TCLR — Clear Baud-Rate Counters (Test)

This bit is used to clear the baud-rate counter chain during factory testing. TCLR is zero and cannot be set while the MCU is in normal operating modes.

Bit 6 — Not implemented; always reads zero.

SCP1 and SCP0 — SCI Baud-Rate Prescaler Selects

These bits control a prescaler whose output is the input for a second divider that is controlled by the SCR2–SCR0 bits (see Table 9-1).

RCKB — SCI Baud-Rate Clock Check (Test)

This bit is used during factory testing to enable the exclusive OR of the receiver clock and transmitter clock to be driven out of the TxD pin. RCKB is zero and cannot be set while in normal operating modes.

SCR2–SCR0 — SCI Baud-Rate Selects

These bits select the baud rate for both the transmitter and the receiver. The prescaler output selected by SCP1 and SCP0 is further divided by the setting of these bits (see Table 9-2).

Table 9-1. Prescaler Highest Baud-Rate Frequency Output

SCP Bit		Clock* Divided By	Crystal Frequency (MHz)				
1	0		8.3886	8.0	4.9152	4.0	3.6864
0	0	1	131.072 K Baud	125.000 K Baud	76.80 K Baud	62.50 K Baud	57.60 K Baud
0	1	3	43.690 K Baud	41.666 K Baud	25.60 K Baud	20.833 K Baud	19.20 K Baud
1	0	4	32.768 K Baud	31.250 K Baud	19.20 K Baud	15.625 K Baud	14.40 K Baud
1	1	13	10.082 K Baud	9600 Baud	5.907 K Baud	4800 Baud	4430 Baud

*The internal processor clock

Table 9-2. Transmit Baud-Rate Output for a Given Prescaler Output

SCR Bit			Divided By	Representative Highest Prescaler Baud-Rate Output				
2	1	0		131.072 K Baud	32.768 K Baud	76.80 K Baud	19.20 K Baud	9600 Baud
0	0	0	1	131.072 K Baud	32.768 K Baud	76.80 K Baud	19.20 K Baud	9600 Baud
0	0	1	2	65.536 K Baud	16.384 K Baud	38.40 K Baud	9600 Baud	4800 Baud
0	1	0	4	32.768 K Baud	8.192 K Baud	19.20 K Buad	4800 Baud	2400 Baud
0	1	1	8	16.384 K Baud	4.096 K Baud	9600 Baud	2400 Baud	1200 Baud
1	0	0	16	8.192 K Baud	2.048 K Baud	4800 Baud	1200 Baud	600 Baud
1	0	1	32	4.096 K Baud	1.024 K Baud	2400 Baud	600 Baud	300 Baud
1	1	0	64	2.048 K Baud	512 Baud	1200 Baud	300 Baud	150 Baud
1	1	1	128	1.024 K Baud	256 Baud	600 Baud	150 Baud	75 Baud

SECTION 10
SERIAL PERIPHERAL INTERFACE

The SPI is a high-speed synchronous serial I/O system. The SPI can be used for simple I/O expansion or for allowing several MCUs to be interconnected in a multimaster configuration. Clock phase and polarity are software programmable to allow direct compatibility with a large number of peripheral devices. The SPI system can be configured as either a master or a slave.

Four basic signal lines are associated with the SPI system: the master-out slave-in (MOSI), the master-in slave-out (MISO), the serial clock (SCK), and the slave select ($\overline{SS}$) pins. The SPI signals are assigned to port D bits 5–2 (see Figure 10-1). SPI outputs must have the corresponding bits set in the port D data direction register (DDRD). However, any SPI input line is forced to be an input, regardless of the DDRD bit settings.

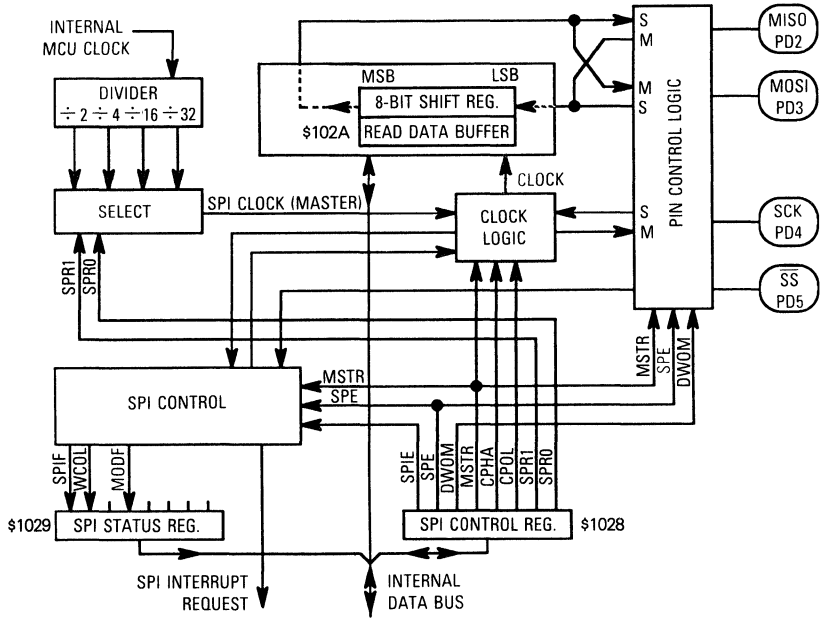


Figure 10-1. SPI Block Diagram

A series of eight SCK clock cycles are generated to synchronize data transfer. When a master device transmits data to a slave device via the MOSI line, the slave responds by sending data to the master device via the MISO line. This exchange implies full-duplex transmission with both data out and data in synchronized to the same clock signal. The byte transmitted is replaced by the byte received, thereby eliminating the need for separate transmit-empty and receiver-full status bits. A single status bit (SPIF) is used to signify that the I/O operation has been completed. Figure 10-1 is a block diagram of the SPI.

The SPI is double buffered on read, but not on write. If a write is attempted during data transfer, the transfer is uninterrupted, and the write is unsuccessful. This condition will cause the write collision (WCOL) bit of the SPSR register to be set. After a data byte is shifted, the SPIF flag of SPSR is set.

In the master mode, the SCK pin is an output. Depending on the CPOL bit of the SPCR register, SCK idles high or low until data is written to the shift register. Once data is written, eight clocks are generated to shift the eight bits of data. Then SCK goes idle again.

In a slave mode, the slave start logic receives a logic low at the $\overline{SS}$ pin and a clock input at the SCK pin. Thus, the slave is synchronized with the master. Data from the master is received serially at the slave MOSI line and loads the 8-bit shift register. Once the 8-bit shift register is loaded, its data is parallel transferred to the read buffer. During a write cycle, data is written into the shift register. The slave then waits for a clock train from the master to shift the data out on the slave's MISO line. Figure 10-2 illustrates the MOSI, MISO, SCK, and $\overline{SS}$ master-slave interconnections.

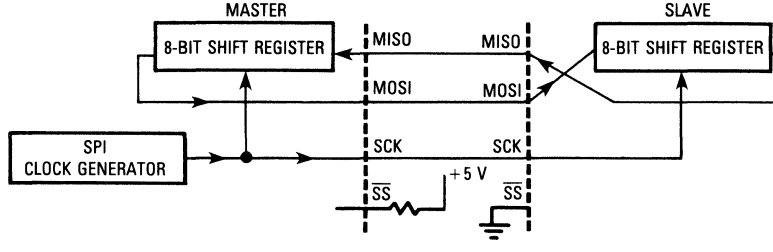


Figure 10-2. SPI Master-Slave Interconnections

10.1 SPI REGISTERS

There are three registers in the SPI that provide control, status, and data storage functions. These registers are described in the following paragraphs.

10.1.1 Serial Peripheral Control Register (SPCR)

	7	6	5	4	3	2	1	0	
$1028	SPIE	SPE	DWOM	MSTR	CPOL	CPHA	SPR1	SPR0	SPCR
RESET:	0	0	0	0	0	0	U	U	

SPIE — Serial Peripheral Interrupt Enable
1 = SPI interrupt requested (if SPIF or MODF of SPSR = 1).
0 = SPI (SPIF) interrupts disabled.

SPE — Serial Peripheral System Enable
1 = SPI system on.
0 = SPI system off.

DWOM — Port D Wired-OR Mode Option
This bit affects all six port D pins together.
1 = Port D outputs act as open-drain outputs.
0 = Port D outputs are normal CMOS outputs.

MSTR — Master Mode Select
1 = Master mode is selected.
0 = Slave mode is selected.

CPOL — Clock Polarity
This bit selects the polarity of the SCK clock. It is used with the CPHA control bit to produce the desired clock-data relationship between master and slave (see Figure 10-3).
1 = SCK line idles high.
0 = SCK line idles low.

CPHA — Clock Phase

This bit selects one of two fundamentally different clock protocols, when considered with the CPOL setting (see Figure 10-3).

1 = Data is sampled midway through the SCK period. $\overline{SS}$ to the slave can be left low between transfers.

0 = Transfer begins when $\overline{SS}$ goes low. Data is sampled on the initial edge of SCK. $\overline{SS}$ to the slave must be negated and then asserted between bytes.

SPR1 and SPR0 — SPI Clock Rate Selects

These two bits select one of four baud rates to be used as SCK if the SPI is set as a master; they have no effect in the slave mode. Their value decodes as follows:

SPR1	SPR0	E Clock Divide By	Frequency at E=2 MHz (Baud Rate)
0	0	2	1.0 MHz
0	1	4	500 kHz
1	0	16	125 kHz
1	1	32	62.5 kHz

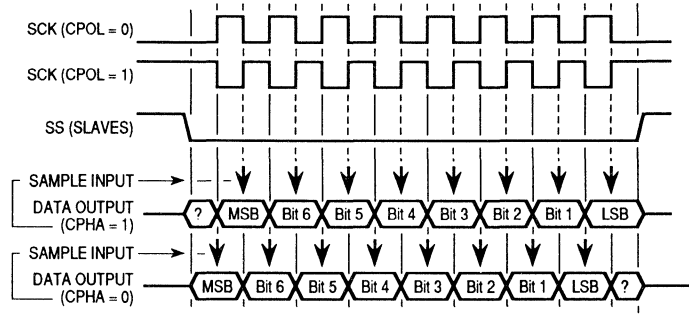


Figure 10-3. SPI Data Clock Timing Diagram

10.1.2 Serial Peripheral Status Register (SPSR)

	7	6	5	4	3	2	1	0	
$1029	SPIF	WCOL	0	MODF	0	0	0	0	SPSR
RESET:	0	0	0	0	0	0	0	0	

SPIF — SPI Transfer Complete Flag

This flag is automatically set when data transfer is complete between processor and external device. The flag is cleared by a read of SPSR followed by a read or write of SPDR.

WCOL — Write Collision

This bit is set automatically when an attempt is made to write to the SPI data register while data is being transferred. The bit is cleared by a read of SPSR followed by a read or write of SPDR.

Bits 5 and 3–0 — Not implemented; always read zero.

MODF — Mode Fault

This bit indicates the possibility of a multimaster conflict for system control and allows a proper exit from system operation to a reset or default system state. MODF is only set when a master device has its $\overline{SS}$ pin pulled low. The SPI system is then terminated by the following actions:

1. The DDRD bits are cleared to disable SPI outputs,
2. The MSTR bit is cleared to reconfigure the SPI as slave,
3. The SPE bit is cleared to disable the SPI, and
4. An SPI interrupt is generated if SPIF is set.

This bit is cleared by a read of SPSR followed by a write to SPCR. The SPE and MSTR control bits can be restored to their original state by user software once the MODF bit has been cleared. The DDRD bits must be restored after a mode fault.

10.1.3 Serial Peripheral Data I/O Register (SPDR)

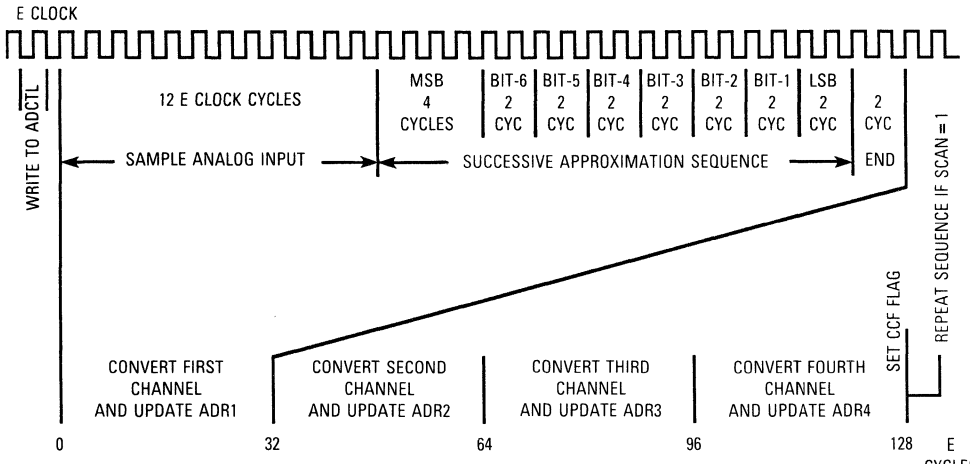
Located at address $1102A, the SPDR is used to transmit and receive data on the serial bus. A write to this register in a master initiates transmission/reception of another byte. A slave writes data to this register for later transmission to a master. When transmission is complete, the SPIF status bit is set in the SPSR register of both the master and slave device. When a read is performed on the SPDR, a buffer is actually being read. The first SPIF must be cleared by the time a second transfer of data from the shift register to the read buffer is initiated, or an overrun condition will exist. In case of an overrun, the byte causing the overrun is lost.

SECTION 11
ANALOG-TO-DIGITAL CONVERTER

The MC68HC11F1 contains an eight-channel, multiplexed-input, successive-approximation, A/D converter with sample and hold. Two dedicated lines (V_{RL} and V_{RH}) are provided for the reference supply voltage input. These pins are used instead of the device power pins to increase the accuracy of the A/D conversion.

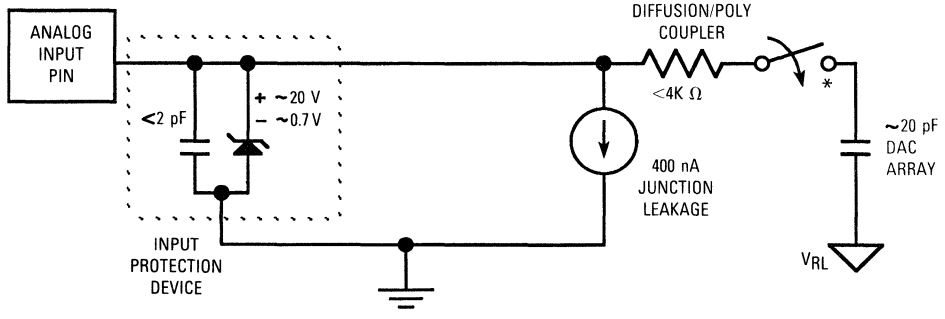
The 8-bit A/D conversions of the MCU are accurate to within ±1 LSB, using the system E clock (±1/2 LSB quantizing errors and ±1/2 LSB all other errors combined). Each conversion is accomplished in 32 MCU E-clock cycles. The CSEL bit of the OPTION register allows selection of an internal RC oscillator that allows the A/D to be used with very low MCU clock rates. A typical conversion cycle requires 16 μs to complete at a bus frequency of 2 MHz.

The A/D converter is ratiometric with no reserved codes. An input voltage equal to V_{RL} converts to $00, and an input voltage equal to V_{RH} converts to $FF (full scale), with no overflow indication. Figure 11-1 shows the detailed sequence for a set of four conversions. This sequence begins one E-clock cycle after a write to the A/D control/status register (ADCTL). Figure 11-2 shows a model of the port E A/D channel inputs. This model is useful for understanding the effects of external circuitry on the accuracy of A/D conversions.



NOTE: Conversion results are built up in the SAR and transferred into ADRx during the END period.
The CCF status flag is set during the END period of the fourth conversion after a write to ADCTL.
This figure assumes CSEL in the OPTION register is 0 so that the E clock is acting as the conversion clock.
IF MULT = 0 all four conversions in the sequence are performed on the same analog channel.

Figure 11-1. A/D Conversion Sequence



*This analog switch is closed only during the 12-cycle sample time.

Figure 11-2. A/D Pin Model

11.1 CHANNEL ASSIGNMENTS

A multiplexer allows the single A/D converter to select one of 16 analog signals. Eight channels correspond to port E input lines to the MCU. Four channels are for internal reference points or test functions, and four channels are reserved for future use. Table 11-1 shows the signals selected by the four channel-select control bits.

Table 11-1. A/D Channel Assignments

CD	CC	CB	CA	Channel Signal	Result in ADRx if MULT = 1
0	0	0	0	AN0	ADR1
0	0	0	1	AN1	ADR2
0	0	1	0	AN2	ADR3
0	0	1	1	AN3	ADR4
0	1	0	0	AN4	ADR1
0	1	0	1	AN5	ADR2
0	1	1	0	AN6	ADR3
0	1	1	1	AN7	ADR4
1	0	0	0	Reserved	ADR1
1	0	0	1	Reserved	ADR2
1	0	1	0	Reserved	ADR3
1	0	1	1	Reserved	ADR4
1	1	0	0	V_{RH} Pin*	ADR1
1	1	0	1	V_{RL} Pin*	ADR2
1	1	1	0	$(V_{RH})/2$*	ADR3
1	1	1	1	Reserved*	ADR4

*This group of channels is used during factory testing.

11.2 SINGLE-CHANNEL OPERATION

There are two variations of single-channel operation. In the first variation (SCAN = 0), the single selected channel is converted four consecutive times with the first result being stored in A/D result register 1 (ADR1), the second in ADR2, the third in ADR3, and the fourth in ADR4. Once the fourth conversion is complete, all conversion activity stops until a new conversion command is written to the ADCTL.

In the second variation (SCAN = 1), conversions continue to be performed on the selected channel with the fifth conversion being stored in ADR1, overwriting the first result. The sixth conversion overwrites the contents of ADR2, etc.

11.3 MULTIPLE-CHANNEL OPERATION

There are two variations in multiple-channel operation. In the first variation (SCAN = 0), the selected group of four channels are converted, each channel being converted once. The first result is stored in register ADR1, the second in ADR2, etc. After the fourth conversion is complete, all conversion activity stops until a new conversion command is written to the ADCTL.

In the second variation (SCAN = 1), conversions continue to be performed on the selected group of channels with the fifth conversion being stored in ADR1, overwriting the first result. The sixth conversion overwrites ADR2, etc.

11.4 A/D CONTROL/STATUS REGISTER (ADCTL)

All the bits of this register may be read or written except bit 7, which is a read-only status indicator, and bit 6, which always reads as zero.

	7	6	5	4	3	2	1	0	
$1030	CCF	0	SCAN	MULT	CD	CC	CB	CA	ADCTL
RESET:	0	0	U	U	U	U	U	U	

CCF — Conversions Complete Flag

This read-only status indicator is set when all four A/D result registers contain valid conversion results. Each time ADCTL is written, this bit is automatically cleared, and a conversion sequence is started. In the continuous modes, conversions continue in a round-robin fashion, and the result registers continue to be updated with current data even though the CCF bit remains set.

NOTE

The user must write to register ADCTL to initiate conversion. To abort a conversion in progress, a write to the ADCTL register initiates a new conversion sequence immediately.

Bit 6 — Not implemented; always reads zero.

SCAN — Continuous Scan Control

When this control bit is clear, the four requested conversions are performed once to fill the four result registers. When SCAN is set, conversions continue in a round-robin fashion with the result registers being updated as data becomes available.

MULT — Multiple-Channel/Single-Channel Control

When this bit is clear, the A/D system is configured to perform four consecutive conversions on the single channel specified by the four channel-select bits (CD–CA). When MULT is set, the A/D system is configured to perform a conversion on each of the four channels, where each result register corresponds to one channel.

NOTE

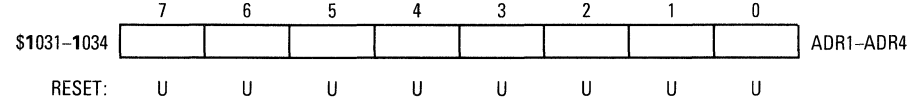
When the multiple-channel continuous scan mode is used, extra care is needed in the design of circuitry driving the A/D inputs. Refer to the Figures 11-1 and 11-2 while reading the following information. The charge on the capacitive D/A converter (DAC) array prior to the sample time is related to the voltage on the previously converted channel. A charge-share situation exists between the internal DAC capacitance and the external circuit capacitance. Although the amount of charge involved is small, the rate at which it is repeated is every 64 μs for an E clock of 2 MHz. The RC charging rate of the external circuit must be balanced against this charge-sharing effect to avoid accuracy errors.

CD–CA — Channel Selects (D–A)

These four bits are used to select one of 16 A/D channels. When a multiple-channel mode is selected (MULT = 1), the two least significant channel select bits (CB and CA) have no meaning. The CD and CC bits specify which group of four channels are to be converted. The channels selected by the four channel-select control bits are shown in Table 11-1.

11.5 A/D RESULT REGISTERS (ADR1–ADR4)

The A/D result registers are read-only registers used to hold an 8-bit conversion result. Writes to these registers have no effect.



Data in the A/D registers is valid when the CCF bit of ADCTL is set, indicating a conversion sequence is complete. If conversion results are needed sooner, refer to Figure 10-3. For example, the ADR1 result is valid 33 cycles after an ADCTL write. Refer to the A/D channel assignments shown in Table 11-1 for the relationship between the channels and the result registers.

11.6 A/D POWERUP AND CLOCK SELECT

A/D powerup is controlled by bit 7 (ADPU) of the OPTION register. When ADPU is cleared, power to the A/D system is disabled. When ADPU is set, the A/D system is enabled. A delay of as much as 100 μs is required after turning on the A/D converter to allow the analog bias voltages to stabilize.

Clock select is controlled by bit 6 (CSEL) of the OPTION register. When CSEL is cleared, the A/D system uses the system E clock. When CSEL is set, the A/D system uses an internal RC clock source that runs at about 1.5 MHz. The MCU E clock is not suitable to drive the A/D system if it is operating below 750 kHz. In this case, the RC internal clock should be selected. A delay of 10 ms is required after changing the CSEL from zero to one to allow the RC oscillator to start and internal bias voltages to settle. The CSEL bit also enables a separate RC oscillator to drive the EEPROM charge pump.

When the A/D system is operating with the MCU E clock, all switching and comparator operations are synchronized to the MCU clocks. This allows the comparator results to be sampled at quiet clock times to minimize noise errors. Because the internal RC oscillator is asynchronous to the MCU clock, noise affects A/D results more while CSEL equals one.

SECTION 12
PROGRAMMING INFORMATION

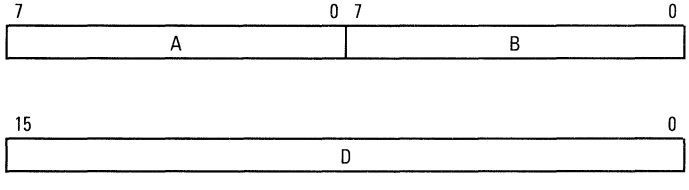
This section depicts the MC68HC11F1 CPU registers, instruction set, and addressing modes.

12.1 PROGRAMMING MODEL

The M68HC11 Family of MCUs has eight central processing unit (CPU) registers available to the programmer. Each register is explained in the following paragraphs.

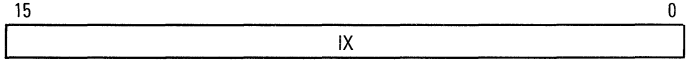
12.1.1 Accumulators (A, B, and D)

Accumulators A and B are general-purpose 8-bit registers used to hold operands and the results of arithmetic calculations or data manipulations. These two accumulators are treated as a single double-byte accumulator (D accumulator) for some instructions.



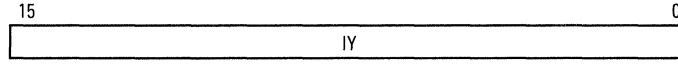
12.1.2 Index Register X (IX)

The IX register is a 16-bit register used for the indexed addressing mode. It provides a 16-bit value that can be added to an 8-bit offset provided in an instruction to create an effective address. The IX register can also be used as either a counter or temporary storage area.



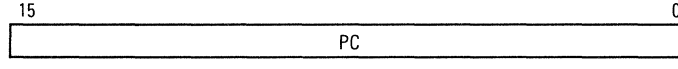
12.1.3 Index Register Y (IY)

The IY register is a 16-bit register used for the indexed addressing mode, similar to the IX register. However, most instructions using the IY register are two-byte opcodes and require an extra byte of machine code and an extra cycle of execution time. The IY register can also be used as a counter or temporary storage area.



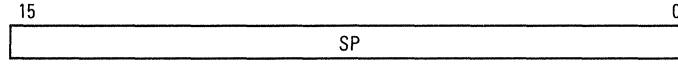
12.1.4 Program Counter (PC)

The PC is a 16-bit register that contains the address of the next instruction to be executed.



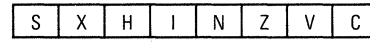
12.1.5 Stack Pointer (SP)

The SP is a 16-bit register that contains the address of the next free location on the stack. The stack is configured as a sequence of last-in first-out (LIFO) read/write registers that allow important data to be stored during interrupts and subroutine calls. Each time a new byte is added to the stack, the SP is decremented. Each time a byte is removed, the SP is incremented.



12.1.6 Condition Code Register (CCR)

The CCR is an 8-bit register in which five bits are used to indicate the results of the instruction just executed and three bits are mask bits for interrupt and stop. These bits can be individually tested by a program, and specific action can be taken as a result of their state. Each bit is explained in the following paragraphs.



12.1.6.1 CARRY/BORROW (C). When set, the C bit indicates that a carry or borrow out of the arithmetic logic unit (ALU) occurred during the last arithmetic operation. This bit is also affected during shift and rotate instructions.

12.1.6.2 OVERFLOW (V). The V bit is set if an arithmetic overflow occurred as a result of the operation; otherwise, the V bit is cleared.

12.1.6.3 ZERO (Z). When set, the Z bit indicates the result of the last arithmetic, logical, or data manipulation operation was zero.

12.1.6.4 NEGATIVE (N). When set, the N bit indicates that the result of the last arithmetic, logical, or data manipulation operation was negative.

12.1.6.5 INTERRUPT MASK (I). The I bit is set by either hardware or program instruction to disable (mask) all maskable interrupt sources, both external or internal.

12.1.6.6 HALF-CARRY (H). The H bit is set during ADD, ABA, and ADC operations to indicate that a carry occurred between bits 3 and 4. This bit is mainly useful in binary-coded decimal (BCD) calculations.

12.1.6.7 X INTERRUPT MASK (X). The X bit is set only by hardware ($\overline{\text{RESET}}$ or $\overline{\text{XIRQ}}$) and is cleared only by the program instructions transfer A to CC register (TAP), or return from interrupt (RTI).

12.1.6.8 STOP DISABLE (S). Under program control, the S bit is set to disable the STOP instruction. It is cleared to enable the STOP instruction. The STOP instruction is treated as no operation (NOP) if the S bit is set.

12.2 INSTRUCTION SET

The CPU of the MC68HC11F1 is basically a proper extension of the MC6801 CPU. In addition to its ability to execute all M6800 and M6801 instructions, 91 new opcodes are provided by the paged opcode map. Major functional additions include a second 16-bit index register (the Y register), two types of 16-by-16 divide instructions, STOP and WAI instructions, and bit manipulation instructions.

Table 12-1 shows all the MC68HC11F1 instructions in all possible addressing modes. For each instruction, the operand construction is shown as well as the number of machine code bytes and the execution time in CPU E-clock cycles. Definitions of "special ops" shown in the Boolean Expression column of this table are found in Figure 12-1.

Table 12-1. Instructions, Addressing Modes, and Execution Times (Sheet 1 of 7)

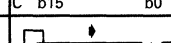
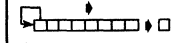
Source Form(s)	Operation	Boolean Expression	Addressing Mode for Operand	Machine Coding (Hexadecimal)		Bytes	Cycle	Condition Codes
				Opcode	Operand(s)			S X H I N Z V C
ABA	Add Accumulators	A+B ⇨ A	INH	1B		1	2	— — ↕ — ↕ ↕ ↕ ↕
ABX	Add B to X	IX+00:B ⇨ IX	INH	3A		1	3	— — — — — — — —
ABY	Add B to Y	IY+00:B ⇨ IY	INH	18 3A		2	4	— — — — — — — —
ADCA (opr)	Add with Carry to A	A+M+C ⇨ A	A IMM	89	ii	2	2	— — ↕ — ↕ ↕ ↕ ↕
			A DIR	99	dd	2	3	
			A EXT	B9	hh ll	3	4	
			A IND,X	A9	ff	2	4	
			A IND,Y	18 A9	ff	3	5	
ADCB (opr)	Add with Carry to B	B+M+C ⇨ B	B IMM	C9	ii	2	2	— — ↕ — ↕ ↕ ↕ ↕
			B DIR	D9	dd	2	3	
			B EXT	F9	hh ll	3	4	
			B IND,X	E9	ff	2	4	
			B IND,Y	18 E9	ff	3	5	
ADDA (opr)	Add Memory to A	A+M ⇨ A	A IMM	8B	ii	2	2	— — ↕ — ↕ ↕ ↕ ↕
			A DIR	9B	dd	2	3	
			A EXT	BB	hh ll	3	4	
			A IND,X	AB	ff	2	4	
			A IND,Y	18 AB	ff	3	5	
ADDB (opr)	Add Memory to B	B+M ⇨ B	B IMM	CB	ii	2	2	— — ↕ — ↕ ↕ ↕ ↕
			B DIR	DB	dd	2	3	
			B EXT	FB	hh ll	3	4	
			B IND,X	EB	ff	2	4	
			B IND,Y	18 EB	ff	3	5	
ADDD (opr)	Add 16-Bit to D	D+M:M+1 ⇨ D	IMM	C3	jj kk	3	4	— — — — ↕ ↕ ↕ ↕
			DIR	D3	dd	2	5	
			EXT	F3	hh ll	3	6	
			IND,X	E3	ff	2	6	
			IND,Y	18 E3	ff	3	7	
ANDA (opr)	AND A with Memory	A•M ⇨ A	A IMM	84	ii	2	2	— — — — ↕ ↕ 0 —
			A DIR	94	dd	2	3	
			A EXT	B4	hh ll	3	4	
			A IND,X	A4	ff	2	4	
			A IND,Y	18 A4	ff	3	5	
ANDB (opr)	AND B with Memory	B•M ⇨ B	B IMM	C4	ii	2	2	— — — — ↕ ↕ 0 —
			B DIR	D4	dd	2	3	
			B EXT	F4	hh ll	3	4	
			B IND,X	E4	ff	2	4	
			B IND,Y	18 E4	ff	3	5	
ASL (opr)	Arithmetic Shift Left		EXT	78	hh ll	3	6	— — — — ↕ ↕ ↕ ↕
			IND,X	68	ff	2	6	
			IND,Y	18 68	ff	3	7	
ASLA			A INH	48		1	2	
ASLB			B INH	58		1	2	
ASLD	Arithmetic Shift Left Double		INH	05		1	3	— — — — ↕ ↕ ↕ ↕
ASR (opr)	Arithmetic Shift Right		EXT	77	hh ll	3	6	— — — — ↕ ↕ ↕ ↕
			IND,X	67	ff	2	6	
			IND,Y	18 67	ff	3	7	
ASRA			A INH	47		1	2	
ASRB			B INH	57		1	2	
BCC (rel)	Branch if Carry Clear	? C=0	REL	24	rr	2	3	— — — — — — — —
BCLR (opr) (msk)	Clear Bit(s)	M•($\overline{mm}$) ⇨ M	DIR	15	dd mm	3	6	— — — — ↕ ↕ 0 —
			IND,X	1D	ff mm	3	7	
			IND,Y	18 1D	ff mm	4	8	
BCS (rel)	Branch if Carry Clear	? C=1	REL	25	rr	2	3	— — — — — — — —
BEQ (rel)	Branch if=Zero	? Z=1	REL	27	rr	2	3	— — — — — — — —

Table 12-1. Instructions, Addressing Modes, and Execution Times (Sheet 2 of 7)

Source Form(s)	Operation	Boolean Expression	Addressing Mode for Operand	Machine Coding (Hexadecimal)		Bytes	Cycle	Condition Codes
				Opcode	Operand(s)			S X H I N Z V C
BGE (rel)	Branch if ≥ Zero	? N ⊕ V = 0	REL	2C	rr	2	3	— — — — — — — —
BGT (rel)	Branch if > Zero	? Z + (N ⊕ V) = 0	REL	2E	rr	2	3	— — — — — — — —
BHI (rel)	Branch if Higher	? C + Z = 0	REL	22	rr	2	3	— — — — — — — —
BHS (rel)	Branch if Higher or Same	? C = 0	REL	24	rr	2	3	— — — — — — — —
BITA (opr)	Bit(s) Test A with Memory	A•M	A IMM	85	ii	2	2	— — — — ↕ ↕ 0 —
			A DIR	95	dd	2	3	
			A EXT	B5	hh ll	3	4	
			A IND,X	A5	ff	2	4	
			A IND,Y	18 A5	ff	3	5	
BITB (opr)	Bit(s) Test B with Memory	B•M	B IMM	C5	ii	2	2	— — — — ↕ ↕ 0 —
			B DIR	D5	dd	2	3	
			B EXT	F5	hh ll	3	4	
			B IND,X	E5	ff	2	4	
			B IND,Y	18 E5	ff	3	5	
BLE (rel)	Branch if ≤ Zero	? Z + (N ⊕ V) = 1	REL	2F	rr	2	3	— — — — — — — —
BLO (rel)	Branch if Lower	? C = 1	REL	25	rr	2	3	— — — — — — — —
BLS (rel)	Branch if Lower or Same	? C + Z = 1	REL	23	rr	2	3	— — — — — — — —
BLT (rel)	Branch If < Zero	? N ⊕ V = 1	REL	2D	rr	2	3	— — — — — — — —
BMI (rel)	Branch if Minus	? N = 1	REL	2B	rr	2	3	— — — — — — — —
BNE (rel)	Branch if Not = Zero	? Z = 0	REL	26	rr	2	3	— — — — — — — —
BPL (rel)	Branch if Plus	? N = 0	REL	2A	rr	2	3	— — — — — — — —
BRA (rel)	Branch Always	? 1 = 1	REL	20	rr	2	3	— — — — — — — —
BRCLR(opr) (msk) (rel)	Branch if Bit(s) Clear	? M•mm = 0	DIR	13	dd mm rr	4	6	— — — — — — — —
			IND,X	1F	ff mm rr	4	7	
			IND,Y	18 1F	ff mm rr	5	8	
BRN (rel)	Branch Never	? 1 = 0	REL	21	rr	2	3	— — — — — — — —
BRSET(opr) (msk) (rel)	Branch if Bit(s) Set	? $(\overline{M})$•mm = 0	DIR	12	dd mm rr	4	6	— — — — — — — —
			IND,X	1E	ff mm rr	4	7	
			IND,Y	18 1E	ff mm rr	5	8	
BSET (opr) (msk)	Set Bit(s)	M + mm ⇩ M	DIR	14	dd mm	3	6	— — — — ↕ ↕ 0 —
			IND,X	1C	ff mm	3	7	
			IND,Y	18 1C	ff mm	4	8	
BSR (rel)	Branch to Subroutine	See Special Ops	REL	8D	rr	2	6	— — — — — — — —
BVC (rel)	Branch if Overflow Clear	? V = 0	REL	28	rr	2	3	— — — — — — — —
BVS (rel)	Branch if Overflow Set	? V = 1	REL	29	rr	2	3	— — — — — — — —
CBA	Compare A to B	A − B	INH	11		1	2	— — — — ↕ ↕ ↕ ↕
CLC	Clear Carry Bit	0 ⇩ C	INH	0C		1	2	— — — — — — — 0
CLI	Clear Interrupt Mask	0 ⇩ I	INH	0E		1	2	— — — 0 — — — —
CLR (opr)	Clear Memory Byte	0 ⇩ M	EXT	7F	hh ll	3	6	— — — — 0 1 0 0
			IND,X	6F	ff	2	6	
			IND,Y	18 6F	ff	3	7	
CLRA	Clear Accumulator A	0 ⇩ A	A INH	4F		1	2	— — — — 0 1 0 0
CLRB	Clear Accumulator B	0 ⇩ B	B INH	5F		1	2	— — — — 0 1 0 0
CLV	Clear Overflow Flag	0 ⇩ V	INH	0A		1	2	— — — — — — 0 —
CMPA (opr)	Compare A to Memory	A − M	A IMM	81	ii	2	2	— — — — ↕ ↕ ↕ ↕
			A DIR	91	dd	2	3	
			A EXT	B1	hh ll	3	4	
			A IND,X	A1	ff	2	4	
			A IND,Y	18 A1	ff	3	5	

Table 12-1. Instructions, Addressing Modes, and Execution Times (Sheet 3 of 7)

Source Form(s)	Operation	Boolean Expression	Addressing Mode for Operand	Machine Coding (Hexadecimal)		Bytes	Cycle	Condition Codes
				Opcode	Operand(s)			S X H I N Z V C
CMPB (opr)	Compare B to Memory	B–M	B IMM	C1	ii	2	2	— — — — ↕ ↕ ↕ ↕
			B DIR	D1	dd	2	3	
			B EXT	F1	hh ll	3	4	
			B IND,X	E1	ff	2	4	
			B IND,Y	18 E1	ff	3	5	
COM (opr)	Ones Complement Memory Byte	$FF–M ⇩ M	EXT	73	hh ll	3	6	— — — — ↕ ↕ 0 1
			IND,X	63	ff	2	6	
			IND,Y	18 63	ff	3	7	
COMA	Ones Complement A	$FF–A ⇩ A	A INH	43		1	2	— — — — ↕ ↕ 0 1
COMB	Ones Complement B	$FF–B ⇩ B	B INH	53		1	2	— — — — ↕ ↕ 0 1
CPD (opr)	Compare D to Memory 16-Bit	D – M:M+1	IMM	1A 83	jj kk	4	5	— — — — ↕ ↕ ↕ ↕
			DIR	1A 93	dd	3	6	
			EXT	1A B3	hh ll	4	7	
			IND,X	1A A3	ff	3	7	
			IND,Y	CD A3	ff	3	7	
CPX (opr)	Compare X to Memory 16-Bit	IX – M:M+1	IMM	8C	jj kk	3	4	— — — — ↕ ↕ ↕ ↕
			DIR	9C	dd	2	5	
			EXT	BC	hh ll	3	6	
			IND,X	AC	ff	2	6	
			IND,Y	CD AC	ff	3	7	
CPY (opr)	Compare Y to Memory 16-Bit	IY – M:M+1	IMM	18 8C	jj kk	4	5	— — — — ↕ ↕ ↕ ↕
			DIR	18 9C	dd	3	6	
			EXT	18 BC	hh ll	4	7	
			IND,X	1A AC	ff	3	7	
			IND,Y	18 AC	ff	3	7	
DAA	Decimal Adjust A	Adjust Sum to BCD	INH	19		1	2	— — — — ↕ ↕ ↕ ↕
DEC (opr)	Decrement Memory Byte	M – 1 ⇩ M	EXT	7A	hh ll	3	6	— — — — ↕ ↕ ↕ —
			IND,X	6A	ff	2	6	
			IND,Y	18 6A	ff	3	7	
DECA	Decrement Accumulator A	A – 1 ⇩ A	A INH	4A		1	2	— — — — ↕ ↕ ↕ —
DECB	Decrement Accumulator B	B – 1 ⇩ B	B INH	5A		1	2	— — — — ↕ ↕ ↕ —
DES	Decrement Stack Pointer	SP – 1 ⇩ SP	INH	34		1	3	— — — — — — — —
DEX	Decrement Index Register X	IX – 1 ⇩ IX	INH	09		1	3	— — — — — ↕ — —
DEY	Decrement Index Register Y	IY – 1 ⇩ IY	INH	18 09		2	4	— — — — — ↕ — —
EORA (opr)	Exclusive OR A with Memory	A ⊕ M ⇩ A	A IMM	88	ii	2	2	— — — — ↕ ↕ 0 —
			A DIR	98	dd	2	3	
			A EXT	B8	hh ll	3	4	
			A IND,X	A8	ff	2	4	
			A IND,Y	18 A8	ff	3	5	
EORB (opr)	Exclusive OR B with Memory	B ⊕ M ⇩ B	B IMM	C8	ii	2	2	— — — — ↕ ↕ 0 —
			B DIR	D8	dd	2	3	
			B EXT	F8	hh ll	3	4	
			B IND,X	E8	ff	2	4	
			B IND,Y	18 E8	ff	3	5	
FDIV	Fractional Divide 16 by 16	D/IX ⇩ IX; r ⇩ D	INH	03		1	41	— — — — — ↕ ↕ ↕
IDIV	Integer Divide 16 by 16	D/IX ⇩ IX; r ⇩ D	INH	02		1	41	— — — — — ↕ 0 ↕
INC (opr)	Increment Memory Byte	M + 1 ⇩ M	EXT	7C	hh ll	3	6	— — — — ↕ ↕ ↕ —
			IND,X	6C	ff	2	6	
			IND,Y	18 6C	ff	3	7	
INCA	Increment Accumulator A	A + 1 ⇩ A	A INH	4C		1	2	— — — — ↕ ↕ ↕ —
INCB	Increment Accumulator B	B + 1 ⇩ B	B INH	5C		1	2	— — — — ↕ ↕ ↕ —
INS	Increment Stack Pointer	SP + 1 ⇩ SP	INH	31		1	3	— — — — — — — —

Table 12-1. Instructions, Addressing Modes, and Execution Times (Sheet 4 of 7)

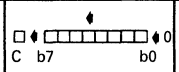
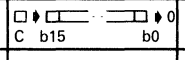
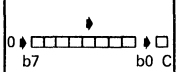
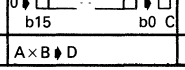
Source Form(s)	Operation	Boolean Expression	Addressing Mode for Operand	Machine Coding (Hexadecimal) Opcode	Operand(s)	Bytes	Cycle	Condition Codes S X H I N Z V C
INX	Increment Index Register X	IX+1 ⇨ IX	INH	08		1	3	— — — — — ↕ — —
INY	Increment Index Register Y	IY+1 ⇨ IY	INH	18 08		2	4	— — — — — ↕ — —
JMP (opr)	Jump	See Special Ops	EXT	7E	hh ll	3	3	— — — — — — — —
			IND,X	6E	ff	2	3	
			IND,Y	18 6E	ff	3	4	
JSR (opr)	Jump to Subroutine	See Special Ops	DIR	9D	dd	2	5	— — — — — — — —
			EXT	BD	hh ll	3	6	
			IND,X	AD	ff	2	6	
			IND,Y	18 AD	ff	3	7	
LDAA (opr)	Load Accumulator A	M ⇨ A	A IMM	86	ii	2	2	— — — — ↕ ↕ 0 —
			A DIR	96	dd	2	3	
			A EXT	B6	hh ll	3	4	
			A IND,X	A6	ff	2	4	
			A IND,Y	18 A6	ff	3	5	
LDAB (opr)	Load Accumulator B	M ⇨ B	B IMM	C6	ii	2	2	— — — — ↕ ↕ 0 —
			B DIR	D6	dd	2	3	
			B EXT	F6	hh ll	3	4	
			B IND,X	E6	ff	2	4	
			B IND,Y	18 E6	ff	3	5	
LDD (opr)	Load Double Accumulator D	M ⇨ A,M+1 ⇨ B	IMM	CC	jj kk	3	3	— — — — ↕ ↕ 0 —
			DIR	DC	dd	2	4	
			EXT	FC	hh ll	3	5	
			IND,X	EC	ff	2	5	
			IND,Y	18 EC	ff	3	6	
LDS (opr)	Load Stack Pointer	M:M+1 ⇨ SP	IMM	8E	jj kk	3	3	— — — — ↕ ↕ 0 —
			DIR	9E	dd	2	4	
			EXT	BE	hh ll	3	5	
			IND,X	AE	ff	2	5	
			IND,Y	18 AE	ff	3	6	
LDX (opr)	Load Index Register X	M:M+1 ⇨ IX	IMM	CE	jj kk	3	3	— — — — ↕ ↕ 0 —
			DIR	DE	dd	2	4	
			EXT	FE	hh ll	3	5	
			IND,X	EE	ff	2	5	
			IND,Y	CD EE	ff	3	6	
LDY (opr)	Load Index Register Y	M:M+1 ⇨ IY	IMM	18 CE	jj kk	4	4	— — — — ↕ ↕ 0 —
			DIR	18 DE	dd	3	5	
			EXT	18 FE	hh ll	4	6	
			IND,X	1A EE	ff	3	6	
			IND,Y	18 EE	ff	3	6	
LSL (opr)	Logical Shift Left	 C b7 b0	EXT	78	hh ll	3	6	— — — — ↕ ↕ ↕ ↕
			IND,X	68	ff	2	6	
			IND,Y	18 68	ff	3	7	
LSLA			A INH	48		1	2	
LSLB			B INH	58		1	2	
LSLD	Logical Shift Left Double	 C b15 b0	INH	05		1	3	— — — — ↕ ↕ ↕ ↕
LSR (opr)	Logical Shift Right	 b7 b0 C	EXT	74	hh ll	3	6	— — — — 0 ↕ ↕ ↕
			IND,X	64	ff	2	6	
			IND,Y	18 64	ff	3	7	
LSRA			A INH	44		1	2	
LSRB			B INH	54		1	2	
LSRD	Logical Shift Right Double	 b15 b0 C	INH	04		1	3	— — — — 0 ↕ ↕ ↕
MUL	Multiply 8 by 8	A×B ⇨ D	INH	3D		1	10	— — — — — — — ↕

Table 12-1. Instructions, Addressing Modes, and Execution Times (Sheet 5 of 7)

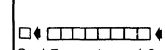
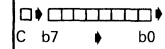
Source Form(s)	Operation	Boolean Expression	Addressing Mode for Operand	Machine Coding (Hexadecimal)				Condition Codes
				Opcode	Operand(s)			S X H I N Z V C
NEG (opr)	Twos Complement Memory Byte	0 − M ⇩ M	EXT IND,X IND,Y	70 60 18 60	hh ll ff ff	3 2 3	6 6 7	— — — — ↕ ↕ ↕ ↕
NEGA	Twos Complement A	0 − A ⇩ A	A INH	40		1	2	— — — — ↕ ↕ ↕ ↕
NEGB	Twos Complement B	0 − B ⇩ B	B INH	50		1	2	— — — — ↕ ↕ ↕ ↕
NOP	No Operation	No Operation	INH	01		1	2	— — — — — — — —
ORAA (opr)	OR Accumulator A (Inclusive)	A + M ⇩ A	A IMM A DIR A EXT A IND,X A IND,Y	8A 9A BA AA 18 AA	ii dd hh ll ff ff	2 2 3 2 3	2 3 4 4 5	— — — — ↕ ↕ 0 —
ORAB (opr)	OR Accumulator B (Inclusive)	B + M ⇩ B	B IMM B DIR B EXT B IND,X B IND,Y	CA DA FA EA 18 EA	ii dd hh ll ff ff	2 2 3 2 3	2 3 4 4 5	— — — — ↕ ↕ 0 —
PSHA	Push A onto Stack	A ⇩ Stk,SP = SP − 1	A INH	36		1	3	— — — — — — — —
PSHB	Push B onto Stack	B ⇩ Stk,SP = SP − 1	B INH	37		1	3	— — — — — — — —
PSHX	Push X onto Stack (Lo First)	IX ⇩ Stk,SP = SP − 2	INH	3C		1	4	— — — — — — — —
PSHY	Push Y onto Stack (Lo First)	IY ⇩ Stk,SP = SP − 2	INH	18 3C		2	5	— — — — — — — —
PULA	Pull A from Stack	SP = SP + 1,A ⇧ Stk	A INH	32		1	4	— — — — — — — —
PULB	Pull B from Stack	SP = SP + 1,B ⇧ Stk	B INH	33		1	4	— — — — — — — —
PULX	Pull X from Stack (Hi First)	SP = SP + 2,IX ⇧ Stk	INH	38		1	5	— — — — — — — —
PULY	Pull Y from Stack (Hi First)	SP = SP + 2,IY ⇧ Stk	INH	18 38		2	6	— — — — — — — —
ROL (opr)	Rotate Left		EXT IND,X IND,Y	79 69 18 69	hh ll ff ff	3 2 3	6 6 7	— — — — ↕ ↕ ↕ ↕
ROLA			A INH	49		1	2	
ROLB			B INH	59		1	2	
ROR (opr)	Rotate Right		EXT IND,X IND,Y	76 66 18 66	hh ll ff ff	3 2 3	6 6 7	— — — — ↕ ↕ ↕ ↕
RORA			A INH	46		1	2	
RORB			B INH	56		1	2	
RTI	Return from Interrupt	See Special Ops	INH	3B		1	12	↕ ↕ ↕ ↕ ↕ ↕ ↕ ↕
RTS	Return from Subroutine	See Special Ops	INH	39		1	5	— — — — — — — —
SBA	Subtract B from A	A − B ⇩ A	INH	10		1	2	— — — — ↕ ↕ ↕ ↕
SBCA (opr)	Subtract with Carry from A	A − M − C ⇩ A	A IMM A DIR A EXT A IND,X A IND,Y	82 92 B2 A2 18 A2	ii dd hh ll ff ff	2 2 3 2 3	2 3 4 4 5	— — — — ↕ ↕ ↕ ↕
SBCB (opr)	Subtract with Carry from B	B − M − C ⇩ B	B IMM B DIR B EXT B IND,X B IND,Y	C2 D2 F2 E2 18 E2	ii dd hh ll ff ff	2 2 3 2 3	2 3 4 4 5	— — — — ↕ ↕ ↕ ↕
SEC	Set Carry	1 ⇩ C	INH	OD		1	2	— — — — — — — 1
SEI	Set Interrupt Mask	1 ⇩ I	INH	OF		1	2	— — — 1 — — — —
SEV	Set Overflow Flag	1 ⇩ V	INH	OB		1	2	— — — — — — 1 —

Table 12-1. Instructions, Addressing Modes, and Execution Times (Sheet 6 of 7)

Source Form(s)	Operation	Boolean Expression	Addressing Mode for Operand	Machine Coding (Hexadecimal)		Bytes	Cycle	Condition Codes
				Opcode	Operand(s)			S X H I N Z V C
STAA (opr)	Store Accumulator A	A ⇨ M	A DIR	97	dd	2	3	— — — — ↕ ↕ 0 —
			A EXT	B7	hh ll	3	4	
			A IND,X	A7	ff	2	4	
			A IND,Y	18 A7	ff	3	5	
STAB (opr)	Store Accumulator B	B ⇨ M	B DIR	D7	dd	2	3	— — — — ↕ ↕ 0 —
			B EXT	F7	hh ll	3	4	
			B IND,X	E7	ff	2	4	
			B IND,Y	18 E7	ff	3	5	
STD (opr)	Store Accumulator D	A ⇨ M,B ⇨ M+1	DIR	DD	dd	2	4	— — — — ↕ ↕ 0 —
			EXT	FD	hh ll	3	5	
			IND,X	ED	ff	2	5	
			IND,Y	18 ED	ff	3	6	
STOP	Stop Internal Clocks		INH	CF		1	2	— — — — — — — —
STS (opr)	Store Stack Pointer	SP ⇨ M:M+1	DIR	9F	dd	2	4	— — — — ↕ ↕ 0 —
			EXT	BF	hh ll	3	5	
			IND,X	AF	ff	2	5	
			IND,Y	18 AF	ff	3	6	
STX (opr)	Store Index Register X	IX ⇨ M:M+1	DIR	DF	dd	2	4	— — — — ↕ ↕ 0 —
			EXT	FF	hh ll	3	5	
			IND,X	EF	ff	2	5	
			IND,Y	CD EF	ff	3	6	
STY (opr)	Store Index Register Y	IY ⇨ M:M+1	DIR	18 DF	dd	3	5	— — — — ↕ ↕ 0 —
			EXT	18 FF	hh ll	4	6	
			IND,X	1A EF	ff	3	6	
			IND,Y	18 EF	ff	3	6	
SUBA (opr)	Subtract Memory from A	A − M ⇨ A	A IMM	80	ii	2	2	— — — — ↕ ↕ ↕ ↕
			A DIR	90	dd	2	3	
			A EXT	B0	hh ll	3	4	
			A IND,X	A0	ff	2	4	
			A IND,Y	18 A0	ff	3	5	
SUBB (opr)	Subtract Memory from B	B − M ⇨ B	B IMM	C0	ii	2	2	— — — — ↕ ↕ ↕ ↕
			B DIR	D0	dd	2	3	
			B EXT	F0	hh ll	3	4	
			B IND,X	E0	ff	2	4	
			B IND,Y	18 E0	ff	3	5	
SUBD (opr)	Subtract Memory from D	D − M:M+1 ⇨ D	IMM	83	jj kk	3	4	— — — — ↕ ↕ ↕ ↕
			DIR	93	dd	2	5	
			EXT	B3	hh ll	3	6	
			IND,X	A3	ff	2	6	
			IND,Y	18 A3	ff	3	7	
SWI	Software Interrupt	See Special Ops	INH	3F		1	14	— — — 1 — — — —
TAB	Transfer A to B	A ⇨ B	INH	16		1	2	— — — — ↕ ↕ 0 —
TAP	Transfer A to CC Register	A ⇨ CCR	INH	06		1	2	↕ ↕ ↕ ↕ ↕ ↕ ↕ ↕
TBA	Transfer B to A	B ⇨ A	INH	17		1	2	— — — — ↕ ↕ 0 —
TEST	TEST (Only in Test Modes)	Address Bus Counts	INH	00		1	*	— — — — — — — —
TPA	Transfer CC Register to A	CCR ⇨ A	INH	07		1	2	— — — — — — — —
TST (opr)	Test for Zero or Minus	M − 0	EXT	7D	hh ll	3	6	— — — — ↕ ↕ 0 0
			IND,X	6D	ff	2	6	
			IND,Y	18 6D	ff	3	7	
TSTA		A − 0	A INH	4D		1	2	— — — — ↕ ↕ 0 0
TSTB		B − 0	B INH	5D		1	2	— — — — ↕ ↕ 0 0
TSX	Transfer Stack Pointer to X	SP+1 ⇨ IX	INH	30		1	3	— — — — — — — —
TSY	Transfer Stack Pointer to Y	SP+1 ⇨ IY	INH	18 30		2	4	— — — — — — — —

Table 12-1. Instructions, Addressing Modes, and Execution Times (Sheet 7 of 7)

Source Form(s)	Operation	Boolean Expression	Addressing Mode for Operand	Machine Coding (Hexadecimal)		Bytes	Cycle	Condition Codes
				Opcode	Operand(s)			S X H I N Z V C
TXS	Transfer X to Stack Pointer	IX − 1 ⇨ SP	INH	35		1	3	— — — — — — — —
TYS	Transfer Y to Stack Pointer	IY − 1 ⇨ SP	INH	18 35		2	4	— — — — — — — —
WAI	Wait for Interrupt	Stack Regs & WAIT	INH	3E		1	**	— — — — — — — —
XGDX	Exchange D with X	IX ⇨ D, D ⇨ IX	INH	8F		1	3	— — — — — — — —
XGDY	Exchange D with Y	IY ⇨ D, D ⇨ IY	INH	18 8F		2	4	— — — — — — — —

Cycle

*Infinity or until reset occurs

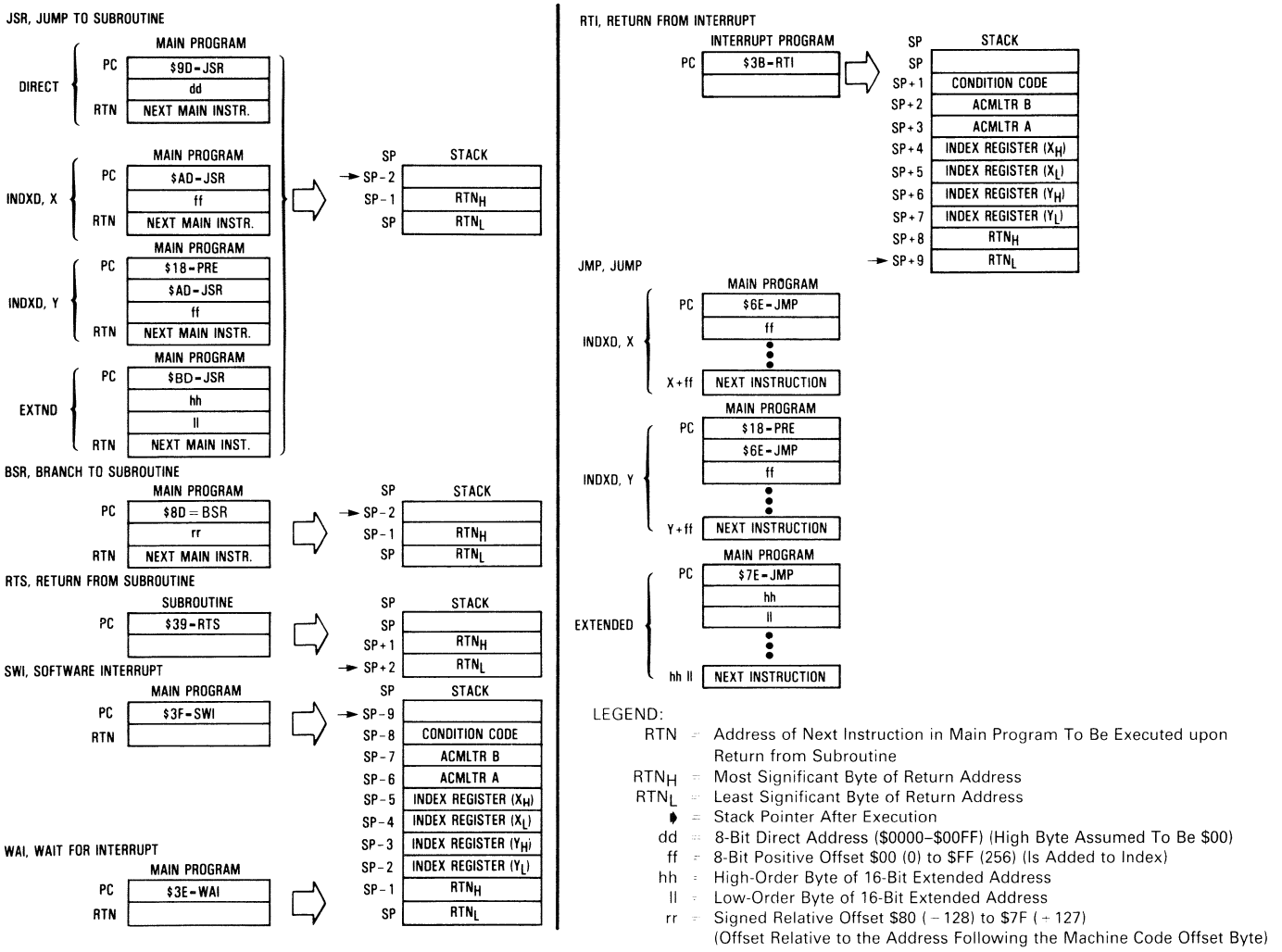
**12 Cycles are used beginning with the opcode fetch. A wait state is entered which remains in effect for an integer number of MPU E-clock cycles (n) until an interrupt is recognized. Finally, two additional cycles are used to fetch the appropriate interrupt vector (14+n total).

Operands

dd = 8-Bit Direct Address ($0000–$00FF) (High Byte Assumed to be $00)
ff = 8-Bit Positive Offset $00 (0) to $FF (255) (Is Added to Index)
hh = High-Order Byte of 16-Bit Extended Address
ii = One Byte of Immediate Data
jj = High-Order Byte of 16-Bit Immediate Data
kk = Low-Order Byte of 16-Bit Immediate Data
ll = Low-Order Byte of 16-Bit Extended Address
mm = 8-Bit Mask (Set Bits to be Affected)
rr = Signed Relative Offset $80 (−128) to $7F (+127)
(Offset Relative to the Address Following the Machine Code Offset Byte)

Condition Codes

— Bit not changed
0 Bit always cleared
1 Bit always set
⇕ Bit cleared or set, depending on operation
⇩ Bit can be cleared, cannot become set



LEGEND:

RTN	=	Address of Next Instruction in Main Program To Be Executed upon Return from Subroutine
RTN_H	=	Most Significant Byte of Return Address
RTN_L	=	Least Significant Byte of Return Address
↓	=	Stack Pointer After Execution
dd	=	8-Bit Direct Address ($0000–$00FF) (High Byte Assumed To Be $00)
ff	=	8-Bit Positive Offset $00 (0) to $FF (256) (Is Added to Index)
hh	=	High-Order Byte of 16-Bit Extended Address
ll	=	Low-Order Byte of 16-Bit Extended Address
rr	=	Signed Relative Offset $80 (−128) to $7F (+127) (Offset Relative to the Address Following the Machine Code Offset Byte)

Figure 12-1. Special Operations

12.3 ADDRESSING MODES

This MCU uses six different addressing modes that provide the programmer with an opportunity to optimize code for all situations. Some instructions require an additional byte (prebyte) before the opcode to accommodate a multipage opcode map. Refer to Figure 12-2 for a map of the opcodes.

The term effective address (EA) is used in describing the various addressing modes. The effective address is the address from which the argument for an instruction is fetched or stored.

12.3.1 Immediate

In the immediate addressing mode, the operand is contained in the byte immediately following the opcode. These are two- or three-byte instructions (four-byte if a prebyte is required).

12.3.2 Direct

In the direct addressing mode, the low-order byte of the operand address is contained in a single byte following the opcode, and the high-order byte of an address is assumed to be $00. Direct addressing allows the user to directly address $0000-00FF using two-byte instructions. Execution time is reduced by eliminating the additional memory access. In most applications, this 256-byte area is reserved for frequently referenced data. In this MCU, software can configure the memory map so that internal RAM, and/or internal registers or external memory space can occupy these addresses.

12.3.3 Extended

In the extended addressing mode, the effective address of the argument is contained in the two bytes following the opcode byte. These are three-byte instructions (four-byte if a prebyte is required). One or two bytes are needed for the opcode and two bytes for the effective address.

12.3.4 Indexed

In the indexed addressing mode, one of the index registers, either X or Y, is used in calculating the effective address. In this case, the effective address is variable and depends on two factors: the current contents of the index register (X or Y) being used and the 8-bit unsigned offset contained in the

instruction. This addressing mode allows the programmer to reference any memory location in the 64K-byte address space. These two- or three-byte instructions consist of the opcode, the 8-bit offset, and perhaps a prebyte.

12.3.5 Relative

The relative addressing mode is used only in branch instructions. In relative addressing, the contents of the 8-bit signed byte following the opcode (the offset) is added to the PC. This addition is performed if, and only if, the branch conditions are true. Otherwise, control proceeds to the next instruction. Relative addressing instructions are usually two-byte instructions.

12.3.6 Inherent

In the inherent addressing mode, all the information necessary to execute the instruction is contained in the opcode. Operations specifying only the index register or accumulator and the control instruction with no other arguments are included in this addressing mode. These are one- or two-byte instructions.

12.3.7 Prebyte

To expand the number of instructions used in this MCU, a prebyte has been added to certain instructions. The instructions affected are usually associated with index register Y. Accessing opcodes from pages 2, 3, or 4 (see Figure 12-2) would require a prebyte.

		DIR			OPCODE MAP PAGE 1					ACCA				ACCB				
		INH	INH	REL	INH	ACCA	ACCB	IND,X	EXT	IMM	DIR	IND,X	EXT	IMM	DIR	IND,X	EXT	
MSB		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111	
LSB		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
0000	0	TEST*	SBA	BRA	TSX	NEG				SUB								0
0001	1	NOP	CBA	BRN	INS					CMP								1
0010	2	IDIV	BRSET	BHI	PULA					SBC								2
0011	3	FDIV	BRCLR	BLS	PULB	COM				SUBD				ADDD				3
0100	4	LSRD	BSET	BCC	DES	LSR				AND								4
0101	5	ASLD	BCLR	BCS	TXS					BIT								5
0110	6	TAP	TAB	BNE	PSHA	ROR				LDA								6
0111	7	TPA	TBA	BEQ	PSHB	ASR					STA				STA			7
1000	8	INX	PG 2	BVC	PULX	ASL				EOR								8
1001	9	DEX	DAA	BVS	RTS	ROL				ADC								9
1010	A	CLV	PG 3	BPL	ABX	DEC				ORA								A
1011	B	SEV	ABA	BMI	RTI					ADD								B
1100	C	CLC	BSET	BGE	PSHX	INC				CPX				LDD				C
1101	D	SEC	BCLR	BLT	MUL	TST				BSR	JSR			PG 4	STD			D
1110	E	CLI	BRSET	BGT	WAI			JMP		LDS				LDX				E
1111	F	SEI	BRCLR	BLE	SWI	CLR				XGDX	STS			STOP	STX			F
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	

IND,X

		OPCODE MAP PAGE 2 (18xx)								ACCA				ACCB				
		INH			INH			IND,Y		IMM	DIR	IND,Y	EXT	IMM	DIR	IND,Y	EXT	
MSB		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111	
LSB		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
0000	0				TSY			NEG				SUB				SUB		0
0001	1											CMP				CMP		1
0010	2											SBC				SBC		2
0011	3							COM				SUBD				ADDD		3
0100	4							LSR				AND				AND		4
0101	5				TYS							BIT				BIT		5
0110	6							ROR				LDA				LDA		6
0111	7							ASR				STA				STA		7
1000	8	INY			PULY			ASL				EOR				EOR		8
1001	9	DEY						ROL				ADC				ADC		9
1010	A				ABY			DEC				ORA				ORA		A
1011	B											ADD				ADD		B
1100	C		BSET		PSHY			INC		CPY						LDD		C
1101	D		BCLR					TST				JSR				STD		D
1110	E		BRSET					JMP				LDS		LDY				E
1111	F		BRCLR					CLR		XGDY		STS			STY			F
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	

IND,Y

Figure 12-2. Opcode Map (Sheet 1 of 2)

OPCODE MAP PAGE 3 (1Axx)									ACCA				ACCB			
									IMM	DIR	IND,X	EXT			IND,X	
MSB		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
LSB		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0																0
0001	1																1
0010	2																2
0011	3									CPD							3
0100	4																4
0101	5																5
0110	6																6
0111	7																7
1000	8																8
1001	9																9
1010	A																A
1011	B																B
1100	C											CPY					C
1101	D																D
1110	E															LDY	E
1111	F															STY	F
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

OPCODE MAP PAGE 4 (CDxx)									ACCA				ACCB			
											IND,Y				IND,Y	
MSB		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
LSB		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0																0
0001	1																1
0010	2																2
0011	3											CPD					3
0100	4																4
0101	5																5
0110	6																6
0111	7																7
1000	8																8
1001	9																9
1010	A																A
1011	B																B
1100	C											CPX					C
1101	D																D
1110	E															LDX	E
1111	F															STX	F
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

Figure 12-2. Opcode Map (Sheet 2 of 2)

SECTION 13
ELECTRICAL SPECIFICATIONS

This section contains the electrical specifications and associated timing information for the MC68HC11F1 MCU.

13.1 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	T_A	T_L to T_H −40 to 85	°C
Storage Temperature Range	T_{stg}	−55 to 150	°C
Current Drain per Pin* Excluding V_{DD}, V_{SS}, V_{RH}, and V_{RL}	I_D	25	mA

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

This device contains protective circuitry against damage due to high static voltages or electrical fields; however, it is advised that normal precautions be taken to avoid application of any voltages higher than maximum-rated voltages to this high-impedance circuit. Reliability of operation is enhanced if unused inputs are tied to an appropriate logic voltage level (e.g., either GND or V_{DD}).

13.2 THERMAL CHARACTERISTICS

Characteristic	Symbol	Value	Unit
Thermal Resistance Plastic 68-Pin Quad Pack (PLCC)	θ_{JA}	50	°C/W

13.3 POWER CONSIDERATIONS

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

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

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 $P_{I/O} < P_{INT}$ and can be neglected.

The following is an approximate relationship between P_D and T_J (if $P_{I/O}$ is neglected):

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

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

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

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.

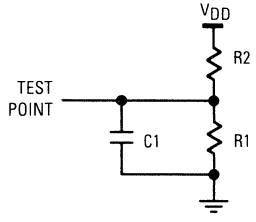
13.4 DC ELECTRICAL CHARACTERISTICS

(V_{DD} = 5.0 Vdc + 10%; V_{SS} = 0 Vdc, $T_A = T_L$ to T_H, unless otherwise noted)

Characteristics		Symbol	Min	Max	Unit
Output Voltage	All Outputs	V_{OL}	—	0.1	V
$I_{Load} = \pm 10.0\ \mu A$ (see Note 1)	All Outputs Except $\overline{RESET}$ and MODA	V_{OH}	$V_{DD} - 0.1$	—	
Output High Voltage	All Outputs Except $\overline{RESET}$,	V_{OH}	$V_{DD} - 0.8$	—	V
$I_{Load} = -0.8$ mA, $V_{DD} = 4.5$ V (see Note 1)	XTAL, and MODA				
Output Low Voltage	All Outputs Except XTAL	V_{OL}	—	0.4	V
$I_{Load} = 1.6$ mA					
Input High Voltage	All Inputs Except $\overline{RESET}$	V_{IH}	$0.7 \times V_{DD}$	$V_{DD} + 0.3$	V
	$\overline{RESET}$		$0.8 \times V_{DD}$	$V_{DD} + 0.3$	
Input Low Voltage	All Inputs	V_{IL}	$V_{SS} - 0.3$	$0.2 \times V_{DD}$	V
I/O Ports, Three-State Leakage	PA7–PA0, PC7–PC0, PD5–PD0, PG7–PG0	I_{OZ}	—	± 10	μA
$V_{in} = V_{IH}$ or V_{IL}	MODA/$\overline{LIR}$, $\overline{RESET}$				
Input Current (see Note 2)		I_{in}			μA
$V_{in} = V_{DD}$ or V_{SS}	$\overline{IRQ}$, $\overline{XIRQ}$		—	± 1	
$V_{in} = V_{DD}$ or V_{SS}	MODB/V_{stby}		—	± 10	
RAM Standby Voltage	Powerdown	V_{SB}	4.0	V_{DD}	V
RAM Standby Current	Powerdown	I_{SB}	—	20	μA
Total Supply Current (see Note 3)					
RUN:		I_{DD}			
Single-Chip Mode			—	15	mA
Expanded-Nonmultiplexed Mode			—	27	mA
WAIT: (All Peripheral Functions Shut Down)		W_{IDD}			
Single-Chip Mode			—	6	mA
Expanded-Nonmultiplexed Mode			—	10	mA
STOP: (No Clocks)		S_{IDD}			
Single-Chip Mode			—	100	μA
Input Capacitance	PE7–PE0, $\overline{IRQ}$, $\overline{XIRQ}$, EXTAL	C_{in}	—	8	pF
	PA7–PA0, PC7–PC0, PD5–PD0, PG7–PG0, MODA/$\overline{LIR}$, $\overline{RESET}$		—	12	
Power Dissipation	Single-Chip Mode	P_D	—	85	mW
	Expanded-Nonmultiplexed Mode		—	150	

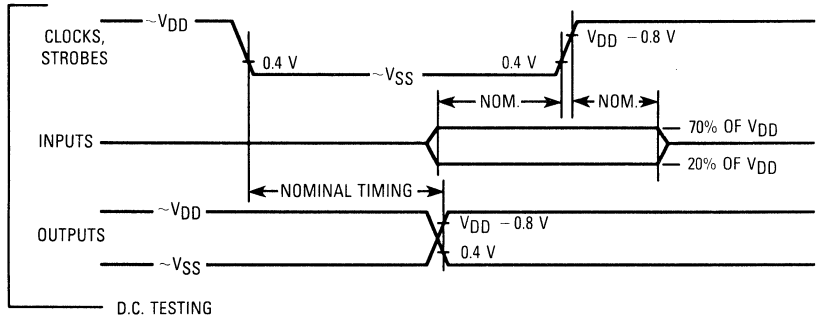
NOTES:

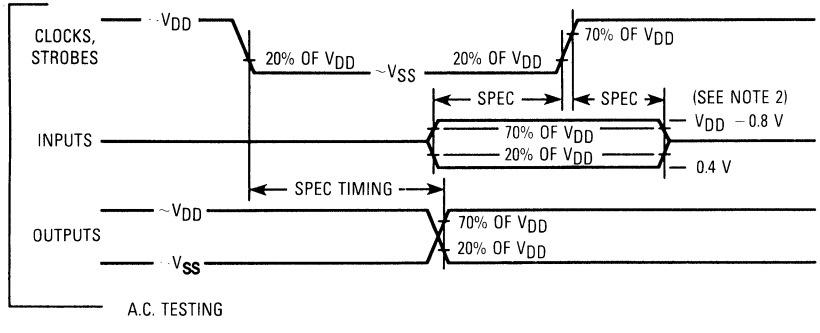
1. V_{OH} specification for $\overline{RESET}$ and MODA is not applicable because they are open-drain pins. V_{OH} specification not applicable to ports C, D, and G in wired-OR mode.
2. See A/D specifications for leakage current for port E.
3. All ports configured as inputs,
   $V_{IL} \leq 0.2$ V,
   $V_{IH} \geq V_{DD} - 0.2$ V,
   No dc loads,
   EXTAL is driven with a square wave, and
   $t_{cyc} = 476.5$ ns.



Equivalent Test Load[1]

Pins	R1	R2	C1
4XOUT	3.26 kΩ	2.38 kΩ	30 pF
PA7-PA3, PB7-PB0, PC7-PC0, PD5, PD0 PF7-PF0 PG7-PG0 E, R/$\overline{W}$	3.26 kΩ	2.38 kΩ	90 pF
PD4-PD1	3.26 kΩ	2.38 kΩ	200 pF





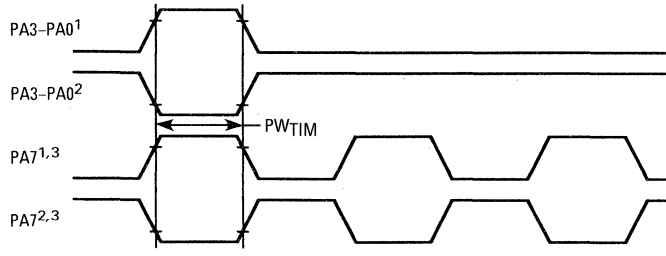
NOTES:
1. Full test loads are applied during all ac electrical timing measurements.
2. During ac timing measurements, inputs are driven to 0.4 V and V_{DD} – 0.8 V while timing measurements are taken at the 20% and 70% of V_{DD} points.

Figure 13-1. Test Methods

13.5 CONTROL TIMING (V_{DD} = 5.0 Vdc ± 10%, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H)

Characteristic	Symbol	1.0 MHz		2.0 MHz		2.1 MHz		Unit
		Min	Max	Min	Max	Min	Max	
Frequency of Operation	f_o	dc	1.0	dc	2.0	dc	2.1	MHz
E-Clock Period	t_{cyc}	1000	—	500	—	476	—	ns
Crystal Frequency	f_{XTAL}	—	4.0	—	8.0	—	8.4	MHz
External Oscillator Frequency	4 f_o	dc	4.0	dc	8.0	dc	8.4	MHz
Processor Control Setup Time ($t_{PCS} = 1.4\ t_{cyc} - 50$ ns) (See Figures 13-3, 13-5, and 13-6)	t_{PCS}	—	200	—	75	—	69	ns
Reset Input Pulse Width (To Guarantee External Reset Vector) (see Note 1 and Figure 13-3) (Minimum Input Time; May Be Pre-empted by Internal Reset)	PW_{RSTL}	8 1	— —	8 1	— —	8 1	— —	t_{cyc}
Mode Programming Setup Time (See Figure 13-3)	t_{MPS}	2	—	2	—	2	—	t_{cyc}
Mode Programming Hold Time (See Figure 13-3)	t_{MPH}	0	—	0	—	0	—	ns
Interrupt Pulse Width, $\overline{IRQ}$ Edge Sensitive Mode ($PW_{IRQ} = t_{cyc} + 20$ ns) (See Figure 13-4 and 13-6)	PW_{IRQ}	1020	—	520	—	496	—	ns
Wait Recovery Startup Time (See Figure 13-5)	t_{WRS}	—	4	—	4	—	4	t_{cyc}
Timer Pulse Width Input Capture, Pulse Accumulator Input ($PW_{TIM} = t_{cyc} + 20$ ns) (See Figure 13-2)	PW_{TIM}	1020	—	520	—	496	—	ns

NOTES:
1. $\overline{RESET}$ will be 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.
2. All timing is shown with respect to 20% V_{DD} and 70% V_{DD} unless otherwise noted.



NOTES:
1. Rising edge sensitive input.
2. Falling edge sensitive input.
3. Maximum pulse accumulator clocking rate is E frequency divided by 2.

Figure 13-2. Timer Inputs Timing Diagram

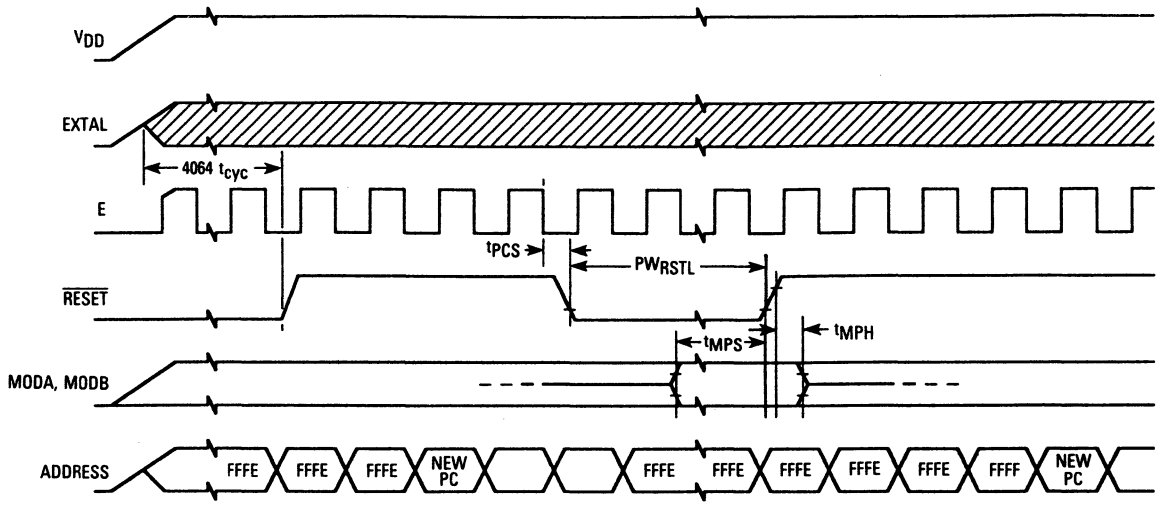
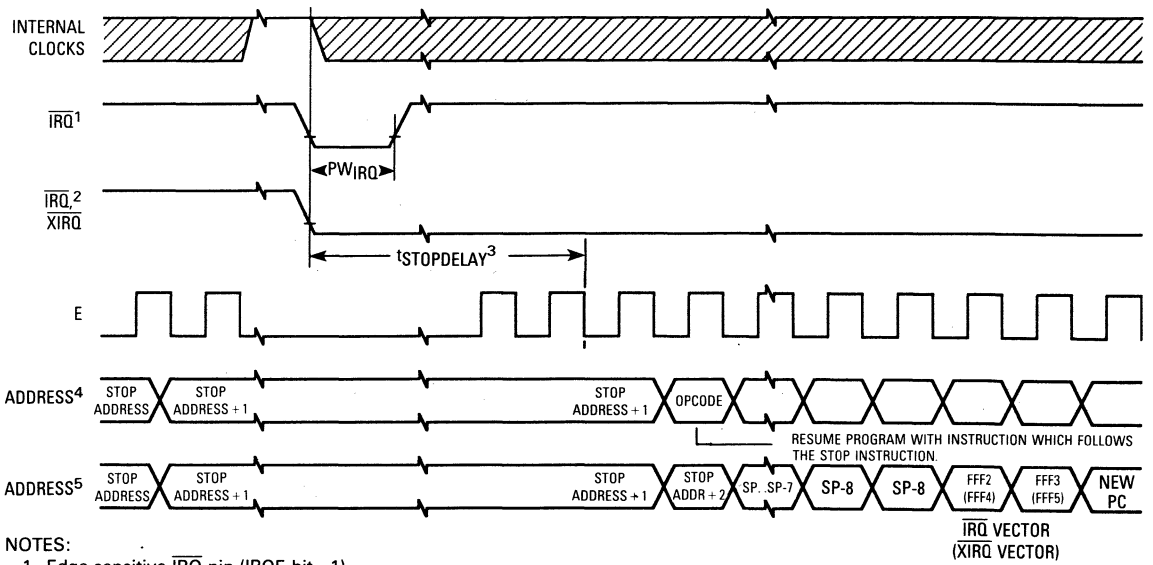
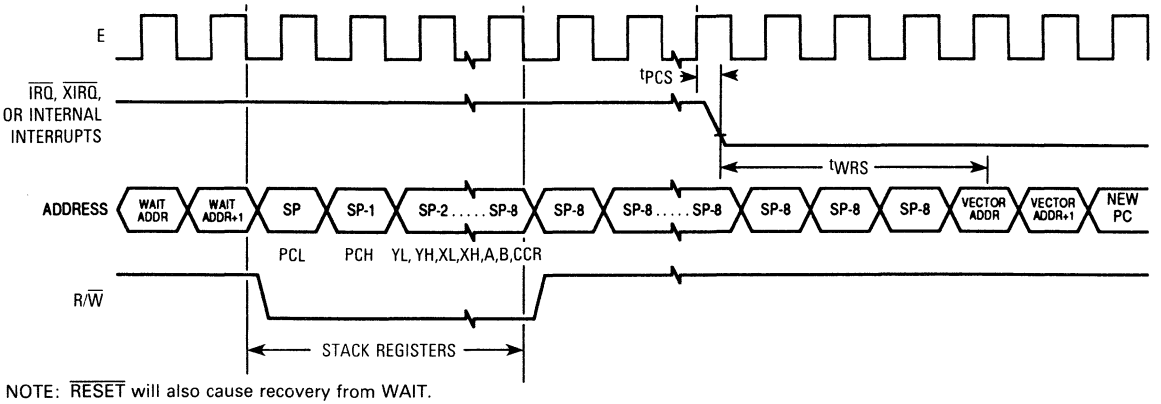


Figure 13-3. POR External Reset Timing Diagram



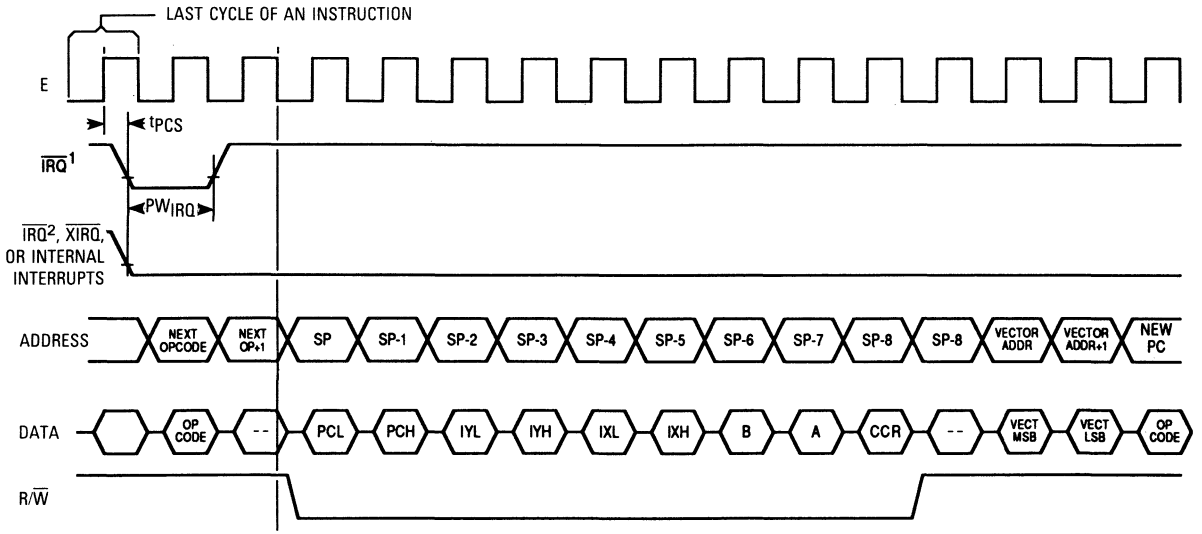
NOTES:
1. Edge sensitive $\overline{IRQ}$ pin (IRQE bit = 1).
2. Level sensitive $\overline{IRQ}$ pin (IRQE bit = 0).
3. $t_{STOPDELAY}$ = 4064 t_{cyc} if DLY bit = 1 or 4 t_{cyc} if DLY = 0.
4. $\overline{XIQ}$ with X bit in CCR = 1.
5. $\overline{IRQ}$ or ($\overline{XIRQ}$ with X bit in CCR = 0.

Figure 13-4. STOP Recovery Timing Diagram



NOTE: $\overline{\text{RESET}}$ will also cause recovery from WAIT.

Figure 13-5. WAIT Recovery from Interrupt Timing Diagram



NOTES:
1. Edge-sensitive $\overline{IRQ}$ pin (IRQE bit = 1).
2. Level-sensitive $\overline{IRQ}$ pin (IRQE bit = 0).

Figure 13-6. Interrupt Timing Diagram

13.6 PERIPHERAL PORT TIMING (V_{DD} = 5.0 Vdc ± 10%, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H)

Characteristic	Symbol	1.0 MHz		2.0 MHz		2.1 MHz		Unit
		Min	Max	Min	Max	Min	Max	
Frequency of Operation (E Clock Frequency)	f_O	1.0	1.0	2.0	2.0	2.1	2.1	MHz
E-Clock Period	t_{cyc}	1000	—	500	—	476	—	ns
Peripheral Data Setup Time (MCU Read of Ports A, C, D, E, and G)	t_{PDSU}	100	—	100	—	100	—	ns
Peripheral Data Hold Time MCU Read of Ports A, C, D, E, and G)	t_{PDH}	50	—	50	—	50	—	ns
Delay Time, Peripheral Data Write (MCU Writes to Port A) (MCU Writes to Ports B, C, D, F, and G) $t_{PWD} = 1/4\ t_{cyc} + 100$ ns	t_{PWD}	— —	200 350	— —	200 225	— —	200 219	ns

NOTES:
1. Port C, D, and G timing is valid for active drive (DWOM bit not set in SPCR register, and CWOM and GWOM bits not set in OPT2 register).
2. All timing is shown with respect to 20% V_{DD} and 70% V_{DD} unless otherwise noted.

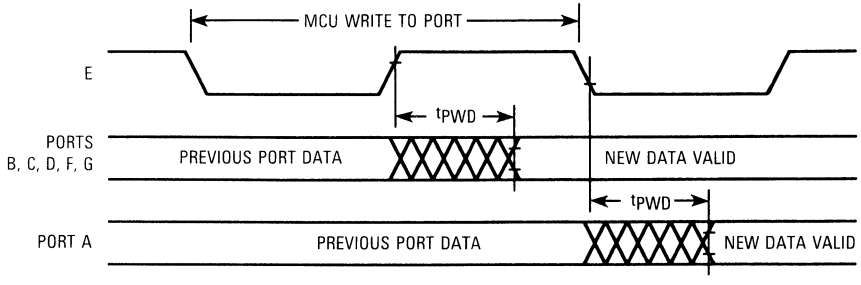


Figure 13-7. Port Write Timing Diagram

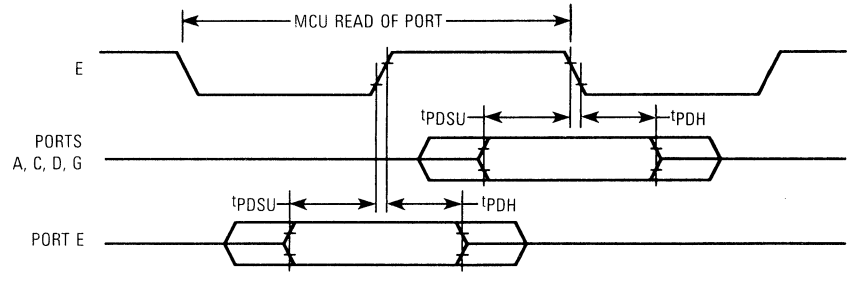


Figure 13-8. Port Read Timing Diagram

13.7 A/D CONVERTER CHARACTERISTICS

(V_{DD} = 5.0 Vdc ± 10%, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H, 750 kHz ≤ E ≤ 2.1 MHz)

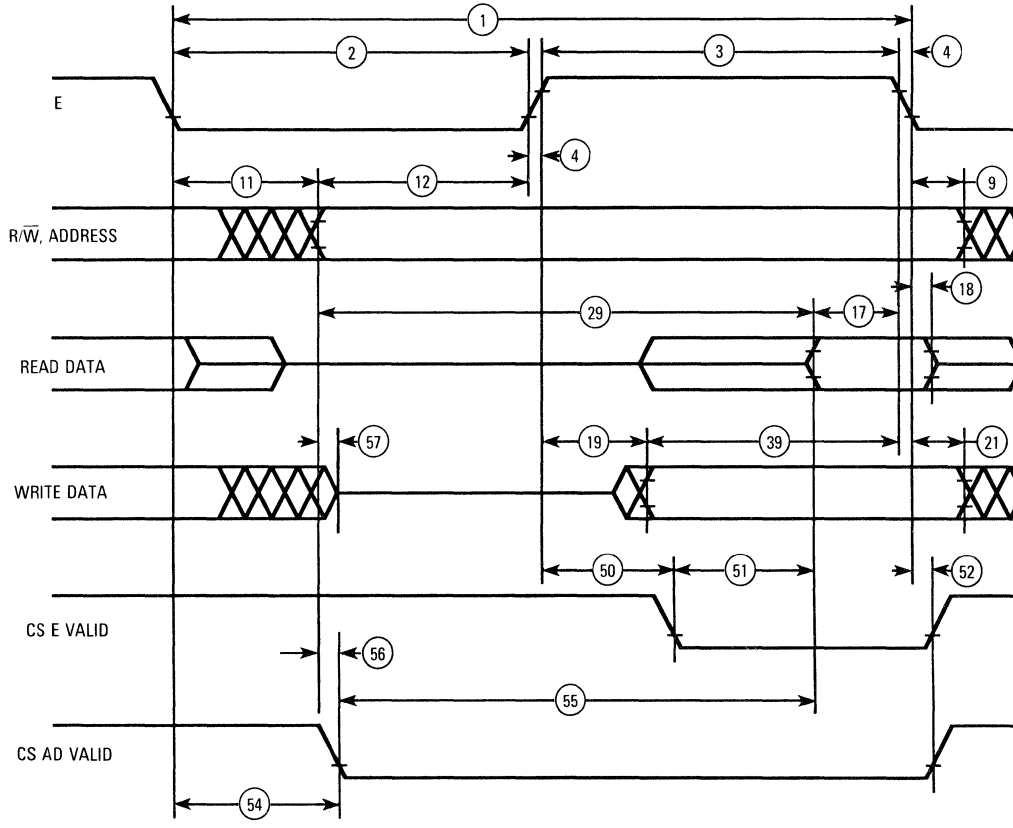
Characteristic	Parameter	Min	Absolute	Max	Unit
Resolution	Number of Bits Resolved by the A/D	—	8	—	Bits
Nonlinearity	Maximum Deviation from the Ideal A/D Transfer Characteristics	—	—	±1.2	LSB
Zero Error	Difference Between the Output of an Ideal and an Actual A/D for Zero Input Voltage	—	—	±1.2	LSB
Full-Scale Error	Difference Between the Output of an Ideal and an Actual A/D for Full-Scale Input Voltage	—	—	±1.2	LSB
Total Unadjusted Error	Maximum Sum of Nonlinearity, Zero Error, and Full-Scale Error	—	—	±1.2	LSB
Quantization Error	Uncertainty Due to Converter Resolution	—	—	±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 a. E Clock b. Internal RC Oscillator	 — —	 — —	 ±1 ±2	LSB
Conversion Range	Analog Input Voltage Range	V_{RL}	—	V_{RH}	V
V_{RH}	Maximum Analog Reference Voltage (see Note 2)	V_{RL}	—	$V_{DD}+0.1$	V
V_{RL}	Minimum Analog Reference Voltage (see Note 2)	$V_{SS}-0.1$	—	R_{RH}	V
ΔV_R	Minimum Difference between V_{RH} and V_{RL} (see Note 2)	3	—	—	V
Conversion Time	Total Time to Perform a Single Analog-to-Digital Conversion: a. E Clock b. Internal RC Oscillator	 — —	 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	Hex
Sample Acquisition Time	Analog Input Acquisition Sampling Time: a. E Clock b. Internal RC Oscillator	 — —	 12 —	 — 12	 t_{cyc} μs
Sample Hold Capacitance	Input Capacitance during Sample PE7–PE0	—	20 (Typ)	—	pF
Input Leakage	Input Leakage on A/D Pins PE7–PE0 V_{RL}, V_{RH}	— —	— —	400 1.0	nA μA

NOTES:
1. Source impedances greater than 10 KΩ will adversely affect accuracy, due mainly to input leakage.
2. Performance verified down to 2.5 V ΔV_R, but accuracy is tested and guaranteed at ΔV_R = 5 V ± 10%.

13.8 EXPANSION BUS TIMING (V_{DD} = 5.0 Vdc ± 10%, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H, see Figure 13-9)

Num	Characteristic	Symbol	2.1 MHz		Unit
			Min	Max	
1	Cycle Time (see Note 1)	t_{cyc}	476	—	ns
2	Pulse Width, E Low (1/2 t_{cyc} − 23 ns)	PW_{EL}	215	—	ns
3	Pulse Width, E High (1/2 t_{cyc} − 28 ns) (see Note 1)	PW_{EH}	210	—	ns
4	E Rise and Fall Time	t_r, t_f	—	20	ns
9	Address Hold Time (1/8 t_{cyc})	t_{AH}	60	—	ns
11	Address Delay Time (1/8 t_{cyc} + 60 ns)	t_{AD}	—	120	ns
12	Address Valid Time ($PW_{EL} - t_{AD}$)	t_{AV}	95	—	ns
17	Read Data Setup Time (see Note 1)	t_{DSR}	30	—	ns
18	Read Data Hold Time	t_{DHR}	0	—	ns
19	Write Data Delay Time	t_{DDW}	—	40	ns
21	Write Data Hold Time	t_{DHW}	30	—	ns
29	MPU Address Access Time ($t_{cyc} - t_f - t_{DSR} - t_{AD}$) (see Note 1)	t_{ACCA}	306	—	ns
39	Write Data Setup Time ($PW_{EH} - t_{DDW}$) (see Note 1)	t_{DSW}	170	—	ns
50	E Valid Chip Select Delay Time	t_{ECSD}	—	50	ns
51	E Valid Chip Select Access Time ($PW_{EH} - t_{ECSD} - t_{DSR}$) (see Note 1)	t_{ECSA}	130	—	ns
52	Chip Select Hold Time	t_{CH}	0	20	ns
54	Address Valid Chip Select Delay Time (1/4 t_{cyc} + 40 ns)	t_{ACSD}	—	160	ns
55	Address Valid Chip Select Access Time ($t_{cyc} - t_f - t_{DSR} - t_{ACSD}$) (see Note 1)	t_{ACSA}	266	—	ns
56	Address Valid to Chip Select Time	t_{AVCS}	0	—	ns
57	Address Valid to Data Three-State Time	t_{AVDZ}	—	10	ns

NOTES:
1. Indicates a parameter affected by clock stretching. Add $n \times t_{cyc}$ to parameter value, where n = 1, 2, or 3 (depends on values written to CSSTRH register).
2. All timing is shown with respect to 20% V_{DD} and 70% V_{DD}.



NOTE: Measurement points shown are 20% and 70% V_{DD}.

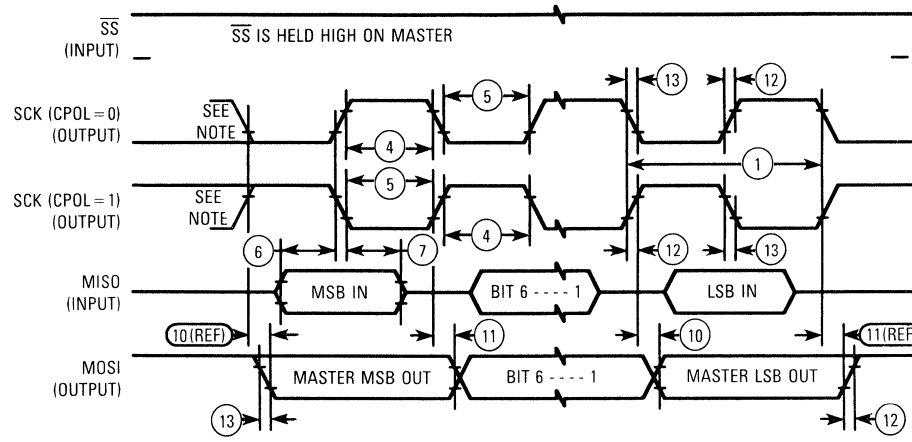
Figure 13-9. Expansion Bus Timing Diagram

13.9 SERIAL PERIPHERAL INTERFACE TIMING

(V_{DD} = 5.0 Vdc ± 10%, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H, see Figure 13-10)

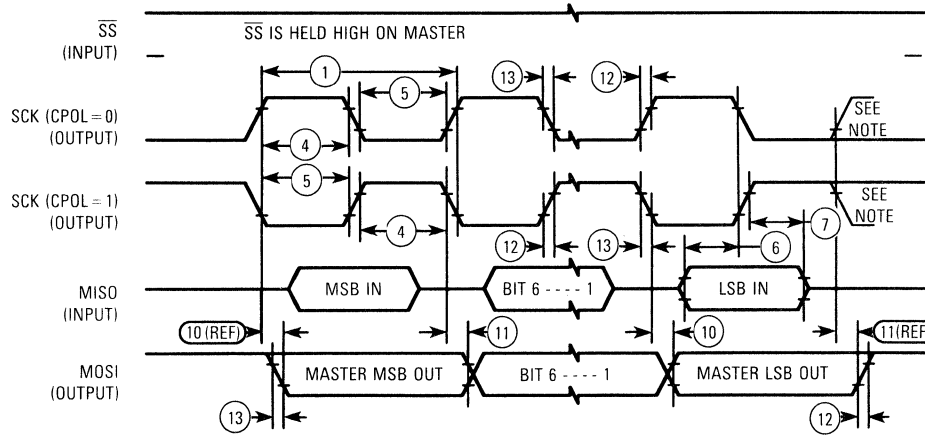
Num	Characteristic	Symbol	Min	Max	Unit
	Operating Frequency Master Slave	 $f_{op(m)}$ $f_{op(s)}$	 dc dc	 0.5 2.1	f_{op} MHz
1	Cycle Time Master Slave	 $t_{cyc(m)}$ $t_{cyc(s)}$	 2.0 480	 — —	t_{cyc} ns
2	Enable Lead Time Master Slave	 $t_{lead(m)}$ $t_{lead(s)}$	 * 240	 — —	 ns ns
3	Enable Lag Time Master Slave	 $t_{lag(m)}$ $t_{lag(s)}$	 * 240	 — —	 ns ns
4	Clock (SCK) High Time Master Slave	 $t_{w(SCKH)m}$ $t_{w(SCKH)s}$	 340 190	 — —	 ns ns
5	Clock (SCK) Low Time Master Slave	 $t_{w(SCKL)m}$ $t_{w(SCKL)s}$	 340 190	 — —	 ns ns
6	Data Setup Time (Inputs) Master Slave	 $t_{su(m)}$ $t_{su(s)}$	 100 100	 — —	 ns ns
7	Data Hold Time (Inputs) Master Slave	 $t_{h(m)}$ $t_{h(s)}$	 100 100	 — —	 ns ns
8	Access Time (Time to Data Active from High-Impedance State) Slave	 t_a	 0	 120	 ns
9	Disable Time (Hold Time to High-Impedance State) Slave	 t_{dis}	 —	 240	 ns
10	Data Valid (After Enable Edge)**	$t_{v(s)}$	—	240	ns
11	Data Hold Time (Outputs) (After Enable Edge)	t_{ho}	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	 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

*Signal product depends on software.
**Assumes 200 pF load on all SPI pins.
NOTE:
1. All timing is shown with respect 20% V_{DD} and 70% V_{DD} unless otherwise noted.



NOTE: This first clock edge is generated internally but is not seen at the SCK pin.

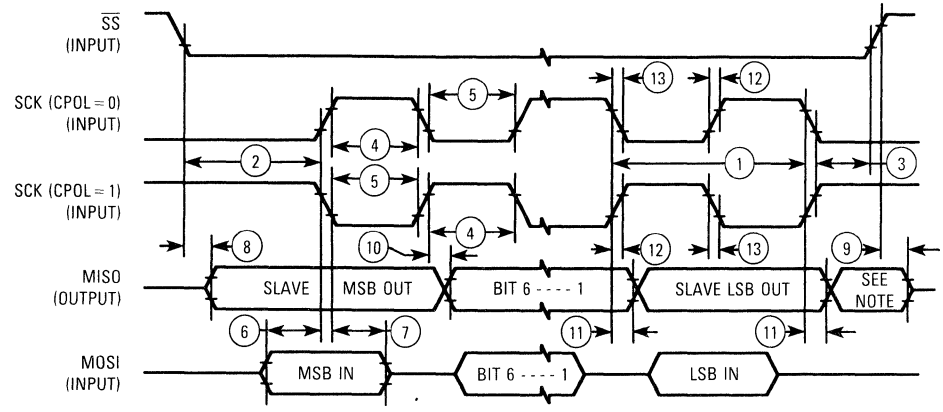
(a) SPI Master Timing (CPHA = 0)



NOTE: The last clock edge is generated internally but is not seen at the SCK pin.

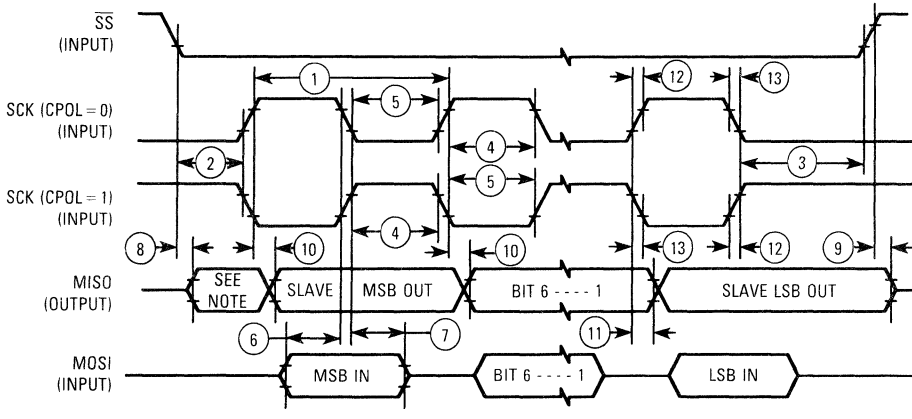
(b) SPI Master Timing (CPHA = 1)

Figure 13-10. SPI Timing Diagrams (Sheet 1 of 2)



NOTE: Not defined but normally MSB of character just received.

(c) SPI Slave Timing (CPHA = 0)



NOTE: Not defined but normally LSB of character previously transmitted.

(d) SPI SLave Timing (CPHA = 1)

Figure 13-10. SPI Timing Diagrams (Sheet 2 of 2)

13.10 EEPROM CHARACTERISTICS (V_{DD} = 5.0 Vdc ± 10%, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H)

Characteristic		−40 to 85°C	Unit
Programming Time (see Note 1)	Under 1.0 MHz with RC Oscillator Enabled	10	ms
	1.0 to 2.0 MHz with RC Oscillator Disabled	20	
	2.0 MHz (or Anytime RC Oscillator Enabled)	10	
Erase Time (see Note 1)		10	ms
Write Erase Endurance (see Note 2)		10,000	Cycles
Data Retention (see Note 2)		10	Years

NOTES:
1. The RC oscillator must be enabled (by setting the CSEL bit in the OPTION register) for EEPROM programming and erasure when the E-clock frequency is below 1.0 MHz.
2. See current quarterly Reliability Monitor report for current failure rate information.

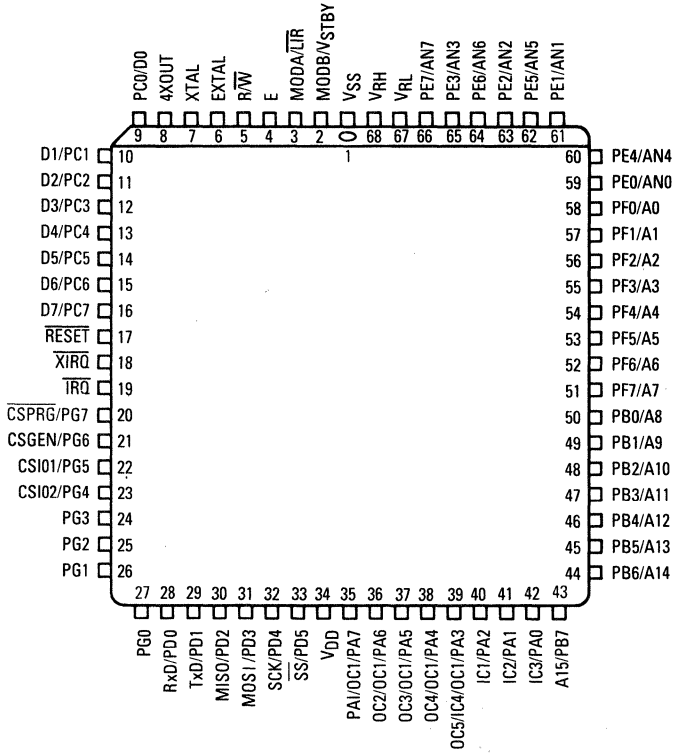
SECTION 14
MECHANICAL DATA

The following section contains the pin assignments, packaging dimensions, and ordering information for the MC68HC11F1 MCU.

14.1 ORDERING INFORMATION

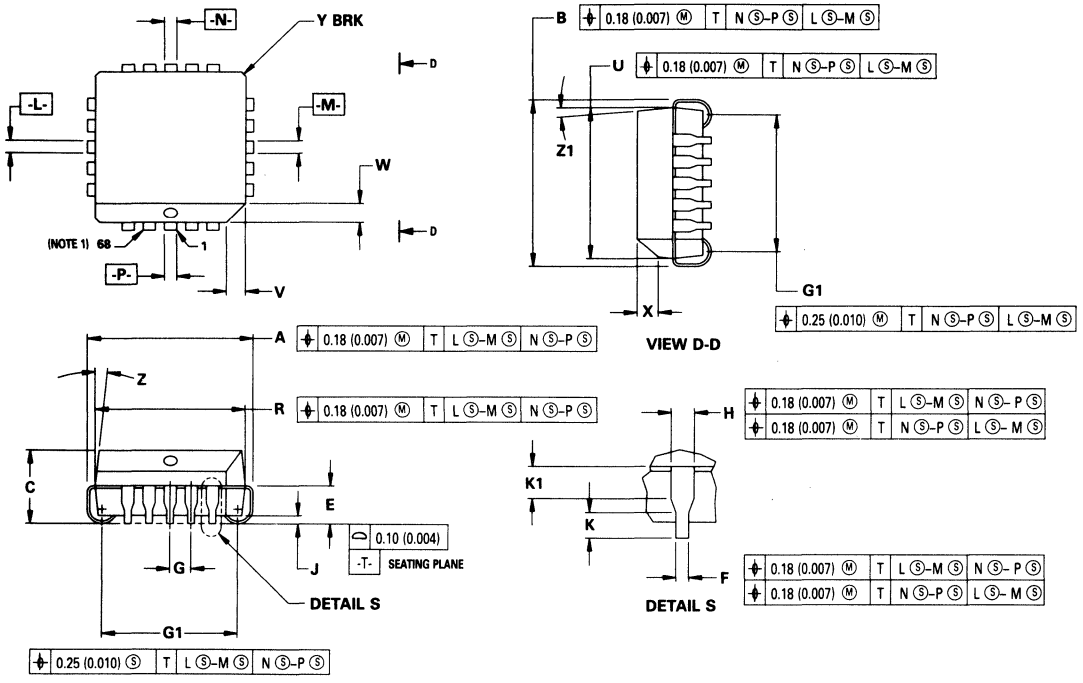
Package Type	Temperature	Part Number
PLCC (FN Suffix)	−40°C to 85°C	MC68HC11F1FN

14.2 PIN ASSIGNMENTS



14.3 PACKAGE DIMENSIONS

FN SUFFIX
PLASTIC LEADED
CHIP CARRIER
CASE 779-02



DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	25.02	25.27	0.985	0.995
B	25.02	25.27	0.985	0.995
C	4.20	4.57	0.165	0.180
E	2.29	2.79	0.090	0.110
F	0.33	0.48	0.013	0.019
G	1.27 BSC		0.050 BSC	
H	0.66	0.81	0.026	0.032
J	0.51	—	0.020	—
K	0.64	—	0.025	—
R	24.13	24.28	0.950	0.956
U	24.13	24.28	0.950	0.956
V	1.07	1.21	0.042	0.048
W	1.07	1.21	0.042	0.048
X	1.07	1.42	0.042	0.056
Y	—	0.50	—	0.020
Z	2°	10°	2°	10°
G1	23.12	23.62	0.910	0.930
K1	1.02	—	0.040	—
Z1	2°	10°	2°	10°

NOTES:
1. DUE TO SPACE LIMITATION, CASE 779-02 SHALL BE REPRESENTED BY A GENERAL (SMALLER) CASE OUTLINE DRAWING RATHER THAN SHOWING ALL 68 LEADS.
2. DATUMS -L-, -M-, -N-, AND -P- DETERMINED WHERE TOP OF LEAD SHOULDER EXIT PLASTIC BODY AT MOLD PARTING LINE.
3. DIM G1, TRUE POSITION TO BE MEASURED AT DATUM -T-, SEATING PLANE.
4. DIM R AND U DO NOT INCLUDE MOLD PROTRUSION. ALLOWABLE MOLD PROTRUSION IS 0.25 (0.010) PER SIDE.
5. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
6. CONTROLLING DIMENSION: INCH.

Literature Distribution Centers:

USA: Motorola Literature Distribution; P.O. Box 20912; Phoenix, Arizona 85036.
EUROPE: Motorola Ltd.; European Literature Center; 88 Tanners Drive, Blakelands, Milton Keynes, MK14 5BP, England.
JAPAN: Nippon Motorola Ltd.; 4-32-1, Nishi-Gotanda, Shinagawa-ku, Tokyo 141 Japan.
ASIA-PACIFIC: Motorola Semiconductors H.K. Ltd.; Silicon Harbour Center, No. 2 Dai King Street, Tai Po Industrial Estate, Tai Po, N.T., Hong Kong.

MOTOROLA

A23022-2 PRINTED IN USA 10/90 IMPERIAL LITHO 75138 18,000 MCU YGACAA