Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	HC11
Core Size	8-Bit
Speed	2MHz
Connectivity	SCI, SPI
Peripherals	POR, WDT
Number of I/O	38
Program Memory Size	-
Program Memory Type	ROMless
EEPROM Size	-
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	A/D 8x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	52-LCC (J-Lead)
Supplier Device Package	52-PLCC (19.13x19.13)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mc68hc11e0cfn2

Table of Contents

Section 3. Analog-to-Digital (A/D) Converter

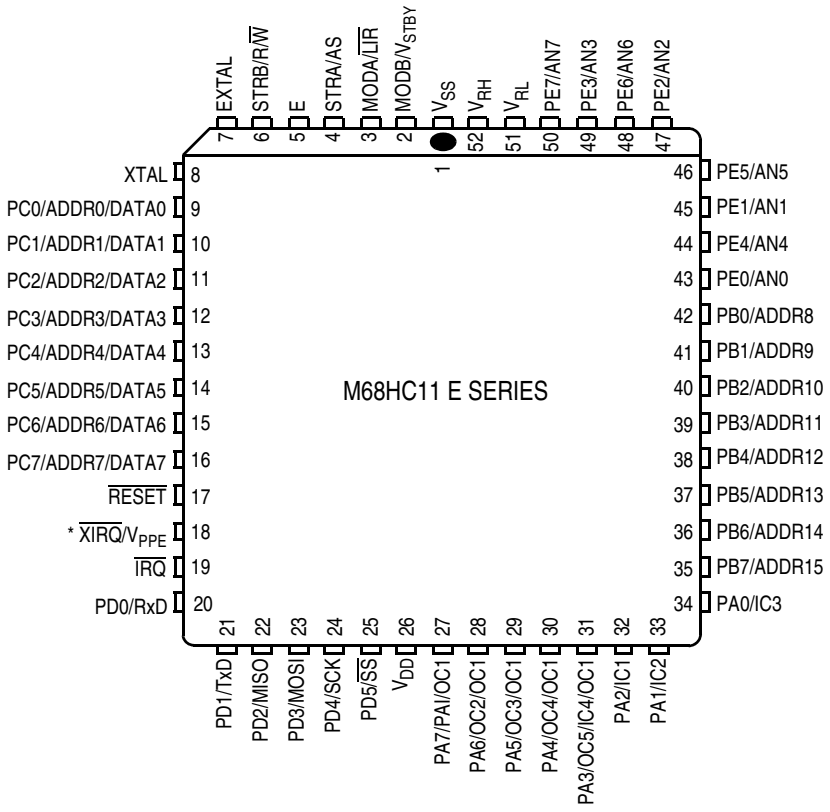
Section 4. Central Processor Unit (CPU)

1.4 Pin Descriptions

M68HC11 E-series MCUs are available packaged in:

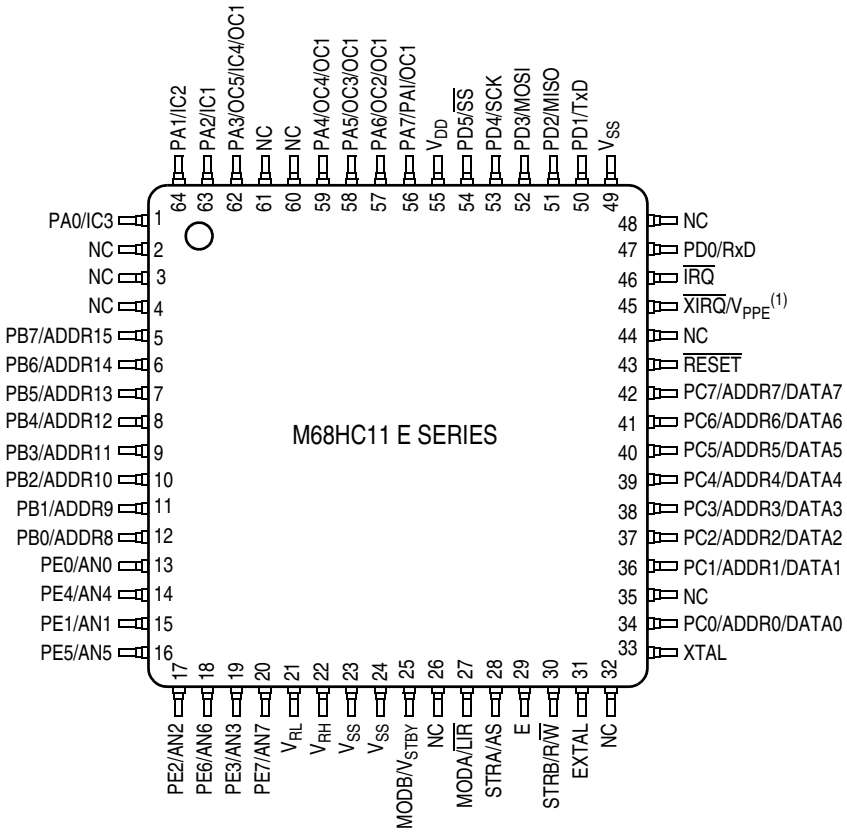
- 52-pin plastic-leaded chip carrier (PLCC)
- 52-pin windowed ceramic leaded chip carrier (CLCC)
- 52-pin plastic thin quad flat pack, 10 mm x 10 mm (TQFP)
- 64-pin quad flat pack (QFP)
- 48-pin plastic dual in-line package (DIP), MC68HC811E2 only
- 56-pin plastic shrink dual in-line package, .070-inch lead spacing (SDIP)

Most pins on these MCUs serve two or more functions, as described in the following paragraphs. Refer to **Figure 1-2**, **Figure 1-3**, **Figure 1-4**, **Figure 1-5**, and **Figure 1-6** which show the M68HC11 E-series pin assignments for the PLCC/CLCC, QFP, TQFP, SDIP, and DIP packages.



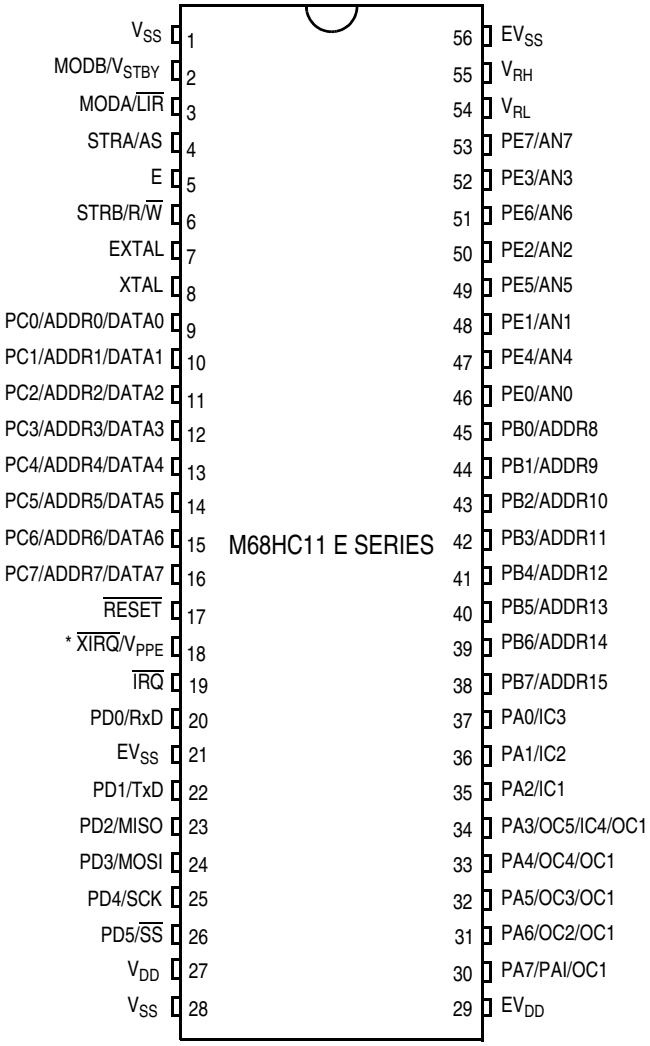
* V_{PPE} applies only to devices with EPROM/OTPROM.

Figure 1-2. Pin Assignments for 52-Pin PLCC and CLCC



1. V_{PPE} applies only to devices with EPROM/OTPROM.

Figure 1-3. Pin Assignments for 64-Pin QFP



* V_{PPE} applies only to devices with EPROM/OTPROM.

Figure 1-5. Pin Assignments for 56-Pin SDIP

The V_{STBY} pin is used to input random-access memory (RAM) standby power. When the voltage on this pin is more than one MOS threshold (about 0.7 volts) above the V_{DD} voltage, the internal RAM and part of the reset logic are powered from this signal rather than the V_{DD} input. This allows RAM contents to be retained without V_{DD} power applied to the MCU. Reset must be driven low before V_{DD} is removed and must remain low until V_{DD} has been restored to a valid level.

1.4.7.1 V_{RL} and V_{RH}

These two inputs provide the reference voltages for the analog-to-digital (A/D) converter circuitry:

- V_{RL} is the low reference, typically 0 Vdc.
- V_{RH} is the high reference.

For proper A/D converter operation:

- V_{RH} should be at least 3 Vdc greater than V_{RL}.
- V_{RL} and V_{RH} should be between V_{SS} and V_{DD}.

1.4.8 STRA/AS

The strobe A (STRA) and address strobe (AS) pin performs either of two separate functions, depending on the operating mode:

- In single-chip mode, STRA performs an input handshake (strobe input) function.
- In the expanded multiplexed mode, AS provides an address strobe function.

AS can be used to demultiplex the address and data signals at port C. Refer to **Section 2. Operating Modes and On-Chip Memory**.

1.4.9 STRB/R/$\overline{W}$

The strobe B (STRB) and read/write (R/$\overline{W}$) pin act as either an output strobe or as a data bus direction indicator, depending on the operating mode.

In single-chip operating mode, STRB acts as a programmable strobe for handshake with other parallel devices. Refer to **Section 6. Parallel Input/Output (I/O) Ports** for further information.

In expanded multiplexed operating mode, R/$\overline{W}$ is used to indicate the direction of transfers on the external data bus. A low on the R/$\overline{W}$ pin indicates 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 low during consecutive data bus write cycles, such as a double-byte store. It is possible for data to be driven out of port C, if internal read visibility (IRV) is enabled and an internal address is read, even though R/$\overline{W}$ is in a high-impedance state. Refer to **Section 2. Operating Modes and On-Chip Memory** for more information about IRVNE (internal read visibility not E).

A normal mode is selected when MODB is logic 1 during reset. One of three reset vectors is fetched from address $FFFA–$FFFF, and program execution begins from the address indicated by this vector. If MODB is logic 0 during reset, the special mode reset vector is fetched from addresses $BFFA–$BFFF, and software has access to special test features. Refer to **Section 5. Resets and Interrupts**.

Address: $103C

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	RBOOT(1)	SMOD(1)	MDA(1)	IRV(NE)(1)	PSEL3	PSEL2	PSEL1	PSEL0
Resets:								
Single chip:	0	0	0	0	0	1	1	0
Expanded:	0	0	1	0	0	1	1	0
Bootstrap:	1	1	0	0	0	1	1	0
Test:	0	1	1	1	0	1	1	0

1. The reset values depend on the mode selected at the RESET pin rising edge.

Figure 2-9. Highest Priority I-Bit Interrupt and Miscellaneous Register (HPRIO)

RBOOT — Read Bootstrap ROM Bit

Valid only when SMOD is set (bootstrap or special test mode); can be written only in special modes

0 = Bootloader ROM disabled and not in map

1 = Bootloader ROM enabled and in map at $BE00–$BFFF

SMOD and MDA — Special Mode Select and Mode Select A Bits

The initial value of SMOD is the inverse of the logic level present on the MODB pin at the rising edge of reset. The initial value of MDA equals the logic level present on the MODA pin at the rising edge of reset. These two bits can be read at any time. They can be written anytime in special modes. MDA can be written only once in normal modes. SMOD cannot be set once it has been cleared.

Input		Mode	Latched at Reset	
MODB	**MODA**		**SMOD**	**MDA**
1	0	Single chip	0	0
1	1	Expanded	0	1
0	0	Bootstrap	1	0
0	1	Special test	1	1

2.5 EEPROM

Some E-series devices contain 512 bytes of on-chip EEPROM. The MC68HC811E2 contains 2048 bytes of EEPROM with selectable base address. All E-series devices contain the EEPROM-based CONFIG register.

2.5.1 EEPROM and CONFIG Programming and Erasure

The erased state of an EEPROM bit is 1. During a read operation, bit lines are precharged to 1. The floating gate devices of programmed bits conduct and pull the bit lines to 0. Unprogrammed bits remain at the precharged level and are read as ones. Programming a bit to 1 causes no change. Programming a bit to 0 changes the bit so that subsequent reads return 0.

When appropriate bits in the BPROT register are cleared, the PPROG register controls programming and erasing the EEPROM. The PPROG register can be read or written at any time, but logic enforces defined programming and erasing sequences to prevent unintentional changes to EEPROM data. When the EELAT bit in the PPROG register is cleared, the EEPROM can be read as if it were a ROM.

The on-chip charge pump that generates the EEPROM programming voltage from V_{DD} uses MOS capacitors, which are relatively small in value. The efficiency of this charge pump and its drive capability are affected by the level of V_{DD} and the frequency of the driving clock. The load depends on the number of bits being programmed or erased and capacitances in the EEPROM array.

The clock source driving the charge pump is software selectable. When the clock select (CSEL) bit in the OPTION register is 0, the E clock is used; when CSEL is 1, an on-chip resistor-capacitor (RC) oscillator is used.

The EEPROM programming voltage power supply voltage to the EEPROM array is not enabled until there has been a write to PPROG with EELAT set and PGM cleared. This must be followed by a write to a valid EEPROM location or to the CONFIG address, and then a write to PPROG with both the EELAT and EPGM bits set. Any attempt to set both EELAT and EPGM during the same write operation results in neither bit being set.

2.5.1.1 Block Protect Register

This register prevents inadvertent writes to both the CONFIG register and EEPROM. The active bits in this register are initialized to 1 out of reset and can be cleared only during the first 64 E-clock cycles after reset in the normal modes. When these bits are cleared, the associated EEPROM section and the CONFIG register can be programmed or erased. EEPROM is only visible if the EEON bit in the CONFIG register is set. The bits in the BPROT register can be written to 1 at any time to protect EEPROM and the CONFIG register. In test or bootstrap modes, write protection is inhibited and BPROT can be written repeatedly. Address ranges for protected areas of EEPROM differ significantly for the MC68HC811E2. Refer to **Figure 2-16**.

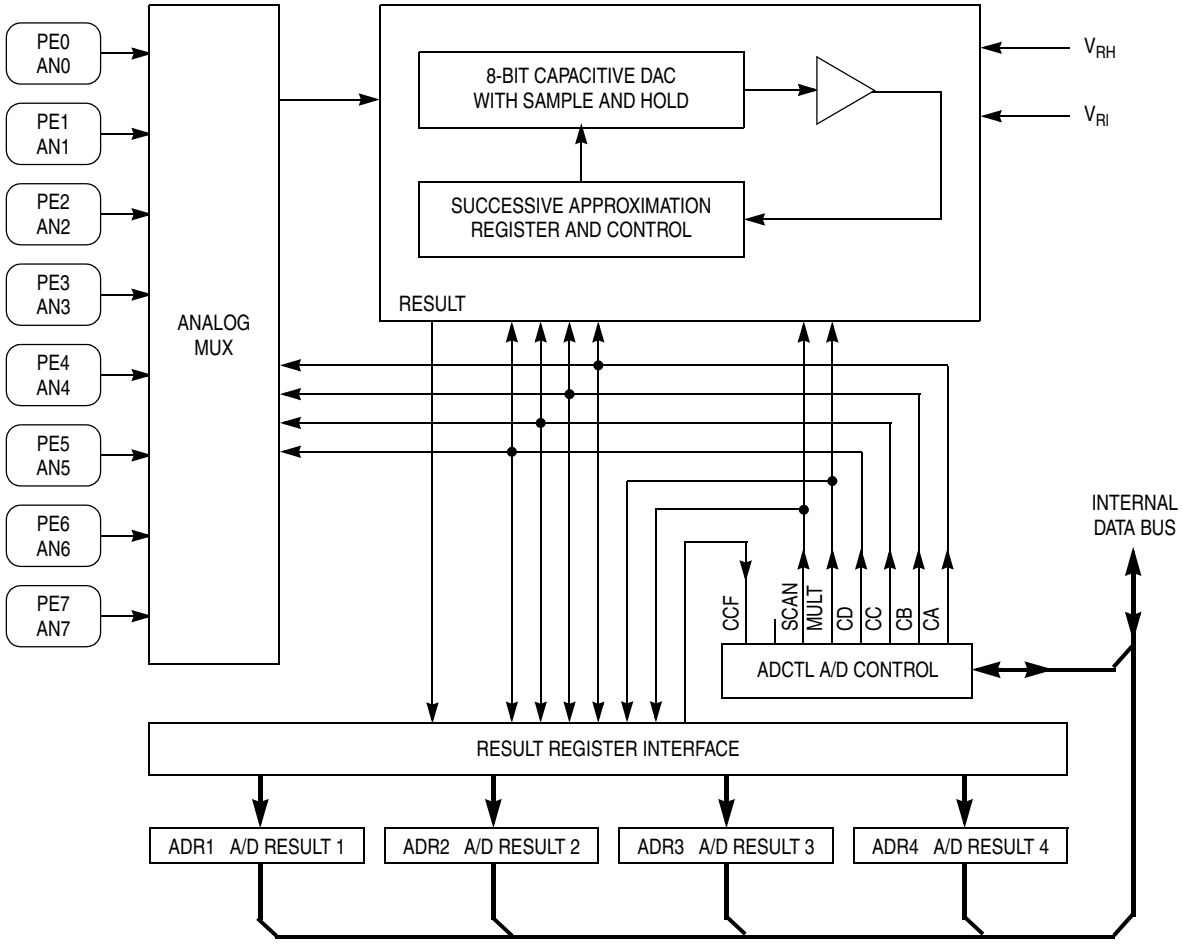
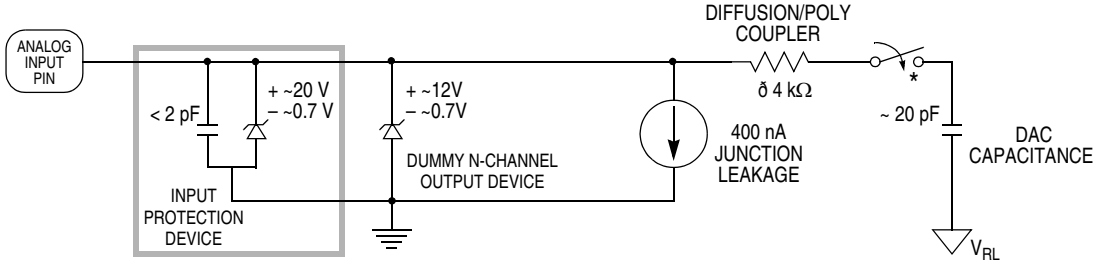


Figure 3-1. A/D Converter Block Diagram



* THIS ANALOG SWITCH IS CLOSED ONLY DURING THE 12-CYCLE SAMPLE TIME.

Figure 3-2. Electrical Model of an A/D Input Pin (Sample Mode)

Table 3-1. Converter Channel Assignments

Channel Number	Channel Signal	Result in ADRx if MULT = 1
1	AN0	ADR1
2	AN1	ADR2
3	AN2	ADR3
4	AN3	ADR4
5	AN4	ADR1
6	AN5	ADR2
7	AN6	ADR3
8	AN7	ADR4
9 – 12	Reserved	—
13	V_{RH}(1)	ADR1
14	V_{RL}(1)	ADR2
15	$(V_{RH})/2$(1)	ADR3
16	Reserved(1)	ADR4

1. Used for factory testing

3.6 Single-Channel Operation

The two types of single-channel operation are:

1. When SCAN = 0, the single selected channel is converted four consecutive times. The first result is stored in A/D result register 1 (ADR1), and the fourth result is stored in ADR4. After the fourth conversion is complete, all conversion activity is halted until a new conversion command is written to the ADCTL register.
2. When SCAN = 1, conversions continue to be performed on the selected channel with the fifth conversion being stored in register ADR1 (overwriting the first conversion result), the sixth conversion overwriting ADR2, and so on.

3.7 Multiple-Channel Operation

The two types of multiple-channel operation are:

1. When SCAN = 0, a selected group of four channels is converted one time each. The first result is stored in A/D result register 1 (ADR1), and the fourth result is stored in ADR4. After the fourth conversion is complete, all conversion activity is halted until a new conversion command is written to the ADCTL register.
2. When SCAN = 1, conversions continue to be performed on the selected group of channels with the fifth conversion being stored in register ADR1 (replacing the earlier conversion result for the first channel in the group), the sixth conversion overwriting ADR2, and so on.

operation of the CPU continues uninterrupted until the I bit is cleared. After any reset, the I bit is set by default and can only be cleared by a software instruction. When an interrupt is recognized, the I bit is set after the registers are stacked, but before the interrupt vector is fetched. After the interrupt has been serviced, a return-from-interrupt instruction is normally executed, restoring the registers to the values that were present before the interrupt occurred. Normally, the I bit is 0 after a return from interrupt is executed. Although the I bit can be cleared within an interrupt service routine, "nesting" interrupts in this way should only be done when there is a clear understanding of latency and of the arbitration mechanism. Refer to **Section 5. Resets and Interrupts**.

4.2.6.6 Half Carry (H)

The H bit is set when a carry occurs between bits 3 and 4 of the arithmetic logic unit during an ADD, ABA, or ADC instruction. Otherwise, the H bit is cleared. Half carry is used during BCD operations.

4.2.6.7 X Interrupt Mask (X)

The XIRQ mask (X) bit disables interrupts from the $\overline{\text{XIRQ}}$ pin. After any reset, X is set by default and must be cleared by a software instruction. When an $\overline{\text{XIRQ}}$ interrupt is recognized, the X and I bits are set after the registers are stacked, but before the interrupt vector is fetched. After the interrupt has been serviced, an RTI instruction is normally executed, causing the registers to be restored to the values that were present before the interrupt occurred. The X interrupt mask bit is set only by hardware ($\overline{\text{RESET}}$ or $\overline{\text{XIRQ}}$ acknowledge). X is cleared only by program instruction (TAP, where the associated bit of A is 0; or RTI, where bit 6 of the value loaded into the CCR from the stack has been cleared). There is no hardware action for clearing X.

4.2.6.8 STOP Disable (S)

Setting the STOP disable (S) bit prevents the STOP instruction from putting the M68HC11 into a low-power stop condition. If the STOP instruction is encountered by the CPU while the S bit is set, it is treated as a no-operation (NOP) instruction, and processing continues to the next instruction. S is set by reset; STOP is disabled by default.

4.3 Data Types

The M68HC11 CPU supports four data types:

1. Bit data
2. 8-bit and 16-bit signed and unsigned integers
3. 16-bit unsigned fractions
4. 16-bit addresses

5.2.2 External Reset ($\overline{\text{RESET}}$)

The CPU distinguishes between internal and external reset conditions by sensing whether the reset pin rises to a logic 1 in less than two E-clock cycles after an internal device releases reset. When a reset condition is sensed, the $\overline{\text{RESET}}$ pin is driven low by an internal device for four E-clock cycles, then released. Two E-clock cycles later it is sampled. If the pin is still held low, the CPU assumes that an external reset has occurred. If the pin is high, it indicates that the reset was initiated internally by either the COP system or the clock monitor.

CAUTION: *Do not connect an external resistor capacitor (RC) power-up delay circuit to the reset pin of M68HC11 devices because the circuit charge time constant can cause the device to misinterpret the type of reset that occurred.*

5.2.3 Computer Operating Properly (COP) Reset

The MCU includes a COP system to help protect against software failures. When the COP is enabled, the software is responsible for keeping a free-running watchdog timer from timing out. When the software is no longer being executed in the intended sequence, a system reset is initiated.

The state of the NOCOP bit in the CONFIG register determines whether the COP system is enabled or disabled. To change the enable status of the COP system, change the contents of the CONFIG register and then perform a system reset. In the special test and bootstrap operating modes, the COP system is initially inhibited by the disable resets (DISR) control bit in the TEST1 register. The DISR bit can subsequently be written to 0 to enable COP resets.

The COP timer rate control bits CR[1:0] in the OPTION register determine the COP timeout period. The system E clock is divided by 2^{15} and then further scaled by a factor shown in **Table 5-1**. After reset, these bits are 0, which selects the fastest timeout period. In normal operating modes, these bits can be written only once within 64 bus cycles after reset.

Table 5-1. COP Timer Rate Select

CR[1:0]	Divide E/2^{15} By	XTAL = 4.0 MHz Timeout – 0 ms, + 32.8 ms	XTAL = 8.0 MHz Timeout – 0 ms, + 16.4 ms	XTAL = 12.0 MHz Timeout – 0 ms, + 10.9 ms	XTAL = 16.0 MHz Timeout – 0 ms, + 8.2 ms
0 0	1	32.768 ms	16.384 ms	10.923 ms	8.19 ms
0 1	4	131.072 ms	65.536 ms	43.691 ms	32.8 ms
1 0	16	524.28 ms	262.14 ms	174.76 ms	131 ms
1 1	64	2.098 s	1.049 s	699.05 ms	524 ms
	E =	1.0 MHz	2.0 MHz	3.0 MHz	4.0 MHz

Section 8. Serial Peripheral Interface (SPI)

8.1 Introduction

The serial peripheral interface (SPI), an independent serial communications subsystem, allows the MCU to communicate synchronously with peripheral devices, such as:

- Frequency synthesizers
- Liquid crystal display (LCD) drivers
- Analog-to-digital (A/D) converter subsystems
- Other microprocessors

The SPI is also capable of inter-processor communication in a multiple master system. The SPI system can be configured as either a master or a slave device. When configured as a master, data transfer rates can be as high as one-half the E-clock rate (1.5 Mbits per second for a 3-MHz bus frequency). When configured as a slave, data transfers can be as fast as the E-clock rate (3 Mbits per second for a 3-MHz bus frequency).

8.2 Functional Description

The central element in the SPI system is the block containing the shift register and the read data buffer. The system is single buffered in the transmit direction and double buffered in the receive direction. This means that new data for transmission cannot be written to the shifter until the previous transfer is complete; however, received data is transferred into a parallel read data buffer so the shifter is free to accept a second serial character. As long as the first character is read out of the read data buffer before the next serial character is ready to be transferred, no overrun condition occurs. A single MCU register address is used for reading data from the read data buffer and for writing data to the shifter.

The SPI status block represents the SPI status functions (transfer complete, write collision, and mode fault) performed by the serial peripheral status register (SPSR). The SPI control block represents those functions that control the SPI system through the serial peripheral control register (SPCR).

Refer to **Figure 8-1**, which shows the SPI block diagram.

9.3.3 Timer Input Capture 4/Output Compare 5 Register

Use TI4/O5 as either an input capture register or an output compare register, depending on the function chosen for the PA3 pin. To enable it as an input capture pin, set the I4/O5 bit in the pulse accumulator control register (PACTL) to logic level 1. To use it as an output compare register, set the I4/O5 bit to a logic level 0. Refer to **9.7 Pulse Accumulator**.

Register name: Timer Input Capture 4/Output Compare 5 (High) Address: $101E

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
Reset:	1	1	1	1	1	1	1	1

Register name: Timer Input Capture 4/Output Compare 5 (Low) Address: $101F

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

Figure 9-7. Timer Input Capture 4/Output Compare 5 Register Pair (TI4/O5)

9.4 Output Compare

Use the output compare (OC) function to program an action to occur at a specific time — when the 16-bit counter reaches a specified value. For each of the five output compare functions, there is a separate 16-bit compare register and a dedicated 16-bit comparator. The value in the compare register is compared to the value of the free-running counter on every bus cycle. When the compare register matches the counter value, an output compare status flag is set. The flag can be used to initiate the automatic actions for that output compare function.

To produce a pulse of a specific duration, write a value to the output compare register that represents the time the leading edge of the pulse is to occur. The output compare circuit is configured to set the appropriate output either high or low, depending on the polarity of the pulse being produced. After a match occurs, the output compare register is reprogrammed to change the output pin back to its inactive level at the next match. A value representing the width of the pulse is added to the original value, and then written to the output compare register. Because the pin state changes occur at specific values of the free-running counter, the pulse width can be controlled accurately at the resolution of the free-running counter, independent of software latencies. To generate an output signal of a specific frequency and duty cycle, repeat this pulse-generating procedure.

The five 16-bit read/write output compare registers are: TOC1, TOC2, TOC3, and TOC4, and the TI4/O5. TI4/O5 functions under software control as either IC4 or OC5. Each of the OC registers is set to $FFFF on reset. A value written to an OC

10.11 Peripheral Port Timing

Characteristic[1] [2]	Symbol	1.0 MHz		2.0 MHz		3.0 MHz		Unit
		Min	Max	Min	Max	Min	Max	
Frequency of operation E-clock frequency	f_o	dc	1.0	dc	2.0	dc	3.0	MHz
E-clock period	t_{CYC}	1000	—	500	—	333	—	ns
Peripheral data setup time MCU read of ports A, C, D, and E	t_{PDSU}	100	—	100	—	100	—	ns
Peripheral data hold time MCU read of ports A, C, D, and E	t_{PDH}	50	—	50	—	50	—	ns
Delay time, peripheral data write t_{PWD} = 1/4 t_{CYC}+ 100 ns MCU writes to port A MCU writes to ports B, C, and D	t_{PWD}	 — —	 200 350	 — —	 200 225	 — —	 200 183	ns
Port C input data setup time	t_{IS}	60	—	60	—	60	—	ns
Port C input data hold time	t_{IH}	100	—	100	—	100	—	ns
Delay time, E fall to STRB t_{DEB} = 1/4 t_{CYC}+ 100 ns	t_{DEB}	—	350	—	225	—	183	ns
Setup time, STRA asserted to E fall[3]	t_{AES}	0	—	0	—	0	—	ns
Delay time, STRA asserted to port C data output valid	t_{PCD}	—	100	—	100	—	100	ns
Hold time, STRA negated to port C data	t_{PCH}	10	—	10	—	10	—	ns
3-state hold time	t_{PCZ}	—	150	—	150	—	150	ns

1. V_{DD} = 5.0 Vdc ± 10%, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H, all timing is shown with respect to 20% V_{DD} and 70% V_{DD}, unless otherwise noted
2. Ports C and D timing is valid for active drive. (CWOM and DWOM bits are not set in PIOC and SPCR registers, respectively.)
3. If this setup time is met, STRB acknowledges in the next cycle. If it is not met, the response may be delayed one more cycle.

Section 11. Ordering Information and Mechanical Specifications

11.1 Introduction

This section provides ordering information for the E-series devices grouped by:

- Standard devices
- Custom ROM devices
- Extended voltage devices

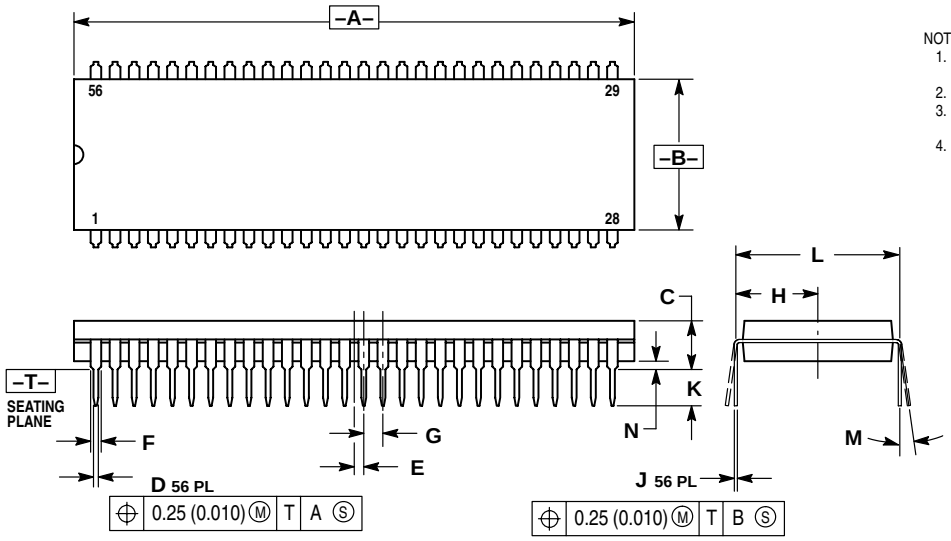
In addition, mechanical specifications for the following packaging options:

- 52-pin plastic-leaded chip carrier (PLCC)
- 52-pin windowed ceramic-leaded chip carrier (CLCC)
- 64-pin quad flat pack (QFP)
- 52-pin thin quad flat pack (TQFP)
- 56-pin shrink dual in-line package with .070-inch lead spacing (SDIP)
- 48-pin plastic DIP (.100-inch lead spacing), MC68HC811E2 only

11.2 Standard Device Ordering Information

Description	CONFIG	Temperature	Frequency	MC Order Number
52-pin plastic leaded chip carrier (PLCC)				
BUFFALO ROM	$0F	–40°C to +85°C	2 MHz	MC68HC11E9BCFN2
			3 MHz	MC68HC11E9BCFN3
No ROM	$0D	–40°C to +85°C	2 MHz	MC68HC11E1CFN2
			3 MHz	MC68HC11E1CFN3
		–40°C to +105°C	2 MHz	MC68HC11E1VFN2
		–40°C to +125°C	2 MHz	MC68HC11E1MFN2
No ROM, no EEPROM	$0C	–40°C to +85°C	2 MHz	MC68HC11E0CFN2
			3 MHz	MC68HC11E0CFN3
		–40°C to +105°C	2 MHz	MC68HC11E0VFN2
		–40°C to +125°C	2 MHz	MC68HC11E0MFN2

11.9 56-Pin Dual in-Line Package (Case 859)

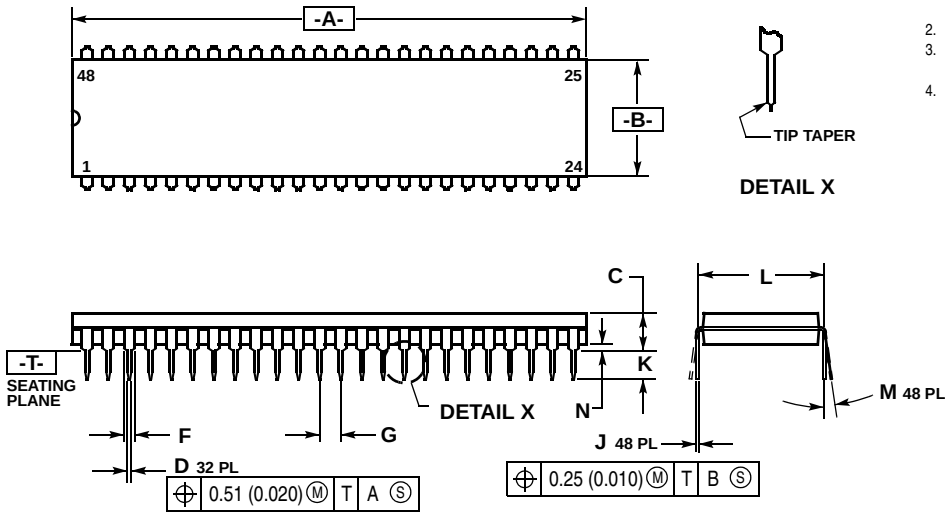


NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: INCH.
3. DIMENSION L TO CENTER OF LEAD WHEN FORMED PARALLEL.
4. DIMENSIONS A AND B DO NOT INCLUDE MOLD FLASH. MAXIMUM MOLD FLASH 0.25 (0.010)

	INCHES		MILLIMETERS	
DIM	MIN	MAX	MIN	MAX
A	2.035	2.065	51.69	52.45
B	0.540	0.560	13.72	14.22
C	0.155	0.200	3.94	5.08
D	0.014	0.022	0.36	0.56
E	0.035 BSC		0.89 BSC	
F	0.032	0.046	0.81	1.17
G	0.070 BSC		1.778 BSC	
H	0.300 BSC		7.62 BSC	
J	0.008	0.015	0.20	0.38
K	0.115	0.135	2.92	3.43
L	0.600 BSC		15.24 BSC	
M	0 °	15 °	0 °	15 °
N	0.020	0.040	0.51	1.02

11.10 48-Pin Plastic DIP (Case 767)

NOTE: *The MC68HC811E2 is the only member of the E series that is offered in a 48-pin plastic dual in-line package.*



NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: INCH.
3. DIMENSION L TO CENTER OF LEAD WHEN FORMED PARALLEL.
4. DIMENSIONS A AND B DO NOT INCLUDE MOLD FLASH. MAXIMUM MOLD FLASH 0.25 (0.010).

	INCHES		MILLIMETERS	
DIM	MIN	MAX	MIN	MAX
A	2.415	2.445	61.34	62.10
B	0.540	0.560	13.72	14.22
C	0.155	0.200	3.94	5.08
D	0.014	0.022	0.36	0.55
F	0.040	0.060	1.02	1.52
G	0.100 BSC		2.54 BSC	
H	0.070 BSC		1.79 BSC	
J	0.008	0.015	0.20	0.38
K	0.115	0.150	2.92	3.81
L	0.600 BSC		15.24 BSC	
M	0×	15×	0×	15×
N	0.020	0.040	0.51	1.01

pins during reset, the selected mode, and the state of the MDA, SMOD, and RBOOT control bits. Refer to the composite memory map and information in **Table 1** for the following discussion.

The MDA control bit is determined by the state of the MODA pin as the MCU leaves reset. MDA selects between single-chip and expanded operating modes. When MDA is 0, a single-chip mode is selected, either normal single-chip mode or special bootstrap mode. When MDA is 1, an expanded mode is selected, either normal expanded mode or special test mode.

The SMOD control bit is determined by the inverted state of the MODB pin as the MCU leaves reset. SMOD controls whether a normal mode or a special mode is selected. When SMOD is 0, one of the two normal modes is selected, either normal single-chip mode or normal expanded mode. When SMOD is 1, one of the two special modes is selected, either special bootstrap mode or special test mode. When either special mode is in effect (SMOD = 1), certain privileges are in effect, for instance, the ability to write to the mode control bits and fetching the reset and interrupt vectors from $BFxx rather than $FFxx.

Table 1. Mode Selection Summary

Input Pins		Mode Selected	Control Bits in HPRIO		
MODB	MODA		RBOOT	SMOD	MDA
1	0	Normal single chip	0	0	0
0	0	Normal expanded	0	0	1
0	0	Special bootstrap	1	1	0
0	1	Special test	0	1	1

The alternate vector locations are achieved by simply driving address bit A14 low during all vector fetches if SMOD = 1. For special test mode, the alternate vector locations assure that the reset vector can be fetched from external memory space so the test system can control MCU operation. In special bootstrap mode, the small boot ROM is enabled in the memory map by RBOOT = 1 so the reset vector will be fetched from this ROM and the bootloader firmware will control MCU operation.

```
                        * MEMORY CONFIGURATION EQUATES
                        *
B600                    EEPMSTR  EQU    $B600          Start of EEPROM
B7FF                    EEPMEND  EQU    $B7FF          End of EEPROM
                        *
D000                    EPRMSTR  EQU    $D000          Start of EPROM
FFFF                    EPRMEND  EQU    $FFFF          End of EPROM
                        *
0000                    RAMSTR   EQU    $0000
01FF                    RAMEND   EQU    $01FF

                        * DELAY CONSTANTS
                        *
0DB0                    DELAYS   EQU    3504           Delay at slow baud
021B                    DELAYF   EQU    539            Delay at fast baud
                        *
1068                    PROGDEL  EQU    4200           2 ms programming delay
                        *                       At 2.1 MHz

                        ***************************************************
BF00                     ORG    $BF00
                        ***************************************************

                        * Next two instructions provide a predictable place
                        * to call PROGRAM and UPLOAD even if the routines
                        * change size in future versions.
                        *
BF00 7EBF13             PROGRAM  JMP    PRGROUT        EPROM programming utility
BF03                    UPLOAD   EQU    *              Upload utility

                        ***************************************************
                        * UPLOAD - Utility subroutine to send data from
                        * inside the MCU to the host via the SCI interface.
                        * Prior to calling UPLOAD set baud rate, turn on SCI
                        * and set Y=first address to upload.
                        * Bootloader leaves baud set, SCI enabled, and
                        * Y pointing at EPROM start ($D000) so these default
                        * values do not have to be changed typically.
                        * Consecutive locations are sent via SCI in an
                        * infinite loop. Reset stops the upload process.
                        ***************************************************
BF03 CE1000                      LDX    #$1000         Point to internal registers
BF06 18A600             UPLOOP   LDAA   0,Y            Read byte
BF09 1F2E80FC                    BRCLR  SCSR,X $80 *   Wait for TDRE
BF0D A72F                        STAA   SCDAT,X        Send it
BF0F 1808                        INY
BF11 20F3                        BRA    UPLOOP         Next...
```

```
Symbol Table:

Symbol Name     Value  Def.#  Line Number Cross Reference

BAUD             002B *00037  00160 00180
BAUDOK           BF8A *00183  00178
BEGIN            BF54 *00155  00250
DELAYF           021B *00061  00163
DELAYS           0DB0 *00060  00181
DONEIT           BF47 *00142  00124
EEPMEND          B7FF *00050
EEPMSTR          B600 *00049  00175
ELAT             0020 *00043  00125 00128
EPGM             0001 *00044  00128
EPRMEND          FFFF *00053
EPRMSTR          D000 *00052  00206
NEWONE           BF9B *00196  00189
NOTZERO          BF7E *00176  00174
OC1F             0080 *00034  00136 00139
PORTD            0008 *00029  00168
PPROG            003B *00041  00126 00129 00140
PRGROUT          BF13 *00110  00074
PROGDEL          1068 *00063  00205
PROGRAM          BF00 *00074
RAMEND           01FF *00056  00156 00201
RAMSTR           0000 *00055  00184 00207
SCCR2            002D *00038  00162 00167 00169
SCDAT            002F *00040  00091 00118 00122 00145 00172 00197 00199
SCSR             002E *00039  00090 00116 00121 00143 00171 00189
SPCR             0028 *00036  00158
STAR             BFAA *00204  00194
TCNT             000E *00030  00134
TFLG1            0023 *00032  00137 00139
TOC1             0016 *00031  00135 00164 00182 00187
UPLOAD           BF03 *00075
UPLOOP           BF06 *00089  00093
WAIT             BF8E *00186  00202
WAIT1            BF1F *00120  00147
WTLOOP           BF90 *00188  00193

                   Errors: None
                   Labels: 35
     Last Program Address: $BFFF
     Last Storage Address: $0000
            Program Bytes: $0100  256
            Storage Bytes: $0000  0
```