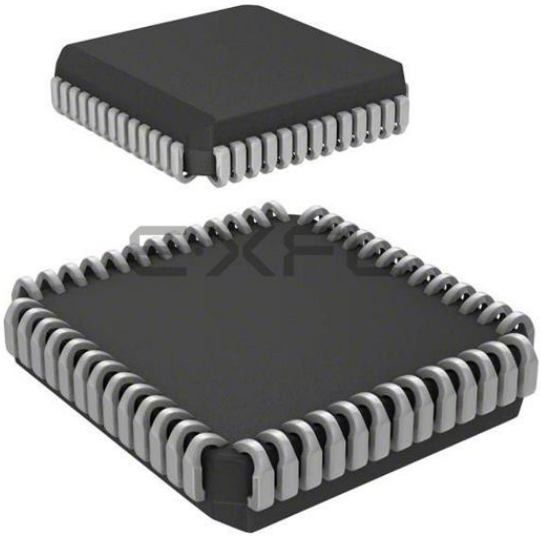


Welcome to E-XFL.COM

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)	3V ~ 5.5V
Data Converters	A/D 8x8b
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	52-LCC (J-Lead)
Supplier Device Package	52-PLCC (19.1x19.1)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc68l11e0fne2

Chapter 3
Analog-to-Digital (A/D) Converter

Chapter 4
Central Processor Unit (CPU)

Chapter 5
Resets and Interrupts

1.4.2 $\overline{\text{RESET}}$

A bidirectional control signal, $\overline{\text{RESET}}$, acts as an input to initialize the MCU to a known startup state. It also acts as an open-drain output to indicate that an internal failure has been detected in either the clock monitor or computer operating properly (COP) watchdog circuit. 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 a reset has occurred. See Figure 1-7 and Figure 1-8.

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.

Because the CPU is not able to fetch and execute instructions properly when V_{DD} falls below the minimum operating voltage level, reset must be controlled. A low-voltage inhibit (LVI) circuit is required primarily for protection of EEPROM contents. However, since the configuration register (CONFIG) value is read from the EEPROM, protection is required even if the EEPROM array is not being used.

Presently, there are several economical ways to solve this problem. For example, two good external components for LVI reset are:

1. The Seiko S0854HN (or other S805 series devices):
   a. Extremely low power (2 µA)
   a. TO-92 package
   a. Limited temperature range, –20°C to +70°C
   a. Available in various trip-point voltage ranges
2. The Freescale MC34064:
   a. TO-92 or SO-8 package
   a. Draws about 300 µA
   a. Temperature range –40°C to 85°C
   a. Well controlled trip point
   a. Inexpensive

Refer to Chapter 5 Resets and Interrupts for further information.

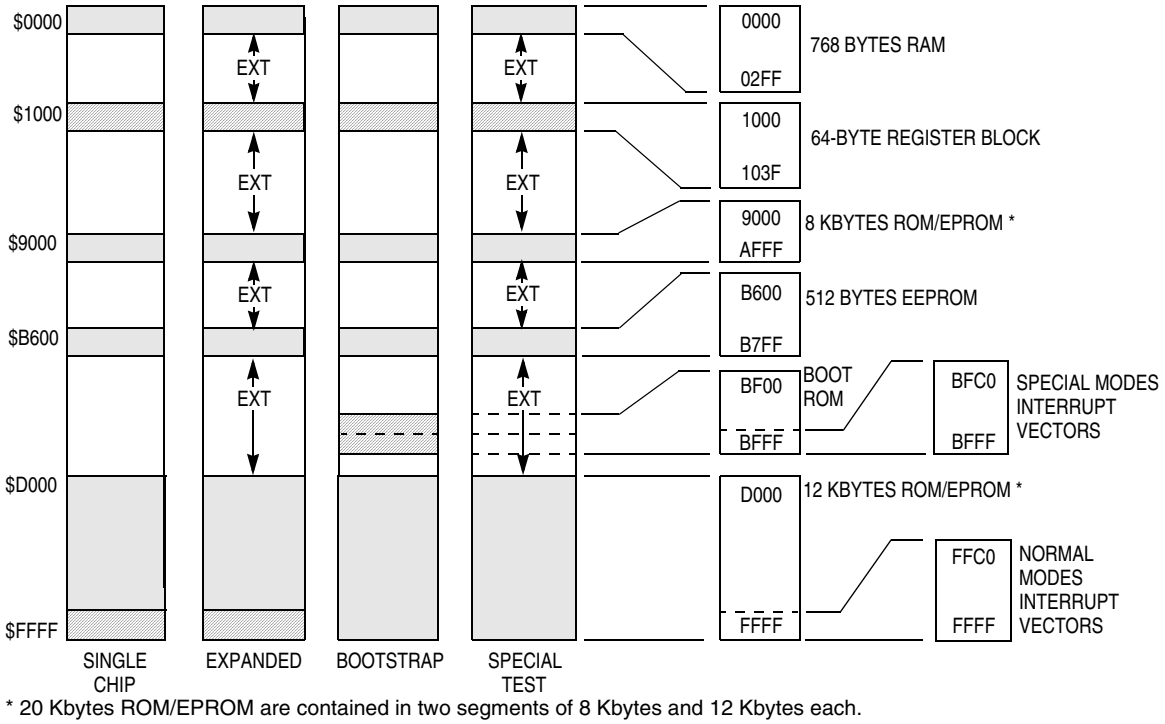
1.4.3 Crystal Driver and External Clock Input (XTAL and EXTAL)

These two pins provide the interface for either a crystal or a CMOS- compatible clock to control the internal clock generator circuitry. The frequency applied to these pins is four times higher than the desired E-clock rate.

The XTAL pin must be left unterminated when an external CMOS- compatible clock input is connected to the EXTAL pin. The XTAL output is normally intended to drive only a crystal. Refer to Figure 1-9 and Figure 1-10.

CAUTION

In all cases, use caution around the oscillator pins. Load capacitances shown in the oscillator circuit are specified by the crystal manufacturer and should include all stray layout capacitances.



* 20 Kbytes ROM/EPROM are contained in two segments of 8 Kbytes and 12 Kbytes each.

Figure 2-5. Memory Map for MC68HC(7)11E20

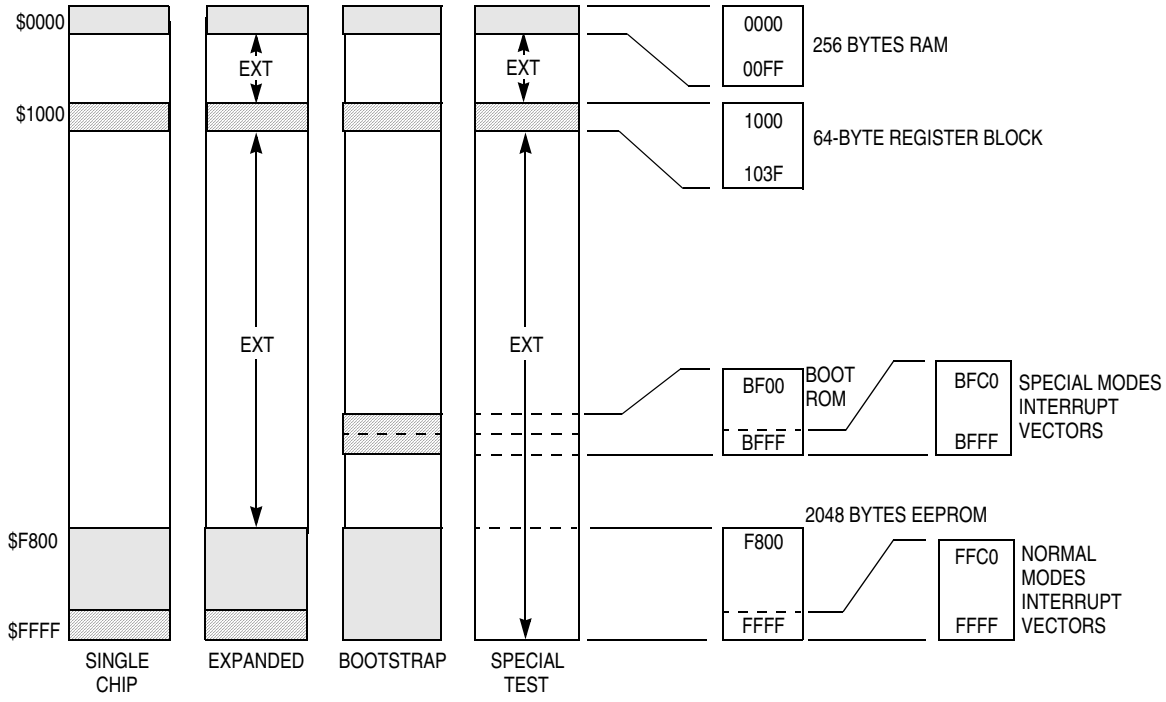


Figure 2-6. Memory Map for MC68HC811E2

0	0	Bootstrap	1	0
0	1	Special test	1	1

IRV(NE) — Internal Read Visibility (Not E) Bit

IRVNE can be written once in any mode. In expanded modes, IRVNE determines whether IRV is on or off. In special test mode, IRVNE is reset to 1. In all other modes, IRVNE is reset to 0. For the MC68HC811E2, this bit is IRV and only controls the internal read visibility function.

0 = No internal read visibility on external bus

1 = Data from internal reads is driven out the external data bus.

In single-chip modes this bit determines whether the E clock drives out from the chip. For the MC68HC811E2, this bit has no meaning or effect in single-chip and bootstrap modes.

0 = E is driven out from the chip.

1 = E pin is driven low. Refer to the following table.

Mode	IRVNE Out of Reset	E Clock Out of Reset	IRV Out of Reset	IRVNE Affects Only	IRVNE Can Be Written
Single chip	0	On	Off	E	Once
Expanded	0	On	Off	IRV	Once
Bootstrap	0	On	Off	E	Once
Special test	1	On	On	IRV	Once

PSEL[3:0] — Priority Select Bits

Refer to Chapter 5 Resets and Interrupts.

2.3.3 System Initialization

Registers and bits that control initialization and the basic operation of the MCU are protected against writes except under special circumstances. Table 2-2 lists registers that can be written only once after reset or that must be written within the first 64 cycles after reset.

Table 2-2. Write Access Limited Registers

Operating Mode	Register Address	Register Name	Must be Written in First 64 Cycles	Write Anytime
SMOD = 0	$x024	Timer interrupt mask 2 (TMSK2)	Bits [1:0], once only	Bits [7:2]
	$x035	Block protect register (BPROT)	Clear bits, once only	Set bits only
	$x039	System configuration options (OPTION)	Bits [5:4], bits [2:0], once only	Bits [7:6], bit 3
	$x03C	Highest priority I-bit interrupt and miscellaneous (HPRIO)	See HPRIO description	See HPRIO description
	$x03D	RAM and I/O map register (INIT)	Yes, once only	—
SMOD = 1	$x024	Timer interrupt mask 2 (TMSK2)	—	All, set or clear
	$x035	Block protect register (BPROT)	—	All, set or clear
	$x039	System configuration options (OPTION)	—	All, set or clear
	$x03C	Highest priority I-bit interrupt and miscellaneous (HPRIO)	See HPRIO description	See HPRIO description
	$x03D	RAM and I/O map register (INIT)	—	All, set or clear

NOSEC — Security Disable Bit

NOSEC is invalid unless the security mask option is specified before the MCU is manufactured. If the security mask option is omitted NOSEC always reads 1. The enhanced security feature is available in the MC68S711E9 MCU. The enhancement to the standard security feature protects the EPROM as well as RAM and EEPROM.

0 = Security enabled
1 = Security disabled

NOCOP — COP System Disable Bit

Refer to Chapter 5 Resets and Interrupts.

1 = COP disabled
0 = COP enabled

ROMON — ROM/EPROM/OTPROM Enable Bit

When this bit is 0, the ROM or EPROM is disabled and that memory space becomes externally addressed. In single-chip mode, ROMON is forced to 1 to enable ROM/EPROM regardless of the state of the ROMON bit.

0 = ROM disabled from the memory map
1 = ROM present in the memory map

EEON — EEPROM Enable Bit

When this bit is 0, the EEPROM is disabled and that memory space becomes externally addressed.

0 = EEPROM removed from the memory map
1 = EEPROM present in the memory map

2.3.3.2 RAM and I/O Mapping Register

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

Address: $103D

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

Figure 2-12. RAM and I/O Mapping Register (INIT)

RAM[3:0] — RAM Map Position Bits

These four bits, which specify the upper hexadecimal digit of the RAM address, control position of RAM in the memory map. RAM can be positioned at the beginning of any 4-Kbyte page in the memory map. It is initialized to address $0000 out of reset. Refer to Table 2-4.

REG[3:0] — 64-Byte Register Block Position

These four bits specify the upper hexadecimal digit of the address for the 64-byte block of internal registers. The register block, positioned at the beginning of any 4-Kbyte page in the memory map, is initialized to address $1000 out of reset. Refer to Table 2-5.

2.4.3 EPROM and EEPROM Programming Control Register

The EPROM and EEPROM programming control register (PPROG) enables the EPROM programming voltage and controls the latching of data to be programmed.

- For MC68HC711E9, PPROG is also the EEPROM programming control register.
- For the MC68HC711E20, EPROM programming is controlled by the EPROG register and EEPROM programming is controlled by the PPROG register.

Address: $103B

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	ODD	EVEN	ELAT(1)	BYTE	ROW	ERASE	EELAT	EPGM
Reset:	0	0	0	0	0	0	0	0

1. MC68HC711E9 only

Figure 2-14. EPROM and EEPROM Programming Control Register (PPROG)

ODD — Program Odd Rows in Half of EEPROM (Test) Bit

Refer to 2.5 EEPROM.

EVEN — Program Even Rows in Half of EEPROM (Test) Bit

Refer to 2.5 EEPROM.

ELAT — EPROM/OTPROM Latch Control Bit

When ELAT = 1, writes to EPROM cause address and data to be latched and the EPROM/OTPROM cannot be read. ELAT can be read any time. ELAT can be written any time except when EPGM = 1; then the write to ELAT is disabled.

0 = EPROM address and data bus configured for normal reads
1 = EPROM address and data bus configured for programming

For the MC68HC711E9:

a. EPGM enables the high voltage necessary for both EEPROM and EPROM/OTPROM programming.

b. ELAT and EELAT are mutually exclusive and cannot both equal 1.

BYTE — Byte/Other EEPROM Erase Mode Bit

Refer to 2.5 EEPROM.

ROW — Row/All EEPROM Erase Mode Bit

Refer to 2.5 EEPROM.

ERASE — Erase Mode Select Bit

Refer to 2.5 EEPROM.

EELAT — EEPROM Latch Control Bit

Refer to 2.5 EEPROM.

EPGM —EPROM/OTPROM/EEPROM Programming Voltage Enable Bit

EPGM can be read any time and can be written only when ELAT = 1 (for EPROM/OTPROM programming) or when EELAT = 1 (for EEPROM programming).

0 = Programming voltage to EPROM/OTPROM/EEPROM array disconnected
1 = Programming voltage to EPROM/OTPROM/EEPROM array connected

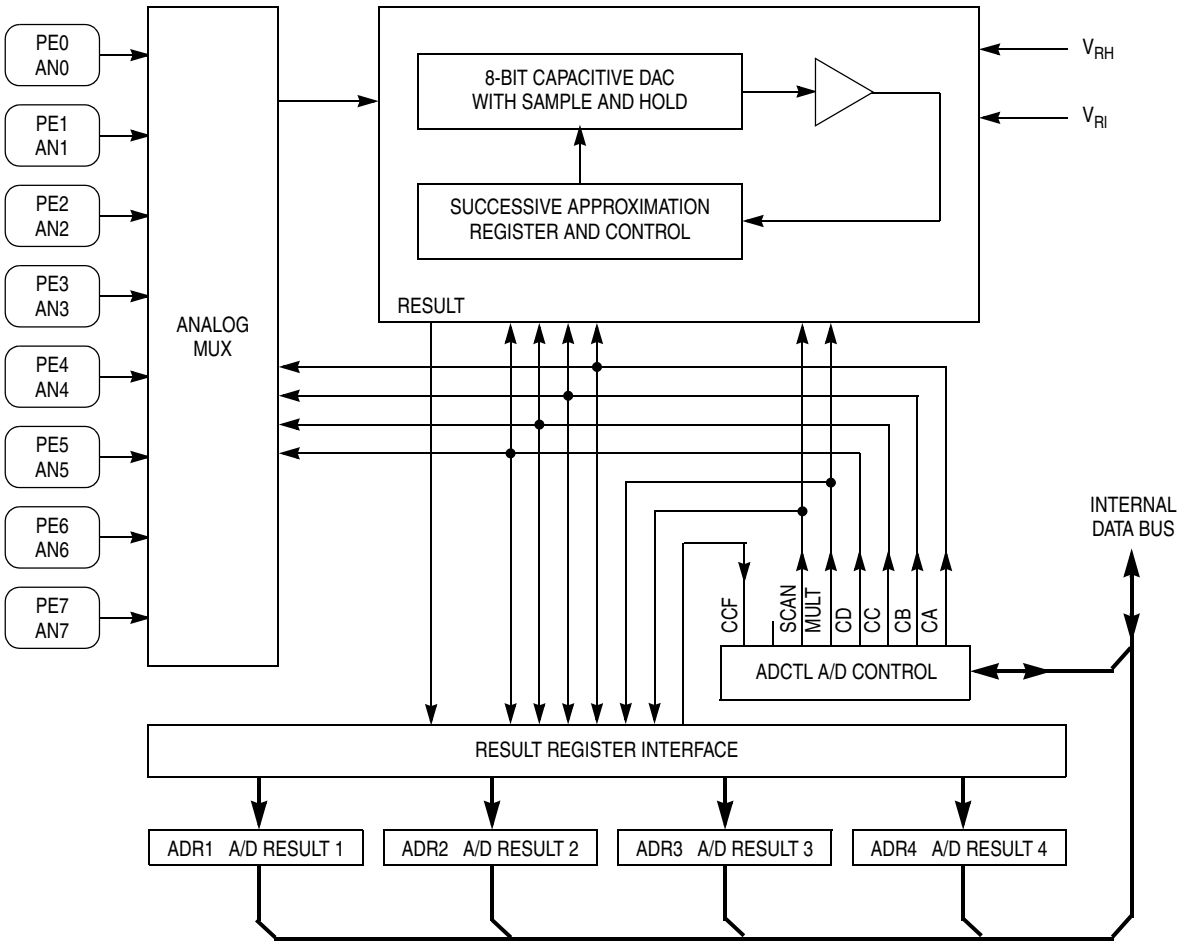
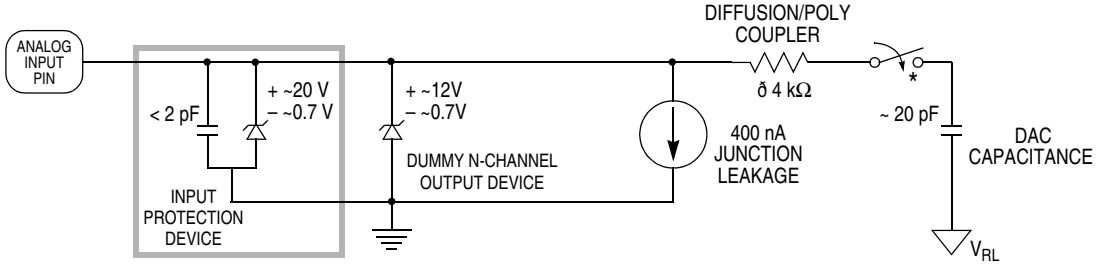


Figure 3-1. A/D Converter Block Diagram



* 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)

3.2.3 Digital Control

All A/D converter operations are controlled by bits in register ADCTL. In addition to selecting the analog input to be converted, ADCTL bits indicate conversion status and control whether single or continuous conversions are performed. Finally, the ADCTL bits determine whether conversions are performed on single or multiple channels.

3.2.4 Result Registers

Four 8-bit registers ADR[4:1] store conversion results. Each of these registers can be accessed by the processor in the CPU. The conversion complete flag (CCF) indicates when valid data is present in the result registers. The result registers are written during a portion of the system clock cycle when reads do not occur, so there is no conflict.

3.2.5 A/D Converter Clocks

The CSEL bit in the OPTION register selects whether the A/D converter uses the system E clock or an internal RC oscillator for synchronization. When E-clock frequency is below 750 kHz, charge leakage in the capacitor array can cause errors, and the internal oscillator should be used. When the RC clock is used, additional errors can occur because the comparator is sensitive to the additional system clock noise.

3.2.6 Conversion Sequence

A/D converter operations are performed in sequences of four conversions each. A conversion sequence can repeat continuously or stop after one iteration. The conversion complete flag (CCF) is set after the fourth conversion in a sequence to show the availability of data in the result registers. Figure 3-3 shows the timing of a typical sequence. Synchronization is referenced to the system E clock.

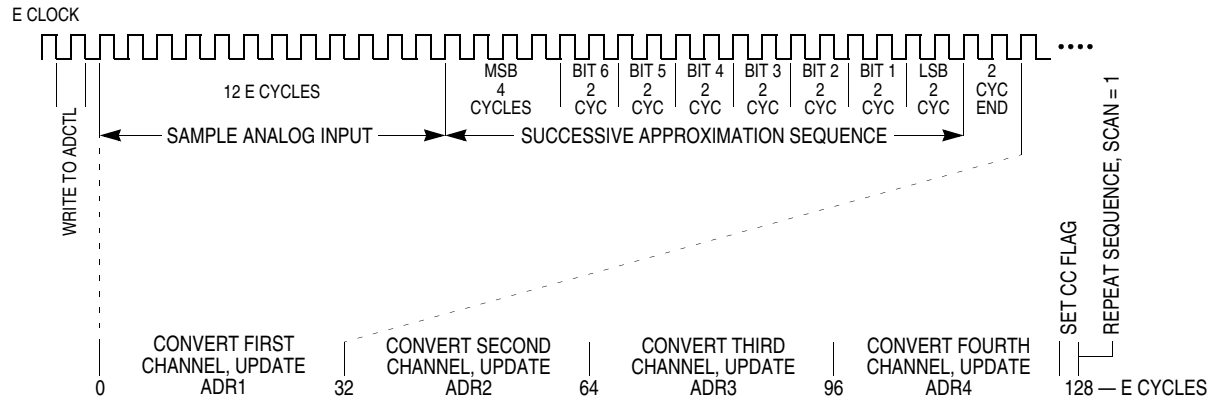


Figure 3-3. A/D Conversion Sequence

5.2.5 System Configuration Options Register

Address:	$1039							
	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	ADPU	CSEL	IRQE(1)	DLY(1)	CME		CR1(1)	CR0(1)
Reset:	0	0	0	1	0	0	0	0

1. Can be written only once in first 64 cycles out of reset in normal mode or at any time in special modes

= Unimplemented

Figure 5-2. System Configuration Options Register (OPTION)

ADPU — Analog-to-Digital Converter Power-Up Bit

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

CSEL — Clock Select Bit

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

IRQE — Configure $\overline{\text{IRQ}}$ for Edge-Sensitive-Only Operation Bit

0 = $\overline{\text{IRQ}}$ is configured for level-sensitive operation.
1 = $\overline{\text{IRQ}}$ is configured for edge-sensitive-only operation.

DLY — Enable Oscillator Startup Delay Bit

Refer to Chapter 2 Operating Modes and On-Chip Memory and Chapter 3 Analog-to-Digital (A/D) Converter.

CME — Clock Monitor Enable Bit

This control bit can be read or written at any time and controls whether or not the internal clock monitor circuit triggers a reset sequence when the system clock is slow or absent. When it is clear, the clock monitor circuit is disabled, and when it is set, the clock monitor circuit is enabled. Reset clears the CME bit.

0 = Clock monitor circuit disabled
1 = Slow or stopped clocks cause reset

Bit 2 — Unimplemented

Always reads 0

CR[1:0] — COP Timer Rate Select Bit

The internal E clock is first divided by 2^{15} before it enters the COP watchdog system. These control bits determine a scaling factor for the watchdog timer. See Table 5-1 for specific timeout settings.

Table 5-4. Interrupt and Reset Vector Assignments

Vector Address	Interrupt Source	CCR Mask Bit	Local Mask
FFC0, C1 – FFD4, D5	Reserved	—	—
FFD6, D7	SCI serial system • SCI receive data register full • SCI receiver overrun • SCI transmit data register empty • SCI transmit complete • SCI idle line detect	I	RIE RIE TIE TCIE ILIE
FFD8, D9	SPI serial transfer complete	I	SPIE
FFDA, DB	Pulse accumulator input edge	I	PAII
FFDC, DD	Pulse accumulator overflow	I	PAOVI
FFDE, DF	Timer overflow	I	TOI
FFE0, E1	Timer input capture 4/output compare 5	I	I4/O5I
FFE2, E3	Timer output compare 4	I	OC4I
FFE4, E5	Timer output compare 3	I	OC3I
FFE6, E7	Timer output compare 2	I	OC2I
FFE8, E9	Timer output compare 1	I	OC1I
FFEA, EB	Timer input capture 3	I	IC3I
FFEC, ED	Timer input capture 2	I	IC2I
FFEE, EF	Timer input capture 1	I	IC1I
FFF0, F1	Real-time interrupt	I	RTII
FFF2, F3	$\overline{\text{IRQ}}$ (external pin)	I	None
FFF4, F5	$\overline{\text{XIRQ}}$ pin	X	None
FFF6, F7	Software interrupt	None	None
FFF8, F9	Illegal opcode trap	None	None
FFFA, FB	COP failure	None	NOCOP
FFFC, FD	Clock monitor fail	None	CME
FFFE, FF	$\overline{\text{RESET}}$	None	None

5.5.1 Interrupt Recognition and Register Stacking

An interrupt can be recognized at any time after it is enabled by its local mask, if any, and by the global mask bit in the CCR. Once an interrupt source is recognized, the CPU responds at the completion of the instruction being executed. Interrupt latency varies according to the number of cycles required to complete the current instruction. When the CPU begins to service an interrupt, the contents of the CPU registers are pushed onto the stack in the order shown in Table 5-5. After the CCR value is stacked, the I bit and the X bit, if $\overline{\text{XIRQ}}$ is pending, are set to inhibit further interrupts. The interrupt vector for the highest priority pending source is fetched and execution continues at the address specified by the vector. At the

Register name: Timer Output Compare 3 Register (High) Address: $101A

	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 Output Compare 3 Register (Low) Address: $101B

	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 9-10. Timer Output Compare 3 Register Pair (TOC3)

Register name: Timer Output Compare 4 Register (High) Address: $101C

	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 Output Compare 4 Register (Low) Address: $101D

	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 9-11. Timer Output Compare 4 Register Pair (TOC4)

9.4.2 Timer Compare Force Register

The CFORC register allows forced early compares. FOC[1:5] correspond to the five output compares. These bits are set for each output compare that is to be forced. The action taken as a result of a forced compare is the same as if there were a match between the OCx register and the free-running counter, except that the corresponding interrupt status flag bits are not set. The forced channels trigger their programmed pin actions to occur at the next timer count transition after the write to CFORC.

The CFORC bits should not be used on an output compare function that is programmed to toggle its output on a successful compare because a normal compare that occurs immediately before or after the force can result in an undesirable operation.

Address: $100B

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	FOC1	FOC2	FOC3	FOC4	FOC5			
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 9-12. Timer Compare Force Register (CFORC)

9.4.5 Timer Counter Register

The 16-bit read-only TCNT register contains the prescaled value of the 16-bit timer. A full counter read addresses the most significant byte (MSB) first. A read of this address causes the least significant byte (LSB) to be latched into a buffer for the next CPU cycle so that a double-byte read returns the full 16-bit state of the counter at the time of the MSB read cycle.

Register name: Timer Counter Register (High) Address: $100E

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

Register name: Timer Counter Register (Low) Address: $100F

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

= Unimplemented

Figure 9-15. Timer Counter Register (TCNT)

9.4.6 Timer Control Register 1

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

Address: $1020

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	OM2	OL2	OM3	OL3	OM4	OL4	OM5	OL5
Reset:	0	0	0	0	0	0	0	0

Figure 9-16. Timer Control Register 1 (TCTL1)

OM[2:5] — Output Mode Bits

OL[2:5] — Output Level Bits

These control bit pairs are encoded to specify the action taken after a successful OCx compare. OC5 functions only if the I4/O5 bit in the PACTL register is clear. Refer to Table 9-3 for the coding.

Table 9-3. Timer Output Compare Actions

OMx	OLx	Action Taken on Successful Compare
0	0	Timer disconnected from output pin logic
0	1	Toggle OCx output line
1	0	Clear OCx output line to 0
1	1	Set OCx output line to 1

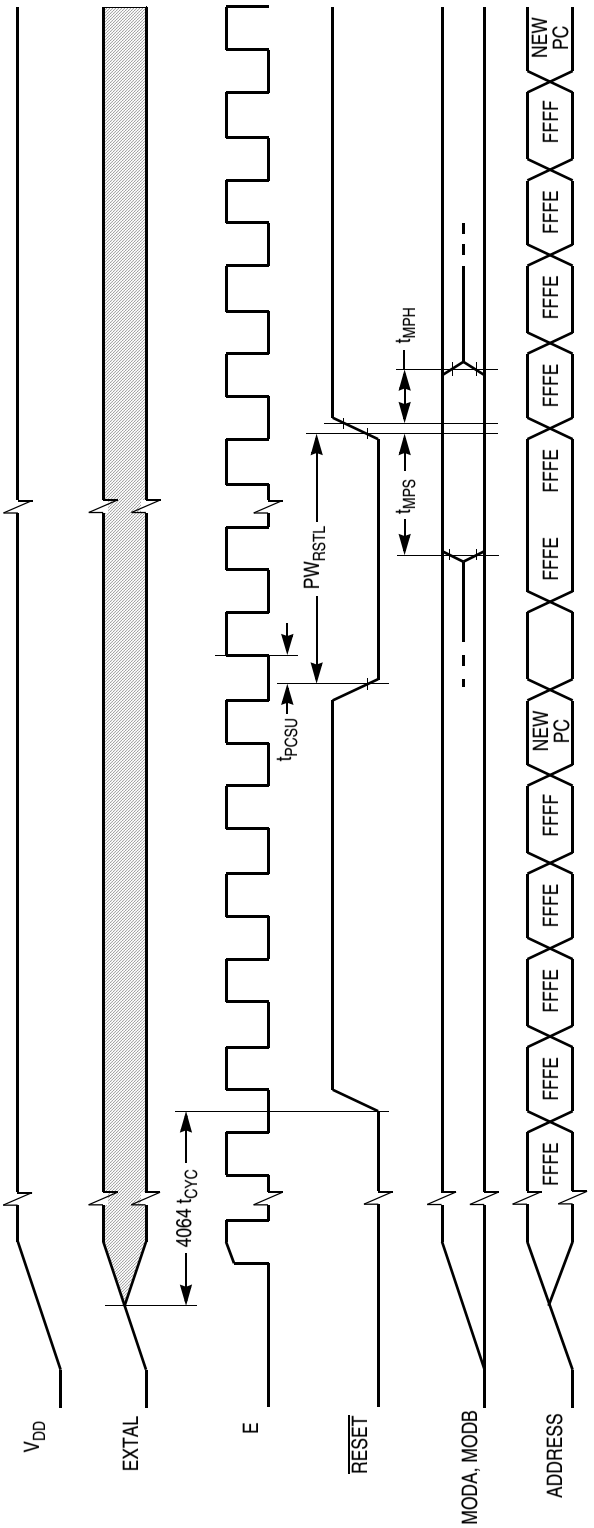


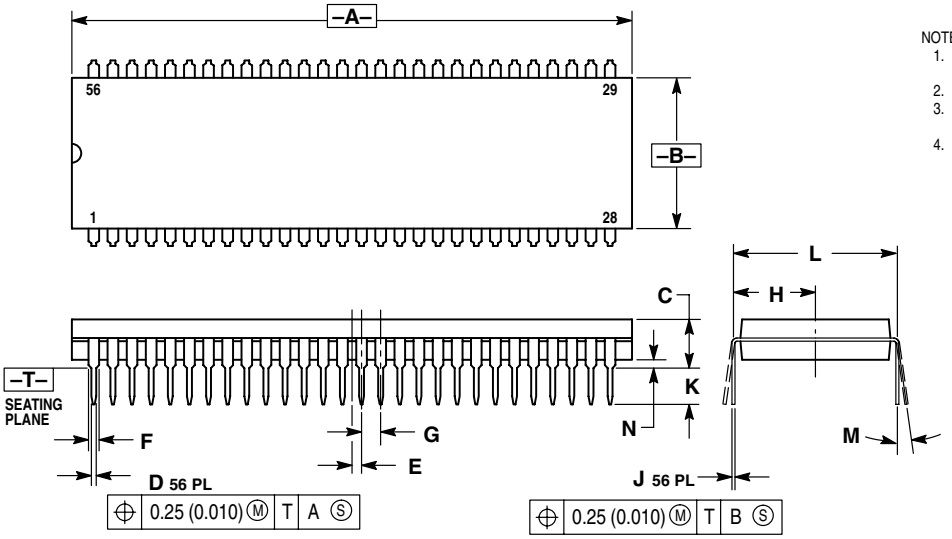
Figure 10-3. POR External Reset Timing Diagram

10.14 MC68L11E9/E20 Analog-to-Digital Converter Characteristics

Characteristic[1]	Parameter[2]	Min	Absolute	Max	Unit
Resolution	Number of bits resolved by A/D converter	—	8	—	Bits
Non-linearity	Maximum deviation from the ideal A/D transfer characteristics	—	—	±1	LSB
Zero error	Difference between the output of an ideal and an actual for 0 input voltage	—	—	±1	LSB
Full scale error	Difference between the output of an ideal and an actual A/D for full-scale input voltage	—	—	±1	LSB
Total unadjusted error	Maximum sum of non-linearity, zero error, and full-scale error	—	—	±1/2	LSB
Quantization error	Uncertainty because of 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	—	—	±2	LSB
Conversion range	Analog input voltage range	V_{RL}	—	V_{RH}	V
V_{RH}	Maximum analog reference voltage	V_{RL}	—	V_{DD} + 0.1	V
V_{RL}	Minimum analog reference voltage	V_{SS} –0.1	—	V_{RH}	V
ΔV_R	Minimum difference between V_{RH} and V_{RL}	3.0	—	—	V
Conversion time	Total time to perform a single analog-to-digital conversion: E clock 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: E clock Internal RC oscillator	 — —	 12 —	 — 12	 t_{CYC} µs
Sample/hold capacitance	Input capacitance during sample PE[7:0]	—	20 typical	—	pF
Input leakage	Input leakage on A/D pins PE[7:0] V_{RL}, V_{RH}	 — —	 — —	 400 1.0	 nA µA

1. V_{DD} = 3.0 Vdc to 5.5 Vdc, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H, 750 kHz ≤ E ≤ 2.0 MHz, unless otherwise noted
2. Source impedances greater than 10 kΩ affect accuracy adversely because of input leakage.

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



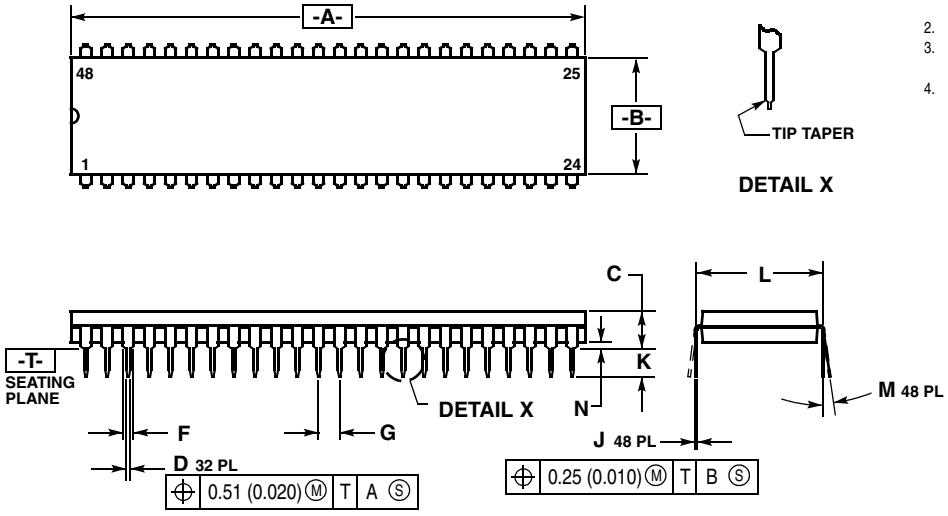
NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: INCH.
3. DIMENSION L TO CENTER OF LEAD WHEN FORMED PARALLEL.
4. DIMENSIONS A AND B DO NOT INCLUDE MOLD FLASH. MAXIMUM MOLD FLASH 0.25 (0.010)

	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

- Extensive on-line MCU information via the CHIPINFO command. View memory map, vectors, register, and pinout information pertaining to the device being emulated
- Host software supports:
  – An editor
  – An assembler and user interface
  – Source-level debug
  – Bus state analysis
  – IBM® mouse

A.5 SPGMR11 — Serial Programmer for M68HC11 MCUs

The SPGMR11 is a modular EPROM/EEPROM programming tool for all M68HC11 devices. The programmer features interchangeable adapters that allow programming of various M68HC11 package types.

Programmer features include:

- Programs M68HC11 Family devices that contain an EPROM or EEPROM array.
- Can be operated as a stand-alone programmer connected to a host computer or connected between a host computer and the M68HC11 modular development system (MMDS11) station module
- Uses plug-in programming adapters to accommodate a variety of MCU devices and packages
- On-board programming voltage circuit eliminates the need for an external 12-volt supply.
- Includes programming software and a user's manual
- Includes a +5-volt power cable and a DB9 to DB25 connector adapter

® IBM is a registered trademark145 of International Business Machines Corporation.

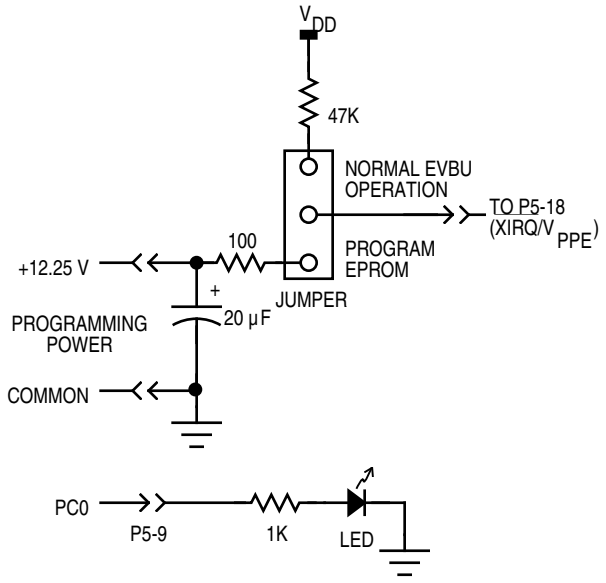


Figure 8. PC-to-MCU Programming Circuit

Lines 50–95 read in the small bootloader from DATA statements at the end of the listing. The source code for this bootloader is presented in the DATA statements. The bootloaded code makes port C bit 0 low, initializes the X and Y registers for use by the EPROM programming utility routine contained in the boot ROM, and then jumps to that routine. The hexadecimal values read in from the DATA statements are converted to binary values by a subroutine. The binary values are then saved as one string (BOOTCODE$).

The next long section of code (lines 97–1250) reads in the S records from an external disk file (in this case, BUF34.S19), converts them to integer, and saves them in an array. The techniques used in this section show how to convert ASCII S records to binary form that can be sent (bootloaded) to an M68HC11.

This S-record translator only looks for the S1 records that contain the actual object code. All other S-record types are ignored.

When an S1 record is found (lines 1000–1024), the next two characters form the hex byte giving the number of hex bytes to follow. This byte is converted to integer by the same subroutine that converted the bootloaded code from the DATA statements. This BYTECOUNT is adjusted by subtracting 3, which accounts for the address and checksum bytes and leaves just the number of object-code bytes in the record.

Starting at line 1100, the 2-byte (4-character) starting address is converted to decimal. This address is the starting address for the object code bytes to follow. An index into the CODE% array is formed by subtracting the base address initialized at the start of the program from the starting address for this S record.

A FOR-NEXT loop starting at line 1130 converts the object code bytes to decimal and saves them in the CODE% array. When all the object code bytes have been converted from the current S record, the program loops back to find the next S1 record.

```
 52 D000           EPRMSTR  EQU    $D000          Start of EPROM
 53 FFFF           EPRMEND  EQU    $FFFF          End of EPROM
 54                *
 55 0000           RAMSTR   EQU    $0000
 56 01FF           RAMEND   EQU    $01FF
 57
 58                * DELAY CONSTANTS
 59                *
 60 0DB0           DELAYS   EQU    3504           Delay at slow baud
 61 021B           DELAYF   EQU    539            Delay at fast baud
 62                *
 63 1068           PROGDEL  EQU    4200           2 ms programming delay
 64                *                       At 2.1 MHz
 65
 66                ************************************************
 67 BF00                     ORG    $BF00
 68                ************************************************
 69
 70                * Next two instructions provide a predictable place
 71                * to call PROGRAM and UPLOAD even if the routines
 72                * change size in future versions.
 73                *
 74 BF00 7EBF13     PROGRAM  JMP    PRGROUT        EPROM programming utility
 75 BF03           UPLOAD   EQU    *              Upload utility
 76
 77                ************************************************
 78                * UPLOAD - Utility subroutine to send data from
 79                * inside the MCU to the host via the SCI interface.
 80                * Prior to calling UPLOAD set baud rate, turn on SCI
 81                * and set Y=first address to upload.
 82                * Bootloader leaves baud set, SCI enabled, and
 83                * Y pointing at EPROM start ($D000) so these default
 84                * values do not have to be changed typically.
 85                * Consecutive locations are sent via SCI in an
 86                * infinite loop. Reset stops the upload process.
 87                ************************************************
 88 BF03 CE1000             LDX    #$1000         Point to internal registers
 89 BF06 18A600   UPLOOP    LDAA   0,Y            Read byte
 90 BF09 1F2E80FC           BRCLR  SCSR,X $80 *   Wait for TDRE
 91 BF0D A72F               STAA   SCDAT,X        Send it
 92 BF0F 1808               INY
 93 BF11 20F3               BRA    UPLOOP         Next...
 94
 95                ************************************************
 96                * PROGRAM - Utility subroutine to program EPROM.
 97                * Prior to calling PROGRAM set baud rate, turn on SCI
 98                * set X=2ms prog delay constant, and set Y=first
 99                * address to program. SP must point to RAM.
100                * Bootloader leaves baud set, SCI enabled, X=4200
101                * and Y pointing at EPROM start ($D000) so these
102                * default values don't have to be changed typically.
103                * Delay constant in X should be equivalent to 2 ms
104                *  at 2.1 MHz X=4200; at 1 MHz X=2000.
105                * An external voltage source is required for EPROM
106                * programming.
```

How to Reach Us:

Home Page:
www.freescale.com

E-mail:
support@freescale.com

USA/Europe or Locations Not Listed:
Freescale Semiconductor
Technical Information Center, CH370
1300 N. Alma School Road
Chandler, Arizona 85224
+1-800-521-6274 or +1-480-768-2130
support@freescale.com

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
support@freescale.com

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064
Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor Hong Kong Ltd.
Technical Information Center
2 Dai King Street
Tai Po Industrial Estate
Tai Po, N.T., Hong Kong
+800 2666 8080
support.asia@freescale.com

For Literature Requests Only:
Freescale Semiconductor Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-441-2447 or 303-675-2140
Fax: 303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

M68HC11E
Rev. 5.1, 07/2005

freescale™
semiconductor