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	4MHz
Connectivity	SCI, SPI
Peripherals	POR, PWM, WDT
Number of I/O	62
Program Memory Size	24KB (24K x 8)
Program Memory Type	OTP
EEPROM Size	640 x 8
RAM Size	768 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	84-LCC (J-Lead)
Supplier Device Package	84-PLCC (29.29x29.29)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc711k4cfne4

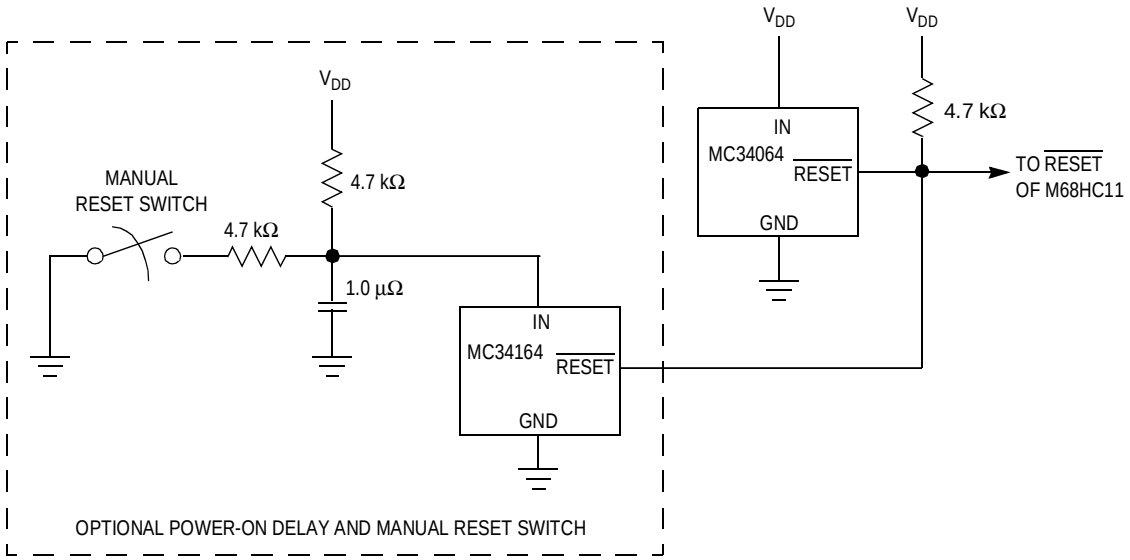


Figure 2-5. External Reset Circuit

It is important to protect the MCU against corruption of RAM and EEPROM during power transitions. This can be done with a low-voltage interrupt (LVI) circuit which holds the $\overline{\text{RESET}}$ pin low when V_{DD} drops below the minimum operating level. **Figure 2-5** shows a suggested reset circuit that incorporates two LVI devices and an external switch.

2.5 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.

When an external CMOS-compatible clock input is connected to the EXTAL pin, the XTAL pin must be left unterminated.

CAUTION: *In all cases, use caution around the oscillator pins.*

Load capacitances shown in **Figure 2-6** are specified by the crystal manufacturer and should include all stray layout capacitances.

Table 2-2. Port Signal Summary

Port/Bit	Single-Chip and Bootstrap Modes	Expanded and Special Test Modes
PA0	PA0/IC3	
PA1	PA1/IC2	
PA2	PA2/IC1	
PA3	PA3/OC5/IC4/and-or OC1	
PA4	PA4/OC4/and-or OC1	
PA5	PA5/OC3/and-or OC1	
PA6	PA6/OC2/and-or OC1	
PA7	PA7/PAI/and-or OC1	
PB[7:0]	PB[7:0]	ADDR[15:8]
PC[7:0]	PC[7:0]	DATA[7:0]
PD0	PD0/RxD	
PD1	PD1/TxD	
PD2	PD2/MISO	
PD3	PD3/MOSI	
PD4	PD4/SCK	
PD5	PD5/$\overline{SS}$	
PE[7:0]	PE[7:0]/AN[7:0]	
PF[7:0]	PF[7:0]	ADDR[7:0]
PG0	PG0	PG0/XA13
PG1	PG1	PG1/XA14
PG2	PG2	PG2/XA15
PG3	PG3	PG3/XA16
PG4	PG4	PG4/XA17
PG5	PG5	PG5/XA18
PG6	PG6	PG6
PG7	PG7	PG7/R/$\overline{W}$
PH0	PH0/PW1	
PH1	PH1/PW2	
PH2	PH2/PW3	
PH3	PH3/PW4	
PH4	PH4	PH4/CSIO
PH5	PH5	PH5/CSGP1
PH6	PH6	PH6/CSGP2
PH7	PH7	PH7/CSPROG

there are no special requirements for alignment of instructions or operands.

3.5 Opcodes and Operands

The M68HC11 Family of microcontrollers uses 8-bit opcodes. Every instruction requires a unique opcode for each of its addressing modes. The resulting number of opcodes exceeds the 256 available in an 8-bit binary number. A 4-page opcode map has been implemented to accommodate the extra instructions. An additional byte, called a prebyte, directs the processor from page 0 of the opcode map to one of the other three pages. As its name implies, the additional byte precedes the opcode.

A complete instruction consists of a prebyte, if any, an opcode, and zero to three operands. The operands contain information the CPU needs for executing the instruction. Complete instructions can be from one to five bytes long.

3.6 Addressing Modes

Six addressing modes can be used to access memory:

1. Immediate
2. Direct
3. Extended
4. Indexed
5. Inherent
6. Relative

All modes except inherent mode use an effective address. The effective address is the memory address where the argument is fetched or stored or the address from which execution is to proceed. The effective address can be specified within an instruction or it can be calculated.

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$0028	Serial Peripheral Control Register (SPCR) See page 174.	Read: Write:	SPIE	SPE	DWOM	MSTR	CPOL	CPHA	SPR1	SPR0
		Reset:	0	0	0	0	0	1	U	U
$0029	Serial Peripheral Status Register (SPSR) See page 176.	Read: Write:	SPIF	WCOL	0	MODF	0	0	0	0
		Reset:	0	0	0	0	0	0	0	0
$002A	Serial Peripheral Data Register (SPDR) See page 177.	Read: Write:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Reset:	Undefined after reset							
$002B	EPROM Programming Control Register (EPROG)(1) See page 91.	Read: Write:	R	0	ELAT	EXCOL	EXROW	0	0	EPGM
		Reset:	0	0	0	0	0	0	0	0

1. Present only in EPROM (711) devices

$002C	Port Pullup Assignment Register (PPAR) See page 147.	Read: Write:	0	0	0	0	HPPUE	GPPUE	FPPUE	BPPUE
		Reset:	0	0	0	0	1	1	1	1
$002D	Port G Assignment Register (PGAR) See page 235.	Read: Write:	0	0	PGAR5	PGAR4	PGAR3	PGAR2	PGAR1	PGAR0
		Reset:	0	0	0	0	0	0	0	0
$002E	System Configuration Options 3 Register (OPT3)(2) See page 132.	Read: Write:		SM						
		Reset:	0	0	0	0	0	0	0	0

2. Not available on M68HC11K4 devices

$002F	Reserved		R	R	R	R	R	R	R	R
$0030	Analog-to-Digital Control/Status Register (ADCTL) See page 227.	Read: Write:	CCF	0	SCAN	MULT	CD	CC	CB	CA
		Reset:	0	0	U	U	U	U	U	U
$0031	Analog-to-Digital Results Register 1 (ADR1) See page 229.	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	Undefined after reset							

= Unimplemented R = Reserved U = Undefined

Figure 4-1. Register and Control Bit Assignments (Sheet 5 of 11)

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$0062	Pulse Width Modulation Timer Prescaler Register (PWSCAL) See page 215.	Read: Write:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Reset:	0	0	0	0	0	0	0	0
$0063	Pulse Width Modulation Timer Enable Register (PWEN) See page 216.	Read: Write:	TPWSL	DISCP	0	0	PWEN4	PWEN3	PWEN2	PWEN1
		Reset:	0	0	0	0	0	0	0	0
$0064	Pulse Width Modulation Timer Counter 1 Register (PWCNT1) See page 217.	Read: Write:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Reset:	0	0	0	0	0	0	0	0
$0065	Pulse Width Modulation Timer Counter 2 Register (PWCNT2) See page 217.	Read: Write:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Reset:	0	0	0	0	0	0	0	0
$0066	Pulse Width Modulation Timer Counter 3 Register (PWCNT3) See page 217.	Read: Write:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Reset:	0	0	0	0	0	0	0	0
$0067	Pulse Width Modulation Timer Counter 4 Register (PWCNT4) See page 217.	Read: Write:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Reset:	0	0	0	0	0	0	0	0
$0068	Pulse Width Modulation Timer Period 1 Register (PWPER1) See page 218.	Read: Write:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Reset:	0	0	0	0	0	0	0	0
$0069	Pulse Width Modulation Timer Period 2 Register (PWPER2) See page 218.	Read: Write:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Reset:	0	0	0	0	0	0	0	0
$006A	Pulse Width Modulation Timer Period 3 Register (PWPER3) See page 218.	Read: Write:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Reset:	0	0	0	0	0	0	0	0
$006B	Pulse Width Modulation Timer Period 4 Register (PWPER4) See page 218.	Read: Write:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Reset:	0	0	0	0	0	0	0	0

= Unimplemented R = Reserved U = Undefined

Figure 4-1. Register and Control Bit Assignments (Sheet 9 of 11)

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$006C	Pulse Width Modulation Timer Duty Cycle 1 Register (PWDTY1) See page 219.	Read: Write:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Reset:	0	0	0	0	0	0	0	0
$006D	Pulse Width Modulation Timer Duty Cycle 2 Register (PWDTY2) See page 219.	Read: Write:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Reset:	0	0	0	0	0	0	0	0
$006E	Pulse Width Modulation Timer Duty Cycle 3 Register (PWDTY3) See page 219.	Read: Write:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Reset:	0	0	0	0	0	0	0	0
$006F	Pulse Width Modulation Timer Duty Cycle 4 Register (PWDTY4) See page 219.	Read: Write:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Reset:	0	0	0	0	0	0	0	0
$0070	SCI Baud Rate Control Register High (SCBDH) See page 158.	Read: Write:	BTST	BSPL	0	SBR12	SBR11	SBR10	SBR9	SBR8
		Reset:	0	0	0	0	0	0	0	0
$0071	SCI Baud Rate Control Register Low (SCBDL) See page 158.	Read: Write:	SBR7	SBR6	SBR5	SBR4	SBR3	SBR2	SBR1	SBR0
		Reset:	0	0	0	0	0	1	0	0
$0072	SCI Control Register 1 (SCCR1) See page 160.	Read: Write:	LOOPS	WOMS	0	M	WAKE	ILT	PE	PT
		Reset:	U	U	0	0	0	0	0	0
$0073	SCI Control Register 2 (SCCR2) See page 161.	Read: Write:	TIE	TCIE	RIE	ILIE	TE	RE	RWU	SBK
		Reset:	0	0	0	0	0	0	0	0
$0074	SCI Status Register 1 (SCSR1) See page 162.	Read: Write:	TDRE	TC	RDRF	IDLE	OR	NF	FE	PF
		Reset:	0	0	0	0	0	0	0	0
$0075	SCI Status Register 2 (SCSR2) See page 164.	Read:	0	0	0	0	0	0	0	RAF
		Write:								
		Reset:	1	1	0	0	0	0	0	0
$0076	SCI Data Register (SCDR) See page 165.	Read: Write:	R8	T8	0	0	0	0	0	0
		Reset:	Undefined after reset							

= Unimplemented R = Reserved U = Undefined

Figure 4-1. Register and Control Bit Assignments (Sheet 10 of 11)

4.6.4 Bootloader ROM

The bootloader program occupies 512 bytes of bootstrap ROM at addresses $BE00–$BFFF. It is active only in special modes when the RBOOT bit in the HPRIO register is set.

4.7 EPROM/OTPROM (M68HC711K4 and M68HC711KS2)

The M68HC711K4 devices include 24 Kbytes of on-chip EPROM (OTPROM in non-windowed packages). The M68HC711KS2 has 32 Kbytes of EPROM.

The two methods available to program the EPROM are:

- Downloading data through the serial communication interface (SCI) in bootstrap or special test mode
- Programming individual bytes from memory

Before proceeding with programming:

- Ensure that the CONFIG register ROMON bit is set.
- Ensure that the $\overline{\text{IRQ}}$ pin is pulled to a high level.
- Apply 12 volts to the $\overline{\text{XIRQ}}/V_{PP}$ pin.

Program the EPROM only at room temperature. Place an opaque label over the quartz window on windowed parts after programming.

4.7.1 Programming the EPROM with Downloaded Data

The MCU can download EPROM data through the SCI while in the special test or bootstrap modes. This can be done either with custom software, also downloaded through the SCI, or with a built-in utility program in bootstrap ROM. In either case, the 12-volt nominal programming voltage must be present on the $\overline{\text{XIRQ}}/V_{PP}$ pin.

To use the bootstrap ROM utility, download a 3-byte program consisting of a single jump instruction to $BF00, the starting address of the resident EPROM programming utility. The utility program sets the X and Y index

This procedure programs one byte into EPROM. On entry, accumulator A contains the byte of data to be programmed and X contains the target EPROM address.

```
EPROG LDAB  #$20
      STAB  $002B  Set ELAT bit to enable EPROM latches.
                   (EPGM must be 0.)
      STAA  $0,X   Store data to EPROM address
      LDAB  #$21
      STAB  $002B  Set EPGM bit with ELAT=1
                   to enable EPROM programming voltage
      JSR   DLYEP  Delay 1-2 ms
      CLR   $002B  Turn off programming voltage and set to
                   READ mode
```

4.8 EEPROM and the CONFIG Register

The 640-byte on-board EEPROM is enabled by the EEON bit in the CONFIG register and located on a 4-K boundary determined by the INIT2 register (**4.6.3 EEPROM**). An internal charge pump supplies the programming voltage for the EEPROM, eliminating the need for an external high-voltage supply.

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 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

execution of the illegal opcode, which can lead to stack overflow, the service routine should reinitialize the stack pointer.

5.5.1.3 Software Interrupt (SWI)

SWI cannot be masked by virtue of the fact that it is a software instruction. It is not inhibited by the global mask bits in the CCR. Execution of SWI sets the I mask bit, so other interrupts are inhibited until user software clears the I bit or SWI terminates with an RTI instruction.

5.5.2 Maskable Interrupts

All maskable interrupts are generated by on-chip peripherals, with the exception of the $\overline{\text{IRQ}}$ pin. This input can be connected through a wired-OR network to external devices. When one of these devices pulls $\overline{\text{IRQ}}$ low, a software accessible interrupt flag is set. When enabled, this flag causes a constant request for interrupt service. After the flag is cleared, the service request is released. $\overline{\text{IRQ}}$ is low-level sensitive by default, but can be set for falling-edge sensitivity by the IRQE bit in the OPTION register (see **Figure 5-6**).

Address: $0039

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	ADPU	CSEL	IRQE(1)	DLY(1)	CME	FCME(1)	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 modes or at any time in special modes

Figure 5-6. System Configuration Options Register (OPTION)

IRQE — Configure IRQ for Edge-Sensitive Operation Bit

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

0 = Low-level recognition
1 = Falling-edge recognition

5.6 Reset and Interrupt Priority

A hardware priority scheme determines which reset or interrupt is serviced first when simultaneous requests occur.

The six highest-priority interrupt sources are not maskable. The priority arrangement for these sources is:

1. POR or $\overline{\text{RESET}}$ pin
2. Clock monitor reset
3. COP watchdog reset
4. $\overline{\text{XIRQ}}$ interrupt
5. Illegal opcode interrupt
6. Software interrupt (SWI)

The maskable interrupt sources have this priority arrangement:

1. IRQ
2. Real-time interrupt
3. Timer input capture 1
4. Timer input capture 2
5. Timer input capture 3
6. Timer output compare 1
7. Timer output compare 2
8. Timer output compare 3
9. Timer output compare 4
10. Timer input capture 4/output compare 5
11. Timer overflow
12. Pulse accumulator overflow
13. Pulse accumulator input edge
14. SPI transfer complete
15. SCI system

6. Steps 2-5 are repeated until the entire message is sent.
7. The line returns to idle status.

7.4 Transmit Operation

Transmission starts by writing a data character to the 2-byte SCI data register (SCDRH and SCDRL). The MCU parallel-loads the character into a serial shift register which shifts the data out on the transmission pin. This double-buffered operation allows transmission of the current character while the MCU loads the next one. The output of the serial shift register drives the TxD pin as long as the transmit enable (TE) bit of serial communication control register 2 (SCCR2) is set.

Two flags in serial communication status register 1 (SCSR1) alert the MCU of transmission status. The TDRE (transmit data register empty) flag is set when the SCDR loads its contents into the shift register; this flag can generate an interrupt if the TIE (transmit interrupt enable) bit in SCCR2 is set. The TC (transmit complete) flag is set when transmission is complete (line idle); this can also generate an interrupt if the TCIE (transmit complete interrupt enable) bit in SCSR1 is set. The TDRE and TC flags are normally set when software sets the TE bit to enable the transmitter. See **Figure 5-10. Interrupt Priority Resolution Within SCI System** for a flow diagram of SCI interrupts.

If interrupts are not enabled, the status flags can be read by software (polled) to determine when the corresponding conditions exist. Status flags are set automatically by hardware logic conditions, but must be cleared by software. The software clearing sequence for these flags is automatic. Functions that are normally performed in response to the status flags also satisfy the conditions of the clearing sequence.

When software clears the TE bit, the TxD pin reverts to its general-purpose I/O function (PD1). The transmitter completes transmission of a character in progress before actually shutting down; other characters waiting in the transmit queue are lost. The TC and TDRE flags are set at the completion of this last character, even though TE has been disabled. Only an MCU reset can abort transmission in midcharacter.

9.4.1 Timer Counter Register

	Bit 7	6	5	4	3	2	1	Bit 0
Address:	$000E — TCNT (High)							
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
Address:	$000F — TCNT (Low)							
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 9-3. Timer Counter Register (TCNT)

TCNT reflects the current value in the free-running counter. Input capture functions use this number to mark the time of an external event, and output compare functions use it to determine the time at which to generate an event.

In normal modes, TCNT is a read-only register. Writes to TCNT in normal modes have no effect. TCNT can be read and written in special modes.

9.5.1 Timer Input Capture Registers

	Bit 7	6	5	4	3	2	1	Bit 0
Address:	$0010 — TIC1 (High)							
Read: Write:	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
Reset:	Unaffected by reset							
Address:	$0011 — TIC1 (Low)							
Read: Write:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Reset:	Unaffected by reset							
Address:	$0012 — TIC2 (High)							
Read: Write:	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
Reset:	Unaffected by reset							
Address:	$0013 — TIC2 (Low)							
Read: Write:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Reset:	Unaffected by reset							
Address:	$0014— TIC3 (High)							
Read: Write:	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
Reset:	Unaffected by reset							
Address:	$0015 — TIC3 (Low)							
Read: Write:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Reset:	Unaffected by reset							

Figure 9-8. Timer Input Capture Registers (TIC1–TIC3)

9.6.4 Timer Interrupt Mask 1 Register

Address: $0022

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	OC1I	OC2I	OC3I	OC4I	I4/O5I	IC1I	IC2I	IC3I
Reset:	0	0	0	0	0	0	0	0

Figure 9-16. Timer Interrupt Mask 1 Register (TMSK1)

Bits in TMSK1 correspond bit for bit with flag bits in TFLG1.

OC1I–OC4I — Output Compare x Interrupt Enable Bits

If the OCxI enable bit is set when the OCxF flag bit is set, a hardware interrupt sequence is requested.

I4/O5I — Input Capture 4 or Output Compare 5 Interrupt Enable Bit

If I4/O5I is set when OC5 is enabled and the I4/O5F flag bit is set, a hardware interrupt sequence is requested.

9.6.5 Timer Control 1 Register

Address: $0020

	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 9-17. Timer Control Register 1 (TCTL1)

OM[2:5] and OL[2:5] — Output Mode and Output Level Bits

Use these bit pairs as indicated in **Table 9-4** to specify the action taken after a successful OCx compare.

The MULT bit in ADCTL determines whether one or four channels are converted. When MULT = 0, one channel is converted. The selected channel is sampled and the conversion results are written to ADR1; the same channel is sampled again, and the conversion results are stored in ADR2, and so on. In continuous mode, the same channel continues to be sampled and converted. Setting the MULT bit converts four different channels; each result register contains the conversion result of a different channel. In continuous mode, the same four channels are converted in each sequence.

ADCTL bits [3:0] select the particular channel(s) to be converted. See **Table 10-1**.

Table 10-1. A/D Converter Channel Selection

Channel Select Control Bits CD:CC:CB:CA	Channel Signal	Result Location When MULT = 1
0000	AN0	ADR1
0001	AN1	ADR2
0010	AN2	ADR3
0011	AN3	ADR4
0100	AN4	ADR1
0101	AN5	ADR2
0110	AN6	ADR3
0111	AN7	ADR4
10XX	Reserved	—
1100	V_{RH}(1)	ADR1
1101	V_{RL}(1)	ADR2
1110	(V_{RH})/2(1)	ADR3
1111	Reserved(1)	ADR4

1. Used for factory testing

11.3 Memory Expansion

The M68HC11K Family devices employ a register-based paging scheme to extend their address range beyond the physical 64-Kbyte limit of the 16 CPU address lines. Pages are selected using the expansion address lines XA[18:13] available on port G. This selection can be facilitated by the chip-select lines on port H, discussed in **11.4 Chip Selects**. The M68HC11KS devices do not provide these features since they lack the required port G and port H lines. Refer to **Figure 1-2. M68HC11KS Family Block Diagram**.

11.3.1 Memory Size and Address Line Allocation

To access expanded memory, the user first allocates portion(s) of the 64 Kbyte address space, or window(s), through which the CPU will view external memory. One or two windows can be designated, and the size of each window can be 0 (disabled), 8, 16, or 32 Kbytes.

Expanded memory is addressed with a combination of the CPU's normal address lines ADDR[15:0] and the expansion address lines XA[18:13]. The expansion address lines select a memory bank, and the CPU's normal address lines select a particular location within the bank. The size of the window(s) and the number of memory banks determine exactly which expansion address lines are used. The port G assignment register (PGAR) controls which port G pins function as expanded address lines. Any port G pins not allocated for memory expansion can serve as general-purpose input/output (GPIO). When a configuration uses any of the lower three expansion address lines XA[15:13] they replace the CPU's equivalent address lines (ADDR[15:13]). **Table 11-1** shows how address and expansion lines are allocated for various combinations of memory banks and window size.

Table 11-1. CPU Address and Address Expansion Signals

Number of Banks	Window Size			
	8 Kbytes	16 Kbytes	32 Kbytes	32 Kbytes (Window Based at $4000)
2	ADDR[12:0] XA13	ADDR[13:0] XA14	ADDR[14:0] XA15	ADDR[13:0] XA[15:14]
4	ADDR[12:0] XA[14:13]	ADDR[13:0] XA[15:14]	ADDR[14:0] XA[16:15]	ADDR[13:0] XA[16:14]
8	ADDR[12:0] XA[15:13]	ADDR[13:0] XA[16:14]	ADDR[14:0] XA[17:15]	ADDR[13:0] XA[17:14]
16	ADDR[12:0] XA[16:13]	ADDR[13:0] XA[17:14]	ADDR[14:0] XA[18:15]	ADDR[13:0] XA[18:14]
32	ADDR[12:0] XA[17:13]	ADDR[13:0] XA[18:14]	— —	— —
64	ADDR[12:0] XA[18:13]	— —	— —	— —

The base address for each window must be an integer multiple of the window size, with one exception. When the window size is 32 Kbytes, the base address can be at $4000 as well as the 32-Kbyte multiples $0000 and $8000.

This special case requires a modification in address line deployment. Normally, when the bank size is 32 Kbytes and the bank address is $0000 or $8000, CPU address lines ADDR[14:0] select individual bytes within the 32-Kbyte space and the ADDR[14:0] pins are connected to address lines A[14:0] of the memory device. When the base address is $4000, the CPU address signal ADDR14 must be inverted to allow 32 Kbytes of contiguous memory. To do this, the CPU drives the inverted ADDR14 signal onto the XA14 pin when the window is active, and the non-inverted CPU ADDR14 signal onto the XA14 pin when the window is not active. Therefore, address 14 of the memory device must be connected to expansion line XA14 rather than normal address line ADDR14.

If the two memory windows overlap, window 1 has priority, and only the portion of window 2 that does not overlap window 1 remains active. If a

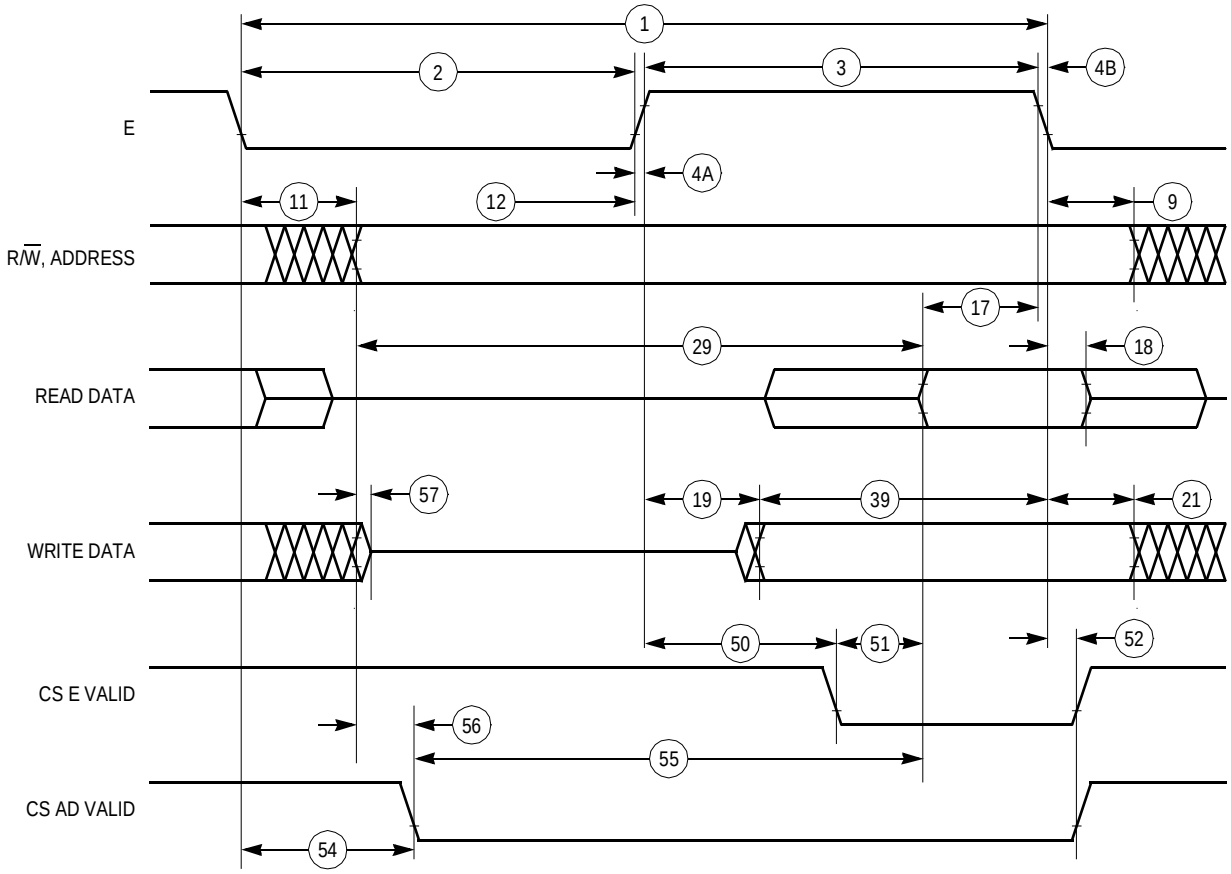
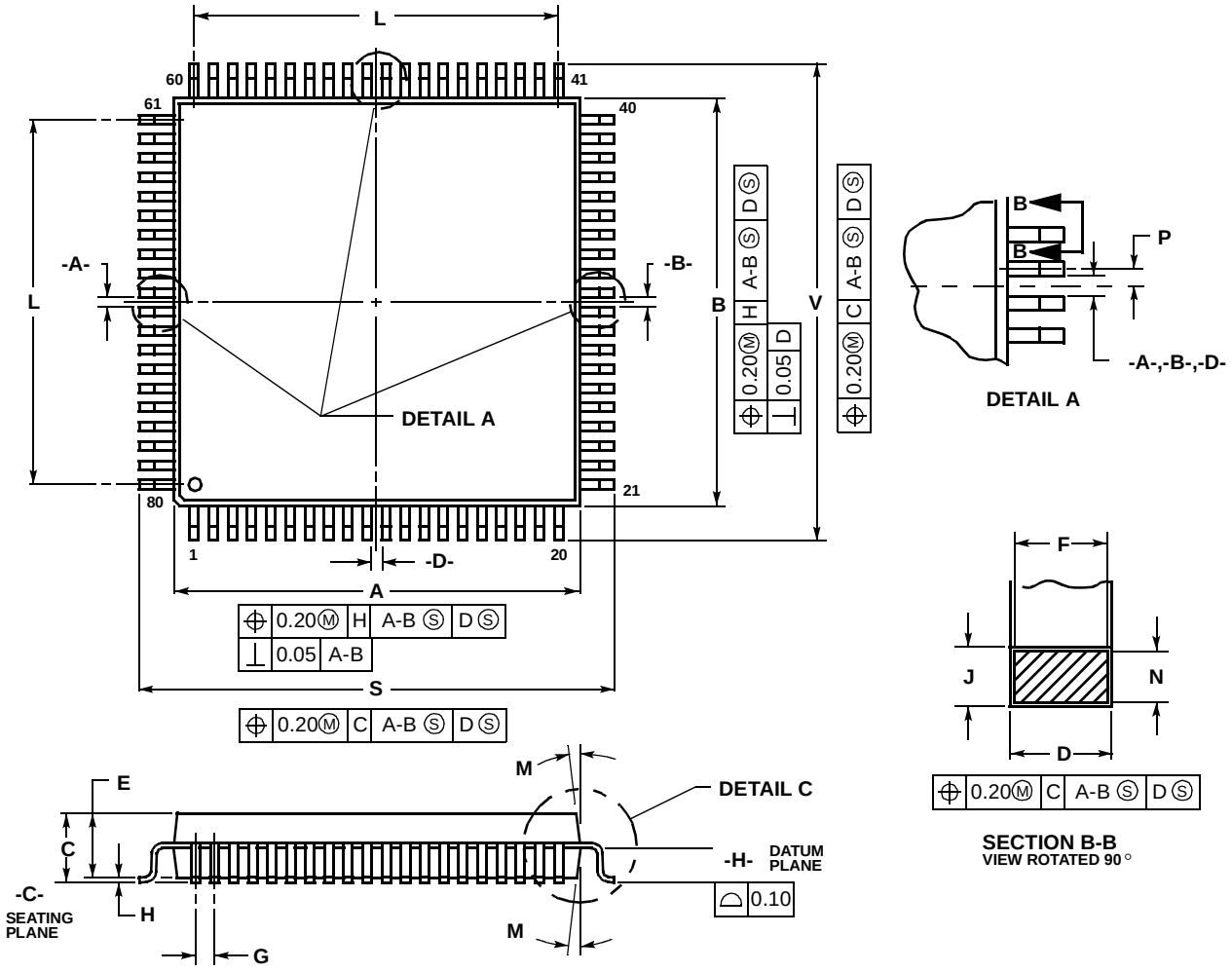
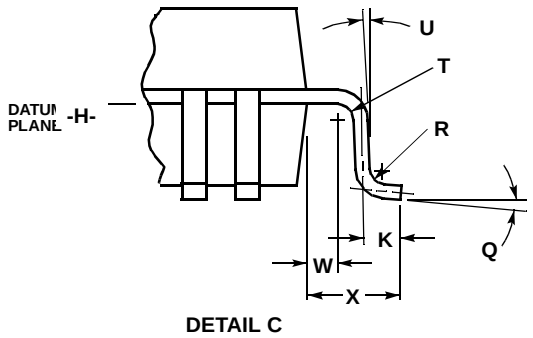


Figure 12-9. Expansion Bus Timing

13.5 80-Pin Quad Flat Pack (Case 841B)





NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: MILLIMETER.
3. DATUM PLANE -H- IS LOCATED AT BOTTOM OF LEAD AND IS COINCIDENT WITH THE LEAD WHERE THE LEAD EXITS THE PLASTIC BODY AT THE BOTTOM OF THE PARTING LINE.
4. DATUMS -A-, -B- AND -D- TO BE DETERMINED AT DATUM PLANE -H-.
5. DIMENSIONS S AND V TO BE DETERMINED AT SEATING PLANE -C-.
6. DIMENSIONS A AND B DO NOT INCLUDE MOLD PROTRUSION. ALLOWABLE PROTRUSION IS 0.25 PER SIDE. DIMENSIONS A AND B DO INCLUDE MOLD MISMATCH AND ARE DETERMINED AT DATUM PLANE -H-.
7. DIMENSION D DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL BE 0.08 TOTAL IN EXCESS OF THE D DIMENSION AT MAXIMUM MATERIAL CONDITION. DAMBAR CANNOT BE LOCATED ON THE LOWER RADIUS OR THE FOOT.

	MILLIMETERS	
DIM	MIN	MAX
A	13.90	14.10
B	13.90	14.10
C	2.15	2.45
D	0.22	0.38
E	2.00	2.40
F	0.22	0.33
G	0.65 BSC	
H	---	0.25
J	0.13	0.23
K	0.65	0.95
L	12.35 REF	
M	5°	10°
N	0.13	0.17
P	0.325 BSC	
Q	0°	7°
R	0.13	0.30
S	16.95	17.45
T	0.13	---
U	0°	---
V	16.95	17.45
W	0.35	0.45
X	1.6 REF	

Ordering Information