

# MOTOROLA

## SEMICONDUCTOR

### TECHNICAL DATA

## MC68HC705L5

### *Technical Summary*

## 8-Bit Microcontroller Unit

### Introduction

The MC68HC705L5 is a member of Motorola's high-density, complementary metal-oxide semiconductor (HCMOS) M68HC05 Family of 8-bit microcontrollers (MCUs). The M68HC05 Family is based on the customer-specified integrated circuit (CSIC) design strategy. All MCUs in the family use the popular M68HC05 central processor unit (CPU), and the family includes a selection of subsystems, memory sizes and types, and package types.

### Features

- Popular M68HC05 CPU
- Memory-Mapped Input/Output (I/O) Registers
- 8208 Bytes of Erasable Programmable ROM (EPROM)
- 256 Bytes of Static RAM (SRAM)
- General-Purpose Data Pins: 14 Bidirectional Pins, 10 Input-Only Pins and 15 Output-Only Pins
- Full-Duplex Serial Peripheral Interface (SPI)
- Liquid Crystal Display (LCD) Driver with Four Backplane Outputs and 39 Frontplane Outputs
- Fully Static Operation (No Minimum Clock Speed)
- Two On-Chip Oscillators with Crystal/Ceramic Resonator Connections
- 16-Bit Capture/Compare Timer
- 8-Bit Event Counter
- Key Wakeup Interrupt Circuit with Eight Inputs
- Two External Interrupt Input Pins
- Real-Time Interrupt Circuit
- Bootloader Mode
- Power-Saving STOP and WAIT Modes
- Single 2.2–6.0-Volt Power Requirement
- 8 × 8 Unsigned Multiply Instruction
- Programmable Port A, B, and C Pull-Up Resistors
- Programmable Port A, C, D, and E Open-Drain Outputs
- Programmable Edge-Sensitive or Edge- and Level-Sensitive Interrupt Trigger
- RESET Pin Pull-Up Resistor (Mask Option)
- OSC1 Pin/OSC2 Pin Feedback Resistor (Mask Option)
- XOSC1 Pin/XOSC2 Pin Feedback and Damping Resistors (Mask Option)
- 80-Pin Quad Flat Pack (QFP)

This document contains information on a new product. Specifications and information herein are subject to change without notice.


**MOTOROLA**

MC68HC705L5TS/D

**~~This Page Intentionally Left Blank~~**

## Table of Contents

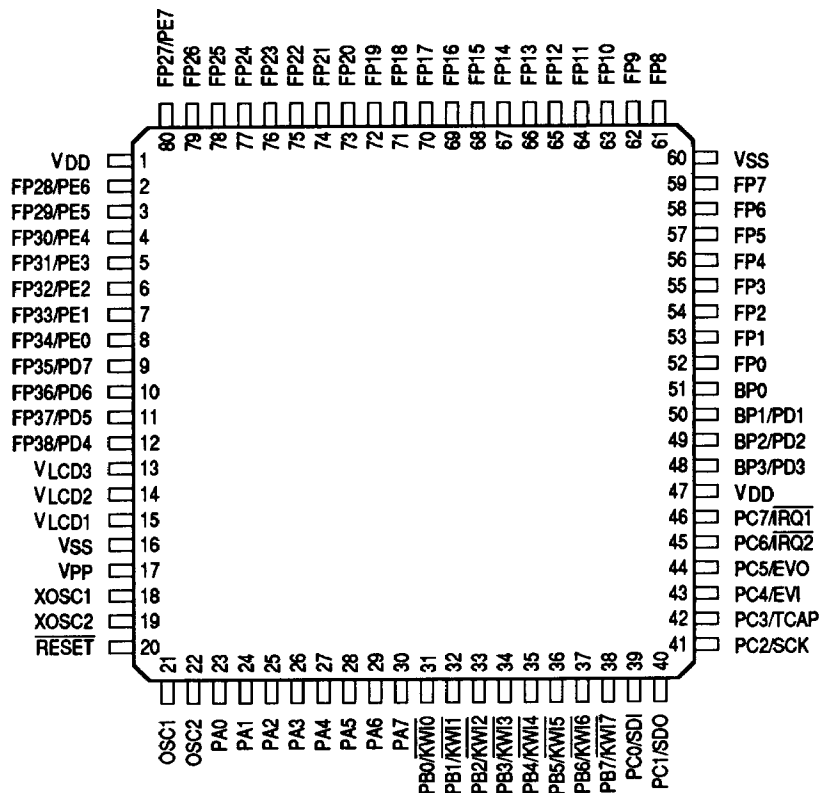
Introduction.....	1
Features .....	1
Pin Assignments .....	5
Order Numbers.....	5
MCU Structure.....	6
Central Processor Unit Registers .....	7
Memory Map.....	8
Option Memory Map.....	12
I/O Ports.....	19
Resets and Interrupts.....	23
Resets.....	23
Interrupts .....	23
Clock Distribution and Control.....	27
OSC and XOSC.....	28
System Clock .....	28
Time Base .....	29
Timers.....	32
Timer 1.....	33
Timer 2.....	38
Liquid Crystal Display Driver.....	43
Serial Peripheral Interface.....	48
Bootloader Mode.....	51
Parallel Bootloader Operation.....	52
Serial Bootloader Operation .....	52
Bootloader Functions .....	53
EPROM .....	55
EPROM Erasing .....	55
EPROM Programming.....	55

## I/O Register Index

Register	Address	Page
DDRA.....	Port A Data Direction Register..... Option Map \$0000.....	14
DDRC.....	Port C Data Direction Register..... Option Map \$0002.....	15
RCR1.....	Resistor Control Register 1..... Option Map \$0008.....	15
RCR2.....	Resistor Control Register 2..... Option Map \$0009.....	16
WOM1.....	Wired-Or Mode Register 1..... Option Map \$000A.....	16
WOM2.....	Wired-Or Mode Register 2..... Option Map \$000B.....	17
KWIEN.....	Key Wakeup Input Enable Register..... Option Map \$000E.....	17
MOSR.....	Mask Option Status Register..... Option Map \$000F.....	18
PORTA.....	Port A Data Register..... \$0000.....	20
PORTB.....	Port B Data Register..... \$0001.....	20
PORTC.....	Port C Data Register..... \$0002.....	21
PORTD.....	Port D Data Register..... \$0003.....	21
PORTE.....	Port E Data Register..... \$0004.....	22
INTCR.....	Interrupt Control Register..... \$0008.....	25
INTSR.....	Interrupt Status Register..... \$0009.....	26
SPCR.....	SPI Control Register..... \$000A.....	49
SPSR.....	SPI Status Register..... \$000B.....	50
SPDR.....	SPI Data Register..... \$000C.....	50
TBCR1.....	Time Base Control Register 1..... \$0010.....	29
TBCR2.....	Time Base Control Register 2..... \$0011.....	30
T1CR.....	Timer 1 Control Register..... \$0012.....	34
T1SR.....	Timer 1 Status Register..... \$0013.....	34
ICR.....	Input Capture Register..... \$0014–\$0015.....	35
T1OCR.....	Timer 1 Output Compare Register..... \$0016–\$0017.....	36
T1CNTR.....	Timer 1 Counter Register..... \$0018–\$0019.....	36
T1ALTCNTR.....	Timer 1 Alternate Counter Register..... \$001A–\$001B.....	37
T2CR.....	Timer 2 Control Register..... \$001C.....	39
T2SR.....	Timer 2 Status Register..... \$001D.....	41
T2OCR.....	Timer 2 Output Compare Register..... \$001E.....	42
T2CNTR.....	Timer 2 Counter Register..... \$001F.....	42
LCDCR.....	LCD Control Register..... \$0020.....	43
LCDDR1–20.....	LCD Data Registers 1–20..... \$0021–\$0034.....	44
EPROGR.....	EPROM Programming Register..... \$003D.....	55
MR.....	Miscellaneous Register..... \$003E.....	28

## Pin Assignments

The following figure shows the pinout of the 80-pin QFP.



NOTE: Pin 17 is for EPROM programming and should be connected to  $V_{DD}$  during normal operation.

## Pin Assignments

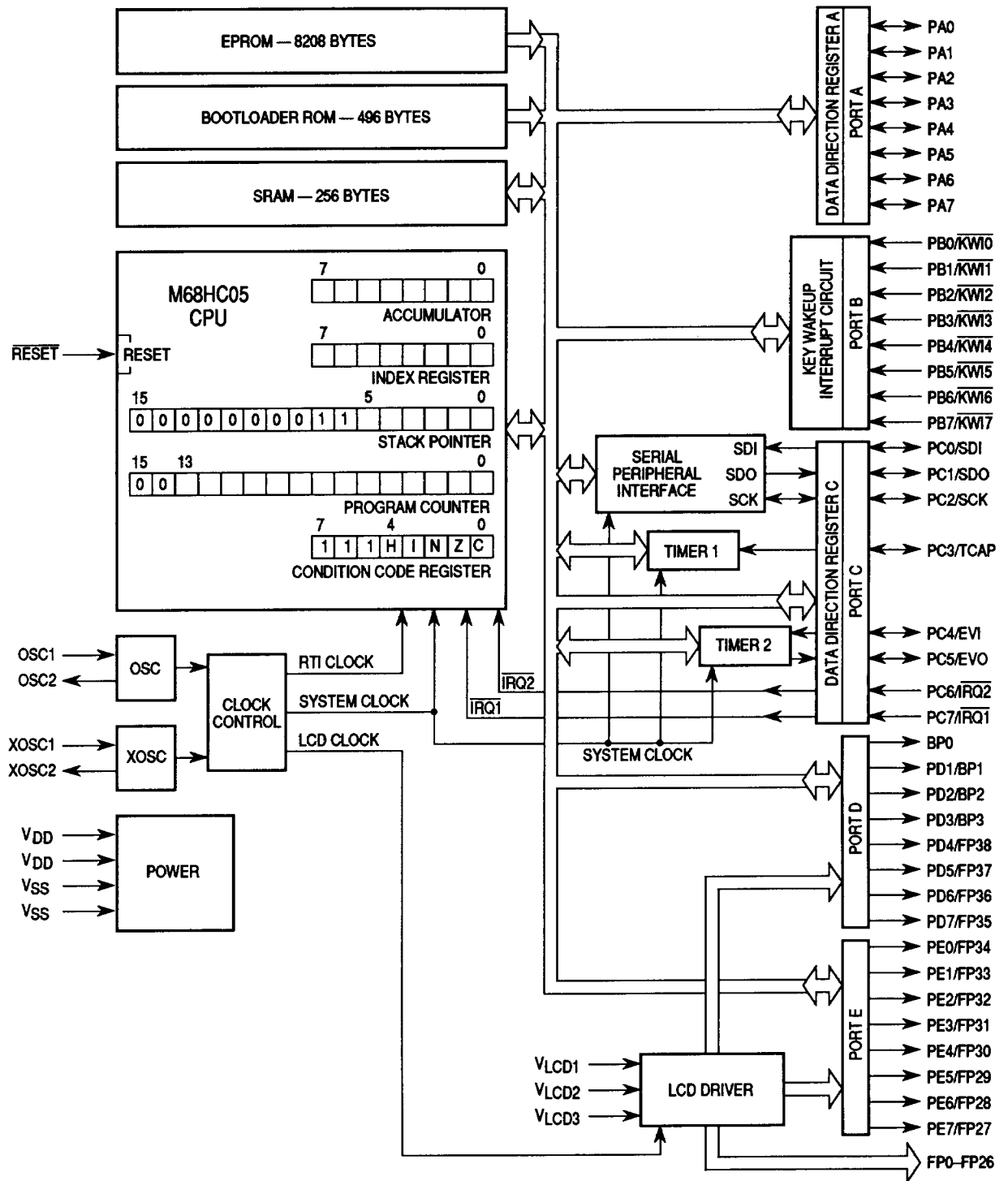
## Order Numbers

Use the following number when ordering this package type.

Package Type	Order Number
80-Pin QFP	MC68HC705L5FU

# MCU Structure

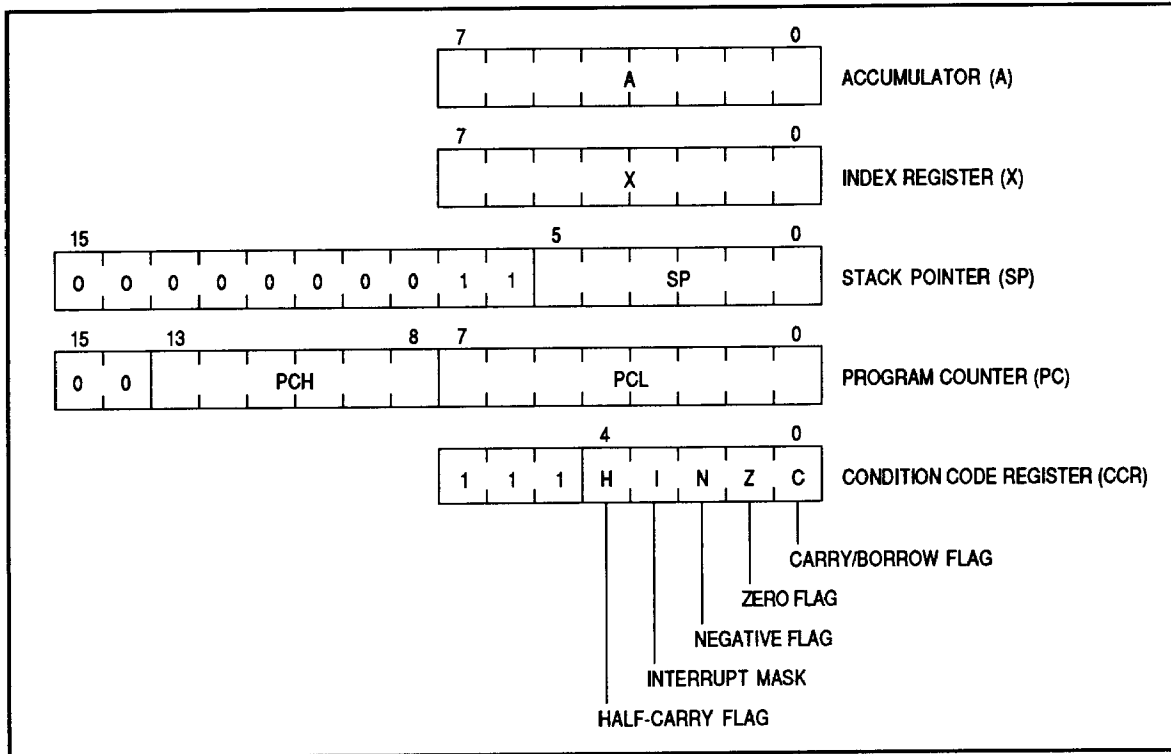
The following figure shows the organization of the MC68HC705L5.



MCU Structure

## Central Processor Unit Registers

Five CPU registers are available to the programmer.



### CPU Registers

The accumulator is a general-purpose 8-bit register. The CPU uses the accumulator to hold operands and results of arithmetic and nonarithmetic operations.

The index register contains a value that indicates an operand address in the indexed addressing modes. The index register can also be used as an auxiliary accumulator for temporary storage.

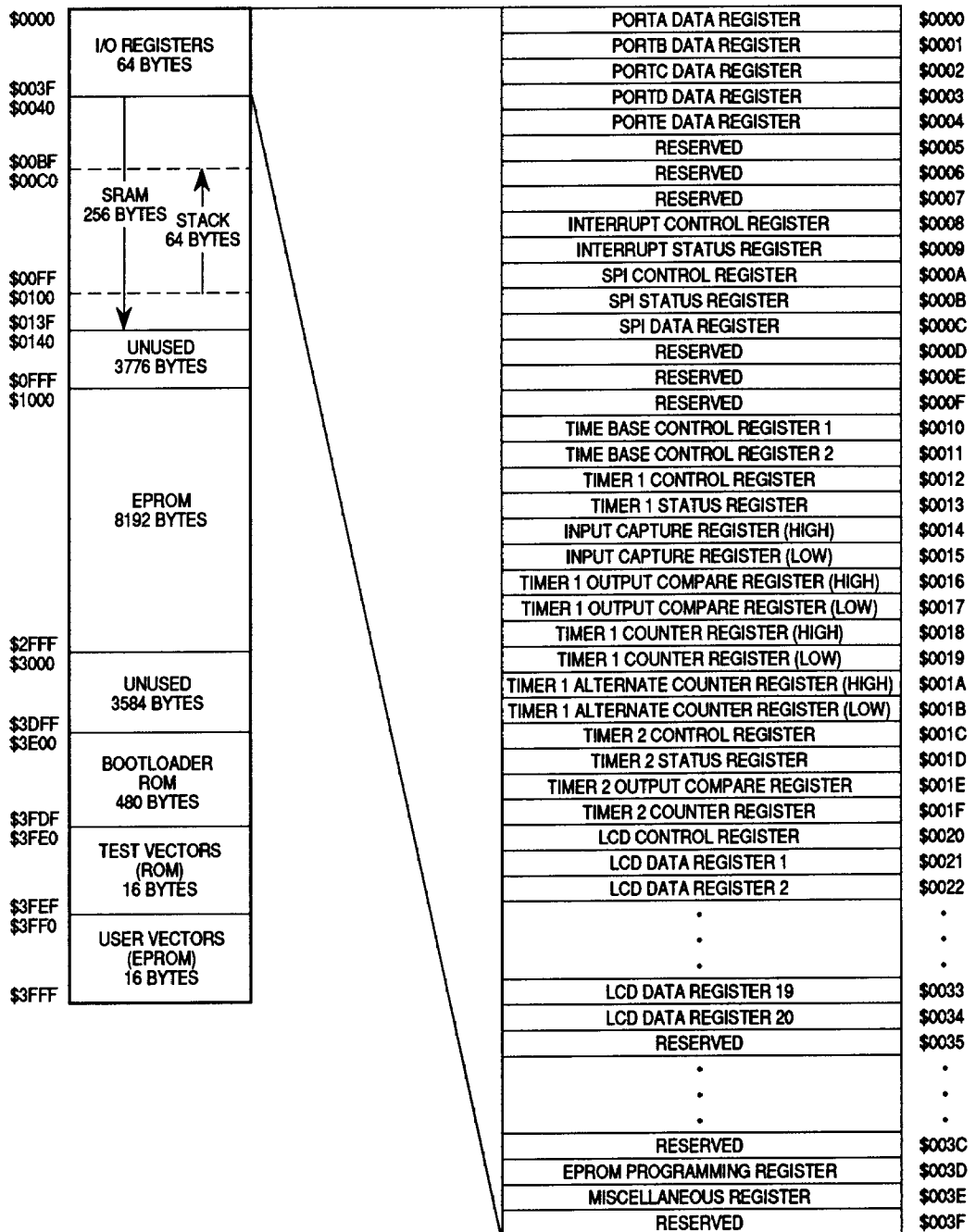
The stack pointer contains the address of the next free location on the stack. The stack pointer decrements as data is pushed onto the stack and increments as data is pulled from the stack.

The program counter contains the address of the next byte for the CPU to fetch.

The condition code register has four status flags that indicate the results of the instruction just executed. A fifth bit is the interrupt mask.

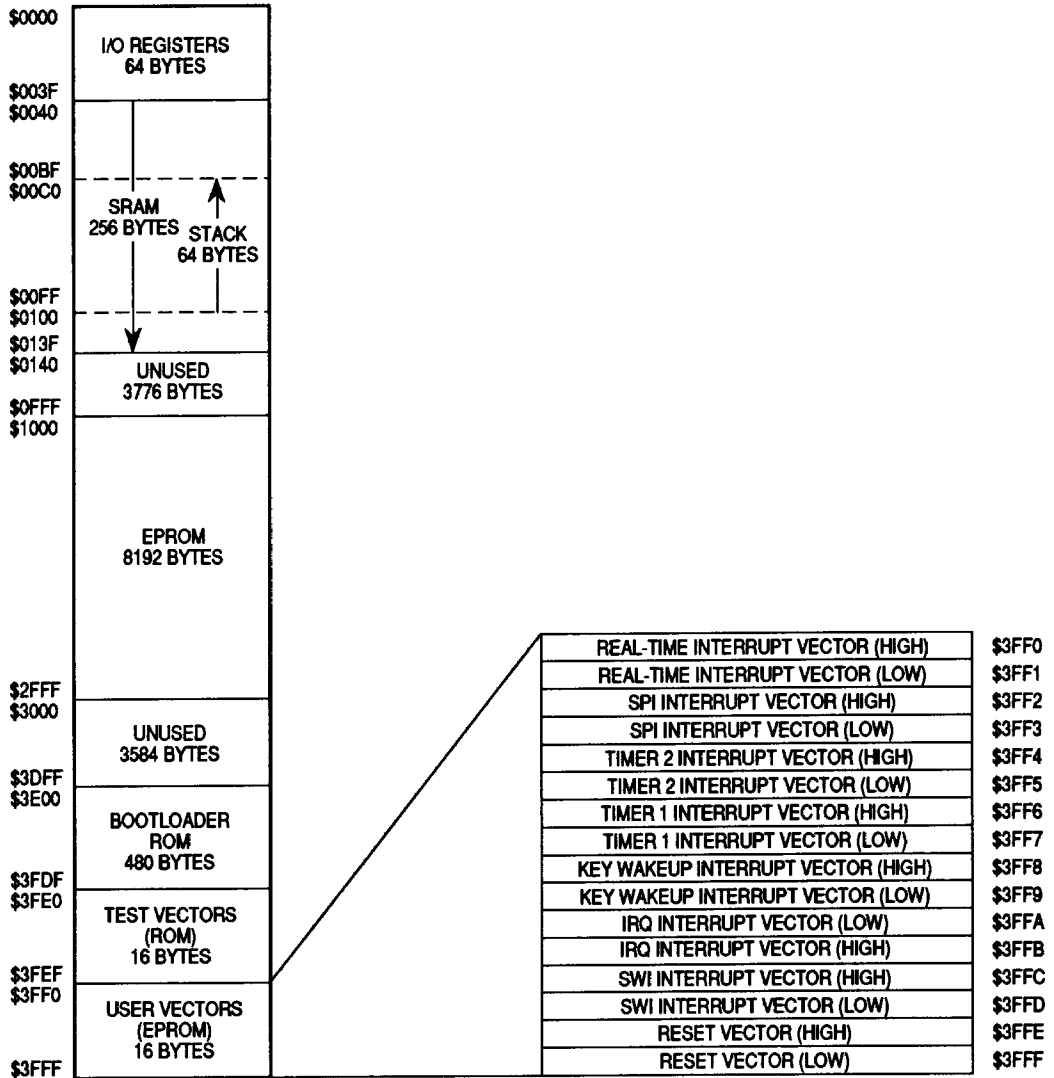
# Memory Map

The MC68HC705L5 can address 16 Kbytes of memory space. The following figures show the organization of the on-chip memory and the function of the I/O registers.



Memory Map





Memory Map (Continued)

\$0000	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0	PORTA
\$0001	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0	PORTB
\$0002	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0	PORTC
\$0003	PD7	PD6	PD5	PD4	PD3	PD2	PD1	1	PORTD
\$0004	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0	PORTE
\$0005	—	—	—	—	—	—	—	—	RESERVED
\$0006	—	—	—	—	—	—	—	—	RESERVED
\$0007	—	—	—	—	—	—	—	—	RESERVED
\$0008	IRQ1E	IRQ2E	0	KWIE	IRQ1S	IRQ2S	0	0	INTCR
\$0009	IRQ1F	IRQ2F	0	KWIF	RIRQ1	RIRQ2	0	RKWIF	INTSR
\$000A	SPIE	SPE	DORD	MSTR	0	0	0	SPR	SPCR
\$000B	SPIF	DCOL	0	0	0	0	0	0	SPSR
\$000C	Bit 7	6	5	4	3	2	1	Bit 0	SPDR
\$000D	—	—	—	—	—	—	—	—	RESERVED
\$000E	—	—	—	—	—	—	—	—	RESERVED
\$000F	—	—	—	—	—	—	—	—	RESERVED
\$0010	TBCLK	0	LCLK	0	0	0	T2R1	T2R0	TBCR1
\$0011	RTIF	RTIE	RTR1	RTR0	RRTIF	0	0	0	TBCR2
\$0012	ICIE	T1OCIE	TOIE	0	0	0	IEDG	0	T1CR
\$0013	ICF	T1OCIF	TOF	0	0	0	0	0	T1SR
\$0014	Bit 15	14	13	12	11	10	9	Bit 8	ICR (HIGH)
\$0015	Bit 7	6	5	4	3	2	1	Bit 0	ICR (LOW)
\$0016	Bit 15	14	13	12	11	10	9	Bit 8	T1OCR (HIGH)
\$0017	Bit 7	6	5	4	3	2	1	Bit 0	T1OCR (LOW)
\$0018	Bit 15	14	13	12	11	10	9	Bit 8	T1CNTR (HIGH)
\$0019	Bit 7	6	5	4	3	2	1	Bit 0	T1CNTR (LOW)
\$001A	Bit 15	14	13	12	11	10	9	Bit 8	T1ALTCNTR (HIGH)
\$001B	Bit 7	6	5	4	3	2	1	Bit 0	T1ALTCNTR (LOW)
\$001C	T2IE	T2OCIE	0	T2CLK	T2IM	T2IL	T2OE	T2OS	T2CR
\$001D	T2IF	T2OCIF	0	0	RT2IF	RT2OCIF	0	0	T2SR
\$001E	Bit 7	6	5	4	3	2	1	Bit 0	T2OCR
\$001F	Bit 7	6	5	4	3	2	1	Bit 0	T2CNTR

### I/O Registers

\$0020	LCDE	DUTY1	DUTY0	0	PEH	PEL	PDH	0	LCDCR
\$0021	F1B3	F1B2	F1B1	F1B0	F0B3	F0B2	F0B1	F0B0	LCDDR1
\$0022	F3B3	F3B2	F3B1	F3B0	F2B3	F2B2	F2B1	F2B0	LCDDR2
\$0023	F5B3	F5B2	F5B1	F5B0	F4B3	F4B2	F4B1	F4B0	LCDDR3
\$0024	F7B3	F7B2	F7B1	F7B0	F6B3	F6B2	F6B1	F6B0	LCDDR4
\$0025	F9B3	F9B2	F9B1	F9B0	F8B3	F8B2	F8B1	F8B0	LCDDR5
\$0026	F11B3	F11B2	F11B1	F11B0	F10B3	F10B2	F10B1	F10B0	LCDDR6
\$0027	F13B3	F13B2	F13B1	F13B0	F12B3	F12B2	F12B1	F12B0	LCDDR7
\$0028	F15B3	F15B2	F15B1	F15B0	F14B3	F14B2	F14B1	F14B0	LCDDR8
\$0029	F17B3	F17B2	F17B1	F17B0	F16B3	F16B2	F16B1	F16B0	LCDDR9
\$002A	F19B3	F19B2	F19B1	F19B0	F18B3	F18B2	F18B1	F18B0	LCDDR10
\$002B	F21B3	F21B2	F21B1	F21B0	F20B3	F20B2	F20B1	F20B0	LCDDR11
\$002C	F23B3	F23B2	F23B1	F23B0	F22B3	F22B2	F22B1	F22B0	LCDDR12
\$002D	F25B3	F25B2	F25B1	F25B0	F24B3	F24B2	F24B1	F24B0	LCDDR13
\$002E	F27B3	F27B2	F27B1	F27B0	F26B3	F26B2	F26B1	F26B0	LCDDR14
\$002F	F29B3	F29B2	F29B1	F29B0	F28B3	F28B2	F28B1	F28B0	LCDDR15
\$0030	F31B3	F31B2	F31B1	F31B0	F30B3	F30B2	F30B1	F30B0	LCDDR16
\$0031	F33B3	F33B2	F33B1	F33B0	F32B3	F32B2	F32B1	F32B0	LCDDR17
\$0032	F35B3	F35B2	F35B1	F35B0	F34B3	F34B2	F34B1	F34B0	LCDDR18
\$0033	F37B3	F37B2	F37B1	F37B0	F36B3	F36B2	F36B1	F36B0	LCDDR19
\$0034	—	—	—	—	F38B3	F38B2	F38B1	F38B0	LCDDR20
\$0035	—	—	—	—	—	—	—	—	RESERVED
.					.				.
.					.				.
.					.				.
\$003C	—	—	—	—	—	—	—	—	RESERVED
\$003D	—	—	—	—	—	0	ELAT	EPGM	EPROGR
\$003E	FTUP	STUP	0	0	SYS1	SYS0	FOSCE	OPTM	MR
\$003F	—	—	—	—	—	—	—	—	RESERVED

I/O Registers (Continued)

## Option Memory Map

The option memory map contains control registers to implement functions that otherwise would be mask options. The two data direction registers, DDRA and DDRC, are also in the option memory map. When the OPTM bit in the miscellaneous register (MR) is set, the option memory map appears. The option memory map replaces addresses \$0000–\$000F in the memory map and controls the following options:

**In the resistor control register 1 (RCR1) at \$0008:**

1. Port B pull-up resistors (RAH and RAL bits)
2. Port A pull-up resistors (RBH and RBL bits)

**In the resistor control register 2 (RCR2) at \$0009:**

3. Port C pull-up resistors (RC7–0 bits)

**In the wired-or mode register 1 (WOM1) at \$000A:**

4. Port D open-drain mode (DWOMH and DWOML bits)
5. Port E open-drain mode (EWOMH and EWOML bits)
6. Port A open-drain mode (AWOMH and AWOML bits)

**In the wired-or mode register 2 (WOM2) at \$000B:**

7. Port C open-drain mode (CWOM5–0 bits)

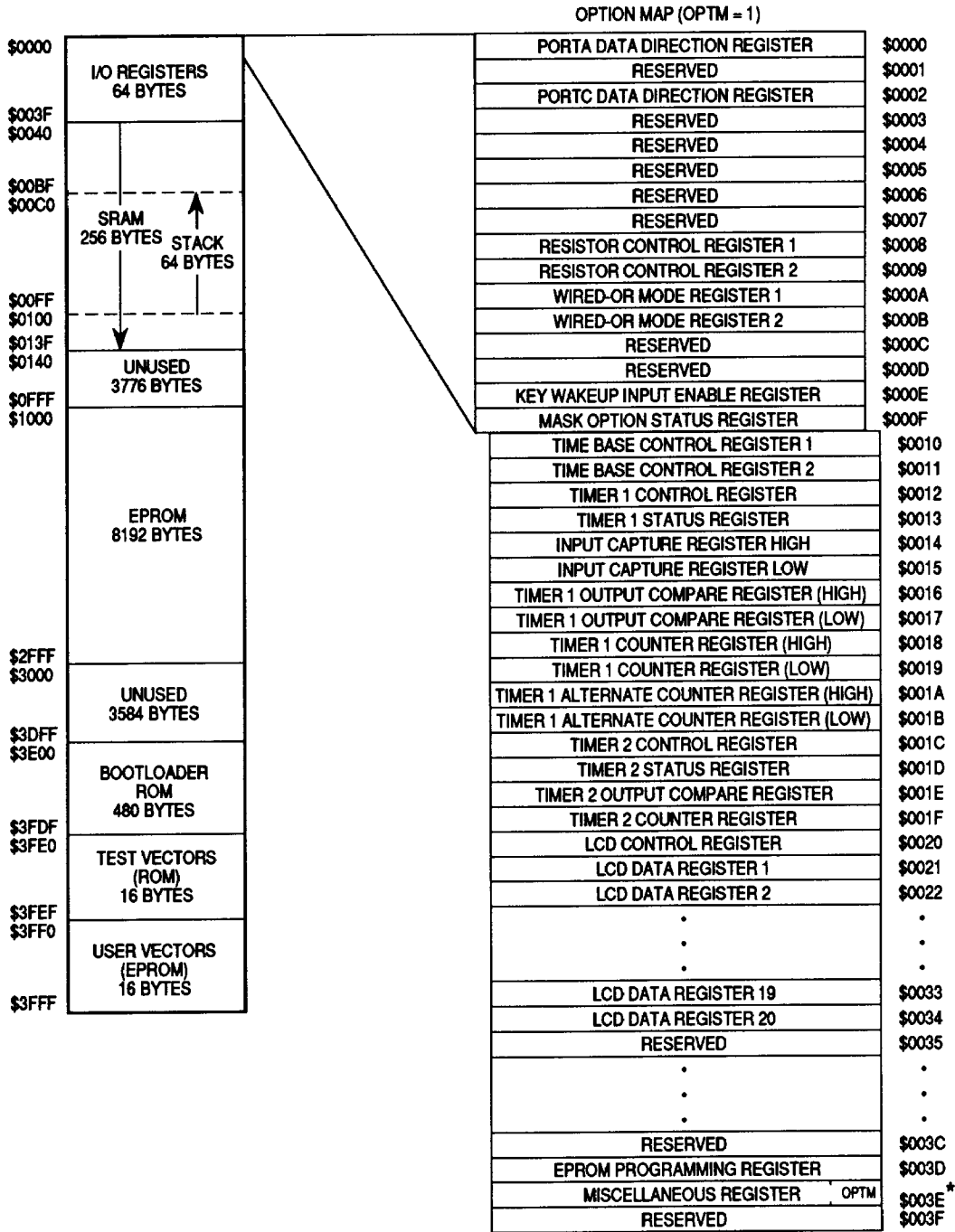
**In the key wakeup input enable register (KWIEN) at \$000E:**

8. Key wakeup input enable (KWIE7–0 bits)

**The mask option status register (MOSR) at \$000F shows the status of the following mask options:**

1.  $\overline{\text{RESET}}$  pin pull-up resistor (RSTR bit)
2. OSC1/OSC2 feedback resistor (OSCR bit)
3. XOSC1/XOSC2 feedback resistor and damping resistor at XOSC2 (XOSCR bit)

The following figures show the organization of the option memory map and the function of the option map registers.



\*The option map appears when the OPTM bit is set. OPTM is bit 0 in the miscellaneous register. The option map replaces addresses \$0000-\$000F of the main memory map while OPTM is set.

### Option Memory Map

\$0000	DDRA7	DDRA6	DDRA5	DDRA4	DDRA3	DDRA2	DDRA1	DDRA0	DDRA
\$0001	—	—	—	—	—	—	—	—	RESERVED
\$0002	0	0	DDRC5	DDRC4	DDRC3	DDRC2	DDRC1	DDRC0	DDRC
\$0003	—	—	—	—	—	—	—	—	RESERVED
\$0004	—	—	—	—	—	—	—	—	RESERVED
\$0005	—	—	—	—	—	—	—	—	RESERVED
\$0006	—	—	—	—	—	—	—	—	RESERVED
\$0007	—	—	—	—	—	—	—	—	RESERVED
\$0008	0	0	0	0	RBH	RBL	RAH	RAL	RCR1
\$0009	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	RCR2
\$000A	DWOMH	DWOML	EWOMH	EWOML	0	0	AWOMH	AWOML	WOM1
\$000B	0	0	CWOM5	CWOM4	CWOM3	CWOM2	CWOM1	CWOM0	WOM2
\$000C	—	—	—	—	—	—	—	—	RESERVED
\$000D	—	—	—	—	—	—	—	—	RESERVED
\$000E	KWIE7	KWIE6	KWIE5	KWIE4	KWIE3	KWIE2	KWIE1	KWIE0	KWIEN
\$000F	RSTR	OSCR	XOSCR	0	0	0	0	0	MOSR

### Option Map I/O Registers

**DDRA — Port A Data Direction Register**

**Option Map \$0000**

	Bit 7	6	5	4	3	2	1	Bit 0
	DDRA7	DDRA6	DDRA5	DDRA4	DDRA3	DDRA2	DDRA1	DDRA0
RESET:	0	0	0	0	0	0	0	0

**DDRA7–DDRA0 — Port A Data Direction Bits**

These read/write bits determine whether the PA7–PA0 pins are inputs or outputs.

1 = Corresponding port pin configured as output

0 = Corresponding port pin configured as input

---

**DDRC — Port C Data Direction Register****Option Map \$0002**

	Bit 7	6	5	4	3	2	1	Bit 0
	0	0	DDRC5	DDRC4	DDRC3	DDRC2	DDRC1	DDRC0
RESET:	0	0	0	0	0	0	0	0

**DDRC5–DDRC0 — Port C Data Direction Bits**

These read/write bits determine whether pins PC5–PC0 are inputs or outputs.

- 1 = Corresponding port pin configured as output
- 0 = Corresponding port pin configured as input

---

**RCR1 — Resistor Control Register 1****Option Map \$0008**

	Bit 7	6	5	4	3	2	1	Bit 0
	0	0	0	0	RBH	RBL	RAH	RAL
RESET:	0	0	0	0	0	0	0	0

**RBH — Resistors B High**

This read/write bit connects internal pull-up resistors to the upper four port B pins.

- 1 = Pull-up resistors connected to pins PB7–PB4
- 0 = No pull-up resistors connected to pins PB7–PB4

**RBL — Resistors B Low**

This read/write bit connects internal pull-up resistors to the lower four port B pins.

- 1 = Pull-up resistors connected to pins PB3–PB0
- 0 = No pull-up resistors connected to pins PB3–PB0

**RAH — Resistors A High**

This read/write bit connects internal pull-up resistors to the upper four port A pins.

- 1 = Pull-up resistors connected to pins PA7–PA4
- 0 = No pull-up resistors connected to pins PA7–PA4

**RAL — Resistors A Low**

This read/write bit connects internal pull-up resistors to the lower four port A pins.

- 1 = Pull-up resistors connected to pins PA3–PA0
  - 0 = No pull-up resistors connected to pins PA3–PA0
-

---

**RCR2 — Resistor Control Register 2****Option Map \$0009**

	Bit 7	6	5	4	3	2	1	Bit 0
	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0
RESET:	0	0	0	0	0	0	0	0

**RC7–RC0 — Resistors C**

These read/write bits connect internal pull-up resistors to the port C pins.

1 = Pull-up resistor connected to the corresponding pin

0 = No pull-up resistor connected to the corresponding pin

---

**WOM1 — Wired-Or Mode Register 1****Option Map \$000A**

	Bit 7	6	5	4	3	2	1	Bit 0
	DWOMH	DWOML	EWOMH	EWOML	0	0	AWOMH	AWOML
RESET:	0	0	0	0	0	0	0	0

**DWOMH — Port D Wired-Or Mode (High Bits)**

This read/write bit configures the upper four port D pins as open-drain outputs. The PDH bit in the LCD control register must be set, configuring PD7–PD4 as general-purpose outputs.

1 = PD7–PD4 configured as open-drain outputs

0 = PD7–PD4 configured as normal outputs

**DWOML — Port D Wired-Or Mode (Low Bits)**

This read/write bit configures port D pins PD3–PD1 as open-drain outputs. The DUTY1 and DUTY0 bits in the LCD control register must be programmed to configure the PDx pin as a general-purpose output.

1 = PD3–PD1 configured as open-drain outputs

0 = PD3–PD1 configured as normal outputs

**EWOMH — Port E Wired-Or Mode (High Bits)**

This read/write bit configures the upper four port E pins as open-drain outputs. The PEH bit in the LCD control register must be set, configuring PE7–PE4 as general-purpose outputs.

1 = PE7–PE4 configured as open-drain outputs

0 = PE7–PE4 configured as normal outputs

**EWOML — Port E Wired-Or Mode (Low Bits)**

This read/write bit configures the lower four port E pins as open-drain outputs. The PEL bit in the LCD control register must be set, configuring PE3–PE0 as general imer purpose outputs.

1 = PE3–PE0 configured as open-drain outputs

0 = PE3–PE0 not configured as open-drain outputs



### AWOMH — Port A Wired-Or Mode (High Bits)

This read/write bit configures the upper four port A pins as open-drain outputs. The DDRAx bit in the port A data direction register must be set, configuring the PAX pin as an output.

- 1 = PA7–PA4 configured as open-drain outputs
- 0 = PA7–PA4 not configured as open-drain outputs

### AWOML — Port A Wired-Or Mode (Low Bits)

This read/write bit configures the lower four port A pins as open-drain outputs. The DDRAx bit in the port A data direction register must be set, configuring the PAX pin as an output.

- 1 = PA3–PA0 configured as open-drain outputs
- 0 = PA3–PA0 configured as normal outputs

---

### WOM2 — Wired-Or Mode Register 2

Option Map \$000B

Bit 7	6	5	4	3	2	1	Bit 0
0	0	CWOM5	CWOM4	CWOM3	CWOM2	CWOM1	CWOM0

RESET: 0 0 0 0 0 0 0 0

### CWOM5–CWOM0 — Port C Wired-Or Mode

These read/write bits configure the port C pins PC5–PC0 as open-drain outputs. The DDRCx bit in the port C data direction register must be set, configuring the PCx pin as an output.

- 1 = PCx configured as open-drain output
- 0 = PCx configured as normal output

---

### KWIEN — Key Wakeup Input Enable Register

Option Map \$000E

Bit 7	6	5	4	3	2	1	Bit 0
KWIE7	KWIE6	KWIE5	KWIE4	KWIE3	KWIE2	KWIE1	KWIE0

RESET: 0 0 0 0 0 0 0 0

### KWIE7–KWIE0 — Key Wakeup Enable

These read/write bits enable the KW17–KW10 pins to function as key wakeup interrupt inputs.

- 1 = KW1x input enabled
- 0 = KW1x input disabled

**MOSR — Mask Option Status Register**

**Option Map \$000F**

	Bit 7	6	5	4	3	2	1	Bit 0
	RSTR	OSCR	XOSCR	0	0	0	0	0
RESET:	U	U	U	0	0	0	0	0

(U = UNAFFECTED)

This read-only register indicates the status of three mask options.

**RSTR — Reset Resistor**

This flag is set when the mask-optional internal pull-up resistor is connected to the  $\overline{\text{RESET}}$  pin.

- 1 = Pull-up resistor connected to  $\overline{\text{RESET}}$  pin
- 0 = No pull-up resistor on  $\overline{\text{RESET}}$  pin

**OSCR — OSC Resistor**

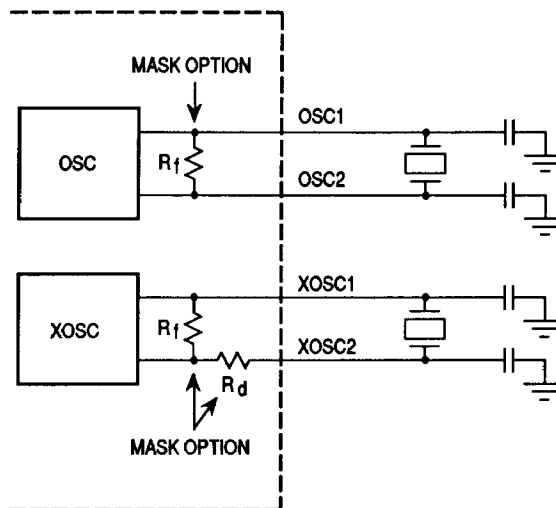
This flag is set when the mask-optional internal feedback resistor,  $R_f$ , is connected between OSC1 and OSC2.

- 1 = Feedback resistor connected
- 0 = No feedback resistor connected

**XOSCR — XOSC Resistor**

This flag is set when the mask-optional internal feedback resistor,  $R_f$ , is connected between XOSC1 and XOSC2, and the mask-optional damping resistor,  $R_d$ , is connected to XOSC2.

- 1 = Feedback and damping resistors connected
- 0 = No feedback or damping resistors connected

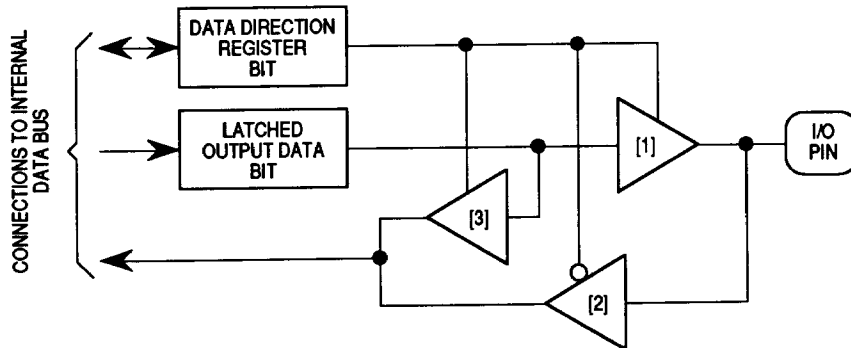


**OSC/XOSC Resistor Mask Options**

## I/O Ports

The MC68HC705L5 has 14 bidirectional I/O pins, 10 input-only pins, and 15 output-only pins. The contents of the data direction registers (DDRs) determine whether each bidirectional I/O pin is an input or an output.

When an I/O pin is programmed as an output, reading the associated port bit actually reads the value of the output data latch and not the voltage on the pin itself. When a pin is programmed as an input, reading the port bit reads the voltage level on the I/O pin. The output data latch can always be written, regardless of the state of its DDR bit.



- [1] This output buffer enables the latched data to drive the pin when DDR bit is 1 (output mode).
- [2] This input buffer is enabled when DDR bit is 0 (input mode).
- [3] This input buffer is enabled when DDR bit is 1 (output mode).

### Parallel I/O Circuit

#### I/O Pin Functions

R/ $\overline{W}$ *	DDR	I/O Pin Functions
0	0	The I/O pin is in input mode. Data is written into the output data latch.
0	1	Data is written into the output data latch, which drives the I/O pin.
1	0	The state of the I/O pin is read.
1	1	The I/O pin is in output mode. The output data latch is read.

\* $\overline{RW}$  is an internal signal.

The following figures detail the four I/O port data registers.

---

**PORTA — Port A Data Register**

**\$0000**

Bit 7	6	5	4	3	2	1	Bit 0
PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0

RESET: NOT CHANGED BY RESET

Port A is an eight-bit bidirectional port. The DDRA7–0 bits in the port A data direction register in the option map determine the data direction of pins PA7–PA0. At power-up, all eight pins are configured as inputs.

On-chip port A pull-up resistors are selectable in resistor control register 1 in the option memory map.

Port A outputs are configurable as open-drain outputs in wired-OR mode register 1 in the option memory map.

---

**PORTB — Port B Data Register**

**\$0001**

Bit 7	6	5	4	3	2	1	Bit 0
PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0

RESET: NOT CHANGED BY RESET

ALTERNATE

FUNCTION: KWI7 KWI6 KWI5 KWI4 KWI3 KWI2 KWI1 KWI0

Port B is an eight-bit input-only port that shares its pins with the key wakeup interrupt system.

On-chip port B pull-up resistors are selectable in resistor control register 1 in the option memory map.

---

**PORTC — Port C Data Register****\$0002**

Bit 7	6	5	4	3	2	1	Bit 0
PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0

RESET: NOT CHANGED BY RESET

ALTERNATE  
FUNCTION: IRQ1 IRQ2 EVO EVI TCAP SCK SDO SDI

Port C is an eight-bit port that shares its pins with several subsystems. PC7 and PC6 are input-only pins that function as external interrupt inputs when external interrupts are enabled by the IRQ1E and IRQ2E bits in the interrupt control register.

PC5 is a bidirectional pin that functions as the timer 2 output (EVO) when the EVO is enabled by the T2OE bit in the timer 2 control register.

PC4 and PC3 are bidirectional pins that can function as the timer 2 input (EVI) and the timer 1 input (TCAP).

PC2–0 are bidirectional pins that function as serial peripheral interface (SPI) pins when the SPI is enabled by the SPE bit in the SPI control register.

On-chip port C pull-up resistors are selectable in resistor control register 2 in the option memory map.

Port C outputs are configurable as open-drain outputs in wired-OR mode register 2 in the option memory map.

---

**PORTD — Port D Data Register****\$0003**

Bit 7	6	5	4	3	2	1	Bit 0
PD7	PD6	PD5	PD4	PD3	PD2	PD1	1

RESET: 1 1 1 1 1 1 1 1

ALTERNATE  
FUNCTION: FP35 FP36 FP37 FP38 BP3 BP2 BP1

Port D is a seven-bit general-purpose output-only port that shares its pins with the LCD drivers.

Port D outputs are configurable as open-drain outputs in wired-OR mode register 1 in the option memory map.

---

**PORTE — Port E Data Register****\$0004**

	Bit 7	6	5	4	3	2	1	Bit 0
	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0
RESET:	1	1	1	1	1	1	1	1
ALTERNATE FUNCTION:	FP27	FP28	FP29	FP30	FP31	FP32	FP33	FP34

Port E is an eight-bit general-purpose output port that shares its pins with the LCD frontplane drivers.

Port E outputs are configurable as open-drain outputs in wired-OR mode register 1 in the option memory map.

---

## Resets and Interrupts

There are three reset conditions that force the CPU to a user-defined starting address. There are also seven ways to interrupt normal processing to service an external or subsystem event.

### Resets

A reset occurs under the following conditions:

- Power-on reset (POR) — A POR is generated when a positive transition occurs on the VDD pin.
- External reset — A reset is generated when a logical zero is applied to the  $\overline{\text{RESET}}$  pin.

The following internal actions occur on reset:

- All implemented data direction bits are cleared, making all I/O pins inputs.
- The stack pointer is loaded with \$FF.
- The global interrupt mask (I bit) in the condition code register is set, inhibiting interrupts.
- The timer clock divider stages are reset. Timer 1 is loaded with \$FFFC.
- Timer 2 is loaded with \$01.
- All local interrupt enable bits are cleared to disable interrupts.
- The STOP latch is cleared to enable MCU clocks.
- The WAIT latch is cleared to wake the CPU from the WAIT mode.
- The program counter is loaded with the reset vector.
- The SYS1 and SYS0 bits in the miscellaneous register are preset to 1 and 0, selecting OSC ÷ 64 as the system clock.

### Interrupts

The CPU can be interrupted in the following ways:

- Software interrupt (SWI) — The SWI instruction causes a nonmaskable interrupt to be executed.
- External interrupt — An interrupt signal on either the  $\overline{\text{IRQ1}}$  pin or the  $\overline{\text{IRQ2}}$  pin causes an external interrupt if the corresponding interrupt enable bit (IRQ1E or IRQ2E) is set. Both external interrupts use the same interrupt vector. The interrupt signal can be either of the following conditions:

- A falling edge
- Either a falling edge or a low level

The states of the IRQ1S and IRQ2S bits determine which condition on the  $\overline{\text{IRQ}}$  pins the CPU recognizes as an interrupt signal. The IRQ1E and IRQ2E bits enable external interrupts.

- Key wakeup interrupt — The key wakeup enable register (\$000E in the option memory map) enables the key wakeup pins,  $\overline{\text{KWI7}}$ – $\overline{\text{KWI0}}$ . See **Option Memory Map**. A falling edge on one of the enabled key wakeup pins sets the KWIF flag. If the KWIE bit is also set, a key wakeup interrupt is requested.
- Timer 1 interrupt — The CPU recognizes a timer 1 interrupt if one of the three timer 1 interrupt flags (ICF, T1OCIF, or TOF) is set while its corresponding timer 1 interrupt enable bit (ICIE, T1OCIE, or TOIE) is set.

- Timer 2 interrupt — The CPU recognizes a timer 2 interrupt if one of the two timer 2 interrupt flags (T2IF or T2OCIF) is set while its corresponding timer 2 interrupt enable bit (T2IE or T2OCIE) is set.
- Serial peripheral interface (SPI) interrupt — After each one-byte data transfer, the SPI sets the SPIF flag. If the SPI interrupt enable bit (SPIE) is also set, an interrupt is requested.
- Real-time interrupt — The real-time interrupt circuit sets the RTIF flag at the end of the selected real-time interrupt period. If the RTIE bit is also set, a real-time interrupt is requested.

The following actions occur as a result of an interrupt:

- The CPU registers are stored in the stack in the order PCL, PCH, X, A, CCR.
- The interrupt mask (I bit) in the condition code register is set to prevent additional interrupts.
- The program counter is loaded with the appropriate interrupt vector.
- The RTI (return from interrupt) instruction causes the CPU registers to be recovered from the stack in the order CCR, A, X, PCH, PCL. Normal processing resumes.

#### Reset/Interrupt Vector Assignments

Type	Source	Local Mask	Global Mask	Priority (1 = High)	Vector Address
Reset	Reset Pin POR	None	None	1	\$3FFE-\$3FFF
		None	None	1	\$3FFE-\$3FFF
SWI	None	None	None	*	\$3FFC-\$3FFD
External	IRQ1 Pin	IRQ1E	I	2	\$3FFA-\$3FFB
	IRQ2 Pin	IRQ2E	I	2	\$3FFA-\$3FFB
KWI	KWIF	KWIE	I	3	\$3FF8-\$3FF9
Timer 1	ICF	ICIE	I	4	\$3FF6-\$3FF7
	T1OCIF	T1OCIE	I	4	\$3FF6-\$3FF7
	TOF	TOIE	I	4	\$3FF6-\$3FF7
Timer 2	T2IF	T2IE	I	5	\$3FF4-\$3FF5
	T2OCIF	T2OCIE	I	5	\$3FF4-\$3FF5
SPI	SPIF	SPIE	I	6	\$3FF2-\$3FF3
RTI	RTIF	RTIE	I	7	\$3FF0-\$3FF1

\*Same level as an instruction



---

**INTCR — Interrupt Control Register****\$0008**

	Bit 7	6	5	4	3	2	1	Bit 0
	IRQ1E	IRQ2E	0	KWIE	IRQ1S	IRQ2S	0	0
RESET:	0	0	0	0	0	0	0	0

**IRQ1E — IRQ1 Enable**

This read/write bit enables IRQ1 interrupts.

1 = IRQ1 interrupts enabled

0 = IRQ1 interrupts disabled

**IRQ2E — IRQ2 Enable**

This read/write bit enables IRQ2 interrupts.

1 = IRQ2 interrupts enabled

0 = IRQ2 interrupts disabled

**KWIE — Key Wakeup Interrupt Enable**

This read/write bit enables key wakeup interrupts.

1 = Key wakeup interrupts enabled

0 = Key wakeup interrupts disabled

**IRQ1S — IRQ1 Sensitivity**This read/write bit determines the  $\overline{\text{IRQ1}}$  pin interrupt trigger sensitivity.1 = Negative edge on  $\overline{\text{IRQ1}}$  pin triggers interrupt request0 = Negative edge or low level on  $\overline{\text{IRQ1}}$  pin triggers interrupt request**IRQ2S — IRQ2 Sensitivity**This read/write bit determines the  $\overline{\text{IRQ2}}$  pin interrupt trigger sensitivity.1 = Negative edge on  $\overline{\text{IRQ2}}$  pin triggers interrupt request0 = Negative edge or low level on  $\overline{\text{IRQ2}}$  pin triggers interrupt request

---

**INTSR — Interrupt Status Register****\$0009**

	Bit 7	6	5	4	3	2	1	Bit 0
	IRQ1F	IRQ2F	0	KWIF	RIRQ1	RIRQ2	0	RKWIF
RESET:	0	0	0	0	0	0	0	0

**IRQ1F — IRQ1 Flag**

This read-only bit is set when an interrupt signal occurs on the IRQ1 pin. Clear the IRQ1F flag by writing a one to the RIRQ1 bit.

**IRQ2F — IRQ2 Flag**

This read-only bit is set when an interrupt signal occurs on the IRQ2 pin. Clear the IRQ2F flag by writing a one to the RIRQ2 bit.

**KWIF — Key Wakeup Interrupt Flag**

This read-only bit is set when a key wakeup interrupt signal occurs on one of the KW17–0 pins. Clear the KWIF bit by writing a one to the RKWIF bit.

**RIRQ1 — Reset IRQ1 Flag**

Setting this write-only bit clears the IRQ1F bit. Writing a zero to RIRQ1 has no effect. RIRQ1 always reads as zero.

**RIRQ2 — Reset IRQ2 Flag**

Setting this write-only bit clears the IRQ2F bit. Writing a zero to RIRQ2 has no effect. RIRQ2 always reads as zero.

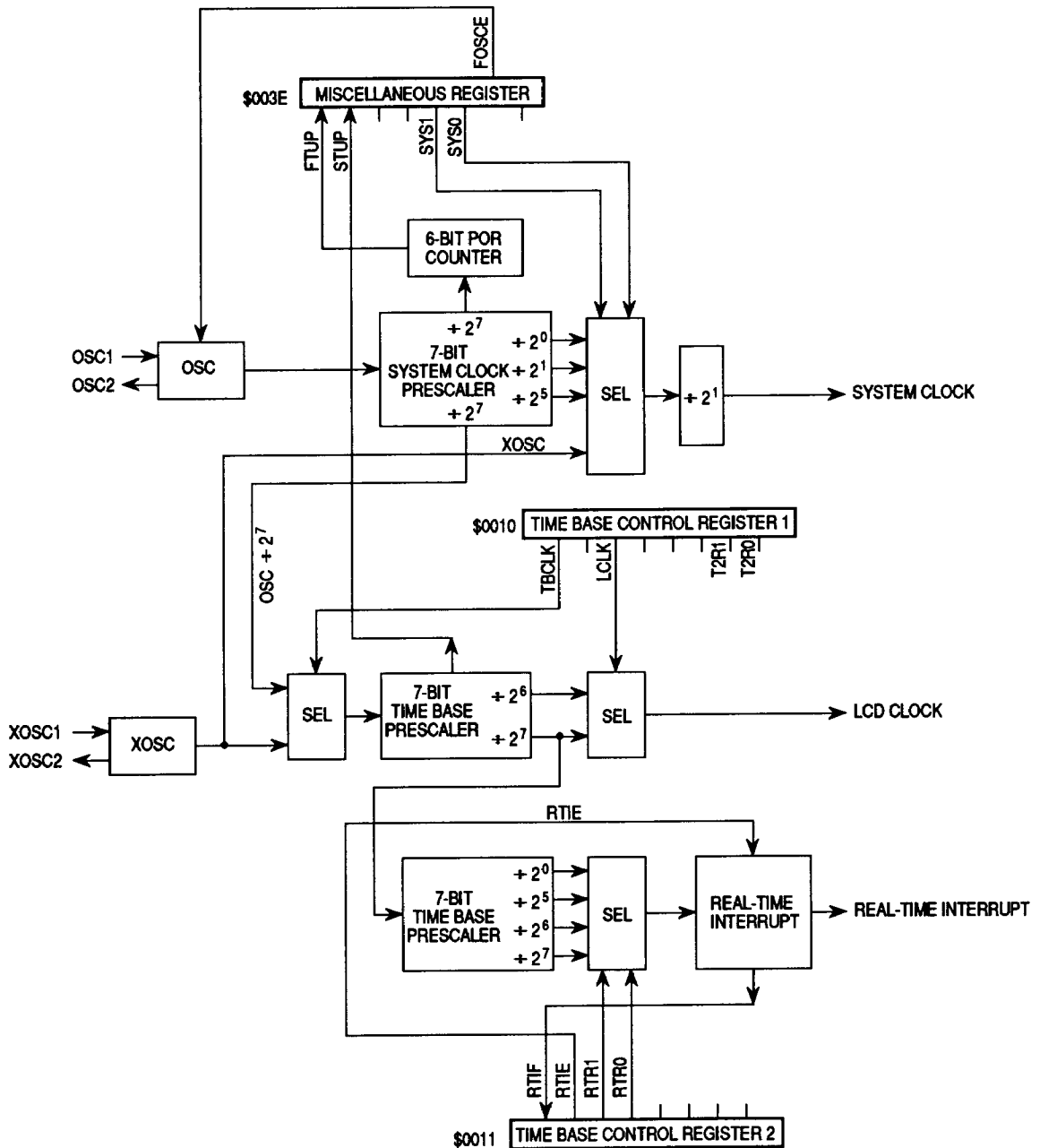
**RKWIF — Reset Key Wakeup Interrupt Flag**

Setting this write-only bit clears the KWIF bit. Writing a zero to RKWIF has no effect. RKWIF always reads as zero.

---

## Clock Distribution and Control

The following figure shows how the oscillator inputs are prescaled and selected to provide the internal clock signals.



Clock Distribution

## OSC and XOSC

OSC is the main oscillator. OSC divided by 2, 4, or 64 can be selected as the clock source for the system clock. OSC divided by 128 can be selected as the clock source for the time base. Clearing the FOSCE bit in the miscellaneous register turns off OSC and presets the 7-bit system clock prescaler and the 6-bit power-on reset (POR) counter to \$0078. Setting FOSCE starts OSC again, and the POR counter overflow sets the FTUP bit after 8072 counts.

XOSC is the alternate oscillator. It runs as long as the MCU is powered up. XOSC can be selected as the clock source for the system clock and the time base.

## System Clock

A 7-bit divider prescales the OSC clock to produce the system clock. The system clock drives the CPU, the serial peripheral interface subsystem, and the two timers. The SYS1 and SYS0 bits in the miscellaneous register select either OSC divided by 2, 4, or 64 or XOSC divided by 2 to drive the system clock. Reset automatically selects OSC divided by 64.

---

### MR — Miscellaneous Register

**\$003E**

	Bit 7	6	5	4	3	2	1	Bit 0
	FTUP	STUP	0	0	SYS1	SYS0	FOSCE	OPTM
RESET:	U	U	0	0	1	0	1	0

(U = UNAFFECTED)

#### FTUP — OSC Time Up

This read-only flag is cleared by power-on reset or by shutting down the OSC oscillator (clearing the FOSCE bit). If the FOSCE bit is set, the POR counter overflow sets the FTUP bit again after 8072 counts.

- 1 = OSC clock available for the system clock
- 0 = Power-on reset or OSC disabled (FOSCE = 0)

#### STUP — XOSC Time Up

This read-only flag is cleared at power-on and is set after 8072 counts of the time base prescaler. The STUP bit then stays set until power-down.

- 1 = XOSC clock available for the system clock
- 0 = XOSC not stabilized or no connection on XOSC1 and XOSC2 pins

#### SYS1–0 — System Clock

These read/write bits select one of four system clock sources, as the following table shows.

### System Clock Frequency Selection

SYS1:SYS0	Divide Ratio	System Clock Frequency		
		OSC = 4.0 MHz	OSC = 4.1943 MHz	XOSC = 32.768 kHz
00	OSC + 2	2.0 MHz	2.0972 MHz	—
01	OSC + 4	1.0 MHz	1.0486 MHz	—
10	OSC + 64	62.5 kHz	65.536 kHz	—
11	XOSC + 2	—	—	16.384 kHz

#### FOSCE — OSC Enable

This read/write bit turns on OSC. After 8072 clocks, an overflow from the POR counter sets the FTUP bit in the miscellaneous register. When OSC turns off, the FTUP bit is cleared, and the 7-bit OSC divider and the 6-bit POR counter are preset to \$0078.

- 1 = OSC enabled
- 0 = OSC disabled

#### OPTM — Option Map

This read/write bit selects the option memory map.

- 1 = Option memory map selected
- 0 = Option memory map not selected

#### Time Base

The time base produces the LCD clock and the real-time interrupt clock. OSC divided by 128 or XOSC can drive the time base. At power-on, the 14-bit time base prescaler is initialized to \$0078. After 8072 counts it sets the STUP bit in the miscellaneous register.

#### TBCR1 — Time Base Control Register 1

**\$0010**

	Bit 7	6	5	4	3	2	1	Bit 0
	TBCLK	0	LCLK	0	0	0	T2R1	T2R0
RESET:	0	0	0	0	0	0	0	0

#### TBCLK — Time Base Clock

This read/write bit selects the clock source for the time base.

- 1 = OSC divided by 128
- 0 = XOSC

### LCLK — LCD Clock

This read/write bit selects the clock source for the LCD driver.

1 = OSC divided by 16384 or XOSC divided by 128

0 = OSC divided by 8192 or XOSC divided by 64

The following table shows how TBCLK and LCLK control the LCD clock.

#### LCD Clock Frequency Selection

TBCLK	LCLK	Divide Ratio	LCD Clock Frequency (Hz)		
			OSC = 4.0 MHz	OSC = 4.1943 MHz	XOSC = 32.768 kHz
1	0	OSC + 2 <sup>13</sup>	488	512	—
1	1	OSC + 2 <sup>14</sup>	244	256	—
0	0	XOSC + 2 <sup>6</sup>	—	—	512
0	1	XOSC + 2 <sup>7</sup>	—	—	256

### T2R1–0 — Timer 2 Rate Bits 1–0

These read/write bits control the timer prescaler as the following table shows. They select the system clock divided by 1, 4, 32, or 256 to produce CLK2. CLK2 may be used to gate the EVI input to timer 2. (See Timers.)

#### Timer Prescaler Control

T2R1:0	CLK2 Frequency
00	System Clock + 1
01	System Clock + 4
10	System Clock + 32
11	System Clock + 256

### TBCR2 — Time Base Control Register 2

**\$0011**

	Bit 7	6	5	4	3	2	1	Bit 0
	RTIF	RTIE	RTR1	RTR0	RRTIF	0	0	0
RESET:	0	0	1	1	0	0	0	0

### RTIF — Real-Time Interrupt Flag

This read-only flag is set by every timeout of the real-time clock.

1 = RTI request

0 = No RTI request

**RTIE — Real-Time Interrupt Enable**

This read/write bit enables real-time interrupts.

1 = RTI interrupts enabled

0 = RTI interrupts disabled

**RTR1–0 — Real-Time Interrupt Rate**

These read/write bits select one of four real-time interrupt rates, as the following table shows.

**Real-Time Interrupt Clock Frequency Selection**

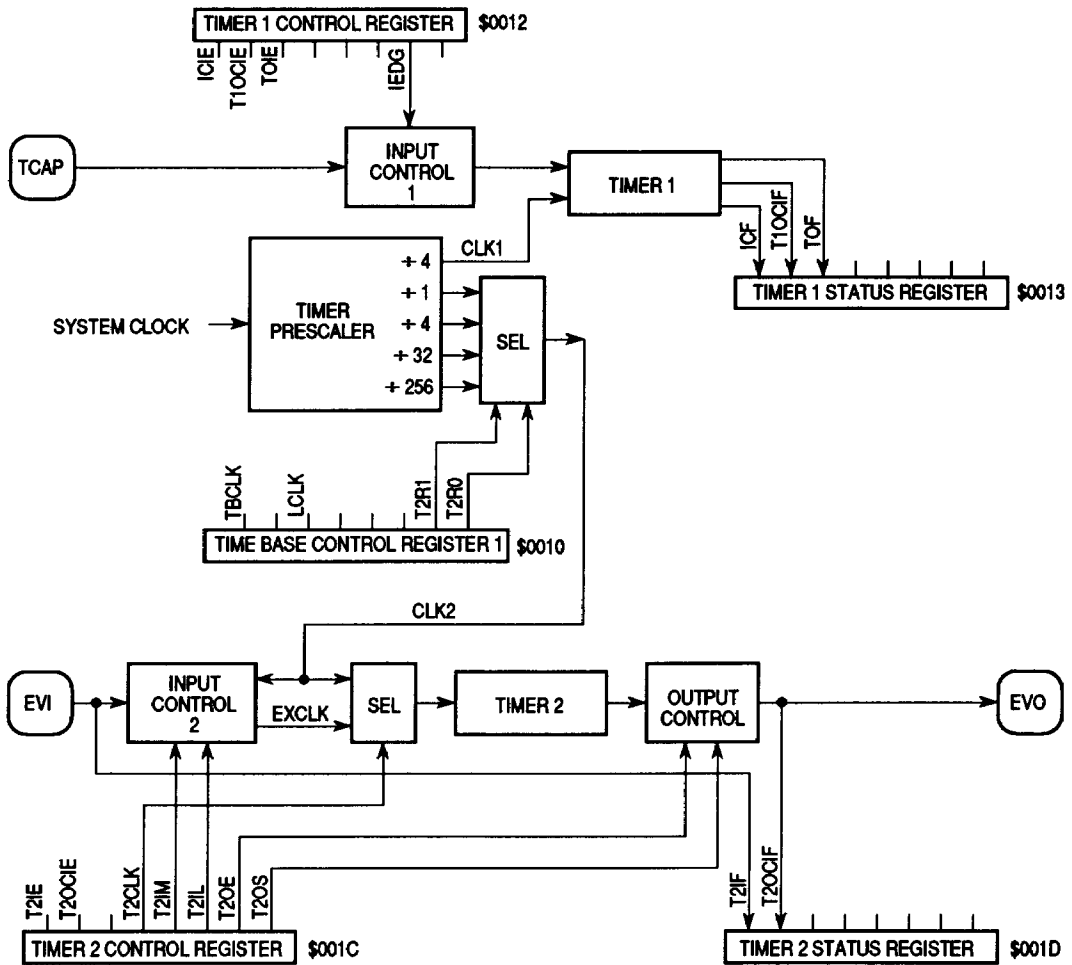
TBCLK	RTR1:0	Divide Ratio	Real-Time Interrupt Frequency (Hz)		
			OSC = 4.0 MHz	OSC = 4.1943 MHz	XOSC = 32.768 kHz
1	00	OSC + 2 <sup>14</sup>	244	256	—
1	01	OSC + 2 <sup>19</sup>	7.63	8.00	—
1	10	OSC + 2 <sup>20</sup>	3.81	4.00	—
1	11	OSC + 2 <sup>21</sup>	1.91	2.00	—
0	00	XOSC + 2 <sup>7</sup>	—	—	256
0	01	XOSC + 2 <sup>12</sup>	—	—	8.00
0	10	XOSC + 2 <sup>13</sup>	—	—	4.00
0	11	XOSC + 2 <sup>14</sup>	—	—	2.00

**RRTIF — Reset Real-Time Interrupt Flag**

RRTIF is a write-only bit and always reads as zero. Setting the RRTIF bit clears the RTIF bit. Writing zero to RRTIF has no effect.

## Timers

The two MC68HC705L5 timers provide two timer input pins and one timer output pin. The following figure shows the structure of the timer system.



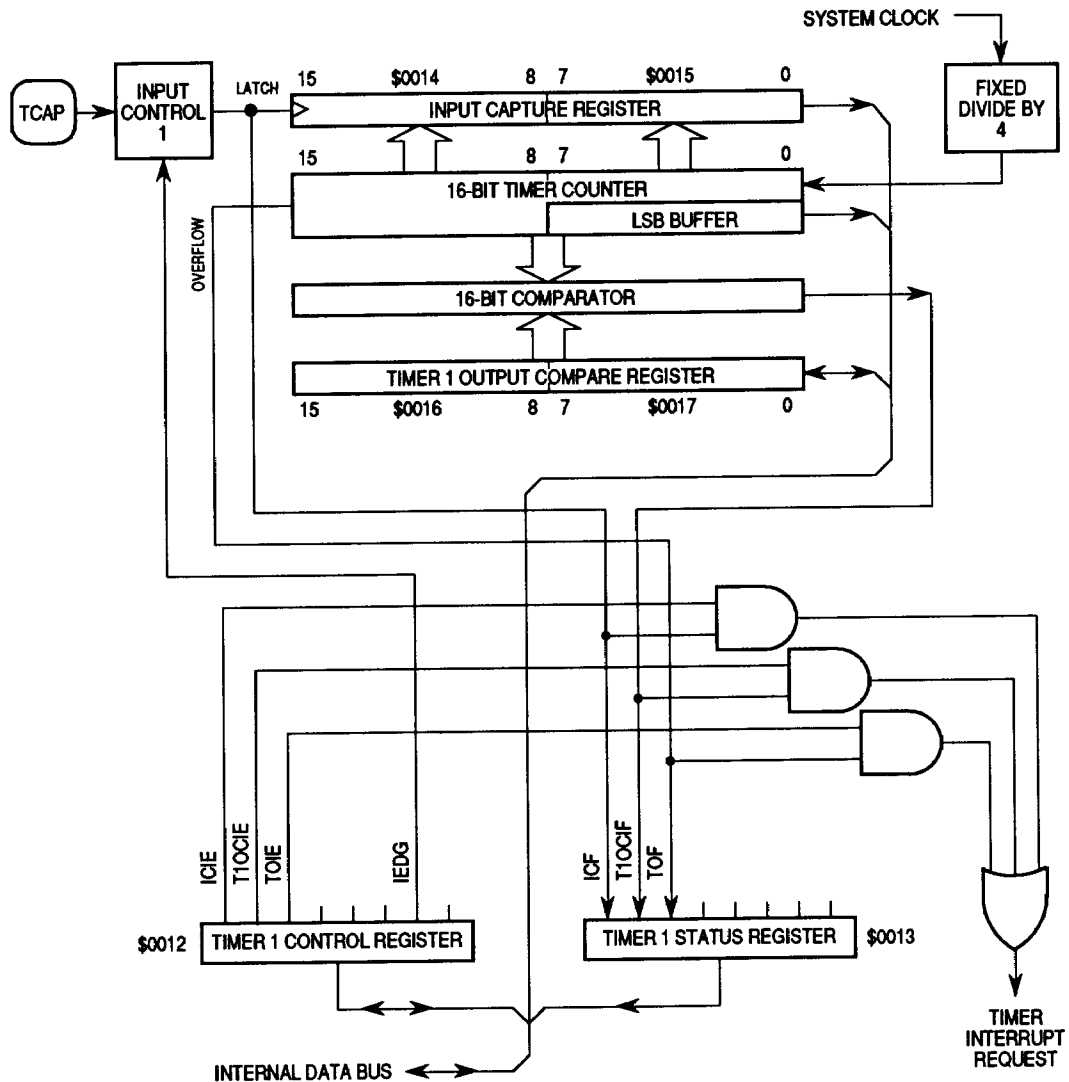
Timer 1 and Timer 2



## Timer 1

Timer 1 is a 16-bit free-running up counter with 16-bit input capture and 16-bit output compare functions. The input capture function latches the counter value and generates an interrupt request when a selected edge (either rising or falling) occurs on the TCAP input pin. Software can later read this value to determine when the edge occurred.

The output compare function generates an interrupt request when the counter value matches the value programmed in the timer 1 output compare register.



Timer 1

---

**T1CR — Timer 1 Control Register****\$0012**

	Bit 7	6	5	4	3	2	1	Bit 0
	ICIE	T1OCIE	TOIE	0	0	0	IEDG	0
RESET:	0	0	0	0	0	0	U	0

**ICIE — Input Capture Interrupt Enable**

This read/write bit enables timer 1 input capture interrupts.

1 = Timer 1 input capture interrupts enabled

0 = Timer 1 input capture interrupts disabled

**T1OCIE — Timer 1 Output Compare Interrupt Enable**

This read/write bit enables timer 1 output compare interrupts.

1 = Timer 1 output compare interrupts enabled

0 = Timer 1 output compare interrupts disabled

**TOIE — Overflow Interrupt Enable**

This read/write bit enables timer 1 overflow interrupts.

1 = Timer 1 overflow interrupts enabled

0 = Timer 1 overflow interrupts disabled

**IEDG — Input Edge**

This read/write bit determines whether a rising or a falling edge on the TCAP pin causes a transfer of the counter contents to the input capture register.

1 = Rising edge transfers counter contents

0 = Falling edge transfers counter contents

---

**T1SR — Timer 1 Status Register****\$0013**

	Bit 7	6	5	4	3	2	1	Bit 0
	ICF	T1OCIF	TOF	0	0	0	0	0
RESET:	0	0	0	0	0	0	0	0

**ICF — Input Capture Flag**

This read-only bit is set when a selected edge occurs on the TCAP pin. Clear the ICF bit by reading the timer status register with ICF set, and then reading the low byte of the input capture register.

**T1OCIF — Timer 1 Output Compare Interrupt Flag**

This read-only bit is set when the counter value matches the contents of the timer 1 output compare register. Clear the T1OCIF bit by reading the timer 1 status register with T1OCIF set, and then reading the low byte of the timer 1 output compare register.

### TOF — Timer Overflow Flag

This read-only bit is set when the counter rolls over from \$FFFF to \$0000. Clear the TOF bit by reading the timer 1 status register with TOF set, and then reading the low byte of the timer 1 counter register.

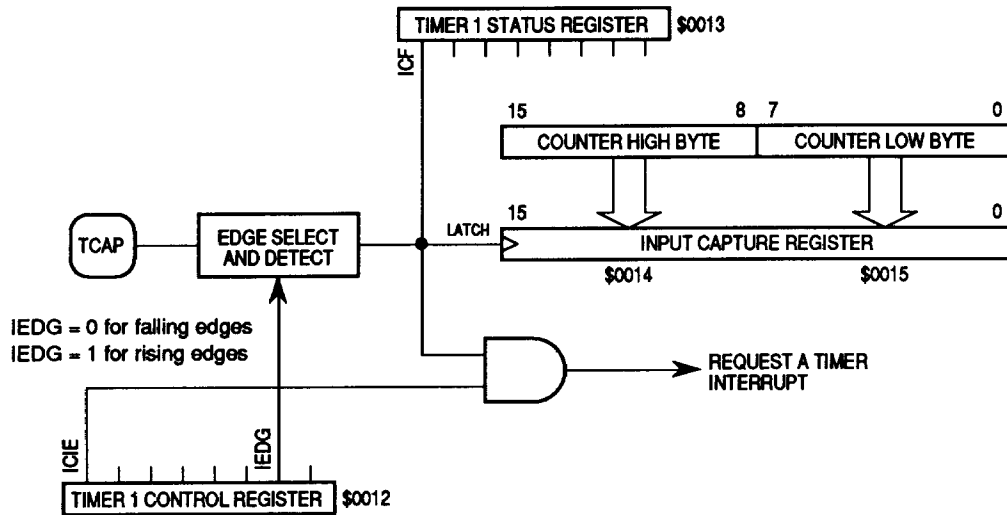
### ICR — Input Capture Register

**\$0014–\$0015**

\$0014	Bit 15	14	13	12	11	10	9	Bit 8	ICR (HIGH)
\$0015	Bit 7	6	5	4	3	2	1	Bit 0	ICR (LOW)

Reset does not affect the input capture register.

When a selected edge occurs on the TCAP pin, the current value of the counter is latched into the input capture register. Reading the high byte of the input capture register inhibits further captures until the low byte is read. Writing to the input capture register has no effect.



**Input Capture Operation**

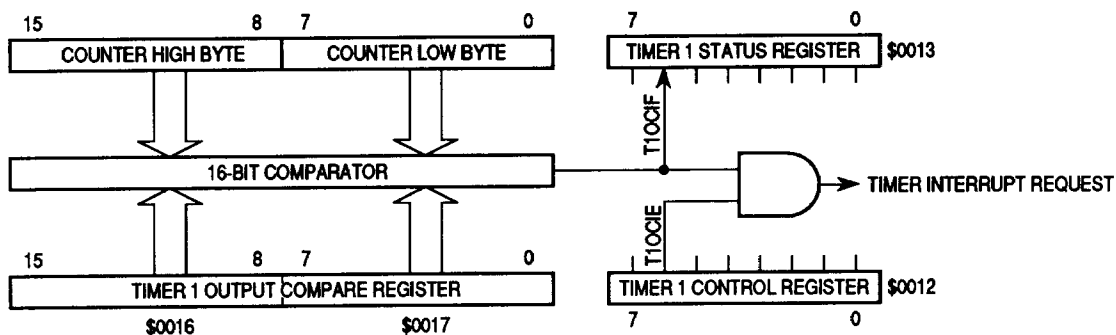
**T1OCR — Timer 1 Output Compare Register**

**\$0016–\$0017**

\$0016	Bit 15	14	13	12	11	10	9	Bit 8	T1OCR (HIGH)
\$0017	Bit 7	6	5	4	3	2	1	Bit 0	T1OCR (LOW)

Reset does not change the output compare register.

When the value of the counter matches the value in the timer 1 output compare register, the T1OCIF bit in the timer 1 status register is set. If the T1OCIE bit in the timer 1 control register is also set, an interrupt is generated. Writing to the high byte of the timer 1 output compare register inhibits timer compares until the low byte is written.



**Output Compare Operation**

**T1CNTR — Timer 1 Counter Register**

**\$0018–\$0019**

\$0018	Bit 15	14	13	12	11	10	9	Bit 8	T1CNTR (HIGH)
\$0019	Bit 7	6	5	4	3	2	1	Bit 0	T1CNTR (LOW)

On reset, the timer 1 counter register is preset to \$FFFC.

The timer 1 counter register contains the current counter value. Reading the high byte causes the low byte to be latched into a buffer. Reading the low byte after reading the timer 1 status register clears the TOF bit. Writing to the timer 1 counter register has no effect.

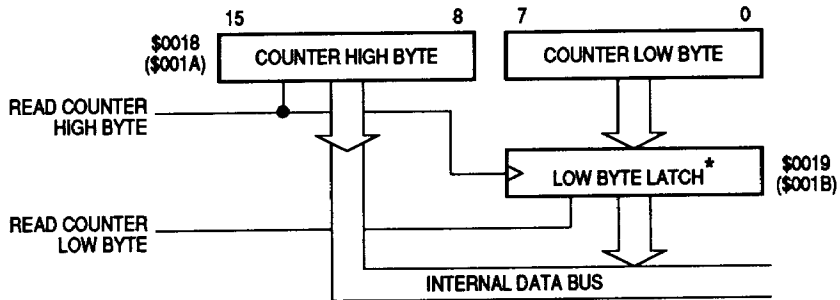
**T1ALTCNTR — Timer 1 Alternate Counter Register**

**\$001A–\$001B**

\$001A	Bit 15	14	13	12	11	10	9	Bit 8	T1ALTCNTR (HIGH)
\$001B	Bit 7	6	5	4	3	2	1	Bit 0	T1ALTCNTR (LOW)

On reset, the timer 1 alternate counter register is preset to \$FFFC.

The timer 1 alternate counter register contains the current counter value. Reading the high byte causes the low byte to be latched into a buffer. Reading the timer 1 alternate counter register does not affect the TOF bit. Writing to this register has no effect.



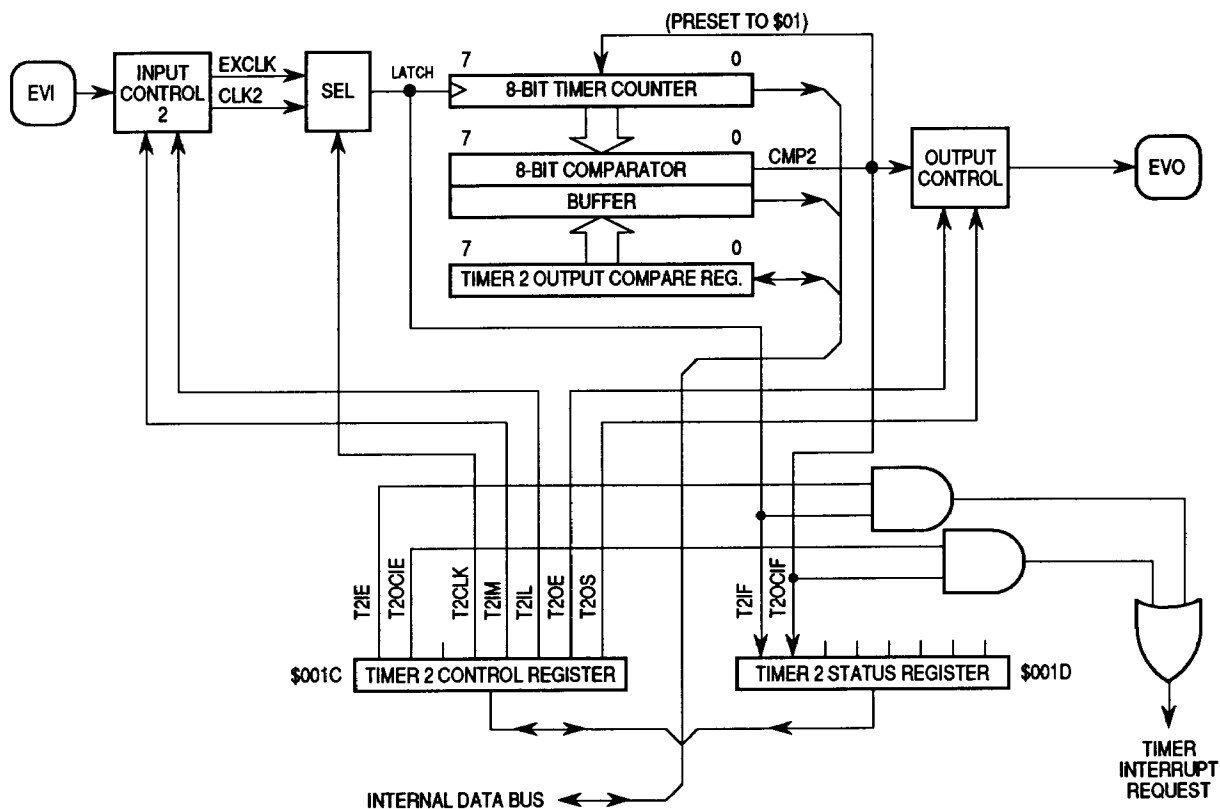
\* The low byte latch is normally transparent. It latches when the high byte of the counter is read and becomes transparent again when the low byte of the counter is read.

**16-Bit Counter Reads**

## Timer 2

Timer 2 is an 8-bit up counter with an 8-bit output compare function, an event input pin, and an event output pin. The output compare function presets the timer 2 counter to \$01 and generates an interrupt request when the counter reaches the value programmed into the output compare register.

Either the EVI input or the prescaled system clock (CLK2) can drive timer 2. In event mode, the EVI input drives timer 2 directly. In gated mode, the EVI input gated by CLK2 drives timer 2.



Timer 2

**T2CR — Timer 2 Control Register****\$001C**

Bit 7	6	5	4	3	2	1	Bit 0
T2IE	T2OCIE	0	T2CLK	T2IM	T2IL	T2OE	T2OS

RESET:           0           0           0           0           0           0           0           0

**T2IE — Timer 2 Interrupt Enable**

This read/write bit enables timer 2 interrupts when the T2IF bit in the timer 2 status register is set.

- 1 = Timer 2 interrupts enabled
- 0 = Timer 2 interrupts disabled

**T2OCIE — Timer 2 Output Compare Interrupt Enable**

This read/write bit enables timer 2 interrupts when the T2OCIF bit in the timer 2 status register is set.

- 1 = Timer 2 Output compare interrupts enabled
- 0 = Timer 2 Output compare interrupts disabled

**T2CLK — Timer 2 clock**

This read/write bit selects the clock source for timer 2.

- 1 = EXCLK from EVI is selected
- 0 = CLK2 from timer prescaler is selected

**T2IM — Timer 2 Input Mode**

This read/write bit selects prescaled system clock (CLK2) gating of the EVI input.

- 1 = Timer 2 driven by CLK2-gated EVI input (gated mode)
- 0 = Timer 2 driven by EVI input directly (event mode)

**T2IL — Timer 2 Input Level**

This read/write bit selects the active edge (rising or falling) of EVI in event mode, and the gate enable level of EVI in gated mode.

- 1 = Rising edge on EVI increments counter (event mode)  
High level on EVI enables counting (gated mode)
- 0 = Falling edge on EVI increments counter (event mode)  
Low level on EVI enables counting (gated mode)

**EVI Mode Selection**

T2IM:T2IL	Mode	Action on Counter
0 0	Event	EVI Falling Edge Increments Counter
0 1	Event	EVI Rising Edge Increments Counter
1 0	Gated	EVI Low Level Enables Counting
1 1	Gated	EVI High Level Enables Counting

### T2OE — Timer 2 Output Enable

This read/write bit determines whether the PC5/EVO pin is a bidirectional I/O pin (PC5) or the timer 2 output pin (EVO). Changing the T2OE bit does not change the PC5/EVO pin function immediately. A synchronization delay occurs until a rising or falling edge occurs on the PC5/EVO pin as selected by the T2OS bit.

- 1 = EVO output enabled
- 0 = PC5 bidirectional I/O enabled

### T2OS — Timer 2 Output Synchronize

This read/write bit selects either a falling or a rising edge on the PC5/EVO pin to change the function of the PC5/EVO pin.

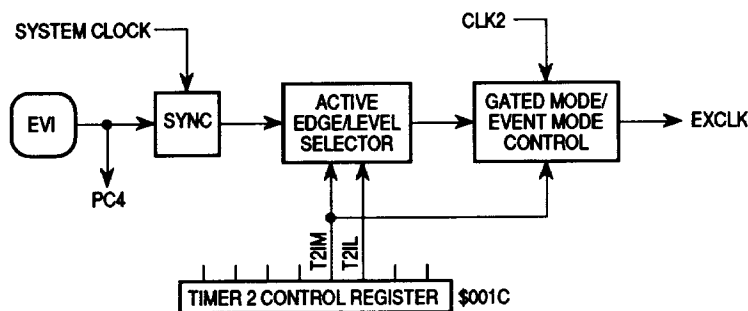
- 1 = Rising edge on PC5/EVO pin changes pin function
- 0 = Falling edge on PC5/EVO pin changes pin function

---

The timer 2 input control circuit synchronizes the EVI input with the falling edge of the system clock. The minimum external clock pulse width should be larger than one system clock pulse width.

The T2IM bit determines whether the EVI pin drives timer 2 directly (event mode) or is gated by CLK2 (gated mode). In event mode the T2IL bit determines which EVI edge increments timer 2. In gated mode the T2IL bit determines which EVI level enables timer 2 to count.

To avoid an illegal timer 2 count, software should preset the timer 2 counter register after changing the T2IM bit.

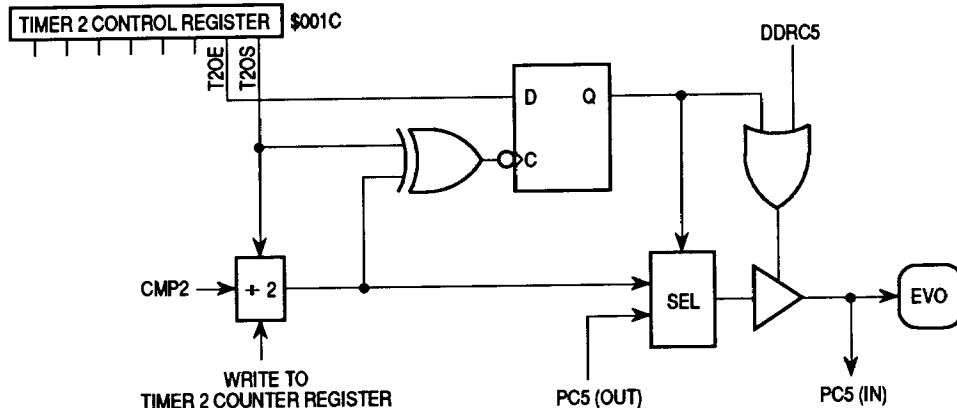


**EVI Input Control**



The timer 2 output control circuit enables and synchronizes the EVO output. A divide by two prescaler divides the timer 2 output compare signal, CMP2, to produce a 50% duty output. The prescaler is initialized to the state of the T2OS bit in the timer 2 control register when the CPU writes to the timer 2 counter register.

The T2OE bit enables the EVO. To avoid an incomplete EVO pulse, T2OE must be changed synchronously with the EVO signal. The T2OS bit selects a synchronizing EVO edge.



**EVO Output Control**

**T2SR — Timer 2 Status Register**

**\$001D**

Bit 7	6	5	4	3	2	1	Bit 0
T2IF	T2OCIF	0	0	RT2IF	RT2OCIF	0	0

RESET:

0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---

**T2IF — Timer 2 Input Flag**

In event mode this read-only bit is set when the event edge occurs on the EVI pin. In gated mode the T2IF bit is set when the trailing edge of the gating signal occurs on EVI.

**T2OCIF — Timer 2 Output Compare Interrupt Flag**

This read-only bit is set when a timer 2 output compare occurs.

**RT2IF — Reset Timer 2 Input Flag**

This write-only bit always reads as zero. Setting RT2IF clears the T2IF bit. Clearing RT2IF has no effect.

**RT2OCIF — Reset Timer 2 Output Compare Interrupt Flag**

This write-only bit always reads as zero. Setting RT2OCIF clears the T2OCIF flag. Clearing RT2OCIF has no effect.

---

**T2OCR — Timer 2 Output Compare Register****\$001E**

	Bit 7	6	5	4	3	2	1	Bit 0
RESET:	1	1	1	1	1	1	1	1

This read/write register contains a user-supplied value that is continuously compared with the value in the timer 2 counter. When the two values match, a comparator presets the timer 2 counter to \$01, and sets the T2OCIF bit in the timer 2 status register.

---

**T2CNTR — Timer 2 Counter Register****\$001F**

	Bit 7	6	5	4	3	2	1	Bit 0
RESET:	0	0	0	0	0	0	0	1

This read/write register contains the current timer 2 counter value. Writing any value to the timer 2 counter register presets the counter to \$01.

---

## Liquid Crystal Display Driver

The features of the programmable liquid crystal display (LCD) driver subsystem include the following:

- Four backplane (BP) output pins and up to 39 frontplane (FP) output pins
- 20 data registers
- Three power supply pins

The following figures detail the LCD control register and the LCD data registers.

### LCDCR — LCD Control Register

**\$0020**

	Bit 7	6	5	4	3	2	1	Bit 0
	LCDE	DUTY1	DUTY0	0	PEH	PEL	PDH	0
RESET:	0	0	0	0	0	0	0	0

#### LCDE — LCD Enable

This read/write bit enables the LCD driver outputs. The LCDE bit does not affect Port D and port E pins that are programmed as general-purpose outputs.

- 1 = BP and FP pins enabled to carry LCD waveforms
- 0 = BP and FP outputs disabled; VDD appears on all implemented BP and FP pins

#### DUTY1–DUTY0 — Duty Select

These read/write bits determine how many BP outputs are used, as the following table shows. Unused BP pins are general-purpose port D pins.

**LCD Duty Selection**

Duty	LCD Control	Pin Function			
	DUTY[1:0]	BP3/PD3	BP2/PD2	BP1/PD1	BP0
1/1	01	PD3	PD2	PD1	BP0
1/2	10	PD3	PD2	BP1	BP0
1/3	11	PD3	BP2	BP1	BP0
1/4	00	BP3	BP2	BP1	BP0

#### PEH — Port E High

This read/write bit enables the upper four bits of port E as general-purpose outputs.

- 1 = PE7–PE4 selected
- 0 = FP27–FP30 selected

#### PEL — Port E Low

This read/write bit enables the lower four bits of port E as general-purpose outputs.

- 1 = PE3–PE0 selected
- 0 = FP31–FP34 selected

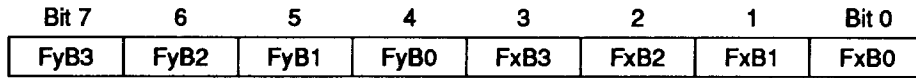
**PDH — Port D High**

This read/write bit enables the upper four bits of port D as general-purpose outputs.

- 1 = PD7–PD4 selected
- 0 = FP35–FP38 selected

**LCDDR1–LCDDR20 — LCD Data Registers 1–20**

**\$0021–\$0034**



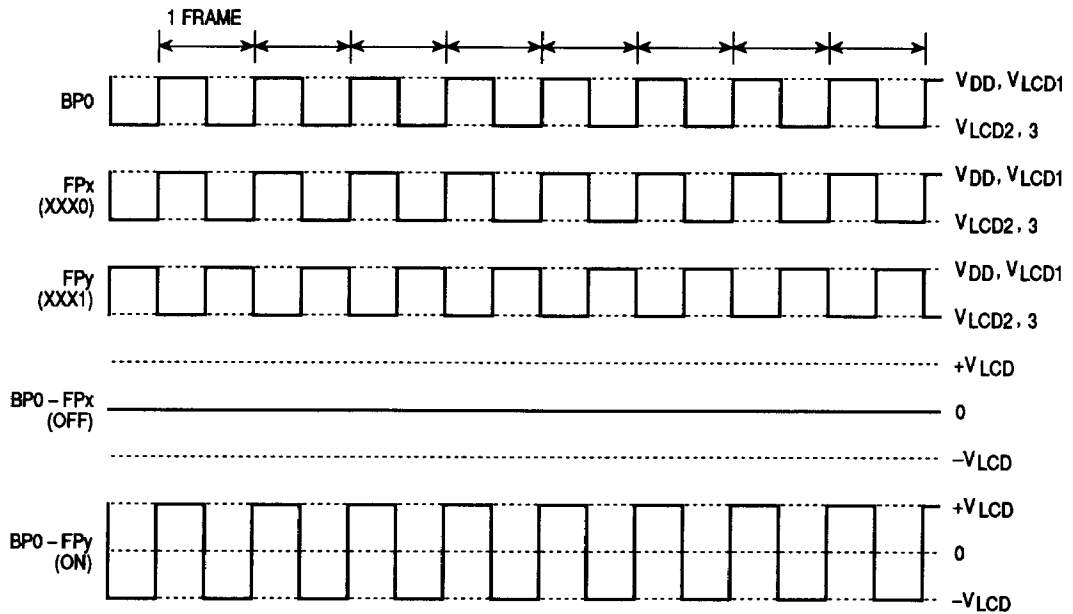
RESET: UNCHANGED BY RESET

Each nibble in an LCD data register controls the waveform of a frontplane driver.

**BP3–BP0 — Backplane 3–0**

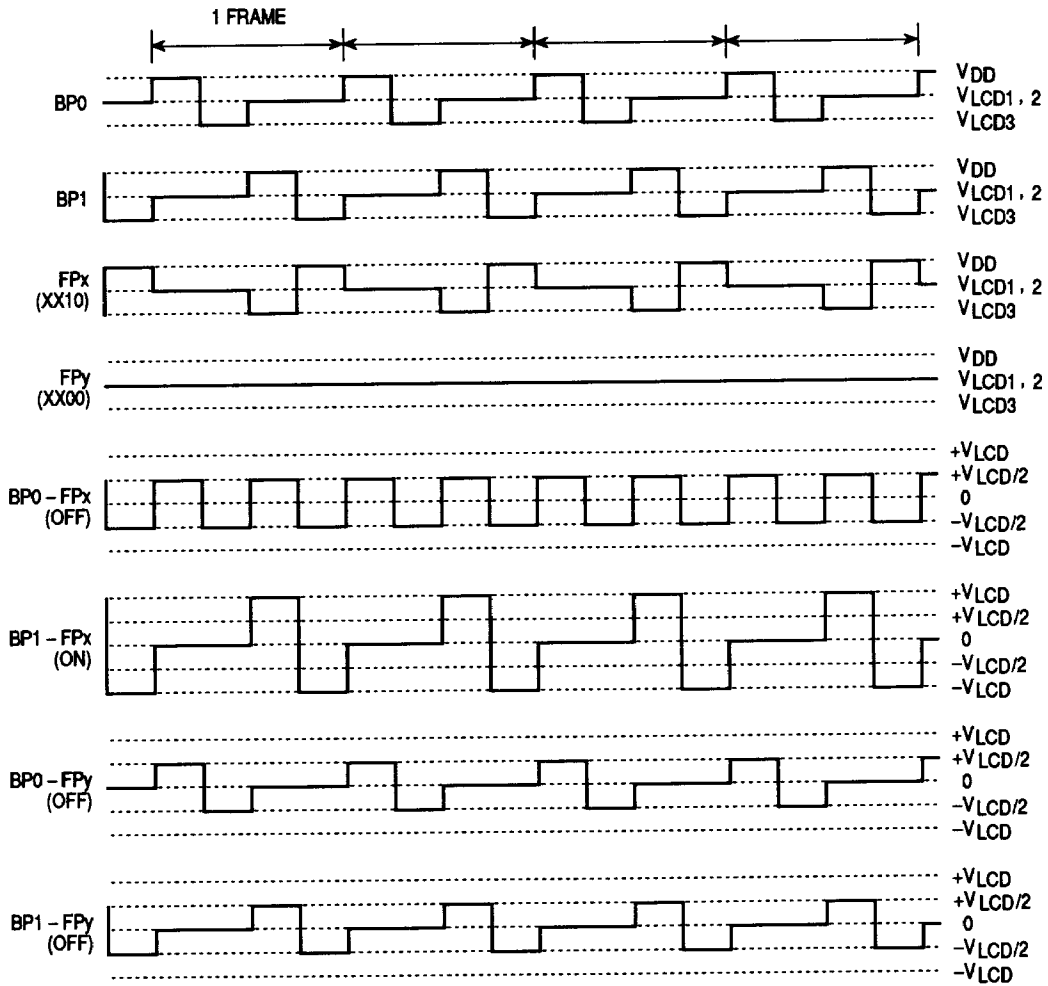
- 1 = Backplane output selected
- 0 = Backplane output deselected

The following figures show example LCD driver waveforms.



Duty = 1/1 (Static)  
 Bias = None ( $V_{LCD1} = V_{DD}$ ,  $V_{LCD2} = V_{LCD3} = V_{DD} - V_{LCD}$ )

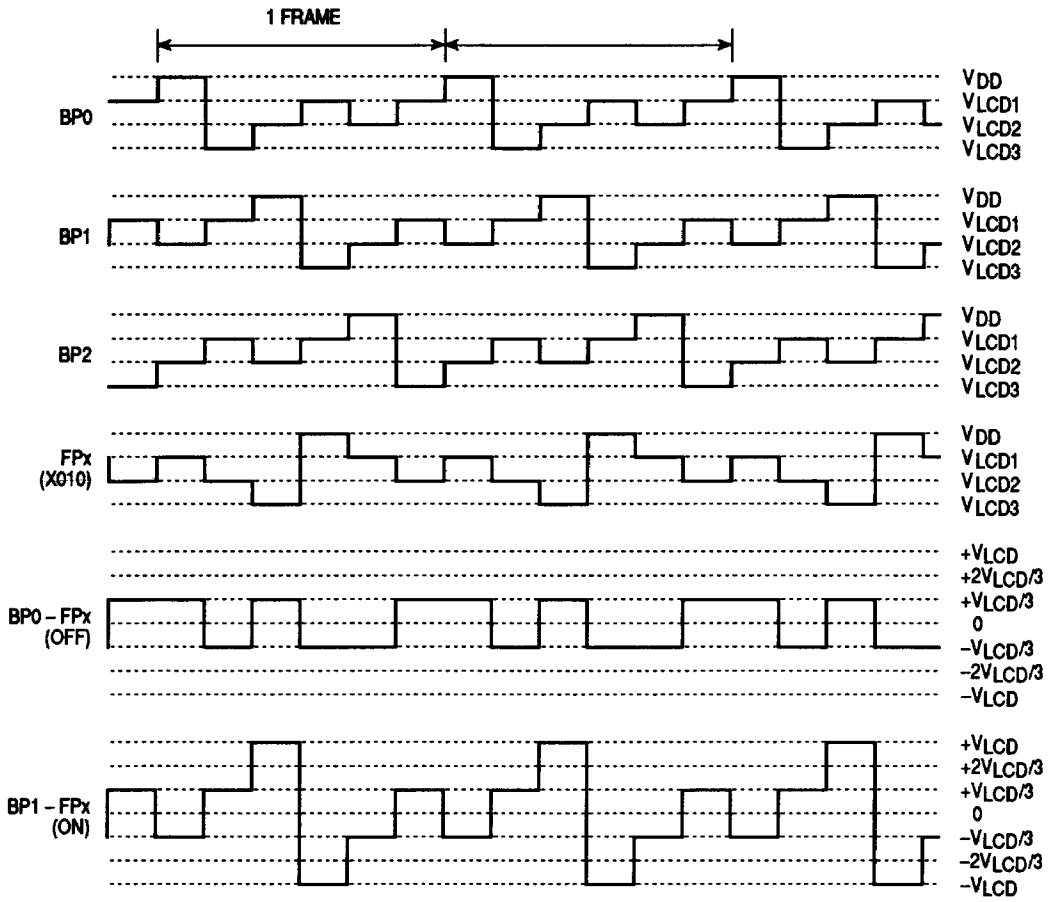
**1/1 Duty and 1/1 Bias**



Duty = 1/2

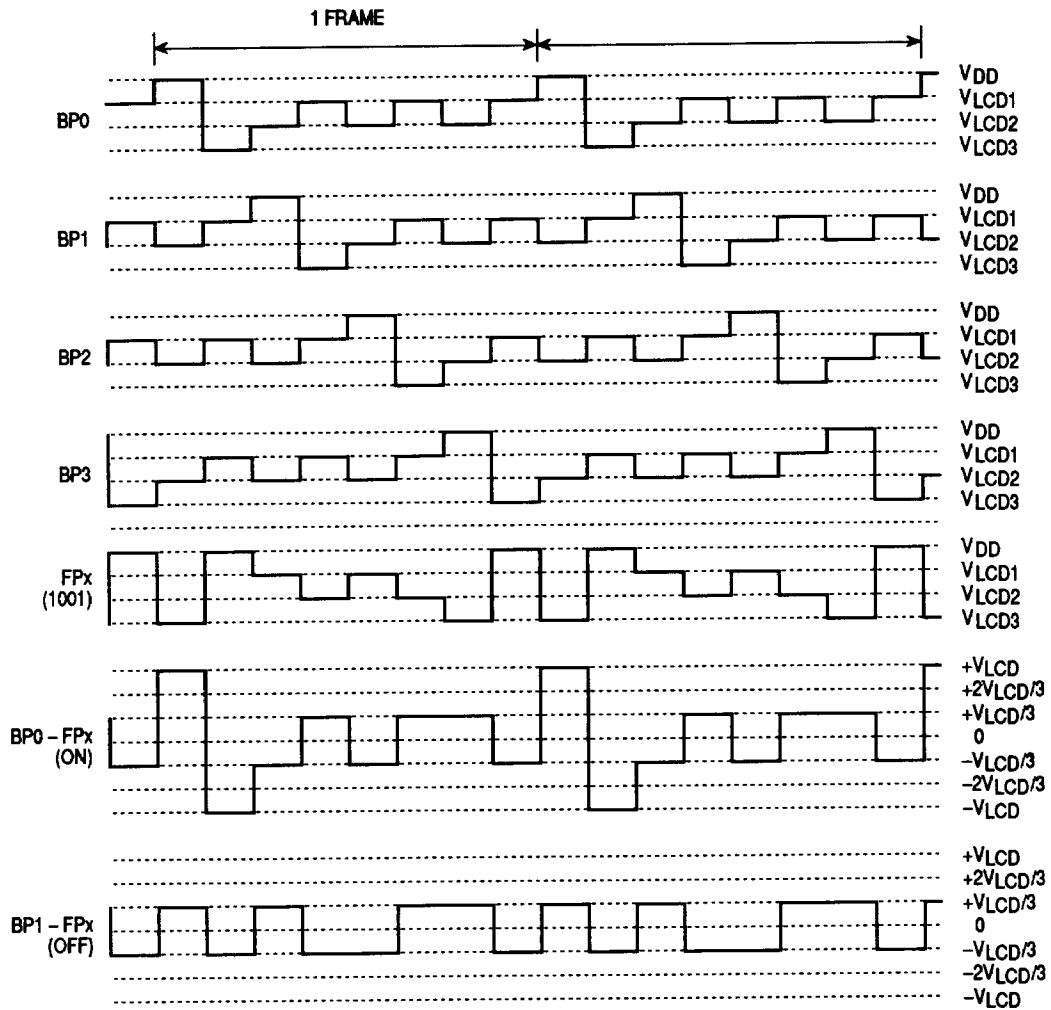
Bias = 1/2 ( $V_{LCD1} = V_{LCD2} = V_{DD} - V_{LCD}/2$ ,  $V_{LCD3} = V_{DD} - V_{LCD}$ )

### 1/2 Duty and 1/2 Bias



Duty = 1/3  
 Bias = 1/3 ( $V_{LCD1} = V_{DD} - V_{LCD}/3$ ,  $V_{LCD2} = V_{DD} - 2V_{LCD}/3$ ,  $V_{LCD3} = V_{DD} - V_{LCD}$ )

**1/3 Duty and 1/3 Bias**



Duty = 1/4

Bias = 1/3 ( $V_{LCD1} = V_{DD} - V_{LCD}/3$ ,  $V_{LCD2} = V_{DD} - 2V_{LCD}/3$ ,  $V_{LCD3} = V_{DD} - V_{LCD}$ )

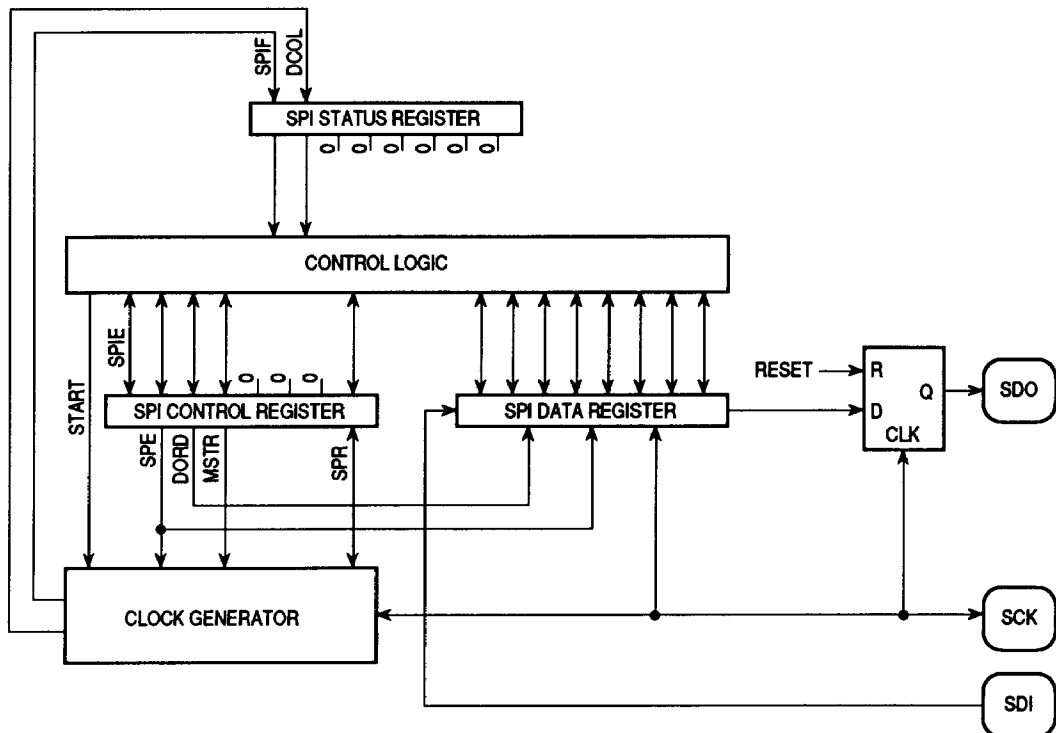
### 1/4 Duty and 1/3 Bias

## Serial Peripheral Interface

The serial peripheral interface (SPI) allows high-speed synchronous serial data transfer between the MC68HC705L5 and peripheral devices or between several MC68HC705L5s. The SPI can be used with simple shift registers to expand the number of parallel I/O pins controlled by the MCU. A large number of more powerful peripherals such as A/D converters and real-time clocks are also compatible with this interface. SPI features include the following:

- Full Duplex, Three Wire Synchronous Data Transfers
- Master or Slave Operation
- 1.05-MHz (Maximum) Bit Frequency
- LSB First or MSB First Data Transfer
- Two Programmable Bit Rates
- End of Transmission Interrupt Flag
- Data Collision Flag Protection
- Wakeup from STOP Mode (Slave Mode Only)

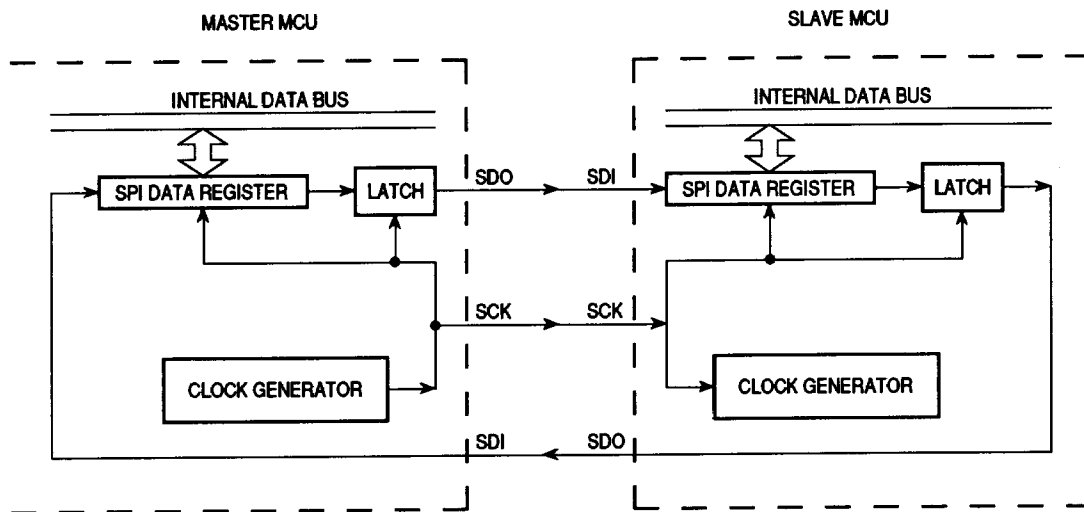
The following figure shows the structure of the SPI.



SPI Structure



The following figure shows the interconnection between master and slave MCUs. In master mode the SCK pin functions as the clock output. In slave mode the SCK pin is an input. Writing to the SPI data register of the master MCU starts the master clock generator, and the data written shifts out the master SDO pin and into the slave SDI pin. After shifting one byte, the master clock generator stops, setting the end of transmission flag (SPIF). If the SPI interrupt enable bit (SPIE) is set, an interrupt is requested.



**SPI Master-Slave Interconnection**

**SPCR — SPI Control Register**

**\$000A**

Bit 7	6	5	4	3	2	1	Bit 0
SPIE	SPE	DORD	MSTR	0	0	0	SPR

RESET:      0      0      0      0      0      0      0      0

**SPIE — SPI Interrupt Enable**

This read/write bit enables SPI interrupts.

- 1 = SPI interrupts enabled
- 0 = SPI interrupts disabled

**SPE — SPI Enable**

This read/write bit enables the SPI system and connects SDI, SDO, and SCK to pins PC0, PC1, and PC2.

- 1 = SPI enabled
- 0 = SPI disabled





The MCU enters bootloader mode when a rising edge occurs on the RESET pin while a logical one is on the PC6/IRQ2 pin, and  $1.8 \times V_{DD}$  is on the IRQ1 pin.

### **Parallel Bootloader Operation**

The bootloader program uses ports C and E to address the external EPROM data and port B to read the data. The following sequence shows how the bootloader downloads a parallel byte from an external EPROM to the on-chip EPROM or RAM.

1. The bootloader puts the address on the port C and port E pins.
2. The bootloader resets the Data Request line (PA6).
3. The bootloader reads the data on port B if the Data Valid line (PC7) is high. In a master/multiple-slave system, the master controller can monitor the Data Request lines of all the slave MCUs. It can then send a Data Valid signal to each slave MCU when all Data Request lines are low and valid data is ready. In a standalone system the Data Request line of the single MCU must be tied to  $V_{DD}$ .
4. The bootloader sets the Data Request line (PA6).

### **Serial Bootloader Operation**

The bootloader program uses ports C and E to address external EPROM data and the PA0 line to read the data. The following sequence shows how the bootloader downloads a serial byte from an external EPROM to the on-chip EPROM or RAM.

1. The bootloader puts the address on the port C and port E pins.
2. The bootloader resets the Data Request line (PA6).
3. The bootloader waits for the Start bit and a byte of data on the SCI RX line (PA0). In a master/multiple-slave system, the master controller can monitor the Data Request lines of all the slave MCUs. It can then send data to each slave MCU when all the Data Request lines are low. The bootloader does not use the Data Valid line in serial operations.
4. The bootloader checks for framing errors. If it finds a framing error, the program stops.
5. If no framing error is found, the bootloader sets the Data Request line.

## Bootloader Functions

The bootloader ROM can perform nine separate functions. When the MCU enters bootloader mode, the states of pins PA2–PA5 select the function to execute.

**Bootloader Functions**

Function	PA5:PA4:PA3:PA2	Action
0	0 0 0 0	Execute Program in RAM
8	1 0 0 0	Program EPROM and Verify (Parallel)
9	1 0 0 1	Program EPROM and Verify (Parallel)
A	1 0 1 0	Verify EPROM (Parallel)
B	1 0 1 1	Load Program into RAM (Parallel)
C	1 1 0 0	Dump EPROM Contents (Serial)
D	1 1 0 1	Modify Memory (Serial)
E	1 1 1 0	Verify EPROM (Serial)
F	1 1 1 1	Load Program into RAM (Serial)

### Function 0 — Execute program in RAM

Function 0 jumps to the first RAM address, \$0040, and executes the program that begins there. The program may have been previously loaded by Function B, Function D, or Function F. The last RAM address, \$013F, is reserved for a user interrupt jump offset. If the program in RAM generates an interrupt, the CPU adds the contents of address \$013F to \$0040 to determine the jump address.

### Function 8 — Program EPROM and verify (parallel)

Function 8 copies from an external EPROM to the on-chip EPROM and then automatically executes Function A. With a 4-MHz oscillator, the programming voltage,  $V_{pp}$ , turns on for 3 ms and turns off for 1 ms while programming each byte. The Program LED flashes 64 times during the copy and remains on while the bootloader executes Function A. When Function A is done, the Finish LED turns on.

### Function 9 — Program EPROM and verify (parallel)

Function 9 is the same as Function 8 except for the  $V_{pp}$  on-time. Before the bootloader starts programming, it reads the program time data at address \$0000 in the external EPROM. With a 4-MHz oscillator,  $V_{pp}$  turns on for approximately  $0.1 \text{ ms} \times$  the program time and turns off for 1 ms while programming each byte.

### Function A — Verify EPROM (parallel)

Function A compares the on-chip EPROM data with the data in an external EPROM. If it finds an error, the program stops and indicates the address of the error on port C and port E. If the data are the same, the Finish LED turns on.

### **Function B — Load program Into RAM (parallel)**

Function B copies from external EPROM to the on-chip RAM. Addresses \$00F2–\$00FF are reserved for bootloader stack. When the copy is complete, the Finish LED turns on and the bootloader jumps to address \$0040.

### **Function C — Dump EPROM contents (serial)**

Function C serially transmits the data from the on-chip EPROM addresses \$1000–\$3FFF but skips addresses \$3000–\$3FEF. When the transmission is complete, the Finish LED turns on and the bootloader stops.

### **Function D — Memory modify (serial)**

Function D is an EPROM editing routine that transmits serially in ASCII code to allow direct connection to a personal computer. The bootloader program does not decode the entire ASCII table. ASCII characters \$30–\$39 decode as 0–9; characters \$41–\$46 and \$61–\$66 decode as A–F, and character \$0D decodes as a carriage return (CR).

Function D presents the following prompt:

**AAAA DD**

AAAA is the current address and DD is the data at that address.

Use the following commands to change the address and the data:

1. CR — Increments the address
2. <dd> CR — Changes the data and increments the address
3. <aaaa> — Changes the address
4. CTRL-D — Displays the 256 bytes that follow the current address
5. CTRL-G — Executes program from the current address
6. CTRL-L — Load S-record, program EPROM, and ignore checksum

If only one character is entered before a carriage return, the character is ignored. If three characters are entered before a carriage return, the last character entered is ignored.

Function D reserves addresses \$00E0–\$00FF for the bootloader stack and workspace.

### **Function E — Verify EPROM (serial)**

Function E compares the on-chip EPROM data with the data on the PA0 input (SCI RX). If it finds an error, the program stops and indicates the address of the error on port C and port E. If the data are the same, the Finish LED turns on.

### **Function F — Load program Into RAM (serial)**

Function F is the same as Function B, except Function F receives the data on line PA0. For the bootloader stack \$00F2–\$00FF are reserved. When finished, LED turns on and bootloader stops.

If the bootloader detects a framing error, the program stops and the error address appears on the port C and port E pins. PA6 (DREQ) stays low.

## EPROM

The memory map includes 8208 bytes of EPROM. Addresses \$1000–\$2FFF contain 8192 bytes of EPROM. Sixteen EPROM bytes for user-defined interrupt and reset vectors occupy addresses \$3FF0–\$3FFF.

### EPROM Erasing

The erased state of an EPROM byte is \$FF. Erase the EPROM with high-density, 2537-angstrom ultraviolet light (UV) before programming it. The recommended dose is 15 Ws/cm<sup>2</sup>. Position the UV light source 1 in. (2.5 cm) from the EPROM. Do not use a shortwave filter.

### EPROM Programming

The V<sub>DD</sub> and V<sub>PP</sub> pins supply EPROM programming power. The EPROM programming register shown in the next figure contains the control bits for programming the EPROM.

---

#### EPROGR — EPROM Programming Register

\$003D

	Bit 7	6	5	4	3	2	1	Bit 0
	—	—	—	—	—	0	ELAT	EPGM
RESET:	0	0	0	0	0	0	0	0

#### ELAT — EPROM Bus Latch

This read/write bit configures the EPROM address bus and data bus for programming. When ELAT is set, writing to the EPROM latches the data and address. To set ELAT, the programming voltage must be present on the V<sub>pp</sub> pin.

- 1 = EPROM address bus and data bus configured for EPROM programming
- 0 = Buses configured for normal operation

#### EPGM — EPROM Programming

This read/write bit can be set only when ELAT = 1. EPGM turns on the EPROM programming power and resets automatically when ELAT is cleared.

- 1 = EPROM programming power switched on
- 0 = EPROM programming power switched off

---

Complete the following steps to program a byte of EPROM:

1. Set the ELAT bit.
2. Write to the EPROM address.
3. Set the EPGM bit to apply the programming voltage for the appropriate length of time.
4. Clear the ELAT and EPGM bits.