68HC05L16
68HC705L16

SPECIFICATION
(General Release)

CSIC System Design Group
Austin, Texas

TABLE OF CONTENTS

TABLE OF CONTENTS

Section **Title** **Page**

SECTION 4
RESET/INTERRUPT STRUCTURE

SECTION 5
LOW-POWER MODES

SECTION 6
PARALLEL I/O

SECTION 7
OSCILLATORS/CLOCK DISTRIBUTIONS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF FIGURES

LIST OF TABLES

SECTION 1
GENERAL DESCRIPTION

The MC68HC(7)05L16 is an 80-pin microcontroller unit (MCU) with highly sophisticated on-chip peripheral functions. The memory map includes 16K bytes of user ROM (MC68HC05L16) or EPROM (MC68HC705L16) and 512 bytes of RAM. The MCU has five parallel ports: A, B, C, D, and E. The MC68HC(7)05L16 includes a time base circuit, 8- and 16-bit timers, a computer operating properly (COP) watchdog timer, liquid crystal display (LCD) drivers, and a simple serial peripheral interface (SSPI).

1.1 Features

Features of the MC68HC(7)05L16 MCU include the following:

- Low Cost/HC05 Core
- 80-Pin Quad Flat Pack (QFP)
- 16,400 Bytes of Mask ROM or EPROM, Including 16 Bytes of User Vectors and 512 Bytes of On-Chip RAM
- 16 Bidirectional Input-Output (I/O) Lines
- Eight Input-Only Lines
- 15 Output-Only Lines, Including 8-Bit Key Wake-up Interrupts
- Pull-Up Resistors
- Open-Drain Output Port Options
- Two Interrupt Request (IRQ) Inputs
- 16-Bit Timer with Input Capture and Output Compare (Timer 1)
- 8-Bit Event Counter/Modulus Clock Divider (Timer 2)
- Simple Serial Peripheral Interface (SSPI)
- LCD Drivers (One to Four Backplane Drivers) x (27 to 39 Frontplane Drivers)

- On-Chip Time Base Circuits with COP Watchdog Timer and Time Base Interrupts
- Dual Oscillators and Selectable System Clock Frequency
- Power Saving STOP Mode/WAIT Mode

1.2 MCU Structure

Figure 1-1 shows the structure of the MC68HC(7)05L16 MCU.

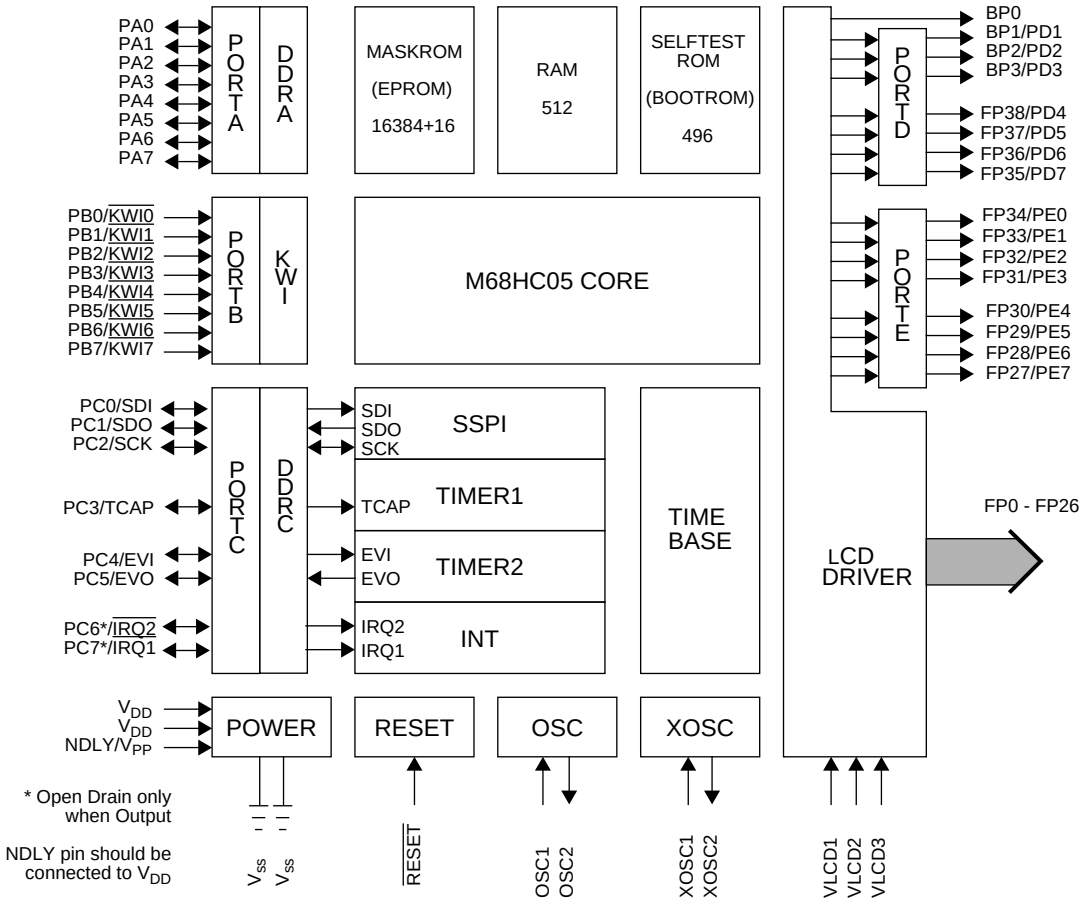


Figure 1-1. Block Diagram of the MC68HC(7)05L16

NOTE

A line over a signal name indicates an active low signal. For example, $\overline{\text{RESET}}$ is active low.

1.3 Mask Options

There are three mask options on the MC68HC05L16:

1. RSTR ($\overline{\text{RESET}}$ pin pull up resistor),
2. OSCR (OSC feedback resistor), and
3. XOSCR (XOSC feedback/damping resistor).

See **2.2.6 Mask Option Status Register (MOSR)**. MC68HC705L16 has no mask options.

1.4 Modes of Operation

The MC68HC05L16 has the following operating modes: single-chip mode (SCM) and self-check mode. In the MC68HC705L16, the self-check mode becomes bootstrap mode.

The single-chip mode, also called the user mode, allows maximum use of pins for on-chip peripheral functions.

The self-check capability of MC68HC05L16 provides an internal check to determine if the device is functional.

The bootstrap load mode is provided for EPROM programming, dumping EPROM contents, and loading programs into the internal RAM and executing them. This is a very versatile mode because there are essentially no limitations on the special purpose program that is boot-loaded into the internal RAM.

Table 1-1. Pin Configurations

PIN NBR	SCM, BOOT, SELF-CHECK	I/O
23	PA0	I/O
24	PA1	I/O
25	PA2	I/O
26	PA3	I/O
27	PA4	I/O
28	PA5	I/O
29	PA6	I/O
30	PA7	I/O
31	PB0/$\overline{\text{KWI0}}$	I
32	PB1/$\overline{\text{KWI1}}$	I
33	PB2/$\overline{\text{KWI2}}$	I
34	PB3/$\overline{\text{KWI3}}$	I
35	PB4/$\overline{\text{KWI4}}$	I
36	PB5/$\overline{\text{KWI5}}$	I
37	PB6/$\overline{\text{KWI6}}$	I
38	PB7/$\overline{\text{KWI7}}$	I
39	PC0/SDI	I/O
40	PC1/SDO	I/O
41	PC2/SCK	I/O
42	PC3/TCAP	I/O
43	PC4/EVI	I/O
44	PC5/$\overline{\text{EVO}}$	I/O
45	PC6*/$\overline{\text{IRQ2}}$	I
46	PC7*/$\overline{\text{IRQ1}}$	I
17	NDLY/V_{PP}	I
47	V_{DD}	I
1	V_{DD}	I
60	V_{SS}	O
16	V_{SS}	O
21	OSC1	I
22	OSC2	O
18	XOSC1	I
19	XOSC2	O
15	VLCD1	I
14	VLCD2	I
13	VLCD3	I
48	BP3/PD3	O
49	BP2/PD2	O
50	BP1/PD1	O
51	BP0	O

PIN NBR	SCM, BOOT, SELF-CHECK	I/O
52	FP0	O
53	FP1	O
54	FP2	O
55	FP3	O
56	FP4	O
57	FP5	O
58	FP6	O
59	FP7	O
61	FP8	O
62	FP9	O
63	FP10	O
64	FP11	O
65	FP12	O
66	FP13	O
67	FP14	O
68	FP15	O
69	FP16	O
70	FP17	O
71	FP18	O
72	FP19	O
73	FP20	O
74	FP21	O
75	FP22	O
76	FP23	O
77	FP24	O
78	FP25	O
79	FP26	O
80	FP27/PE7	O
2	FP28/PE6	O
3	FP29/PE5	O
4	FP30/PE4	O
5	FP31/PE3	O
6	FP32/PE2	O
7	FP33/PE1	O
8	FP34/PE0	O
9	FP35/PD7	O
10	FP36/PD6	O
11	FP37/PD5	O
12	FP38/PD4	O

* Open Drain only when Output

1.4.1 Mode Entry

Mode entry is done at the rising edge of the $\overline{\text{RESET}}$ pin. Once the device enters one of the four modes, the mode cannot be changed by software, only by an external reset.

At the rising edge of the $\overline{\text{RESET}}$ pin, the device latches the states of $\overline{\text{IRQ1}}$ and $\overline{\text{IRQ2}}$ and places itself in the specified mode. While the $\overline{\text{RESET}}$ pin is low, all pins are configured as single-chip mode. The following table shows the states of $\overline{\text{IRQ1}}$ and $\overline{\text{IRQ2}}$ for each mode entry.

Table 1-2. Mode Select Summary

Modes	$\overline{\text{RESET}}$	PC6/$\overline{\text{IRQ1}}$	PC7/$\overline{\text{IRQ2}}$
Single-Chip (User) Mode	↑	V_{SS} or V_{DD}	V_{SS} or V_{DD}
Self-Check Mode (MC68HC05L16) Bootstrap Mode (MC68HC705L16)	↑	V_{TST}	V_{DD}

High voltage V_{TST} = 2 x V_{DD} is required to select modes other than single-chip mode.

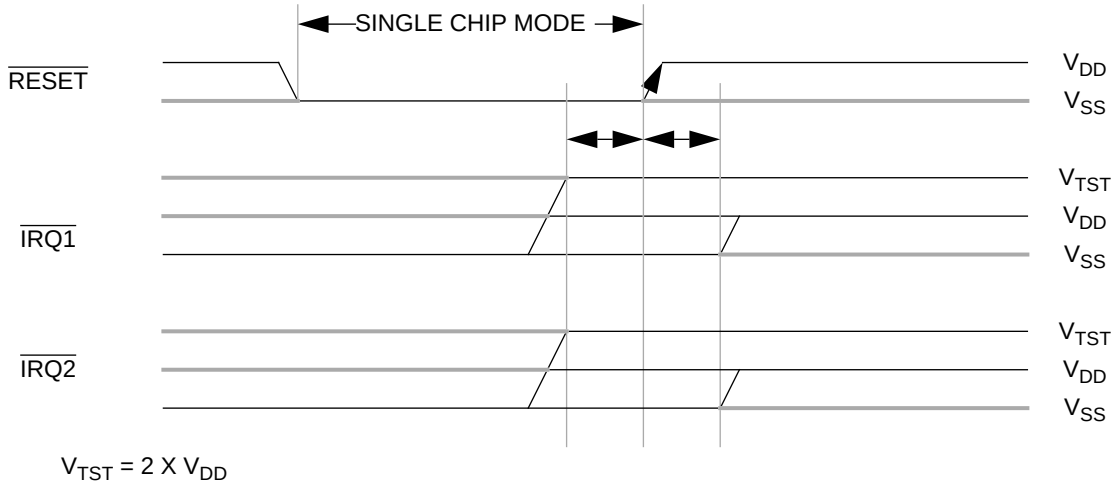


Figure 1-2. MC68HC(7)05L16 Mode Entry Diagram

1.4.2 Single-Chip Mode (SCM)

In this mode, all address and data bus activity occurs within the MCU. Thus, no external pins are required for these functions. The single-chip mode allows the maximum number of I/O pins for on-chip peripheral functions, for example, port A through port E, and LCD drivers.

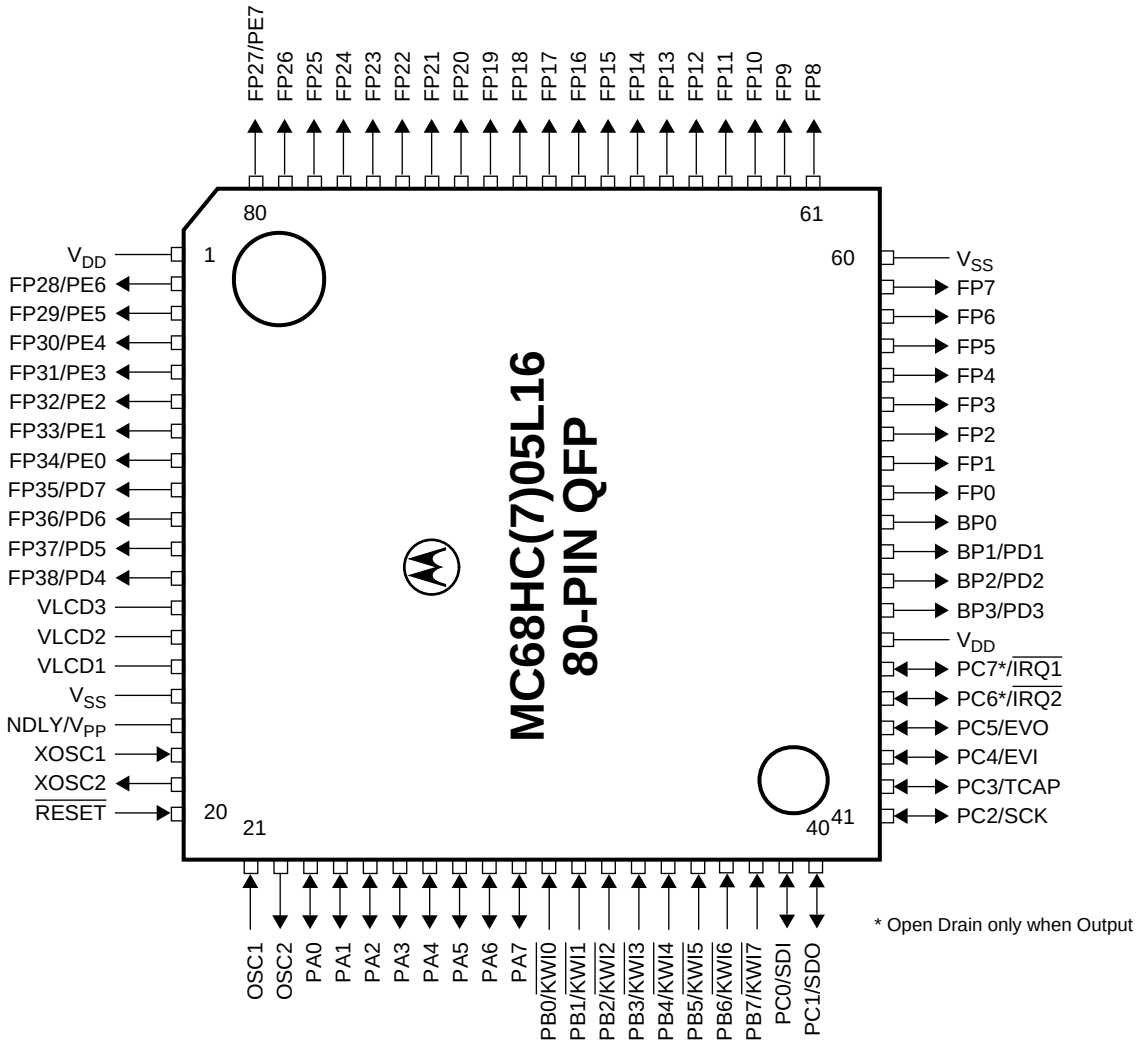


Figure 1-3. Pin Assignments for Single-Chip Mode

1.4.3 Self-Check/Bootstrap Mode

In this mode, the reset vector is fetched from a 496-byte internal self-check (bootstrap for MC68HC705L16) ROM at $FE00-$FFEF. The self-check ROM contains a self-check program to test the functions of internal modules. The bootstrap ROM contains a small program which loads a program into the internal RAM and then passes control to that program at location $0040 or executes EPROM programming sequence and dumps EPROM contents.

Since these modes are not normal user modes, all of the privileged control bits are accessible. This allows the self-check/bootstrap mode to be used for self test of the device.

SECTION 2
MEMORY

The MC68HC05L16 contains a 16,384-byte mask ROM, 496 bytes of self-check ROM, and 512 bytes of RAM. An additional 16 bytes of mask ROM are provided for user vectors at $FFF0 through $FFFF.

In the MC68HC705L16 (EPROM device), the 16,384-byte ROM and 16-byte user vector areas are replaced by EPROM, and self-check ROM is called bootstrap ROM.

Memory map for the MCU is shown in Figure 2-1.

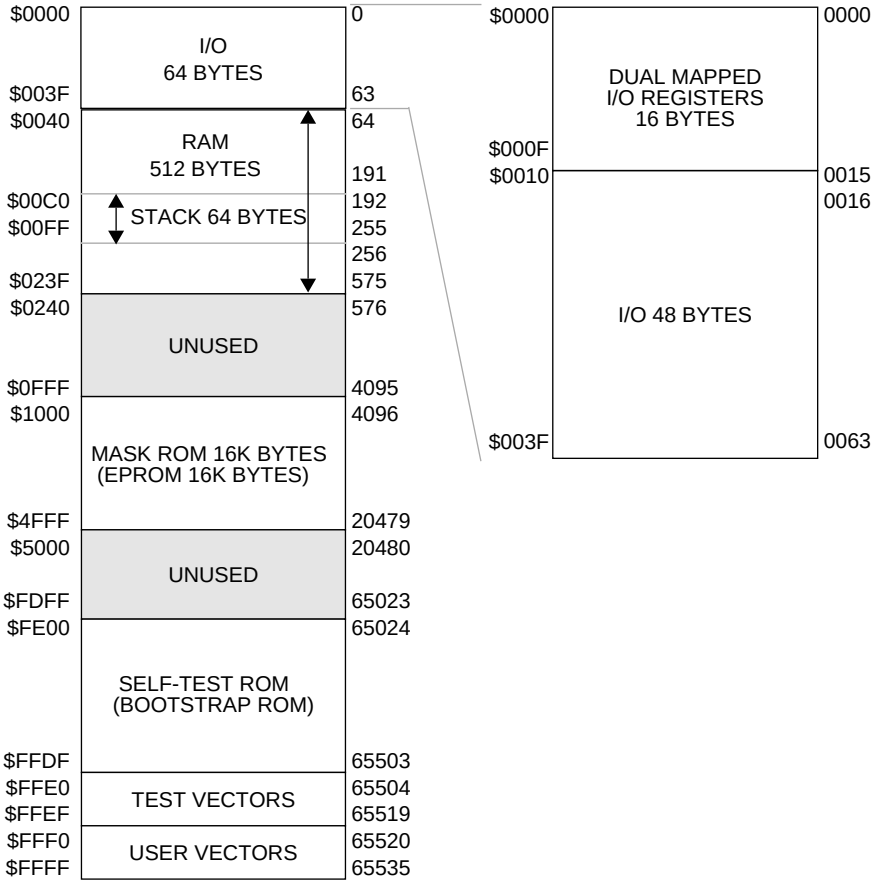


Figure 2-1. MC68HC(7)05L16 Memory Map

2.1 Summary of Internal Registers and I/O Map

An explanation of how to interpret the register figures, used in this document, is shown in Figure 2-2.

Refer to Figures 2-3 through 2-6 for a more detailed memory map of the I/O register section.

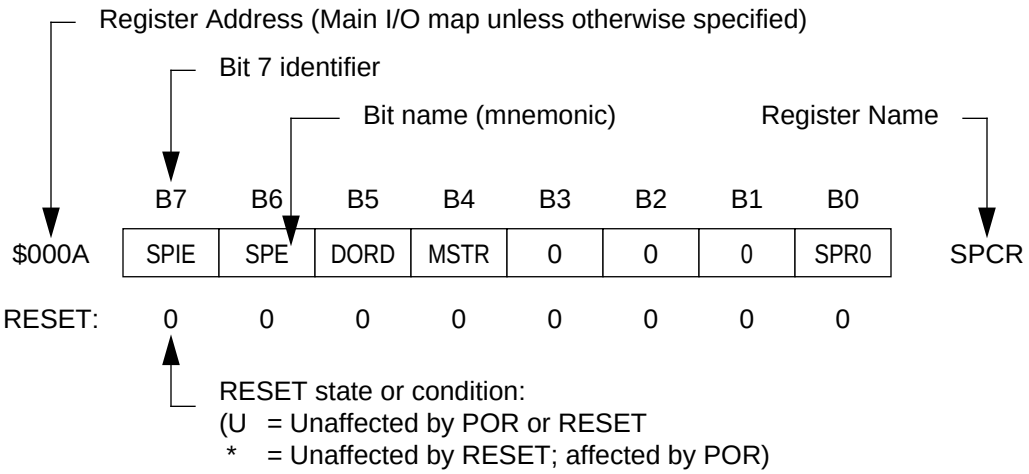


Figure 2-2. Register Description Key

INTERNAL REGISTERS — MAIN I/O MAP (OPTM = 0)

	B7	B6	B5	B4	B3	B2	B1	B0	
$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	SPR0	SPCR
$000B	SPIF	DCOL	0	0	0	0	0	0	SPSR
$000C	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0	SPDR
$000D	—	—	—	—	—	—	—	—	(reserved)
$000E	—	—	—	—	—	—	—	—	(reserved)
$000F	—	—	—	—	—	—	—	—	(reserved)
	B7	B6	B5	B4	B3	B2	B1	B0	

Figure 2-3. Main I/O Map ($0000-$000F)

INTERNAL REGISTERS — I/O MAP

	B7	B6	B5	B4	B3	B2	B1	B0	
$0010	TBCLK	0	LCLK	0	0	0	T2R1	T2R0	TBCR1
$0001	TBIF	TBIE	TBR1	TBR0	RTBIF	0	COPE	COPC	TBCR2
$0012	ICIE	OC1IE	TOIE	0	0	0	IEDG	0	TCR
$0013	ICF	OC1F	TOF	0	0	0	0	0	TSR
$0014	BIT15	BIT14	BIT13	BIT12	BIT11	BIT10	BIT9	BIT8	ICH
$0015	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0	ICL
$0016	BIT15	BIT14	BIT13	BIT12	BIT11	BIT10	BIT9	BIT8	OC1H
$0017	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0	OC1L
$0018	BIT15	BIT14	BIT13	BIT12	BIT11	BIT10	BIT9	BIT8	TCNTH
$0019	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0	TCNTL
$001A	BIT15	BIT14	BIT13	BIT12	BIT11	BIT10	BIT9	BIT8	ACNTH
$001B	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0	ACNTL
$001C	TI2IE	OC2IE	0	T2CLK	IM2	IL2	OE2	OL2	TCR2
$001D	TI2F	OC2F	0	0	RTI2F	ROC2F	0	0	TSR2
$001E	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0	OC2
$001F	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0	TCNT2
	B7	B6	B5	B4	B3	B2	B1	B0	

Figure 2-4. Main I/O Map ($0010-$001F)

INTERNAL REGISTERS — I/O MAP

	B7	B6	B5	B4	B3	B2	B1	B0	
$0020	LCDE	DUTY1	DUTY0	0	PEH	PEL	PDH	0	LCDCR
$0021	F1B3	F1B2	F1B1	F1B0	F0B3	F0B2	F0B1	F0B0	LCDR1
$0022	F3B3	F3B2	F3B1	F3B0	F2B3	F2B2	F2B1	F2B0	LCDR2
$0023	F5B3	F5B2	F5B1	F5B0	F4B3	F4B2	F4B1	F4B0	LCDR3
$0024	F7B3	F7B2	F7B1	F7B0	F6B3	F6B2	F6B1	F6B0	LCDR4
$0025	F9B3	F9B2	F9B1	F9B0	F8B3	F8B2	F8B1	F8B0	LCDR5
$0026	F11B3	F11B2	F11B1	F11B0	F10B3	F10B2	F10B1	F10B0	LCDR6
$0027	F13B3	F13B2	F13B1	F13B0	F12B3	F12B2	F12B1	F12B0	LCDR7
$0028	F15B3	F15B2	F15B1	F15B0	F14B3	F14B2	F14B1	F14B0	LCDR8
$0029	F17B3	F17B2	F17B1	F17B0	F16B3	F16B2	F16B1	F16B0	LCDR9
$002A	F19B3	F19B2	F19B1	F19B0	F18B3	F18B2	F18B1	F18B0	LCDR10
$002B	F21B3	F21B2	F21B1	F21B0	F20B3	F20B2	F20B1	F20B0	LCDR11
$002C	F23B3	F23B2	F23B1	F23B0	F22B3	F22B2	F22B1	F22B0	LCDR12
$002D	F25B3	F25B2	F25B1	F25B0	F24B3	F24B2	F24B1	F24B0	LCDR13
$002E	F27B3	F27B2	F27B1	F27B0	F26B3	F26B2	F26B1	F26B0	LCDR14
$002F	F29B3	F29B2	F29B1	F29B0	F28B3	F28B2	F28B1	F28B0	LCDR15

Figure 2-5. Main I/O Map ($0020-$002F)

INTERNAL REGISTERS — I/O MAP

	B7	B6	B5	B4	B3	B2	B1	B0	
$0030	F31B3	F31B2	F31B1	F31B0	F30B3	F30B2	F30B1	F30B0	LCDR16
$0031	F33B3	F33B2	F33B1	F33B0	F32B3	F32B2	F32B1	F32B0	LCDR17
$0032	F35B3	F35B2	F35B1	F35B0	F34B3	F34B2	F34B1	F34B0	LCDR18
$0033	F37B3	F37B2	F37B1	F37B0	F36B3	F36B2	F36B1	F36B0	LCDR19
$0034	0	0	0	0	F38B3	F38B2	F38B1	F38B0	LCDR20
$0035	—	—	—	—	—	—	—	—	(reserved)
$0036	—	—	—	—	—	—	—	—	(reserved)
$0037	—	—	—	—	—	—	—	—	(reserved)
$0038	—	—	—	—	—	—	—	—	(reserved)
$0039	—	—	—	—	—	—	—	—	(reserved)
$003A	—	—	—	—	—	—	—	—	(reserved)
$003B	—	—	—	—	—	—	—	—	(reserved)
$003C	—	—	—	—	—	—	—	—	(reserved)
$003D	0	0	0	0	0	0	ELAT	PGM	PCR*
$003E	FTUP	STUP	0	0	SYS1	SYS0	FOSCE	OPTM	MISC
$003F	—	—	—	—	—	—	—	—	(reserved)
	B7	B6	B5	B4	B3	B2	B1	B0	

* PCR is not available on MC68HC05L16.

Figure 2-6. Main I/O Map ($0030-$003F)

2.2 Option Map for I/O Configurations

Most of the I/O configurations are done in the option map see Figure 2-7. Some options still remain as mask options for MC68HC05L16 such as pull-up resistor for $\overline{\text{RESET}}$ pin and resistors for OSC1/OSC2 and XOSC1/XOSC2 pins. These mask options may be read by the MOSR ($000F) in the option map.

The option map is located at $0000 through $000F of the main memory map and it is available when OPTM bit in the MISC register ($003E) is set. Main registers at $0000 through $000F are not available when OPTM = 1.

Data direction registers of I/O ports are available in the option map.

SYSTEM CONFIGURATION — OPTION MAP (OPTM = 1)

	B7	B6	B5	B4	B3	B2	B1	B0	
$0000	DDRA7	DDRA6	DDRA5	DDRA4	DDRA3	DDRA2	DDRA1	DDRA0	DDRA
$0001	—	—	—	—	—	—	—	—	(reserved)
$0002	DDRC7	DDRC6	DDRC5	DDRC4	DDRC3	DDRC2	DDRC1	DDRC0	DDRC
$0003	PDM7	PDM6	PDM5	PDM4	0	0	0	0	PDMUX
$0004	PEM7	PEM6	PEM5	PEM4	PEM3	PEM2	PEM1	PEM0	PEMUX
$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	1	1	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*
	B7	B6	B5	B4	B3	B2	B1	B0	

* MOSR is not available on MC68HC705L16.

Figure 2-7. Option Map ($0000–$000F)

2.2.1 Resistor Control Register 1 (RCR1)

	B7	B6	B5	B4	B3	B2	B1	B0	
$0008	0	0	0	0	RBH	RBL	RAH	RAL	RCR1
RESET:	0	0	0	0	0	0	0	0	

READ: anytime

WRITE: anytime

Bits 7 through 4 — Reserved
These bits are not used and always read as logic zero.

RBH — Port B Pull-up Resistor (H)
When this bit is set, pull-up resistors are connected to the upper four bits of port B. This bit is cleared on reset.

RBL — Port B Pull-up Resistor (L)
When this bit is set, pull-up resistors are connected to the lower four bits of port B. This bit is cleared on reset.

RAH — Port A Pull-up Resistor (H)
When this bit is set, pull-up resistors are connected to the upper four bits of port A. This bit is cleared on reset.

RAL — Port A Pull-up Resistor (L)
When this bit is set, pull-up resistors are connected to the lower four bits of port A. This bit is cleared on reset.

2.2.2 Resistor Control Register 2 (RCR2)

	B7	B6	B5	B4	B3	B2	B1	B0	
$0009	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	RCR2
RESET:	0	0	0	0	0	0	0	0	

READ: anytime

WRITE: anytime

RC*x* — Port C Pull-up Resistor (Bit *x*)

When RC*x* bit is set, the pull-up resistor is connected to the corresponding bit of port C. This bit is cleared on reset.

2.2.3 Open-Drain Output Control Register 1 (WOM1)

	B7	B6	B5	B4	B3	B2	B1	B0	
$000A	DWOMH	DWOML	EWOMH	EWOML	0	0	AWOMH	AWOML	WOM1
RESET:	0	0	0	0	0	0	0	0	

READ: anytime

WRITE: anytime

DWOMH — Port D Open-Drain Mode (H)

When this bit is set, the upper four bits of port D are configured as open-drain outputs if these bits are selected as port D output by the PDH bit in the LCDCR. This bit is cleared on reset.

DWOML — Port D Open-Drain Mode (L)

When this bit is set, the lower three bits of port D are configured as open-drain outputs if the corresponding BP*x* pin is not used by the LCD driver. This bit is cleared on reset.

EWOMH — Port E Open-Drain Mode (H)

When this bit is set, the upper four bits of port E (that are configured as I/O output by PEH bit in the LCDCR) are configured as open-drain outputs. This bit is cleared on reset.

EWOML — Port E Open-Drain Mode (L)

When this bit is set, the lower four bits of port E (that are configured as I/O output by PEL bit in the LCDCR) are configured as open-drain outputs. This bit is cleared on reset.

Bits 3 and 2 — Reserved

These bits are not used and always return logic zero.

AWOMH — Port A Open-Drain Mode (H)

When this bit is set, the upper four bits of port A that are configured as output (corresponding DDRA bit set) become open-drain outputs. This bit is cleared on reset.

AWOML — Port E Open-Drain Mode (L)

When this bit is set, the lower four bits of port A that are configured as output (corresponding DDRA bit set) become open-drain outputs. This bit is cleared on reset.

2.2.4 Open-Drain Output Control Register 2 (WOM2)

	B7	B6	B5	B4	B3	B2	B1	B0	
$000B	1	1	CWOM5	CWOM4	CWOM3	CWOM2	CWOM1	CWOM0	WOM2
RESET:	1	1	0	0	0	0	0	0	

READ: anytime

WRITE: anytime

Bits 7 through 6 — Reserved

These bits are not used and always read as logic one. PC6 and PC7 when configured as output are always open drain.

CWOM*x* — Port C Open-Drain Mode (bit *x*)

When CWOM*x* bit is set, port C bits *x* are configured as open-drain outputs if DDRC*x* is set. This bit is cleared on reset.

2.2.5 Key Wake-Up Input Enable Register (KWIEN)

	B7	B6	B5	B4	B3	B2	B1	B0	
$000E	KWIE7	KWIE6	KWIE5	KWIE4	KWIE3	KWIE2	KWIE1	KWIE0	KWIEN
RESET:	0	0	0	0	0	0	0	0	

READ: anytime

WRITE: anytime

KWIE*x* — Key Wake-up Input Enable (bit *x*)
When KWIE*x* bit is set, the KWI*x* (PB*x*) input is enabled for key wake-up interrupt. This bit is cleared on reset.

2.2.6 Mask Option Status Register (MOSR)

The mask option status register (MOSR) indicates the state of mask options specified prior to production of MC68HC05L16. There are no mask options for MC68HC705L16 and thus this register has no meaning.

	B7	B6	B5	B4	B3	B2	B1	B0	
$000F	RSTR	OSCR	XOSCR	0	0	0	0	0	MOSR
RESET:	U	U	U	0	0	0	0	0	

READ: anytime for MC68HC05L16; not available on MC68HC705L16

WRITE: no effect

RSTR — $\overline{\text{RESET}}$ Pin Pull-up Resistor

When this bit is set, it indicates an internal pull up resistor is attached to $\overline{\text{RESET}}$ pin by mask option.

OSCR — OSC Feedback Resistor

When this bit is set, it indicates that an internal feedback resistor is attached between OSC1 and OSC2 by mask option.

XOSCR — OSC Feedback Resistor

When this bit is set, it indicates that an internal feedback resistor is attached between XOSC1 and XOSC2. The damping resistor at the XOSC2 pin is attached by mask option.

Bits 4 through 0 — Reserved

These bits are not used and always read as logic zero.

2.3 RAM

The 512 byte internal RAM is positioned at $0040 through $023F in the memory map. The lower 192 bytes are positioned in the page zero which are accessible by the direct addressing mode, but the upper 320 bytes of page zero are used for the CPU stack area.

Care should be taken if the stack area is used for data storage.

The RAM is implemented with static cells and retains its contents during the STOP and WAIT modes.

2.4 Self-Check ROM (MC68HC05L16)

Self-check ROM is 496 bytes of mask ROM positioned at $FE00 through $FFEF. This ROM contains self-check programs and reset/interrupt vectors in the self-check mode.

2.5 Boot ROM (MC68HC705L16)

Boot ROM is 496 bytes of mask ROM positioned at $FE00 through $FFEF. This ROM contains bootstrap loader programs and reset/interrupt vectors in the bootstrap mode. The bootstrap loader programs include:

- EPROM Programming and Verification
- Dumping EPROM Contents
- Loading Programs into the Internal RAM
- Executing Programs in the Internal RAM

2.6 Mask ROM (MC68HC05L16)

The 16K-byte user ROM is positioned at $1000 through $4FFF, and an additional 16 bytes of ROM are located at $FFF0 through $FFFF for user vectors. In mask ROM devices, NDLY/V_{PP} pin is not used and should be connected to V_{DD}; EPROM program control register (PCR) is not implemented.

2.7 EPROM (MC68HC705L16)

The 16K-byte EPROM is positioned at $1000 through $4FFF, and the additional 16 bytes of EPROM are located at $FFF0 through $FFFF for user vectors. The erased state of EPROM is read as $FF and EPROM power is supplied from V_{PP} pin and V_{DD} pin.

The program control register (PCR) is provided for EPROM programming and testing. The functions of EPROM depend on the device mode.

In the user mode, ELAT and PGM bits in the PCR are available for the user programming and remaining test bits become read-only bits. The V_{PP} pin should be tied to 5 V or programming voltage.

2.7.1 Programming Sequence

The following sequence is taken:

- Set the ELAT Bit
- Write the Data to the Address to be Programmed
- Set the PGM Bit
- Delay for an Appropriate Amount of Time
- Clear the PGM and the ELAT Bit

The last item may be done on a single CPU write. It is important to remember that an external programming voltage must be applied to the V_{PP} pin while programming, but it should be equal to V_{DD} during normal operations.

2.7.2 Program Control Register (PCR)

A program control register is provided for EPROM programming. This register is available only in the MC68HC705L16 (EPROM device).

	B7	B6	B5	B4	B3	B2	B1	B0	
$003D	0	0	0	0	0	0	ELAT	PGM	PCR
RESET:	0	0	0	0	0	0	0	0	

READ: In user mode, bits 2 through 7 read as logic zero

WRITE: Bits 2 through 7 are reserved.

Bits 7 through 3 — Reserved

These bits are reserved and read as logic zero in user mode.

Bit 2 — Reserved

This bit is not used and always reads as logic zero.

ELAT — EPROM LATch control

0 = EPROM address and data bus configured for normal reads

1 = EPROM address and data bus configured for programming (writes to EPROM cause address and data to be latched). EPROM is in programming mode and cannot be read if ELAT is logic one. This bit should not be set when no programming voltage is applied to the V_{PP} pin.

PGM — EPROM ProGraM command

0 = Programming power is switched OFF from EPROM array.

1 = Programming power is switched ON to EPROM array. If ELAT ≠ 1, then PGM = 0.

SECTION 3
CENTRAL PROCESSOR UNIT

The MC68HC(7)05L16 MCU contains five registers as shown in the programming model of Figure 3-1.

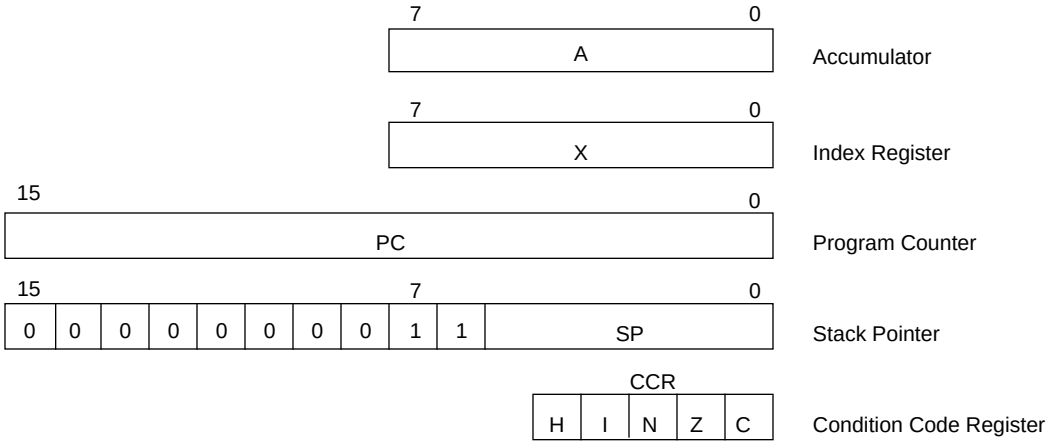


Figure 3-1. Programming Model

3.1 Accumulator (A)

The accumulator is a general purpose 8-bit register used to hold operands and results of arithmetic calculations or data manipulations.

3.2 Index Register (X)

The index register is an 8-bit register used for the indexed addressing value to create an effective address. The index register may also be used as a temporary storage area.

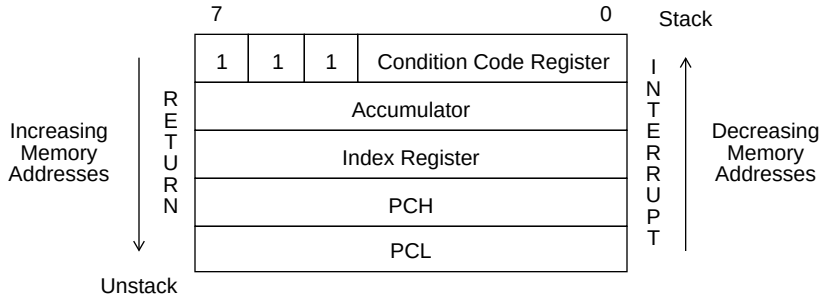
3.3 Program Counter (PC)

The program counter is a 16-bit register that contains the address of the next byte to be fetched.

3.4 Stack Pointer (SP)

The stack pointer contains the address of the next free location on the stack. During an MCU reset or the reset stack pointer (RSP) instruction, the stack pointer is set to location $00FF. The stack pointer is then decremented as data is pushed onto the stack and incremented as data is pulled from the stack.

When accessing memory, the 10 most significant bits are permanently set to 0000000011. These eight zero bits are appended to the six least significant register bits to produce an address within the range of $00FF to $00C0. Subroutines and interrupts may use up to 64 (decimal) locations. If 64 locations are exceeded, the stack pointer wraps around and loses the previously stored information. A subroutine call occupies two locations on the stack; an interrupt uses five locations. See Figure 3-2.



Note: Since the Stack Pointer decrements during pushes, the PCL is stacked first, followed by PCH, etc. Pulling from the stack is in the reverse order.

Figure 3-2. Stacking Order

3.5 Condition Code Register (CCR)

The CCR is a 5-bit register in which four bits are used to indicate the results of the instruction just executed, and the fifth bit indicates whether interrupts are masked. These bits can be individually tested by a program, and specific actions can be taken as a result of their state. Each bit is explained in the following paragraphs.

3.5.1 Half-Carry Flag (H)

This bit is set during ADD and ADC operations to indicate that a carry occurred between bits 3 and 4.

3.5.2 Interrupt Mask (I)

When this bit is set, the timer and external interrupt are masked (disabled). If an interrupt occurs while this bit is set, the interrupt is latched and processed as soon as the interrupt bit is cleared.

3.5.3 Negative Flag (N)

When set, this bit indicates that the result of the last arithmetic, logical, or data manipulation was negative.

3.5.4 Zero Flag (Z)

When set, this bit indicates that the result of the last arithmetic, logical, or data manipulation was zero.

3.5.5 Carry/Borrow Flag (C)

When set, this bit indicates that a carry or borrow out of the arithmetic logical unit (ALU) occurred during the last arithmetic operation. This bit is also affected during bit test and branch instructions and during shifts and rotates.

3.6 Arithmetic/Logic Unit (ALU)

The ALU performs the arithmetic and logical operations defined by the instruction set.

The binary arithmetic circuits decode instructions and set up the ALU for the selected operation. Most binary arithmetic is based on the addition algoirthm, carrying out subtraction as negative addition. Multiplication is not performed as a discrete operation but as a chain of addition and shift operations within the ALU. The multiply instruction (MUL) requires 11 internal processor cycles to complete this chain of operations.

SECTION 4
RESET/INTERRUPT STRUCTURE

In user operating modes, the reset/interrupt vectors are located at the top of the address space ($FFF0 through $FFFF). In the self-check (bootstrap) mode, the reset/interrupt vectors are located at $FFE0 through $FFEF in the internal self-check (bootstrap) ROM. For the remainder of this section, a user operating mode will be assumed. The following table shows the address assignments for the vectors.

Table 4-1. Interrupt Vector Assignments

VECTOR ADDRESS	INTERRUPT SOURCE		MASKED BY	LOCAL MASK	PRIORITY (1=HIGHEST)
FFF0-F1	Time Base		I BIT	TBIE	7
FFF2-F3	SSPI		I BIT	SPIE	6
FFF4-F5	TIMER 2	TI2I	I BIT	TI2IE	5
		OC2I	I BIT	OC2IE	5
FFF6-F7	TIMER 1	ICI	I BIT	ICIE	4
		OC1I	I BIT	OC1IE	4
		TOI	I BIT	TOIE	4
FFF8-F9	KWI		I BIT	KWIE	3
FFFA-FB	IRQ	$\overline{\text{IRQ1}}$	I BIT	IRQ1E	2
		$\overline{\text{IRQ2}}$	I BIT	IRQ2E	2
FFFC-FD	SWI		none	none	*
FFFE-FF	RESET	COP	none	COPE	1
		$\overline{\text{RESET}}$ PIN	none	none	1
		POWER-ON	none	none	1

*Same level as an instruction.

Upon reset, the I bit in the condition code register is set and interrupts are disabled (masked). When an interrupt occurs, the I bit is automatically set by hardware after stacking the condition code register (CCR). All interrupts in the MC68HC(7)05L16 follow a fixed hardware priority circuit to resolve simultaneous requests.

Each interrupt has a software programmable interrupt mask bit which may be used to selectively inhibit automatic hardware response. In addition, the I bit in the condition code register acts as a class inhibit mask to inhibit all sources in the I-bit class. $\overline{\text{RESET}}$ and software interrupt (SWI) are not masked by the I bit in the condition code register.

SWI is an instruction rather than a prioritized asynchronous interrupt source. In a sense, it is lower in priority than any source because once any interrupt sequence has begun, SWI cannot override it. In another sense, it is higher in priority than any hardware sources except reset because once the SWI opcode is fetched, no other sources can be honored until after the first instruction in the SWI service routine has been executed. SWI causes the I mask bit in the CC register to be set.

4.1 Interrupts of the MC68HC(7)05L16

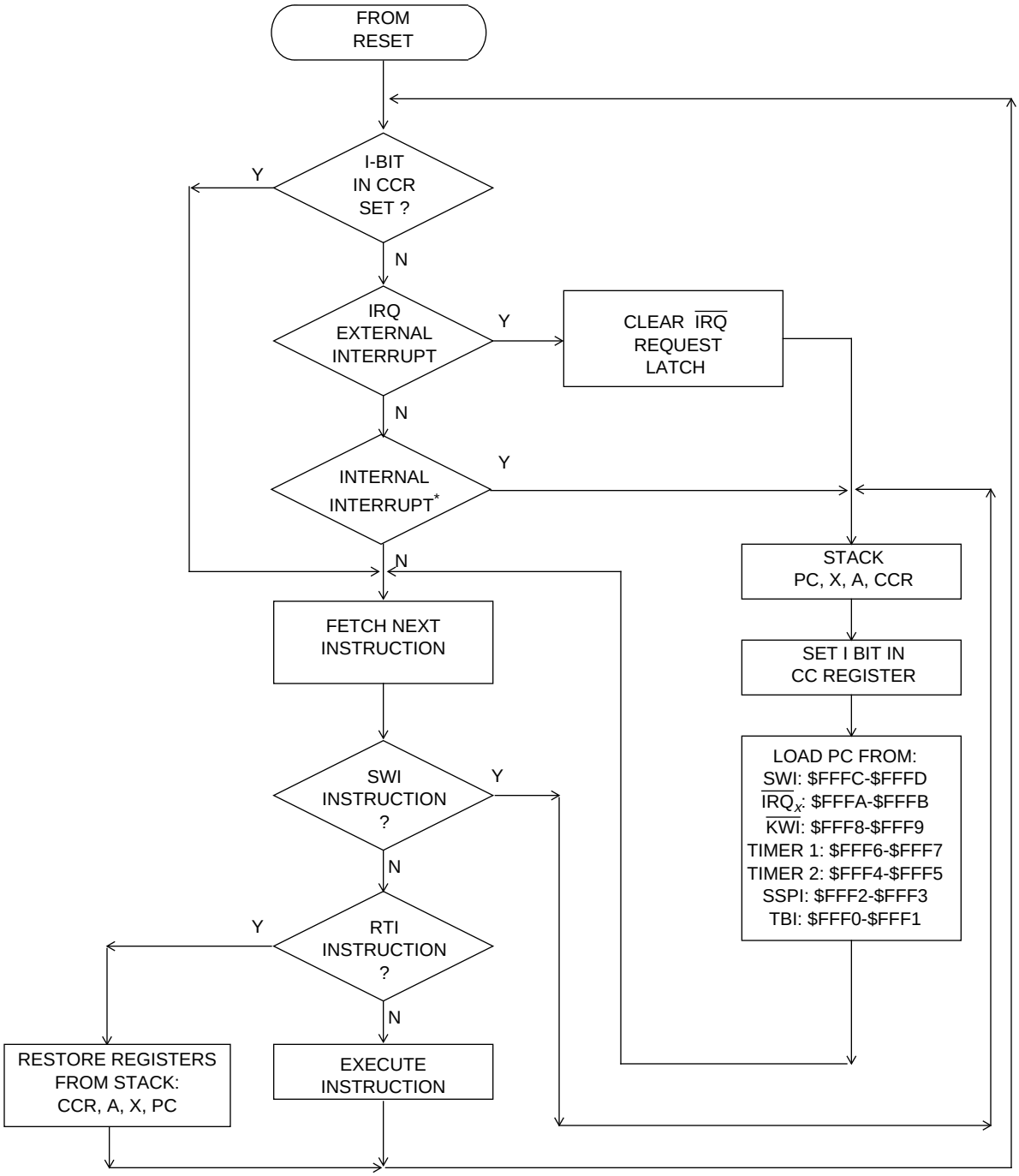
There are six hardware interrupt sources in the MC68HC(7)05L16: $\overline{\text{IRQ1}}$ and $\overline{\text{IRQ2}}$, key wake-up interrupt ($\overline{\text{KWI}}$), timer 1 (TOI, ICI, OC1I), timer 2 (TI2I, OC2I), serial transfer complete interrupt (SSPI), and time base interrupt (TBI).

4.1.1 $\overline{\text{IRQ1}}$ and $\overline{\text{IRQ2}}$

Two external interrupt request inputs, $\overline{\text{IRQ1}}$ and $\overline{\text{IRQ2}}$, share the same vector address at $FFFA and $FFFB.

Bits IRQ1S and IRQ2S in interrupt control register (INTCR) control whether $\overline{\text{IRQ1}}$ and $\overline{\text{IRQ2}}$, respectively, respond only to the falling edge or falling edge and low level to trigger an interrupt. The $\overline{\text{IRQ1}}$ and $\overline{\text{IRQ2}}$ are enabled by IRQ1E and IRQ2E bits and IRQ1F and IRQ2F bits are provided as an indicator in the interrupt status register (INTSR). Since the IRQ6(7)F can be set by either the pins or the data latches of PC6(7), be sure to clear the flags by software before setting the IRQ6(7)E bit.

The $\overline{\text{IRQ1}}$ and the $\overline{\text{IRQ2}}$ pins are shared with port C bit 7 and bit 6, respectively, and IRQ*x* pin states can be determined by reading port C pins. The BIL and BIH instructions apply only to the $\overline{\text{IRQ1}}$ input.



Note:
* $\overline{KWI}$, Timer1, Timer2, SSPI, and TBI

Figure 4-1. Interrupt Flowchart

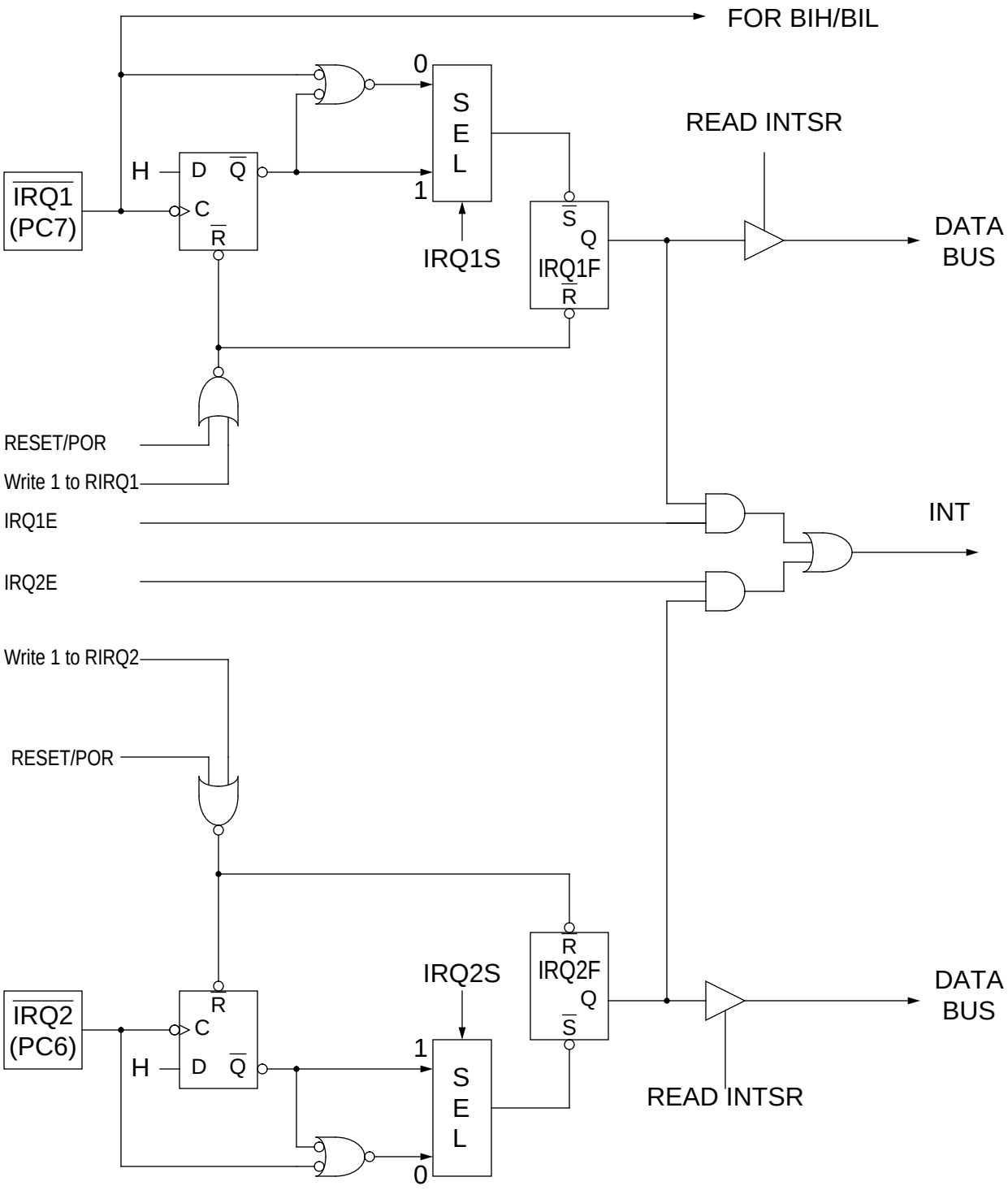


Figure 4-2. $\overline{\text{IRQ1}}$ and $\overline{\text{IRQ2}}$ Block Diagram

4.1.2 Key Wake-Up Interrupt ($\overline{KWI}$)

Eight key wake-up inputs ($\overline{KWI0}$-$\overline{KWI7}$) share pins with port B. Each key wake-up input is enabled by the corresponding bit in the KWIEN register which resides in the options map, and $\overline{KWI}$ is enabled by the KWIE bit in the INTCR.

When a falling edge is detected at one of the enabled key wake-up inputs, the KWIF bit in the INTSR is set and $\overline{KWI}$ is generated if KWIE = 1. Each input has a latch which responds only to the falling edge at the pin, and all input latches are cleared at the same time by clearing KWIF bit. See Figure 4-3.

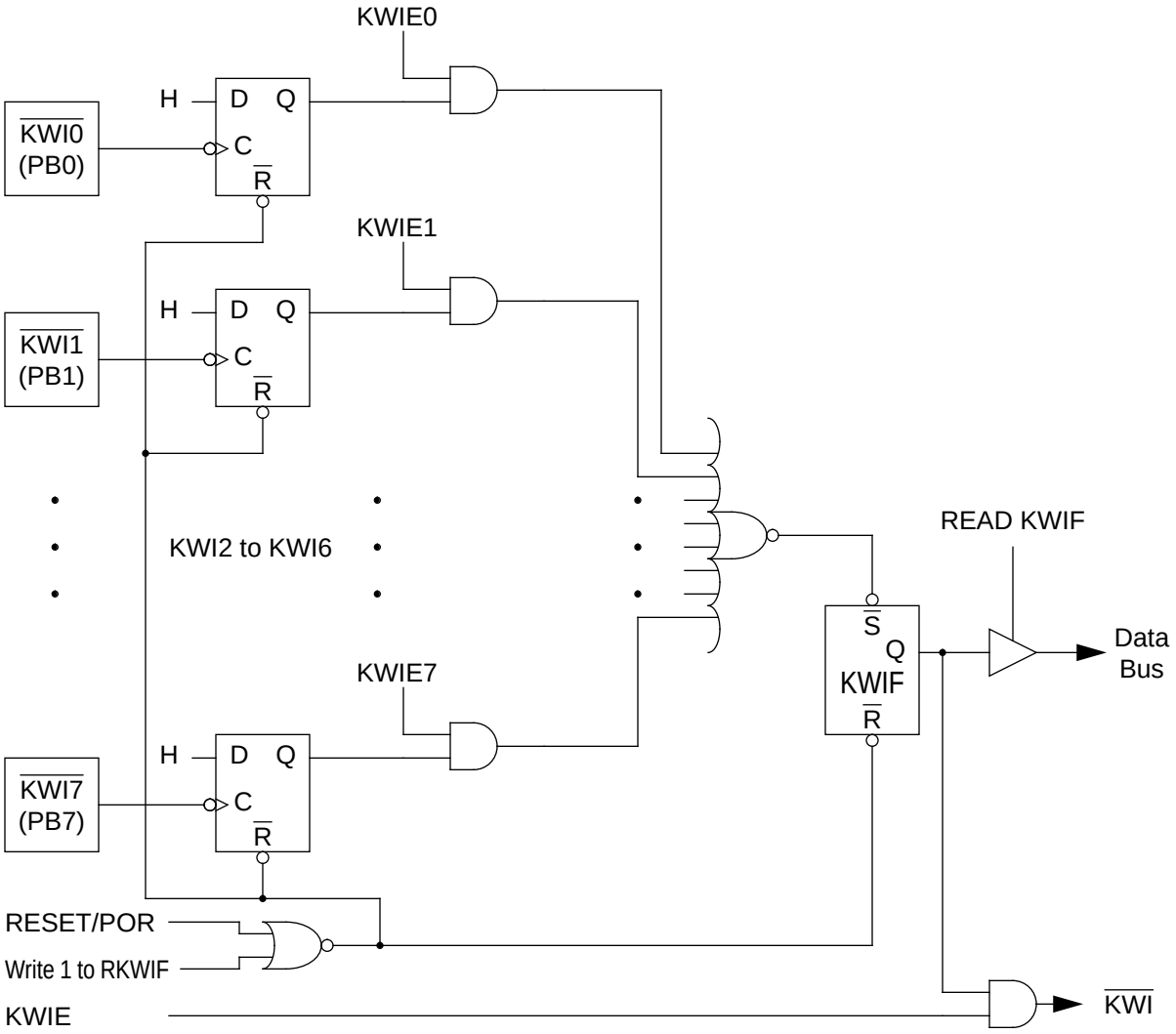


Figure 4-3. Key Wake-up Interrupt ($\overline{KWI}$)

4.1.3 Timer 1 Interrupt

Three timer 1 interrupts (TOI, ICI, and OC1I) share the same interrupt vector at $FFF6 and $FFF7. See **9.1 Timer 1.**

4.1.4 Timer 2 Interrupt

Two timer 2 interrupts (TI2I and OC2I) share the same interrupt vector at $FFF4 and $FFF5. See **9.2.1 Timer Control Register 2 (TCR2).**

4.1.5 SSPI Interrupt

The SSPI transfer complete interrupt uses the vector at $FFF2 and $FFF3. See **SECTION 8 SIMPLE SERIAL PERIPHERAL INTERFACE (SSPI).**

4.1.6 Time Base Interrupt

The time base interrupt uses the vector at $FFF0 and $FFF1. See **7.4 Time Base.**

4.1.7 Interrupt Control Register (INTCR)

	B7	B6	B5	B4	B3	B2	B1	B0	
$0008	IRQ1E	IRQ2E	0	KWIE	IRQ1S	IRQ2S	0	0	INTCR
RESET:	0	0	0	0	0	0	0	0	

READ: anytime

WRITE: anytime

IRQ1E — IRQ1 Interrupt Enable

IRQ1E bit enables IRQ1 interrupt when IRQ1F is set. This bit is cleared on reset.

0 = IRQ1 Interrupt is disabled
1 = IRQ1 Interrupt is enabled

IRQ2E — IRQ2 Interrupt Enable

IRQ2E bit enables IRQ2 interrupt when IRQ2F is set. This bit is cleared on reset.

0 = IRQ2 Interrupt is disabled
1 = IRQ2 Interrupt is enabled

Bit 5 — reserved

This bit is not used and is always read as logic zero.

KWIE — Key Wake-up Interrupt (KWI) Enable

KWIE bit enables key wake-up interrupt when KWIF is set. This bit is cleared on reset.

0 = KWI is disabled
1 = KWI is enabled

IRQ1S — IRQ1 Select Edge Sensitive Only

0 = IRQ1 is configured for low LEVEL and negative edge sensitive
1 = IRQ1 is configured to respond only to negative EDGEs

IRQ2S — IRQ2 Select Edge Sensitive Only

0 = IRQ2 is configured for low LEVEL and negative edge sensitive
1 = IRQ2 is configured to respond only to negative EDGEs

Bits 1 and 0 — reserved

These bits are not used and always read as logic zero.

4.1.8 Interrupt Status Register (INTSR)

	B7	B6	B5	B4	B3	B2	B1	B0	
$0009	IRQ1F	IRQ2F	0	KWIF	RIRQ1	RIRQ2	0	RKWIF	INTSR
RESET:	0	0	0	0	0	0	0	0	

READ: anytime
(Bits 3 through 0 are write-only bits and always read as logic zero)

WRITE: anytime
(Bits 7 through 4 are read-only bits and write has no effect)

IRQ1F — IRQ1 Interrupt Flag

When IRQ1S = 0, the falling edge or low level at $\overline{\text{IRQ1}}$ pin sets IRQ1F. When IRQ1S = 1, only the falling edge sets IRQ1F bit. If IRQ1E bit and this bit are set, an interrupt is generated. This bit is a read-only bit and is cleared by writing a logic one to the RIRQ1 bit. Reset clears this bit.

IRQ2F — IRQ2 Interrupt Flag

When IRQ2S = 0, the falling edge or low level at $\overline{\text{IRQ2}}$ pin sets IRQ2F. When IRQ2S = 1, only the falling edge sets IRQ2F bit. If IRQ2E bit and this bit are set, an interrupt is generated. This bit is a read-only bit and is cleared by writing a a logic one to the RIRQ2 bit. Reset clears this bit.

Bit 5 — reserved

This bit is not used and is always read as logic zero.

KWIF — Key Wake-up Interrupt Flag

When KWIE*x* bit in the KWIEN register is set, the falling edge at KWI*x* pin sets KWIF bit. If KWIE bit and this bit are set, an interrupt is generated. This bit is a read-only bit, and clearing KWIF is accomplished by writing a logic one to the RKWIF bit. Reset clears this bit.

RIRQ1 — Reset IRQ1 Flag

The RIRQ1 bit is a write-only bit and is always read as logic zero. Writing a logic one to this bit clears the IRQ1F bit and writing logic zero to this bit has no effect.

RIRQ2 — Reset IRQ2 Flag

The RIRQ2 bit is a write-only bit and always read as logic zero. Writing a logic one to this bit clears the IRQ2F bit and writing a logic zero to this bit has no effect.

Bit 1 — reserved

This bit is not used and is always read as logic zero.

RKWIF — Reset KWI Flag

The RKWIF bit is a write-only bit and is always read as logic zero. Writing a logic one to this bit clears the KWIF bit and writing a logic zero to this bit has no effect.

SECTION 5
LOW-POWER MODES

The MCU has two power-saving modes, STOP and WAIT. Flowcharts of these modes are shown in Figure 5-2.

5.1 STOP

The STOP instruction places the MCU in its lowest power consumption mode. In STOP Mode, the internal main oscillator OSC is turned off, halting all internal processing, including timer operations (timer 1, timer 2, and COP watchdog timer). Sub oscillator XOSC does not stop oscillating. Therefore, if XOSC is used as the clock source for COP, COP is still functional in STOP Mode. See **SECTION 7 OSCILLATORS/CLOCK DISTRIBUTIONS.**

During the STOP mode, the timer prescaler is cleared. The I bit in the CCR is cleared to enable external interrupts. All other registers and memory remain unaltered. All input/output lines remain unchanged. The processor can be brought out of the STOP mode only by $\overline{\text{RESET}}$ or an interrupt from $\overline{\text{IRQ1}}$, $\overline{\text{IRQ2}}$, $\overline{\text{KWI}}$, SSPI (slave mode only), or TBI. See **SECTION 7 OSCILLATORS/CLOCK DISTRIBUTIONS.**

5.2 WAIT

The WAIT instruction places the MCU in a low-power consumption mode, but the WAIT mode consumes more power than the STOP mode. All CPU action is suspended, but on-chip peripherals and oscillators remain active. Any interrupt or reset (including a COP reset) will cause the MCU to exit the WAIT mode.

During the WAIT mode, the I bit in the CCR is cleared to enable interrupts. All other registers, memory, and input/output lines remain in their previous state. The timers may be enabled to allow a periodic exit from the WAIT mode. WAIT mode must be exited and the COP must be reset to prevent a COP timeout.

The reduction of power in the WAIT mode depends on how many of the on-chip peripheral's clocks can be shut down. Therefore, the amount of power that will be consumed is very dependant on the application and it would be prohibitive to test all parts for all variations. For these reasons, the values stated in the data sheet reflect typical application conditions after initial characterization of silicon.

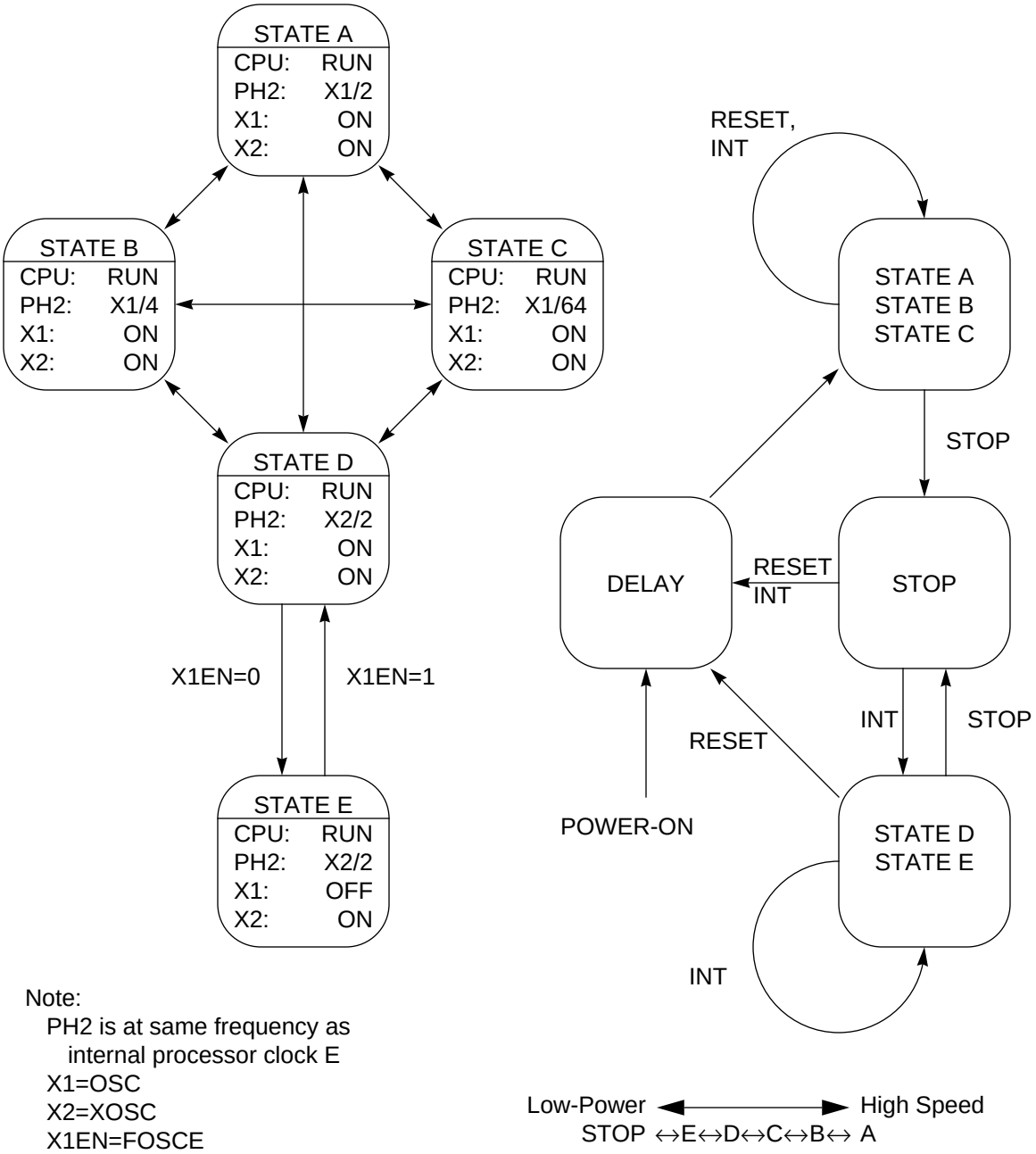


Figure 5-1. Clock State and STOP Recovery/POR Delay Diagrams

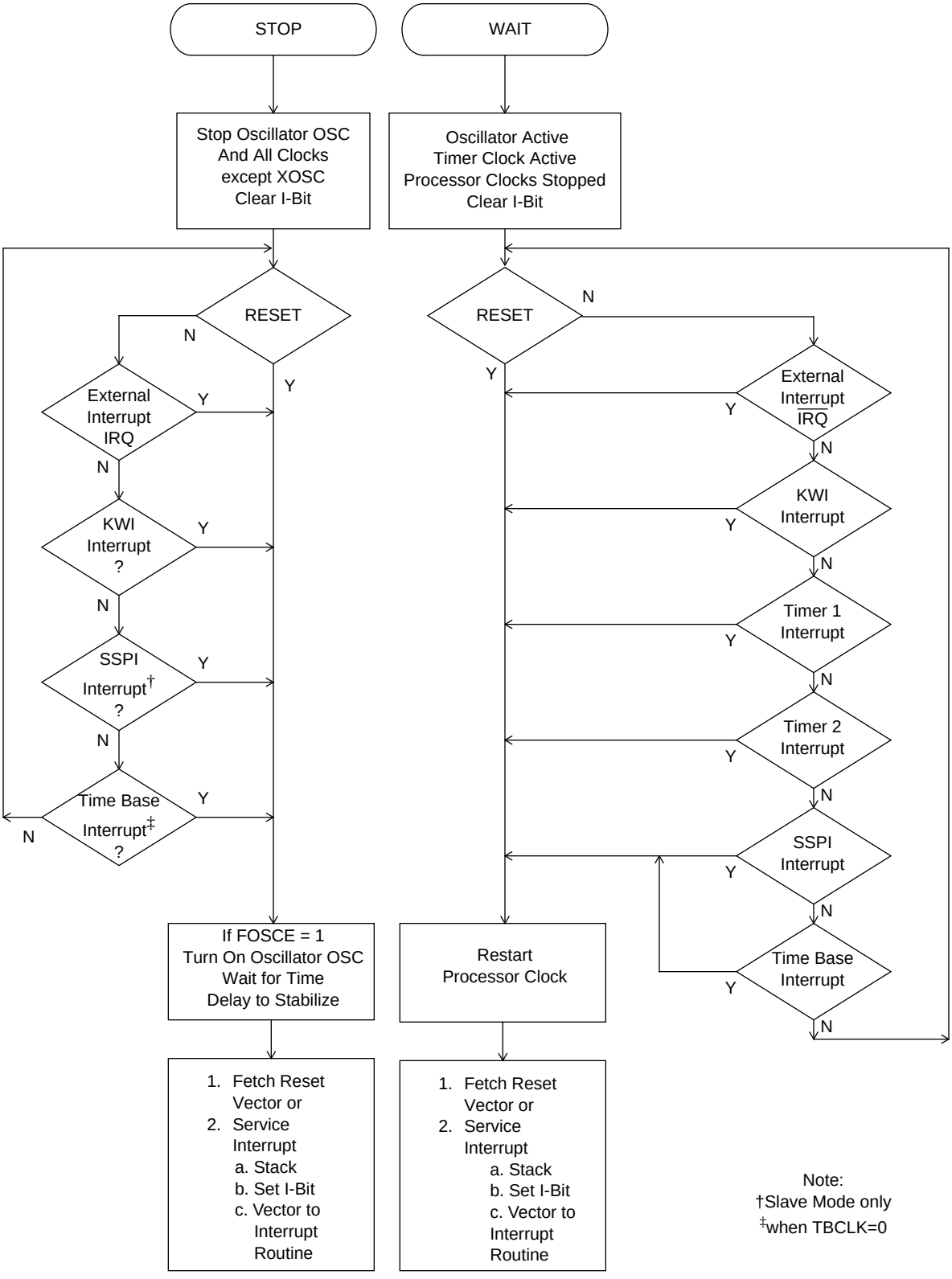


Figure 5-2. STOP/WAIT Flowcharts

SECTION 6
PARALLEL I/O

The MCU has five parallel ports: A, B, C, D, and E. Ports A and C have eight I/O pins, port B has eight input-only pins, port D has seven output-only pins, and port E has eight output-only pins. Most of these 39 I/O pins serve multiple purposes depending on the configuration of the MCU system. The configuration is in turn controlled by hardware mode selection as well as internal control registers.

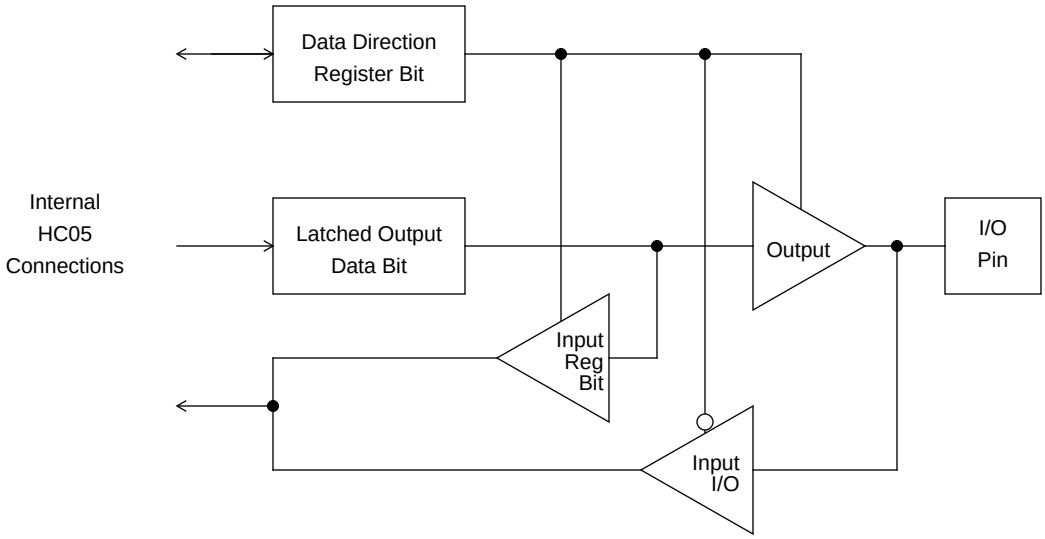


Figure 6-1. Port I/O Circuitry for One Bit

6.1 Port A

Port A is an 8-bit bidirectional general purpose port. The data direction of a port A pin is determined by its corresponding DDRA bit.

When a port A pin is programmed as an output by the corresponding DDRA bit, data in the PORTA data register becomes output data to the pin. This data is returned when PORTA register is read.

Open drain or CMOS outputs are selected by AWOMH and AWOML bits in the WOM1 register. If the AWOMH bit is set, the P-channel drivers of bit 7 through bit 4 output buffers are disabled (open drain). If the AWOML bit is set, the P-channel drivers of bit 3 through bit 0 output buffers are disabled (open drain).

When a bit is programmed as input by the corresponding DDRA bit, the pin level is read by the CPU.

Port A has optional pull-up resistors. When the RAH or RAL bit in the RCR1 is set, pull-up resistors are attached to the upper four bits or lower four bits of port A pins, respectively. (The typical resistor values are 50 KΩ at V_{DD} = 3 V.) When a pin outputs a low level, the pull-up resistor is disconnected regardless of RAH or RAL bit state.

6.1.1 Port A Data Register (PORTA)

	B7	B6	B5	B4	B3	B2	B1	B0	
$0000	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0	PORTA
RESET:	U	U	U	U	U	U	U	U	

READ: anytime
(returns pin level if DDR set to input; returns output data latch if DDR set to output)

WRITE: anytime
(data stored in an internal latch; drives pin only if DDR set for output)

RESET: becomes high-impedance inputs

6.1.2 Port A Data Direction Register (DDRA)

	B7	B6	B5	B4	B3	B2	B1	B0	
option map $0000	DDRA7	DDRA6	DDRA5	DDRA4	DDRA3	DDRA2	DDRA1	DDRA0	DDRA
RESET:	0	0	0	0	0	0	0	0	

READ: anytime (when OPTM = 1)

WRITE: anytime (when OPTM = 1)

RESET: cleared to $00 (all general purpose I/O configured for input)

DDRA*x* — Port A Data Direction Register bit *x*

0 = configure I/O pin PA*x* to input
1 = configure I/O pin PA*x* to output

6.2 Port B

Port B pins serve two basic functions: $\overline{\text{KWI}}$ input pins and general purpose input pins.

Each $\overline{\text{KWI}}$ input is enabled or disabled by the corresponding KWIE*x* bit in the KWIEN register, and the usage of the $\overline{\text{KWI}}$ input does not affect the general purpose input function.

Port B pin states may be read anytime regardless of the configurations. Since there is no output drive logic associated with port B, there is no DDRB register and the write to the PORTB register has no meaning.

Pull-up resistors are provided for both upper and lower four bits of port B pins which are controlled by the RBH and RBL bits, respectively, in the RCR1 register. (The typical resistor values are 50 KΩ at V_{DD} = 3 V.)

6.2.1 Port B Data Register (PORTB)

	B7	B6	B5	B4	B3	B2	B1	B0	
$0001	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0	PORTB
RESET:	U	U	U	U	U	U	U	U	

READ: anytime
(returns pin level)

WRITE: has no meaning or effect

RESET: unaffected; always input port

6.3 Port C

Port C pins share functions with several on-chip peripherals. A pin function is controlled by the enable bit of each associated peripheral.

Bit 7 and bit 6 of port C are general purpose I/O pins and IRQ input pins. The DDRC7 and DDRC6 bits determine whether the pin states or the data latch states should be read by the CPU. Since IRQ1F or IRQ2F can be set by either the pins or the data latches, when using IRQs, be sure to clear the flags by software before enabling the IRQ1E or IRQ2E bits.

When configured for output port, PC6 and PC7 are open drain only. When V_{DD} output is required, pull-up resistor must be enabled.

The PC5 pin is a general purpose I/O pin and the direction of the pin is determined by the DDRC5 bit in the data direction register C (DDRC). When the event output (EVO) is enabled, the PC5 is configured as an event output pin and the DDRC5 bit has meaning only for the read of PC5 bit in the PORTC register; if the DDRC5 is set, the PC5 data latch is read by the CPU. Otherwise, PC5 pin level (EVO state) is read. When EVO is disabled, DDRC5 bit decides the idling state of EVO (if DDRC5 = 1). This PC5/EVO output has the capability to drive 10 mA source current when ($V_{OH} \geq V_{DD}$ - 0.8V).

The PC4 and PC3 pins share functions with the timer input pins (EVI and TCAP). These bits are not affected by the usage of timer input functions and the directions of pins are always controlled by the DDRC4 and DDRC3 bits. Also, the DDRC4 and DDRC3 bits determine whether the pin states or data latch states should be read by the CPU.

NOTE

Since the TCAP pin is shared with the PC3 I/O pin, changing the state of the PC3 DDR or data register can cause an unwanted TCAP interrupt. This can be handled by clearing the ICIE bit before changing the configuration of PC3, and clearing any pending interrupts before enabling ICIE.

NOTE

Since the EVI pin is shared with the PC4 I/O pin, DDRC4 should always be cleared whenever EVI is used. EVI should not be used when DDRC4 is high.

The PC2 through PC0 pins are shared with the simple serial peripheral interface (SSPI). When the SSPI is not used (SPE = 0), DDRC2 through DDRC0 bits control the directions of the pins, and when the SSPI is enabled, the pins are configured as serial clock output or input (SCK), serial data output (SDO), and serial data input (SDI). The direction of the SCK depends on the MSTR bit in the SPCR. When PORTC is read, the value read will be determined by the data direction register. When the port is configured for input (DDRC2, DDRC1, or DDRC0 equal to logic zero) the pin state is read. When the port is configured for output (DDRC2, DDRC1, or DDRC0 equal to logic zero), output data latch is read.

Each port C pin has a pull-up resistor option which is controlled by the corresponding RCR2 register bit. (The typical resistor values are 10 KΩ at V_{DD} = 3 V.) When a pin outputs low, the resistor is disconnected regardless of an RCR2 register bit being set.

Bit 5 through bit 0 have open drain or CMOS output options, which are controlled by the corresponding WOM2 register bits. These open drain or CMOS output options may be selected for either the general purpose output ports or the peripheral outputs (EVO, SCK, and SDO).

6.3.1 Port C Data Register (PORTC)

	B7	B6	B5	B4	B3	B2	B1	B0	
$0002	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0	PORTC
RESET:	U	U	U	U	U	U	U	U	

READ: anytime
(returns pin level if DDR set to input;
returns output data latch if DDR set to output)

WRITE: anytime
(data stored in an internal latch; drives pin only if DDR set for output
writes do not change pin state when pin configured for SDO, SCK, and EVO peripheral output.)

RESET: becomes high-impedance inputs

6.3.2 Port C Data Direction Register (DDRC)

option map $0002	B7	B6	B5	B4	B3	B2	B1	B0	
	DDRC7	DDRC6	DDRC5	DDRC4	DDRC3	DDRC2	DDRC1	DDRC0	DDRC
RESET:	0	0	0	0	0	0	0	0	

READ: anytime (when OPTM = 1)

WRITE: anytime (when OPTM = 1)

RESET: cleared to $00 (all general purpose I/O configured for input)

DDRC*x* — Port C Data Direction Register bit *x*

The timer and SSPI force the I/O state to be an output for each port C line associated with an enabled output function such as SDO and EVO. For these cases, the data direction bits will not change.

0 = configure I/O pin PC*x* to input
1 = configure I/O pin PC*x* to output

6.4 Port D

Port D pins serve one of two basic functions depending on the MCU mode selected: LCD frontplane and backplane driver outputs, or general purpose output pins. Since port D is an output-only port, there is no DDRD register. In place of DDRD is port D MUX control register (PDMUX). Bits 7 through 4 of this register control the port/LCD muxing of port D bits 7 through 4 respectively on a bit-wide basis. These bits are cleared on reset, and writing a logic one to any bit will turn that pin into a port output. This function is superceded by the PDH bit in the LCD control register. When PDH is set, the upper four bits of port D become port outputs regardless of the state of the PDMUX bits.These outputs have the capability to drive 10 mA sink current when ($V_{OL} \leq V_{SS}$ + 0.8 V).

On reset, all port D outputs are disconnected from the pins and the port D data latches are set to a logic one.

The pin connections of the lower three bits of port D depend on the LCD duty selection by the DUTY1 and DUTY0 bits in the LCDCR. When the LCD duty is not 1/4, the unused backplane driver(s) is (are) replaced by the port D output pin(s) automatically.

If DWOMH bit or DWOML bit in the WOM1 register is set, the P-channel drivers of output buffers at the upper four bits or lower three bits, respectively, are disabled (open drain mode). These open drain controls do not apply to the pins which are configured as frontplane or backplane driver outputs.

6.4.1 Port D Data Register (PORTD)

	B7	B6	B5	B4	B3	B2	B1	B0	
$0003	PD7	PD6	PD5	PD4	PD3	PD2	PD1	1	PORTD
RESET:	1	1	1	1	1	1	1	1	

READ: anytime
(returns output data latch; bit 0 is always read logic one)

WRITE: anytime
(writes do not change pin state when configured for LCD driver output)

RESET: all bits set to logic one and output ports disconnected from the pins
(LCD is enabled on RESET

6.4.2 Port D MUX Register (PDMUX)

option map $0003	B7	B6	B5	B4	B3	B2	B1	B0	
	PDM7	PDM6	PDM5	PDM4	0	0	0	0	PDMUX
RESET:	0	0	0	0	0	0	0	0	

READ: anytime (when OPTM = 1, bits 3 through 0 always read logic zero)

WRITE: anytime (writes have no affect if PDH is set)

RESET: all bits cleared (LCD is enabled)

PDM*x* Port D MUX Control bit *x*

0 = configure pin PD*x* to LCD
1 = configure pin PD*x* to output

6.5 Port E

Port E pins serve one of two basic functions depending on the MCU mode selected: LCD frontplane driver outputs or general purpose output pins. Since port E is an output-only port, there is no DDRE register. In place of DDRE is port E MUX control register (PEMUX). Bits 7 through 0 of this register control the port/ LCD muxing of port E bits 7 through 0 respectively on a bit-wide basis. These bits are cleared on reset, and writing a logic one to any bit will turn that pin into a port output. This function is superceded by the PEH and PEL bits in the LCD control register. When PEH is set, the upper four bits of port E become port outputs regardless of the state of the PEMUX bits. Likewise, when PEL is set, the lower four bits of port E become outputs. These outputs have the capability to drive 10 mA sink current when ($V_{OL} \leq V_{SS}$ + 0.8 V).

On reset, all port E outputs are disconnected from the pins and the port E data latches are set to logic one.

If EWOMH bit or EWOML bit in the WOM1 register is set, the P-channel driver of output buffers at the upper or lower four bits, respectively, are disabled (open drain mode). These open drain controls do not apply to the pins which are configured as frontplane driver outputs.

6.5.1 Port E Data Register (PORTE)

	B7	B6	B5	B4	B3	B2	B1	B0	
$0004	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0	PORTE
RESET:	1	1	1	1	1	1	1	1	

READ: anytime
(returns output data latch)

WRITE: anytime
(writes do not change pin state when configured for LCD driver output)

RESET: all bits set to logic one and output ports disconnected from the pins
(LCD is enabled on RESET)

6.5.2 Port E MUX Register (PEMUX)

option map $0004	B7	B6	B5	B4	B3	B2	B1	B0	
	PEM7	PEM6	PEM5	PEM4	PEM3	PEM2	PEM1	PEM0	PEMUX
RESET:	0	0	0	0	0	0	0	0	

READ: anytime (when OPTM = 1)

WRITE: anytime (writes have no affect if PEH/L is set)

RESET: all bits cleared (LCD is enabled)

PEM*x* Port E MUX Control bit *x*

0 = configure pin PE*x* to LCD
1 = configure pin PE*x* to output

SECTION 7
OSCILLATORS/CLOCK DISTRIBUTIONS

There are two oscillator blocks: OSC and XOSC. Several combinations of the clock distributions are allowed for the modules in the MC68HC(7)05L16. Refer to the following block diagram.

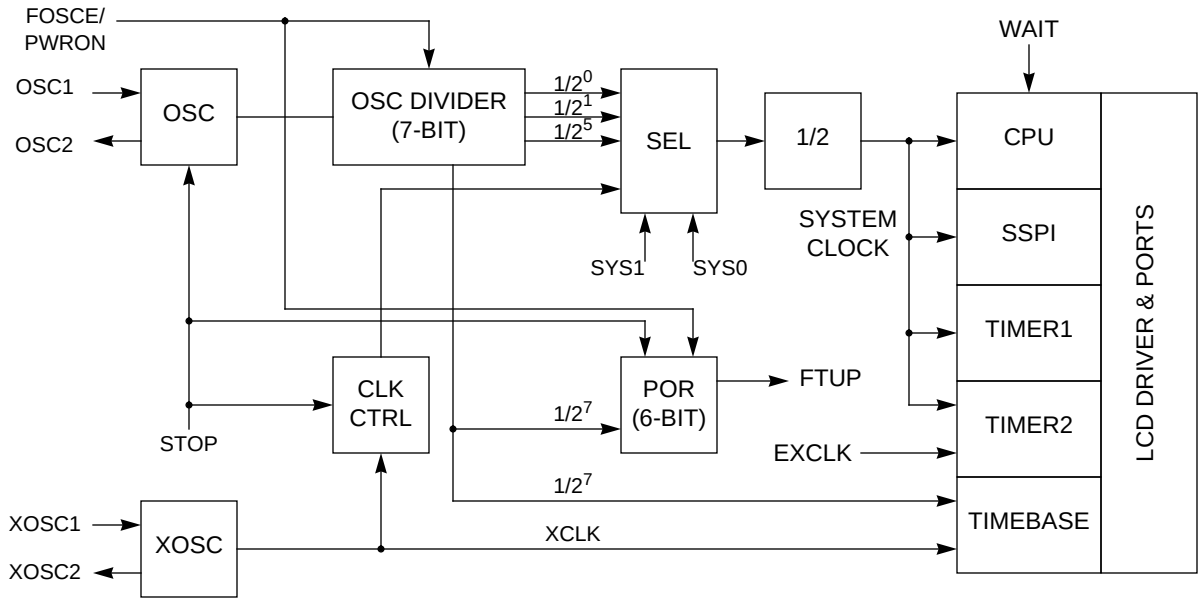


Figure 7-1. Clock Signal Distribution

7.1 OSC Clock Divider and POR Counter

The OSC clock is divided by a 7-bit counter which is used for the system clock, time base, and POR counter. Clocks divided by 2, 4, and 64 are available for the system clock selections and a clock divided by 128 is provided for the time base and POR counter.

The POR counter is a 6-bit clock counter that is driven by the OSC divided by 128. The overflow of this counter is used for setting FTUP bit, release of power-on reset (POR), and resuming operation from STOP mode.

The 7-bit divider and POR counter are initialized to $0078 by the following conditions.

- Power-on detection
- When FOSCE bit is cleared

7.2 System Clock Control

The system clock is provided for all internal modules except time base.

Both OSC and XOSC are available as the system clock source. The divide ratio is selected by the SYS1 and SYS0 bits in the MISC register.

Table 7-1. System Bus Clock Frequency Selection

SYS1	SYS0	Divide Ratio	CPU Bus Frequency (Hz)		
			OSC = 4.0 M	OSC = 4.1943 M	XOSC = 32.768 K
0	0	OSC Divided by 2	2.0 M	2.0972 M	—
0	1	OSC Divided by 4	1.0 M	1.0486 M	—
1	0	OSC Divided by 64	62.5 K	65.536 K	—
1	1	XOSC Divided by 2	—	—	16.384 K

By default, OSC/64 is selected on reset.

NOTE

Do not switch the system clock to XOSC (SYS1:0 = 11) when XOSC clock is not available. XOSC clock is available when STUP flag is set.

NOTE

Do not switch the system clock to OSC (SYS1:0 = 00, 01, or 10) when OSC clock is not available. OSC clock is available when FTUP flag is set.

7.3 OSC and XOSC

The secondary oscillator (XOSC) runs continuously after power-up. The main oscillator (OSC) can be stopped to conserve power via the STOP instruction or the FOSCE bit in the MISC register. The effects of restarting the OSC will vary depending on the current state of the MCU, including SYS0:1 and FOSCE.

7.3.1 OSC on Line

If the system clock is OSC, FOSCE should remain logic one. Executing the STOP instruction in this condition will halt OSC, put the MCU into a low-power mode and clear the 6-bit POR counter. The 7-bit divider is not initialized. Exiting STOP with external IRQ or reset re-starts the oscillator. When the POR counter overflows, internal reset is released and execution can begin. The stabilization time will vary between 8064 and 8192 counts.

NOTE

Exiting STOP with external reset will always return the MCU to the state as defined by the default register definitions, for example, SYS0:1 = 1:0, FOSCE = 1.

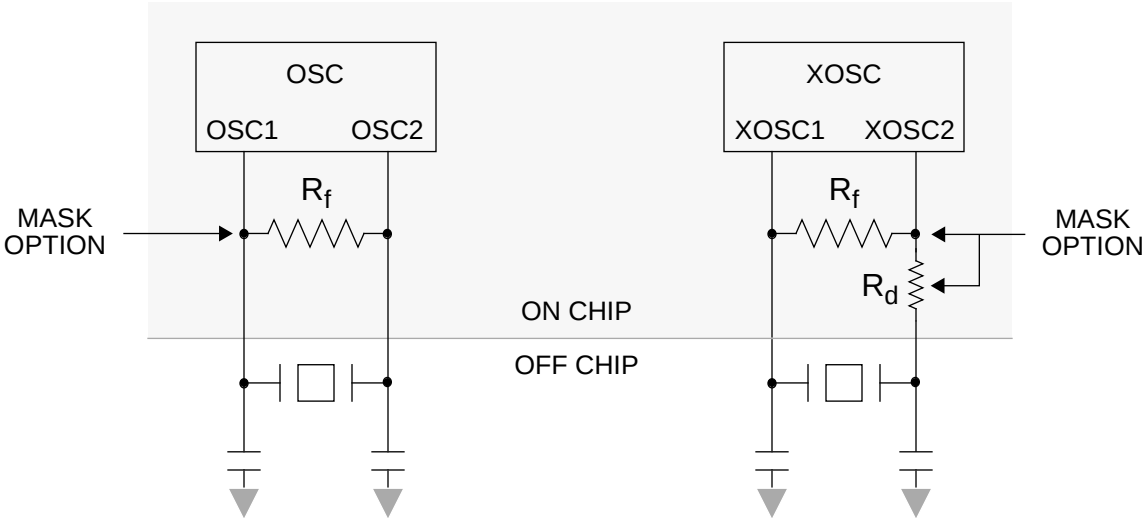


Figure 7-2. OSC1, OSC2, XOSC1, and XOSC2 Mask Options

7.3.2 XOSC on Line

If XOSC is the system clock (SYS0:1 = 1:1), OSC can be stopped either by the STOP instruction or by clearing the FOSCE bit.

The sub oscillator (XOSC) never stops except during power down. This clock may also be used as the clock source of the system clock and time base. FTUP bit indicates that XOSC clock is available.

OSC and XOSC pins have options for feedback and damping resistor implementations. These options are set through mask option and may be read through the MOSR register.

NOTE

When XOSC is not used, XOSC1 input pin should be connected to RESET pin.

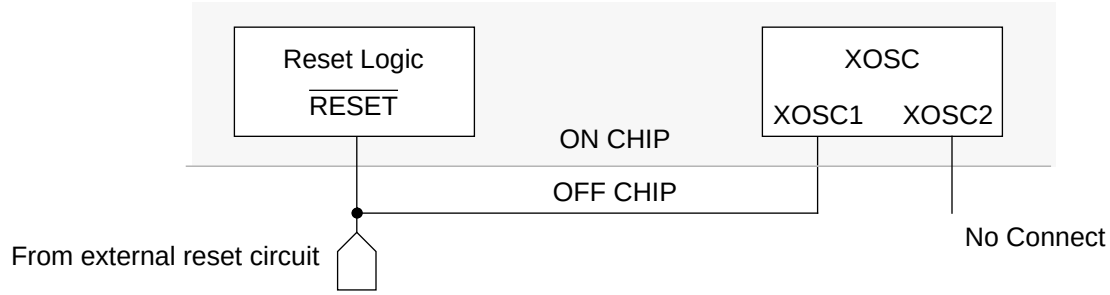


Figure 7-3. Unused XOSC1 pin

7.3.2.1 XOSC with FOSCE = 1

If the system clock is XOSC and FOSCE = 1, executing the STOP instruction will halt OSC, put the MCU into a low-power mode and clear the 6-bit POR counter. The 7-bit divider is not initialized. Exiting STOP with external IRQ re-starts the oscillator; however, execution begins immediately using XOSC. When the POR counter overflows, FTUP is set signaling that OSC is stable and OSC can be used as the system clock. The stabilization time will vary between 8064 and 8192 counts.

7.3.2.2 XOSC with FOSCE = 0

If XOSC is the system clock, clearing FOSCE will stop OSC and preset the 7-bit divider and 6-bit POR counter to $0078. Execution will continue with XOSC, and when FOSCE is set again, OSC will re-start. When the POR counter overflows, FTUP is set, signaling that OSC is stable and OSC can be used as the system clock. The stabilization time will be 8072 counts.

7.3.2.3 XOSC with FOSCE = 0 and STOP

If XOSC is the system clock and FOSCE is cleared, further power reduction can be achieved by executing the STOP instruction. In this case, OSC is stopped, the 7-bit divider and 6-bit POR counter are preset to $0078 (since FOSCE = 0) and execution is halted. Exiting STOP with external IRQ does not re-start the OSC; however, execution begins immediately using XOSC. OSC may be re-started by setting FOSCE, and when the POR counter overflows, FTUP will be set signaling that OSC is stable and can be used as the system clock. The stabilization time will be 8072 counts.

7.3.2.4 STOP and WAIT Modes

During STOP mode, the main oscillator (OSC) is shut down and the clock path from the second oscillator (XOSC) is disconnected. All modules except time base are halted. Entering STOP mode clears FTUP flag in the MISC register, and initializes POR counter. The STOP mode is exited by $\overline{\text{RESET}}$, $\overline{\text{IRQ1}}$, $\overline{\text{IRQ2}}$, KWI, SSPI (slave mode), or time base interrupt.

If OSC is selected as system clock source during STOP mode, CPU resumes after the overflow of POR counter and this overflow also sets FTUP status flag.

If XOSC is selected as system clock source during STOP mode, no stop recovery time is required for exiting STOP mode because XOSC never stops. Re-start of the main oscillator depends on the FOSCE bit.

During WAIT mode, only the CPU clocks are halted and the peripheral modules are not affected. The WAIT mode is exited by $\overline{\text{RESET}}$ and any interrupts.

Table 7-2. Recovery Time Requirements

BEFORE RESET OR INTERRUPT			POWER-ON RESET	EXTERNAL RESET	EXIT STOP MODE BY INTERRUPT
CPU CLOCK SOURCE	**STOP**	**FOSCE**			
—	—	—	WAIT	—	—
OSC (OSC ON)	OUT	1	—	NO WAIT	—
OSC (OSC OFF)	OUT	0*	—	WAIT	—
	IN	1	—	WAIT	WAIT
	IN†	0†	—	WAIT	WAIT
XOSC (OSC ON)	OUT	1	—	NO WAIT	—
XOSC (OSC OFF)	OUT	0	—	WAIT	—
	IN	1	—	WAIT	NO WAIT
	IN	0	—	WAIT	NO WAIT

NOTE:
* This case has no meaning for the applications
† This case never occurs

7.4 Time Base

Time base is a 14-bit up-counter which is clocked by XOSC input or OSC input divided by 128. TBCLK bit in the TBCR1 register selects the clock source.

This 14-bit divider is initialized to $0078 only upon power-on reset (POR). After counting 8072 clocks, the STUP bit in the MISC register is set.

The divided clocks from the time base are used for LCDCLK, STUP, TBI, and COP.

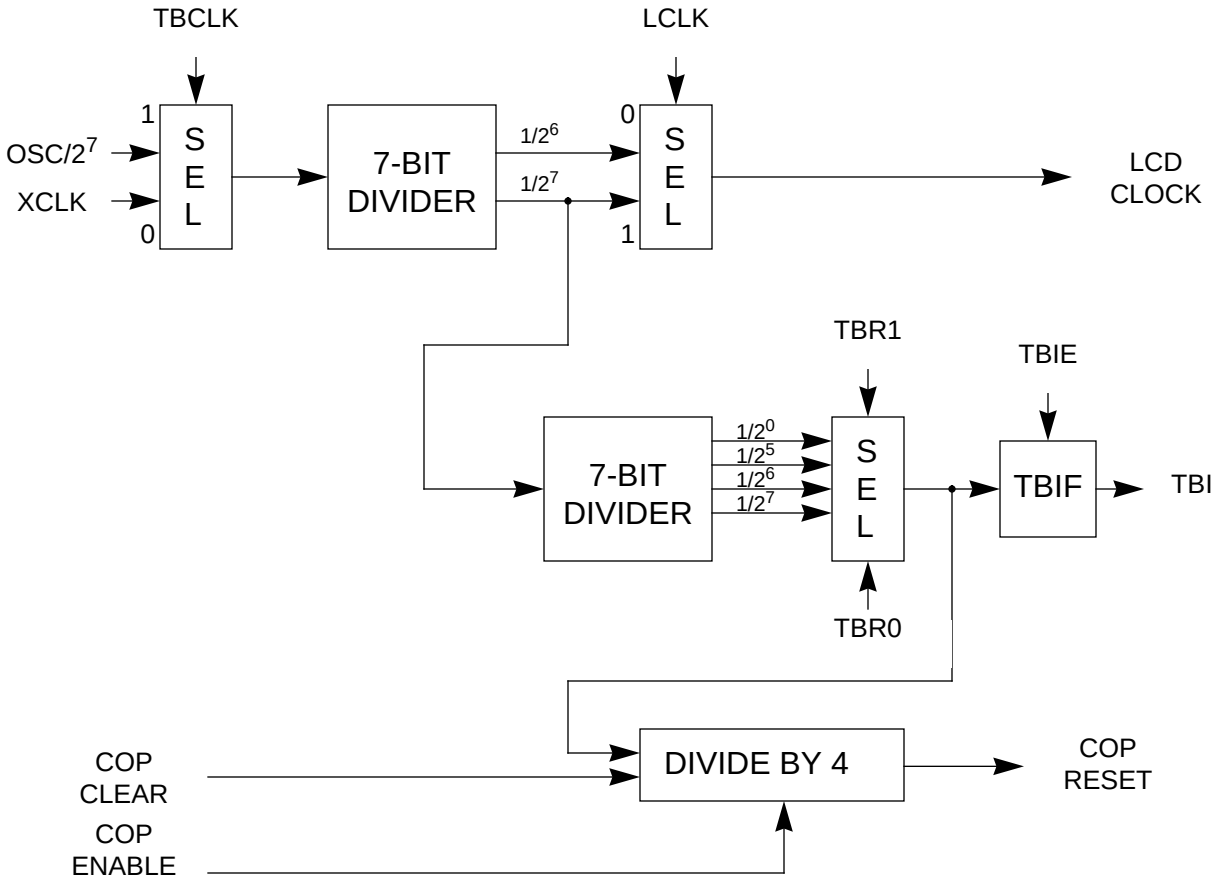


Figure 7-4. Time Base Clock Divider

7.4.1 LCDCLK

The clocks divided by 64 and 128 are used as LCD clocks at the LCD driver module, and clocks are selected by the LCLK bit in the TBCR1.

7.4.2 STUP

Time base divider is initialized to $0078 by the power-on detection and when the count reaches 8072, STUP flag in the MISC register is set. Once STUP flag is set, it is never cleared until power down.

7.4.3 TBI

Time base interrupts may be generated every 0.5, 0.25, 0.125, or 0.0039 seconds with 32.768 kHz crystal at XOSC pins.

Time base interrupt flag (TBIF) is set every period and interrupt is requested if the enable bit (TBIE) is set. The clock divided by 128, 4096, 8192, or 16,384 is used to set TBIF, and this clock is selected by the TBR1 and TBR0 bits in the TBCR2 register.

Table 7-3. Time Base Interrupt Frequency

TBCR2		Divide Ratio	Frequency (Hz)		
TBR1	TBR0		OSC = 4.0 MHz	OSC = 4.1943 MHz	XOSC = 32.768 KHz
0	0	TBCLK / 128	244	256	256
0	1	TBCLK / 4096	7.63	8.00	8.00
1	0	TBCLK / 8192	3.81	4.00	4.00
1	1	TBCLK / 16384	1.91	2.00	2.00

7.4.4 COP

The computer operating properly (COP) watchdog timer is controlled by the COPE and COPC bits in the TBCR2 register.

The COP uses the same clock as TBI that is selected by the TBR1 and TBR0 bits. The TBI is divided by four and overflow of this divider generates COP timeout reset if the COP enable (COPE) bit is set. The COP timeout reset has the same vector address as POR and external $\overline{\text{RESET}}$. To prevent the COP timeout, the COP divider is cleared by writing a logic one to the COP clear (COPC) bit.

When the time base divider is driven by the OSC clock, clock for the divider is suspended during STOP mode or when FOSCE is a logic zero. This may cause COP period stretching or no COP timeout reset when processing errors occur. It is recommended that XOSC clock be used for the COP functions to avoid these problems.

When the time base (COP) divider is driven by the XOSC clock, the divider does not stop counting and the COPC bit must be triggered to prevent the COP timeout.

Table 7-4. COP Time-Out Period

TBCR2		**COP Period (ms)**					
TBR1	**TBR0**	**OSC = 4.0 MHz**		**OSC = 4.1943 MHz**		**XOSC = 32.768 KHz**	
		Min	**Max**	**Min**	**Max**	**Min**	**Max**
0	0	12.3	16.4	11.7	15.6	11.7	15.6
0	1	393	524	375	500	375	500
1	0	786	1048	750	1000	750	1000
1	1	1573	2097	1500	2000	1500	2000

7.4.5 Time Base Control Register 1 (TBCR1)

	B7	B6	B5	B4	B3	B2	B1	B0	
$0010	TBCLK	0	LCLK	0	0	0	T2R1	T2R0	TBCR1
RESET:	0	0	0	0	0	0	0	0	

READ: anytime

WRITE: anytime (only one write is allowed on bit 7 after reset)

TBCLK — Time Base Clock

The TBCLK bit selects time base clock source. This bit is cleared on reset. After reset, write to this bit is allowed only once.

0 = XOSC clock is selected
1 = OSC clock divided by 128 is selected

BIT 6 — Reserved

This bit is not used and always read as logic zero.

LCLK — LCD Clock

The LCLK bit selects clock for the LCD driver. This bit is cleared on reset.

0 = divide by 64 is selected
1 = divide by 128 is selected

Bits 4 through 2 — Reserved

These bits are not used and always read as logic zero.

T2R1/T2R0 — Timer 2 Prescale Rate select bits

T2R1 and T2R0 select timer 2 clock rate. See **Section 9.2** for more detail.

7.4.6 Time Base Control Register 2 (TBCR2)

	B7	B6	B5	B4	B3	B2	B1	B0	
$0011	TBIF	TBIE	TBR1	TBR0	RTBIF	0	COPE	COPC	TBCR2
RESET:	0	0	1	1	0	0	0	0	

READ: anytime
(bits 3 and 0 are write-only bits and always read as logic zero)

WRITE: anytime
(bit 7 is read-only bit and write has no effect; bit 1 is one-time write bit)

TBIF — Time Base Interrupt Flag

The TBIF bit is set every timeout interval of the time base counter. This bit is a read-only bit and is cleared by writing a logic one to the RTBIF bit. Reset clears TBIF bit. Time base interrupt period between reset and first TBIF depends on the time elapsed during reset, since time base divider is not initialized on reset.

TBIE — Time Base Interrupt Enable

The TBIE bit enables the time base interrupt capability. If TBIF = 1 and TBIE = 1, the time base interrupt is generated.

1 = time base interrupt requested when TBIF = 1
0 = time base interrupt is disabled

TBR1/TBR0 — Time Base Interrupt Rate Select

The TBR1 and TBR0 bits select one of four rates for the time base interrupt period. The TBS rate is also related to the COP timeout reset period. These bits are set to logic one on reset.

Table 7-5. Time Base Interrupt Frequency

TBCR2		Divide Ratio	Frequency (Hz)		
TBR1	TBR0		OSC = 4.0 MHz	OSC = 4.1943 MHz	XOSC = 32.768 KHz
0	0	TBCLK / 128	244	256	256
0	1	TBCLK / 4096	7.63	8.00	8.00
1	0	TBCLK / 8192	3.81	4.00	4.00
1	1	TBCLK / 16384	1.91	2.00	2.00

RTBIF — Reset TBS Interrupt Flag

The RTBIF bit is a write-only bit and is always read as logic zero. Writing logic one to this bit clears the TBIF bit and writing logic zero to this bit has no effect.

Bit 2 — Reserved

This bit is not used and is always read as logic zero.

COPE — COP Enable

When the COPE bit is logic one, COP reset function is enabled. This bit is cleared on reset (including COP timeout reset) and write to this bit is allowed only once after reset.

COPC — COP Clear

Writing logic one to COPC bit clears the 2-bit divider to prevent COP time out. (The COP timeout period depends on the TBI rate.) This bit is a write-only bit and returns to logic zero when read.

7.5 Miscellaneous Register (MISC)

	B7	B6	B5	B4	B3	B2	B1	B0	
$003E	FTUP	STUP	0	0	SYS1	SYS0	FOSCE	OPTM	MISC
RESET:	U	U	0	0	1	0	1	0	

READ: anytime

WRITE: Bits 7 through 4: no effect
Bits 3 through 1: anytime (care must be taken when changing these bits)
Bit 0: anytime

FTUP — OSC Time Up Flag
Power-on detection and clearing FOSCE bit clears this bit. This bit is set by the overflow of the POR counter. Reset does not affect this bit.
READ:
0 = during POR or OSC shut down
1 = OSC clock is available for the system clock

STUP — XOSC Time Up Flag
Power-on detection clears this bit. This bit is set after the time base has counted 8072 clocks. Reset does not affect this bit.
READ:
0 = XOSC is not stabilized or no signal on XOSC1–XOSC2 pins
1 = XOSC clock is available for the system clock

Bits 5 and 4 — Reserved
These bits are not used and always read as logic zero.

SYS 1:0 — System Clock Select
These two bits select the system clock source. On reset, the SYS1 and SYS0 bits are initialized to 1 and 0, respectively.

Table 7-6. System Bus Clock Frequency Selection

SYS1	SYS0	Divide Ratio	CPU Bus Frequency (Hz)		
			OSC = 4.0 M	OSC = 4.1943 M	XOSC = 32.768 K
0	0	OSC Divided by 2	2.0 M	2.0972 M	—
0	1	OSC Divided by 4	1.0 M	1.0486 M	—
1	0	OSC Divided by 64	62.5 K	65.536 K	—
1	1	XOSC Divided by 2	—	—	16.384 K

NOTE

Do not switch the system clock to XOSC (SYS1:0 = 11) when XOSC clock is not available. XOSC clock is available when STUP flag is set.

NOTE

Do not switch the system clock to OSC (SYS1:0 = 00, 01, or 10) when OSC clock is not available. OSC clock is available when FTUP flag is set.

FOSCE — Fast (Main) Oscillator Enable

The FOSCE bit controls the main oscillator activity. This bit should not be cleared by the CPU when the main oscillator is selected as the system clock source.

When this bit is cleared:

1. OSC is shut down.
2. 7-bit divider at the OSC input and POR counter are initialized to $0078.
3. FTUP flag is cleared.

When this bit is set:

1. Main oscillator starts again.
2. FTUP flag is set by the POR counter overflow (8072 clocks).

OPTM — Option Map Select

The OPTM bit selects one of two register maps at $0000-$000F. This bit is cleared on reset.

0 = Main register map is selected
1 = Option map is selected

7.6 V_{PP} Pin

In normal operation mode (SCM), this pin should be tied to V_{DD} level.

SECTION 8
SIMPLE SERIAL PERIPHERAL INTERFACE (SSPI)

The simple serial peripheral Interface (SSPI) of the MC68HC(7)05L16 is a master/ slave synchronous serial communication module.

SSPI uses a three-wire protocol: data input, data output, and serial clock. In this format, the clock is not being included in the data stream and must be provided as a separate signal.

When the SSPI is enabled (SPE = 1), bit 0 through bit 2 of the port C become SDI (serial data in), SDO (serial data out), and SCK (serial clocK) pins. The corresponding DDRC bit does not change the direction of the pin.

The MSTR bit decides the SSPI operation mode. SCK pin is configured as output in the master mode and configured as input in the slave mode.

The DORD bit in the serial peripheral control register (SPCR) selects the data transmission order. When DORD is set, LSB of serial data is shifted out/in first. When DORD is clear, serial data is shifted from/to the MSB.

Master serial clock speed is selected by SPR bit in the SPCR. An interrupt may be generated by the completion of a transfer.

8.1 Features

- Full Duplex, Three-Wire Synchronous Transfers
- Master or Slave Operation
- Programmable Data Transmission Order (LSB or MSB first)
- 1.05 MHz (maximum) Transmission Bit Frequency at 2.1 MHz CPU Bus
- Two Programmable Transmission Bit Rates
- End of Transmission Interrupt Flag
- Wake up from STOP Mode (slave mode only)

8.2 Functional Descriptions

In master mode, the clock start logic is triggered by CPU (detection of CPU write to the 8-bit shift register (SPDR)). The SCK is based on the internal processor clock. This clock is also used in the 3-bit counter and 8-bit shift register. See Figure 8-2**.**

When data is written to the 8-bit shift register of the master device, it is then shifted out to the SDO pin for application to the slave device. At the same time, data applied from the slave device via the SDI pin is shifted into the 8-bit shift register.

After 8-bit data is shifted in/out, SCK stops and SPIF is set. If SPIE is enabled, an interrupt request is generated. The slave device in the STOP mode wakes up by this interrupt. Further transfers (writes to SPDR) are inhibited while SPIF is a logic one.

The MASTER-SLAVE basic interconnection is illustrated in Figure 8-1

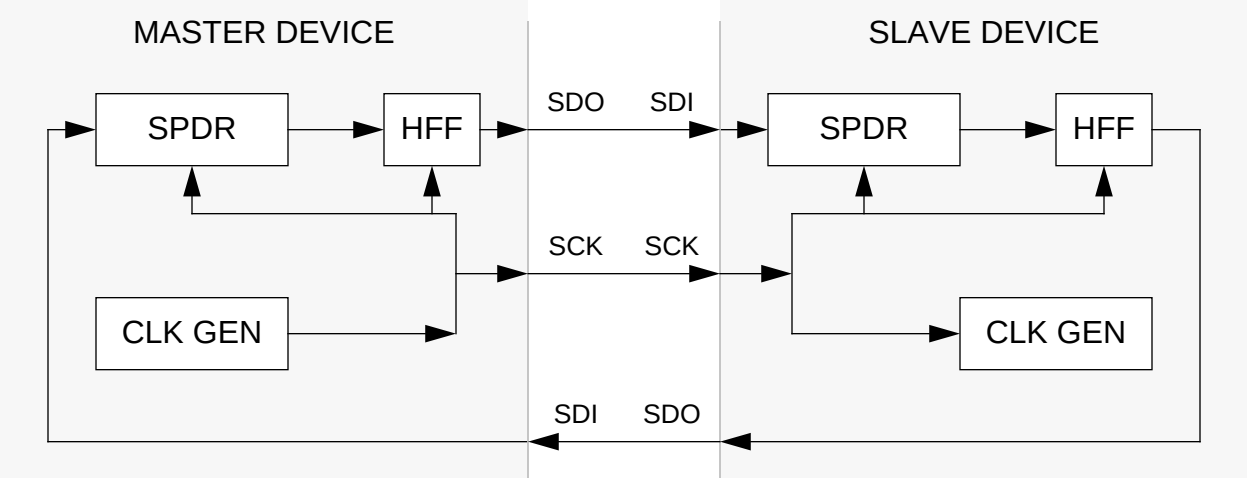


Figure 8-1. SSPI Master-Slave Interconnection

8.3 Internal Block Descriptions

This section describes the main blocks in the SSPI module.

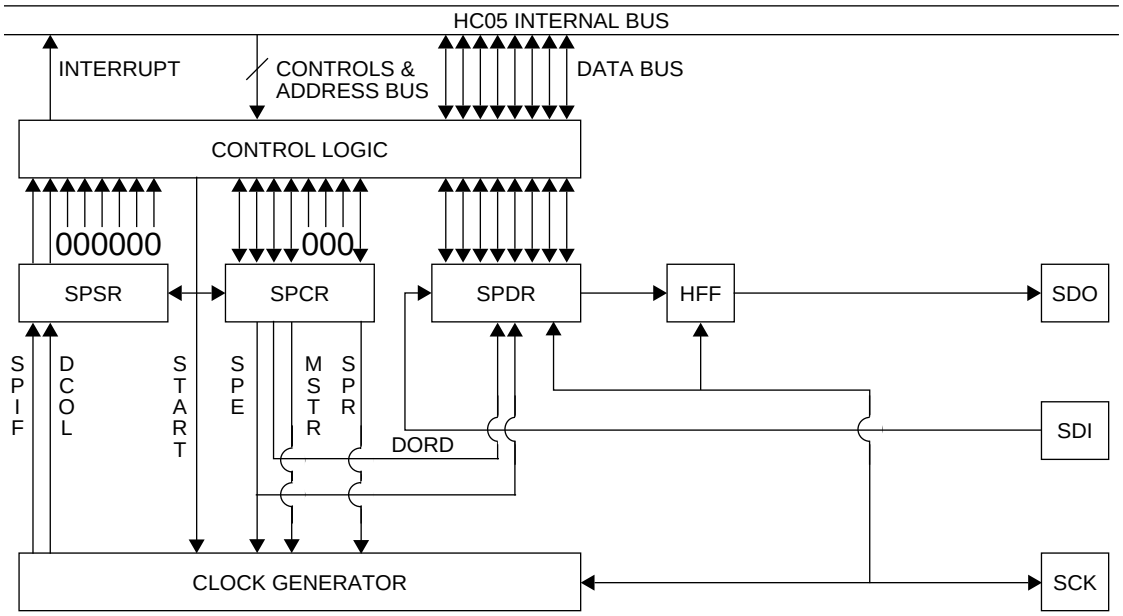


Figure 8-2. SSPI Block Diagram

8.3.1 Control

This block is an interface to HC05 internal bus and generates a START signal when a write to the SPDR is detected in the master mode. It also generates an interrupt request to CPU.

8.3.2 SPDR

This register is an 8-bit shift register called serial peripheral data register (SPDR). The DORD bit in the SPCR determines the bus connection between the internal data bus and SPDR. This register can be read and written by CPU.

8.3.3 SPCR

Serial peripheral control register (SPCR). The bits in this register control SSPI functions.

8.3.4 SPSR

Serial peripheral status register (SPSR). This register mainly works for setting flags such as SPIF and DCOL.

8.3.5 CLKGEN

In master mode, this block generates SCK when CPU writes to the data register (SPDR) and the clock rate is selected by SPR bit in control register.

In the slave mode, external clock from the SCK pin is used instead of master mode clock, and SPR has no affect.

This clock generator includes a 3-bit clock counter. Overflow of this counter sets SPIF.

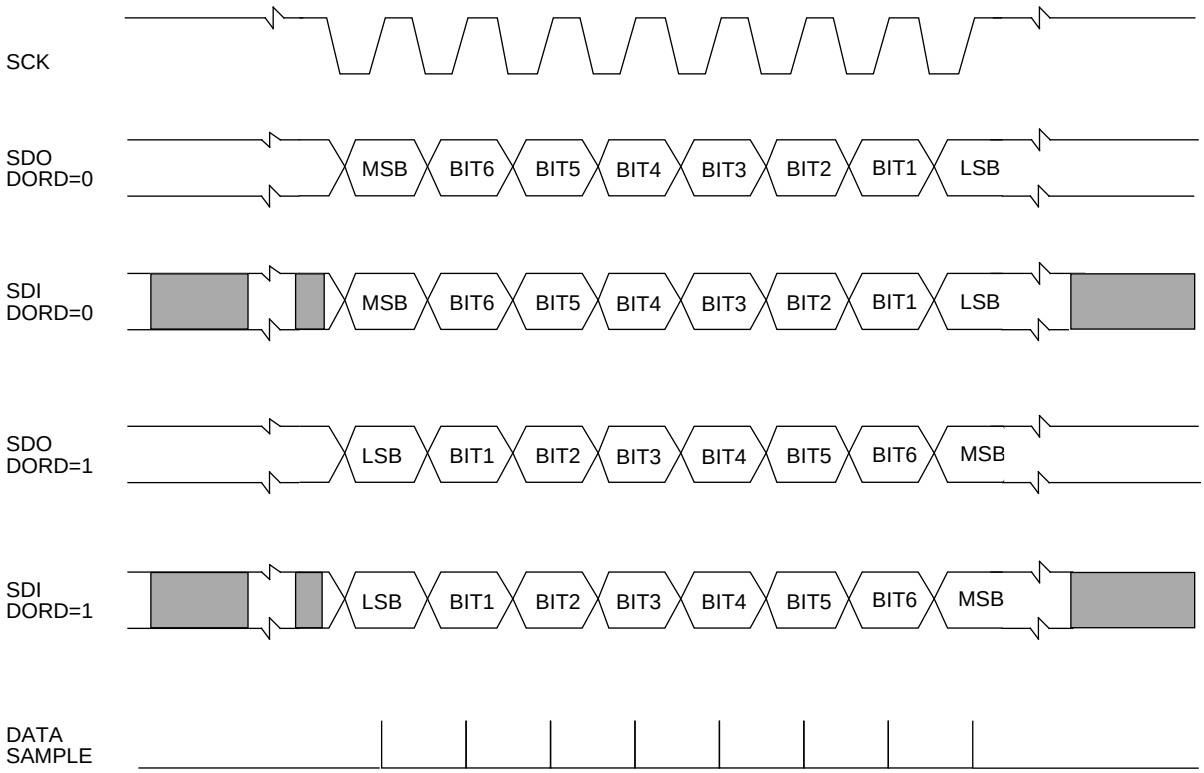


Figure 8-3. SSPI Clock-Data Timing Diagram

8.4 Signal Descriptions

Three basic signals (SDI, SDO, and SCK) are described in the following paragraphs. The relationship among SCK, SDI, and SDO is shown in Figure 8-3.

8.4.1 SSPI Data I/O (SDI and SDO)

The two serial data lines, SDI for input and SDO for output, are connected to PC0 and PC1, respectively, when SSPI is enabled (SPE = 1).

At the falling edge of SCK, a serial data bit is transmitted out of the SDO pin. At the rising edge of SCK, a serial data bit on the SDI pin is sampled internally.

When data is transmitted to other devices via the SDO line, the receiving data is shifted into the shift register through SDI pin. This implies full duplex transmission with both data-out and data-in synchronized with the same clock signal. Thus, the byte transmitted is replaced by the byte received and eliminates the need for separate transmit-empty and receiver-full status bits. A single status bit SPIF is used to signify the completion of data transfer.

8.4.2 Serial Clock (SCK)

SCK is used for synchronization of both input and output data streams through its SDI and SDO pins.

The MASTER and SLAVE devices are capable of exchanging a data byte during a sequence of eight clock pulses. Since the SCK is generated by the MASTER, SLAVE data transfer is accomplished by synchronization to SCK.

The MASTER generates the SCK through a circuit driven by the internal processor clock, and uses the SCK to latch incoming SLAVE device data on the SDI pin and shift out data to the SLAVE via SDO pin. The SPR bit in the SPCR of the MASTER selects the transmission clock rate.

The SLAVE device receives the SCK from the MASTER device, and uses the SCK to latch incoming MASTER device data on the SDI pin and shifts out data to the MASTER via SDO pin. The SPR bit in the SPCR of the SLAVE has no meaning.

NOTE

PC2/SCK should be at V_{DD} level before SSPI is enabled. This can be done with internal or external pull-up resistor, or by setting DDRC2 = 1 and PC2 = 1 prior to enabling the SSPI. Otherwise, the circuit will not initialize correctly.

8.5 Registers

There are three registers in the SSPI which provide control, status, and data storage functions. These three registers are serial peripheral control register (SPCR location $000A), serial peripheral status register (SPSR location $000B), and serial peripheral data register (SPDR location $000C).

8.5.1 Serial Peripheral Control Register (SPCR)

	B7	B6	B5	B4	B3	B2	B1	B0	
$000A	SPIE	SPE	DORD	MSTR	0	0	0	SPR	SPCR
RESET:	0	0	0	0	0	0	0	0	

READ: anytime

WRITE: should not be written during transmission

SPIE — SSPI Interrupt Enable

If the serial peripheral interrupt enable (SPIE) bit is set, an interrupt is generated when SPIF in the SPSR is set and I bit (interrupt mask bit) in the condition code register (CCR) is clear.

During STOP mode, an SSPI request is accepted only in the slave mode. Interrupt in the master mode will be pending until STOP mode is exited. STOP instruction does not change SPIF and SPIE.

0 = disable SSPI interrupt
1 = enable SSPI interrupt

SPE — SSPI Enable

When the SSPI enable (SPE) bit is set, the SSPI system is enabled and connected to the port C pins.

Clearing the SPE bit initializes all control logic in the SSPI modules and disconnects the SSPI from port C pins.

This bit is cleared on reset.

0 = disable SSPI
1 = enable SSPI

DORD — Data Transmission ORDer

When this bit is set, the data in the 8-bit shift register (SPDR) is shifted in/out from LSB. When this bit is cleared, the data in the SPDR is shifted in/out from MSB.

This bit is cleared on reset.

0 = MSB first
1 = LSB first

MSTR — MaSTeR Mode Select

The MSTR bit determines whether the device is in master mode or slave mode.

In the master mode (MSTR = 1), SCK pin is configured as an output and the serial clock is generated by the internal clock generator when the CPU writes to the SPDR.

In the slave mode (MSTR = 0), SCK pin is configured as an input and the serial clock is applied externally. This bit is cleared on reset.

0 = slave mode
1 = master mode

Bits 3 through — reserved

These bits are not used and are fixed to zero.

SPR — SSPI Clock Rate Select

This serial peripheral clock rate bit selects one of two bit rates of SCK. This bit is cleared on reset.

0 = internal processor clock divided by 2
1 = internal processor clock divided by 16

8.5.2 Serial Peripheral Status Register (SPSR)

	B7	B6	B5	B4	B3	B2	B1	B0	
$000B	SPIF	DCOL	0	0	0	0	0	0	SPSR
RESET:	0	0	0	0	0	0	0	0	

READ: anytime

WRITE: not applicable

SPIF — Serial Transfer Complete Flag

The serial peripheral data transfer complete flag bit notifies the user that a data transfer between the MC68HC(7)05L16 and external device has been completed. With the completion of the data transfer, the rising edge of the eighth pulse sets SPIF, and if SPIE is set, SSPI is generated. However, during STOP, the interrupt request is serviced only in slave mode. STOP execution never affects the SPIF flag or SPIE.

When SPIF is set, the ninth clock from the clock generator or from the SCK pin is inhibited.

Clearing the SPIF bit is done by a software sequence of accessing the SPSR while SPIF bit is set followed by accessing SPDR (8-bit shift register). This also clears DCOL bit mentioned below. While SPIF is set, all writes to the SPDR are inhibited until SPSR is read by the CPU.

SPIF bit is a read-only bit and cleared on reset.

0 = data transfer not complete
1 = data transfer complete

DCOL — Data COLlision

The data collision bit notifies the user that an attempt was made to write or read the serial peripheral data register while a data transfer was taking place with an external device. The transfer continues uninterrupted; therefore, a write will be unsuccessful, and a data read will be incorrect.

A data collision only sets the DCOL bit and does not generate SSPI interrupt. The DCOL bit indicates only the occurrence of data collision.

Clearing the DCOL bit is done by a software sequence of accessing the SPSR while SPIF is set followed by accessing the SPDR. Both SPIF and DCOL bits will be cleared by this sequence.

The DCOL bit is cleared on reset.

0 = no data collision
1 = data collision occurred

Bits 5 through 0 — reserved

These bits are not used and are fixed to zero.

8.5.3 Serial Peripheral Data Register (SPDR)

	B7	B6	B5	B4	B3	B2	B1	B0	
$000C	MSB	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	LSB	SPDR
RESET:	U	U	U	U	U	U	U	U	

READ: a read during transmission causes DCOL to be set

WRITE: a write during transmission causes DCOL to be set

The SPDR is used to transmit and receive data on the serial bus.

In master mode, a write to this register initiates transmission/reception of a data byte.

The SPIF status bit is set at the completion of data byte transmission. A write to the SPDR is inhibited while this register is shifting (a write attempt sets DCOL) or when the SPIF bit is set without reading SPSR. Data collision never affects the receiving and transmitting data in SPDR.

A write or read of the SPDR after accessing the SPSR with SPIF set will clear the SPIF and DCOL bits.

The ability to access SPDR is inhibited when a transmission is taking place. It is important to read the discussion defining DCOL and SPIF bits to understand the limits on using the SPDR.

When SSPI is not used (SPE = 0), the SPDR can be used as a general purpose data storage register.

8.6 Port Function

The SSPI shares I/O pins with PC0 through PC2. When SPE is set, PC0 becomes SDI input, PC1 becomes SDO output and PC2 becomes SCK. The direction of SCK depends on MSTR bit. Setting DDRC bits 0 through 2 does not change the data direction of the pin to output, but instead changes the source of data when PC0 through PC2 is read. If DDRC*x* = 1, Port C bit-*x* data latch is read and if DDRC*x* = 0, PORTC*x* pin level is read by the CPU.

When SPE is clear, SSPI is disconnected from I/O pins and PC0 through PC2 are used as general purpose I/O pins. See **Section 6.3.**

SECTION 9
TIMER SYSTEM

The MC68HC(7)05L16 has two timer modules: timer 1 with a 16-bit counter and timer 2 with an 8-bit counter. Timer 1 has one input pin (TCAP) and no output pin. Timer 2 has one input pin (EVI) and one output pin (EVO). The following block diagram describes the timer system of MC68HC(7)05L16.

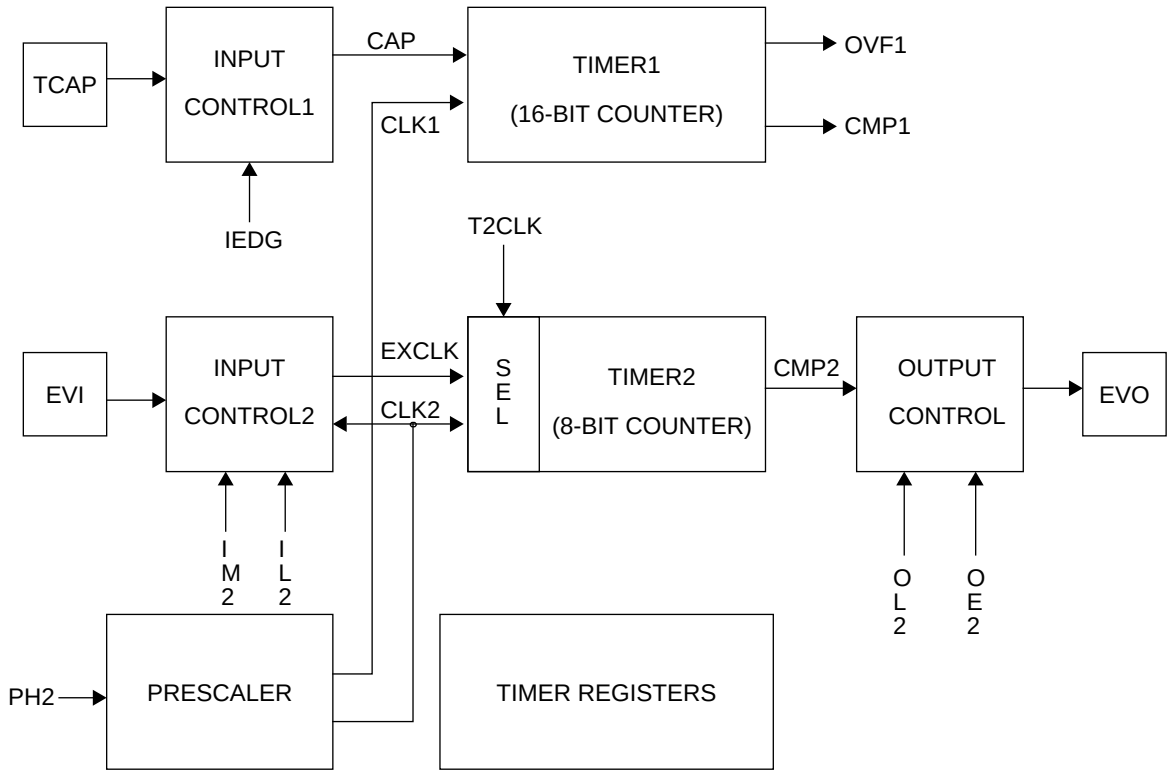


Figure 9-1. Timer System Block Diagram

9.1 Timer 1

The timer 1 consists of a 16-bit software-programmable counter driven by a fixed divide-by-four prescaler. This timer can be used for many purposes, including input waveform measurements while simultaneously generating an output compare interrupt. Pulse widths can vary from several microseconds to many seconds. See Figure 9-2**.**

Because the timer has a 16-bit architecture, each specific functional segment (capability) is represented by two registers. These registers contain the high and low byte of that functional segment. Generally, accessing the low byte of a specific timer function allows full control of that function; however, an access of the high byte inhibits that specific timer function until the low byte is also accessed.

NOTE

The I bit in the CCR should be set while manipulating both the high and low byte register of a specific timer function to ensure that an interrupt does not occur

---

.

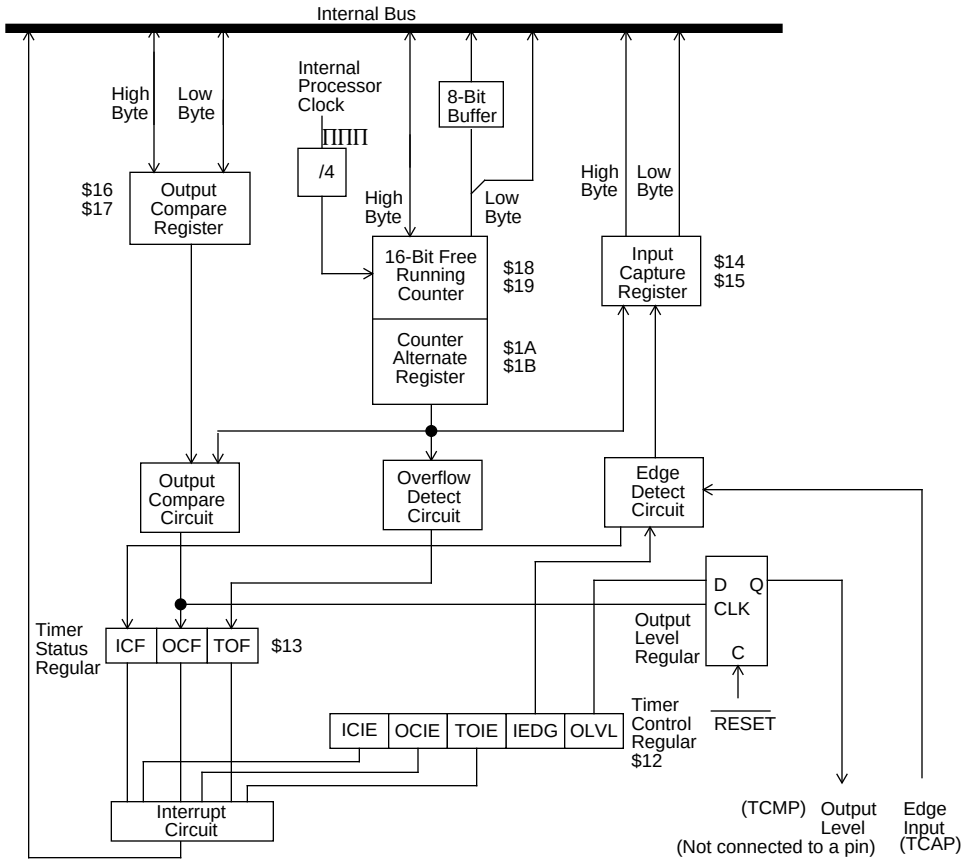


Figure 9-2. Timer 1 Block Diagram

9.1.1 Counter

The key element in the programmable timer is a 16-bit, free-running counter or counter register, preceded by a prescaler that divides the internal processor clock by four. The prescaler gives the timer a resolution of 2.0 microseconds if the internal bus clock is 2.0 MHz. The counter is incremented during the low portion of the internal bus clock. Software can read the counter at anytime without affecting its value.

The double-byte, free-running counter can be read from either of two locations, $18-$19 (counter register) or $1A-$1B (counter alternate register). A read from only the least significant byte (LSB) of the free-running counter ($19, $1B) receives the count value at the time of the read. If a read of the free-running counter or counter alternate register first addresses the most significant byte (MSB) ($18, $1A), the LSB ($19, $1B) is transferred to a buffer. This buffer value remains fixed after the first MSB read, even if the user reads the MSB several times. This buffer is accessed when reading the free-running counter or counter

alternate register LSB ($19 or $1B) and, thus, completes a read sequence of the total counter value. In reading either the free-running counter or counter alternate register, if the MSB is read, the LSB must also be read to complete the sequence.

The counter alternate register differs from the counter register in one respect: A read of the counter register MSB can clear the timer overflow flag (TOF). Therefore, the counter alternate register can be read at anytime without the possibility of missing timer overflow interrupts due to clearing of the TOF.

The free-running counter is configured to $FFFC during reset and is always a read-only register. During a power-on reset, the counter is also preset to $FFFC and begins running after the oscillator start-up delay. Because the free-running counter is 16 bits preceded by a fixed divided-by-4 prescaler, the value in the free-running counter repeats every 262,144 internal bus clock cycles. When the counter rolls over from $FFFF to $0000, the TOF bit is set. An interrupt can also be enabled when counter rollover occurs by setting its interrupt enable bit (TOIE).

9.1.2 Output Compare Register

The 16-bit output compare register is made up of two 8-bit registers at locations $16 (MSB) and $17 (LSB). The output compare register is used for several purposes, such as indicating when a period of time has elapsed. All bits are readable and writable and are not altered by the timer hardware or reset. If the compare function is not needed, the 2 bytes of the output compare register can be used as storage locations.

The output compare register contents are compared with the contents of the free-running counter continually, and if a match is found, the corresponding output compare flag (OCF) bit is set. The output compare register values should be changed after each successful comparison to establish a new elapsed timeout. An interrupt can also accompany a successful output compare provided the corresponding interrupt enable bit (OCIE) is set.

After a processor write cycle to the output compare register containing the MSB ($16), the output compare function is inhibited until the LSB ($17) is also written. The user must write both bytes (locations) if the MSB is written first. A write made only to the LSB ($17) will not inhibit the compare function. The free-running counter is updated every four internal bus clock cycles. The minimum time required to update the output compare register is a function of the program rather than the internal hardware.

The processor can write to either byte of the output compare register without affecting the other byte.

9.1.3 Input Capture Register

Two 8-bit registers, which make up the 16-bit input capture register, are read-only and are used to latch the value of the free-running counter after the corresponding input capture edge detector senses a defined transition. The level transition which triggers the counter transfer is defined by the corresponding input edge bit (IEDG). Reset does not affect the contents of the input capture register.

The result obtained by an input capture will be one more than the value of the free-running counter on the rising edge of the internal bus clock preceding the external transition. This delay is required for internal synchronization. Resolution is one count of the free-running counter, which is four internal bus clock cycles.

The free-running counter contents are transferred to the input capture register on each proper signal transition regardless of whether the input capture flag (ICF) is set or clear. The input capture register always contains the free-running counter value that corresponds to the most recent input capture.

After a read of the input capture register ($14) MSB, the counter transfer is inhibited until the LSB ($15) is also read. This characteristic causes the timer used in the input capture software routine and its interaction with the main program to determine the minimum pulse period.

A read of the input capture register LSB ($15) does not inhibit the free-running counter transfer since they occur on opposite edges of the internal bus clock.

NOTE

Since the TCAP pin is shared with the PC3 I/O pin, changing the state of the PC3 DDR or data register can cause an unwanted TCAP interrupt. This can be handled by clearing the ICIE bit before changing the configuration of PC3 and clearing any pending interrupts before enabling ICIE.

9.1.4 Timer Control Register (TCR) $12

The TCR is a read/write register containing five control bits. Three bits enable interrupts associated with the timer status register flags ICF, OCF, and TOF.

	B7	B6	B5	B4	B3	B2	B1	B0	
$0012	ICIE	OC1IE	TOIE	0	0	0	IEDG	OLVL	TCR
RESET:	0	0	0	0	0	0	U	0	

READ: anytime

WRITE: anytime

ICIE — Input Capture Interrupt Enable
1 = interrupt enabled
0 = interrupt disabled

OC1IE — Output Compare 1 Interrupt Enable
1 = interrupt enabled
0 = interrupt disabled

TOIE — Timer Overflow Interrupt Enable
1 = interrupt enabled
0 = interrupt disabled

IEDG — Input Edge
Value of input edge determines which level transition on TCAP pin will trigger free-running counter transfer to the input capture register

Reset does not affect the IEDG bit (U = unaffected).
1 = positive edge
0 = megative edge

Bits 2 through 4 — Not used
Always read logic zero

OLVL — Not used
Always read logic zero

9.1.5 Timer Status Register (TSR) $13

The TSR is a read-only register containing three status flag bits.

	B7	B6	B5	B4	B3	B2	B1	B0	
$0013	ICF	OC1F	TOF	0	0	0	0	0	TSR
RESET:	U	U	U	0	0	0	0	0	

READ: anytime

WRITE: no effect

ICF — Input Capture Flag

1 = Flag set when selected polarity edge is sensed by input capture edge detector

0 = Flag cleared when TSR and input capture low register ($15) are accessed

OC1F — Output Compare 1 Flag

1 = Flag set when output compare register contents match the free-running counter contents

0 = Flag cleared when TSR and output compare low register ($17) are accessed

TOF — Timer Overflow Flag

1 = Flag set when free-running counter transition from $FFFF to $0000 occurs

0 = Flag cleared when TSR and counter low register ($19) are accessed

Bits 0 through 4 — Not used

Always read logic zero

Accessing the timer status register satisfies the first condition required to clear status bits. The remaining step is to access the register corresponding to the status bit.

A problem can occur when using the timer overflow function and reading the free-running counter at random times to measure an elapsed time. Without incorporating the proper precautions into software, the timer overflow flag could unintentionally be cleared if:

1. The timer status register is read or written when TOF is set.

2. The LSB of the free-running counter is read but not for the purpose of servicing the flag.

The counter alternate register at address $1A and $1B contains the same value as the free-running counter (at address $18 and $19); therefore, this alternate register can be read at anytime without affecting the timer overflow flag in the timer status register.

9.1.6 Timer During WAIT Mode

The CPU clock halts during the WAIT mode, but timer 1 remains active. If interrupts are enabled, a timer interrupt will cause the processor to exit the WAIT mode.

9.1.7 Timer During STOP Mode

In the STOP mode, timer 1 stops counting and holds the last count value if STOP is exited by an interrupt. If RESET is used, the counter is forced to $FFFC. During STOP, if at least one valid input capture edge occurs at the TCAP pin, the input capture detect circuit is armed. This does not set any timer flags or wake up the MCU. When the MCU does wake up, there is an active input capture flag and data from the first valid edge that occurred during the STOP Mode. If RESET is used to exit STOP mode, then no input capture flag or data remains, even if a valid input capture edge occurred.

9.2 Timer 2

Timer 2 is an 8-bit event counter which has one compare register, one event input pin (EVI), and one event output pin (EVO). The event counter is clocked by the external clock (EXCLK) or prescaled system clock (CLK2), selected by the T2CLK bit in the TCR2 register. The EXCLK may be EVI direct or EVI gated by CLK2, which is selected by the IM2 bit at the EVI block (refer to the EVI description).

Timer 2 may be used as a modulus clock divider with EVO pin, free-running counter (when compare register is $00), or periodic interrupt timer.

The timer counter 2 (TCNT2) is an 8-bit up counter with preset input. The counter is preset to $01 by a CMP2 signal from the comparator or by a CPU write to it that is done while the system clock (PH2) is low.

The CLK2 from the prescaler or the EXTCLK from the EVI block is selected as timer clock by the T2CLK bit in the TCR2 register. The CLK2 and the EXCLK are synchronized to the falling edge of system clock in the prescaler and the EVI blocks. The minimum pulse width of CLK2 is the same as the system clock, and the minimum pulse width of EXCLK (event mode) is one PH2 cycle. When the EXCLK (event mode) is selected, 50% duty is not guaranteed.

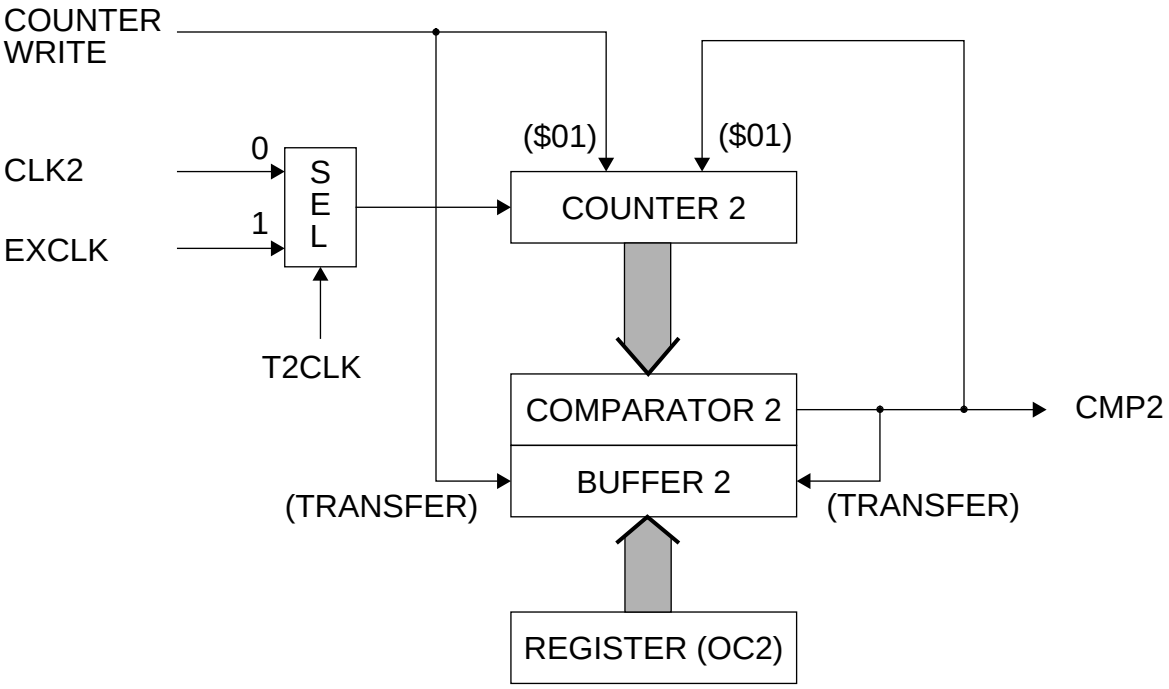


Figure 9-3. Timer 2 Block Diagram

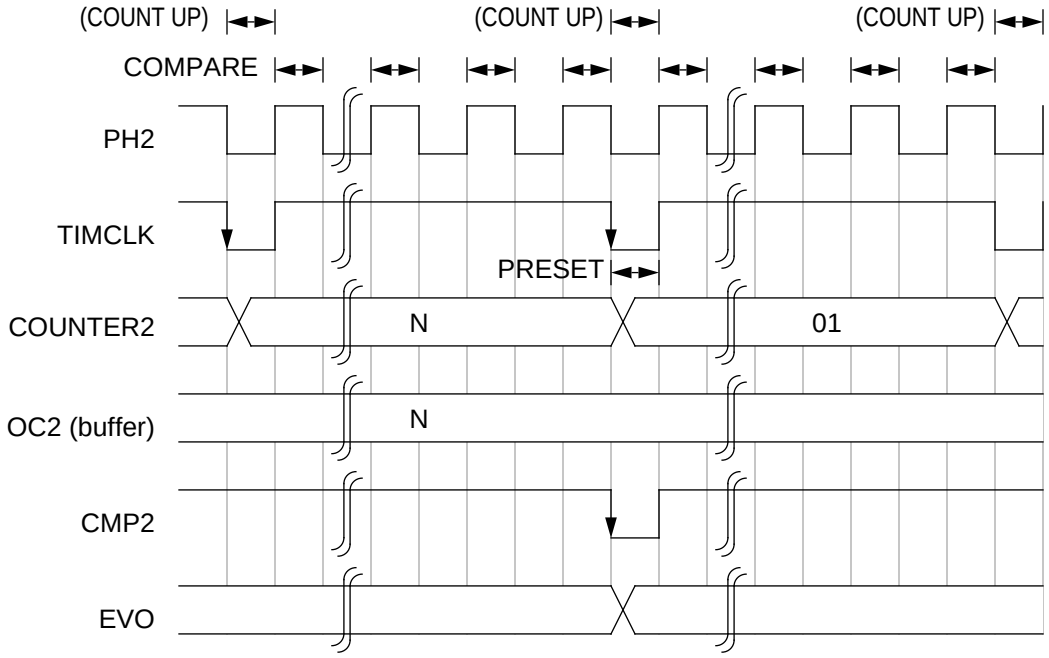
The counter is incremented by the falling edge of the timer clock and the period between two falling edges is defined as one timer cycle in the following description.

The compare register (OC2) is provided for comparison with the timer counter 2 (TCNT2). The OC2 data is transferred to the buffer register when the counter is preset by a CPU write or by a compare output (CMP2). This buffer register is compared with the timer counter 2 (TCNT2).

The comparison between the counter and the OC2 buffer register is done when the system clock is high in each bus cycle. If counter matches with the OC2 buffer register, the comparator latches this result during the current timer cycle. When the next timer cycle begins, the comparator outputs CMP2 signal (if the compare match is detected during previous timer cycle). This CMP2 is used in the counter preset, data transfer to the buffer register, setting OC2F in the TSR2, and the EVO block. The counter preset overrides the counter increment.

The OC2F bit may generate interrupt requests if the OC2IE bit in the TCR2 is set.

OC2=2,3,4,...,FF,0



OC2=1

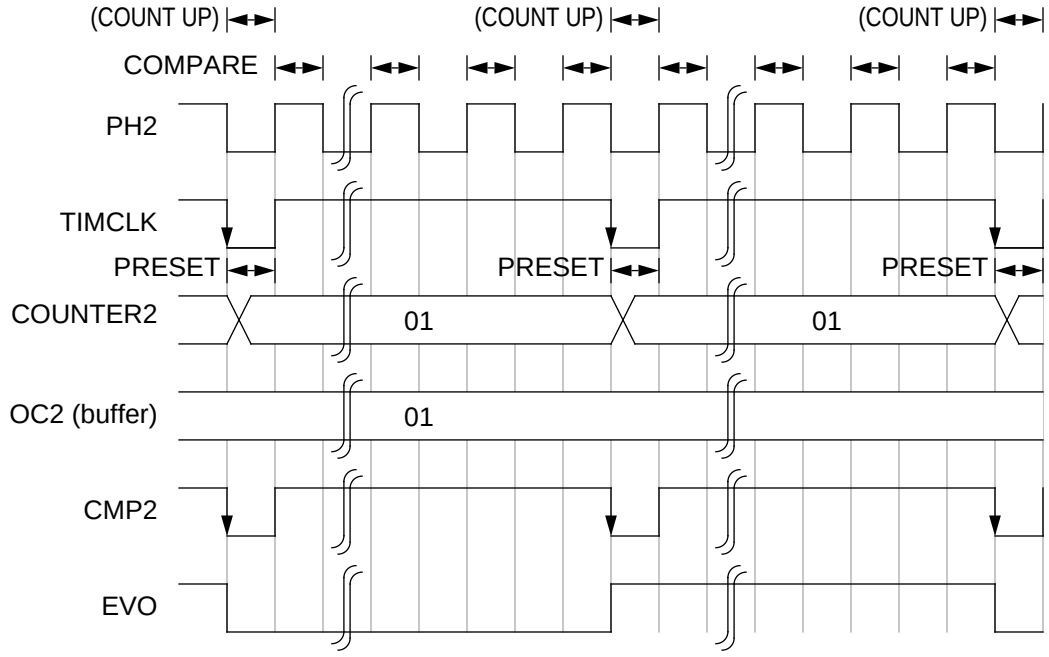
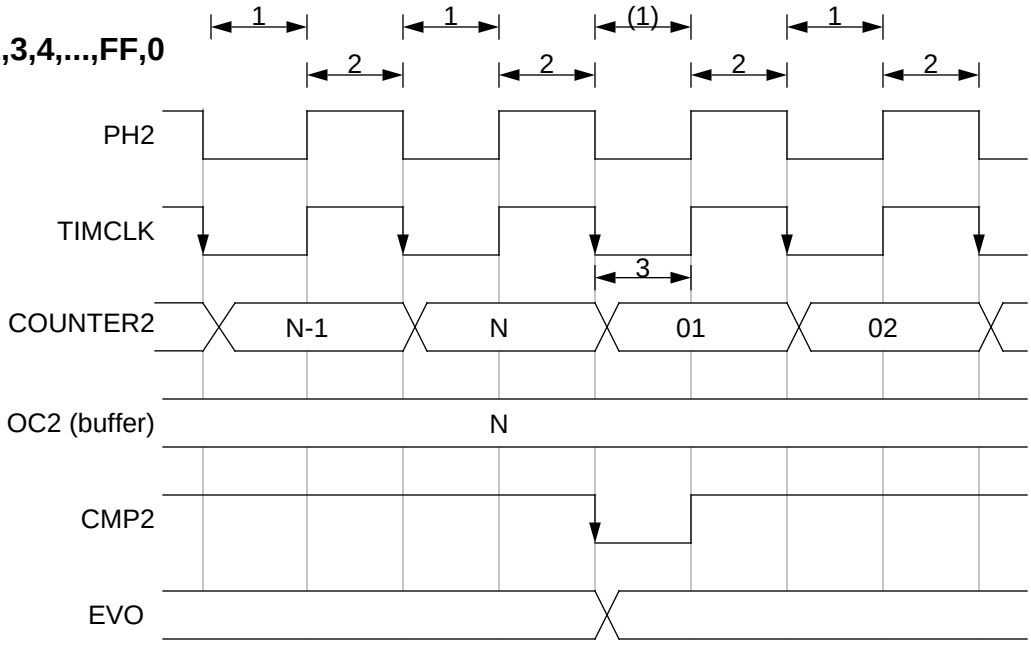


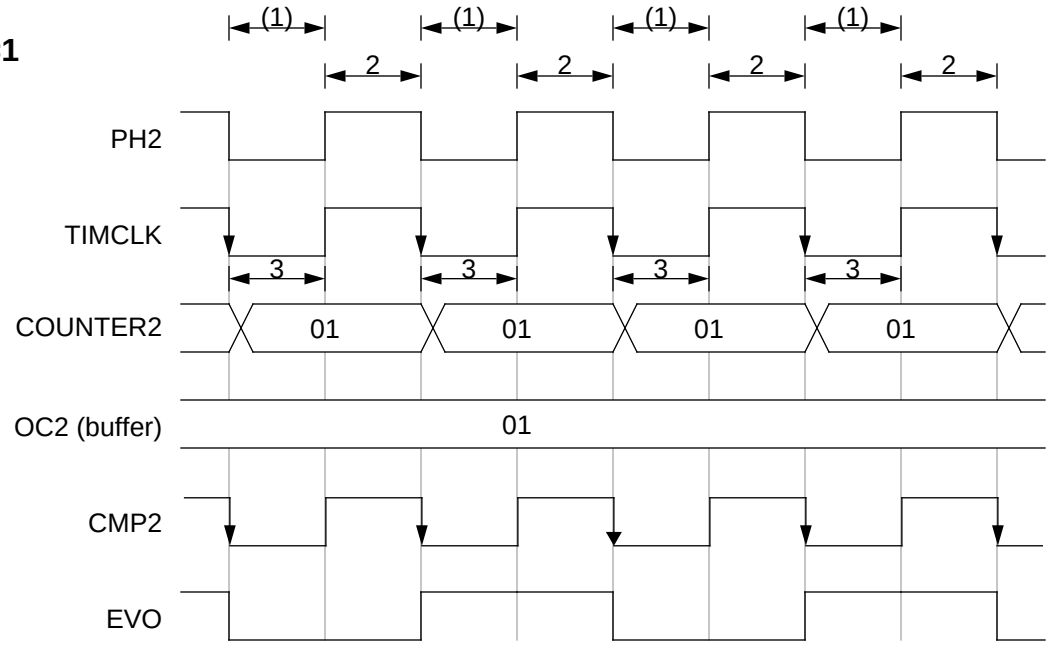
Figure 9-4. Timer 2 Timing Diagram for f(PH2) > f(TIMCLK)

OC2=2,3,4,...,FF,0



1. COUNT UP
2. COMPARE
3. PRESET (that overrides COUNT UP)

OC2=1



1. COUNT UP
2. COMPARE
3. PRESET (that overrides COUNT UP)

Figure 9-5. Timer 2 Timing Diagram for f(PH2) = f(TIMCLK)

9.2.1 Timer Control Register 2 (TCR2)

	B7	B6	B5	B4	B3	B2	B1	B0	
$001C	TI2IE	OC2IE	0	T2CLK	IM2	IL2	OE2	OL2	TCR2
RESET:	0	0	0	0	0	0	0	0	

READ: anytime

WRITE: anytime

TI2IE — Timer Input 2 Interrupt Enable

TI2IE bit enables timer input 2 (EVI) interrupt when TI2F is set. This bit is cleared on reset.t

0 = timer input 2 interrupt is disabled
1 = timer input 2 interrupt is enabled

OC2IE — Compare 2 Interrupt Enable

OC2IE bit enables compare 2 (CMP2) interrupt when compare match is detected (OC2F is set). This bit is cleared on reset.

0 = compare 2 interrupt is disabled
1 = compare 2 interrupt is enabled

Bit 5 — reserved

This bit is not used and is always read as logic zero.

T2CLK — Timer 2 Clock Select

The T2CLK bit selects clock source for the timer counter 2. This bit is cleared on reset.

0 = CLK2 from prescaler is selected
1 = EXCLK from EVI input block is selected

IM2 — Timer Input 2 Mode Select

The IM2 bit selects whether EVI input is gated by CLK2 or not gated by CLK2. This bit is cleared on reset.

0 = EVI is not gated by CLK2 (event mode)
1 = EVI is gated by CLK2 (gate mode)

IL2 — Timer Input 2 Active Edge (Level) select

The IL2 bit selects the active edge of EVI to increment counter for the event mode (IM2 = 0) or gate enable level of EVI for the gate mode (IM2 = 1). This bit is cleared on reset.

0 = falling edge is selected (event mode)
low level enables counting (gate mode)

1 = rising edge is selected (event mode)
high level enables counting (gate mode)

Table 9-1. EVI Modes Selection

IM2	IL2	ACTION ON CLOCK
0	0	FALLING EDGE OF EVI INCREMENTS COUNTER
0	1	RISING EDGE OF EVI INCREMENTS COUNTER
1	0	LOW LEVEL ON EVI ENABLES COUNTING
1	1	HIGH LEVEL ON EVI ENABLES COUNTING

OE2 — Timer Output 2 (EVO) Output Enable

The OE2 bit enables EVO output on PC5 pin. When this bit is changed, control of the pin is delayed (synchronized) until the next active edge of EVO is selected by OL2 bit. This bit is cleared on reset.

0 = EVO output is disabled

1 = EVO output is enabled

OL2 — Timer Output 2 Edge Select for Synchronization

The OL2 bit selects which edge of EVO clock should be synchronized by the OE2 bit control. The OL2 bit also decides the initial value of the CMP2 divider, when counter 2 is written to by the CPU. This bit is cleared on reset.

0 = The falling edge of EVO switches EVO output and PC5 if OE2 bit has bee changed.

1 = The rising edge of EVO switches EVO output and PC5 if OE2 bit has been changed.

9.2.2 Timer Status Register 2 (TSR2)

	B7	B6	B5	B4	B3	B2	B1	B0	
$001D	TI2F	OC2F	0	0	RTI2F	ROC2F	0	0	TSR2
RESET:	0	0	0	0	0	0	0	0	

READ: anytime
(bits 3 and 2 are write-only bits and always read as logic zero)

WRITE: anytime
(bits 7 and 6 are read-only bits and write has no effect)

TI2F — Timer Input 2 (EVI) Interrupt Flag

In event mode the event edge sets TI2F. In gated time accumulation mode the trailing edge of the gate signal at the EVI input pin sets TI2F. When TI2IE bit and this bit are set, an interrupt is generated. This bit is a read-only bit and writes have no effect. The TI2F is cleared by writing a logic one to the RTI2F bit and on reset.

OC2F — Compare 2 Interrupt Flag

The OC2F bit is set when compare match is detected between counter 2 and OC2 register. When OC2IE bit and this bit are set, an interrupt is generated. This bit is a read-only bit and writes have no effect. The OC2F is cleared by writing a logic one to ROC2F bit and on reset.

Bits 5 and 4 — reserved

These bits are not used and always read as logic zero.

RTI2F — Reset Timer Input 2 Flag

The RTI2F bit is a write-only bit and always reads as logic zero. Writing logic one to this bit clears TI2F bit and writing a logic zero to this bit has no effect.

ROC2F — Reset Output Compare 2 Flag

The ROC2F bit is a write-only bit and always reads as logic zero. Writing logic one to this bit clears OC2F bit and writing a logic zero to this bit has no effect.

Bits 1 and 0 — reserved

These bits are not used and always read as logic zero.

9.2.3 Output Compare Register 2 (OC2)

	B7	B6	B5	B4	B3	B2	B1	B0	
$001E	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0	OC2
RESET:	1	1	1	1	1	1	1	1	

READ: anytime

WRITE: anytime

The OC2 register data is transferred to the buffer register when the CPU writes to TCNT2, when the CMP2 presets the TCNT2, or when system resets.

When the OC2 buffer register matches the TCNT2 register, the OC2F bit in the TSR2 register is set and TCNT2 is preset to $01.

9.2.4 Timer Counter 2 (TCNT2)

	B7	B6	B5	B4	B3	B2	B1	B0	
$001F	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0	TCNT2
RESET:	0	0	0	0	0	0	0	1	

READ: anytime

WRITE: anytime
(TCNT2 becomes $01 by any write data)

TCNT2 is incremented by the falling edge of the timer clock (which is synchronized and has the same timing as falling edge of PH2).

The TCNT2 register is compared with OC2 buffer register and initialized to $01 if it matches. It is also initialized to $01 on reset and any CPU write to this register.

The CPU read of this counter should be done while PH2 is high. Data may be latched by the local or main data bus while PH2 is low.

9.2.5 Time Base Control Register 1 (TBCR1)

	B7	B6	B5	B4	B3	B2	B1	B0	
$0010	TBCLK	0	LCLK	0	0	0	T2R1	T2R0	TBCR1
RESET:	0	0	0	0	0	0	0	0	

READ: anytime

WRITE: anytime

T2R1/0 — Prescale Rate Select Bits for Timer 2

The T2R1 and T2R0 bits select prescale rate of CLK2 for timer 2 and timer input 2. These bits are cleared on reset.

Table 9-2. Time Base Prescale Rate Selection

T2R1	T2R0	SYSTEM CLOCK DIVIDED BY
0	0	1
0	1	4
1	0	32
1	1	256

9.2.6 Timer Input 2 (EVI)

The Event Input (EVI) is used as an external clock input for timer 2.

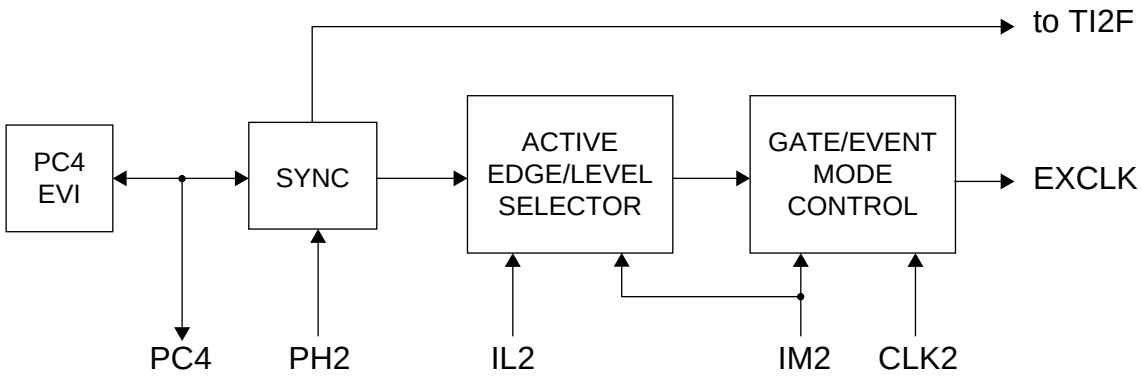


Figure 9-6. EVI Block Diagram

Since the external clock may be asynchronous to the internal clock, this input has a synchronizer which samples external clock by the internal system clock. (The input transition synchronizes to the falling edge of PH2. Therefore, to be measured, the minimum pulse width for EVI must be larger than one system clock.)

The IM2 and IL2 bits in the TCR2 determine how this synchronized external clock is used. IM2 bit decides between event mode and gate mode, and IL2 bit decides which level or edge is activated.

In the event mode (IM2 = 0), the external clock drives the timer 2 counter directly and the active edge at the EVI pin is selected by the IL2 bit. When active edge is detected, the TI2F bit in the TCR2 is set.

Table 9-3. EVI Modes Selection

IM2	IL2	ACTION ON CLOCK
0	0	FALLING EDGE OF EVI INCREMENTS COUNTER
0	1	RISING EDGE OF EVI INCREMENTS COUNTER
1	0	LOW LEVEL ON EVI ENABLES COUNTING
1	1	HIGH LEVEL ON EVI ENABLES COUNTING

NOTE

Since the EVI pin is shared with the PC4 I/O pin, DDRC4 should always be cleared whenever EVI is used. EVI should not be used when DDRC4 is high.

In the gate mode (IM2 = 1), the EVI input is gated by CLK2 from the prescaler and gate output drives the timer 2 counter. IL2 bit decides active level of the external input. When the transition from active level to inactive level is detected, the TI2F bit is set.

Changing the IM2 bit may cause an illegal count up of TCNT2, thus presetting TCNT2 after initializing IM2 is required.

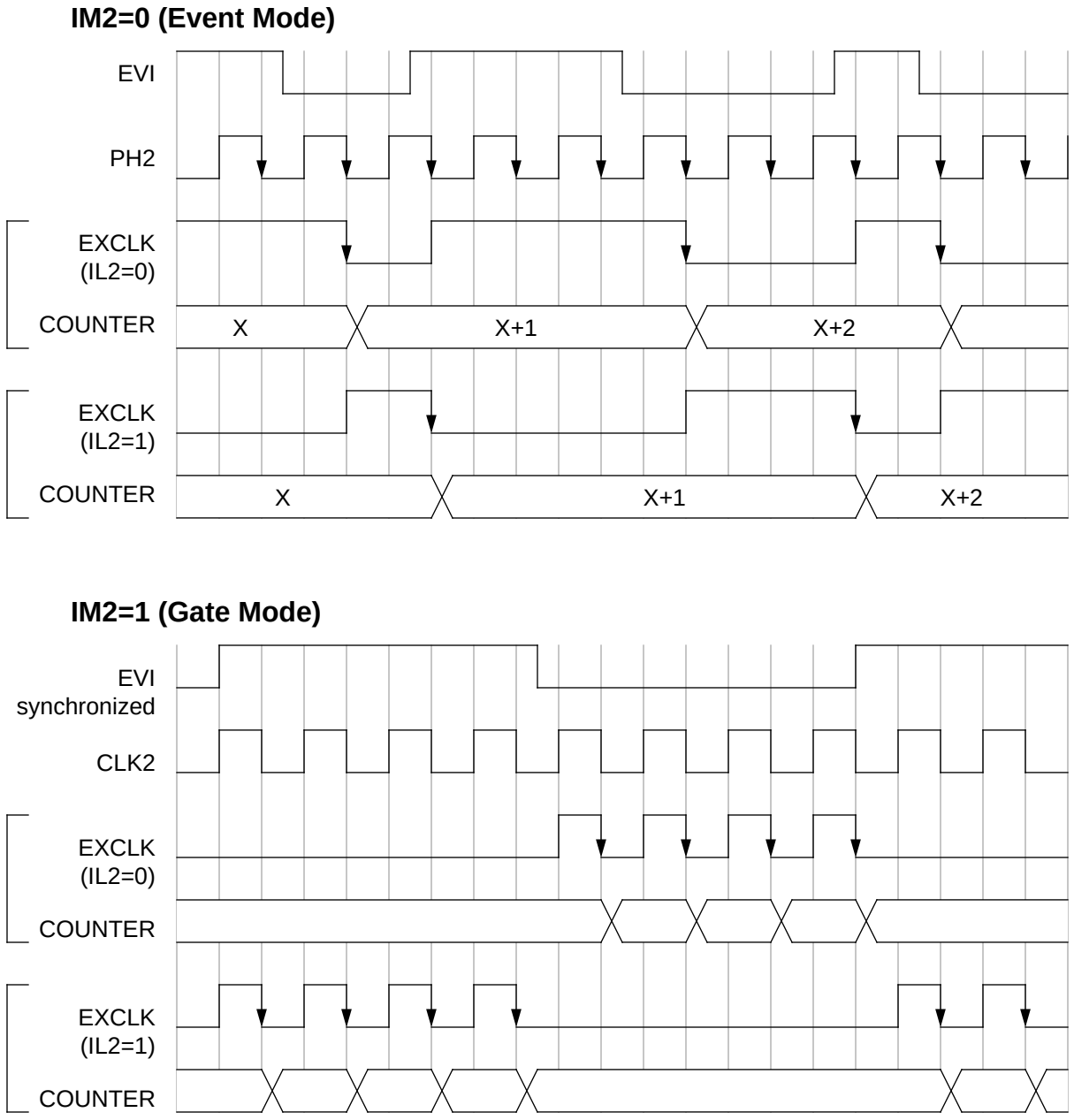


Figure 9-7. EVI Timing Diagram

9.2.7 Event Output (EVO)

The EVO pin is the clock output pin of timer 2. The compare output from the timer 2 (CMP2) is divided in this block for 50% duty output signal. This 1/2 divider is initialized to the level of the OL2 bit when the timer counter 2 is written to by the CPU (initialized).

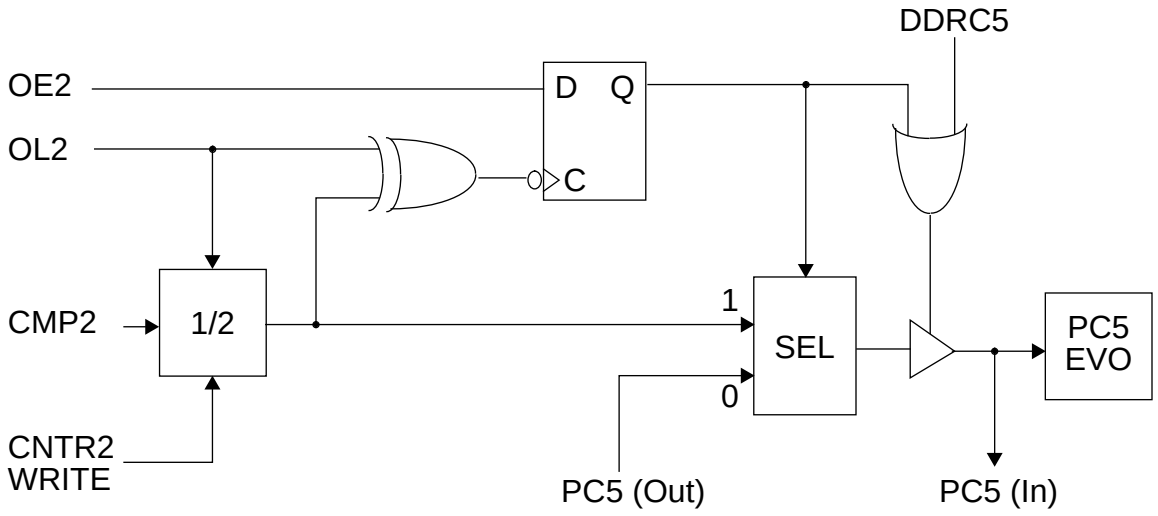


Figure 9-8. EVO Block Diagram

When the OE2 bit in the timer control register 2 (TCR2) is set, the EVO output is activated and when OE2 is cleared EVO is deactivated. These controls must be done synchronously to the EVO output signal to avoid an incomplete pulse on the pin. The OL2 bit in the TCR2 decides which edge of EVO should be synchronized.

When DDRC5 bit is set or the synchronized output enable is high (clock on), the output buffer at the EVO/PC5 pin is enabled. If DDRC5 bit is set to one, the pin state during the idling condition (clock off) depends on the PC5 output data latch. If DDRC5 is cleared, the pin becomes high impedance during clock off.

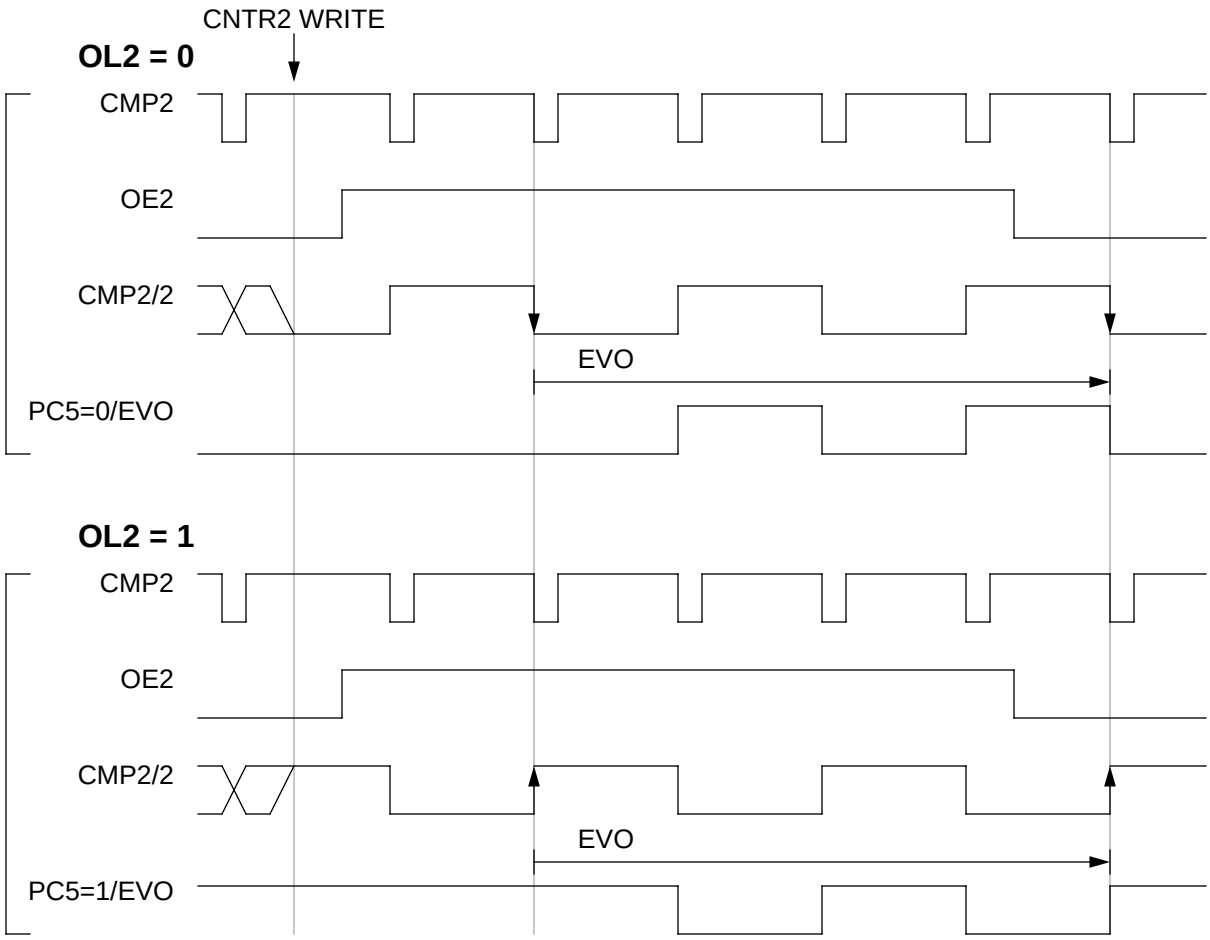


Figure 9-9. EVO Timing Diagram

9.3 Prescaler

The 8-bit prescaler in the timer system divides system clock (PH2) and provides divided clock to each timer and event input.

CLK1 for the timer 1 is a fixed frequency clock (PH2/PH4).

CLK2 for the timer 2 is selected by T2R1 and T2R0 bits in the TBCR1, and this clock is also used as the event input for the gate mode. The CLK2 transitions must be synchronous to the falling edge of PH2.

Table 9-4. Time Base Prescale Rate Selection

T2R1	T2R0	SYSTEM CLOCK DIVIDED BY
0	0	1
0	1	4
1	0	32
1	1	256

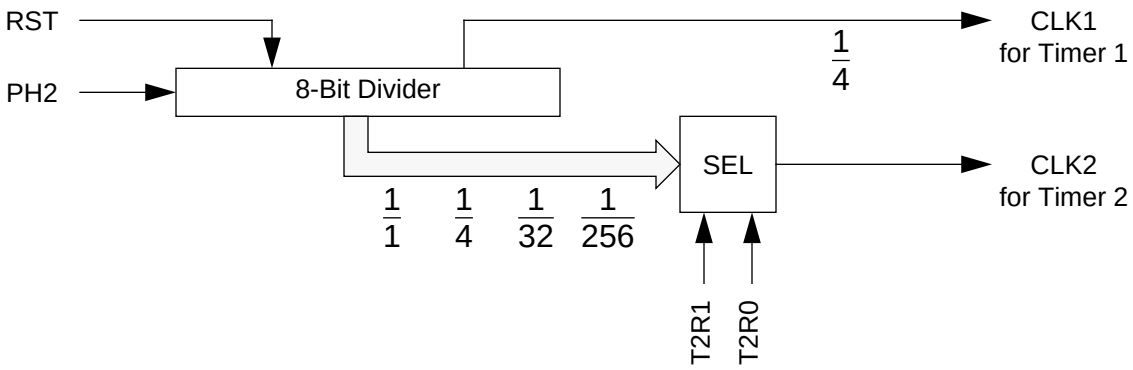


Figure 9-10. Prescaler Block Diagram

SECTION 10
LCD DRIVER

The LCD driver may be configured with four backplanes (BP) and 39 frontplanes (FP) maximum. The V_{DD} voltage is the highest level of the output waveform and the lower three levels are applied from VLCD1, VLCD2, and VLCD3 inputs.

On reset, LCD enable bit (LCDE) in the LCD control register (LCDCR) is cleared (LCD drivers at a disabled state) and all BP pins and FP pins output V_{DD} levels.

The LCD clock is generated by the time base module and LCLK bit in the TBCR1 selects clock frequency.

10.1 LCD Waveform Examples

The following figures illustrate the LCD timing examples.

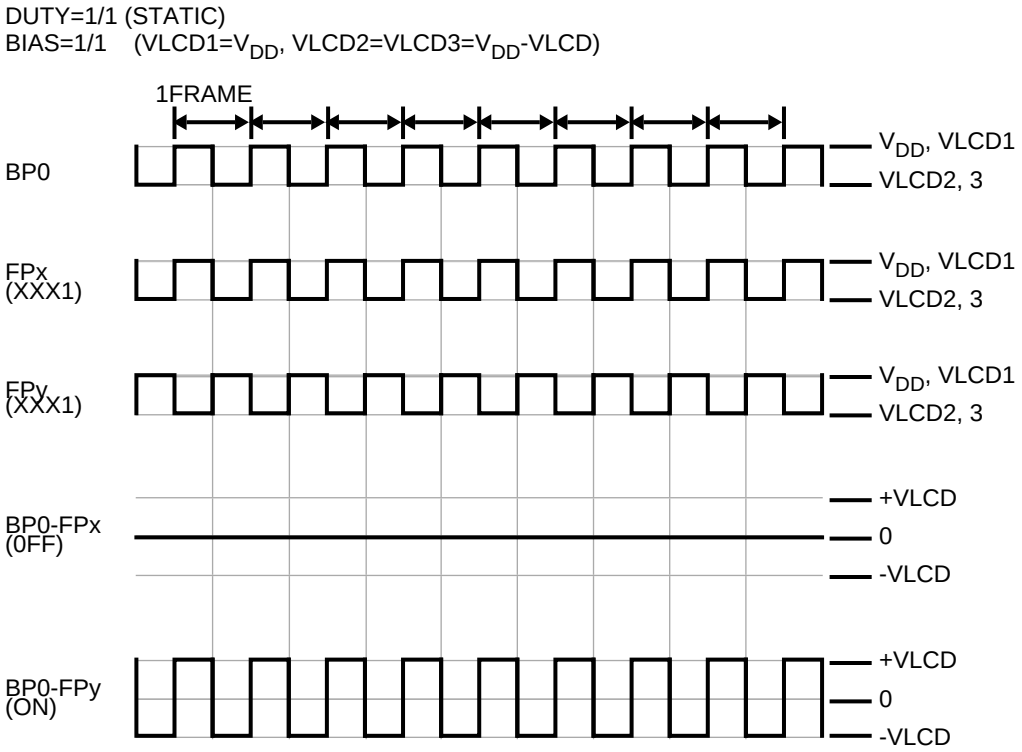


Figure 10-1. LCD 1/1 Duty and 1/1 Bias Timing Diagram

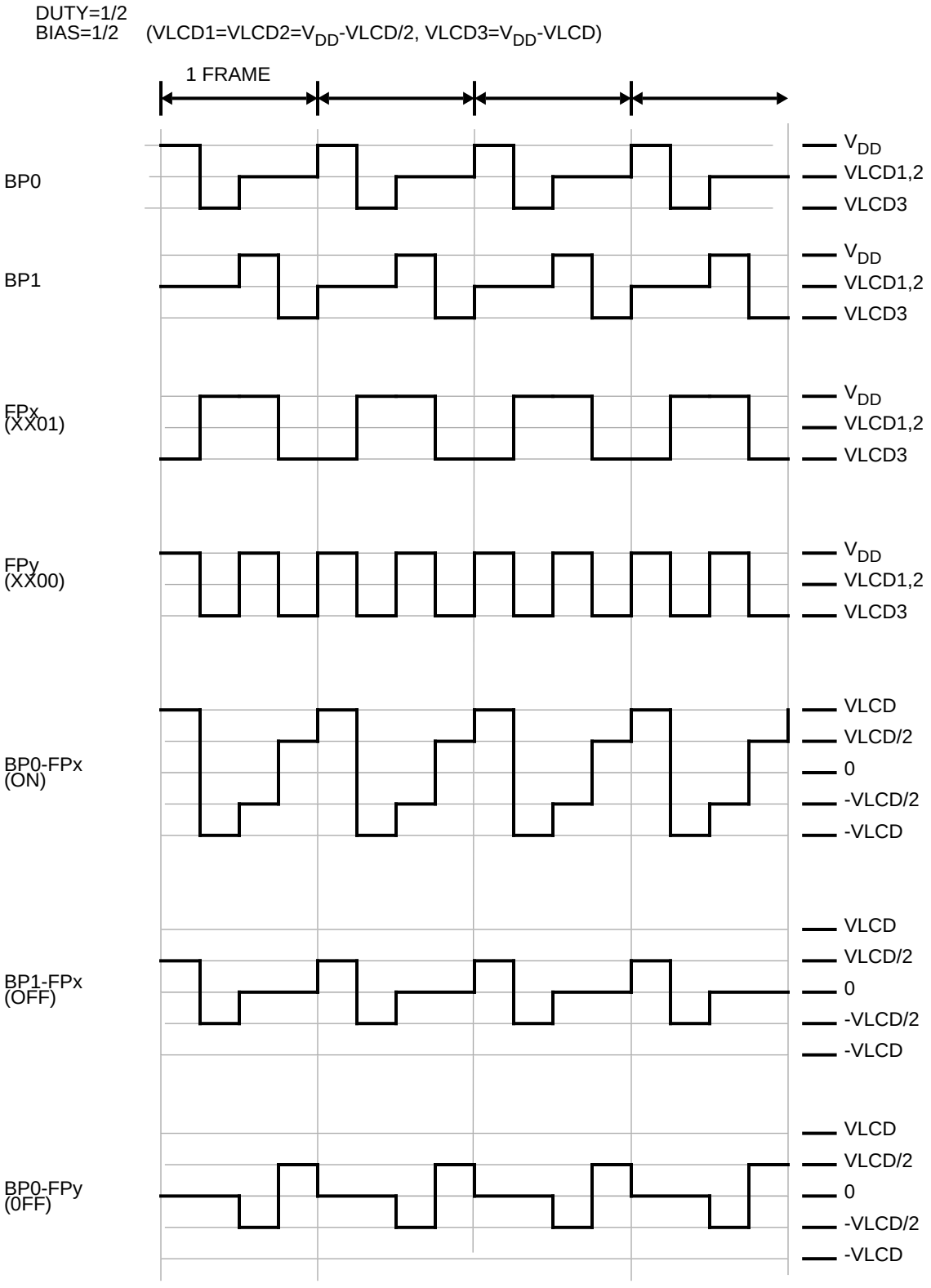


Figure 10-2. LCD 1/2 Duty and 1/2 Bias Timing Diagram

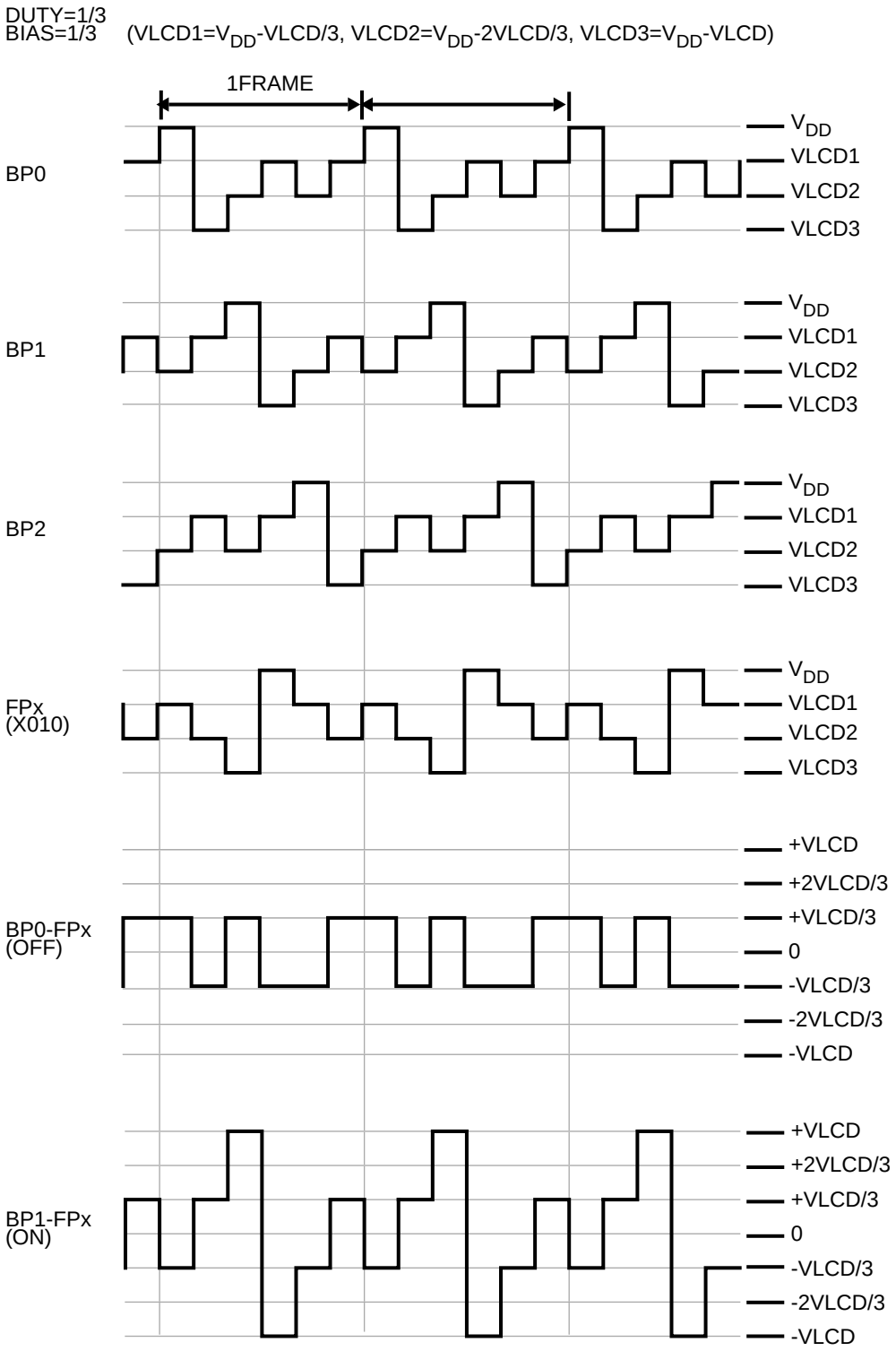


Figure 10-3. LCD 1/3 Duty and 1/3 Bias Timing Diagram

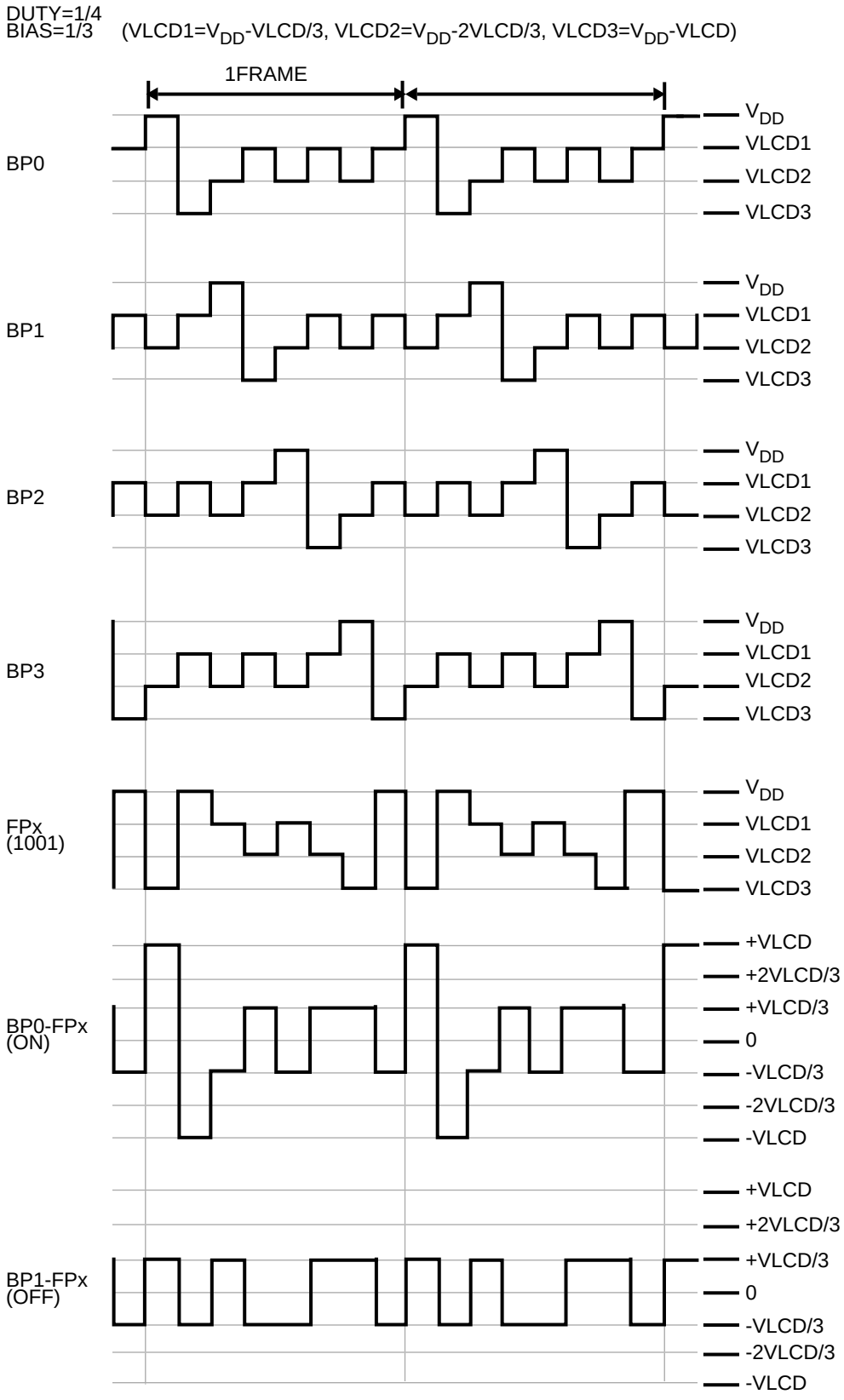


Figure 10-4. LCD 1/4 Duty and 1/3 Bias Timing Diagram

10.2 Backplane Driver and Port Selection

The number of backplane (port D) pins depends on the LCD duty. It is automatically selected by DUTY1 and DUTY0 bits in the LCD control register (LCDCR). On reset, these bits are cleared and 1/4 duty is selected.

Table 10-1. Backplane and Port Selection

DUTY	LCD Control		Pin Selection			
	DUTY1	DUTY0	BP3/PD3	BP2/PD2	BP1/PD1	BP0
1 / 1	0	1	PD3	PD2	PD1	BP0
1 / 2	1	0	PD3	PD2	BP1	BP0
1 / 3	1	1	PD3	BP2	BP1	BP0
1 / 4	0	0	BP3	BP2	BP1	BP0

10.3 Frontplane Driver and Port Selection

The number of frontplane (FP) pins depends on the number of port D and E bits. If port bits are selected as a parallel output port, the number of the FP pins is decreased to 27 at minimum case. The selections between frontplane and port (nibble wide) are done by the PEH, PEL, and PDH bits in the LCDCR. These bits can also be controlled on a bit-wide basis by using the PDMUX and PEMUX registers. PDH, PEH, and PEL have priority over the mux registers. On reset, port D and port E bits are disconnected and FP27 through FP38 are connected to the pin.

Table 10-2. Frontplane and Port Selection

FP / Port Control					Pin Selection		
PEH	PEL	PDH	PDMx	PEMx	FP27:30/ PE7:4	FP31:34/ PE3:0	FP35:38/ PD7:4
		0	0				FP35:38
		0	1				varied
		1	X				PD7:4
	0			0		FP31:34	
	0			1		varied	
	1			X		PE3:0	
0				0	FP27:30		
0				1	varied		
1				X	PE7:4		

10.4 LCD Control Register (LCDCR)

	B7	B6	B5	B4	B3	B2	B1	B0	
$0020	LCDE	DUTY1	DUTY0	0	PEH	PEL	PDH	0	LCDCR
RESET:	0	0	0	0	0	0	0	0	

READ: anytime

WRITE: anytime

LCDE — LCD Output Enable

The LCDE bit enables all BP and FP outputs. (This bit does not affect PEH, PEL, or PDH bits.) This bit is cleared on reset.

0 = all dedicated FP pins output highest (V_{DD}) level; BP and FP pins shared with an output port data

1 = all BP and FP pins output LCD waveforms

DUTY1 and DUTY0 — LCD Duty Select

The DUTY1 and DUTY0 bits select the duty of LCD driver. The number of BP pins is related to this duty selection. The unused BP pin is used as port D pin. Default duty is 1/4 duty. These bits are cleared on reset. See Table 10-1. Backplane and Port Selection.

Bit 4 — reserved

This bit is not used and always reads as logic zero.

PEH — Select Port E (H)

The PEH bit enables the upper four bits of port E instead of LCD drivers. This bit is cleared on reset. See **10.3 Frontplane Driver and Port Selection**.

0 = FP27 through FP30 are selected

1 = PE7 through PE4 are selected

PEL — Select Port E (L)

The PEL bit enables lower four bits of port E instead of LCD drivers. This bit is cleared on reset. See **10.3 Frontplane Driver and Port Selection**.

0 = FP31 through FP34 are selected

1 = PE3 through PE0 are selected

PDH — Select Port D (H)

The PDH bit enables upper four bits of port D instead of LCD drivers. This bit is cleared on reset. See **10.3 Frontplane Driver and Port Selection**.

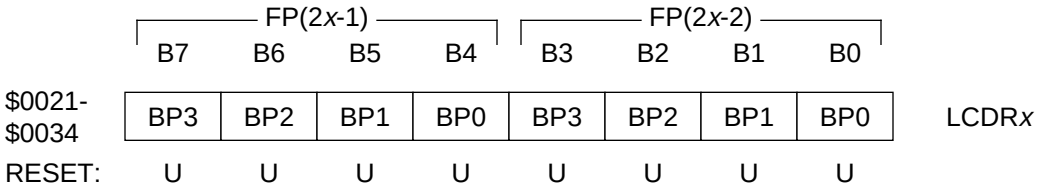
0 = FP35 through FP38 are selected

1 = PD7 through PD4 are selected

Bit 0 — reserved

This bit is not used and is always read as logic zero.

10.5 LCD Data Register (LCDR*x*)



READ: anytime

WRITE: anytime

Data in the LCDR*x* (LCDR1 through LCDR20) controls the waveform of the two frontplane drivers. Bits 0 through 3 and bits 4 through 7 of this register decide the waveforms at the BP0 through BP3 timings. If LCD duty is not 1/4, the register bit for the unused backplane has no meaning. The upper four bits of LCDR20 are not implemented and unknown data may be read.

0 = output deselect waveform at the corresponding backplane timing
1 = output select waveform at the corresponding backplane timing

Table 10-3. Frontplane Data Register Bit Usage

Duty	Frontplane Data Register Bit Usage							
	7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
1 / 1	—	—	—	BP0	—	—	—	BP0
1 / 2	—	—	BP1	BP0	—	—	BP1	BP0
1 / 3	—	BP2	BP1	BP0	—	BP2	BP1	BP0
1 / 4	BP3	BP2	BP1	BP0	BP3	BP2	BP1	BP0

SECTION 11
INSTRUCTION SET

This section describes the addressing modes and the types of instructions.

11.1 Addressing Modes

The MCU uses 10 different addressing modes to provide the programmer with an opportunity to optimize the code for all situations. The various indexed addressing modes make it possible to locate data tables, code conversion tables, and scaling tables anywhere in the memory space. Short indexed accesses are single-byte instructions; the longest instructions (three bytes) permit accessing tables throughout memory. Short and long absolute addressing is also included. One- or two-byte direct addressing instructions access all data bytes in most applications. Extended addressing permits jump instructions to reach all memory.

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

11.1.1 Immediate

In the immediate addressing mode, the operand is contained in the byte immediately following the opcode. The immediate addressing mode is used to access constants that do not change during program execution (for example, a constant used to initialize a loop counter).

11.1.2 Direct

In the direct addressing mode, the effective address of the argument is contained in a single byte following the opcode byte. Direct addressing allows the user to directly address the lowest 256 bytes in memory ($0000-$00FF) with a single two-byte instruction.

11.1.3 Extended

In the extended addressing mode, the effective address of the argument is contained in the two bytes following the opcode byte. Instructions with extended addressing mode are capable of referencing arguments anywhere in memory with

a single three-byte instruction. When using the Motorola assembler, the user need not specify whether an instruction uses direct or extended addressing. The assembler automatically selects the shortest form of the instruction.

11.1.4 Relative

The relative addressing mode is only used in branch instructions. In relative addressing, the contents of the 8-bit signed offset byte (which is the last byte of the instruction) is added to the PC if, and only if, the branch conditions are true. Otherwise, control proceeds to the next instruction. The span of relative addressing is from –128 to +127 from the address of the next opcode. The programmer need not calculate the offset when using the Motorola assembler, since it calculates the proper offset and checks to see that it is within the span of the branch.

11.1.5 Indexed, No Offset

In the indexed, no offset addressing mode, the effective address of the argument is contained in the 8-bit index register. This addressing mode can access the first 256 memory locations ($0000-$00FF). These instructions are only one byte long. This mode is often used to move a pointer through a table or to hold the address of a frequently referenced RAM or I/O location.

11.1.6 ndexed, 8-Bit Offset

In the indexed, 8-bit offset addressing mode, the effective address is the sum of the contents of the unsigned 8-bit index register and the unsigned byte following the opcode. The addressing mode is useful for selecting the K^{th} element in an n element table. With this two-byte instruction, K would typically be in X with the address of the beginning of the table in the instruction. As such, tables may begin anywhere within the first 256 addressable locations and could extend as far as location 510 ($01FE). This is the last location which can be accessed in this way.

11.1.7 Indexed, 16-Bit Offset

In the indexed, 16-bit offset addressing mode, the effective address is the sum of the contents of the unsigned 8-bit index register and the two unsigned bytes following the opcode. This address mode can be used in a manner similar to indexed, 8-bit offset except that this three-byte instruction allows tables to be anywhere in memory. As with direct and extended addressing, the Motorola assembler determines the shortest form of indexed addressing.

11.1.8 Bit Set/Clear

In the bit set/clear addressing mode, the bit to be set or cleared is part of the opcode, and the byte following the opcode specifies the direct addressing of the byte in which the specified bit is to be set or cleared. Any read/write bit in the first 256 locations of memory, including I/O, can be selectively set or cleared with a single two-byte instruction.

11.1.9 Bit Test and Branch

The bit test and branch addressing mode is a combination of direct addressing and relative addressing. The bit that is to be tested and its condition (set or clear) is included in the opcode. The address of the byte to be tested is in the single byte immediately following the opcode byte. The signed relative 8-bit offset in the third byte is added to the PC if the specified bit is set or cleared in the specified memory location. This single three-byte instruction allows the program to branch based on the condition of any readable bit in the first 256 locations of memory. The span of branching is from –128 to +127 from the address of the next opcode. The state of the tested bit is also transferred to the carry bit of the condition code register.

11.1.10 Inherent

In the inherent addressing mode, all the information necessary to execute the instruction is contained in the opcode. Operations specifying only the index register and/or accumulator, as well as the control instruction with no other arguments, are included in this mode. These instructions are 1 byte long.

11.1.11 Software Interrupt (SWI)

The SWI is an executable instruction and a non-maskable interrupt: It is executed regardless of the state of the I bit in the CCR. If the I bit is zero (interrupts enabled), SWI executes after interrupts which were pending when the SWI was fetched, but before interrupts generated after the SWI was fetched. The interrupt service routine address is specified by the contents of memory locations $FFFC and $FFFD.

11.2 Instruction Set

The MCU has a set of 62 basic instructions. They can be divided into five different types: register/memory, read-modify-write, branch, bit manipulation, and control. The following paragraphs briefly explain each type. For more information on the instruction set, refer to the *M6805 HMOS Family User's Manual* (M6805UM/AD3) or the *MC68HC05C4 Technical Data* (MC68HC05C4/D).

11.2.1 Register/Memory Instructions

Most of these instructions use two operands. One operand is either the accumulator or the index register. The other operand is obtained from memory using one of the addressing modes. The jump unconditional (JMP) and jump to subroutine (JSR) instructions have no register operand. Refer to the following instruction list.

Function	Mnemonic
Load A from Memory	LDA
Load X from Memory	LDX
Store A in Memory	STA
Store X in Memory	STX
Add Memory to A	ADD
Add Memory and Carry to A	ADC
Subtract Memory	SUB
Subtract Memory from A with Borrow	SBC
AND Memory to A	AND
OR Memory with A	ORA
Exclusive OR Memory with A	EOR
Arithmetic Compare A with Memory	CMP
Arithmetic Compare X with Memory	CPX
Bit Test Memory with A (Logical Compare)	BIT
Jump Unconditional	JMP
Jump to Subroutine	JSR
Multiply	MUL

11.2.2 Read-Modify-Write Instructions

These instructions read a memory location or a register, modify or test its contents, and write the modified value back to memory or to the register. The test for negative or zero (TST) instruction is an exception to the read-modify-write sequence since it does not modify the value. Refer to the following list of instructions.

Function	Mnemonic
Increment	INC
Decrement	DEC
Clear	CLR
Complement	COM
Negate (Twos Complement)	NEG
Rotate Left through Carry	ROL
Rotate Right through Carry	ROR
Logical Shift Left	LSL
Logical Shift Right	LSR
Arithmetic Shift Right	ASR
Test for Negative or Zero	TST

11.2.3 Branch Instructions

This set of instructions branches if a particular condition is met; otherwise, no operation is performed. Branch instructions are two-byte instructions. Refer to the following list for branch instructions.

Function	Mnemonic
Branch Always	BRA
Branch Never	BRN
Branch if Higher	BHI
Branch if Lower or Same	BLS
Branch if Carry Clear	BCC
Branch if Higher or Same	BHS
Branch if Carry Set	BCS
Branch if Lower	BLO
Branch if Not Equal	BNE
Branch if Equal	BEQ
Branch if Half Carry Clear	BHCC
Branch if Half Carry Set	BHCS
Branch if Plus	BPL
Branch if Minus	BMI
Branch if Interrupt Mask Bit is Clear	BMC
Branch if Interrupt Mask Bit is Set	BMS
Branch if Interrupt Line is Low	BIL
Branch if Interrupt Line is High	BIH
Branch to Subroutine	BSR

11.2.4 Bit Manipulation Instructions

The MCU is capable of setting or clearing any writable bit which resides in the first 256 bytes of the memory space where all port registers, port DDRs, timer, timer control, and on-chip RAM reside. An additional feature allows the software to test and branch on the state of any bit within these 256 locations. The bit set, bit clear, bit test, and branch functions are all implemented with a single instruction. For test and branch instructions, the value of the bit tested is also placed in the carry bit of the condition code register. These instructions are also read-modify-write instructions. Do not bit manipulate write-only locations. Refer to the following list for bit manipulation instructions.

Function	Mnemonic
Branch if Bit n is Set	BRSET n (n = 0. . .7)
Branch if bit n is Clear	BRCLR n (n = 0. . .7)
Set Bit n	BSET n (n = 0. . .7)
Clear Bit n	BCLR n (n = 0. . .7)

11.2.5 Control Instructions

These instructions are register reference instructions and are used to control processor operation during program execution. Refer to the following list for control instructions.

Function	Mnemonic
Transfer A to X	TAX
Transfer X to A	TXA
Set Carry Bit	SEC
Clear Carry Bit	CLC
Set Interrupt Mask Bit	SEI
Clear Interrupt Mask Bit	CLI
Software Interrupt	SWI
Return from Subroutine	RTS
Return from Interrupt	RTI
Reset Stack Pointer	RSP
No-Operation	NOP
Stop	STOP
Wait	WAIT

SECTION 12
ELECTRICAL SPECIFICATIONS

This section contains parametric and timing information.

12.1 Maximum Ratings

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum-rated voltages to this high-impedance circuit. For proper operation, it is recommended that V_{IN} and V_{OUT} be constrained to the range $V_{SS} \le (V_{IN}$ or $V_{OUT}) \le V_{DD}$. Reliability of operation is enhanced if unused inputs are connected to an appropriate logic voltage level, either V_{SS} or V_{DD}.

Table 12-1. Maximum Ratings

Rating	Symbol	Value	Unit
Supply Voltage	V_{DD} V_{LCD1} V_{LCD2} V_{LCD3}	-0.3 to +7.0 V_{SS} -0.3 to V_{DD} +0.3 V_{SS} -0.3 to V_{DD} +0.3 V_{SS} -0.3 to V_{DD} +0.3	V V V V
Input Voltage	V_{in}	V_{SS} -0.3 to V_{DD} +0.3	V
Bootloader/Self-Check Mode ($\overline{IRQ}$ Pin Only)	V_{in}	V_{SS} -0.3 to 2 X V_{DD} +0.3	V
Output Voltage	V_{out}	V_{SS} -0.3 to V_{DD} +0.3	V
Current Drain Per Pin Excluding V_{DD} and V_{SS}	I	12.5	mA
Operating Temperature Range MC68HC(7)05L16 (Standard)	T_A	T_L to T_H 0 to +70	°C
Storage Temperature Range	T_{stg}	-55 to +150	°C

12.2 Thermal Characteristics

Table 12-2. Thermal Characteristics

Characteristic	Symbol	Value	Unit
Thermal Resistance 80-pin Plastic QFP	θ_{JA}	120	°C/W

12.3 Recommended Operating Conditions

($+2.2 \le V_{DD} \le +5.5$Vdc, V_{SS} = 0Vdc, $T_L \le T_A \le T_H$, unless otherwise noted)

Table 12-3. Recommended Operation Conditions

Rating	Symbol	Value			Unit
		Min	Typ	Max	
(f_{OP} = 2.1 MHz)		4.5	5.0	5.5	V
(f_{OP} = 1.0 MHz)	V_{DD}	2.2	—	5.5	V
Supply Voltage	V_{LCD1}	V_{DD} - 1/3 V_{LCD}			V
	V_{LCD2}	V_{DD} - 2/3 V_{LCD}			V
	V_{LCD3}	V_{DD} - 3/3 V_{LCD}			V
Fast Clock Oscillation Frequency	f_{OSC}	—	3.52	4.2	MHz
External capacitance (f_{OSC} = 3.52 MHz)	C1 C2	— —	33 33	— —	pF pF
Slow Clock Oscillation Frequency	f_{XOSC}	—	32.768	—	MHz
External capacitance (f_{XOSC} = 32.768 KHz)	CX1 CX2	— —	18 22	— —	pF pF

12.4 DC Electrical Characteristics (5.0 Vdc)

($+4.5 \le V_{DD} \le +5.5$Vdc, V_{SS} = 0Vdc, $T_L \le T_A \le T_H$, unless otherwise noted)

Table 12-4. DC Electrical Characteristics (V_{DD} = 5.0 Vdc)

Characteristic	Symbol	Min	Typ	Max	Unit
Output Voltage					
I_{LOAD} = 10.0 µA	V_{OL}	—	—	0.1	V
I_{LOAD} = -10.0 µA	V_{OH}	V_{DD}- 0.1	—	—	V
Output High Voltage (V_{DD} = 5.0 V) (I_{LOAD} -0.4 mA) PA0:7, PC0:5, PD1:3	V_{OH}	V_{DD}- 0.8	—	—	V
Output Low Voltage (V_{DD} = 5.0 V) (I_{LOAD} = 0.8 mA) PA0:7, PC0:7, PD1:3	V_{OL}	—	—	0.4	V
Input High Voltage PA0:7, PB0:7, PC0:7, PD1:3, $\overline{\text{RESET}}$, OSC1, XOSC1	V_{IH}	0.7 X V_{DD}	—	V_{DD}	V
Input Low Voltage PA0:7, PB0:7, PC0:7, PD1:3, $\overline{\text{RESET}}$, OSC1, XOSC1	V_{IL}	V_{SS}	—	0.3 x V_{DD}	V
Supply Current (see Notes)					
Run (fop = 2.0 MHz)	I_{DD}	—	6.0	12.0	mA
Wait (fop = 2.0 MHz)	I_{DD}	—	3.0	6.0	mA
Stop					
No clock	I_{DD}	—	3.0	10.0	µA
XOSC=32.768KHz, V_{DD}=5.0V, T_A= +25 ^oC	I_{DD}	—	17.0	—	µA
Input Current (with pullups disabled) PA0:7, PB0:7, PC0:7, PD1:3, $\overline{\text{RESET}}$, OSC1, XOSC1	I_{IN}	—	—	±1.0	µA
Input Current (with pullups enabled, V_{DD} = 5.0V)					
PA0:7	I_{IN}	50	180	400	µA
PB0:7	I_{IN}	50	180	400	µA
PC0:5	I_{IN}	200	700	1400	µA
PC6:7	I_{IN}	30	125	300	µA
LCD pin output impedance					
FP0:26	Zo,FP	—	10	20	KΩ
BP0:3	Zo,BP	—	5	18	KΩ

NOTES:

1. All values shown reflect average measurements.
2. Typical values at midpoint of voltage range, 25^{o}C only.
3. Run (Operating) I_{DD}, Wait I_{DD} : Measured using external square wave clock source (f_{OSC} =4.2 MHz), all inputs 0.2 V from rail (V_{SS} or V_{DD}); no dc loads, less than 50 pF on all outputs, C_l = 20pF on OSC2.
4. Wait, Stop I_{DD}: All ports configured as inputs, V_{IL} = 0.2 V, V_{IH} = V_{DD} - 0.2 V.
5. Stop I_{DD} measured with OSC1 = V_{SS}.
6. Wait I_{DD} is affected linearly by the OSC2 capacitance.
7. Input Current measured with output transistor turned off and V_{IN} = 0V

12.5 DC Electrical Characteristics (3.3 Vdc)

($+3.0 \le V_{DD} < +4.5$Vdc, V_{SS} = 0Vdc, $T_L \le T_A \le T_H$, unless otherwise noted)

Table 12-5. DC Electrical Characteristics (V_{DD} = 3.3 Vdc)

Characteristic	Symbol	Min	Typ	Max	Unit
Output Voltage I_{LOAD} = 10.0 µA I_{LOAD} = -10.0 µA	 V_{OL} V_{OH}	 — V_{DD}- 0.1	 — —	 0.1 —	 V V
Output High Voltage (I_{LOAD} -0.4 mA) PA0:7, PC0:5, PD1:3 (I_{LOAD} -10 mA) PC5	 V_{OH} V_{OH}	 V_{DD}- 0.8 V_{DD}- 0.8	 —	 —	 V
Output Low Voltage (I_{LOAD} = 0.8 mA) PA0:7, PC0:7, PD1:3	V_{OL}	—	—	0.4	V
Input High Voltage PA0:7, PB0:7, PC0:7, PD1:3, $\overline{\text{RESET}}$, OSC1, XOSC1	V_{IH}	0.7 X V_{DD}	—	V_{DD}	V
Input Low Voltage PA0:7, PB0:7, PC0:7, PD1:3, $\overline{\text{RESET}}$, OSC1, XOSC1	V_{IL}	V_{SS}	—	0.3 x V_{DD}	V
Supply Current (see Notes) Run (f_{OP} = 1.0 MHz) Wait (fop = 1.0 MHz) Stop No clock XOSC=32.768KHz, V_{DD}= 3.3 V, T_A= +25 ^oC	 I_{DD} I_{DD} I_{DD} I_{DD}	 — — — —	 1.8 0.8 2.0 8.0	 4.0 2.0 10.0 —	 mA mA µA µA
Input Current (with pullups disabled) PA0:7, PB0:7, PC0:7, PD1:3, $\overline{\text{RESET}}$, OSC1, XOSC1	I_{IN}	—	—	±1.0	µA
Input Current (with pullups enabled, V_{DD} = 3.3V) PA0:7 PB0:7 PC0:5 PC6:7	 I_{IN} I_{IN} I_{IN} I_{IN}	 20 20 50 30	 75 75 300 100	 300 300 1000 ** 250 **	 µA µA µA µA
LCD pin output impedance FP0:26 BP0:3	 Zo,FP Zo,BP	 — —	 10 5	 20 18	 KΩ KΩ

NOTES:

1. All values shown reflect average measurements.
2. Typical values at midpoint of voltage range, 25^oC only.
3. Run (Operating) I_{DD}, Wait I_{DD} : Measured using external square wave clock source (f_{OSC} =2.0 MHz), all inputs 0.2 V from rail (V_{SS} or V_{DD}); no dc loads, less than 50 pF on all outputs, C_I = 20 pF on OSC2.
4. Wait, Stop I_{DD}: All ports configured as inputs, V_{IL} = 0.2 V, V_{IH} = V_{DD} - 0.2 V.
5. Stop I_{DD} measured with OSC1 = V_{SS}.
6. Wait I_{DD} is affected linearly by the OSC2 capacitance.
7. Input Current measured with output transistor turned off and V_{IN} = 0V

12.6 DC Electrical Characteristics (2.7 Vdc)

($+2.2 \le V_{DD} < +3.0$ Vdc, $V_{SS} = 0$Vdc, $T_L \le T_A \le T_H$, unless otherwise noted)

Table 12-6. DC Electrical Characteristics (V_{DD} = 2.7 Vdc)

Characteristic	Symbol	Min	Typ	Max	Unit
Output Voltage I_{LOAD} = 10.0 µA I_{LOAD} = -10.0 µA	 V_{OL} V_{OH}	 — V_{DD}- 0.1	 — —	 0.1 —	 V V
Output High Voltage (I_{LOAD} -0.4 mA) PA0:7, PC0:5, PD1:3	V_{OH}	V_{DD}- 0.6	—	—	V
Output Low Voltage (I_{LOAD} = 0.8 mA) PA0:7, PC0:7, PD1:3	V_{OL}	—	—	0.3	V
Input High Voltage PA0:7, PB0:7, PC0:7, PD1:3, $\overline{\text{RESET}}$, OSC1, XOSC1	V_{IH}	0.7 X V_{DD}	—	V_{DD}	V
Input Low Voltage PA0:7, PB0:7, PC0:7, PD1:3, $\overline{\text{RESET}}$, OSC1, XOSC1	V_{IL}	V_{SS}	—	0.3 x V_{DD}	V
Supply Current (see Notes) Run (fop = 1.0 MHz) Wait (fop = 1.0 MHz) Stop No clock XOSC=32.768KHz, V_{DD}= 2.7 V, T_A= +25 ^{o}C	 I_{DD} I_{DD} I_{DD} I_{DD}	 — — — —	 0.7 0.4 1.5 5.0	 2.2 1.4 10.0 —	 mA mA µA µA
Input Current (with pullups disabled) PA0:7, PB0:7, PC0:7, PD1:3, $\overline{\text{RESET}}$, OSC1, XOSC1	I_{IN}	—	—	±1.0	µA
Input Current (with pullups enabled, V_{DD} = 2.7 V) PA0:7 PB0:7 PC0:5 PC6:7	 I_{IN} I_{IN} I_{IN} I_{IN}	 5 5 30 20	 50 50 200 85	 150 150 600 200	 µA µA µA µA
LCD pin output impedance FP0:26 BP0:3	 Zo,FP Zo,BP	 — —	 10 5	 20 18	 KΩ KΩ

NOTES:

1. All values shown reflect average measurements.
2. Typical values at midpoint of voltage range, 25^{o}C only.
3. Run (Operating) I_{DD}, Wait I_{DD} : Measured using external square wave clock source (f_{OSC} =2.0 MHz), all inputs 0.2 V from rail (V_{SS} or V_{DD}); no dc loads, less than 50 pF on all outputs, C_I = 20pF on OSC2.
4. Wait, Stop I_{DD}: All ports configured as inputs, V_{IL} = 0.2 V, V_{IH} = V_{DD} - 0.2 V.
5. Stop I_{DD} measured with OSC1 = V_{SS}.
6. Wait I_{DD} is affected linearly by the OSC2 capacitance.
7. Input Current measured with output transistor turned off and V_{IN} = 0V

12.7 Control Timing (3.3 Vdc AND 5.0 Vdc)

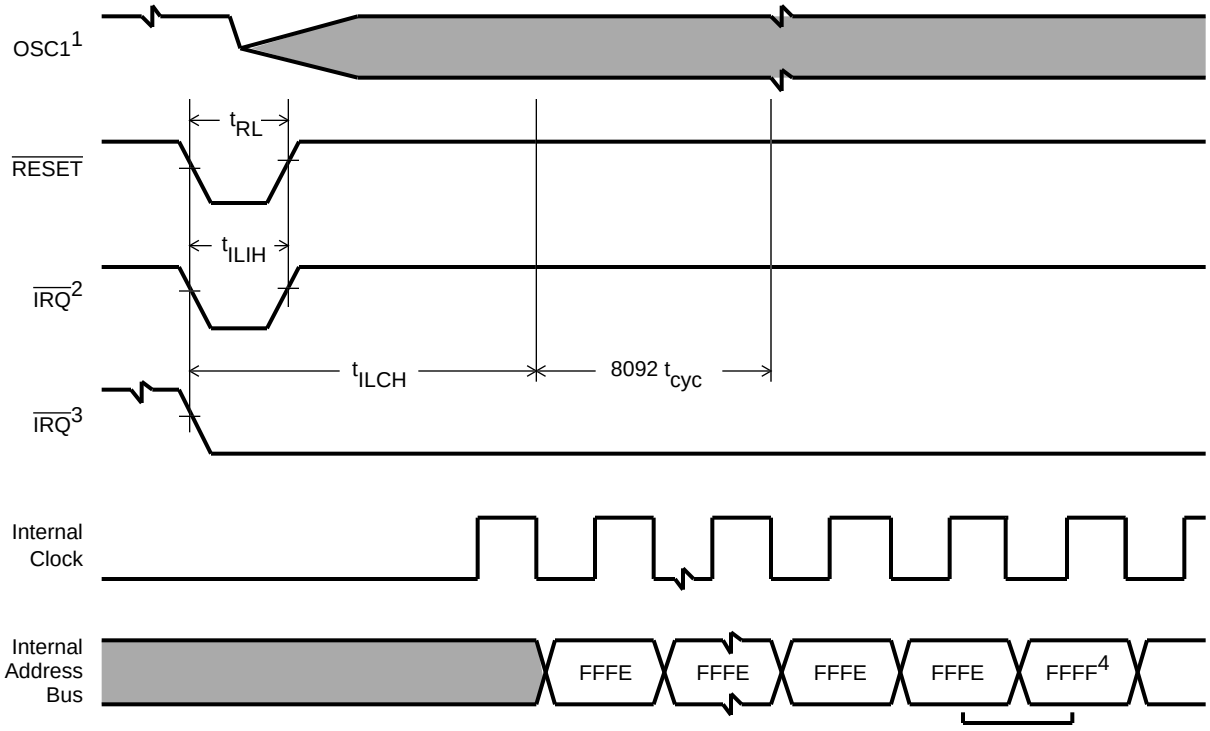
(+2.2 ≤ V_{DD} < +5.5Vdc, V_{SS} = 0Vdc, $T_L \le T_A \le T_H$, unless otherwise noted)

Table 12-7. Control Timing (2.2 Vdc to 5.5 Vdc)

Characteristic	Symbol	Min	Max	Unit
Frequency of Oscillation (OSC) Crystal External Clock	 f_{OSC} f_{OSC}	 — DC	 4.2 4.2	 MHz MHz
Internal Operating Frequency, Crystal or External Clock (f_{OSC}/2) V_{DD} = 4.5 V_{DC} to 5.5 V_{DC} V_{DD} = 2.2 V_{DC} to 5.5 V_{DC}	 f_{op} f_{op}	 — —	 2.1 1.0	 MHz MHz
Cycle Time (Fast OSC selected) V_{DD} = 4.5 V_{DC} to 5.5 V_{DC} V_{DD} = 2.2 V_{DC} to 5.5 V_{DC}	 t_{CYC} t_{CYC}	 480 1.0	 — —	 ns µs
$\overline{\text{RESET}}$ Pulse Width (when bus clock active)	t_{RL}	1.5	—	t_{cyc}
Timer Resolution Input Capture (TCAP) pulse width	 t_{RESL} t_{TH}, t_{TL}	 4.0 284	 — —	 t_{cyc} ns
Interrupt Pulse Width Low (Edge-Triggered)	t_{ILIH}	284	—	ns
Interrupt Pulse Period	t_{ILIL}	see note	—	t_{cyc}
OSC1 Pulse Width (external clock input)	t_{OH},t_{OL}	110	—	ns

NOTES:

1. The minimum period T_{ILIL} should not be less than the number of cycle times it takes to execute the interrupt service routine plus 21 t_{CYC}.
2. The system clock divider configuration (SYS1:0 bits) should be selected such that the internal operating frequency (f_{OP}) does not exceed value specified in f_{OP} for a given f_{OSC}.



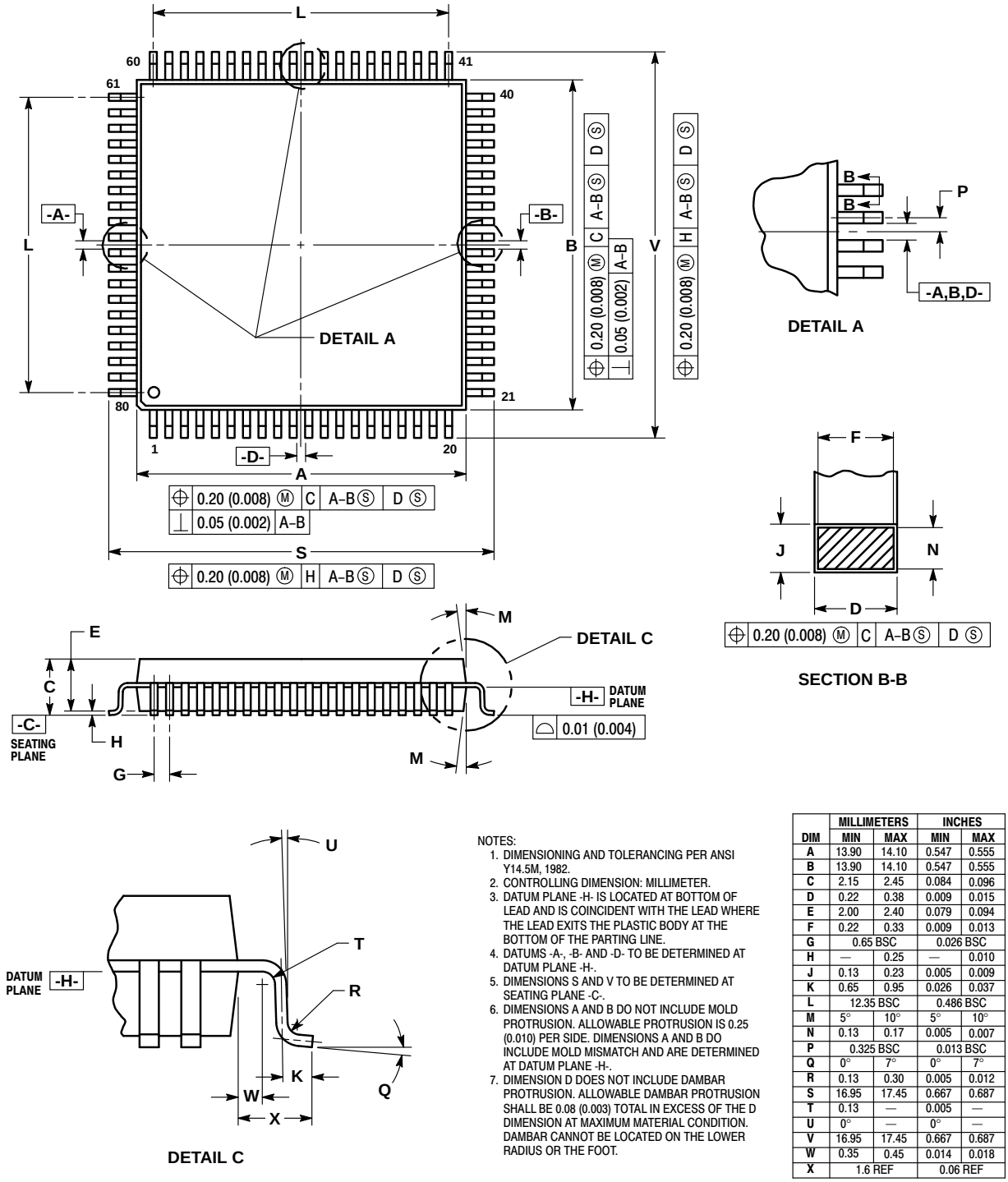
NOTES:
1. Represents the internal gating of the OSC1 pin
2. IRQ pin edge-sensitive mask option.
3. IRQ pin level and edge-sensitive mask option.
4. RESET vector address shown for timing example.

Figure 12-1. Stop Recovery Timing Diagram

SECTION 13
MECHANICAL SPECIFICATIONS

This section describes the dimensions of the following package: quad flat pack (QFP).

13.1 QUAD FLAT PACK (QFP)



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 (0.010) 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 (0.003) TOTAL IN EXCESS OF THE D DIMENSION AT MAXIMUM MATERIAL CONDITION. DAMBAR CANNOT BE LOCATED ON THE LOWER RADIUS OR THE FOOT.

	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	13.90	14.10	0.547	0.555
B	13.90	14.10	0.547	0.555
C	2.15	2.45	0.084	0.096
D	0.22	0.38	0.009	0.015
E	2.00	2.40	0.079	0.094
F	0.22	0.33	0.009	0.013
G	0.65 BSC		0.026 BSC	
H	—	0.25	—	0.010
J	0.13	0.23	0.005	0.009
K	0.65	0.95	0.026	0.037
L	12.35 BSC		0.486 BSC	
M	5°	10°	5°	10°
N	0.13	0.17	0.005	0.007
P	0.325 BSC		0.013 BSC	
Q	0°	7°	0°	7°
R	0.13	0.30	0.005	0.012
S	16.95	17.45	0.667	0.687
T	0.13	—	0.005	—
U	0°	—	0°	—
V	16.95	17.45	0.667	0.687
W	0.35	0.45	0.014	0.018
X	1.6 REF		0.06 REF	

Figure 13-1. MC68HC(7)05L16FU (Case #841B-01)

SECTION 14
ORDERING INFORMATION

This section contains instructions for ordering custom-masked ROM MCUs.

14.1 MCU Ordering Forms

To initiate an order for a ROM-based MCU, first obtain the current ordering form for the MCU from a Motorola representative. Submit the following items when ordering MCUs:

- A current MCU ordering form that is **completely filled out** (Contact your Motorola sales office for assistance.)
- A copy of the customer specification if the customer specification deviates from the Motorola specification for the MCU
- Customer's application program on one of the media listed in **14.2 Application Program Media**

The current MCU ordering form is also available through the Motorola Freeware Bulletin Board Service (BBS). The telephone number is (512) 891-FREE. After making the connection, type bbs in lower-case letters. Then press the return key to start the BBS software.

14.2 Application Program Media

Please deliver the application program to Motorola in one of the following media:

- Macintosh®[1] 3 1/2-inch diskette (double-sided 800K or double-sided high-density 1.4M)
- MS-DOS®[2] or PC-DOS™[3] 3 1/2-inch diskette (double-sided 720K or double-sided high-density 1.44M)
- MS-DOS® or PC-DOS™ 5 1/4-inch diskette (double-sided double-density 360K or double-sided high-density 1.2M)

Use positive logic for data and addresses.

When submitting the application program on a diskette, clearly label the diskette with the following information:

- Customer name
- Customer part number
- Project or product name
- File name of object code
- Date
- Name of operating system that formatted diskette
- Formatted capacity of diskette

On diskettes, the application program must be in Motorola's S-record format (S1 and S9 records), a character-based object file format generated by M6805 cross assemblers and linkers.

NOTE

Begin the application program at the first user ROM location. Program addresses must correspond exactly to the available on-chip user ROM addresses as shown in the memory map. **Write $00 in all non-user ROM locations or leave all non-user ROM locations blank**. Refer to the current MCU ordering form for additional requirements. Motorola may request pattern re-submission if non-user areas contain any non-zero code.

---

1. Macintosh is a registered trademark of Apple Computer, Inc.
2. MS-DOS is a registered trademark of Microsoft Corporation.
3. PC-DOS is a trademark of International Business Machines Corporation.

If the memory map has two user ROM areas with the same address, then write the two areas in separate files on the diskette. Label the diskette with both file names.

In addition to the object code, a file containing the source code can be included. Motorola keeps this code confidential and uses it only to expedite ROM pattern generation in case of any difficulty with the object code. Label the diskette with the file name of the source code.

14.3 ROM Program Verification

The primary use for the on-chip ROM is to hold the customer's application program. The customer develops and debugs the application program and then submits the MCU order along with the application program.

Motorola inputs the customer's application program code into a computer program that generates a listing verify file. The listing verify file represents the memory map of the MCU. The listing verify file contains the user ROM code and may also contain non-user ROM code, such as self-check code. Motorola sends the customer a computer printout of the listing verify file along with a listing verify form.

To aid the customer in checking the listing verify file, Motorola will program the listing verify file into customer-supplied blank preformatted Macintosh or DOS disks. All original pattern media are filed for contractual purposes and are not returned.

Check the listing verify file thoroughly, then complete and sign the listing verify form and return the listing verify form to Motorola. The signed listing verify form constitutes the contractual agreement for the creation of the custom mask.

14.4 ROM Verification Units (RVUs)

After receiving the signed listing verify form, Motorola manufactures a custom photographic mask. The mask contains the customer's application program and is used to process silicon wafers. The application program cannot be changed after the manufacture of the mask begins. Motorola then produces 10 MCUs, called RVUs, and sends the RVUs to the customer. RVUs are usually packaged in unmarked ceramic and tested to 5 Vdc at room temperature. RVUs are not tested to environmental extremes because their sole purpose is to demonstrate that the customer's user ROM pattern was properly implemented. The 10 RVUs are free of charge with the minimum order quantity. These units are not to be used for qualification or production. RVUs are not guaranteed by Motorola Quality Assurance.

14.5 MC Order Numbers

Table 14-1 shows the MC order numbers for the available package types.

Table 14-1. MC Order Numbers

MC Order Number	Operating Temperature Range
MC68HC05L16FU	–0° to 70°C
MC68HC(7)05L16FU	–0° to 70°C

NOTE: FU = Quad Flat Pack (QFP)

How to reach us:

MFAX: RMFAX0@email.sps.mot.com - TOUCHTONE (602) 244-6609
INTERNET: http://Design-net.com
USA/EUROPE: Motorola Literature Distribution; P.O. Box 20912; Phoenix, Arizona 85036. 1-800-441-2447
JAPAN: Nippon Motorola Ltd.; Tatsumi-SPD-JLDC, Toshikatsu Otsuki, 6F Seibu-Butsuryu-Center, 3-14-3 Tatsumi Koto-Ku, Tokyo 135, Japan. 03-3521-8315
HONG KONG: Motorola Semiconductors H.K. Ltd.; 8B Tai Ping Industrial Park, 51 Ting Tok Road, Tai Po, N.T., Hong Kong. 852-26629298

MOTOROLA

HC05L16GRS/D