

68HC05CT4 68HC705CT4

SPECIFICATION (General Release)

© January 2, 1996

CSIC System Design Group
Austin, Texas

Motorola reserves the right to make changes without further notice to any products herein to improve reliability, function or design. Motorola does not assume any liability arising out of the application or use of any product or circuit described herein; neither does it convey any license under its patent rights nor the rights of others. Motorola products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Motorola product could create a situation where personal injury or death may occur. Should Buyer purchase or use Motorola products for any such unintended or unauthorized application, Buyer shall indemnify and hold Motorola and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Motorola was negligent regarding the design or manufacture of the part.

TABLE OF CONTENTS

Section		Page
---------	--	------

SECTION 1 GENERAL DESCRIPTION

1.1	MASK OPTIONS	1-2
1.2	SIGNAL DESCRIPTION	1-5
1.2.1	V_{DD} and V_{SS}	1-5
1.2.2	V_{DD2} and V_{SS2}	1-5
1.2.3	IRQ (Maskable Interrupt Request)	1-5
1.2.4	OSC1 and OSC2	1-6
1.2.5	RESET	1-7
1.2.6	PA0-PA7	1-7
1.2.7	PB0-PB7	1-7
1.2.8	PC0-PC7/PWM	1-7
1.2.9	PD0/CMP3+, PD1/CMP1-, PD2/CMP12+, PD3/CMP2-, PD4/SDIO, PD5/SCK, and PD6/TCMP	1-7
1.2.10	TCAP	1-7
1.2.11	FinT and FinR	1-7
1.2.12	PDoutT and PDoutR	1-7

SECTION 2 MEMORY

2.1	MEMORY MAP	2-1
2.1.1	ROM	2-1
2.1.2	ROM Security	2-1
2.1.3	RAM	2-1
2.2	INPUT/OUTPUT PROGRAMMING	2-4
2.2.1	Port A	2-4
2.2.2	Port B	2-4
2.2.3	Port C	2-4
2.2.4	Port D	2-5
2.2.5	Input/Output Port Pin Programming	2-5

SECTION 3 CENTRAL PROCESSING UNIT (CPU)

3.1	ACCUMULATOR (A)	3-2
3.2	INDEX REGISTER (X)	3-2
3.3	CONDITION CODE REGISTER (CCR)	3-2
3.3.1	Half Carry (H)	3-2
3.3.2	Interrupt (I)	3-2
3.3.3	Negative (N)	3-2
3.3.4	Zero (Z)	3-2
3.3.5	Carry/Borrow (C)	3-3

TABLE OF CONTENTS

Section	Page
3.4 STACK POINTER (SP)	3-3
3.5 PROGRAM COUNTER (PC)	3-3

SECTION 4 INTERRUPTS

4.1 CPU INTERRUPT PROCESSING	4-1
4.2 RESET INTERRUPT SEQUENCE	4-2
4.3 SOFTWARE INTERRUPT (SWI)	4-2
4.4 HARDWARE INTERRUPTS	4-4
4.5 EXTERNAL INTERRUPT (IRQ)	4-4
4.6 EXTERNAL INTERRUPT TIMING	4-5
4.7 16-BIT TIMER INTERRUPT	4-5
4.8 SSI INTERRUPT	4-5
4.9 CORE TIMER INTERRUPT	4-5
4.10 COMPARATOR 3 INTERRUPT	4-5

SECTION 5 RESETS

5.1 EXTERNAL RESET (RESET)	5-1
5.2 INTERNAL RESETS	5-3
5.2.1 Power-On Reset (POR)	5-3
5.2.2 Computer Operating Properly Reset (COPR)	5-3
5.2.3 Illegal Address	5-5

SECTION 6 OPERATING MODES

6.1 SINGLE-CHIP MODE	6-1
6.2 LOW-POWER MODES	6-1
6.2.1 Stop Mode	6-1
6.2.2 Stop Recovery	6-2
6.2.3 Wait Mode	6-2
6.2.4 Low-Power Wait	6-3

SECTION 7 16-BIT TIMER

7.1 COUNTER REGISTER - \$18:\$19 AND \$1A:\$1B	7-1
7.2 OUTPUT COMPARE REGISTER — \$16:\$17	7-3
7.3 INPUT CAPTURE REGISTER — \$14:\$15	7-3
7.4 TIMER CONTROL REGISTER (TCR) — \$12	7-4
7.5 TIMER STATUS REGISTER (TSR) — \$13	7-5

TABLE OF CONTENTS

Section	Page
7.6	TIMER DURING WAIT MODE 7-6
7.7	TIMER DURING STOP MODE 7-6
7.8	TIMER POWER SUPPLY SOURCE 7-6

SECTION 8
SYNCHRONOUS SERIAL INTERFACE (SSI)

8.1	SIGNAL FORMAT 8-2
8.1.1	Serial Clock (SCK) 8-2
8.1.2	Serial Data In/Out (SDIO) 8-2
8.2	SSI REGISTERS 8-3
8.2.1	SSI Control Register (SCR) \$1E 8-3
8.2.2	SSI Status Register (SSR) \$1C 8-5
8.2.3	SSI Data Register (SDR) \$1D 8-5
8.3	OPERATION DURING WAIT MODE 8-6
8.4	OPERATION DURING STOP MODE 8-6
8.5	SSI POWER SUPPLY SOURCE 8-6

SECTION 9
CORE TIMER

9.1	CORE TIMER CTRL AND STATUS REGISTER (CTCSR) \$08 9-2
9.2	CORE TIMER COUNTER REGISTER (CTCR) \$09 9-3
9.3	COMPUTER OPERATING PROPERLY (COP) RESET 9-4
9.4	TIMER DURING WAIT MODE 9-4
9.5	CORE TIMER POWER SUPPLY SOURCE 9-4

SECTION 10
DUAL PHASE-LOCKED LOOP (PLL)

10.1	REGISTERS 10-2
10.1.1	Dual Control Register (PLLCR) \$0A 10-2
10.1.2	12-Bit Reference Counter Modulus Register (PLLRC) \$0B:\$0C 10-3
10.1.3	16-Bit Transmit Counter Modulus Register (PLLTX) \$0D:\$0E 10-4
10.1.4	16-Bit Receive Counter Modulus Register (PLLRX) \$0F:\$10 10-4
10.2	PLL POWER SUPPLY SOURCE 10-5

SECTION 11
PULSE WIDTH MODULATOR

11.1	FUNCTIONAL DESCRIPTION 11-1
11.2	REGISTER 11-2
11.3	PWM DATA REGISTER (PWMDR) \$20 11-3
11.4	PWM DURING WAIT MODE 11-3

TABLE OF CONTENTS

Section	Page
11.5 PWM DURING STOP MODE	11-3
11.6 PWM DURING RESET	11-3
11.7 PWM POWER SUPPLY SOURCE	11-3

SECTION 12
COMPARATORS

12.1 COMPARATOR CONTROL/STATUS REGISTER	12-2
12.2 READING COMPARATOR OUTPUTS	12-3
12.3 COMPARATOR DURING WAIT	12-3
12.4 COMPARATOR DURING STOP	12-3
12.5 COMPARATOR POWER SUPPLY SOURCE	12-3

SECTION 13
MISCELLANEOUS REGISTER

13.1 MISCELLANEOUS CONTROL REGISTER.....	13-1
13.2 MISCELLANEOUS REGISTER POWER SUPPLY SOURCE	13-1

SECTION 14
INSTRUCTION SET

14.1 ADDRESSING MODES	14-1
14.1.1 Inherent.....	14-1
14.1.2 Immediate	14-2
14.1.3 Direct	14-2
14.1.4 Extended.....	14-2
14.1.5 Indexed, No Offset.....	14-2
14.1.6 Indexed, 8-Bit Offset	14-2
14.1.7 Indexed, 16-Bit Offset	14-3
14.1.8 Relative	14-3
14.2 INSTRUCTION TYPES.....	14-4
14.2.1 Register/Memory Instructions	14-4
14.2.2 Read-Modify-Write Instructions	14-5
14.2.3 Jump/Branch Instructions	14-5
14.2.4 Bit Manipulation Instructions.....	14-7
14.2.5 Control Instructions.....	14-7
14.3 INSTRUCTION SET SUMMARY	14-8

SECTION 15
ELECTRICAL SPECIFICATIONS

15.1 MAXIMUM RATINGS.....	15-1
15.2 THERMAL CHARACTERISTICS	15-1

TABLE OF CONTENTS

Section	Page
15.3 DC ELECTRICAL CHARACTERISTICS (5.0 V)	15-2
15.4 DC ELECTRICAL CHARACTERISTICS (3.3 V)	15-3
15.5 CONTROL TIMING (3.3 V AND 5.0 V)	15-4
15.6 COMPARATORS	15-4
15.7 PWM TIMING	15-5
15.8 PLL SIGNAL INPUT	15-5

SECTION 16
MECHANICAL SPECIFICATIONS

16.1 44-LEAD PLASTIC LEADED CHIP CARRIER (CASE 777-02)	16-1
16.2 44-LEAD QUAD FLAT PACK (CASE 824A-01)	16-2

SECTION 17
ORDERING INFORMATION

17.1 MCU ORDERING FORMS	17-1
17.2 APPLICATION PROGRAM MEDIA	17-2
17.3 ROM PROGRAM VERIFICATION	17-3
17.4 ROM VERIFICATION UNITS (RVUS)	17-3
17.5 MC ORDER NUMBERS	17-4

APPENDIX A
MC68HC705CT4

A.1 MASK OPTIONS	A-1
A.2 $\overline{\text{IRQ}}/V_{\text{PP}}$ (MASKABLE INTERRUPT REQUEST)	A-1
A.3 MEMORY	A-2
A.4 EPROM	A-2
A.5 BOOTLOADER MODE	A-2
A.5.1 Bootloader Functions	A-2
A.5.2 Programming Register (PROG) \$11	A-7
A.5.3 Mask Option Register (MOR)	A-9
A.6 EPROM ELECTRICAL SPECIFICATIONS	A-10
A.6.1 Maximum Ratings	A-10
A.6.2 DC Electrical Characteristics (5.0 V)	A-10
A.6.3 DC Electrical Characteristics (3.3 V)	A-11
A.6.4 EPROM Control Timing (3.3 V and 5.0 V)	A-11

LIST OF FIGURES

Figure	Title	Page
1-1	MC68HC05CT4 Block Diagram	1-3
1-2	44-Lead PLCC Pinout of the MC68HC05CT4.....	1-4
1-3	44-Lead QFP Pinout of the MC68HC05CT4.....	1-5
1-4	Oscillator Connections	1-6
2-1	MC68HC05CT4 8K Memory Map	2-2
2-2	I/O Registers	2-3
2-3	Port C Pullup Option	2-5
2-4	I/O Circuitry	2-6
3-1	Programming Model.....	3-1
3-2	Stacking Order	3-1
4-1	Interrupt Processing Flowchart	4-3
4-2	IRQ Function Block Diagram.....	4-4
5-1	Reset Block Diagram	5-1
5-2	RESET and POR Timing Diagram	5-2
5-3	COP Watchdog Timer Location	5-4
6-1	Single-Chip Mode Pinout of the MC68HC05CT4.....	6-2
6-2	Stop Recovery Timing Diagram	6-3
6-3	STOP/WAIT Flowchart.....	6-4
7-1	16-Bit Timer Block Diagram	7-2
7-2	Timer Control Register	7-4
7-3	Timer Status Register	7-5
8-1	SSI Block Diagram	8-1
8-2	Serial I/O Port Timing.....	8-3
8-3	SSI Control Register	8-3
8-4	SSI Status Register.....	8-5
8-5	SSI Data Register	8-6

LIST OF FIGURES

Figure	Title	Page
9-1	Core Timer Block Diagram.....	9-1
9-2	Core Timer Control and Status Register	9-2
9-3	Timer Counter Register.....	9-3
10-1	Dual PLL Block Diagram	10-1
10-2	Dual PLL Control Register	10-2
10-3	12-Bit Reference Counter (PLLRC)	10-3
10-4	16-Bit Transmit Counter (PLLTX).....	10-4
10-5	16-Bit Receive Counter (PLLRX)	10-4
10-6	Counter Structure Block Diagram	10-5
11-1	PWM Block Diagram.....	11-1
11-2	PWM Waveforms	11-2
11-3	PWM Data Register	11-3
12-1	Comparator Block Diagram.....	12-1
12-2	Comparator Control/Status Register (CMPCSR) \$22	12-2
13-1	Miscellaneous Control Register (MISCR) \$21	13-1
15-1	Stop Recovery Timing Diagram	15-5
A-1	MC68HC705CT4 8K Memory Map	A-3
A-2	Programmer Interface to Host.....	A-4
A-3	MC68HC705CT4 Bootloader Flowchart.....	A-5
A-4	MC68HC705CT4 Programming Circuit.....	A-6
A-5	Programming Register	A-7
A-6	Mask Option Register.....	A-9

LIST OF TABLES

Table	Title	Page
2-1	I/O PIn Functions	2-5
4-1	Vector Address for Interrupts and Reset.....	4-2
5-1	COP Watchdog Timer Recommendations	5-4
6-1	Operating Mode Conditions	6-1
8-1	SSI Rates	8-4
9-1	RTI and COP Rates at 2.048 MHz.....	9-3
10-1	PLL Reference Counter Select	10-2
14-1	Register/Memory Instructions	14-4
14-2	Read-Modify-Write Instructions.....	14-5
14-3	Jump and Branch Instructions.....	14-6
14-4	Bit Manipulation Instructions	14-7
14-5	Control Instructions	14-7
14-6	Instruction Set Summary.....	14-8
14-7	Opcode Map.....	14-14
17-1	MC Order Numbers.....	17-4
A-1	Bootloader Functions	A-4
A-2	Read/Write Based on Mode and LATCH Bit	A-9

SECTION 1

GENERAL DESCRIPTION

The MC68HC05CT4 is a 44-pin member of the MC68HC05 microcontroller family, which supports cordless telephone applications. The memory map includes 5376 bytes of on-chip ROM and 256 bytes of RAM. The microcontroller unit has three 8-bit input/output (I/O) ports A, B, and C, and one 7-bit I/O port: D. Port C has pull-up options and keyscan capability. The MC68HC05CT4 includes a bird core, a bird timer, a synchronous serial interface (SSI), 16-bit timer, a dual 60 MHz phase-locked loop (PLL), pulse width modulator (PWM), and an on-chip computer operating properly (COP) watchdog circuit.

Features of the MC68HC05CT4 include:

- Low Cost
- HC05 Core
- 44-Pin Plastic Leaded Chip Carrier (PLCC) Package
- 10.24 MHz On-Chip Crystal Oscillator
- 2.048 MHz Internal CPU Speed
- 5136 Bytes of User ROM
- 240 Bytes of Self-Check ROM
- 256 Bytes of On-Chip RAM
- 16-Bit Timer
- 31 Bidirectional I/O Lines
- Power-Saving Stop and Wait Modes
- Core Timer
- Dual 60 MHz PLL
- Simple Serial with Bidirectional Data (SSI)
- Keyscan Interrupt with Pullups on Port C
- Mask Selectable Options:
 - COP Watchdog Timer
 - Edge-Sensitive or Edge and Level-Sensitive Interrupt Trigger
 - Port C Pullups for Keyscan
 - One Channel 6-Bit Pulse Width Modulator (PWM)

- Three Comparators
- ROM Security

1.1 MASK OPTIONS

The MC68HC05CT4 has eight mask options:

- Six Port C Pullups
- The IRQ Sensitivity
- COP Enable/Disable

These are non-programmable options in that they are selected at the time of code submission (when masks are made). These options are as follows:

PC7PU - Port C7 Pullup

This bit enables or disables the pullup on port C bit 7.

- 1 = Enables the pullup
- 0 = Disables the pullup

NOTE

Since port C bit 7 also is shared with the PWM output, PC7PU cannot be selected if PWM output is needed.

PC6PU - Port C6 Pullup

This option enables or disables the pullup on port C bit 6.

- 1 = Enables pullup
- 0 = Disables pullup

PC5PU - Port C5 Pullup

This option enables or disables the pullup on port C bit 5.

- 1 = Enables pullup
- 0 = Disables pullup

PC4PU - Port C4 Pullup

This option enables or disables the pullup on port C bit 4.

- 1 = Enables pullup
- 0 = Disables pullup

PC23PU - Port C Bits 2 and 3 Pullups

When PC23PU = 1, the pullups on port C bits 2 and 3 pin are enabled simultaneously.

When PC23PU = 0, the pullups on port C bits 2 and 3 pin are disabled simultaneously.

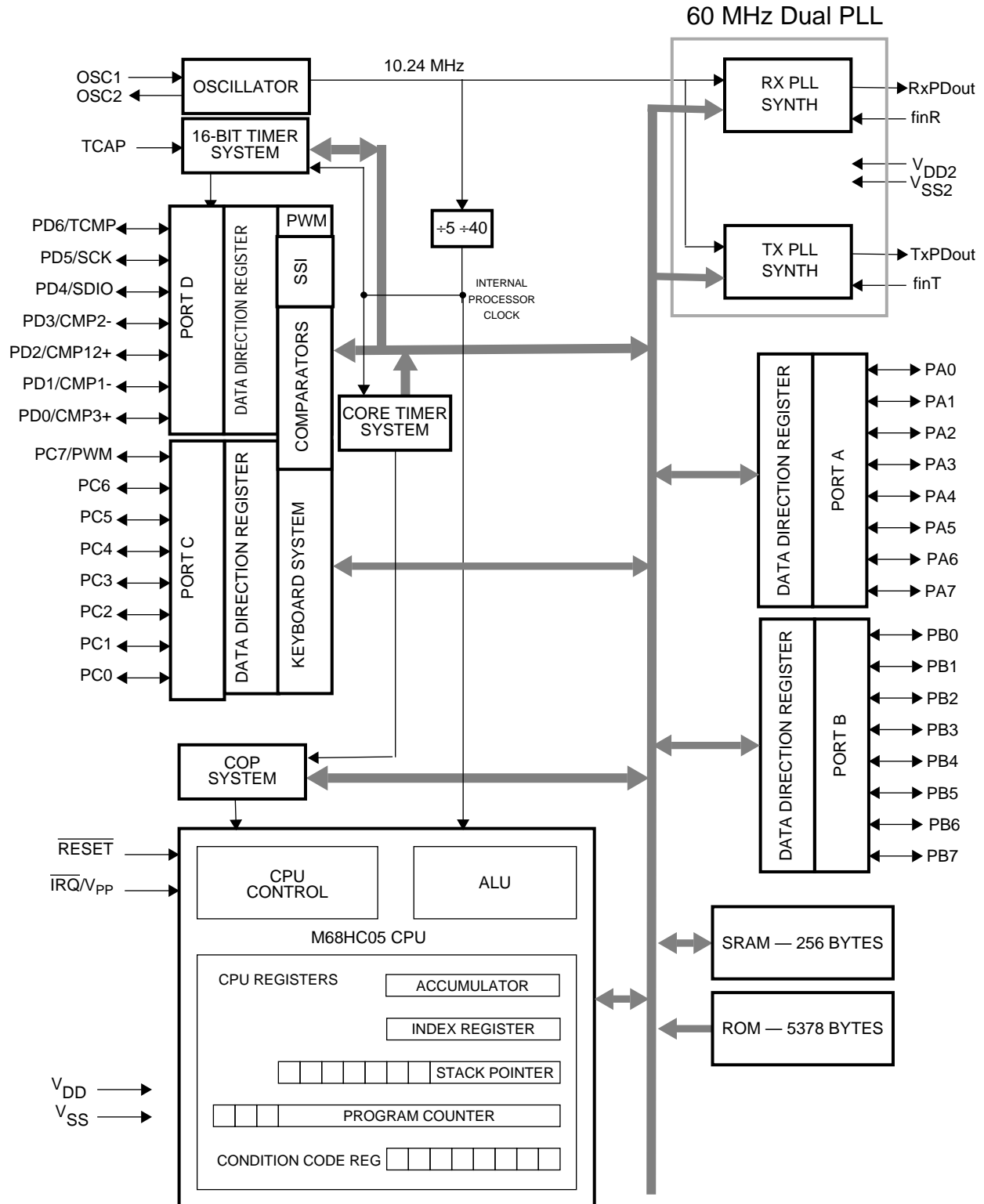


Figure 1-1. MC68HC05CT4 Block Diagram

PC01PU - Port C Bits 0 and 1 Pullups

When PC01PU = 1, the pullups on port C bits 0 and 1 pin are enabled simultaneously.

When PC01PU = 0, the pullups on port C bits 0 and 1 pin are disabled simultaneously.

COP - COP Enable

When the COP option is selected, the COP watchdog timer is enabled.

When the COP option is deselected, the COP watchdog timer is disabled.

IRQ - IRQ Sensitivity

When the IRQ option is selected (IRQEN = 1), edge and level-sensitive IRQ is enabled.

When the IRQ option is deselected (IRQEN = 0), edge-only sensitive IRQ is enabled.

NOTE

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

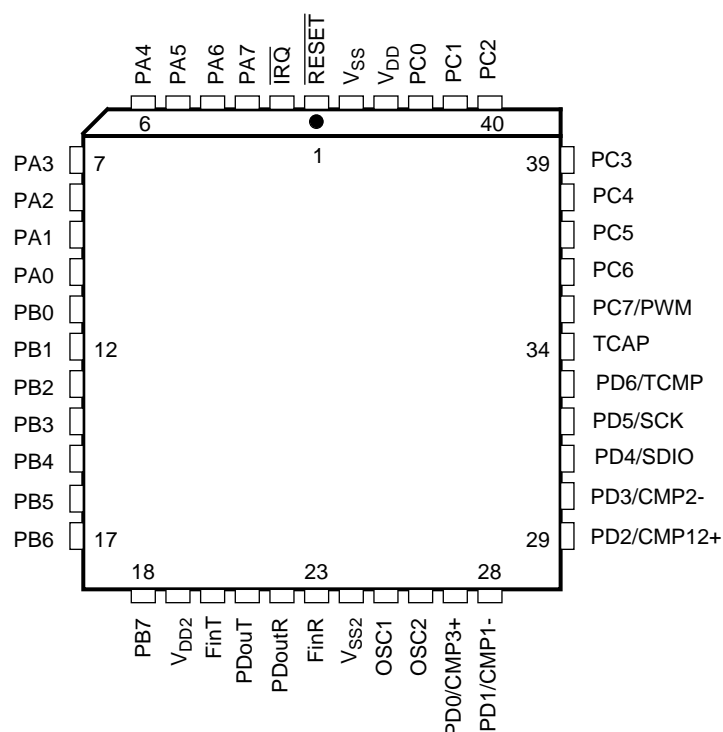


Figure 1-2. 44-Lead PLCC Pinout of the MC68HC05CT4

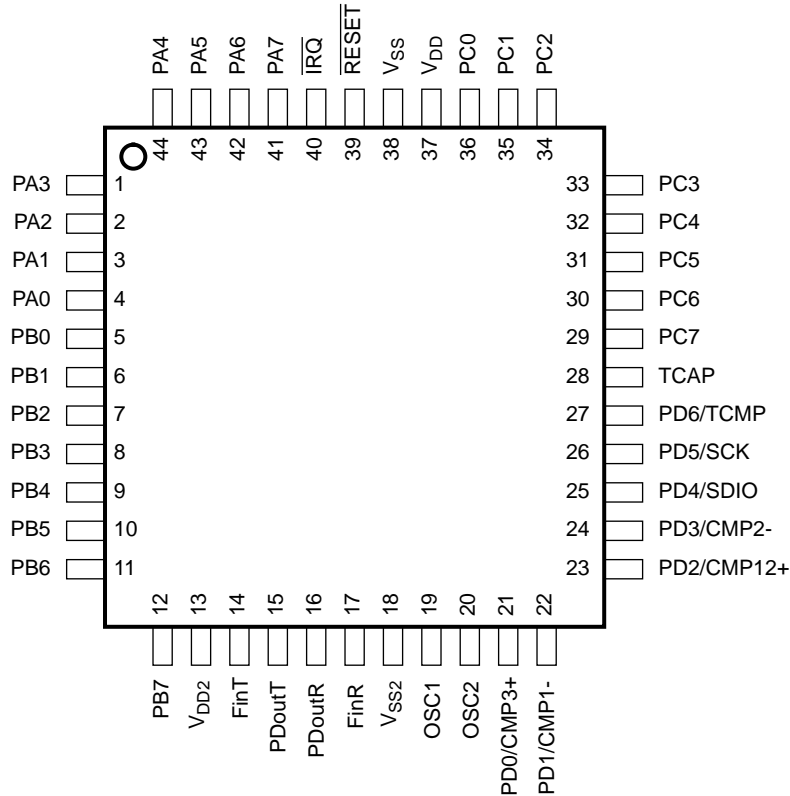


Figure 1-3. 44-Lead QFP Pinout of the MC68HC05CT4

1.2 SIGNAL DESCRIPTION

The following paragraphs provide a description of the signals.

1.2.1 V_{DD} and V_{SS}

Power is supplied to the microcontroller's digital circuits using these two pins. V_{DD} is the positive supply and V_{SS} is ground.

1.2.2 V_{DD2} and V_{SS2}

Power is supplied to noise-susceptible circuitry such as the phase-locked loop (PLL), comparators, and oscillators that require "cleaner" supplies using these two pins. V_{DD2} is the positive supply and V_{SS2} is ground.

1.2.3 \overline{IRQ} (Maskable Interrupt Request)

This pin has a mask option as specified by the user that provides one of two different choices of interrupt triggering sensitivity. The options are:

1. Negative edge-sensitive triggering only
2. Both negative-edge-sensitive and level-sensitive triggering.

The microcontroller unit (MCU) completes the current instruction before it responds to the interrupt request. When $\overline{\text{IRQ}}$ goes low for at least one t_{ILIH} , a logic one is latched internally to signify that an interrupt has been requested. When the MCU completes its current instruction, the interrupt latch is tested. If the interrupt latch contains a logic one and the interrupt mask bit (I bit) in the condition code register is clear, the MCU then begins the interrupt sequence.

If the option is selected to include level-sensitive triggering, the $\overline{\text{IRQ}}$ input requires an external resistor to V_{DD} for wire-OR operation.

The $\overline{\text{IRQ}}$ pin contains an internal Schmitt trigger as part of its input to improve noise immunity.

NOTE

The voltage on this pin affects the mode of operation. See **SECTION 6 OPERATING MODES**.

1.2.4 OSC1 and OSC2

These pins provide control input for an on-chip clock oscillator circuit that drives the PLL reference source. A crystal resonator, a ceramic resonator, or an external signal connects to these pins providing a system clock. The oscillator frequency is selectable between five or forty times the internal bus rate. Typical oscillator configurations and component values are shown in Figure 1-4. The manufacturer of the crystal should be consulted, since actual component values are dependent on the type of crystal used.

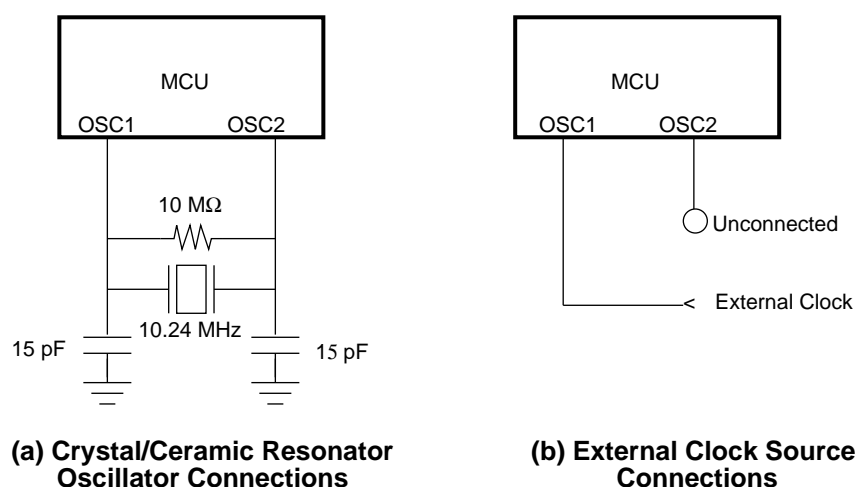


Figure 1-4. Oscillator Connections

1.2.5 **RESET**

This active low pin is used to reset the MCU to a known start-up state by pulling **RESET** low. The **RESET** pin contains an internal Schmitt trigger as part of its input to improve noise immunity. See **SECTION 5 RESETS**.

1.2.6 **PA0-PA7**

These eight I/O lines comprise port A. The state of any pin is software programmable, and all port A lines are configured as input during power-on or reset.

1.2.7 **PB0-PB7**

These eight I/O lines comprise port B. The state of any pin is software programmable and all port B lines are configured as input during power-on or reset.

1.2.8 **PC0-PC7/PWM**

These eight I/O lines comprise port C. All port C lines are configured as input during power-on or reset. Port C has pullup devices and interrupt capability by pin; however, the state of any pin is determined by the user at the time of code submission. For a detailed description of I/O programming, refer to **2.2.5 Input/Output Port Pin Programming**. PC7 is shared with the output of the pulse-width modulation (PWM) function.

1.2.9 **PD0/CMP3+, PD1/CMP1-, PD2/CMP12+, PD3/CMP2-, PD4/SDIO, PD5/SCK, and PD6/TCMP**

These seven port lines comprise port D. The state of any pin is software programmable and the lines are configured as input during power-on or reset. PD4 and PD5 are shared with the SSI subsystem, PD6 is shared with the 16-bit timer subsystem, and PD0 through PD3 are shared with the comparators. For a detailed description on I/O programming, refer to **2.2.5 Input/Output Port Pin Programming**.

1.2.10 **TCAP**

This pin is used for the 16-bit timer input capture operation. Depending on the value of the IEDG bit in the timer control register (TCR), the appropriate level of transition on TCAP will be monitored. When the correct level of transition has occurred, the free-running counter will be transferred to the input capture register.

1.2.11 **FinT and FinR**

These pins are inputs to the PLL transmit and the receive counters, respectively. They typically are driven by the loop VCO and are also AC-coupled. The minimum input signal level is 200 mV peak to peak @ 60.0 MHz.

1.2.12 **PDoutT and PDoutR**

These PLL pins are three-state outputs of the transmit and receive phase detectors, respectively, for use as either loop error signals or phase detector signals.

SECTION 2 MEMORY

This section describes the organization of the on-chip memory. The MC68HC05CT4 8K memory map is shown in Figure 2-1 and the I/O registers in Figure 2-2.

2.1 MEMORY MAP

The MC68HC05CT4 has an 8-Kbyte memory map consisting of user ROM, RAM, self-check ROM, and I/O.

2.1.1 ROM

The user ROM consists of 5120 bytes of ROM located from \$0B00 to \$1EFF and 16 bytes of user vectors located from \$1FF0 to \$1FFF.

The self-check ROM is located from \$1F00 to \$1FEF.

Twelve of the user vectors, \$1FF4 through \$1FFF, are dedicated to reset and interrupt vectors. The four remaining locations — \$1FF0, \$1FF1, \$1FF2, and \$1FF3 — are general-purpose user ROM locations.

2.1.2 ROM Security

Security has been incorporated into the CT4 to prevent external viewing of the ROM contents. This feature assists in keeping customer-developed software proprietary.

2.1.3 RAM

The user RAM consists of 256 bytes of a shared stack area. The RAM starts at address \$0030 and ends at address \$012F. The stack begins at address \$00FF. The stack pointer can access 64 bytes of RAM in the range \$00FF to \$00C0.

NOTE

Using the stack area for data storage or temporary work locations requires care to prevent it from being overwritten due to stacking from an interrupt or subroutine call.

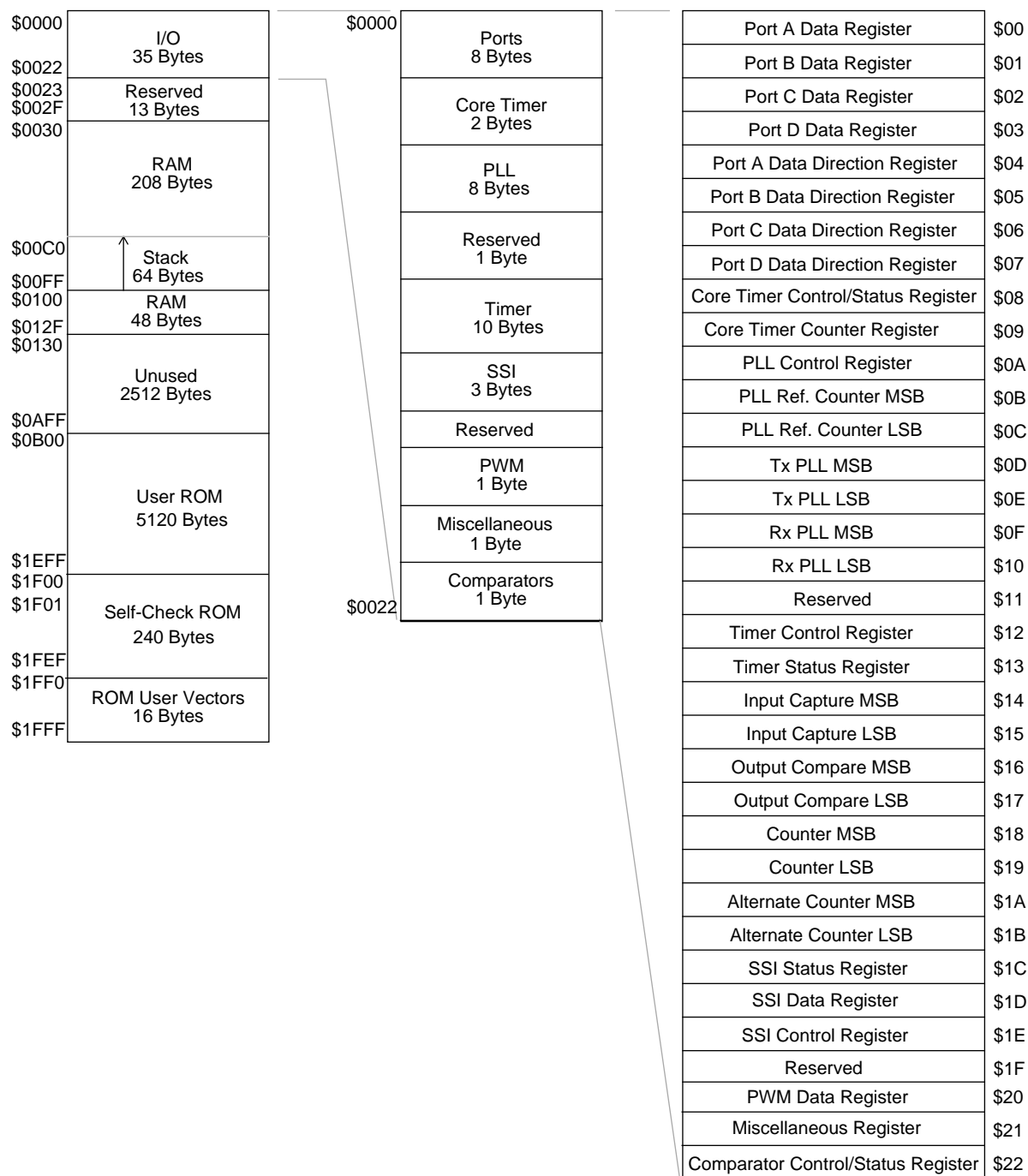


Figure 2-1. MC68HC05CT4 8K Memory Map

ADDR	REGISTER	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
\$00	PORT A DATA								
\$01	PORT B DATA								
\$02	PORT C DATA								
\$03	PORT D DATA	0							
\$04	PORT A DDR								
\$05	PORT B DDR								
\$06	PORT C DDR								
\$07	PORT D DDR	0							
\$08	TIMER CONTROL & STATUS REGISTER	CTOF	RTIF	TOFE	RTIE	TOFC	RTFC	RT1	RT0
\$09	TIMER COUNTER REGISTER								
\$0A	PLL CONTROL REGISTER	0	TLOCK	RLOCK	REFON	TXON	RXON	PLS1	PLS0
\$0B	PLL REFERENCE COUNTER MSB	0	0	0	0				
\$0C	PLL REFERENCE COUNTER LSB								
\$0D	PLL TX COUNTER MSB								
\$0E	PLL TX COUNTER LSB								
\$0F	PLL RX COUNTER MSB								
\$10	PLL RX COUNTER LSB								
\$11	reserved								
\$12	TIMER CONTROL REGISTER	ICIE	OCIE	TOIE	0	0	TON	IEDGE	OLVL
\$13	TIMER STATUS REGISTER	ICF	OCF	0	0	0	0	0	0
\$14	TIMER INPUT CAPTURE MSB								
\$15	TIMER INPUT CAPTURE LSB								
\$16	TIMER OUTPUT COMPARE MSB								
\$17	TIMER OUTPUT COMPARE LSB								
\$18	TIMER COUNTER MSB								
\$19	TIMER COUNTER LSB								
\$1A	TIMER ALTERNATE COUNTER MSB								
\$1B	TIMER ALTERNATE COUNTER LSB								
\$1C	SSI STATUS REGISTER	SF	DCOL	0	0	0	0	0	0
\$1D	SSI DATA REGISTER								
\$1E	SSI CONTROLREGISTER	SIE	SE	LSBF	MSTR	CPOL	T/R	SR1	SR0
\$1F	reserved								
\$20	PWM DATA	0	0						
\$21	MISCELLANEOUS REGISTER	0	0	0	0	SPEED	COE	PWME	0
\$22	COMPARATOR CONTROL/STATUS	CMP3	CMP2	CMP1	CM3IE	CM3IC	CEN3	CEN2	CEN1

Figure 2-2. I/O Registers

2.2 INPUT/OUTPUT PROGRAMMING

In user mode 31 lines are arranged as one 7-bit and three 8-bit ports. Most of these port pins are programmable as either inputs or outputs under software control of the data direction registers, though some are input only.

NOTE

To avoid a glitch on the output pins, write data to the I/O port data register before writing a logic one to the corresponding data direction register.

2.2.1 Port A

Port A is an 8-bit bidirectional port that does not share any of its pins with other subsystems. The port A data register is at \$0000 and the data direction register (DDR) is at \$0004. Reset does not affect the data registers, but clears the data direction registers, thereby returning the ports to inputs. Writing a logic one to a DDR bit sets the corresponding port bit to output mode.

2.2.2 Port B

Port B is an 8-bit bidirectional port. The port B data register is at \$0001 and the data direction register (DDR) is at \$0005. Reset does not affect the data registers, but clears the data direction registers, thereby returning the ports to inputs. Writing a logic one to a DDR bit sets the corresponding port bit to output mode.

2.2.3 Port C

Port C is an 8-bit bidirectional port. The port C data register is at \$0002 and the data direction register (DDR) is at \$0006. Reset does not affect the data registers, but clears the data direction registers, thereby returning the ports to inputs. Writing a logic one to a DDR bit sets the corresponding port pin to output mode. Each of the port C pins has an optional pullup device. When the DDR bit is cleared and the pullup device is enabled, the pin will be a pullup *and* an interrupt pin. The edge- or edge- and level-sensitivity of the IRQ pin also pertains to the enabled port C pins. Care needs to be taken when using port C pins that have the pullup enabled. Before switching from an output to an input, the data should be preconditioned to a logic one to prevent an interrupt from occurring. Port C bit 7 is also shared with the PWM output. When PC7 is used as the PWM output, its pullup option should not be selected (see Figure 2-3).

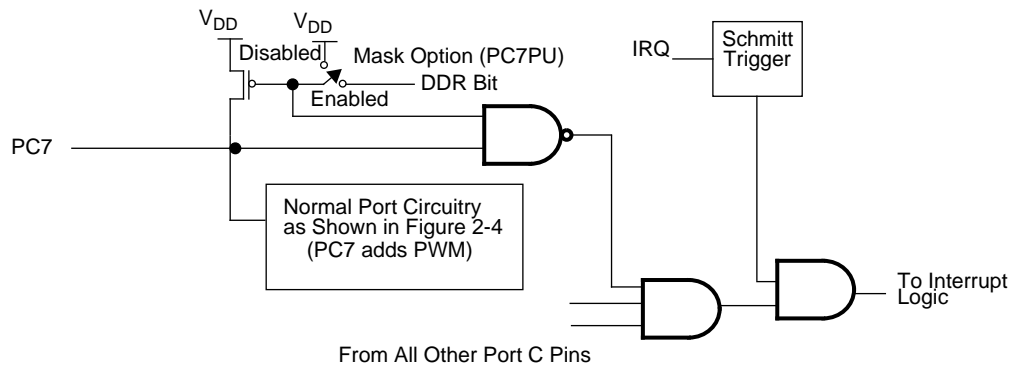


Figure 2-3. Port C Pullup Option

2.2.4 Port D

Port D is a 7-bit bidirectional port. Two of its pins are shared with the SSI subsystem, four are shared with the comparators, and one is shared with the timer. During reset, all seven bits become valid input ports because all special function output drivers associated with the timer and SSI subsystems are disabled.

2.2.5 Input/Output Port Pin Programming

Port pins may be programmed as inputs or outputs under software control. The direction of the pins is determined by the state of the corresponding bit in the port data direction register (DDR). Each I/O port has an associated DDR. Any I/O port pin is configured as an output if its corresponding DDR bit is set to a logic one. A pin is configured as an input if its corresponding DDR bit is cleared to a logic zero.

At power-on or reset, all DDRs are cleared, which configures all pins as inputs. The data direction registers are capable of being written to or read by the processor. During the programmed output state, a read of the data register actually reads the value of the output data latch and not the I/O pin.

Refer to Table 2-1 and to Figure 2-4 for additional information.

Table 2-1. I/O Pin Functions

R/W	DDR	I/O Pin Functions
0	0	The I/O pin is in input mode. Data is written into the output data latch.
0	1	Data is written into the output data latch and output to the I/O pin.
1	0	The state of the I/O pin is read.
1	1	The I/O pin is in an output mode. The output data latch is read.

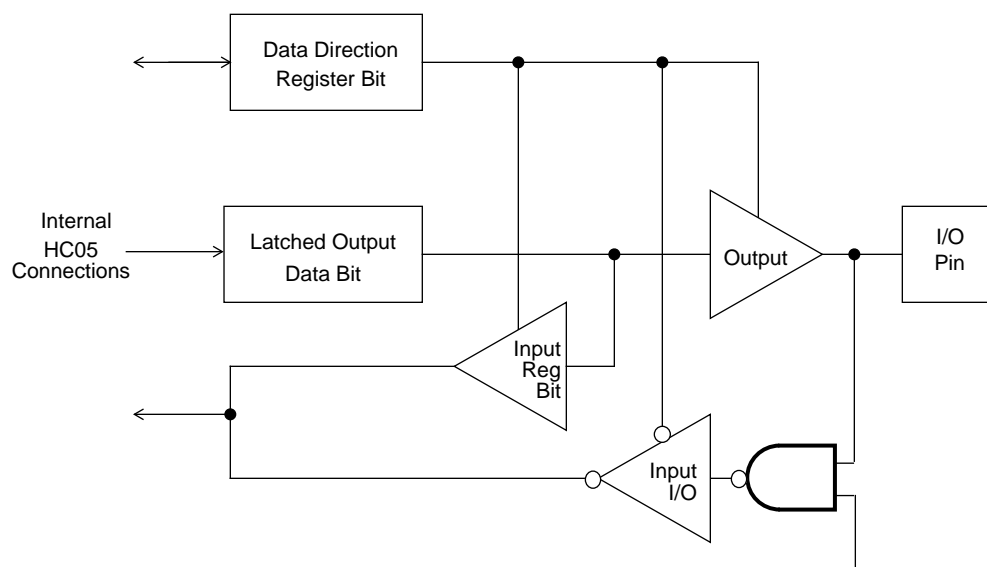


Figure 2-4. I/O Circuitry

SECTION 3

CENTRAL PROCESSING UNIT (CPU)

The MCU contains five registers as shown in Figure 3-1. The interrupt stacking order is shown in Figure 3-2.

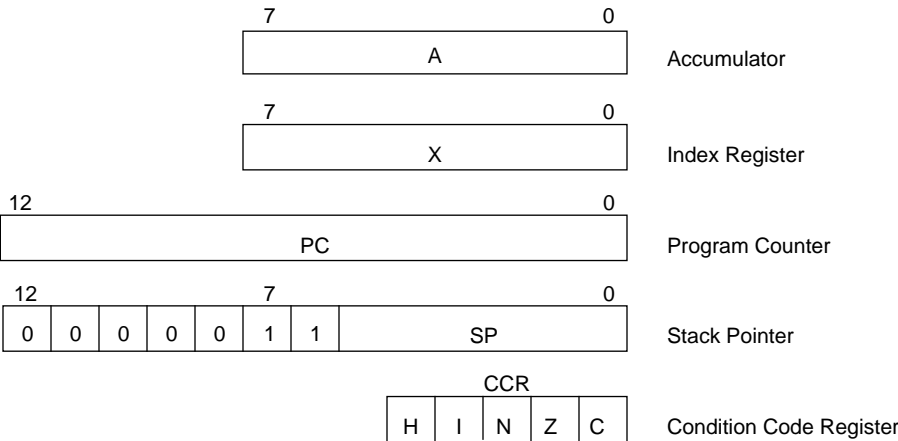


Figure 3-1. Programming Model

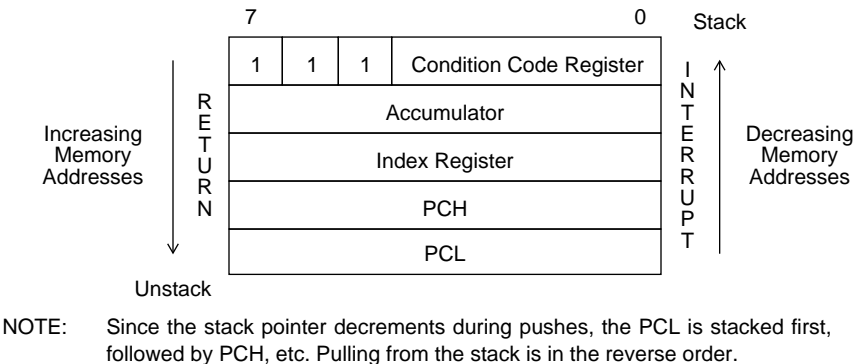
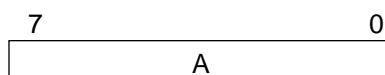


Figure 3-2. Stacking Order

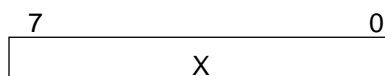
3.1 ACCUMULATOR (A)

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



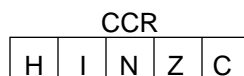
3.2 INDEX REGISTER (X)

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 CONDITION CODE REGISTER (CCR)

The CCR is a 5-bit register in which the H, N, Z, and C bits are used to indicate the results of the instruction just executed, and the I bit is used to enable interrupts. These bits can be tested individually by a program, and specific actions can be taken as a result of their state. Each bit is explained in the following paragraphs.



3.3.1 Half Carry (H)

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

3.3.2 Interrupt (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 I bit is cleared.

3.3.3 Negative (N)

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

3.3.4 Zero (Z)

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

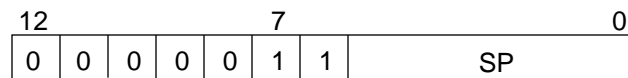
3.3.5 Carry/Borrow (C)

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 also is affected during bit test and branch instructions and during shifts and rotates.

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 then is decremented as data is pushed onto the stack and incremented as data is pulled from the stack.

When accessing memory, the seven most significant bits (MSB) are permanently set to 0000011. These seven 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.

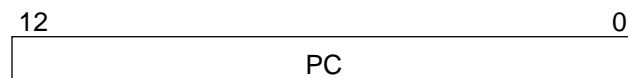


3.5 PROGRAM COUNTER (PC)

The program counter is a 13-bit register that contains the address of the next byte to be fetched.

NOTE

The HC05 CPU core is capable of addressing 16-bit locations. For this implementation, however, the addressing registers are limited to an 8-Kbyte memory map.



SECTION 4 INTERRUPTS

The MCU can be interrupted seven different ways:

1. Non-maskable Software Interrupt Instruction (SWI)
2. External Asynchronous Interrupt ($\overline{\text{IRQ}}$)
3. External Interrupt via IRQ on PC0-PC7
4. Internal 16-bit Timer Interrupt (TIMER)
5. Internal Synchronous Serial Interface (SSI) Interrupt
6. Internal Core Timer Interrupt
7. Comparator 3

4.1 CPU INTERRUPT PROCESSING

Interrupts cause the processor to save register contents on the stack and to set the interrupt mask (I bit) to prevent additional interrupts. Unlike RESET, hardware interrupts do not cause the current instruction execution to be halted, but are considered pending until the current instruction is complete.

If interrupts are not masked (I bit in the CCR is clear) and the corresponding interrupt enable bit is set, the processor will proceed with interrupt processing. Otherwise, the next instruction is fetched and executed. If an interrupt occurs, the processor completes the current instruction, stacks the current CPU register states, sets the I bit to inhibit further interrupts, and finally checks the pending hardware interrupts. If more than one interrupt is pending following the stacking operation, the interrupt with the highest vector location shown in Table 4-1 will be serviced first. The SWI is executed the same as any other instruction, regardless of the I-bit state.

When an interrupt is to be processed, the CPU fetches the address of the appropriate interrupt software service routine from the vector table at locations \$1FF4 through \$1FFF as defined in Table 4-1.

Table 4-1. Vector Address for Interrupts and Reset

Register	Flag Name	Interrupt	CPU Interrupt	Vector Address
N/A	N/A	Reset	RESET	\$1FFE-\$1FFF
N/A	N/A	Software	SWI	\$1FFC-\$1FFD
CMCSR	CMP3	CMP3, External Interrupts**	CMP3/IRQ	\$1FFA-\$1FFB
TSR	OCF, ICF, TOF	16-bit Timer Interrupts	TIMER	\$1FF8-\$1FF9
SSSR	SSIF	SSI Interrupt	SSI	\$1FF6-\$1FF7
CTCSR	TOFE, RTIE	Core Timer Interrupts	TIMER, RTI	\$1FF4-\$1FF5

** External interrupts include IRQ and PORTC sources.

The M68HC05 CPU does not support interruptible instructions. The maximum latency to the first instruction of the interrupt service routine must include the longest instruction execution time plus stacking overhead.

$$\text{Latency} = (\text{Longest instruction execution time} + 10) \times t_{\text{CYC}} \text{ secs}$$

A return to interrupt (RTI) instruction is used to signify when the interrupt software service routine is completed. The RTI instruction causes the register contents to be recovered from the stack and normal processing to resume at the next instruction that was to be executed when the interrupt took place. Figure 4-1 shows the sequence of events that occurs during interrupt processing.

4.2 RESET INTERRUPT SEQUENCE

The RESET function is not in the strictest sense an interrupt; however, it is acted upon in a similar manner as shown in Figure 4-1. A low level input on the RESET pin, or internally generated RST signal, causes the program to vector to its starting address, which is specified by the contents of memory locations \$1FFE and \$1FFF. The I bit in the condition code register is also set. The MCU is configured to a known state during this type of reset as described in **4.1 CPU INTERRUPT PROCESSING**.

4.3 SOFTWARE INTERRUPT (SWI)

The SWI is an executable instruction and a non-maskable interrupt since it is executed regardless of the state of the I bit in the CCR. If the I bit is zero (interrupts enabled), the SWI instruction executes after interrupts that were pending before the SWI was fetched, or before interrupts generated after the SWI was fetched. The interrupt service routine address is specified by the contents of memory locations \$1FFC and \$1FFD.

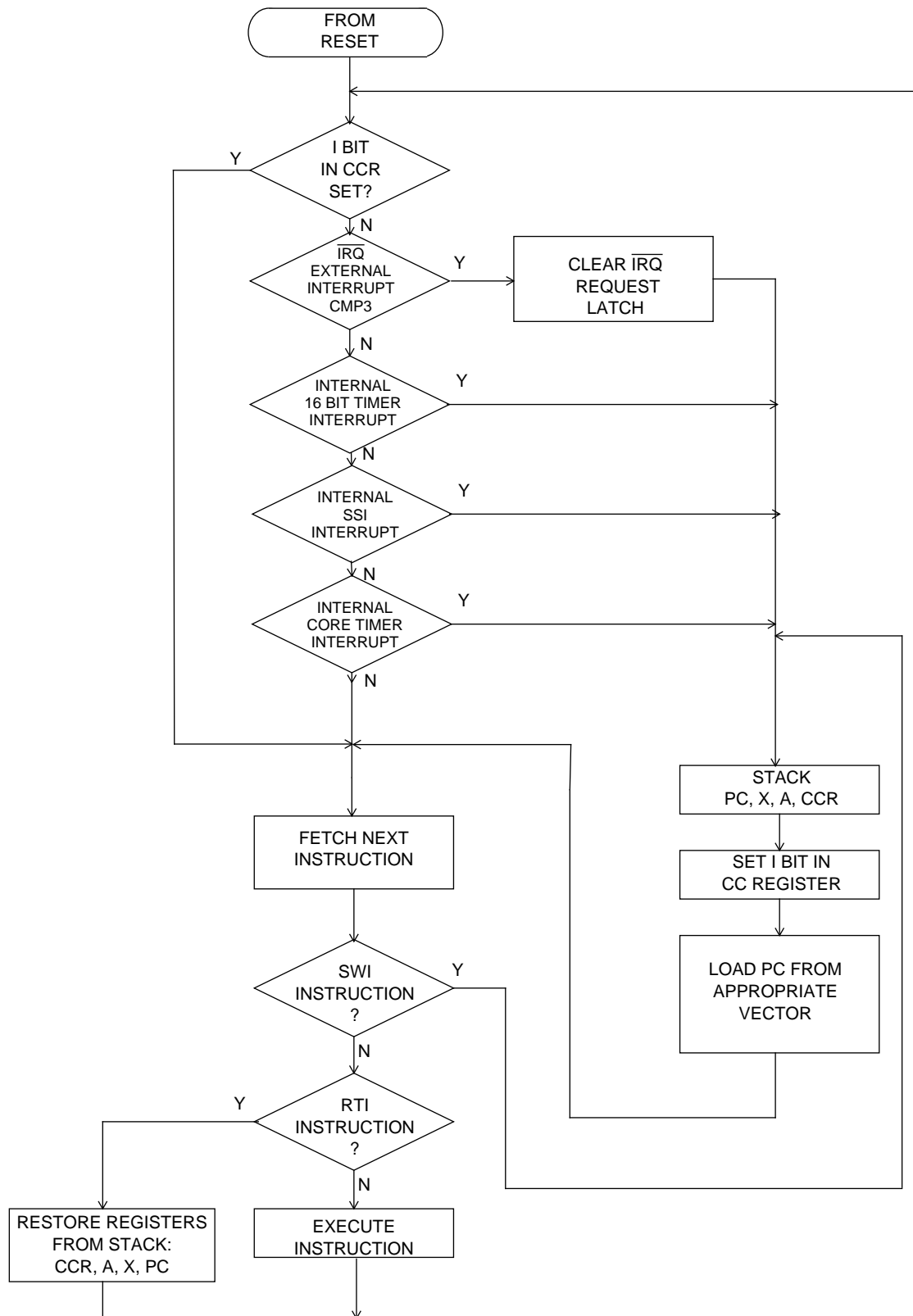


Figure 4-1. Interrupt Processing Flowchart

4.4 HARDWARE INTERRUPTS

All hardware interrupts except $\overline{\text{RESET}}$ are maskable by the I bit in the CCR. If the I bit is set, all hardware interrupts (internal and external) are disabled. Clearing the I bit enables the hardware interrupts. There are two types of hardware interrupts, which are explained in the following sections.

4.5 EXTERNAL INTERRUPT (IRQ)

The $\overline{\text{IRQ}}$ pin provides an asynchronous interrupt to the CPU. A block diagram of the IRQ function is shown in Figure 4-2.

NOTE

The BIH and BIL instructions will apply only to the level on the $\overline{\text{IRQ}}$ pin itself, and not to the output of the logic OR function with the port C IRQ interrupts. The state of the individual port C pins can be checked by reading the appropriate port C pins as inputs.

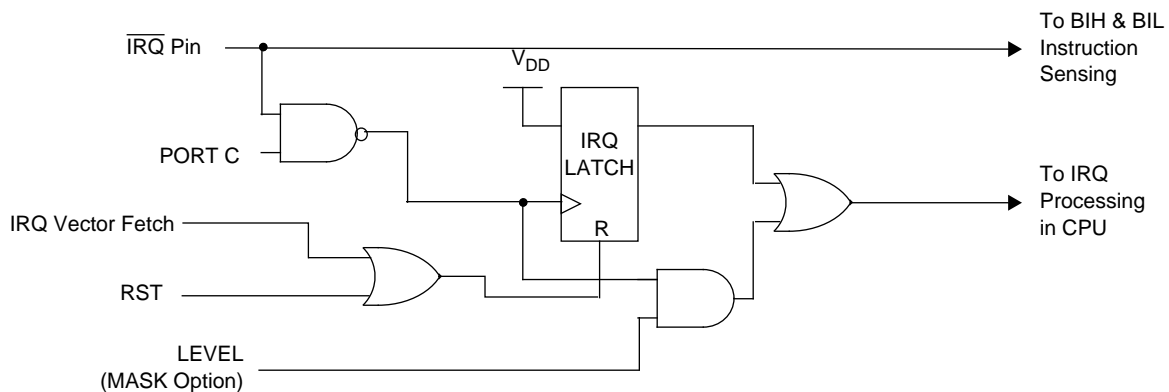


Figure 4-2. IRQ Function Block Diagram

The $\overline{\text{IRQ}}$ pin is one source of an external interrupt. All port C pins (PC0 through PC7) act as other external interrupt sources if the keyscan feature is enabled as specified by the user.

When edge sensitivity is selected for the IRQ interrupt, it is sensitive to the following cases:

1. Falling edge on the $\overline{\text{IRQ}}$ pin.
2. Falling edge on any port C pin with keyscan enabled.

When edge and level sensitivity is selected for the IRQ interrupt, it is sensitive to the following cases:

1. Low level on the $\overline{\text{IRQ}}$ pin.
2. Falling edge on the $\overline{\text{IRQ}}$ pin.
3. Falling edge or low level on any port C pin with keyscan enabled.

4.6 EXTERNAL INTERRUPT TIMING

If the interrupt mask bit (I bit) of the CCR is set, all maskable interrupts (internal and external) are disabled. Clearing the I bit enables interrupts. The interrupt request is latched immediately following the falling edge of the $\overline{\text{IRQ}}$ source. It is then synchronized internally and serviced as specified by the contents of \$1FFA and \$1FFB.

Either a level-sensitive and edge-sensitive trigger or an edge-sensitive-only trigger is available via the mask programmable option for the $\overline{\text{IRQ}}$ pin.

4.7 16-BIT TIMER INTERRUPT

Three different timer interrupt flags cause a timer interrupt whenever they are set and enabled. The interrupt flags are in the timer status register (TSR), and the enable bits are in the timer control register (TCR). Any of these interrupts vector to the same interrupt service routine, located at the address specified by the contents of memory locations \$1FF8 and \$1FF9.

4.8 SSI INTERRUPT

Two different SSI interrupt flags cause an SSI interrupt whenever they are set and enabled. The interrupt flags are in the SSI status register (SSSR), and the enable bits are in the SSI control register (SSCR). Either of these interrupts vector to the same interrupt service routine, located at the address specified by the contents of memory locations \$1FF6 and \$1FF7.

4.9 CORE TIMER INTERRUPT

This timer can create two types of interrupts. A timer overflow interrupt occurs whenever the 8-bit timer rolls over from \$FF to \$00 and the enable bit TOFE is set. A real time interrupt occurs whenever the programmed time elapses and the enable bit RTIE is set. Either of these interrupts vector to the same interrupt service routine, located at the address specified by the contents of memory locations \$1FF4 and \$1FF5.

4.10 COMPARATOR 3 INTERRUPT

Comparator 3 can create an interrupt when its output (CMP3) gets set and the enable bit CM3IE is set. The interrupt service routine is located at the address specified by the contents of memory locations \$1FFA and \$1FFB.

SECTION 5 RESETS

The MCU can be reset from four sources: one external input and three internal restart conditions. The $\overline{\text{RESET}}$ pin is an input with a Schmitt trigger as shown in Figure 5-1. All the internal peripheral modules will be reset by the internal reset signal (RST). Refer to Figure 5-2 for reset timing detail.

5.1 EXTERNAL RESET ($\overline{\text{RESET}}$)

The $\overline{\text{RESET}}$ pin is the only external source of a reset. This pin is connected to a Schmitt trigger input gate to provide an upper and lower threshold voltage separated by a minimum amount of hysteresis. This external reset occurs whenever the $\overline{\text{RESET}}$ pin is pulled below the lower threshold and remains in reset until the $\overline{\text{RESET}}$ pin rises above the upper threshold. This active low input will generate the RST signal and reset the CPU and peripherals. Termination of the external $\overline{\text{RESET}}$ input or the internal COP watchdog reset are the only reset sources that can alter the operating mode of the MCU.

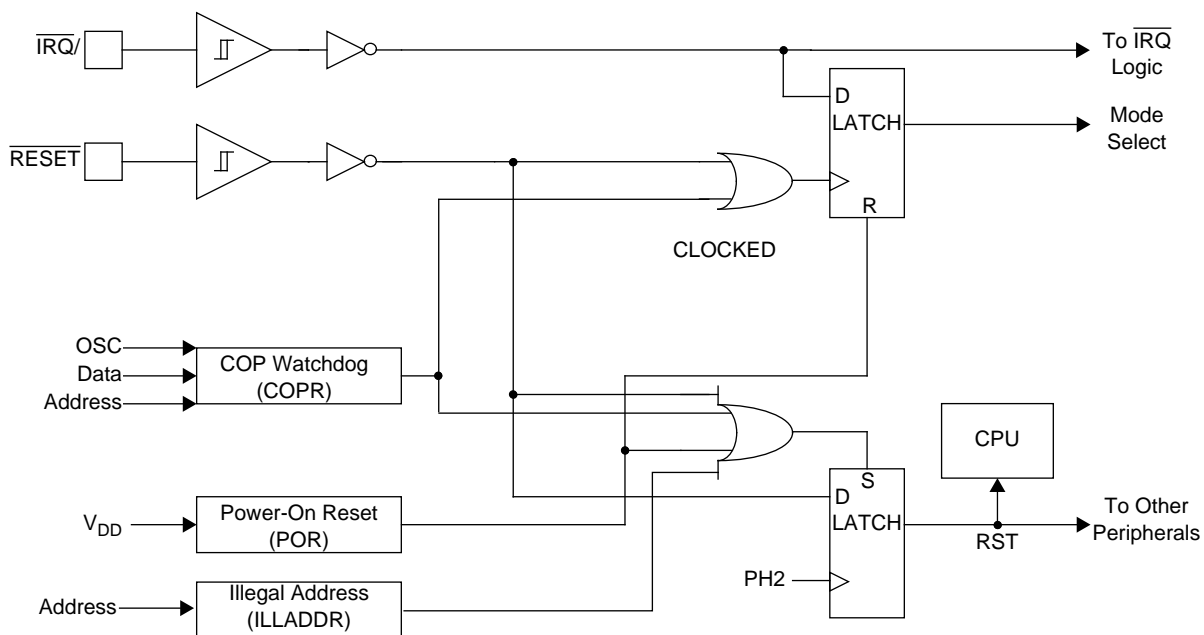
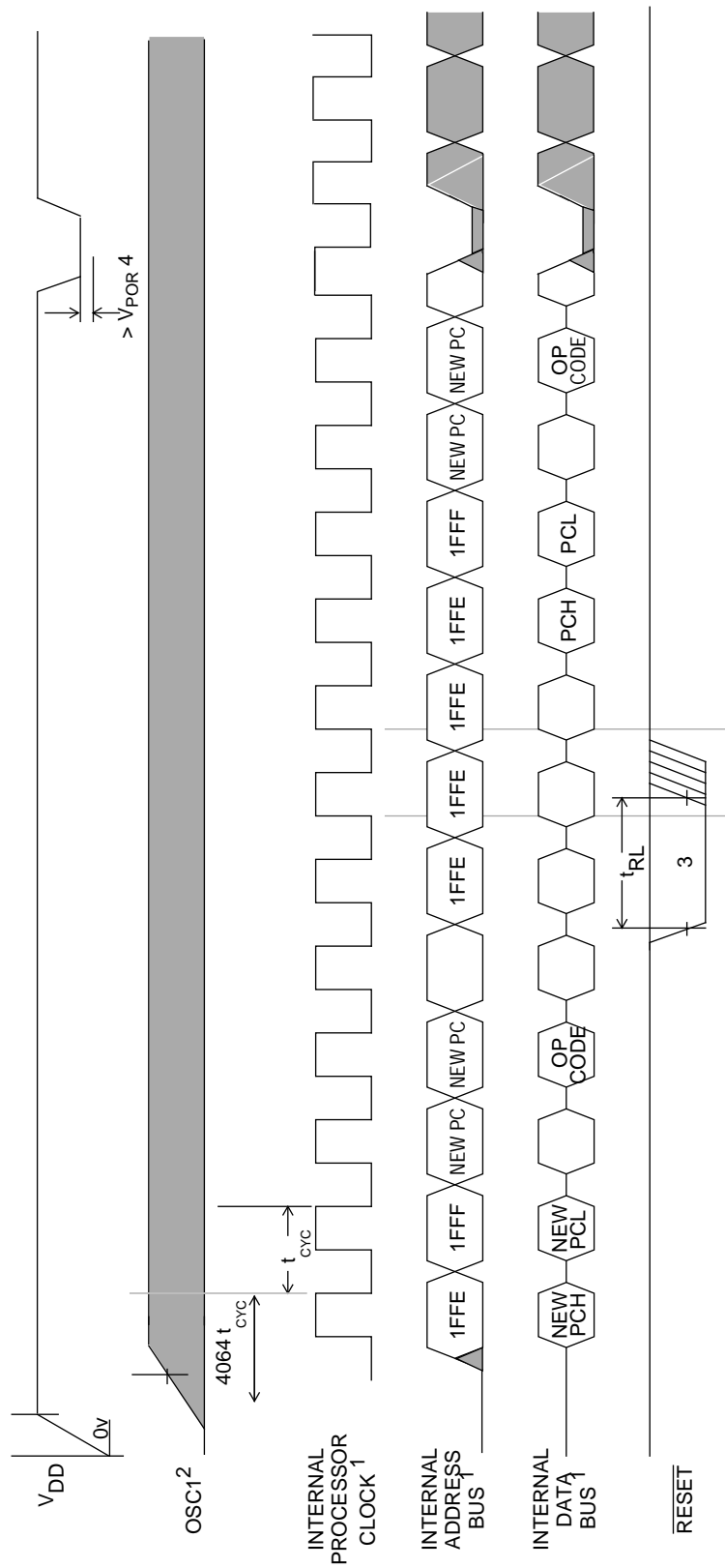


Figure 5-1. Reset Block Diagram



NOTES:

1. Internal timing signal and bus information not available externally
2. OSC1 line is not meant to represent frequency. It is only used to represent time.
3. The next rising edge of the internal processor clock following the rising edge of $\overline{\text{RESET}}$ initiates the reset sequence.
4. V_{DD} must fall to a level lower than V_{POR} to be recognized as a power-on reset.

Figure 5-2. RESET and POR Timing Diagram

NOTE

Activation of the RST signal is generally referred to as **reset** of the device, unless otherwise specified.

5.2 INTERNAL RESETS

The three internally generated resets are the initial power-on reset function, the COP watchdog timer reset, and the illegal address detector. Termination of the external RESET input or the internal COP watchdog timer are the only reset sources that can alter the operating mode of the MCU. The other internal resets do not have any effect on the mode of operation when their reset state ends.

5.2.1 Power-On Reset (POR)

The internal POR is generated on power-up to allow the clock oscillator to stabilize. The POR is strictly for power turn-on conditions and is not able to detect a drop in the power supply voltage (brown-out). There is an oscillator stabilization delay of 4064 internal processor bus clock cycles (PH2) after the oscillator becomes active.

The POR generates the RST signal that resets the CPU. If any other reset function is active at the end of this 4064-cycle delay, the RST signal remains in the reset condition until the other reset condition(s) end. During the POR, the $\overline{\text{RESET}}$ pin is forced low.

5.2.2 Computer Operating Properly Reset (COPR)

The MCU contains a watchdog timer that automatically times out if not reset (cleared) within a specific time by a program reset sequence. If the COP watchdog timer is allowed to time-out, an internal reset is generated to reset the MCU. Regardless of an internal or external RESET, the MCU comes out of a COP reset according to the standard rules of mode selection.

The COP reset function is enabled or disabled by a mask option and is verified during production testing.

Resetting the COP

Writing a 0 to the COPF bit prevents a COP reset. This action resets the counter and begins the time-out period again. The COPF bit is bit 0 of address \$1FF0. A read of address \$1FF0 returns user data programmed at that location.

COP During Wait Mode

The COP continues to operate normally during wait mode. The software should pull the device out of wait mode periodically and reset the COP by writing to the COPF bit to prevent a COP reset.

COP During Stop Mode

When the stop enable mask option is selected, stop mode disables the oscillator circuit and thereby turns the clock off for the entire device. When STOP is executed, the COP counter will hold its current state. If a reset is used to exit stop mode, the COP counter is reset and held until 4064 POR cycles are completed at which time counting will begin. If an external IRQ is used to exit STOP mode, the COP counter does not wait for the completion of the 4064 POR cycles but it does count these cycles. It is recommended, therefore, that the COP is fed before executing the STOP instruction.

COP Watchdog Timer Considerations

The COP watchdog timer is active in all modes of operation if enabled by a mask option. If the COP watchdog timer is selected by a mask option, any execution of the STOP instruction (either intentional or inadvertent due to the CPU being disturbed) causes the oscillator to halt and prevent the COP watchdog timer from timing out. If the COP watchdog timer is selected by a mask option, the COP resets the MCU when it times out. Therefore, it is recommended that the COP watchdog is **disabled** for a system that must have intentional uses of the wait mode for periods longer than the COP time-out period.

The recommended interactions and considerations for the COP watchdog timer, STOP instruction, and WAIT instruction are summarized in Table 5-1.

Table 5-1. COP Watchdog Timer Recommendations

IF the following conditions exist:	THEN the COP Watchdog Timer should be as follows:
WAIT Time	
WAIT Time less than COP Time-Out	Enable or Disable COP by Mask Option
WAIT Time More than COP Time-Out	Disable COP by Mask Option
Any length WAIT Time	Disable COP by Mask Option

COP Register

The COP register is shared with the MSB of an unimplemented user interrupt vector as shown in Figure 5-3. Reading this location returns whatever user data has been programmed at this location. Writing a 0 to the COPR bit in this location clears the COP watchdog timer.

ADDR	REGISTER	R/W	7	6	5	4	3	2	1	0
\$1FF0	UNIMPLEMENTED VECTOR & COP WATCHDOG TIMER	R	x	x	x	x	x	x	x	x
		W								COPR

UNIMPLEMENTED

Figure 5-3. COP Watchdog Timer Location


5.2.3 Illegal Address

An illegal address reset is generated when the CPU attempts to fetch an instruction from either unimplemented address space (\$0130 to \$0AFF) or I/O address space (\$0000 to \$002F).

SECTION 6 OPERATING MODES

The MCU has two modes of operation: single-chip mode and self-check mode, with two oscillator options. Table 6-1 shows the conditions required to go into each mode.

Table 6-1. Operating Mode Conditions

	RESET	IRQ	PB1	PD5	MODE
$V_{TST} = 2 \times V_{DD}$		$V_{SS}-V_{DD}$	$V_{SS}-V_{DD}$	$V_{SS}-V_{DD}$	Single-chip
		V_{TST}	V_{DD}	V_{SS}	Self-check

6.1 SINGLE-CHIP MODE

In single-chip mode, the address and data buses are not available externally, but there are three 8-bit I/O ports and one 7-bit I/O port. This mode allows the MCU to function as a self-contained microcontroller, with maximum use of the pins for on-chip peripheral functions. All address and data activity occurs within the MCU. Single-chip mode is entered on the rising edge of RESET if the IRQ pin is within normal operating range.

6.2 LOW-POWER MODES

The following paragraphs describe the low-power modes.

6.2.1 Stop Mode

The STOP instruction places the MCU in its lowest power consumption mode. In stop mode, the internal oscillator is turned off, halting all internal processing, including timer operation.

During the stop mode, the CTCSR (\$08) bits are altered to remove any pending timer interrupt request and to disable any further timer interrupts. 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 an external interrupt or reset.

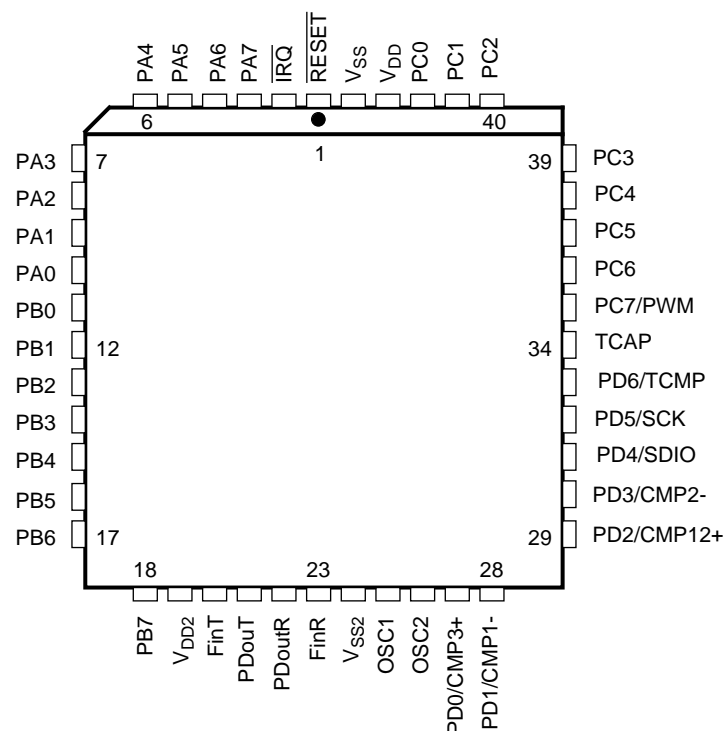


Figure 6-1. Single-Chip Mode Pinout of the MC68HC05CT4

6.2.2 Stop Recovery

The processor can be brought out of the stop mode only by an external interrupt or $\overline{\text{RESET}}$. See Figure 6-2.

6.2.3 Wait Mode

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 the core timer, the oscillator, and any enabled module remain active. Any interrupt or reset will cause the MCU to exit the wait mode. The user must shut off subsystems to reduce power consumption. WAIT current specifications assumes CPU operation only and does not include current consumption by any other subsystems.

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 timer may be enabled to allow a periodic exit from the wait mode.

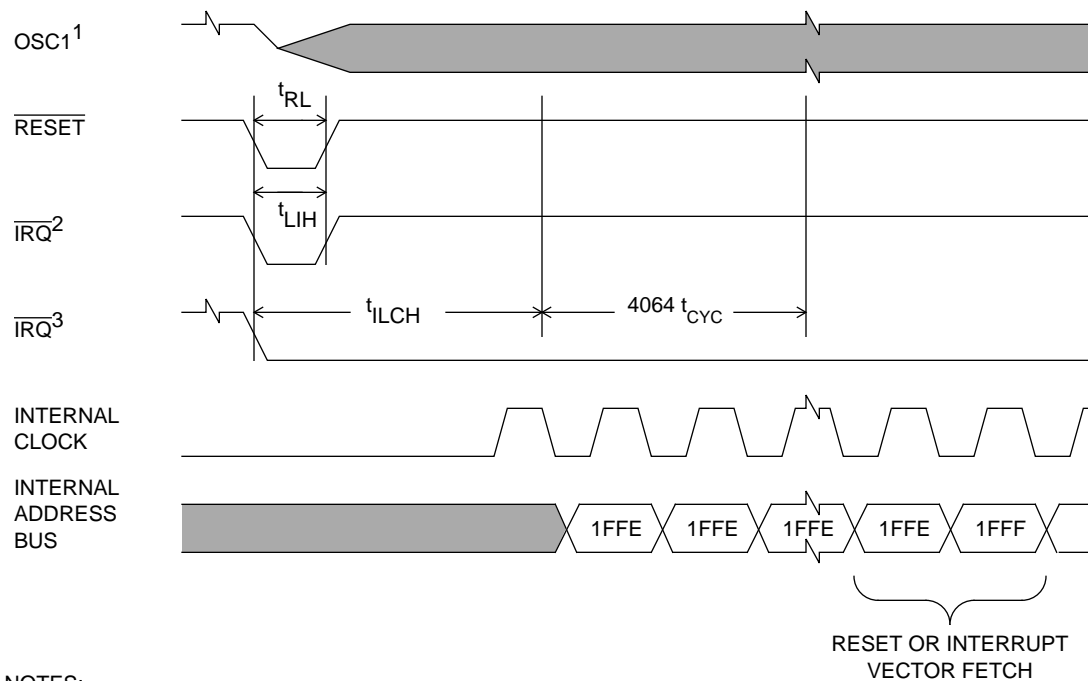
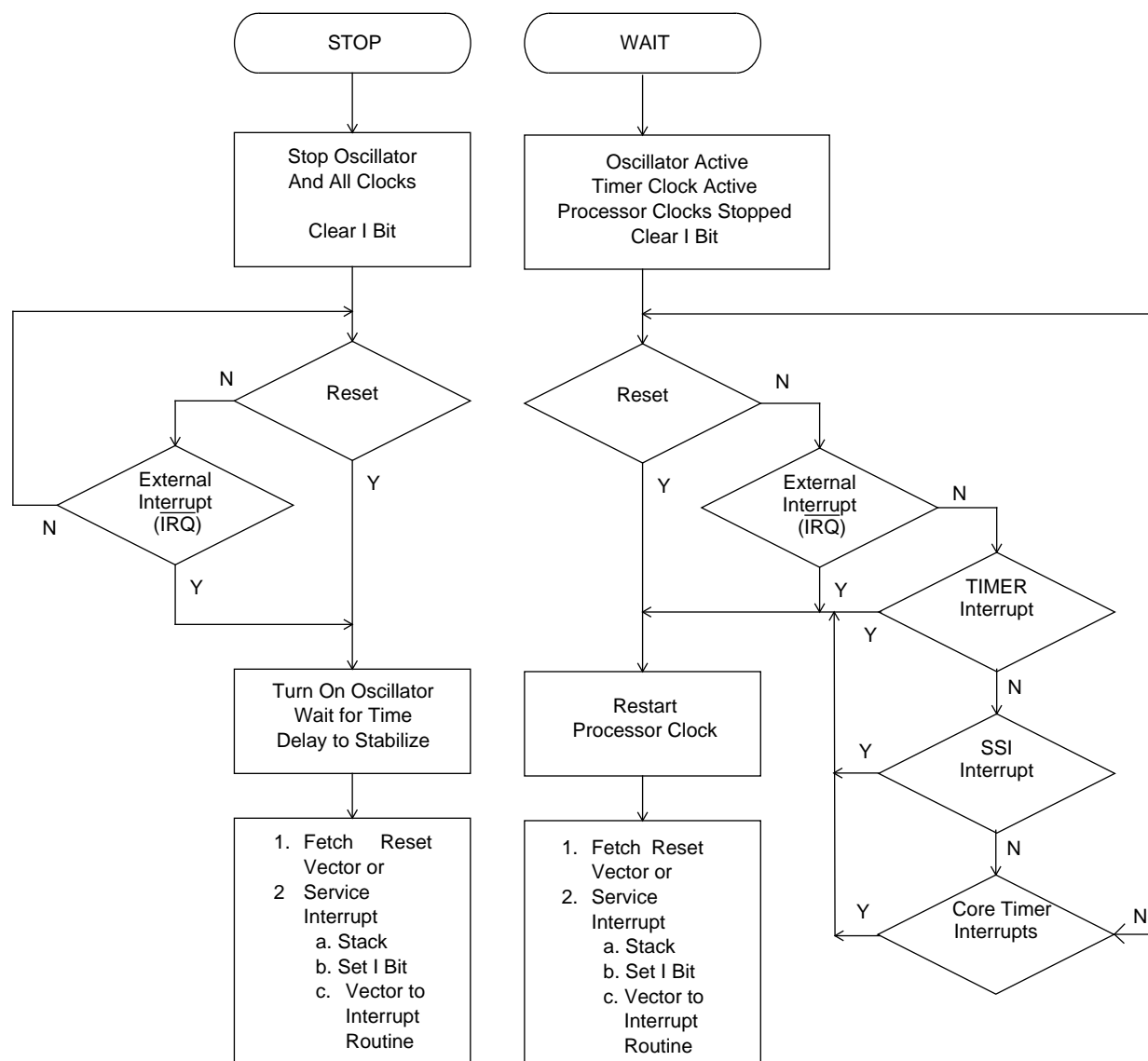


Figure 6-2. Stop Recovery Timing Diagram

6.2.4 Low-Power Wait

When the WAIT mode is entered by executing the WAIT instruction, the oscillator divider changes from a divide-by-5 to a divide-by-40 (additional divide-by-8) to lower the wait current. As a result, this gives a CPU clock rate of 256 KHz if the oscillator is running with a 10.24 MHz crystal. The oscillator divide-by-5 or divide-by-40 option is also controlled by the speed bit located in the miscellaneous control register (\$21). See **SECTION 13 MISCELLANEOUS REGISTER**. When returning from wait mode via an interrupt, the OSC rate prior to entering wait mode is restored.

**Figure 6-3. STOP/WAIT Flowchart**

SECTION 7

16-BIT TIMER

The timer consists of a 16-bit, free-running 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 waveform. Pulse widths can vary from several microseconds to many seconds.

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. 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 registers of a specific timer function to ensure that an interrupt does not occur.

7.1 COUNTER REGISTER - \$18:\$19 AND \$1A:\$1B

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 any time 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 also must be read to complete the sequence.

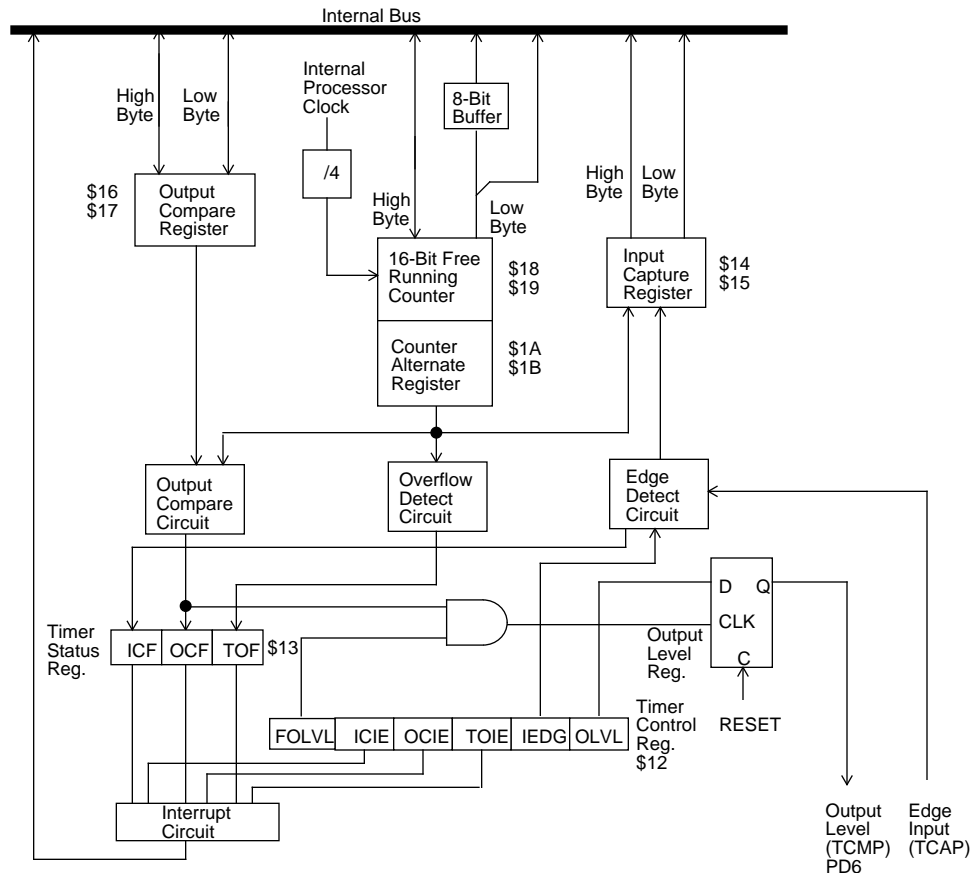


Figure 7-1. 16-Bit Timer Block Diagram

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 any time 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 a read-only register but only when the timer is enabled. During a power-on reset, the counter is also preset to \$FFFC and begins running only after the TON bit in the TIMER control register is set. Because the free-running counter is 16 bits preceded by a fixed divided-by-four 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 roll-over occurs by setting its interrupt enable bit (TOIE).

NOTE

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

7.2 OUTPUT COMPARE REGISTER — \$16:\$17

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 two bytes of the output compare register can be used as storage locations.

The output compare register contents are continually compared with the contents of the free-running counter. If a match is found, the corresponding output compare flag (OCF) bit is set and the corresponding output level (OLVL) bit is clocked to an output level register. The output compare register values and the output level bit should be changed after each successful comparison to establish a new elapsed time-out. 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. The output level (OLVL) bit is clocked to the output level register regardless of whether the output compare flag (OCF) is set or clear.

7.3 INPUT CAPTURE REGISTER — \$14:\$15

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 that 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 is 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 MSB (\$14), the counter transfer is inhibited until the LSB (\$15) is also read. This characteristic causes the time 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.

7.4 TIMER CONTROL REGISTER (TCR) — \$12

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

		7	6	5	4	3	2	1	0
\$12	RD	ICIE	OCIE	TOIE	0	0	TON	IEDG	OLVL
	WR								
	RESET:	0	0	0	0	0	0	0	0

Figure 7-2. Timer Control Register

ICIE — Input Capture Interrupt Enable

- 1 = Interrupt enabled
- 0 = Interrupt disabled

OCIE — Output Compare Interrupt Enable

- 1 = Interrupt enabled
- 0 = Interrupt disabled

TOIE — Timer Overflow Interrupt Enable

- 1 = Interrupt enabled
- 0 = Interrupt disabled

TON — Timer On

When disabled, the timer is initialized to the reset condition.

- 1 = Timer enabled
- 0 = Timer disabled

IEDG — Input Edge

Value of input edge determines which level transition on the TCAP pin will trigger a free-running counter transfer to the input capture register. Reset clears this bit.

- 1 = Positive edge
- 0 = Negative edge

OLVL — Output Level

Value of output level is clocked into the output level register by the next successful output compare and will appear on the TCMP pin

- 1 = High output
- 0 = Low output

7.5 TIMER STATUS REGISTER (TSR) — \$13

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

		7	6	5	4	3	2	1	0
\$13	RD	ICF	OCF	TOF	0	0	0	0	0
	WR								
	RESET:	0	0	0	0	0	0	0	0

Figure 7-3. Timer Status Register

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

Reset clears this bit.

OCF — Output Compare 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

Reset clears this bit.

OF — 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

Reset clears this bit.

Bits 0-4 — Not used

Always read 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 MSB 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 any time without affecting the timer overflow flag in the timer status register.

7.6 TIMER DURING WAIT MODE

The CPU clock halts during the wait mode, but the timer remains active if turned on prior to entering wait mode. If interrupts are enabled, a timer interrupt will cause the processor to exit the wait mode.

7.7 TIMER DURING STOP MODE

In the stop mode, the timer 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 the timer is on and 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, however, 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, no input capture flag or data remains, even if a valid input capture edge occurred.

7.8 TIMER POWER SUPPLY SOURCE

The timer's power is supplied by V_{DD} and V_{SS} . V_{DD2} and V_{SS2} will not be needed since this module is not susceptible to supply noise.

SECTION 8

SYNCHRONOUS SERIAL INTERFACE (SSI)

The synchronous serial interface (SSI) is a two-wire master/slave system including serial clock (SCK) and serial data input/output (SDIO). Data is transferred eight bits at a time. A software programmable option determines whether the SSI transfers data MSB or LSB first and an interrupt may be generated at the completion of each transfer. When operating as a master device, the serial clock speed is selectable from a choice of four rates.

Transmission in master mode is initiated by a write to the SSI data register (SDR). A transfer cannot be initiated in slave mode; the external master initiates the transfer. The programmer must choose between master or slave mode before the SSI is enabled. It is up to the programmer to ensure that only one master exists in the system at any one time. All devices in the system must operate with the same clock polarity and data rates. Slaves should always be disabled before the master is disabled.

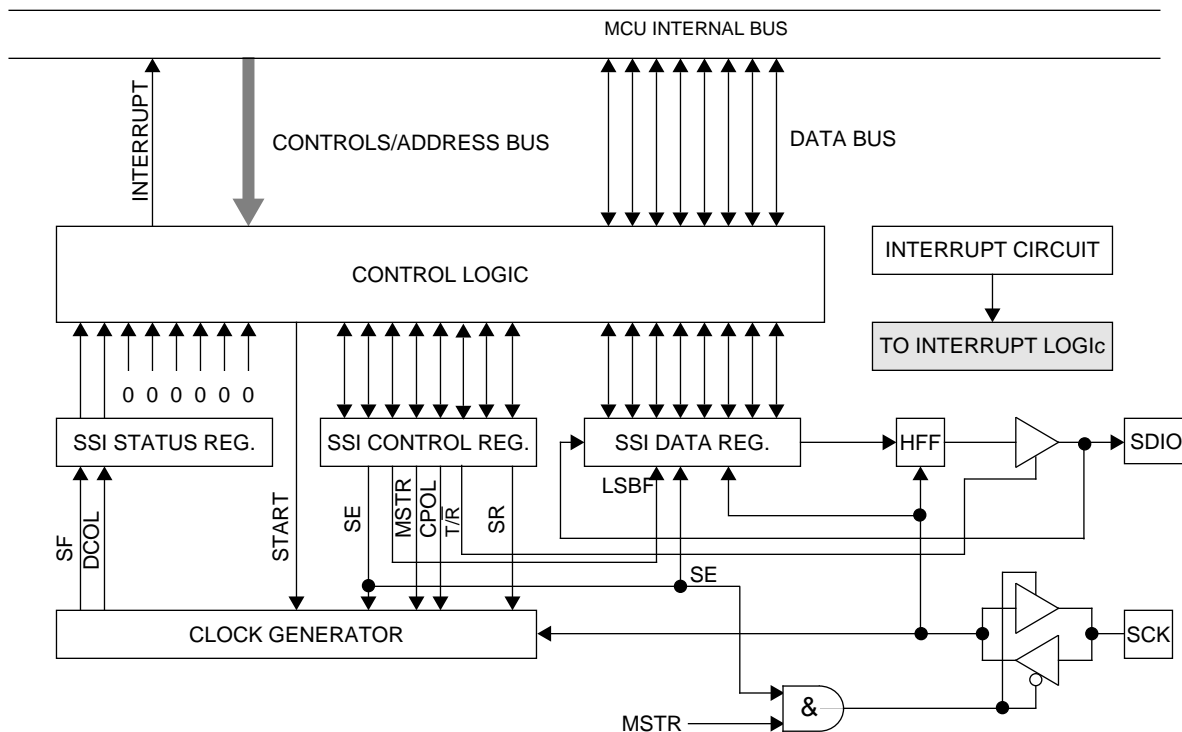


Figure 8-1. SSI Block Diagram

8.1 SIGNAL FORMAT

The SSI is comprised of two main I/O signals that interface with port D serial clock and serial data.

8.1.1 Serial Clock (SCK)

When SE = 0, this pin is a port D bit 5 pin, which follows the port D DDR assignment.

In master mode (MSTR = 1), the serial clock (SCK) pin is an output with four selectable frequencies. This pin will be high (CPOL = 1) or low (CPOL = 0) between transmissions.

In slave mode (MSTR = 0), the SCK pin is an input and the clock must be supplied by an external master with a maximum frequency of $f_{OP}/2$. There is no minimum SCK frequency. This pin should be driven high (CPOL = 1) or low (CPOL = 0) between transmissions by the external master and must be stable before the SSI is first enabled (SE = 1).

Data is always captured at the serial data in/out (SDIO) pin on the rising edge of SCK.

Data is always shifted out and presented at the serial data in/out (SDIO) pin on the falling edge of SCK.

8.1.2 Serial Data In/Out (SDIO)

Prior to enabling the SSI (SE = 0), the serial data in/out (SDIO) pin is a port D bit 4 pin, which follows the port D DDR assignment. When the SSI is enabled (SE = 1) the SDIO pin becomes a high-impedance input pin if the $\overline{T/R}$ bit is low, or idles high if the $\overline{T/R}$ bit is high.

The data can be sent or received in either MSB first format (LSBF = 0) or LSB first format (LSBF = 1).

If (CPOL = 1), the first falling edge of SCK will shift the first data bit out to the SDIO pin. Subsequent falling edges of SCK will shift the remaining data bits out.

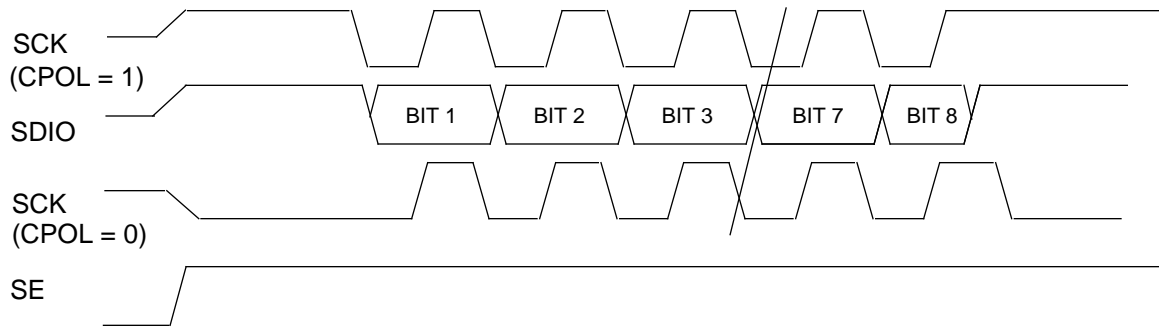
If (CPOL = 0), the first data bit will be driven out to the SDIO pin before the first rising edge of SCK. Subsequent falling edges of SCK will shift the remaining data bits out.

When receiving data in master mode, the $\overline{T/R}$ bit must be low and data must be written to the data register to initiate clock generation.

When transmitting data in master mode, the $\overline{T/R}$ bit must be high.

When receiving data in slave mode, $\overline{T/R}$ bit must be low and the clock and data must be supplied by external device.

When transmitting data in slave mode, $\overline{T/R}$ bit must be high, and data must be written to the data register before the SSI is enabled to ensure that proper data is transferred.

**Figure 8-2. Serial I/O Port Timing**

8.2 SSI REGISTERS

The SSI has three registers: control, status, and data.

8.2.1 SSI Control Register (SCR) \$1E

This register is located at address \$001E and contains seven bits. A reset clears all of these bits, except bit 3 which is set. Writes to this register during a transfer should be avoided, with the exception of clearing the SE bit to disable the SSI.

In addition, the clock polarity, rate, data format and master/slave selection should not be changed while the SSI is enabled (SE = 1) or being enabled. Always disable the SSI first, by clearing the SE bit, before altering these control bits within the SSI control register (SCR).

	7	6	5	4	3	2	1	0
\$1E	RD	SIE	SE	LSBF	MSTR	CPOL	T/R	SR1
	WR							SR0
RESET:	0	0	0	0	1	0	0	0

Figure 8-3. SSI Control Register

SIE — SSI Interrupt Enable

This bit determines whether an interrupt request should be generated when a transfer is complete.

When set, an interrupt request is made if the CPU is in the run or wait mode of operation and status flag bit SF is set.

When cleared, no interrupt requests are made by the SSI.

SE — SSI Enable

When set, this bit enables the SSI, makes PD5 the SCK pin, and makes PD4 the SDIO pin.

When SE is cleared, any transmission in progress is aborted, the bit counter is reset, and pins SCK and SDIO revert to being PD5 and PD4.

LSBF — Least Significant Bit (LSB) First

When set, data is sent and received in a least significant bit (LSB) first format.

When cleared, data is sent and received in a most significant bit (MSB) first format.

MSTR — Master Mode

When set, this bit configures the SSI to the master mode. This means that the transmission is initiated by a write to the data register and the SCK pin becomes an output providing a synchronous data clock at a rate determined by the SR bits.

When cleared, this bit configures the SSI to the slave mode and aborts any transmission in progress. Transfers are initiated by an external master, which should supply the clock to the SCK pin.

CPOL — Clock Polarity

The clock polarity bit controls the state of the SCK pin between transmissions.

When this bit is set, pin SCK is high between transmissions.

When this bit is cleared, pin SCK is low between transmissions.

In both cases the data is latched on the rising edge of SCK for serial input and is valid on the rising edge of SCK for serial output. A reset sets this bit.

NOTE

If the SSI is used as a slave, the SCK input pin must be active before enabling the SSI. For example, if CPOL = 0, SCK must be low; if CPOL = 1, SCK must be high.

T/R — Transmit/Receive

This bit must be set to allow data to be driven on the SDIO pin (transmitting). It must be cleared to disable the SDIO drivers when receiving data. It is cleared by a reset.

SR1, SR0 — SSI Rate

These bits determine the frequency of SCK when in master mode (MSTR = 1). They have no effect in slave mode (MSTR = 0).

Table 8-1. SSI Rates

SR1:SR0	SCK Rates (Hz) at f_{osc} Frequency (10.24 MHz)
00	32 KHz
01	64 KHz
10	128 KHz
11	256 KHz

8.2.2 SSI Status Register (SSR) \$1C

This register is located at address \$001C and contains two bits. Reset clears both of these bits.

	7	6	5	4	3	2	1	0
\$1C	RD	SF	DCOL	0	0	0	0	0
	WR			0	0	0	0	0
RESET:	0	0	0	0	0	0	0	0

Figure 8-4. SSI Status Register

SF — SSI Flag

This bit is set upon occurrence of the last rising clock edge and indicates that a data transfer has taken place. If MSTR = 0 and SIE = 0, this bit has no effect on any further transmissions and can be ignored without problem. However, to write the data register, or if SIE = 1 to clear the interrupt, the SF flag must be clear. If MSTR = 1, the SF flag must be cleared between transfers. The SF flag can be cleared three different ways: (1) by reading the SSR with SF set, followed by a read or write of the serial data register, (2) by a system reset, or, (3) by disabling the SSI. If the SF flag is cleared before the last edge of the next byte, it will be set again.

DCOL — Data Collision

This is a read-only status bit, which indicates that an invalid access to the data register was made. An invalid access can be one of the following conditions:

- An access of the SDR register in the middle of a transfer (after the first falling edge of SCK and before SF is set)
- An access of the SDR register made before an access of the SSR register (after SF is set)

DCOL is cleared by reading the status register with SF set followed by a read or write of the data register. A reset also clears this bit.

8.2.3 SSI Data Register (SDR) \$1D

This register is located at address \$001D and is both the transmit and receive data register. This system is not double buffered, but any writes to this register during transfers are masked and will not destroy the previous contents. The SDR can be read at any time but if a transfer is in progress the results may be ambiguous. The contents of this register could be altered whenever the CPOL bit is altered. This register should be written to only upon completion of a transfer, after the SF flag has been cleared. Otherwise, the new data will not be stored.

For an SSI configured as a master, to initiate a transfer, the data register write must occur after the SSI is enabled.

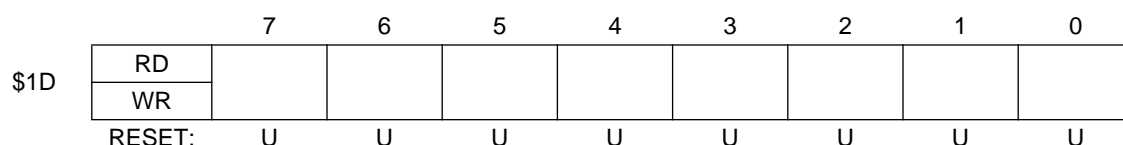


Figure 8-5. SSI Data Register

8.3 OPERATION DURING WAIT MODE

The CPU clock halts during the wait mode, but the SSI remains active. If interrupts are enabled, an SSI interrupt will cause the processor to exit the wait mode.

8.4 OPERATION DURING STOP MODE

In the stop mode, the SSI halts operation. The SDIO and SCK pins will maintain their states.

If the SSI is nearing completion of a transfer when the stop mode is entered, it might be possible for the SSI to generate an interrupt request and thus cause the processor immediately to exit the stop mode. To prevent this occurrence, the programmer should ensure that all transfers are complete before entering the stop mode.

If the SSI is configured to slave mode, further care should be taken in entering stop mode. The SCK pin will still accept a clock from an external master, allowing potentially unwanted transfers to take place and power consumption to be increased. The SSI will not generate interrupt requests in this situation but, on exiting stop mode through some other means, the SF flag may be found to be set and an interrupt request will be generated if SIE is also set at this point.

To avoid these potential problems, it is safer to disable the SSI completely ($SE = 0$) before entering stop mode.

The synchronous serial interface (SSI) is a two-wire master/slave system including serial clock (SCK) and serial data input/output (SDIO). When operating as a master device, the serial clock speed is selectable between four rates.

8.5 SSI POWER SUPPLY SOURCE

The power supplied to the SSI is V_{DD} and V_{SS} , thus keeping the V_{DD2} and V_{SS2} free for the noise susceptible modules.

SECTION 9 CORE TIMER

The core timer for this device is a 12-stage multifunctional ripple counter. Features include timer overflow, power-on reset (POR), real time interrupt (RTI), and COP watchdog timer.

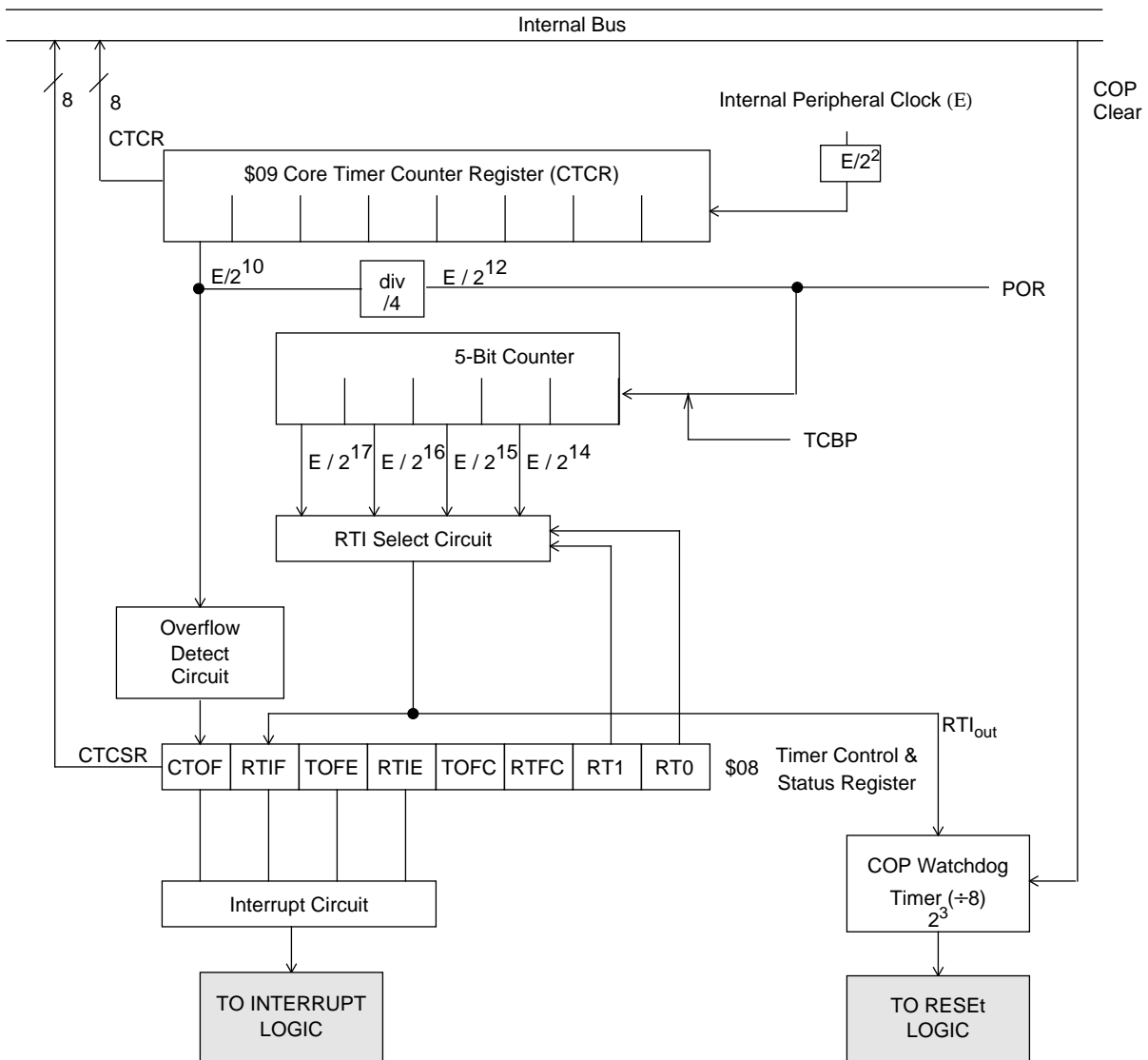


Figure 9-1. Core Timer Block Diagram

As seen in Figure 9-1, the internal peripheral clock is divided by four, and then drives an 8-bit ripple counter. The value of this 8-bit ripple counter can be read by the CPU at any time by accessing the core timer counter register (CTCR) at address \$09. A timer overflow function is implemented on the last stage of this counter, giving a possible interrupt rate of the internal peripheral clock(E)/1024. This point is then followed by three more stages, with the resulting clock (E/16384) driving the real-time interrupt circuit (RTI). The RTI circuit consists of three divider stages with a one-of-four selector. The output of the RTI circuit is further divided by eight to drive the mask optional COP watchdog timer circuit. The RTI rate selector bits and the RTI and CTOF enable bits and flags are located in the timer control and status register at location \$08.

9.1 CORE TIMER CTRL AND STATUS REGISTER (CTCSR) \$08

The CTCSR contains the timer interrupt flag, the timer interrupt enable bits, and the real time interrupt rate select bits. Figure 9-2 shows the value of each bit in the CTCSR when coming out of reset.

	7	6	5	4	3	2	1	0
\$08	RD	CTOF	RTIF	TOFE	RTIE	0	0	RT1
	WR					TOFC	RTFC	
RESET:	0	0	0	0	0	0	0	1

Figure 9-2. Core Timer Control and Status Register

CTOF — Core Timer Overflow

CTOF is a read-only status bit set when the 8-bit ripple counter rolls over from \$FF to \$00. Clearing the CTOF is done by writing a logic one to TOFC. Writing to this bit has no effect. Reset clears CTOF.

RTIF — Real Time Interrupt Flag

The Real-Time Interrupt circuit consists of a three-stage divider and a one-of-four selector. The clock frequency that drives the RTI circuit is $E/2^{14}$ (or $E/16384$) with three additional divider stages giving a maximum interrupt period of 64 milliseconds at a bus rate of 2.048 MHz. RTIF is a clearable, read-only status bit and is set when the output of the chosen (one-of-four selection) stage goes active. Clearing the RTIF is done by writing a logic one to RTFC. Writing has no effect on this bit. Reset clears RTIF.

TOFE — Timer Overflow Enable

When this bit is set, a CPU interrupt request is generated when the CTOF bit is set. Reset clears this bit.

RTIE — Real-Time Interrupt Enable

When this bit is set, a CPU interrupt request is generated when the RTIF bit is set. Reset clears this bit.

TOFC — Timer Overflow Flag Clear

When a logic one is written to this bit, CTOF is cleared. Writing a logic zero has no effect on the CTOF bit. This bit always reads as zero.

RTFC — Real-Time Interrupt Flag Clear

When a logic one is written to this bit, RTIF is cleared. Writing a logic zero has no effect on the RTIF bit. This bit always reads as zero.

RT1:RT0 — Real-Time Interrupt Rate Select

These two bits select one of four taps from the real time interrupt circuit. See Table 9-1. Reset sets these two bits, which selects the lowest periodic rate and gives the maximum time in which to alter these bits if necessary. Care should be taken when altering RT0 and RT1 if the time-out period is imminent or uncertain. If the selected tap is modified during a cycle in which the counter is switching, an RTIF could be missed or an additional one could be generated. To avoid problems, the COP should be cleared before changing RTI taps.

Table 9-1. RTI and COP Rates at 2.048 MHz

RTI RATE 2.048 MHz		RT1:RT0	MINIMUM COP RATES 2.048 MHz	
8 ms	$2^{14}/E$	00	$(2^{17}-2^{14})/E$	56 ms
16 ms	$2^{15}/E$	01	$(2^{18}-2^{15})/E$	112 ms
32 ms	$2^{16}/E$	10	$(2^{19}-2^{16})/E$	224 ms
64 ms	$2^{17}/E$	11	$(2^{20}-2^{17})/E$	448 ms

9.2 CORE TIMER COUNTER REGISTER (CTCR) \$09

The timer counter register is a read-only register that contains the current value of the 8-bit ripple counter at the beginning of the timer chain. This counter is clocked by the CPU clock ($E/4$) and can be used for various functions including a software input capture. Extended time periods can be attained using the TOF function to increment a temporary RAM storage location thereby simulating a 16-bit (or more) counter.

	7	6	5	4	3	2	1	0	
\$09	RD	D7	D6	D5	D4	D3	D2	D1	D0
	WR								
RESET:	0	0	0	0	0	0	0	0	0

Figure 9-3. Timer Counter Register

The power-on cycle clears the entire counter chain and begins clocking the counter. After 4064 cycles, the power-on reset circuit is released, which again clears the counter chain and allows the device to come out of reset. At this point, if $\overline{\text{RESET}}$ is not asserted, the timer starts counting up from zero and normal device operation begins. When $\overline{\text{RESET}}$ is asserted any time during operation (other than POR), the counter chain is cleared.

9.3 COMPUTER OPERATING PROPERLY (COP) RESET

The COP watchdog timer function is implemented on this device by using the output of the RTI circuit and further dividing it by eight. The minimum COP reset rates are listed in Figure 9-1. If the COP circuit times out, an internal reset is generated and the normal reset vector is fetched. Preventing a COP time out, or clearing the COP, is accomplished by writing a logic zero to bit 0 of address \$1FF0. When the COP is cleared, only the final divide-by-eight stage (output of the RTI) is cleared.

If the COP watchdog timer is allowed to time out, an internal reset is generated to reset the MCU. In addition, the $\overline{\text{RESET}}$ pin is pulled low for a minimum of one E-clock cycle for emulation purposes.

The COP remains enabled after execution of the WAIT instruction and all associated operations apply. If the STOP instruction is disabled, execution of STOP instruction causes the CPU to execute a NOP instruction. In addition, the COP is prohibited from being held in RESET. This prevents a device lock-up condition.

This COP's objective is to make it impossible for this device to become stuck or locked-up and to be sure the COP is able to rescue the part from any situation where it might entrap itself in abnormal or unintended behavior. This function is a mask option.

9.4 TIMER DURING WAIT MODE

The CPU clock halts during the wait mode, but the timer remains active. If interrupts are enabled, a timer interrupt will cause the processor to exit the wait mode. The COP is always enabled while in user mode.

9.5 CORE TIMER POWER SUPPLY SOURCE

The core timer is supplied by V_{DD} and V_{SS} . V_{DD2} and V_{SS2} are not needed here because this module is not susceptible to supply noise.

SECTION 10

DUAL PHASE-LOCKED LOOP (PLL)

This dual PLL is similar to that of the MC145162 60-MHz universal programmable dual PLL frequency synthesizer. It is especially designed for CT-1 cordless phone applications.

It features fully programmable 16-bit receive, 16-bit transmit, and 12-bit reference ripple down counters. It also has two independent phase detectors for transmit and receive loops.

The FINTX and FINRX signals are input to the PLL transmit and receive counters, respectively. They are typically driven by the loop VCO and AC-coupled. The minimum input signal level is 200 mVpp @ 60.0 MHz.

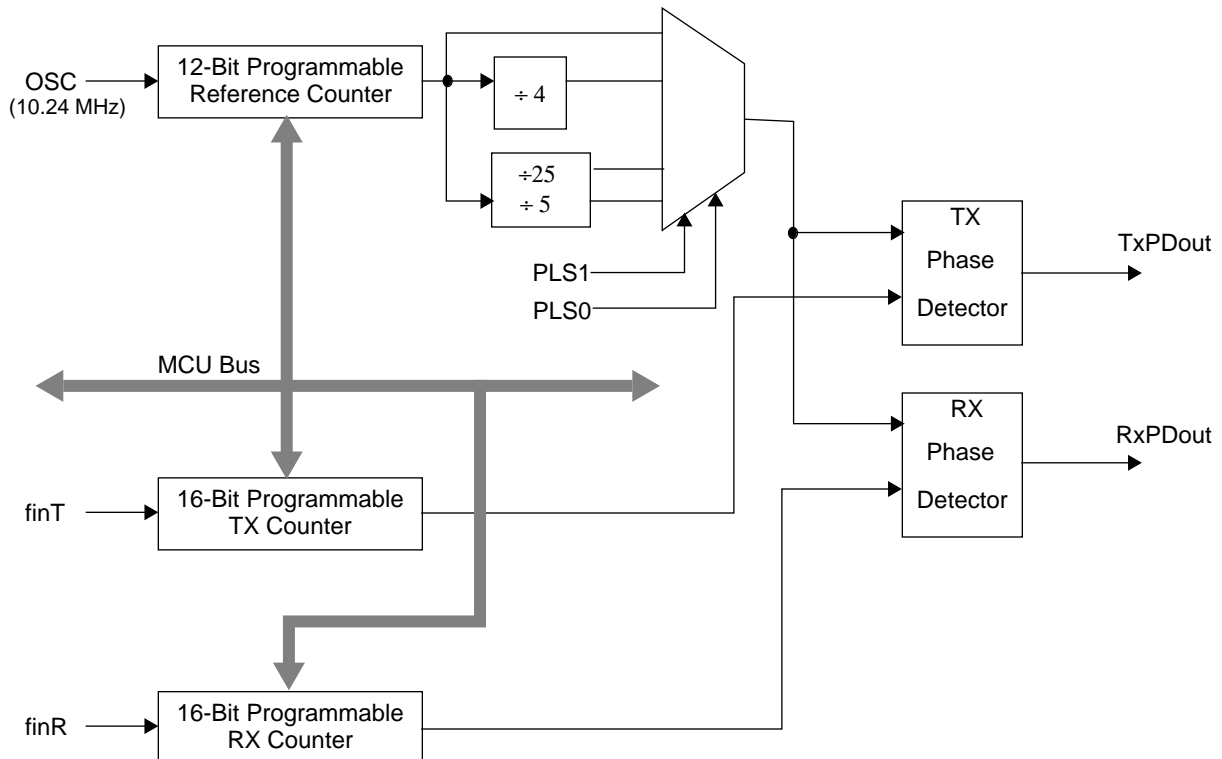


Figure 10-1. Dual PLL Block Diagram

10.1 REGISTERS

The PLL has one 12-bit programmable counter, two 16-bit programmable counters, and one control register.

10.1.1 Dual Control Register (PLLCR) \$0A

The PLLCR contains bits that affect the operation of the PLL.

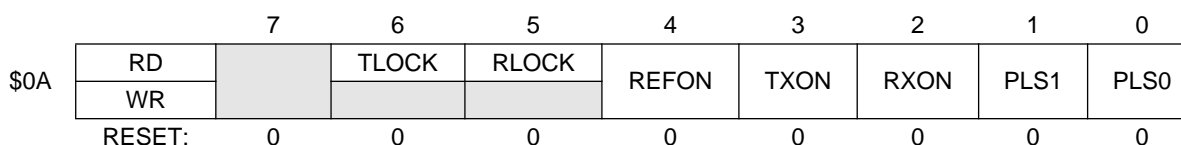


Figure 10-2. Dual PLL Control Register

PLS1, PLS0 — PLL Reference Counter Select

These bits select between the PLL reference counter MUX outputs. This output signal then drives the phase detectors.

Table 10-1. PLL Reference Counter Select

PLS1:PLS0	PLL REFERENCE COUNTER OUTPUT
00	÷ 12-Bit Counter
01	÷ 4 After 12-Bit Counter
10	÷ 5 After 12-Bit Counter
11	÷ 25 After 12-Bit Counter

RXON — RX Counter Enable

When set, this bit enables the PLL receive counter.

When clear, it stops the receive counter in a reset state to save power. It also shuts off the associated phase detector and holds it in three-state. It is recommended that the receive counter be initialized before enabling the receive counter.

TXON — TX Counter Enable

When set, this bit enables the PLL transmit counter.

When clear, it stops the counter in a reset state to save power. It also shuts off the associated phase detector and holds it in three-state. It is recommended that the transmit counter be initialized before it is enabled.

REFON — Reference Counter Enable

When set, this bit enables the PLL reference counter.

When clear, it stops the reference counter in a reset state to save power. It is recommended that the reference counter be initialized before it is enabled.

RLOCK — Receive Lock Detect

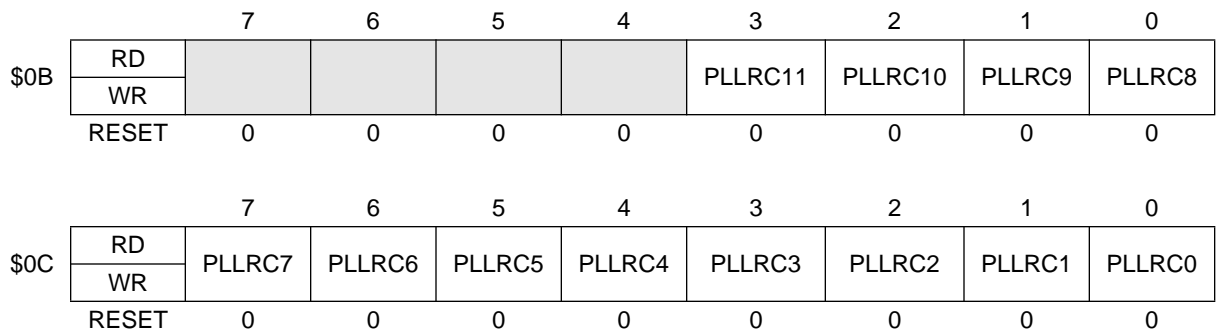
This bit is read only and is not latched. When set, this bit is a real-time indication of an active correction pulse from the phase detect in progress. When clear, no correction is made and the output from the phase detect is three-stated. Using multiple reads at known intervals will allow the user to filter and judge a LOCK condition.

TLOCK — Transmit Lock Detect

This bit is read only and is not latched. When set, this bit is a real-time indication of an active correction pulse from the phase detect in progress. When clear, no correction is made and the output from the phase detect is three-stated. Using multiple reads at known intervals will allow the user to filter and judge a LOCK condition.

10.1.2 12-Bit Reference Counter Modulus Register (PLLRC)\$0B:\$0C

This 2-byte register holds the count for the 12-bit reference counter. The reference counter is shut off and held in reset when the REFON bit is cleared. For proper operation, this register must not be loaded with a value less than \$000F.

**Figure 10-3. 12-Bit Reference Counter (PLLRC)****NOTE**

Bit 4 to bit 7 of \$0B is not used but they are physically there. The user may use these four bits for scratch memory.

10.1.3 16-Bit Transmit Counter Modulus Register (PLLTX) \$0D:\$0E

This 2-byte register holds the count for the 16-bit transmit counter. The transmit counter is shut off and held in reset when the TXON bit is cleared. For proper operation, this register must not be loaded with a value less than \$000F.

		7	6	5	4	3	2	1	0
\$0D	RD	PLLTX15	PLLTX14	PLLTX13	PLLTX12	PLLTX11	PLLTX10	PLLTX9	PLLTX8
	WR								
	RESET	0	0	0	0	0	0	0	0

		7	6	5	4	3	2	1	0
\$0E	RD	PLLTX7	PLLTX6	PLLTX5	PLLTX4	PLLTX3	PLLTX2	PLLTX1	PLLTX0
	WR								
	RESET	0	0	0	0	0	0	0	0

Figure 10-4. 16-Bit Transmit Counter (PLLTX)

10.1.4 16-Bit Receive Counter Modulus Register (PLLRX) \$0F:\$10

This 2-byte register holds the count for the 16-bit receive counter. The receive counter is shut off and held in reset when the RXON bit is cleared. For proper operation, this register must not be loaded with a value less than \$000F.

		7	6	5	4	3	2	1	0
\$0F	RD	PLLRX15	PLLRX14	PLLRX13	PLLRX12	PLLRX11	PLLRX10	PLLRX9	PLLRX8
	WR								
	RESET	0	0	0	0	0	0	0	0

		7	6	5	4	3	2	1	0
\$10	RD	PLLRX7	PLLRX6	PLLRX5	PLLRX4	PLLRX3	PLLRX2	PLLRX1	PLLRX0
	WR								
	RESET	0	0	0	0	0	0	0	0

Figure 10-5. 16-Bit Receive Counter (PLLRX)

The modulus registers mentioned above have specific read/write logic. When the user updates the contents of the modulus registers, the MSB must be written first and is temporarily stored into a temp buffer. This inhibits the transfer of data from this level to the next level, which is the modulus register. When the LSB is written, all 12 or 16 bits (data from the temp buffer and the MCU internal bus) are transferred to the modulus register simultaneously. This prevents the loading of bad data from the modulus register. A read of the data registers is the reverse of a write operation. A read of the LSB buffers the MSB. A subsequent read of the MSB reflects this buffered value. Since only one temporary buffer exists, two sequential MSB writes of different registers will result in only the last data value stored in the temporary buffer. The first value will be lost.

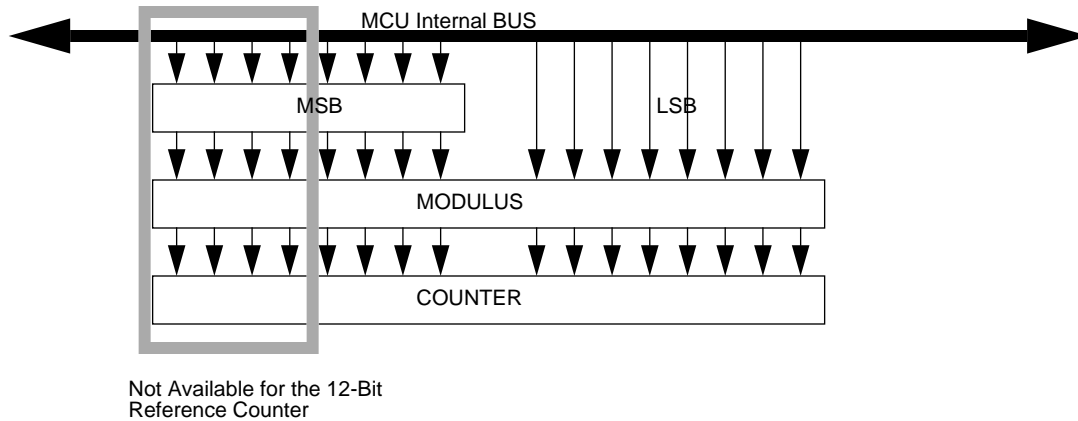


Figure 10-6. Counter Structure Block Diagram

10.2 PLL POWER SUPPLY SOURCE

The PLL is supplied by V_{DD} , V_{DD2} , V_{SS} , and V_{SS2} . V_{DD} and V_{SS} are reserved for the digital circuitry, while V_{DD2} and V_{SS2} are reserved for the analog circuitry such as the phase detect and the amplifiers which are sensitive to supply noise.

SECTION 11

PULSE WIDTH MODULATOR

The pulse width modulator (PWM) system has one 6-bit channel to enable the correct pulse output. The PWM has a fixed frequency of $E/64$, where E is the internal bus frequency. For a PWM output frequency of 32 KHz E must be 2.048 MHz. This corresponds to a 10.24 MHz crystal with the divide-by-five crystal option.

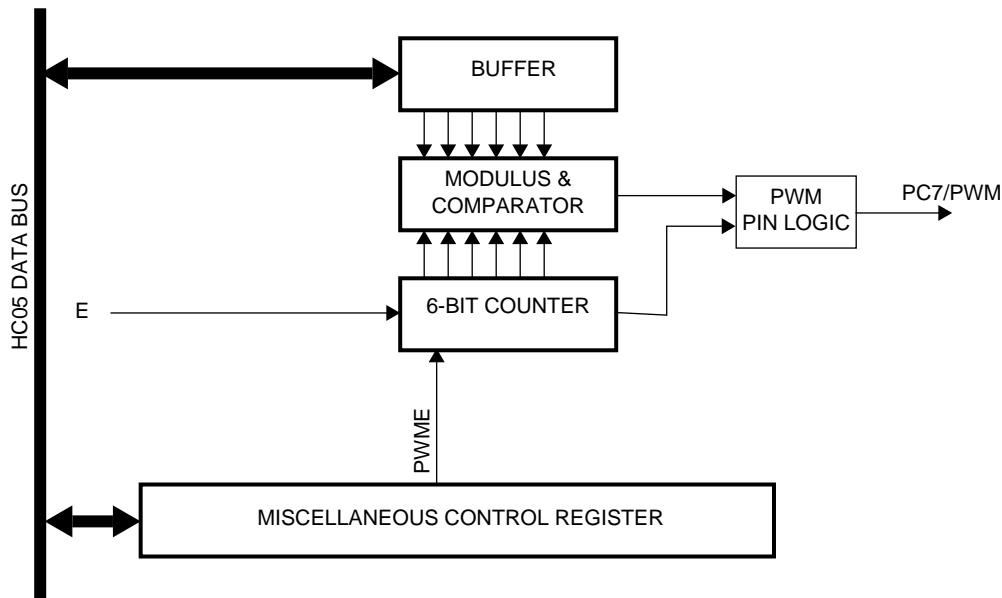


Figure 11-1. PWM Block Diagram

11.1 FUNCTIONAL DESCRIPTION

A \$00 in the PWM data register yields an off output (0%), but a \$3F yields a duty of 63/64 (98.4%).

When not in use, the PWM system can be shut off to save power by clearing the PWME bit in MISCR.

Writes to the PWM data register can be performed at any time without affecting the current PWM output signal. Updates on the PWM output occur at the end of the PWM period (Ex64). At this time, the new value is loaded into the PWM Data

register. If a write to the registers is performed while the PWM is disabled, data is transferred directly to the PWM register. A read of the data register reads the active count in progress, not the buffered value.

After the PWM is enabled (PWM=1) the PWM output remains low for one E cycle. This allows synchronization and will not occur again until the next time the PWM is enabled.

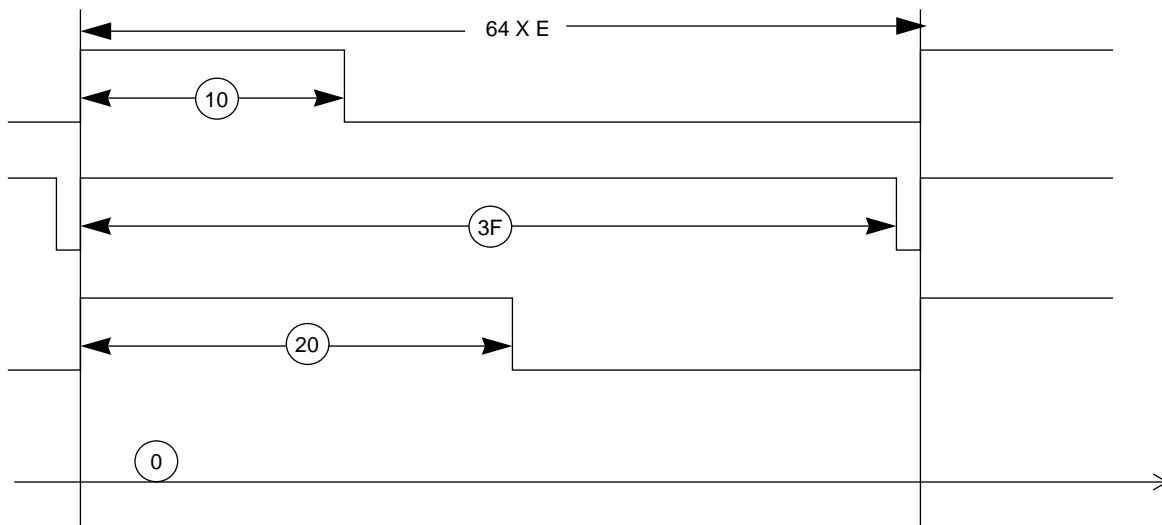


Figure 11-2. PWM Waveforms

11.2 REGISTER

One PWM data register is associated with the PWM system. This register can be written to and read at any time. Writes to the PWM data register are buffered and are not transferred to the active register until the end of the PWM cycle during which the write was executed.

Upon RESET the user should write to the data register prior to enabling the PWM system (for example, prior to setting PWME). This avoids an erroneous duty cycle from being driven.

PWME — PWM Enable

When set, this bit enables the PWM subsystem. Its main function is to allow the user to save power when not using the PWM. When enabled, the clocks are active to the module. When clear, this bit shuts off the clocks to the module and relinquishes control of PC7 to the port C logic (the PC7 pull-up option should not be selected if the PWM is used). Reset clears this bit. This bit is located in the miscellaneous control register found in **SECTION 12 COMPARATORS**.

11.3 PWM DATA REGISTER (PWMDR) \$20

The PWM system has an 8-bit data register that holds the duty cycle for the PWM output; however, only six of these bits are used. The PWM data register is located at \$20. This register is unaffected by reset.

		7	6	5	4	3	2	1	0
\$20	RD	0	0						
	WR								
	RESET:	U	U	U	U	U	U	U	U

Figure 11-3. PWM Data Register

11.4 PWM DURING WAIT MODE

The PWM continues normal operation during wait mode. To decrease power consumption during WAIT, it is recommended that the PWME bit in the miscellaneous control register be cleared if the PWM D/A converter is not being used.

11.5 PWM DURING STOP MODE

In stop mode the oscillator is stopped causing the PWM to cease function. Any signal in process is halted in whatever phase the signal happens to be in. It is recommended that the PWM be disabled before executing the STOP instruction.

11.6 PWM DURING RESET

Upon RESET the PWME bit is cleared. In effect, this disables the PWM system and configures the PC7/PWM pin as a high impedance port C input pin. The user should write to the data register prior to enabling the PWM system (for instance, prior to setting PWME). This avoids an erroneous duty cycle from being driven.

11.7 PWM POWER SUPPLY SOURCE

The PWM is supplied by V_{DD} and V_{SS} , due to the noise generated by the digital circuitry, since this module is not affected by supply noise.

SECTION 12 COMPARATORS

Port D shares its pins with CMP3+/PD0, CMP1-/PD1 and CMP12+/PD2, CMP2-/PD3 of the comparators. The output of the comparators is a status bit internal to the MCU. This circuitry is used for the comparison of analog signals, with a digital output. The comparator circuitry may be powered up by setting the CEN1, CEN2, and the CEN3 bits in the comparator control/status register. This register is located at \$0022 and is cleared by reset. The state of the comparator output bit upon $\overline{\text{RESET}}$ is logic zero. For further electrical information see **SECTION 15 ELECTRICAL SPECIFICATIONS**.

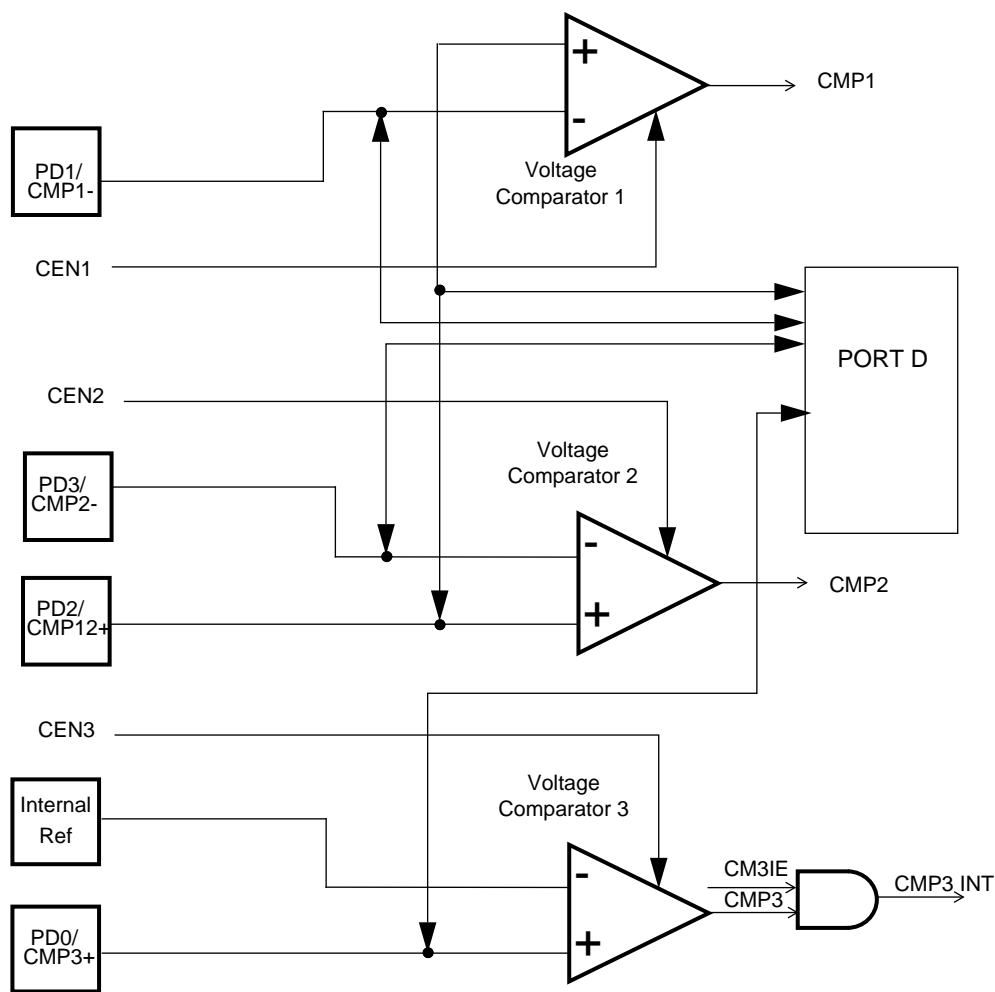


Figure 12-1. Comparator Block Diagram

12.1 COMPARATOR CONTROL/STATUS REGISTER

		7	6	5	4	3	2	1	0
\$22	RD	CMP3	CMP2	CMP1	CM3IE	0	CEN3	CEN2	CEN1
	WR								
	RESET:	0	0	0	0	0	0	0	0

Figure 12-2. Comparator Control/Status Register (CMPCSR) \$22**CMP3 — Comparator 3 Output**

This bit is implemented in two ways: as a real time comparator output or as a latched interrupt source. The state of CM3IE determines which implementation is selected.

CM3IE = 0 (Non-Latched Mode) — This status bit is set when the voltage at CMP3+ (PD0) is larger than that of CMP3- (internal voltage reference $\sim 2 V_{DD} / 3$). Otherwise it is cleared.

CM3IE = 1 (Latched Mode) - This latched status bit is set when the voltage at CMP3+ (PD0) is larger than that of CMP3- (internal voltage reference $\sim 2 V_{DD} / 3$). An interrupt is then initiated (\$1FFA-\$1FFB). It is cleared by only two means:

1. Clearing CM3IE while the voltage at CMP3+ is less than CMP3-.
2. By an external reset.

CMP2 — Comparator 2 Output

This status bit, is set when the voltage at CMP12+ is larger than that of CMP2-, otherwise, it is cleared. Reset clears this bit as the comparator is disabled.

CMP1 — Comparator 1 Output

This status bit is set when the voltage at CMP12+ is larger than that of CMP1-, otherwise, it is cleared. Reset clears this bit as the comparator is disabled.

CM3IE — Comparator 3 Interrupt Enable

This control bit, when set, allows CMP3 to generate an interrupt when it goes high.

CEN3 — Comparator 3 Enable

This control bit, when set, powers up the voltage comparator 3 on PD0. The user must set this bit to allow any functionality of the comparator. After enabling the comparator, the user should delay for t_{CEN} before reading the state of the output or enabling the interrupt. Reset clears this bit.

PD0 remains under control of its DDR bit to enable the user to drive CMP3+ to a desired level. When cleared, the comparator is disabled and consumes negligible power.

CEN2 — Comparator2 Enable

This control bit, when set, powers up the voltage comparator 2 on PD2 and PD3. The user must set this bit in order to allow any functionality of the comparator. After enabling the comparator, the user should delay for t_{CEN} before reading the state of the output. RESET clears this bit.

PD2 and PD3 remains under control of their DDR bits to enable the user to drive CMP2- and CMP12+ to desired levels. When cleared, the comparator is disabled and consumes negligible power.

CEN1 — Comparator1 Enable

This control bit, when set, powers up voltage comparator 1 on PD1 and PD2. The user must set this bit to allow any functionality of the comparator. After enabling the comparator, the user should delay for t_{CEN} before reading the state of the output. Reset clears this bit.

PD1 and PD2 remain under control of their DDR bits to enable the user to drive CMP1- and CMP12+ to desired levels. When cleared, the comparator is disabled and consumes negligible power.

12.2 READING COMPARATOR OUTPUTS

The state of each comparator output bit is internally readable from the CMPCSR.

12.3 COMPARATOR DURING WAIT

The comparator operates normally in WAIT. If the user wishes to save power during the wait mode, the CEN1, CEN2, and CEN3 bits should be cleared before the WAIT instruction.

12.4 COMPARATOR DURING STOP

STOP does not affect the comparator circuit. If the user needs to save power, the CEN1, CEN2, and CEN3 bits should be cleared before the STOP instruction.

12.5 COMPARATOR POWER SUPPLY SOURCE

The comparators are supplied by V_{DD2} and V_{SS2} . These power lines are reserved for noise-susceptible circuitry such as the circuitry found in the comparators.

SECTION 13

MISCELLANEOUS REGISTER

The miscellaneous register exists to hold various system control bits. It is located at address \$0021.

13.1 MISCELLANEOUS CONTROL REGISTER

	7	6	5	4	3	2	1	0
\$21	RD							
	WR	0	0	0	0	SPEED	COE	PWME
	RESET:	0	0	0	0	0	0	0

Figure 13-1. Miscellaneous Control Register (MISCR) \$21

SPEED — CPU Speed Select

This control bit, when set, selects an additional divide-by-eight after the already divide-by-five from the oscillator clock to the internal bus clock. This makes the internal CPU clock a divide-by-40 from the oscillator input.

When clear, the internal bus clock is a divide-by-five from the oscillator input. RESET clears this bit.

COE — 16-Bit Timer Output Compare Enable

This control bit, when set, enables the 16-bit output compare function to come out on PD6/TCMP. When clear, PD6 returns to a general-purpose I/O pin.

PWME — PWM Enable

This control bit, when set, enables the PWM function to come out on PC7/PWM. When clear, PC7 returns to a general-purpose I/O pin.

13.2 MISCELLANEOUS REGISTER POWER SUPPLY SOURCE

The miscellaneous control register is supplied by V_{DD} and V_{SS} . V_{DD2} and V_{SS2} are reserved for noise-susceptible circuitry.

SECTION 14

INSTRUCTION SET

This section describes the addressing modes and instruction types.

14.1 ADDRESSING MODES

The CPU uses eight addressing modes for flexibility in accessing data. The addressing modes define the manner in which the CPU finds the data required to execute an instruction. The eight addressing modes are the following:

- Inherent
- Immediate
- Direct
- Extended
- Indexed, no offset
- Indexed, 8-bit offset
- Indexed, 16-bit offset
- Relative

14.1.1 Inherent

Inherent instructions are those that have no operand, such as return from interrupt (RTI) and stop (STOP). Some of the inherent instructions act on data in the CPU registers, such as set carry flag (SEC) and increment accumulator (INCA). Inherent instructions require no memory address and are one byte long.

14.1.2 Immediate

Immediate instructions are those that contain a value to be used in an operation with the value in the accumulator or index register. Immediate instructions require no memory address and are two bytes long. The opcode is the first byte, and the immediate data value is the second byte.

14.1.3 Direct

Direct instructions can access any of the first 256 memory addresses with two bytes. The first byte is the opcode, and the second is the low byte of the operand address. In direct addressing, the CPU automatically uses \$00 as the high byte of the operand address. BRSET and BRCLR are three-byte instructions that use direct addressing to access the operand and relative addressing to specify a branch destination.

14.1.4 Extended

Extended instructions use only three bytes to access any address in memory. The first byte is the opcode; the second and third bytes are the high and low bytes of the operand address.

When using the Motorola assembler, the programmer does not need to specify whether an instruction is direct or extended. The assembler automatically selects the shortest form of the instruction.

14.1.5 Indexed, No Offset

Indexed instructions with no offset are one-byte instructions that can access data with variable addresses within the first 256 memory locations. The index register contains the low byte of the conditional address of the operand. The CPU automatically uses \$00 as the high byte, so these instructions can address locations \$0000–\$00FF.

Indexed, no offset instructions are often used to move a pointer through a table or to hold the address of a frequently used RAM or I/O location.

14.1.6 Indexed, 8-Bit Offset

Indexed, 8-bit offset instructions are two-byte instructions that can access data with variable addresses within the first 511 memory locations. The CPU adds the unsigned byte in the index register to the unsigned byte following the opcode. The sum is the conditional address of the operand. These instructions can access locations \$0000–\$01FE.

Indexed 8-bit offset instructions are useful for selecting the *k*th element in an *n*-element table. The table can begin anywhere within the first 256 memory locations and could extend as far as location 510 (\$01FE). The *k* value is typically in the index register, and the address of the beginning of the table is in the byte following the opcode.

14.1.7 Indexed, 16-Bit Offset

Indexed, 16-bit offset instructions are three-byte instructions that can access data with variable addresses at any location in memory. The CPU adds the unsigned byte in the index register to the two unsigned bytes following the opcode. The sum is the conditional address of the operand. The first byte after the opcode is the high byte of the 16-bit offset; the second byte is the low byte of the offset. These instructions can address any location in memory.

Indexed, 16-bit offset instructions are useful for selecting the *k*th element in an *n*-element table anywhere in memory.

As with direct and extended addressing the Motorola assembler determines the shortest form of indexed addressing.

14.1.8 Relative

Relative addressing is only for branch instructions. If the branch condition is true, the CPU finds the conditional branch destination by adding the signed byte following the opcode to the contents of the program counter. If the branch condition is not true, the CPU goes to the next instruction. The offset is a signed, two's complement byte that gives a branching range of -128 to $+127$ bytes from the address of the next location after the branch instruction.

When using the Motorola assembler, the programmer does not need to calculate the offset, because the assembler determines the proper offset and verifies that it is within the span of the branch.

14.2 INSTRUCTION TYPES

The MCU instructions fall into the following five categories:

- Register/Memory Instructions
- Read-Modify-Write Instructions
- Jump/Branch Instructions
- Bit Manipulation Instructions
- Control Instructions

14.2.1 Register/Memory Instructions

Most of these instructions use two operands. One operand is in either the accumulator or the index register. The CPU finds the other operand in memory. Table 14-1 lists the register/memory instructions.

Table 14-1. Register/Memory Instructions

Instruction	Mnemonic
Add Memory Byte and Carry Bit to Accumulator	ADC
Add Memory Byte to Accumulator	ADD
AND Memory Byte with Accumulator	AND
Bit Test Accumulator	BIT
Compare Accumulator	CMP
Compare Index Register with Memory Byte	CPX
EXCLUSIVE OR Accumulator with Memory Byte	EOR
Load Accumulator with Memory Byte	LDA
Load Index Register with Memory Byte	LDX
Multiply	MUL
OR Accumulator with Memory Byte	ORA
Subtract Memory Byte and Carry Bit from Accumulator	SBC
Store Accumulator in Memory	STA
Store Index Register in Memory	STX
Subtract Memory Byte from Accumulator	SUB

14.2.2 Read-Modify-Write Instructions

These instructions read a memory location or a register, modify its contents, and write the modified value back to the memory location or to the register. The test for negative or zero instruction (TST) is an exception to the read-modify-write sequence because it does not write a replacement value. Table 14-2 lists the read-modify-write instructions.

Table 14-2. Read-Modify-Write Instructions

Instruction	Mnemonic
Arithmetic Shift Left	ASL
Arithmetic Shift Right	ASR
Clear Bit in Memory	BCLR
Set Bit in Memory	BSET
Clear	CLR
Complement (One's Complement)	COM
Decrement	DEC
Increment	INC
Logical Shift Left	LSL
Logical Shift Right	LSR
Negate (Two's Complement)	NEG
Rotate Left through Carry Bit	ROL
Rotate Right through Carry Bit	ROR
Test for Negative or Zero	TST

14.2.3 Jump/Branch Instructions

Jump instructions allow the CPU to interrupt the normal sequence of the program counter. The unconditional jump instruction (JMP) and the jump to subroutine instruction (JSR) have no register operand. Branch instructions allow the CPU to interrupt the normal sequence of the program counter when a test condition is met. If the test condition is not met, the branch is not performed. All branch instructions use relative addressing.

Bit test and branch instructions cause a branch based on the state of any readable bit in the first 256 memory locations. These three-byte instructions use a combination of direct addressing and relative addressing. The direct address of the byte to be tested is in the byte following the opcode. The third byte is the signed offset byte. The CPU finds the conditional branch destination by adding

the third byte to the program counter if the specified bit tests true. The bit to be tested and its condition (set or clear) is part of the opcode. The span of branching is from -128 to $+127$ from the address of the next location after the branch instruction. The CPU also transfers the tested bit to the carry/borrow bit of the condition code register. Table 14-3 lists the jump and branch instructions.

Table 14-3. Jump and Branch Instructions

Instruction	Mnemonic
Branch if Carry Bit Clear	BCC
Branch if Carry Bit Set	BCS
Branch if Equal	BEQ
Branch if Half-Carry Bit Clear	BHCC
Branch if Half-Carry Bit Set	BHCS
Branch if Higher	BHI
Branch if Higher or Same	BHS
Branch if $\overline{\text{IRQ}}$ Pin High	BIH
Branch if $\overline{\text{IRQ}}$ Pin Low	BIL
Branch if Lower	BLO
Branch if Lower or Same	BLS
Branch if Interrupt Mask Clear	BMC
Branch if Minus	BMI
Branch if Interrupt Mask Set	BMS
Branch if Not Equal	BNE
Branch if Plus	BPL
Branch Always	BRA
Branch if Bit Clear	BRCLR
Branch Never	BRN
Branch if Bit Set	BRSET
Branch to Subroutine	BSR
Unconditional Jump	JMP
Jump to Subroutine	JSR

14.2.4 Bit Manipulation Instructions

The CPU can set or clear any writable bit in the first 256 bytes of memory. Port registers, port data direction registers, timer registers, and on-chip RAM locations are in the first 256 bytes of memory. The CPU can also test and branch based on the state of any bit in any of the first 256 memory locations. Bit manipulation instructions use direct addressing. Table 14-4 lists these instructions.

Table 14-4. Bit Manipulation Instructions

Instruction	Mnemonic
Clear Bit	BCLR
Branch if Bit Clear	BRCLR
Branch if Bit Set	BRSET
Set Bit	BSET

14.2.5 Control Instructions

These register reference instructions control CPU operation during program execution. Control instructions, listed in Table 14-5, use inherent addressing.

Table 14-5. Control Instructions

Instruction	Mnemonic
Clear Carry Bit	CLC
Clear Interrupt Mask	CLI
No Operation	NOP
Reset Stack Pointer	RSP
Return from Interrupt	RTI
Return from Subroutine	RTS
Set Carry Bit	SEC
Set Interrupt Mask	SEI
Stop Oscillator and Enable $\overline{\text{IRQ}}$ Pin	STOP
Software Interrupt	SWI
Transfer Accumulator to Index Register	TAX
Transfer Index Register to Accumulator	TXA
Stop CPU Clock and Enable Interrupts	WAIT

14.3 INSTRUCTION SET SUMMARY

Table 14-6 is an alphabetical list of all M68HC05 instructions and shows the effect of each instruction on the condition code register.

Table 14-6. Instruction Set Summary

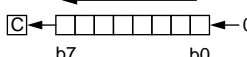
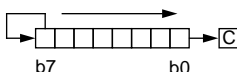
Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			H	I	N	Z	C				
ADC #opr ADC opr ADC opr ADC opr,X ADC opr,X ADC ,X	Add with Carry	$A \leftarrow (A) + (M) + (C)$						IMM	A9	ii	2
								DIR	B9	dd	3
			x	—	↑	↑	↑	EXT	C9	hh ll	4
								IX2	D9	ee ff	5
								IX1	E9	ff	4
								IX	F9		3
ADD #opr ADD opr ADD opr ADD opr,X ADD opr,X ADD ,X	Add without Carry	$A \leftarrow (A) + (M)$						IMM	AB	ii	2
								DIR	BB	dd	3
			x	—	↑	↑	↑	EXT	CB	hh ll	4
								IX2	DB	ee ff	5
								IX1	EB	ff	4
								IX	FB		3
AND #opr AND opr AND opr AND opr,X AND opr,X AND ,X	Logical AND	$A \leftarrow (A) \wedge (M)$						IMM	A4	ii	2
								DIR	B4	dd	3
			—	—	↑	↑	—	EXT	C4	hh ll	4
								IX2	D4	ee ff	5
								IX1	E4	ff	4
								IX	F4		3
ASL opr ASLA ASLX ASL opr,X ASL ,X	Arithmetic Shift Left (Same as LSL)							DIR	38	dd	5
			—	—	↑	↑	↑	INH	48		3
								INH	58		3
								IX1	68	ff	6
								IX	78		5
ASR opr ASRA ASRX ASR opr,X ASR ,X	Arithmetic Shift Right							DIR	37	dd	5
			—	—	↑	↑	↑	INH	47		3
								INH	57		3
								IX1	67	ff	6
								IX	77		5
BCC rel	Branch if Carry Bit Clear	$PC \leftarrow (PC) + 2 + rel ? C = 0$	—	—	—	—	—	REL	24	rr	3
BCLR n opr	Clear Bit n	$Mn \leftarrow 0$						DIR (b0)	11	dd	5
								DIR (b1)	13	dd	5
								DIR (b2)	15	dd	5
								DIR (b3)	17	dd	5
								DIR (b4)	19	dd	5
								DIR (b5)	1B	dd	5
								DIR (b6)	1D	dd	5
								DIR (b7)	1F	dd	5
BCS rel	Branch if Carry Bit Set (Same as BLO)	$PC \leftarrow (PC) + 2 + rel ? C = 1$	—	—	—	—	—	REL	25	rr	3
BEQ rel	Branch if Equal	$PC \leftarrow (PC) + 2 + rel ? Z = 1$	—	—	—	—	—	REL	27	rr	3

Table 14-6. Instruction Set Summary (Continued)

Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			H	I	N	Z	C				
BHCC <i>rel</i>	Branch if Half-Carry Bit Clear	$PC \leftarrow (PC) + 2 + rel ? H = 0$	—	—	—	—	—	REL	28	rr	3
BHCS <i>rel</i>	Branch if Half-Carry Bit Set	$PC \leftarrow (PC) + 2 + rel ? H = 1$	—	—	—	—	—	REL	29	rr	3
BHI <i>rel</i>	Branch if Higher	$PC \leftarrow (PC) + 2 + rel ? C \vee Z = 0$	—	—	—	—	—	REL	22	rr	3
BHS <i>rel</i>	Branch if Higher or Same	$PC \leftarrow (PC) + 2 + rel ? C = 0$	—	—	—	—	—	REL	24	rr	3
BIH <i>rel</i>	Branch if \overline{IRQ} Pin High	$PC \leftarrow (PC) + 2 + rel ? \overline{IRQ} = 1$	—	—	—	—	—	REL	2F	rr	3
BIL <i>rel</i>	Branch if \overline{IRQ} Pin Low	$PC \leftarrow (PC) + 2 + rel ? \overline{IRQ} = 0$	—	—	—	—	—	REL	2E	rr	3
BIT # <i>opr</i> BIT <i>opr</i> BIT <i>opr</i> BIT <i>opr</i> ,X BIT <i>opr</i> ,X BIT ,X	Bit Test Accumulator with Memory Byte	(A) \wedge (M)	—	—	↑	↑	—	IMM DIR EXT IX2 IX1 IX	A5 B5 C5 D5 E5 F5	ii dd hh ll ee ff ff p	2 3 4 5 4 3
BLO <i>rel</i>	Branch if Lower (Same as BCS)	$PC \leftarrow (PC) + 2 + rel ? C = 1$	—	—	—	—	—	REL	25	rr	3
BLS <i>rel</i>	Branch if Lower or Same	$PC \leftarrow (PC) + 2 + rel ? C \vee Z = 1$	—	—	—	—	—	REL	23	rr	3
BMC <i>rel</i>	Branch if Interrupt Mask Clear	$PC \leftarrow (PC) + 2 + rel ? I = 0$	—	—	—	—	—	REL	2C	rr	3
BMI <i>rel</i>	Branch if Minus	$PC \leftarrow (PC) + 2 + rel ? N = 1$	—	—	—	—	—	REL	2B	rr	3
BMS <i>rel</i>	Branch if Interrupt Mask Set	$PC \leftarrow (PC) + 2 + rel ? I = 1$	—	—	—	—	—	REL	2D	rr	3
BNE <i>rel</i>	Branch if Not Equal	$PC \leftarrow (PC) + 2 + rel ? Z = 0$	—	—	—	—	—	REL	26	rr	3
BPL <i>rel</i>	Branch if Plus	$PC \leftarrow (PC) + 2 + rel ? N = 0$	—	—	—	—	—	REL	2A	rr	3
BRA <i>rel</i>	Branch Always	$PC \leftarrow (PC) + 2 + rel ? 1 = 1$	—	—	—	—	—	REL	20	rr	3
BRCLR <i>n opr rel</i>	Branch if bit n clear	$PC \leftarrow (PC) + 2 + rel ? Mn = 0$	—	—	—	—	↑	DIR (b0) DIR (b1) DIR (b2) DIR (b3) DIR (b4) DIR (b5) DIR (b6) DIR (b7)	01 03 05 07 09 0B 0D 0F	dd rr dd rr dd rr dd rr dd rr dd rr dd rr dd rr	5 5 5 5 5 5 5 5
BRSET <i>n opr rel</i>	Branch if Bit n Set	$PC \leftarrow (PC) + 2 + rel ? Mn = 1$	—	—	—	—	↑	DIR (b0) DIR (b1) DIR (b2) DIR (b3) DIR (b4) DIR (b5) DIR (b6) DIR (b7)	00 02 04 06 08 0A 0C 0E	dd rr dd rr dd rr dd rr dd rr dd rr dd rr dd rr	5 5 5 5 5 5 5 5
BRN <i>rel</i>	Branch Never	$PC \leftarrow (PC) + 2 + rel ? 1 = 0$	—	—	—	—	—	REL	21	rr	3

Table 14-6. Instruction Set Summary (Continued)

Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			H	I	N	Z	C				
BSET <i>n opr</i>	Set Bit <i>n</i>	$M_n \leftarrow 1$						DIR (b0)	10	dd	5
								DIR (b1)	12	dd	5
								DIR (b2)	14	dd	5
								DIR (b3)	16	dd	5
								DIR (b4)	18	dd	5
								DIR (b5)	1A	dd	5
								DIR (b6)	1C	dd	5
								DIR (b7)	1E	dd	5
BSR <i>rel</i>	Branch to Subroutine	$PC \leftarrow (PC) + 2$; push (PCL) $SP \leftarrow (SP) - 1$; push (PCH) $SP \leftarrow (SP) - 1$ $PC \leftarrow (PC) + rel$	—	—	—	—	—	REL	AD	rr	6
CLC	Clear Carry Bit	$C \leftarrow 0$	—	—	—	—	0	INH	98		2
CLI	Clear Interrupt Mask	$I \leftarrow 0$	—	0	—	—	—	INH	9A		2
CLR <i>opr</i> CLRA CLR X CLR <i>opr,X</i> CLR ,X	Clear Byte	$M \leftarrow \$00$						DIR	3F	dd	5
		$A \leftarrow \$00$						INH	4F		3
		$X \leftarrow \$00$	—	—	0	1	—	INH	5F		3
		$M \leftarrow \$00$						IX1	6F	ff	6
		$M \leftarrow \$00$						IX	7F		5
CMP # <i>opr</i> CMP <i>opr</i> CMP <i>opr</i> CMP <i>opr,X</i> CMP <i>opr,X</i> CMP ,X	Compare Accumulator with Memory Byte	$(A) - (M)$						IMM	A1	ii	2
								DIR	B1	dd	3
			—	—	↑	↑	↑	EXT	C1	hh ll	4
								IX2	D1	ee ff	5
								IX1	E1	ff	4
								IX	F1		3
COM <i>opr</i> COMA COM X COM <i>opr,X</i> COM ,X	Complement Byte (One's Complement)	$M \leftarrow (\overline{M}) = \$FF - (M)$						DIR	33	dd	5
		$A \leftarrow (\overline{A}) = \$FF - (M)$						INH	43		3
		$X \leftarrow (\overline{X}) = \$FF - (M)$	—	—	↑	↑	1	INH	53		3
		$M \leftarrow (\overline{M}) = \$FF - (M)$						IX1	63	ff	6
		$M \leftarrow (\overline{M}) = \$FF - (M)$						IX	73		5
CPX # <i>opr</i> CPX <i>opr</i> CPX <i>opr</i> CPX <i>opr,X</i> CPX <i>opr,X</i> CPX ,X	Compare Index Register with Memory Byte	$(X) - (M)$						IMM	A3	ii	2
								DIR	B3	dd	3
			—	—	↑	↑	1	EXT	C3	hh ll	4
								IX2	D3	ee ff	5
								IX1	E3	ff	4
								IX	F3		3
DEC <i>opr</i> DECA DEC X DEC <i>opr,X</i> DEC ,X	Decrement Byte	$M \leftarrow (M) - 1$						DIR	3A	dd	5
		$A \leftarrow (A) - 1$						INH	4A		3
		$X \leftarrow (X) - 1$	—	—	↑	↑	—	INH	5A		3
		$M \leftarrow (M) - 1$						IX1	6A	ff	6
		$M \leftarrow (M) - 1$						IX	7A		5
EOR # <i>opr</i> EOR <i>opr</i> EOR <i>opr</i> EOR <i>opr,X</i> EOR <i>opr,X</i> EOR ,X	EXCLUSIVE OR Accumulator with Memory Byte	$A \leftarrow (A) \oplus (M)$						IMM	A8	ii	2
								DIR	B8	dd	3
			—	—	↑	↑	—	EXT	C8	hh ll	4
								IX2	D8	ee ff	5
								IX1	E8	ff	4
								IX	F8		3

Table 14-6. Instruction Set Summary (Continued)

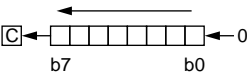
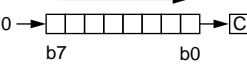
Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			H	I	N	Z	C				
INC <i>opr</i> INCA INCX INC <i>opr</i> ,X INC ,X	Increment Byte	$M \leftarrow (M) + 1$ $A \leftarrow (A) + 1$ $X \leftarrow (X) + 1$ $M \leftarrow (M) + 1$ $M \leftarrow (M) + 1$	—	—	↑	↑	—	DIR INH INH IX1 IX	3C 4C 5C 6C 7C	dd ff	5 3 3 6 5
JMP <i>opr</i> JMP <i>opr</i> JMP <i>opr</i> ,X JMP <i>opr</i> ,X JMP ,X	Unconditional Jump	$PC \leftarrow \text{Jump Address}$	—	—	—	—	—	DIR EXT IX2 IX1 IX	BC C C D C EC FC	dd hh ll ee ff ff	2 3 4 3 2
JSR <i>opr</i> JSR <i>opr</i> JSR <i>opr</i> ,X JSR <i>opr</i> ,X JSR ,X	Jump to Subroutine	$PC \leftarrow (PC) + n$ ($n = 1, 2, \text{ or } 3$) Push (PCL); $SP \leftarrow (SP) - 1$ Push (PCH); $SP \leftarrow (SP) - 1$ $PC \leftarrow \text{Conditional Address}$	—	—	—	—	—	DIR EXT IX2 IX1 IX	BD C D D D ED FD	dd hh ll ee ff ff	5 6 7 6 5
LDA # <i>opr</i> LDA <i>opr</i> LDA <i>opr</i> LDA <i>opr</i> ,X LDA <i>opr</i> ,X LDA ,X	Load Accumulator with Memory Byte	$A \leftarrow (M)$	—	—	↑	↑	—	IMM DIR EXT IX2 IX1 IX	A6 B6 C6 D6 E6 F6	ii dd hh ll ee ff ff	2 3 4 5 4 3
LDX # <i>opr</i> LDX <i>opr</i> LDX <i>opr</i> LDX <i>opr</i> ,X LDX <i>opr</i> ,X LDX ,X	Load Index Register with Memory Byte	$X \leftarrow (M)$	—	—	↑	↑	—	IMM DIR EXT IX2 IX1 IX	AE BE CE DE EE FE	ii dd hh ll ee ff ff	2 3 4 5 4 3
LSL <i>opr</i> LSLA LSLX LSL <i>opr</i> ,X LSL ,X	Logical Shift Left (Same as ASL)		—	—	↑	↑	↑	DIR INH INH IX1 IX	38 48 58 68 78	dd ff	5 3 3 6 5
LSR <i>opr</i> LSRA LSRX LSR <i>opr</i> ,X LSR ,X	Logical Shift Right		—	—	0	↑	↑	DIR INH INH IX1 IX	34 44 54 64 74	dd ff	5 3 3 6 5
MUL	Unsigned Multiply	$X : A \leftarrow (X) \times (A)$	0	—	—	—	0	INH	42		11
NEG <i>opr</i> NEGA NEGX NEG <i>opr</i> ,X NEG ,X	Negate Byte (Two's Complement)	$M \leftarrow -(M) = \$00 - (M)$ $A \leftarrow -(A) = \$00 - (A)$ $X \leftarrow -(X) = \$00 - (X)$ $M \leftarrow -(M) = \$00 - (M)$ $M \leftarrow -(M) = \$00 - (M)$	—	—	↑	↑	↑	DIR INH INH IX1 IX	30 40 50 60 70	ii ff	5 3 3 6 5
NOP	No Operation		—	—	—	—	—	INH	9D		2

Table 14-6. Instruction Set Summary (Continued)

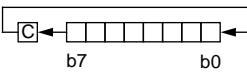
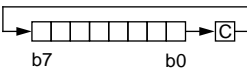
Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			H	I	N	Z	C				
ORA #opr ORA opr ORA opr ORA opr,X ORA opr,X ORA ,X	Logical OR Accumulator with Memory	$A \leftarrow (A) \vee (M)$	—	—	↑	↑	—	IMM DIR EXT IX2 IX1 IX	AA BA CA DA EA FA	ii dd hh ll ee ff ff	2 3 4 5 4 3
ROL opr ROLA ROLX ROL opr,X ROL ,X	Rotate Byte Left through Carry Bit		—	—	↑	↑	↑	DIR INH INH IX1 IX	39 49 59 69 79	dd ff	5 3 3 6 5
ROR opr RORA RORX ROR opr,X ROR ,X	Rotate Byte Right through Carry Bit		—	—	↑	↑	↑	DIR INH INH IX1 IX	36 46 56 66 76	dd ff	5 3 3 6 5
RSP	Reset Stack Pointer	$SP \leftarrow \$00FF$	—	—	—	—	—	INH	9C		2
RTI	Return from Interrupt	$SP \leftarrow (SP) + 1$; Pull (CCR) $SP \leftarrow (SP) + 1$; Pull (A) $SP \leftarrow (SP) + 1$; Pull (X) $SP \leftarrow (SP) + 1$; Pull (PCH) $SP \leftarrow (SP) + 1$; Pull (PCL)	↑	↑	↑	↑	↑	INH	80		6
RTS	Return from Subroutine	$SP \leftarrow (SP) + 1$; Pull (PCH) $SP \leftarrow (SP) + 1$; Pull (PCL)						INH			
SBC #opr SBC opr SBC opr SBC opr,X SBC opr,X SBC ,X	Subtract Memory Byte and Carry Bit from Accumulator	$A \leftarrow (A) - (M) - (C)$	—	—	↑	↑	↑	IMM DIR EXT IX2 IX1 IX	A2 B2 C2 D2 E2 F2	ii dd hh ll ee ff ff	2 3 4 5 4 3
SEC	Set Carry Bit	$C \leftarrow 1$	—	—	—	—	1	INH	99		2
SEI	Set Interrupt Mask	$I \leftarrow 1$	—	1	—	—	—	INH	9B		2
STA opr STA opr STA opr,X STA opr,X STA ,X	Store Accumulator in Memory	$M \leftarrow (A)$	—	—	↑	↑	—	DIR EXT IX2 IX1 IX	B7 C7 D7 E7 F7	dd hh ll ee ff ff	4 5 6 5 4
STOP	Stop Oscillator and Enable \overline{IRQ} Pin		—	0	—	—	—	INH	8E		2
STX opr STX opr STX opr,X STX opr,X STX ,X	Store Index Register In Memory	$M \leftarrow (X)$	—	—	↑	↑	—	DIR EXT IX2 IX1 IX	BF CF DF EF FF	dd hh ll ee ff ff	4 5 6 5 4

Table 14-6. Instruction Set Summary (Continued)

Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			H	I	N	Z	C				
SUB # <i>opr</i> SUB <i>opr</i> SUB <i>opr</i> SUB <i>opr</i> ,X SUB <i>opr</i> ,X SUB ,X	Subtract Memory Byte from Accumulator	$A \leftarrow (A) - (M)$	—	—	↑	↑	↑	IMM DIR EXT IX2 IX1 IX	A0 B0 C0 D0 E0 F0	ii dd hh ll ee ff ff	2 3 4 5 4 3
SWI	Software Interrupt	$PC \leftarrow (PC) + 1$; Push (PCL) $SP \leftarrow (SP) - 1$; Push (PCH) $SP \leftarrow (SP) - 1$; Push (X) $SP \leftarrow (SP) - 1$; Push (A) $SP \leftarrow (SP) - 1$; Push (CCR) $SP \leftarrow (SP) - 1$; $I \leftarrow 1$ PCH \leftarrow Interrupt Vector High Byte PCL \leftarrow Interrupt Vector Low Byte	—	1	—	—	—	INH	83		10
TAX	Transfer Accumulator to Index Register	$X \leftarrow (A)$	—	—	—	—	—	INH	97		2
TST <i>opr</i> TSTA TSTX TST <i>opr</i> ,X TST ,X	Test Memory Byte for Negative or Zero	$(M) - \$00$	—	—	—	—	—	DIR INH INH IX1 IX	3D 4D 5D 6D 7D	dd ff	4 3 3 5 4
TXA	Transfer Index Register to Accumulator	$A \leftarrow (X)$	—	—	—	—	—	INH	9F		2
WAIT	Stop CPU Clock and Enable Interrupts		—	↑	—	—	—	INH	8F		2

A Accumulator
 C Carry/borrow flag
 CCR Condition code register
 dd Direct address of operand
 dd rr Direct address of operand and relative offset of branch instruction
 DIR Direct addressing mode
 ee ff High and low bytes of offset in indexed, 16-bit offset addressing
 EXT Extended addressing mode
 ff Offset byte in indexed, 8-bit offset addressing
 H Half-carry flag
 hh ll High and low bytes of operand address in extended addressing
 I Interrupt mask
 ii Immediate operand byte
 IMM Immediate addressing mode
 INH Inherent addressing mode
 IX Indexed, no offset addressing mode
 IX1 Indexed, 8-bit offset addressing mode
 IX2 Indexed, 16-bit offset addressing mode
 M Memory location
 N Negative flag
 n Any bit

opr Operand (one or two bytes)
 PC Program counter
 PCH Program counter high byte
 PCL Program counter low byte
 REL Relative addressing mode
rel Relative program counter offset byte
 rr Relative program counter offset byte
 SP Stack pointer
 X Index register
 Z Zero flag
 # Immediate value
 ^ Logical AND
 v Logical OR
 ⊕ Logical EXCLUSIVE OR
 () Contents of
 -() Negation (two's complement)
 ← Loaded with
 ? If
 : Concatenated with
 ↑ Set or cleared
 — Not affected

Bit Manipulation				Branch		Read-Modify-Write				Control			Register/Memory				
DIR	DIR	REL	DIR	INH	INH	INH	IX1	IX	INH	INH	IMM	DIR	EXT	IX2	IX1	IX	
0		1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
MSB		LSB															
0	BRSET0 ₃ DIR 2	BSET0 ₅ DIR 2	BRA ₃ REL 2	NEG ₅ DIR 1	NEGA ₃ INH 1	NEG ₃ INH 2	NEG ₆ IX1 1	NEG ₅ IX 1	RTI ₉ INH 1		SUB ₂ IMM 2	SUB ₃ DIR 3	SUB ₄ EXT 3	SUB ₅ IX2 2	SUB ₄ IX1 1	SUB ₃ IX 1	
1	BRCLR0 ₃ DIR 2	BCLR0 ₅ DIR 2	BRN ₃ REL 2						RTS ₆ INH 1		CMP ₂ IMM 2	CMP ₃ DIR 3	CMP ₄ EXT 3	CMP ₅ IX2 2	CMP ₄ IX1 1	CMP ₃ IX 1	
2	BRSET1 ₃ DIR 2	BSET1 ₅ DIR 2	BHI ₃ REL 2		MUL ₁₁ INH 1						SBC ₂ IMM 2	SBC ₃ DIR 3	SBC ₄ EXT 3	SBC ₅ IX2 2	SBC ₄ IX1 1	SBC ₃ IX 1	
3	BRCLR1 ₃ DIR 2	BCLR1 ₅ DIR 2	BLS ₃ REL 2	COM ₅ DIR 1	COMA ₃ INH 1	COMX ₃ INH 2	COM ₆ IX1 1	COM ₅ IX 1	SWI ₁₀ INH 1		CPX ₂ IMM 2	CPX ₃ DIR 3	CPX ₄ EXT 3	CPX ₅ IX2 2	CPX ₄ IX1 1	CPX ₃ IX 1	
4	BRSET2 ₃ DIR 2	BSET2 ₅ DIR 2	BCC ₃ REL 2	LSR ₅ DIR 1	LSRA ₃ INH 1	LSRX ₃ INH 2	LSR ₆ IX1 1	LSR ₅ IX 1			AND ₂ IMM 2	AND ₃ DIR 3	AND ₄ EXT 3	AND ₅ IX2 2	AND ₄ IX1 1	AND ₃ IX 1	
5	BRCLR2 ₃ DIR 2	BCLR2 ₅ DIR 2	BCS/BLO ₃ REL 2								BIT ₂ IMM 2	BIT ₃ DIR 3	BIT ₄ EXT 3	BIT ₅ IX2 2	BIT ₄ IX1 1	BIT ₃ IX 1	
6	BRSET3 ₃ DIR 2	BSET3 ₅ DIR 2	BNE ₃ REL 2	ROR ₅ DIR 1	RORA ₃ INH 1	RORX ₃ INH 2	ROR ₆ IX1 1	ROR ₅ IX 1			LDA ₂ IMM 2	LDA ₃ DIR 3	LDA ₄ EXT 3	LDA ₅ IX2 2	LDA ₄ IX1 1	LDA ₃ IX 1	
7	BRCLR3 ₃ DIR 2	BCLR3 ₅ DIR 2	BEQ ₃ REL 2	ASR ₅ DIR 1	ASRA ₃ INH 1	ASRX ₃ INH 2	ASR ₆ IX1 1	ASR ₅ IX 1	TAX ₁ INH 2		STA ₂ IMM 2	STA ₃ DIR 3	STA ₄ EXT 3	STA ₅ IX2 2	STA ₄ IX1 1	STA ₃ IX 1	
8	BRSET4 ₃ DIR 2	BSET4 ₅ DIR 2	BHCC ₃ REL 2	ASL/LSL ₅ DIR 1	ASLA/LSLA ₃ INH 1	ASLX/LSLX ₃ INH 2	ASL/LSL ₆ IX1 1	ASL/LSL ₅ IX 1	CLC ₂ INH 2		EOR ₂ IMM 2	EOR ₃ DIR 3	EOR ₄ EXT 3	EOR ₅ IX2 2	EOR ₄ IX1 1	EOR ₃ IX 1	
9	BRCLR4 ₃ DIR 2	BCLR4 ₅ DIR 2	BHCS ₃ REL 2	ROL ₅ DIR 1	ROLA ₃ INH 1	ROLX ₃ INH 2	ROL ₆ IX1 1	ROL ₅ IX 1	SEC ₂ INH 2		ADC ₂ IMM 2	ADC ₃ DIR 3	ADC ₄ EXT 3	ADC ₅ IX2 2	ADC ₄ IX1 1	ADC ₃ IX 1	
A	BRSET5 ₃ DIR 2	BSET5 ₅ DIR 2	BPL ₃ REL 2	DEC ₅ DIR 1	DECA ₃ INH 1	DECX ₃ INH 2	DEC ₆ IX1 1	DEC ₅ IX 1	CLI ₂ INH 2		ORA ₂ IMM 2	ORA ₃ DIR 3	ORA ₄ EXT 3	ORA ₅ IX2 2	ORA ₄ IX1 1	ORA ₃ IX 1	
B	BRCLR5 ₃ DIR 2	BCLR5 ₅ DIR 2	BMI ₃ REL 2						SEI ₂ INH 2		ADD ₂ IMM 2	ADD ₃ DIR 3	ADD ₄ EXT 3	ADD ₅ IX2 2	ADD ₄ IX1 1	ADD ₃ IX 1	
C	BRSET6 ₃ DIR 2	BSET6 ₅ DIR 2	BMC ₃ REL 2	INC ₅ DIR 1	INCA ₃ INH 1	INCX ₃ INH 2	INC ₆ IX1 1	INC ₅ IX 1	RSP ₂ INH 2		JMP ₂ IMM 2	JMP ₃ DIR 3	JMP ₄ EXT 3	JMP ₅ IX2 2	JMP ₄ IX1 1	JMP ₃ IX 1	
D	BRCLR6 ₃ DIR 2	BCLR6 ₅ DIR 2	BMS ₃ REL 2	TST ₅ DIR 1	TSTA ₃ INH 1	TSTX ₃ INH 2	TST ₆ IX1 1	TST ₅ IX 1	NOP ₂ INH 2		BSR ₆ REL 2	JSR ₅ DIR 3	JSR ₆ EXT 3	JSR ₇ IX2 2	JSR ₆ IX1 1	JSR ₅ IX	

SECTION 15 ELECTRICAL SPECIFICATIONS

This section contains electrical and timing specifications.

15.1 MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Supply Voltage	V_{DD}	-0.3 to +7.0	V
Input Voltage	V_{IN}	$V_{SS} - 0.3$ to $V_{DD} + 0.3$	V
Self-Check Mode (\overline{IRQ} Pin Only)	V_{IN}	$V_{SS} - 0.3$ to $2 \times V_{DD} + 0.3$	V
Current Drain Per Pin Excluding V_{DD} and V_{SS}	I	25	mA
Operating Temperature Range MC68HC05CT4 (Standard)	T_A	T_L to T_H 0 to +70	$^{\circ}\text{C}$
Storage Temperature Range	T_{STG}	-65 to +150	$^{\circ}\text{C}$

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 $V_{SS} \leq (V_{IN} \text{ or } V_{OUT}) \leq V_{DD}$. Reliability of operation is enhanced if unused inputs are connected to an appropriate logic voltage level (for example, either V_{SS} or V_{DD}).

15.2 THERMAL CHARACTERISTICS

Characteristic	Symbol	Value	Unit
Thermal Resistance Plastic	θ_{JA}	70	$^{\circ}\text{C/W}$

15.3 DC ELECTRICAL CHARACTERISTICS (5.0 V)(V_{DD} = 5.0 Vdc ±10%, V_{SS} = 0 Vdc, T_A = 0 °C to +70 °C, unless otherwise noted)

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
Output High Voltage (I _{LOAD} = -0.8 mA) Port A, Port B, Port C, Port D	V _{OH}	V _{DD} - 0.8	—	—	V
Output Low Voltage (I _{LOAD} = 1.6 mA) Port A, Port B, Port C, Port D	V _{OL}	—	—	0.4	V
Input High Voltage Port A, Port B, Port C, Port D, $\overline{\text{IRQ}}$, $\overline{\text{RESET}}$, OSC1	V _{IH}	0.7 x V _{DD}	—	V _{DD}	V
Input Low Voltage Port A, Port B, Port C, Port D, $\overline{\text{IRQ}}$, $\overline{\text{RESET}}$, OSC1	V _{IL}	V _{SS}	—	0.3 x V _{DD}	V
Input Hysteresis TCAP, $\overline{\text{RESET}}$, $\overline{\text{IRQ}}$, PD4/SCK	V _{HYST}	0.8	0.9	1	V
Supply Current (see Notes) Run Wait Stop 25 °C 0 °C to +70 °C	I _{DD} I _{DD} I _{DD} I _{DD}	— — — —	TBD TBD 100 150	TBD TBD TBD TBD	mA mA μA μA
Supply Current (see Notes) Run Wait Stop 25 °C 0 °C to +70 °C	I _{DD2} I _{DD2} I _{DD2} I _{DD2}	— — — —	TBD TBD TBD TBD	TBD TBD TBD TBD	mA mA μA μA
I/O Ports Hi-Z Leakage Current Port A, Port B, Port D	I _{OZ}	—	—	5	μA
Input Current $\overline{\text{RESET}}$, $\overline{\text{IRQ}}$, OSC1 PC0-PC7 with Pullups Enabled	I _{IN} I _{IN}	— -20	— -125	10 -250	μA μA
Capacitance Ports (as Input or Output) $\overline{\text{RESET}}$, $\overline{\text{IRQ}}$	C _{OUT} C _{INT}	— —	— —	12 8	pF pF

NOTES:

1. All values shown reflect average measurements.
2. Typical values at midpoint of voltage range, 25°C only.
3. Wait I_{DD} and I_{DD2}: Only timer system active.
4. Run (Operating) I_{DD}, I_{DD2}, Wait I_{DD}, and I_{DD2}: Measured using external square wave clock source (f_{OSC} = 10.24 MHz), all inputs 0.2 V from rail; no dc loads, less than 50 pF on all outputs, C_L = 20 pF on OSC2.
5. Wait, Stop I_{DD}: All ports configured as inputs, V_{IL} = 0.2 V, V_{IH} = V_{DD} - 0.2 V.
6. Stop I_{DD} and I_{DD2} are measured with OSC1 = V_{SS}.
7. Wait I_{DD} and I_{DD2} are affected linearly by the OSC2 capacitance.
8. Run, Wait, and Stop I_{DD2} are measured directly off the V_{DD2} power supply.

15.4 DC ELECTRICAL CHARACTERISTICS (3.3 V)(V_{DD} = 3.3 Vdc ±10%, V_{SS} = 0 Vdc, T_A = 0 °C to +70 °C, unless otherwise noted)

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
Output High Voltage (I _{LOAD} = 0.2 mA) Port A, Port B, Port C, Port D	V _{OH}	V _{DD} - 0.3	—	—	V
Output Low Voltage (I _{LOAD} = 1.6 mA) Port A, Port B, Port C, Port D	V _{OL}	—	—	0.3	V
Input High Voltage Port A, Port B, Port C, Port D, $\overline{\text{IRQ}}$, $\overline{\text{RESET}}$, OSC1	V _{IH}	0.7 x V _{DD}	—	V _{DD}	V
Input Low Voltage Port A, Port B, Port C, Port D, $\overline{\text{IRQ}}$, $\overline{\text{RESET}}$, OSC1	V _{IL}	V _{SS}	—	0.3 x V _{DD}	V
Supply Current (see Notes) Run Wait Stop 25 °C 0 °C to +70 °C	I _{DD} I _{DD} I _{DD} I _{DD}	— — — —	TBD TBD 100 150	TBD TBD TBD TBD	mA mA μA μA
Supply Current (see Notes) Run Wait Stop 25 °C 0 °C to +70 °C	I _{DD2} I _{DD2} I _{DD2} I _{DD2}	— — — —	TBD TBD TBD TBD	TBD TBD TBD TBD	mA mA μA μA
I/O Ports Hi-Z Leakage Current Port A, Port B, Port D	I _{OZ}	—	—	5	μA
Input Current RESET, IRQ, OSC1 PC0-PC7 with Pullups Enabled	I _{IN} I _{IN}	— -20	— -50	10 -125	μA μA
Capacitance Ports (as Input or Output) RESET, IRQ	C _{OUT} C _{INT}	— —	— —	12 8	pF pF

NOTES:

1. All values shown reflect average measurements.
2. Typical values at midpoint of voltage range, 25°C only.
3. Wait I_{DD} and I_{DD2}: Only timer system active.
4. Run (Operating) I_{DD}, I_{DD2}, Wait I_{DD}, and I_{DD2}: Measured using external square wave clock source (f_{OSC} = 10.24 MHz), all inputs 0.2 V from rail; no dc loads, less than 50 pF on all outputs, C_L = 20 pF on OSC2.
5. Wait, Stop I_{DD}: All ports configured as inputs, V_{IL} = 0.2 V, V_{IH} = V_{DD} - 0.2 V.
6. Stop I_{DD} and I_{DD2} are measured with OSC1 = V_{SS}.
7. Wait I_{DD} and I_{DD2} are affected linearly by the OSC2 capacitance.
8. Run, Wait, and Stop I_{DD2} are measured directly off the V_{DD2} power supply.

15.5 CONTROL TIMING (3.3 V AND 5.0 V)(V_{DD} = 3.0 to 5.5 Vdc, V_{SS} = 0 Vdc, T_A = 0 °C to +70 °C, unless otherwise noted)

Characteristic	Symbol	Min	Max	Unit
Frequency of Operation				
Crystal	f _{OSC}	—	10.24	MHz
External Clock	f _{OSC}	dc	10.24	MHz
Internal Operating Frequency				
Crystal (f _{OSC} /5)	f _{OP}	—	2.048	MHz
External Clock (f _{OSC} /5)	f _{OP}	dc	2.048	MHz
Cycle Time	t _{CYC}	488	—	ns
Crystal Oscillator Start-up Time	t _{OXOV}	—	100	ms
Stop Recovery Start-up Time (Crystal Oscillator)	t _{ILCH}	—	100	ms
RESET Pulse Width	t _{RL}	1.5	—	t _{CYC}
Interrupt Pulse Width Low (Edge-Triggered)	t _{ILIH}	125	—	ns
Interrupt Pulse Period	t _{ILIL}	*	—	t _{CYC}
OSC1 Pulse Width	t _{OH} , t _{OL}	90	—	ns

* The minimum period T_{ILIL} should not be less than the number of cycle times it takes to execute the interrupt service routine plus 2T t_{CYC}.

15.6 COMPARATORS(V_{DD} = 3.3 Vdc ±10%, V_{SS} = 0 Vdc, T_A = 0 °C to +70 °C, unless otherwise noted)

Characteristic	Symbol	Min	Max	Unit
Input Voltage Range ¹	V _{INT}	V _{SS} - 0.5	V _{DD} +0.5	V
Voltage Comparator Propagation Time (100 mV overdrive) ²	t _{COMP}	—	10	μs
Common Mode Range ³	V _{COM}	V _{SS} +1.5	V _{DD} - 0.5	V
Offset ⁴	V _{OFF}	—	±20	mV
Comparator Enable Time ⁵	t _{CEN}	—	100	ms
Input Current V _{SS} ≤ V _{IN} ≤ V _{DD}	I _{IN}	—	1	μA
Input Current V _{IN} ≤ V _{SS}	I _{IN}	—	4.0	mA
Gain Bandwidth Product		—	10	KHz
Slew Rate	SR	5	—	V/μs
Saturation Voltage (I _{OUT} = -5mA)	V _{SAT}	—	0.5	V
Internal Voltage Reference CMP3-	V _{REF}	x 90%	x 110%	(2/3)V _{DD}

NOTES:

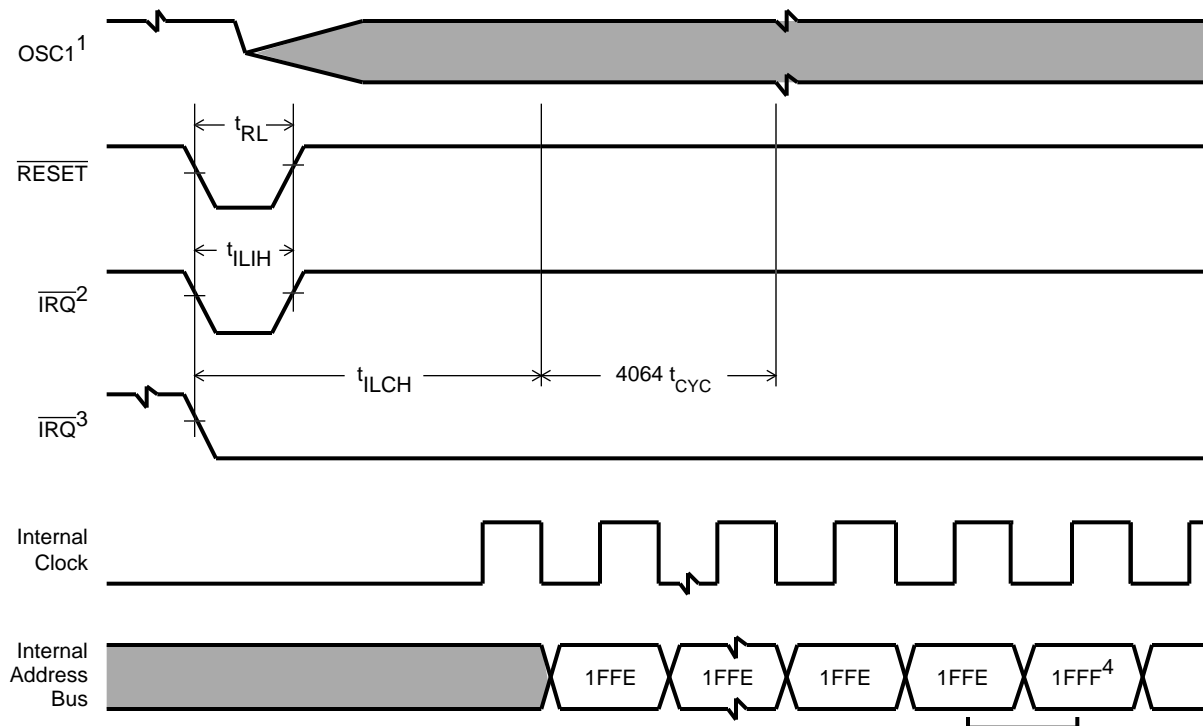
1. The comparator is guaranteed to function over the specified input voltage range with no erroneous outputs.
2. Signal propagation time through the comparators measured with 100 mv of overdrive.
3. Comparator is guaranteed to meet specifications; for example, SR, t_{COMP}, and V_{OFF}.
4. Input offset voltage is guaranteed over the temperature range.
5. Enable time is the time from enabling the comparator with the CEN bit until the comparator is fully functional.

15.7 PWM TIMING

Characteristic	Symbol	Min	Max	Unit
PWM Rise Time	t_{PWMR}	15	35	ns
PWM Fall Time	t_{PWMR}	15	35	ns

15.8 PLL SIGNAL INPUT

Characteristic	Symbol	Min	Max	Unit
Input Voltage on FINTX and FINRX (peak-to-peak)	V_{VCO}	200	—	mV _{pp}
Input Frequency on FINTX and FINRX	f_{VCO}	—	60	MHz

**NOTES:**

1. Represents the internal gating of the OSC1 pin.
2. $\overline{\text{IRQ}}$ pin edge-sensitive mask option.
3. $\overline{\text{IRQ}}$ pin level and edge-sensitive mask option.
4. $\overline{\text{RESET}}$ vector address shown for timing example.

$\overline{\text{RESET}}$ or Interrupt
Vector Fetch

Figure 15-1. Stop Recovery Timing Diagram

SECTION 16

MECHANICAL SPECIFICATIONS

The MC68HC05CT4 is available in a 44-pin plastic leaded chip carrier (PLCC) and a 44-pin quad flat pack (QFP). Package dimensions are provided in this section.

16.1 44-LEAD PLASTIC LEADED CHIP CARRIER (CASE 777-02)

SECTION 17

ORDERING INFORMATION

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

17.1 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 **17.2 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.

17.2 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.

17.3 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.

17.4 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.

17.5 MC ORDER NUMBERS

Table 17-1 shows the MC order numbers for the available package types.

Table 17-1. MC Order Numbers

MC Order Number	Operating Temperature Range
MC68HC05CT4FN	0° to 70°C
MC68HC705CT4FB	0° to 70°C

NOTE: FN = 44-Lead Plastic-Leaded Chip Carrier
FB = 44-Lead Quad Flat Pack

APPENDIX A

MC68HC705CT4

This section describes the differences between the MC68HC705CT4 and the MC68HC05CT4.

The MC68HC705CT4 is a 44-pin member of the MC68HC05 Family that supports the cordless telephone applications. The memory map includes 5136 bytes of user EPROM, 240 bytes of boot ROM, and 256 bytes of RAM. The MCU has three 8-bit input/output (I/O) ports: A, B, C; and one 7-bit I/O, port D. Port C has pullup options and keyscan capability. The MC68HC705CT4 includes a bird core, a bird timer, a synchronous serial I/O (SSI), 16-bit timer, a dual 60 MHz phase-lock loop (PLL), and an on-chip computer operating properly (COP) watchdog circuit.

A.1 MASK OPTIONS

The mask options on the MC68HC705CT4 are handled with eight EPROM bits (MOR).

These options are the port C pullups, the IRQ sensitivity, and the COP enable/disable. **A.4.3 MASK OPTION REGISTER (MOR).**

ROM versions of this device will have these options programmed by the factory.

NOTE

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

A.2 $\overline{\text{IRQ}}/V_{\text{PP}}$ (MASKABLE INTERRUPT REQUEST)

In addition to the functions described earlier, this pin is also used to supply the EPROM with the required programming voltage.

CAUTION

Port C interrupt is an external interrupt provided by port C that is generated when the part comes out of reset. Therefore, an interrupt is pending that is not be serviced until the CLI instruction is executed. This interrupt is produced because the state of mask option register (MOR) is not known until a few clock cycles after the part comes out of reset. After this interrupt occurs, normal operation resumes; unless the part is put in reset again.

A.3 MEMORY

The MC68HC705CT4 has an eight K-byte memory map consisting of user ROM, RAM, bootloader ROM, and I/O. Refer to Figure A-1.

A.4 EPROM

The user EPROM consists of 5120 bytes of EPROM from \$0B00 to \$1EFF and 16 bytes of user vectors from \$1FF0 to \$1FFF.

The bootloader ROM and vectors are located from \$1F01 to \$1FEF.

Twelve of the user vectors, \$1FF4 through \$1FFF, are dedicated to reset and interrupt vectors. The four remaining locations, \$1FF0, \$1FF1, \$1FF2 and \$1FF3, are general-purpose user EPROM locations. Location \$1F00 is the mask option register (MOR).

The contents of the EPROM cannot be prevented from being viewed externally, since security is not incorporated into the MC68HC705CT4.

A.5 BOOTLOADER MODE

Bootloader mode is entered upon the rising edge of $\overline{\text{RESET}}$ if the $\overline{\text{IRQ}}$ is at V_{TST} and the PB1 pin is at logic one. The bootloader code resides in the ROM from \$1F01 to \$1FEF. This program handles copying of user code from an external EPROM into the on-chip EPROM. The bootload function does not have to be done from an external EPROM, but it may be done from a host.

The user code must be a one-to-one correspondence with the internal EPROM addresses.

NOTE

The MCU designer **must** disable the COP hardware in bootloader mode.

A.5.1 Bootloader Functions

Three pins are used to select various bootloader functions: PB0, PB3, and PB4. PB0 is normally a SYNC pin, which is used to synchronize the MCU to an off-chip source driving EPROM data into the MCU. If an external EPROM is used, this pin must be connected to V_{SS} . PB4 and PB3 are used to select a programming mode. Two other pins, PC2 and PC3, are used to drive the PROG LED and the VRF LED, respectively. The programming modes, along with GATE and DRAIN stress, are shown in Table A-1.

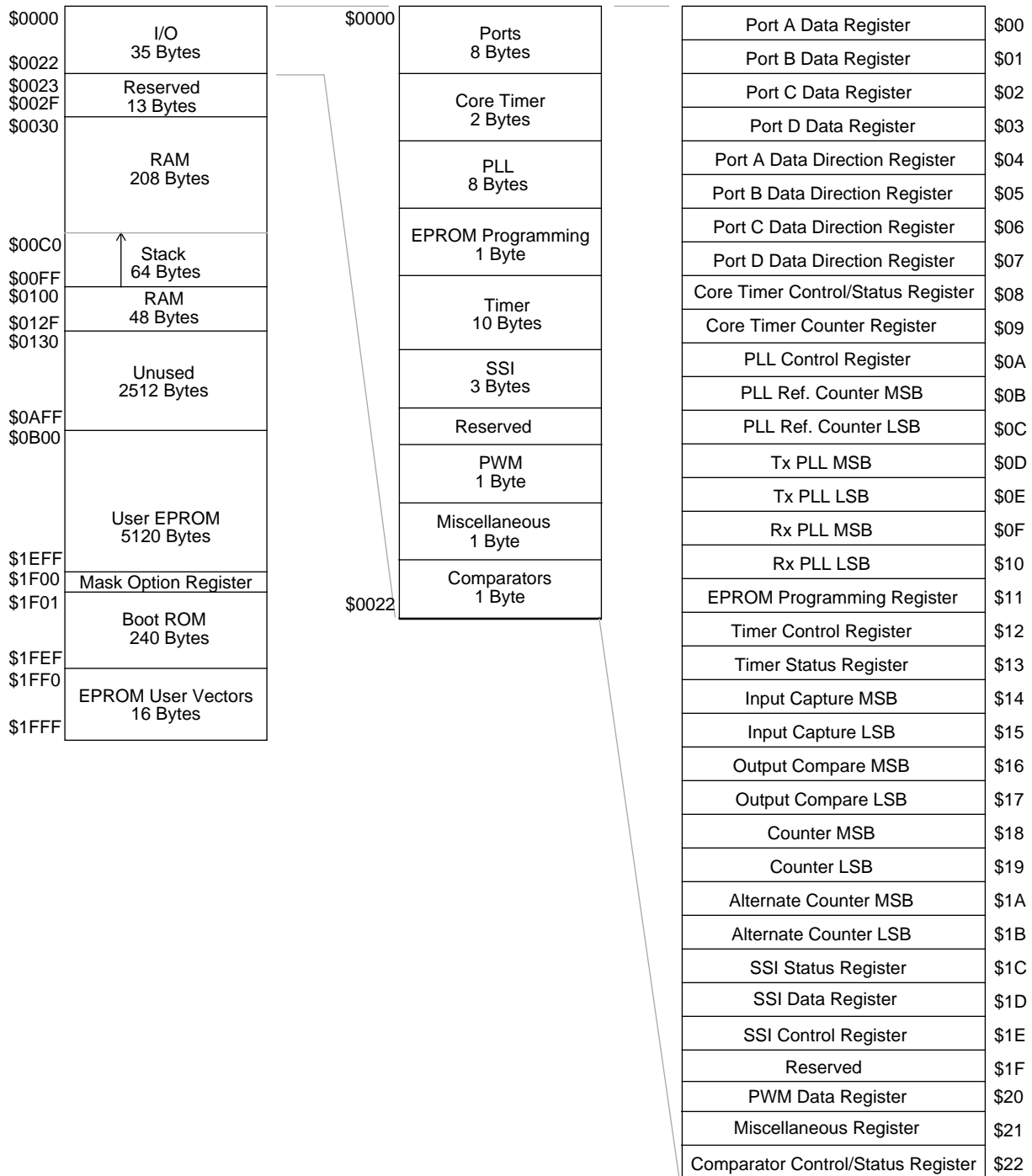
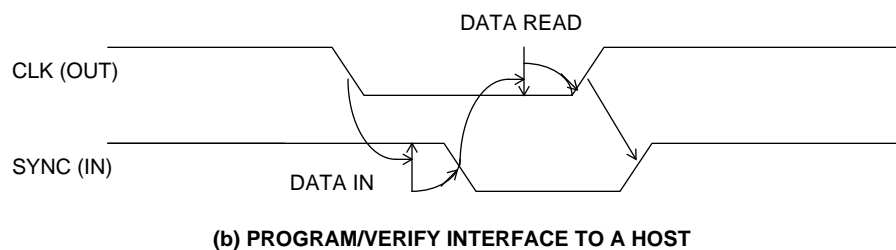
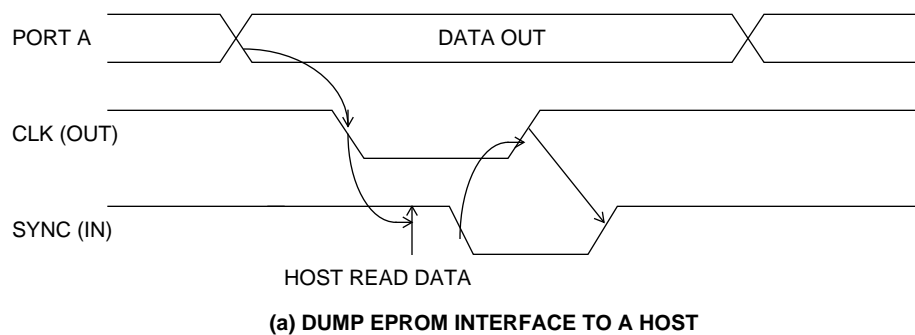


Figure A-1. MC68HC705CT4 8K Memory Map

Table A-1. Bootloader Functions

PB0	PB4	PB3	MODE
SYNC	1	1	PROGRAM/VERIFY
SYNC	1	0	VERIFY ONLY
SYNC	0	0	DUMP EPROM
0	0	1	GATE STRESS
1	0	1	DRAIN STRESS

The bootloader uses an external 12-bit counter to address the memory device containing the code to be copied. This counter requires a clock and a reset function. The 12-bit counter can address up to 4 K-bytes of memory, which means that a port pin has to be used to address the extra memory space.

**Figure A-2. Programmer Interface to Host**

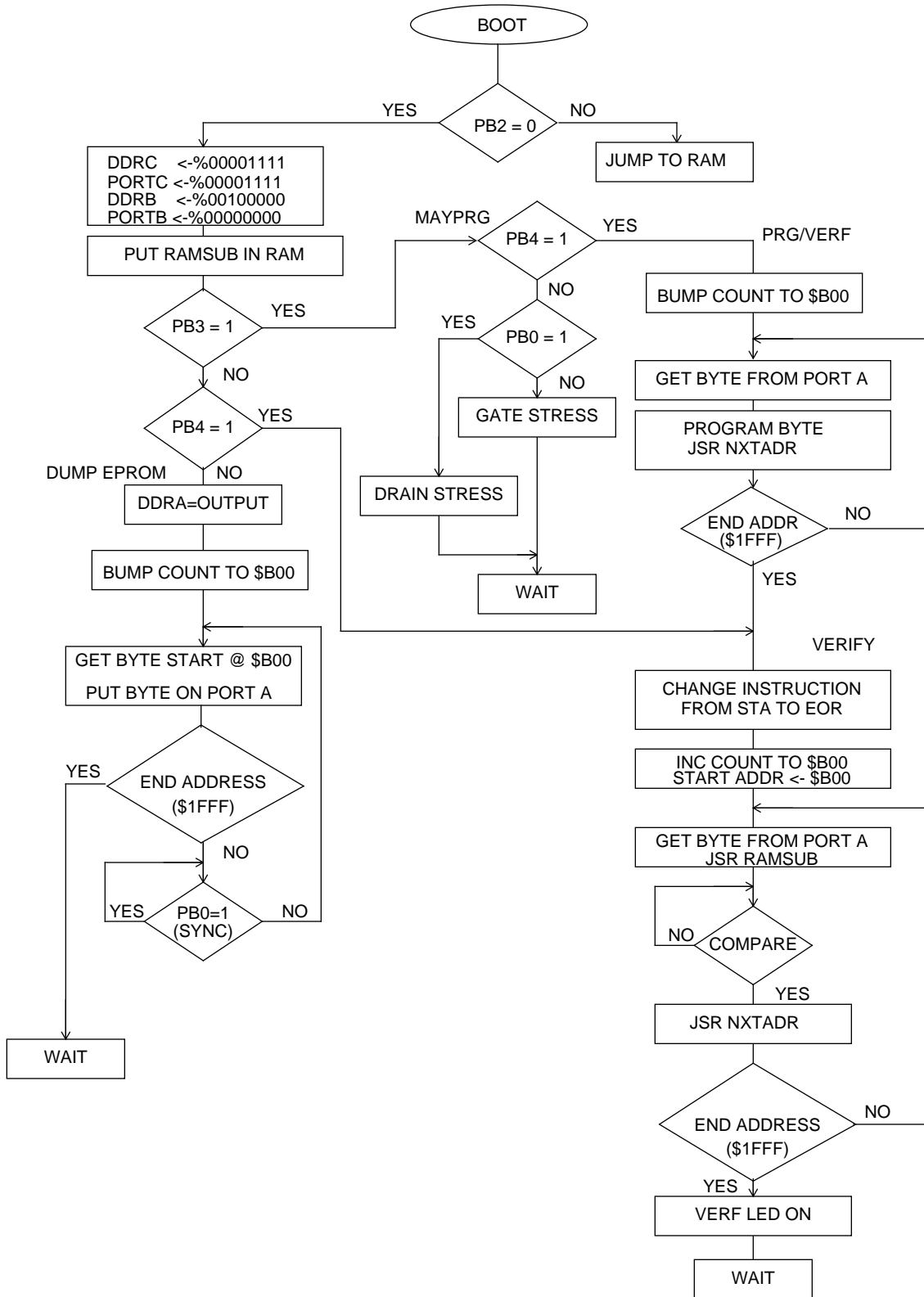


Figure A-3. MC68HC705CT4 Bootloader Flowchart



Figure A-4. MC68HC705CT4 Programming Circuit

A.5.2 Programming Register (PROG) \$11

This register is used to program the EPROM array. Only the LATCH and EPGM bits are available in user mode. To program a byte of EPROM, set LATCH, then write data to the desired address, then set EPGM for t_{EPGM} . EPTST, MBE, TS1, and TS0 are available only in the special modes (boot and test).

	7	6	5	4	3	2	1	0
\$11	MORON	EPTST	MBE	TS1	TS0	LATCH	0	EPGM
RESET:	0	0	0	0	0	0	0	0

Figure A-5. Programming Register

MORON — MOR On

READ: Only in special modes, user mode returns a logic zero

WRITE: Only in special modes

- 1 = Enables the decode to the MOR register latch. This feature makes production testing and emulation easier. It is only valid when IOOFF is cleared to zero.
- 0 = The MOR register is not in the map; INTSELB is not pulled at \$1F00.

EPTST — EPROM Test Mode Enable

READ: Only in special modes, user mode returns a logic zero

WRITE: Only in special modes

- 1 = Enables the functions of TS1 and TS0.
- 0 = TS1 and TS0 functions are disabled.

MBE — Multiple Byte Program Enable

READ: Only in special modes, user mode returns a logic zero

WRITE: Only in special modes

- 1 = Enable the multibyte programming, two bytes at a time. address bit 4 is don't care so that the bytes with addresses 5 = 0 and 5 = 1 are both programmed.
- 0 = Normal programming

TS1 and TS0 — EPROM Test Select Bits

READ: Only in special modes, user mode returns a logic zero

WRITE: Only in special modes

LATCH — EPROM Latch Control

READ: Any time

WRITE: Any time

0 = EPROM address and data bus configured for normal reads

1 = EPROM address and data bus configured for programming. Causes address and data bus to be latched when a write to EPROM is done. EPROM cannot be read if LATCH = 1.

EPGM — EPROM Program Control

READ: Any time

WRITE: Cleared any time, set only if LATCH = 1

0 = Programming power switched off the EPROM array

1 = Programming power switched on to the EPROM array. If LATCH = 0, then EPGM is cleared automatically. EPGM cannot be set if LATCH is not set. LATCH and EPGM cannot both be set on the same write.

A.5.3 Mask Option Register (MOR)

The mask option register, MOR, contains programmable EPROM bits to control mask options.

The MOR register is latched upon reset going away.

	7	6	5	4	3	2	1	0
\$1F00	PC7PU	PC6PU	PC5PU	PC4PU	PC23PU	PC01PU	IRQ	COP

Figure A-6. Mask Option Register

PCXPU - Port C Pullup (X is 7-4)

When set, the PCPU bit enables the pullup on the corresponding port C pin. If the PCPU bit is cleared, the pullup devices are disabled. The erased state of the PCPU bit is to be cleared, thereby disabling the pullup devices.

PC23PU - Port C Bit 2 and 3 Pullups

When set, the PC23PU bit enables the pullups on both port C bits 2 and 3 pins. If the PC23PU bit is cleared, the pullup devices are disabled. The erased state of the PC23PU bit is to be cleared, thereby disabling the pullup devices.

PC01PU - Port C Bit 0 and 1 Pullups

When set, the PC01PU bit enables the pullups on both port C bits 0 and 1 pins. If the PC01PU bit is cleared, the pullup devices are disabled. The erased state of the PC01PU bit is to be cleared, thereby disabling the pullup devices.

COP — COP Enable

When set, this bit enables the COP watchdog timer.
When clear, the COP is disabled.

IRQ — IRQ sensitivity

When set, this bit selects the edge- and level-sensitive IRQ.
When clear, it selects the edge-only-sensitive IRQ.

In test mode, reading and writing the MOR is done according to Table A-3.

Table A-2. MOR Read/Write Based on Mode and LATCH Bit

LATCH	TEST	READ	WRITE
0	0	E	N
1	0	N	PE
0	1	E	R
1	1	R	PE

PE = Program EPROM Byte
R = Register Latch
N = Nothing
E = EPROM Byte

A.6 EPROM ELECTRICAL SPECIFICATIONS

Except for the following, the electrical specifications for the MC68HC705CT4 are the same as those shown in **SECTION 15 ELECTRICAL SPECIFICATIONS** for the MC68HC05CT4.

A.6.1 Maximum Ratings

(Voltages referenced to V_{SS})

Rating	Symbol	Value	Unit
Bootloader Mode (\overline{IRQ} Pin Only)	V_{IN}	$V_{SS} - 0.3$ to $2 \times V_{DD} + 0.3$	V

A.6.2 DC Electrical Characteristics (5.0 V)

($V_{DD} = 5.0 \text{ Vdc} \pm 10\%$, $V_{SS} = 0 \text{ Vdc}$, $T_A = 0^\circ\text{C}$ to $+70^\circ\text{C}$, unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit
EPROM Programming Voltage	V	14.5	—	15.5	V
Supply Current (see Notes)					
Run	I_{DD}	—	TBD	TBD	mA
Wait	I_{DD}	—	TBD	TBD	mA
Stop					
25 °C	I_{DD}	—	TBD	TBD	μA
0 °C to +70 °C	I_{DD}	—	TBD	TBD	μA
Supply Current (see Notes)					
Run	I_{DD2}	—	TBD	TBD	mA
Wait	I_{DD2}	—	TBD	TBD	mA
Stop					
25 °C	I_{DD2}	—	TBD	TBD	μA
0 °C to +70 °C	I_{DD2}	—	TBD	TBD	μA

NOTES:

1. All values shown reflect average measurements.
2. Typical values at midpoint of voltage range, 25°C only.
3. Wait I_{DD} and I_{DD2} : Only timer system active.
4. Run (Operating) I_{DD} , I_{DD2} , Wait I_{DD} , and I_{DD2} : Measured using external square wave clock source ($f_{OSC} = 10.24 \text{ MHz}$), all inputs 0.2 V from rail; no dc loads, less than 50 pF on all outputs, $C_L = 20 \text{ pF}$ on OSC2.
5. Wait, Stop I_{DD} : All ports configured as inputs, $V_{IL} = 0.2 \text{ V}$, $V_{IH} = V_{DD} - 0.2 \text{ V}$.
6. Stop I_{DD} and I_{DD2} are measured with $OSC1 = V_{SS}$.
7. Wait I_{DD} and I_{DD2} are affected linearly by the OSC2 capacitance.
8. Run, Wait, and Stop I_{DD2} are measured directly off the V_{DD2} power supply.

A.6.3 DC Electrical Characteristics (3.3 V)(V_{DD} = 3.3 Vdc ±10%, V_{SS} = 0 Vdc, T_A = 0 °C to +70 °C, unless otherwise noted)


Characteristic	Symbol	Min	Typ	Max	Unit
EPROM Programming Voltage	V	14.5	—	15.5	V
Supply Current (see Notes)					
Run	I _{DD}	—	TBD	TBD	mA
Wait	I _{DD}	—	TBD	TBD	mA
Stop					
25 °C	I _{DD}	—	TBD	TBD	μA
0 °C to +70 °C	I _{DD}	—	TBD	TBD	μA
Supply Current (see Notes)					
Run	I _{DD2}	—	TBD	TBD	mA
Wait	I _{DD2}	—	TBD	TBD	mA
Stop					
25 °C	I _{DD2}	—	TBD	TBD	μA
0 °C to +70 °C	I _{DD2}	—	TBD	TBD	μA

NOTES:

1. All values shown reflect average measurements.
2. Typical values at midpoint of voltage range, 25°C only.
3. Wait I_{DD} and I_{DD2} : Only timer system active.
4. Run (Operating) I_{DD}, I_{DD2}, Wait I_{DD}, and I_{DD2}: Measured using external square wave clock source (f_{OSC} = 10.24 MHz), all inputs 0.2 V from rail; no dc loads, less than 50 pF on all outputs, C_L = 20 pF on OSC2.
5. Wait, Stop I_{DD}: All ports configured as inputs, V_{IL} = 0.2 V, V_{IH} = V_{DD} - 0.2 V.
6. Stop I_{DD} and I_{DD2} are measured with OSC1 = V_{SS}.
7. Wait I_{DD} and I_{DD2} are affected linearly by the OSC2 capacitance.
8. Run, Wait, and Stop I_{DD2} are measured directly off the V_{DD2} power supply.

A.6.4 EPROM Control Timing (3.3 V and 5.0 V)

Characteristic	Symbol	Min	Max	Unit
EPROM Byte Programming Time	t _{EPGM}	—	10.0	ms

Motorola reserves the right to make changes without further notice to any products herein. Motorola makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Motorola assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters can and do vary in different applications. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts. Motorola does not convey any license under its patent rights nor the rights of others. Motorola products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Motorola product could create a situation where personal injury or death may occur. Should Buyer purchase or use Motorola products for any such unintended or unauthorized application, Buyer shall indemnify and hold Motorola and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Motorola was negligent regarding the design or manufacture of the part. Motorola and  are registered trademarks of Motorola, Inc. Motorola, Inc. is an Equal Opportunity/Affirmative Action Employer.

Literature Distribution Centers:

USA: Motorola Literature Distribution; P.O. Box 20912; Phoenix, Arizona 85036.

EUROPE: Motorola Ltd.; European Literature Centre; 88 Tanners Drive, Blakelands, Milton Keynes, MK14 5BP, England.

JAPAN: Nippon Motorola Ltd.; 4-32-1, Nishi-Gotanda, Shinagawa-ku, Tokyo 141 Japan.

ASIA-PACIFIC: Motorola Semiconductors H.K. Ltd.; Silicon Harbour Center, No.2 Dai King Street, Tai Po Industrial Estate, Tai Po, N.T., Hong Kong.



MOTOROLA

HC05CT4GRS/D

