

68HC05P5

SPECIFICATION (General Release)

© November 25, 1996

CSIC System Design Group
Austin, Texas



TABLE OF CONTENTS

| | | |
|------------------|--------------------------------------|-----------|
| SECTION 1 | INTRODUCTION..... | 1 |
| 1.1 | GENERAL | 1 |
| 1.2 | FEATURES | 1 |
| 1.3 | MASK OPTIONS..... | 2 |
| 1.4 | SIGNAL DESCRIPTION | 3 |
| 1.4.1 | V_{DD} AND V_{SS} | 3 |
| 1.4.2 | IRQ | 3 |
| 1.4.3 | OSC1, OSC2 | 3 |
| 1.4.4 | RESET | 3 |
| 1.4.5 | TCMP..... | 3 |
| 1.4.6 | PA0-PA7 | 3 |
| 1.4.7 | PB7/CMPO, PB6/CMP+, PB5/CMP | 4 |
| 1.4.8 | PC0-PC6..... | 4 |
| 1.4.9 | TCAP/PD7 | 4 |
| SECTION 2 | OPERATING MODES..... | 5 |
| 2.1 | USER MODE | 5 |
| 2.2 | SELF-CHECK MODE..... | 6 |
| SECTION 3 | MEMORY | 9 |
| 3.1 | ROM..... | 11 |
| 3.2 | RAM | 11 |
| SECTION 4 | CPU CORE | 13 |
| 4.1 | REGISTERS | 13 |
| 4.1.1 | ACCUMULATOR (A) | 13 |
| 4.1.2 | INDEX REGISTER (X)..... | 13 |
| 4.1.3 | CONDITION CODE REGISTER (CCR)..... | 13 |
| 4.1.3.1 | H - HALF CARRY | 13 |
| 4.1.3.2 | I - INTERRUPT..... | 13 |
| 4.1.3.3 | N - NEGATIVE | 13 |
| 4.1.3.4 | Z - ZERO | 14 |
| 4.1.3.5 | C - CARRY/BORROW | 14 |
| 4.1.4 | STACK POINTER (SP)..... | 14 |
| 4.1.5 | PROGRAM COUNTER (PC) | 14 |
| 4.2 | INSTRUCTION SET..... | 14 |
| 4.2.1 | REGISTER/MEMORY INSTRUCTIONS | 16 |
| 4.2.2 | READ-MODIFY-WRITE INSTRUCTIONS | 17 |
| 4.2.3 | BRANCH INSTRUCTIONS..... | 18 |
| 4.2.4 | BIT MANIPULATION INSTRUCTIONS | 19 |
| 4.2.5 | CONTROL INSTRUCTIONS | 19 |
| 4.3 | ADDRESSING MODES | 20 |

| | | |
|------------------|---|-----------|
| 4.3.1 | IMMEDIATE | 20 |
| 4.3.2 | DIRECT | 20 |
| 4.3.3 | EXTENDED | 20 |
| 4.3.4 | RELATIVE | 20 |
| 4.3.5 | INDEXED, NO OFFSET | 21 |
| 4.3.6 | INDEXED, 8-BIT OFFSET | 21 |
| 4.3.7 | INDEXED, 16-BIT OFFSET | 21 |
| 4.3.8 | BIT SET/CLEAR | 21 |
| 4.3.9 | BIT TEST AND BRANCH | 21 |
| 4.3.10 | INHERENT | 22 |
| 4.4 | RESETS | 22 |
| 4.4.1 | POWER-ON RESET (POR) | 22 |
| 4.4.2 | RESET PIN | 23 |
| 4.4.3 | COMPUTER OPERATING PROPERLY (COP) RESET ... | 23 |
| 4.4.4 | ILLEGAL ADDRESS RESET | 23 |
| 4.5 | INTERRUPTS | 23 |
| 4.5.1 | HARDWARE CONTROLLED INTERRUPT SEQUENCE .. | 24 |
| 4.5.2 | TIMER INTERRUPT | 25 |
| 4.5.3 | EXTERNAL INTERRUPT | 26 |
| 4.5.4 | SOFTWARE INTERRUPT (SWI) | 26 |
| 4.6 | LOW-POWER MODES | 26 |
| 4.6.1 | STOP | 26 |
| 4.6.2 | WAIT | 26 |
| SECTION 5 | TIMER | 29 |
| 5.1 | INTRODUCTION | 29 |
| 5.2 | COUNTER | 30 |
| 5.3 | OUTPUT COMPARE REGISTER | 31 |
| 5.4 | INPUT CAPTURE REGISTER | 31 |
| 5.5 | TIMER CONTROL REGISTER (TCR) \$12 | 32 |
| 5.6 | TIMER STATUS REGISTER (TSR) \$13 | 33 |
| 5.7 | TIMER DURING WAIT MODE | 34 |
| 5.8 | TIMER DURING STOP MODE | 34 |
| SECTION 6 | COMPUTER OPERATING PROPERLY | 35 |
| 6.1 | INTRODUCTION | 35 |
| 6.2 | RESETTING THE COP | 35 |
| 6.3 | COP DURING WAIT MODE | 36 |
| 6.4 | COP DURING STOP MODE | 36 |
| 6.5 | COP DURING SELF-CHECK MODE | 36 |

| | | |
|-------------------|---|-----------|
| SECTION 7 | INPUT/OUTPUT PORTS | 37 |
| 7.1 | PORT A..... | 37 |
| 7.2 | PORT B..... | 37 |
| 7.3 | PORT C..... | 37 |
| 7.4 | PORT D..... | 37 |
| 7.5 | INPUT/OUTPUT PROGRAMMING..... | 37 |
| SECTION 8 | PULSE WIDTH MODULATOR(PWM) | 39 |
| 8.1 | PWM INTRODUCTION..... | 39 |
| 8.2 | FUNCTIONAL DESCRIPTION..... | 39 |
| 8.3 | REGISTERS | 41 |
| 8.3.1 | PWM CONTROL \$0E..... | 41 |
| 8.3.2 | POL1 - PWM1 POLARITY | 41 |
| 8.3.3 | POL0 - PWM0 POLARITY | 41 |
| 8.3.4 | PWM DATA REGISTERS..... | 41 |
| 8.4 | PWM DURING WAIT MODE | 42 |
| 8.5 | PWM DURING STOP MODE..... | 42 |
| 8.6 | PWM DURING RESET | 42 |
| SECTION 9 | COMPARATOR | 43 |
| 9.1 | COMPARATOR CONTROL REGISTER | 43 |
| 9.2 | READING COMPARATOR OUTPUT | 43 |
| 9.3 | COMPARATOR DURING WAIT | 43 |
| 9.4 | COMPARATOR DURING STOP | 44 |
| SECTION 10 | ELECTRICAL SPECIFICATIONS | 45 |
| 10.1 | MAXIMUM RATINGS..... | 45 |
| 10.2 | THERMAL CHARACTERISTICS | 45 |
| 10.3 | DC ELECTRICAL CHARACTERISTICS (5.0 Vdc) | 46 |
| 10.4 | DC ELECTRICAL CHARACTERISTICS (3.0 Vdc) | 47 |
| 10.4 | CONTROL TIMING (5.0 Vdc) | 48 |
| 10.5 | CONTROL TIMING (3.0 Vdc) | 49 |
| 10.5 | COMPARATOR | 50 |
| 10.6 | PWM TIMING..... | 50 |
| SECTION 11 | MECHANICAL SPECIFICATIONS..... | 51 |
| 11.1 | 28-PIN PLASTIC DUAL IN-LINE PACKAGE | 51 |
| 11.2 | 28-PIN SMALL OUTLINE INTEGRATED CIRCUIT PACKAGE | 51 |

| | | |
|-------------------|------------------------------------|-----------|
| SECTION 12 | ORDERING INFORMATION | 53 |
| 12.1 | INTRODUCTION..... | 53 |
| 12.2 | MCU ORDERING FORMS..... | 53 |
| 12.3 | APPLICATION PROGRAM MEDIA | 53 |
| 12.4 | ROM VERIFICATION..... | 55 |
| 12.5 | ROM VERIFICATION UNITS (RVUs)..... | 55 |
| 12.6 | MCU ORDER NUMBERS | 55 |

LIST OF FIGURES

| | | |
|-------------|--|----|
| Figure 1-1: | Block Diagram for the MC68HC05P5 | 2 |
| Figure 2-1: | User Mode Pinout of the MC68HC05P5 | 5 |
| Figure 2-2: | Self-Check Circuit Schematic Diagram | 6 |
| Figure 3-1: | The 8K Memory Map of the MC68HC05P5 | 9 |
| Figure 3-2: | I/O Registers | 10 |
| Figure 4-1: | Hardware Interrupt Flowchart | 25 |
| Figure 4-2: | STOP/WAIT Flowcharts | 27 |
| Figure 5-1: | Timer Block Diagram | 30 |
| Figure 5-2: | Timer Control Register | 32 |
| Figure 5-3: | Timer Status Register | 33 |
| Figure 6-1: | COP Block Diagram | 35 |
| Figure 7-1: | Port I/O Circuitry | 38 |
| Figure 8-1: | PWM Block Diagram | 39 |
| Figure 8-2: | PWM Waveforms (POL = 1) | 40 |
| Figure 8-3: | PWM Waveforms (POL = 0) | 40 |
| Figure 8-4: | PWM Control Register | 41 |
| Figure 8-5: | PWM Data Registers | 42 |
| Figure 9-1: | Comparator Control Register | 43 |
| Figure 9-2: | Port B Comparator Circuitry | 44 |

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

| | | |
|------------|--|----|
| Table 2-1: | Operating Mode Conditions | 5 |
| Table 2-2: | Self-Check Results | 7 |
| Table 4-1: | Vector Address for Interrupts and Reset..... | 24 |
| Table 7-1: | I/O Pin Functions | 38 |
| Table 8-1: | PWM Clock Rate..... | 41 |

THIS PAGE INTENTIONALLY LEFT BLANK

SECTION 1

INTRODUCTION

1.1 GENERAL

The MC68HC05P5 is a 28-pin device based on the MC68HC05P7 with an on-chip analog voltage comparator and a two-channel pulse width modulator (PWM). The memory map includes 3136 bytes of user ROM, and 128 bytes of RAM. The MCU has two I/O ports, A and C, consisting of 15 pins with mask programmable pullup devices. Port B consists of 3 pins, 2 input only and 1 output only, shared with an analog voltage comparator. PWM circuitry does not share the output pins with any other module. Port D has 1 input only pin. The MC68HC05P5 has a 16-bit timer, a mask programmable Computer Operating Properly (COP) Watchdog, a mask programmable STOP disable option, and Illegal Address Reset (ILADR) circuitry to ensure proper functioning of any system.

1.2 FEATURES

- Low cost
- Low noise
- HC05 Core
- 28-pin package
- On-Chip Oscillator with RC or Crystal/Ceramic Resonator Mask Options
- 3136 Bytes of User ROM including 16 user vector locations
- 128 Bytes of On-Chip RAM
- 16-Bit Timer
- 15 Bidirectional I/O Lines, 3 input-only lines, 3 output-only lines
- User Mode and Self-Check Mode
- Power Saving STOP and WAIT Modes
- Edge-Sensitive or Edge- and Level-Sensitive Interrupt Trigger Mask Option
- Port Pullup Mask Option for Ports A and C
- Mask option selectable Watchdog Timer (COP)
- Illegal Address Reset (ILADR)
- STOP disable Mask Option
- Internal Resistor and Capacitors for Oscillator Circuitry

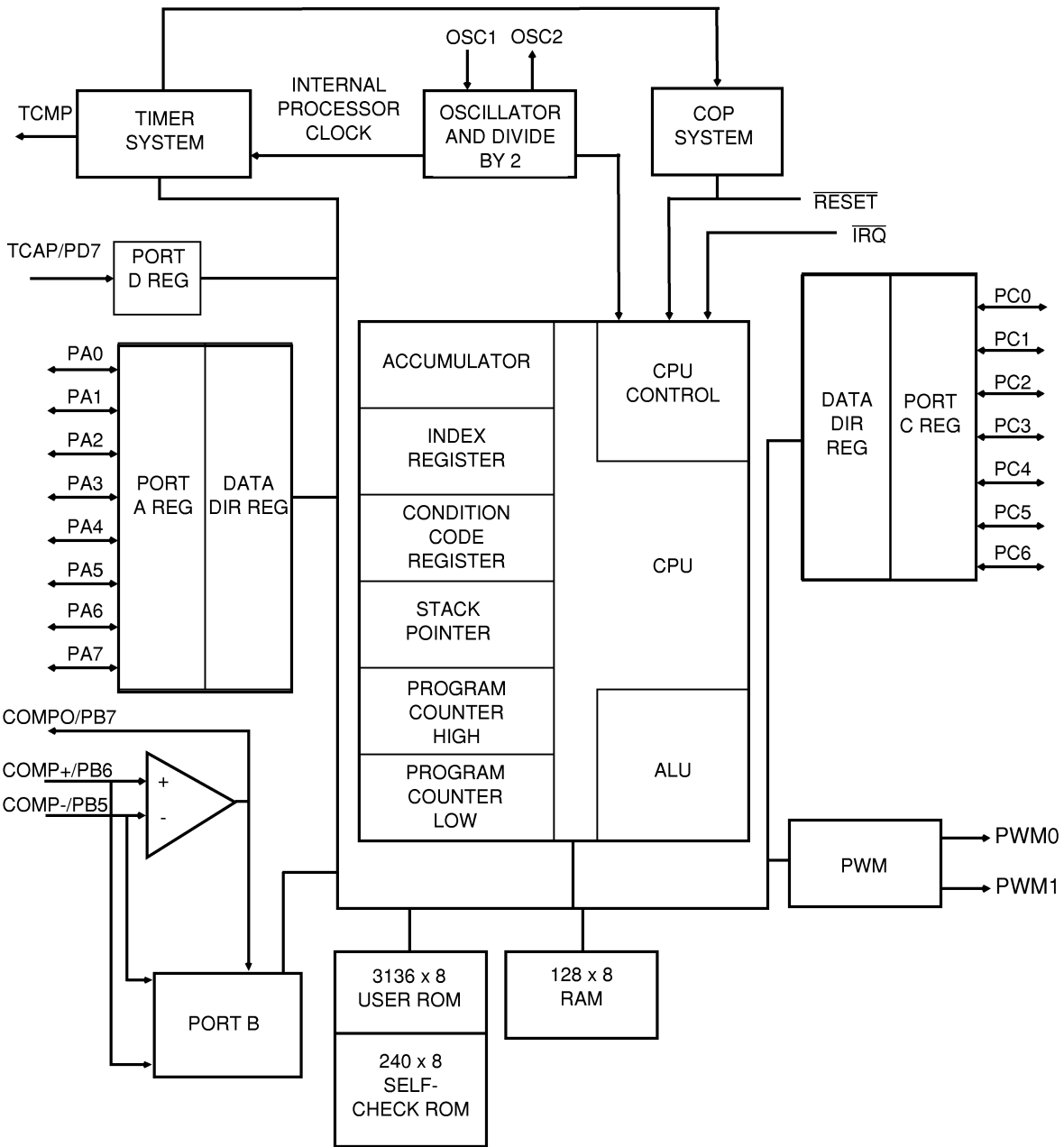


Figure 1-1: Block Diagram for the MC68HC05P5

1.3 MASK OPTIONS

Mask options on the MC68HC05P5 include: CLOCK (RC or Crystal), internal resistor, capacitors for crystal (enable/disable), \overline{IRQ} (edge-sensitive only or edge- and level-sensitive), the COP Watchdog Timer (enable/disable), STOP instruction disable, and pullup devices (enable/disable) for ports A and C.

1.4 SIGNAL DESCRIPTION

1.4.1 V_{DD} AND V_{SS}

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

1.4.2 \overline{IRQ}

This pin has a mask option that provides two different choices of interrupt triggering sensitivity. The \overline{IRQ} pin contains an internal Schmitt trigger as part of its input to improve noise immunity. Refer to **4.5 INTERRUPTS** for more detail.

1.4.3 OSC1, OSC2

These pins provide control input for an on-chip clock oscillator circuit. A crystal, a ceramic resonator, a resistor/capacitor combination, or an external signal connects to these pins providing a system clock. A mask option selects either a crystal/ceramic resonator or a resistor/capacitor as the frequency determining element. The oscillator frequency is two times the internal bus rate. An option to include a 1 M Ω resistor (internally across OSC1 and OSC2) and two 22 pF capacitors (one capacitor from OSC1 to ground and the other from OSC2 to ground) is available if the crystal/ceramic resonator option is chosen. These internal components are disconnected if the RC option is chosen over the crystal option.

1.4.4 RESET

This active low pin is used to reset the MCU to a known start-up state by pulling \overline{RESET} low. The \overline{RESET} pin contains an internal Schmitt trigger as part of its input to improve noise immunity.

1.4.5 TCMP

This pin provides an output for the output compare feature of the on-chip timer system.

1.4.6 PA0-PA7

Port A is an 8-bit bidirectional port. The address of the port A data register is \$0000 and the address of the data direction register (DDR) is \$0004. Reset does not affect the data register, but clears the data direction register, thereby returning the port pins to inputs. Writing a one to a DDR bit sets the corresponding port bit to output mode. Port A has Mask Options to enable/disable the pullups on the port. The pullups are disconnected when the pins are outputs. Refer to **7.5 INPUT/OUTPUT PROGRAMMING** for a detailed description of I/O programming. See **SECTION 10 ELECTRICAL SPECIFICATIONS** for details on pullup devices.

1.4.7 PB7/CMPO, PB6/CMP+, PB5/CMP

Port B is a 3-bit port that is shared with the comparator. PB7 is output only, PB5 and PB6 are input only. PB5, PB6, and PB7 are readable internally. The address of the port B data register is \$0001. Refer to **7.5 INPUT/OUTPUT PROGRAMMING** for a detailed description of I/O programming. When the comparator module is enabled, PB7/CMPO is readable internally at \$0001 bit 7. See **SECTION 9 COMPARATOR** for further information on the comparator.

1.4.8 PC0-PC6

Port C is a 7-bit bidirectional port. The address of the port C data register is \$0002 and the address of the data direction register is \$0006. Reset does not affect the data register, but clears the data direction register, thereby returning the ports to inputs. Writing a one to a DDR bit sets the corresponding port bit to output mode. Port C has a Mask Options to enable/disable the pullups on the port. The pullups are disconnected when the pins are outputs. Refer to **7.5 INPUT/OUTPUT PROGRAMMING** for a detailed description of I/O programming. See **SECTION 10 ELECTRICAL SPECIFICATIONS** for details on pullup devices.

1.4.9 TCAP/PD7

Port D is a 1-bit port. TCAP/PD7 is input-only shared with the timer input capture. The address of the port D data register is \$0003. Reset does not affect the data registers. The TCAP/PD7 pin controls the input capture feature for the on-chip programmable timer. This pin can be read at any time even if the TCAP function is enabled. For detailed information on the TCAP pin, see **5.4 INPUT CAPTURE REGISTER**.

SECTION 2

OPERATING MODES

The MCU has two modes of operation: User Mode mode and Self-Check Mode. **Table 2-1: Operating Mode Conditions** shows the conditions required to go into each mode, where $V_{TST} = 2 \times V_{DD}$.

Table 2-1: Operating Mode Conditions

| RESET | IRQ | TCAP | MODE |
|-------|----------------------|----------------------|------------|
| | V_{SS} to V_{DD} | V_{SS} to V_{DD} | User Mode |
| | V_{TST} | V_{DD} | Self-Check |

2.1 USER MODE

In single-chip mode, the address and data buses are not available externally, but there are two I/O ports (A & C), one 3-bit port (B), and one 1-bit port (D). 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. User Mode is entered on the rising edge of \overline{RESET} if the \overline{IRQ} pin is within normal operating range.

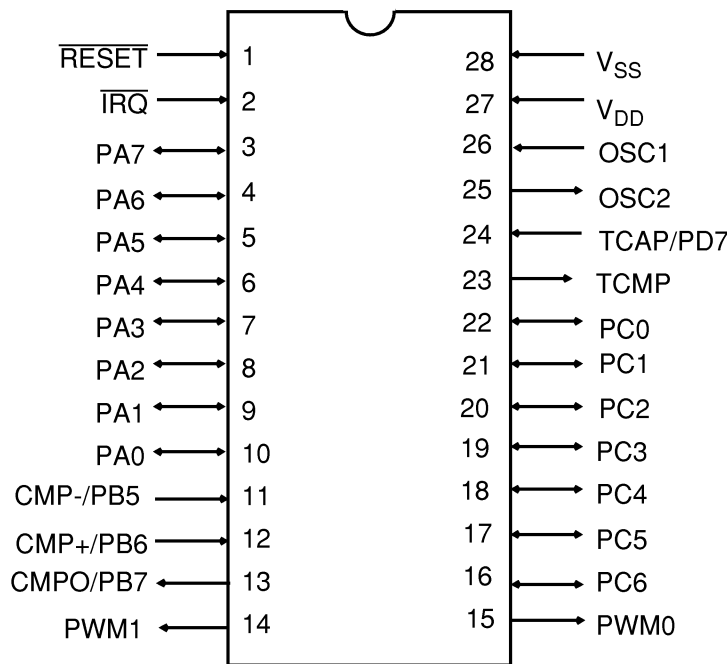


Figure 2-1: User Mode Pinout of the MC68HC05P5

2.2 SELF-CHECK MODE

The Self-Check Mode provides an internal check to determine if the device is functional. See **Figure 2-2: Self-Check Circuit Schematic Diagram**.

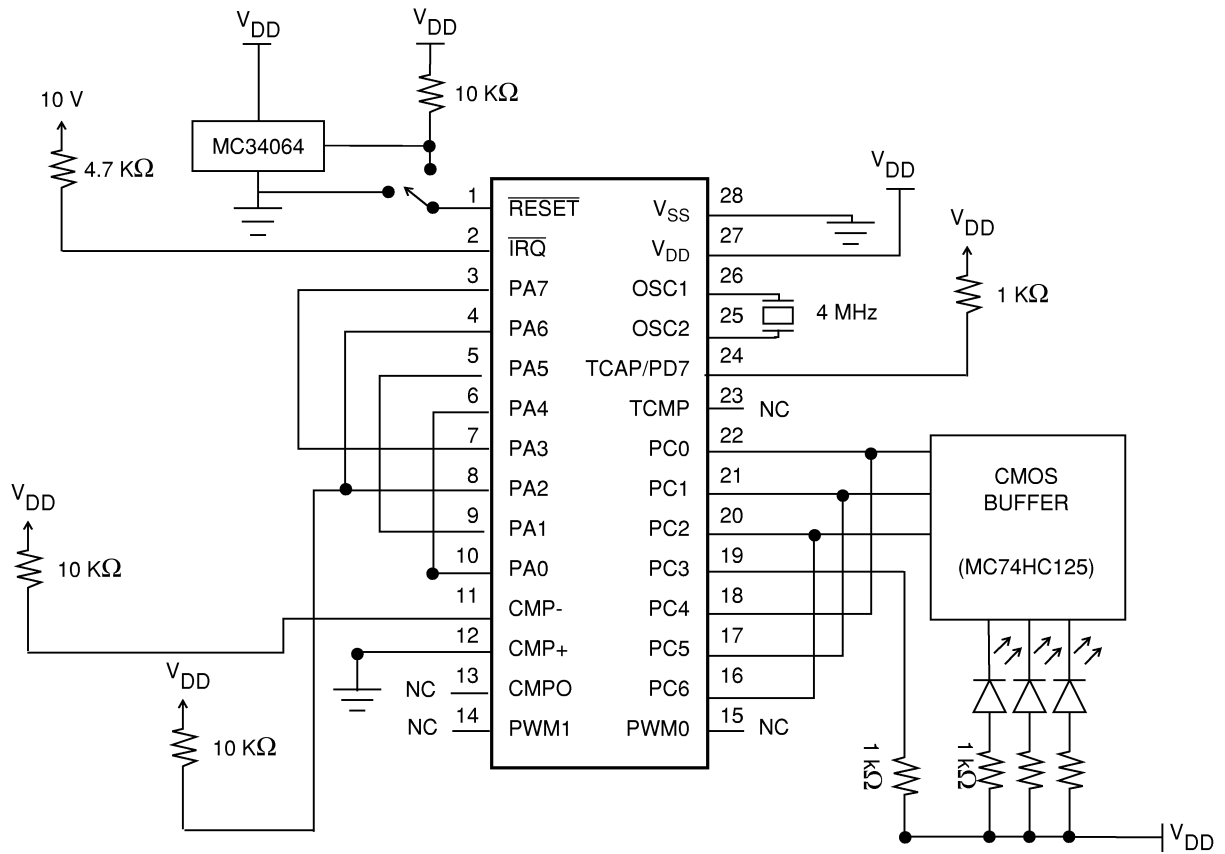


Figure 2-2: Self-Check Circuit Schematic Diagram

The Self-Check Mode is entered on the rising edge of $\overline{\text{RESET}}$ if the $\overline{\text{IRQ}}$ pin is at V_{TST} and the TCAP pin is at logic one. $\overline{\text{RESET}}$ must be held low for 4064 cycles after POR, or for a time t_{RL} for any other reset. After reset, the following tests are performed automatically:

1. I/O – Functionally exercises ports A and C
2. RAM – Counter test for each page zero RAM byte
3. Timer – Tracks counter register and checks OCF flag
4. ROM – Exclusive OR with odd ones parity result

Self-check results (using the LEDs as monitors) are shown in **Table 2-2: Self-Check Results**. The Self-Check program resides at ROM location \$1F00 to \$1FEF.

Table 2-2: Self-Check Results

| PC2 | PC1 | PC0 | Remarks |
|------------|-----|-----|----------------|
| 0 | 0 | 1 | Bad I/O |
| 0 | 1 | 0 | Bad RAM |
| 0 | 1 | 1 | Bad Timer |
| 1 | 0 | 0 | Bad ROM |
| 1 | 1 | 0 | Bad Comparator |
| Flashing | | | Good Device |
| All Others | | | Bad Device |

0 indicates LED is on; 1 indicates LED is off.

THIS PAGE INTENTIONALLY LEFT BLANK

SECTION 3

MEMORY

The MC68HC05P5 has a 8 Kbyte memory map, consisting of user ROM, user RAM, Self-Check ROM, and I/O. See **Figure 3-1: The 8K Memory Map of the MC68HC05P5** and **Figure 3-2: I/O Registers**.

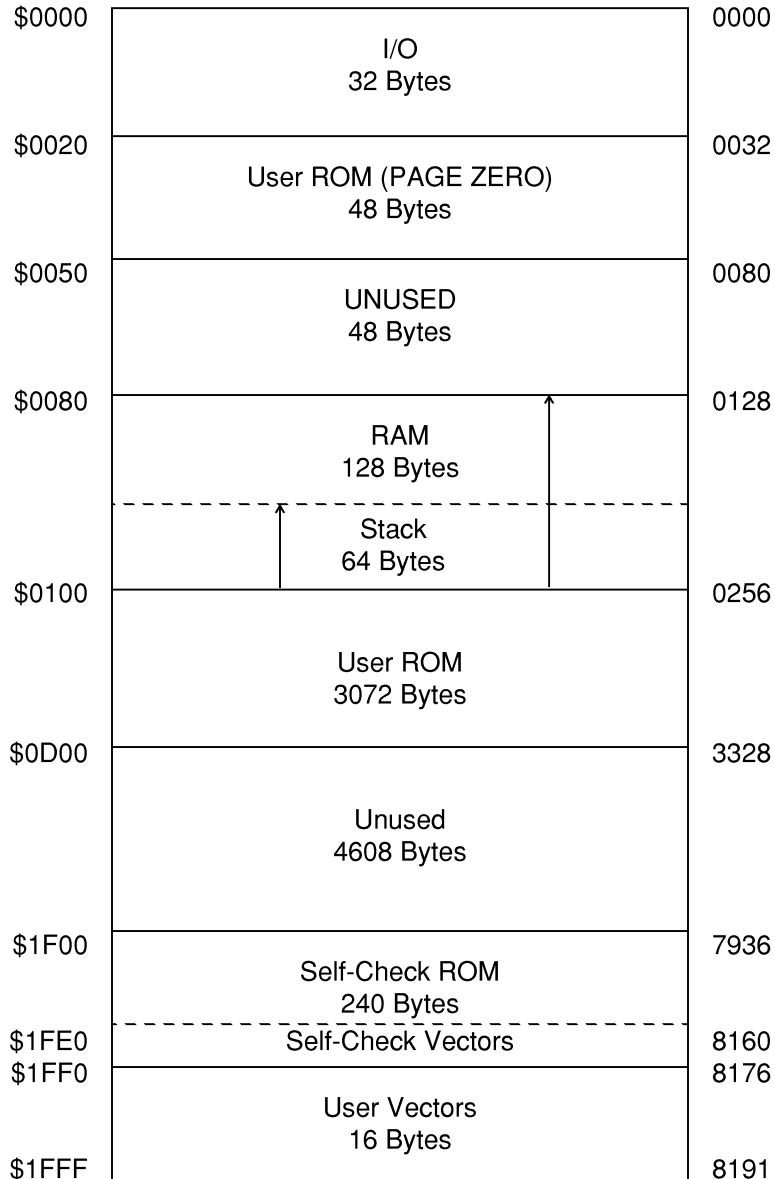


Figure 3-1: The 8K Memory Map of the MC68HC05P5

| ADDRESS 0000 TO 001F | DATA | | | | | | | |
|-------------------------|------|-------|-------|-------|-------|-------|------|-------|
| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| \$00 PORT A DATA | | | | | | | | |
| \$01 PORT B DATA | | | | 0 | 0 | 0 | 0 | 0 |
| \$02 PORT C DATA | 0 | | | | | | | |
| \$03 PORT D DATA | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| \$04 PORT A DDR | | | | | | | | |
| \$05 UNUSED | -- | -- | -- | -- | -- | -- | -- | -- |
| \$06 PORT C DDR | 0 | | | | | | | |
| \$07 UNUSED | -- | -- | -- | -- | -- | -- | -- | -- |
| \$08 UNUSED | -- | -- | -- | -- | -- | -- | -- | -- |
| \$09 UNUSED | -- | -- | -- | -- | -- | -- | -- | -- |
| \$0A UNUSED | -- | -- | -- | -- | -- | -- | -- | -- |
| \$0B UNUSED | -- | - | -- | -- | -- | -- | -- | -- |
| \$0C COMP CONTROL | -- | -- | -- | -- | -- | -- | -- | CEN0 |
| \$0D UNUSED | -- | -- | -- | -- | -- | -- | -- | -- |
| \$0E PWM CONTROL | -- | -- | -- | -- | POL1 | POL0 | RA1 | RA0 |
| \$0F PWM DATA 0 | | | | | | | | |
| \$10 PWM DATA 1 | | | | | | | | |
| \$11 UNUSED | -- | -- | -- | -- | -- | -- | -- | -- |
| \$12 TIMER CONTROL | ICIE | OCIE | TOIE | -- | -- | -- | IEDG | OLVL |
| \$13 TIMER STATUS | ICF | OCF | TOF | -- | -- | -- | -- | -- |
| \$14 CAPTURE HIGH | | | | | | | | |
| \$15 CAPTURE LOW | | | | | | | | |
| \$16 COMPARE HIGH | | | | | | | | |
| \$17 COMPARE LOW | | | | | | | | |
| \$18 COUNTER HIGH | | | | | | | | |
| \$19 COUNTER LOW | | | | | | | | |
| \$1A DUAL TM HIGH | | | | | | | | |
| \$1B DUAL TM LOW | | | | | | | | |
| \$1C UNUSED | -- | -- | -- | -- | -- | -- | -- | -- |
| \$1D UNUSED | -- | -- | -- | -- | -- | -- | -- | -- |
| \$1E UNUSED | -- | -- | -- | -- | -- | -- | -- | -- |
| \$1F TEST REGISTER | -- | MSCAN | ROMON | IOOFF | ILADR | COPEN | TCNT | RAMON |

Figure 3-2: I/O Registers

3.1 ROM

The user ROM consists of 48 bytes of page zero ROM from \$0020 to \$004F, 3072 bytes of ROM from \$0100 to \$0CFF, and 16 bytes of user vectors from \$1FF0 to \$1FFF. The Self-Check ROM and vectors are located from \$1F00 to \$1FEF.

3.2 RAM

The user RAM consists of 128 bytes with a shared stack area. 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.

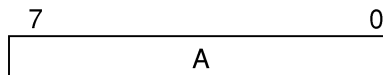
THIS PAGE INTENTIONALLY LEFT BLANK

SECTION 4**CPU CORE****4.1 REGISTERS**

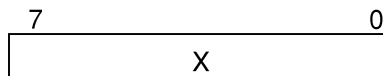
The MCU contains the registers described in the following paragraphs.

4.1.1 ACCUMULATOR (A)

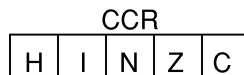
The accumulator is a general purpose 8-bit register used to hold operands and results of arithmetic calculations or data manipulations.

**4.1.2 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.

**4.1.3 CONDITION CODE REGISTER (CCR)**

The CCR is a 5-bit register in which 4 bits are used to indicate the results of the instruction just executed, and the fifth bit indicates whether interrupts are masked. These bits can be individually tested by a program, and specific actions can be taken as a result of their state. Each bit is explained in the following paragraphs.

**4.1.3.1 H - HALF CARRY**

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

4.1.3.2 I - INTERRUPT

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

4.1.3.3 N - NEGATIVE

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

4.1.3.4 Z - ZERO

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

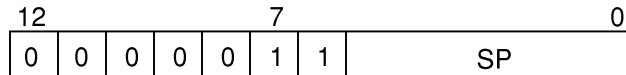
4.1.3.5 C - CARRY/BORROW

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

4.1.4 STACK POINTER (SP)

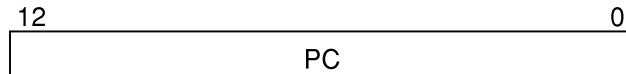
The stack pointer contains the address of the next free location on the stack. During an MCU reset or the reset stack pointer (RSP) instruction, the stack pointer is set to location \$00FF. The stack pointer is then decremented as data is pushed onto the stack and incremented as data is pulled from the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

When accessing memory, the 7 most significant bits are permanently set to 000011. These 7 bits are appended to the 6 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.



4.1.5 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 a 64 Kbyte memory map. For this implementation, however, the addressing registers are limited to an 8 Kbyte memory map.

4.2 INSTRUCTION SET

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

This MCU uses all the instructions available in the M146805 CMOS Family plus one more: the unsigned multiply (MUL) instruction. This instruction allows unsigned multiplication of the contents of the accumulator (A) and the index register (X). The high-order product is then stored in the index register, and the low-order product is stored in the accumulator. A detailed definition of the MUL instruction is shown below.

| | | | | |
|------------------------|---|--------------|------------|----------------|
| Operation | X:A X*A | | | |
| Description | Multiplies the 8 bits in the index register by the 8 bits in the accumulator to obtain a 16-bit unsigned number in the concatenated accumulator and index register. | | | |
| Condition Codes | H: Cleared I: Not affected N: Not affected Z: Not affected C: Cleared | | | |
| Source | MUL | | | |
| Form(s) | Addressing Mode Inherent | Cycles 11 | Bytes 1 | Opcode \$42 |

4.2.1 REGISTER/MEMORY INSTRUCTIONS

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

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

4.2.2 READ-MODIFY-WRITE INSTRUCTIONS

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

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

4.2.3 BRANCH INSTRUCTIONS

This set of instruction branches if a particular condition is met; otherwise, no operation is performed. Branch instructions are 2-byte instructions. Refer to the following list for branch instructions.

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

4.2.4 BIT MANIPULATION INSTRUCTIONS

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

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

4.2.5 CONTROL INSTRUCTIONS

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

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

4.3 ADDRESSING MODES

The MCU uses ten different addressing modes to provide the programmer with an opportunity to optimize the code for all situations. The various indexed addressing modes make it possible to locate data tables, code conversion tables, and scaling tables anywhere in the memory space. Short indexed accesses are single-byte instructions; the longest instructions (3 bytes) permit accessing tables throughout memory. Short and long absolute addressing is also included. One- or 2-byte direct addressing instructions access all data bytes in most applications. Extended addressing permits jump instructions to reach all memory.

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

4.3.1 IMMEDIATE

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

4.3.2 DIRECT

In the direct addressing mode, the effective address of the argument is contained in a single byte following the opcode byte. Direct addressing allows the user to directly address the lowest 256 bytes in memory with a single 2-byte instruction.

4.3.3 EXTENDED

In the extended addressing mode, the effective address of the argument is contained in the 2 bytes following the opcode byte. Instructions with extended addressing mode are capable of referencing arguments anywhere in memory with a single 3-byte instruction. When using the Motorola assembler, the user need not specify whether an instruction uses direct or extended addressing. The assembler automatically selects the shortest form of the instruction.

4.3.4 RELATIVE

The relative addressing mode is only used in branch instructions. In relative addressing, the contents of the 8-bit signed byte (the offset) following the opcode is added to the PC if, and only if, the branch conditions are true. Otherwise, control proceeds to the next instruction. The span of relative addressing is from -126 to +129 from the opcode address. The programmer need not calculate the offset when using the Motorola assembler, since it calculates the proper offset and checks to see that it is within the span of the branch.

4.3.5 INDEXED, NO OFFSET

In the indexed, no offset addressing mode, the effective address of the argument is contained in the 8-bit index register. This addressing mode can access the first 256 memory locations. These instructions are only 1 byte long. This mode is often used to move a pointer through a table or to hold the address of a frequently referenced RAM or I/O location.

4.3.6 INDEXED, 8-BIT OFFSET

In the indexed, 8-bit offset addressing mode, the effective address is the sum of the contents of the unsigned 8-bit index register and the unsigned byte following the opcode. The addressing mode is useful for selecting the Kth element in an n element table. With this 2-byte instruction, K would typically be in X with the address of the beginning of the table in the instruction. As such, tables may begin anywhere within the first 256 addressable locations and could extend as far as location 510 (\$1FE is the last location at which the instruction may begin).

4.3.7 INDEXED, 16-BIT OFFSET

In the indexed, 16-bit offset addressing mode, the effective address is the sum of the contents of the unsigned 8-bit index register and the two unsigned bytes following the opcode. This address mode can be used in a manner similar to indexed, 8-bit offset except that this 3-byte instruction allows tables to be anywhere in memory. As with direct and extended addressing, the Motorola assembler determines the shortest form of indexed addressing.

4.3.8 BIT SET/CLEAR

In the bit set/clear addressing mode, the bit to be set or cleared is part of the opcode, and the byte following the opcode specifies the direct addressing of the byte in which the specified bit is to be set or cleared. Any read/write bit in the first 256 locations of memory, including I/O, can be selectively set or cleared with a single 2-byte instruction.

4.3.9 BIT TEST AND BRANCH

The bit test and branch addressing mode is a combination of direct addressing and relative addressing. The bit that is to be tested and its condition (set or clear), is included in the opcode. The address of the byte to be tested is in the single byte immediately following the opcode byte. The signed relative 8-bit offset in the third byte is added to the PC if the specified bit is set or cleared in the specified memory location. This single 3-byte instruction allows the program to branch based on the condition of any readable bit in the first 256 locations of memory. The span of branching is from -125 to +130 from the opcode address. The state of the tested bit is also transferred to the carry bit of the condition code register.

4.3.10 INHERENT

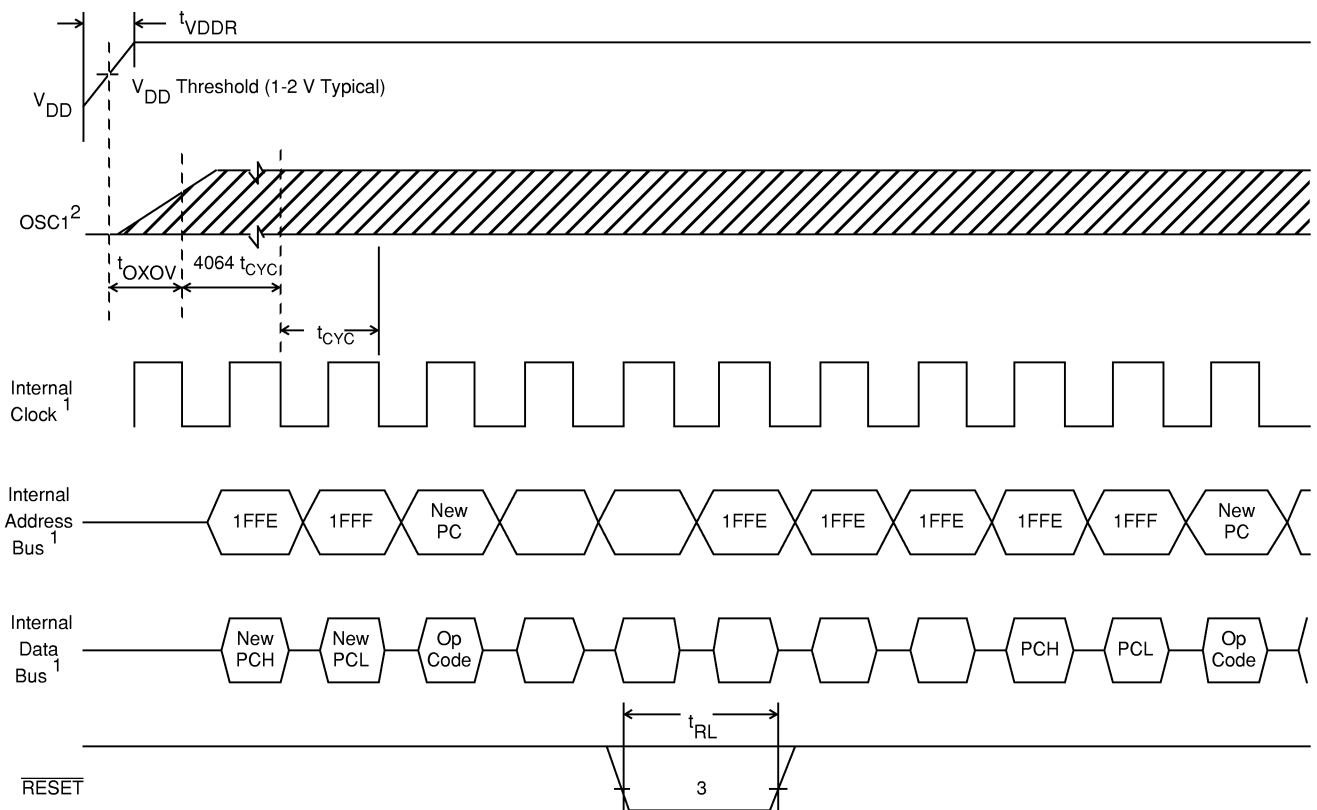
In the inherent addressing mode, all the information necessary to execute the instruction is contained in the opcode. Operations specifying only the index register or accumulator, as well as the control instruction with no other arguments, are included in this mode. These instructions are 1 byte long.

4.4 RESETS

The MCU can be reset three ways: by the initial power-on reset function, by an active low input to the $\overline{\text{RESET}}$ pin, and by a computer operating properly (COP) watchdog-timer time-out.

4.4.1 POWER-ON RESET (POR)

An internal reset is generated on power-up to allow the internal clock generator to stabilize. The power-on reset is strictly for power turn-on conditions and should not be used to detect a drop in the power supply voltage. There is a 4064 internal processor clock cycle (t_{CYC}) oscillator stabilization delay after the oscillator becomes active. If the $\overline{\text{RESET}}$ pin is low at the end of this 4064 cycle delay, the MCU will remain in the $\overline{\text{RESET}}$ condition until $\overline{\text{RESET}}$ goes high.



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.

Figure 4-1: Power-On Reset and $\overline{\text{RESET}}$

4.4.2 RESET PIN

The MCU is reset when a logic zero is applied to the $\overline{\text{RESET}}$ input for a period of one and one-half machine cycles (t_{CYC}).

4.4.3 COMPUTER OPERATING PROPERLY (COP) RESET

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.

The COP reset function is enabled or disabled by a mask option.

See **SECTION 6 COMPUTER OPERATING PROPERLY**, for more information on the COP.

4.4.4 ILLEGAL ADDRESS RESET

When an opcode fetch occurs at an address that is not in the RAM or ROM the part automatically resets.

4.5 INTERRUPTS

The MCU can be interrupted three different ways: the two maskable hardware interrupts ($\overline{\text{IRQ}}$ and timer) and the nonmaskable software interrupt instruction (SWI).

Interrupts cause the processor to save register contents on the stack and to set the interrupt mask (I-bit) to prevent additional interrupts. The RTI instruction causes the register contents to be recovered from the stack and normal processing to resume.

Unlike $\overline{\text{RESET}}$, hardware interrupts do not cause the current instruction execution to be halted, but are considered pending until the current instruction is complete.

NOTE: The current instruction is the one already fetched and being operated on.

When the current instruction is complete, the processor checks all pending hardware interrupts. If interrupts are not masked (CCR I-bit clear) and if the corresponding interrupt enable bit is set, the processor proceeds with interrupt processing; otherwise, the next instruction is fetched and executed.

If both an external interrupt and a timer interrupt are pending at the end of an instruction execution, the external interrupt is serviced first. The SWI is executed the same as any other instruction, regardless of the I-bit state.

Table 4-1: Vector Address for Interrupts and Reset lists vector addresses for all interrupt including reset.

Table 4-1: Vector Address for Interrupts and Reset

| Register | Flag Name | Interrupts | CPU Interrupt | Vector Address |
|----------|-----------|----------------------|---------------------------|----------------|
| N/A | N/A | Reset | $\overline{\text{RESET}}$ | \$1FFE-\$1FFF |
| N/A | N/A | Software | SWI | \$1FFC-\$1FFD |
| N/A | N/A | External Interrupt | IRQ | \$1FFA-\$1FFB |
| TSR | ICF | Timer Input Capture | TIMER | \$1FF8-\$1FF9 |
| TSR | OCF | Timer Output Compare | TIMER | \$1FF8-\$1FF9 |
| TSR | TOF | Timer Overflow | TIMER | \$1FF8-\$1FF9 |

4.5.1 HARDWARE CONTROLLED INTERRUPT SEQUENCE

The following three functions ($\overline{\text{RESET}}$, STOP, and WAIT) are not in the strictest sense an interrupt; however, they are acted upon in a similar manner. Flowcharts for hardware interrupts are shown in **Figure 4-1: Hardware Interrupt Flowchart**, and for STOP and WAIT in **Figure 4-2: STOP/WAIT Flowcharts**. A discussion is provided below. When the STOP disable Mask Option is selected, all STOP instructions become a NOP.

1. $\overline{\text{RESET}}$ - A low input on the $\overline{\text{RESET}}$ input pin 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. Much of the MCU is configured to a known state during this type of reset, as previously described in **4.4 RESETS**.
2. STOP - The STOP instruction causes the oscillator to be turned off and the processor to “sleep” until an external interrupt ($\overline{\text{IRQ}}$) or reset occurs. When the STOP disable Mask Option is selected, all STOP instructions become a NOP.
3. WAIT - The WAIT instruction causes all processor clocks to stop, but leaves the timer clock running. This “rest” state of the processor can be cleared by reset, an external interrupt ($\overline{\text{IRQ}}$), or Timer interrupt. There are no special wait vectors for these individual interrupts.

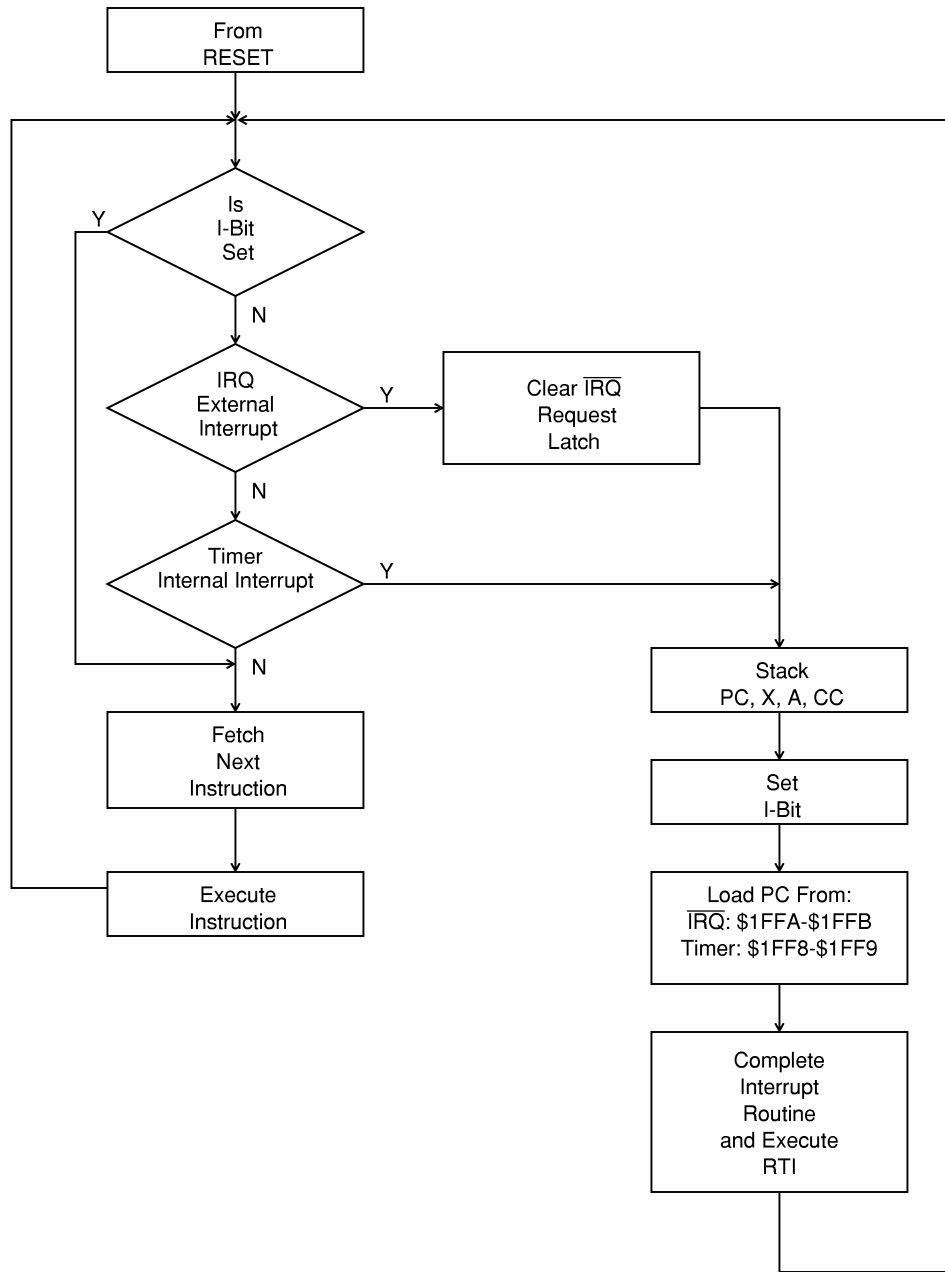


Figure 4-1: Hardware Interrupt Flowchart

4.5.2 TIMER INTERRUPT

There are three different timer interrupt flags that 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 will vector to the same interrupt service routine, located at the address specified by the contents of memory locations \$1FF8 and \$1FF9.

4.5.3 EXTERNAL INTERRUPT

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 \overline{IRQ} . 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 as a mask option.

NOTE: The internal interrupt latch is cleared in the first part of the interrupt service routine; therefore, another external interrupt pulse could be latched and serviced as soon as the I-bit is cleared.

4.5.4 SOFTWARE INTERRUPT (SWI)

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

4.6 LOW-POWER MODES

4.6.1 STOP

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 (and COP Watchdog timer) operation.

During the STOP mode, the TCR 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.

When the STOP disable Mask Option is selected, all STOP instructions become a NOP.

4.6.2 WAIT

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

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

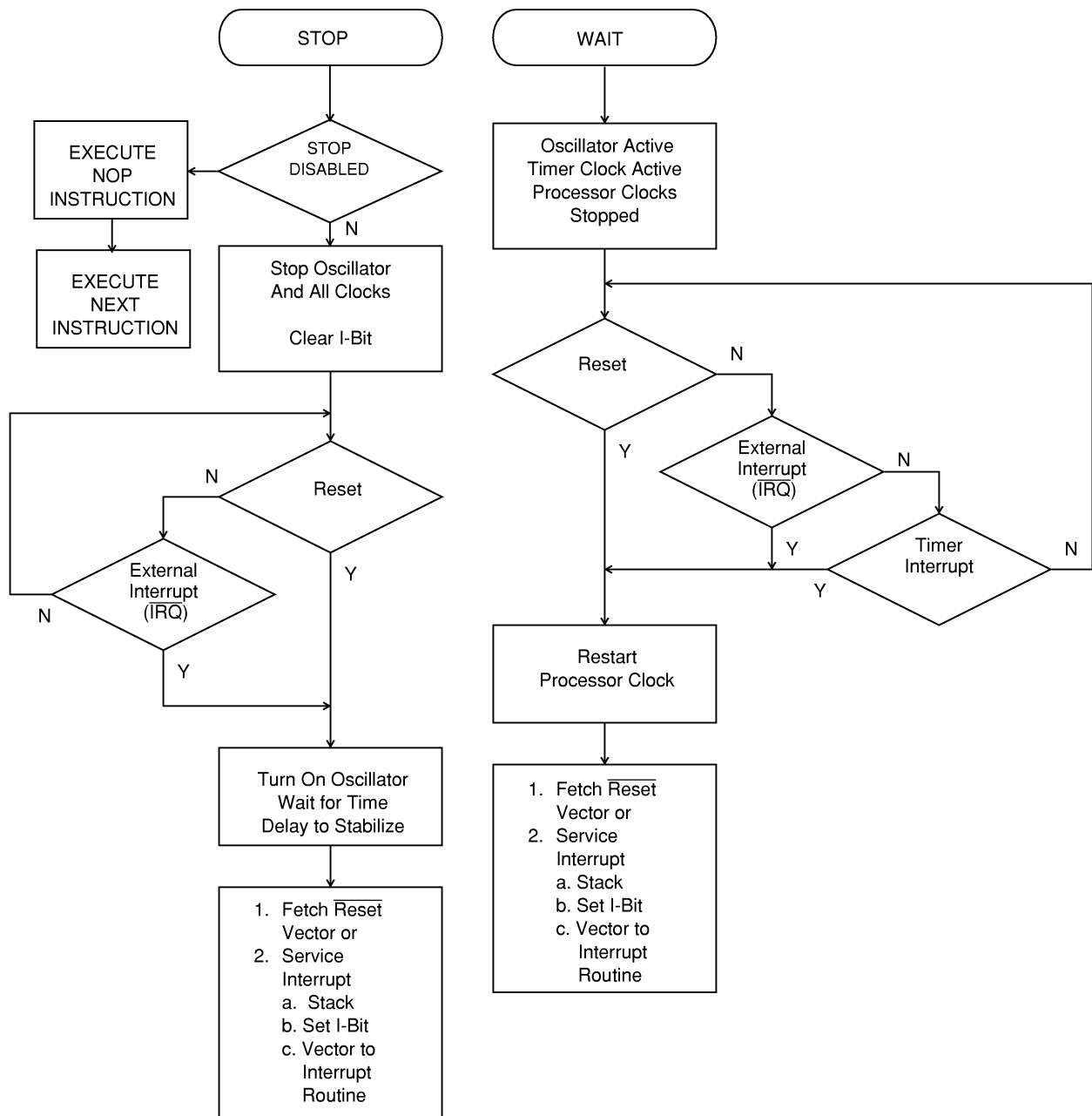


Figure 4-2: STOP/WAIT Flowcharts

THIS PAGE INTENTIONALLY LEFT BLANK

SECTION 5

TIMER

5.1 INTRODUCTION

The timer consists of a 16-bit, software-programmable counter driven by a fixed divide-by-four prescaler. This timer can be used for many purposes, including input waveform measurements while simultaneously generating an output waveform. Pulse widths can vary from several microseconds to many seconds. Refer to **Figure 5-1: Timer Block Diagram** for a timer block diagram.

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

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

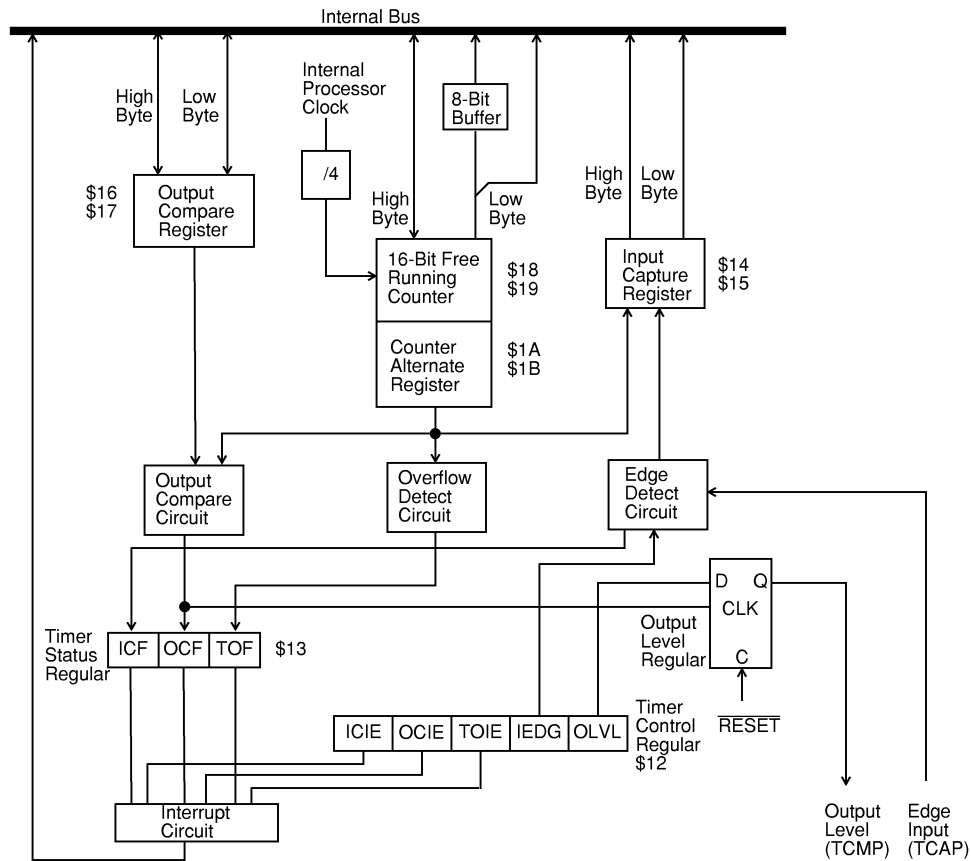


Figure 5-1: Timer Block Diagram

5.2 COUNTER

The key element in the programmable timer is a 16-bit, free-running counter or counter register preceded by a prescaler that divides the internal processor clock by four. The prescaler gives the timer a resolution of 2.0 microseconds if the internal bus clock is 2.0 MHz. The counter is incremented during the low portion of the internal bus clock. Software can read the counter at 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 must also be read to complete the sequence.

The counter alternate register differs from the counter register in one respect: a read of the counter register MSB can clear the timer overflow flag (TOF). Therefore, the counter

alternate register can be read at 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 always a read-only register. During a power-on reset, the counter is also preset to \$FFFC and begins running after the oscillator start-up delay. Since 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 rollover occurs by setting its interrupt enable bit (TOIE).

5.3 OUTPUT COMPARE REGISTER

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

The output compare register contents are compared with the contents of the free-running counter continually, and if a match is found, the corresponding output compare flag (OCF) bit is set 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 to only the LSB (\$17) does 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.

5.4 INPUT CAPTURE REGISTER

Two 8-bit registers, which make up the 16-bit input capture register, are read-only and are used to latch the value of the free-running counter after the corresponding input capture edge detector senses a defined transition. The level transition 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 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.

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.

After a read of the input capture register (\$14) MSB, the counter transfer is inhibited until the LSB (\$15) is also read. This characteristic causes the 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.

5.5 TIMER CONTROL REGISTER (TCR) \$12

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

| | | | | | | | | |
|--------|------|------|------|---|---|---|------|------|
| \$12 | ICIE | OCIE | TOIE | 0 | 0 | 0 | IEDG | OLVL |
| RESET: | 0 | 0 | 0 | 0 | 0 | 0 | U | 0 |

Figure 5-2: 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

IEDG - Input Edge

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

- 1 = Positive edge
- 0 = Negative edge

Reset does not affect the IEDG bit (U=unaffected).

OLVL - Output Level

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

1 = High output

0 = Low output

Bits 2, 3 and 4 - Not used

Always read zero

5.6 TIMER STATUS REGISTER (TSR) \$13

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

| | | | | | | | | |
|--------|-----|-----|-----|---|---|---|---|---|
| \$13 | ICF | OCF | TOF | 0 | 0 | 0 | 0 | 0 |
| RESET: | U | U | U | 0 | 0 | 0 | 0 | 0 |

Figure 5-3: 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

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

TOF - Timer Overflow Flag

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

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

Bits 0-4 - 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 LSB of the free-running counter is read but not for the purpose of servicing the flag.

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

5.7 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.

5.8 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 $\overline{\text{RESET}}$ is used, the counter is forced to \$FFFC. During STOP, if at least one valid input capture edge occurs at the TCAP pin, the input capture detect circuit is armed. This does not set any timer flags or wake up the MCU, but when the MCU does wake up, there is an active input capture flag and data from the first valid edge that occurred during the STOP mode. If $\overline{\text{RESET}}$ is used to exit STOP mode, then no input capture flag or data remains, even if a valid input capture edge occurred.

SECTION 6

COMPUTER OPERATING PROPERLY

6.1 INTRODUCTION

This device includes a “Watchdog” Computer Operating Properly (COP) feature as a mask option. The COP is implemented with an 18-bit ripple counter. This provides a time-out period of 64 milliseconds at a bus rate of 2 MHz. If the COP should time-out, a system reset will occur and the device will be re-initialized in the same fashion as a POR or external RESET.

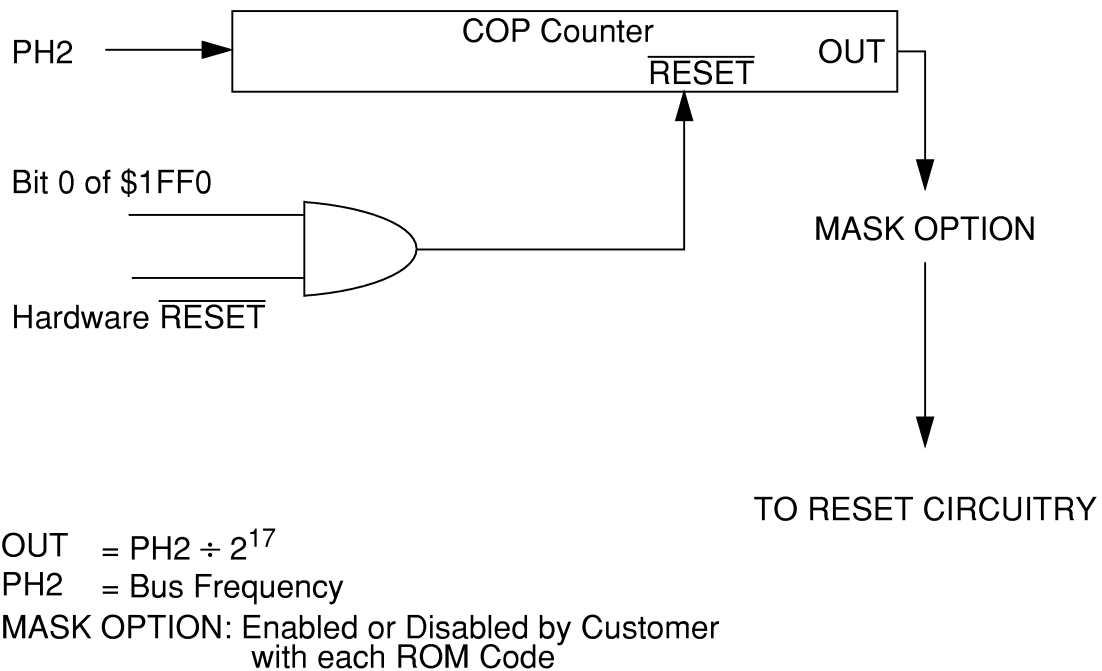


Figure 6-1: COP Block Diagram

6.2 RESETTING THE COP

Preventing a COP reset is done by writing a “0” to the COPR bit. This action resets the counter and begins the time-out period again. The COPR bit is bit 0 of address \$1FF0. A read of address \$1FF0 results in the user-defined ROM data at that location.

6.3 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 COPR bit to prevent a COP reset.

6.4 COP DURING STOP MODE

STOP mode disables the oscillator circuit and thereby turns the clock off for the entire device. The COP counter will be reset when STOP mode is entered. If a reset is used to exit STOP mode, the COP counter will be reset after the 4064 cycles of delay after STOP mode. If an IRQ is used to exit STOP mode, the COP counter will not be reset after the 4064 cycle delay and will have that many cycles already counted when control is returned to the program.

6.5 COP DURING SELF-CHECK MODE

The COP is disabled in Self-Check Mode.

SECTION 7

INPUT/OUTPUT PORTS

In User Mode, there are 15 lines arranged as two I/O ports. These pins, PA0-PA7 and PC0-PC6, are programmable as either inputs or outputs under software control of the data direction registers.

To avoid a glitch on the output pins, write data to the I/O Port Data Register before writing a 1 to the corresponding Data Direction Register.

7.1 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 address \$0000 and the data direction register (DDR) is at address \$0004. Reset does not affect the data registers, but clears the data direction registers, thereby returning the ports to inputs. Writing a 1 to a DDR bit sets the corresponding port bit to output mode.

7.2 PORT B

Port B is a 3-bit port that shares its pins with the on chip comparator. The Port B data register is at \$0001. Two input only (CMP+/PB6 and CMP-/PB5) pins and one output only pin (CMPO/PB7) comprise Port B. There is no DDR for Port B.

7.3 PORT C

Port C is a 7-bit bidirectional port that does not share any of its pins with other subsystems. The Port C data register is at address \$0002 and the data direction register (DDR) is at address \$0006. Reset does not affect the data registers, but clears the data direction registers, thereby returning the ports to inputs. Writing a 1 to a DDR bit sets the corresponding port bit to output mode.

7.4 PORT D

Port D is a 1-bit port that is input only. The Port D data register is at address \$0003. Port D is shared with the timer input capture circuitry. There is no DDR for Port D. For more information, see **5.4 INPUT CAPTURE REGISTER**.

7.5 INPUT/OUTPUT PROGRAMMING

Ports A and C may be programmed as input or output 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 port has an associated DDR. Any Port A or Port C 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 Port A and Port C 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. See **Table 7-1: I/O Pin Functions** and **Figure 7-1: Port I/O Circuitry**.

Table 7-1: I/O Pin Functions

| R/W | DDR | I/O Pin Function |
|-----|-----|---|
| 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 of 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. |

R/W is an internal signal.

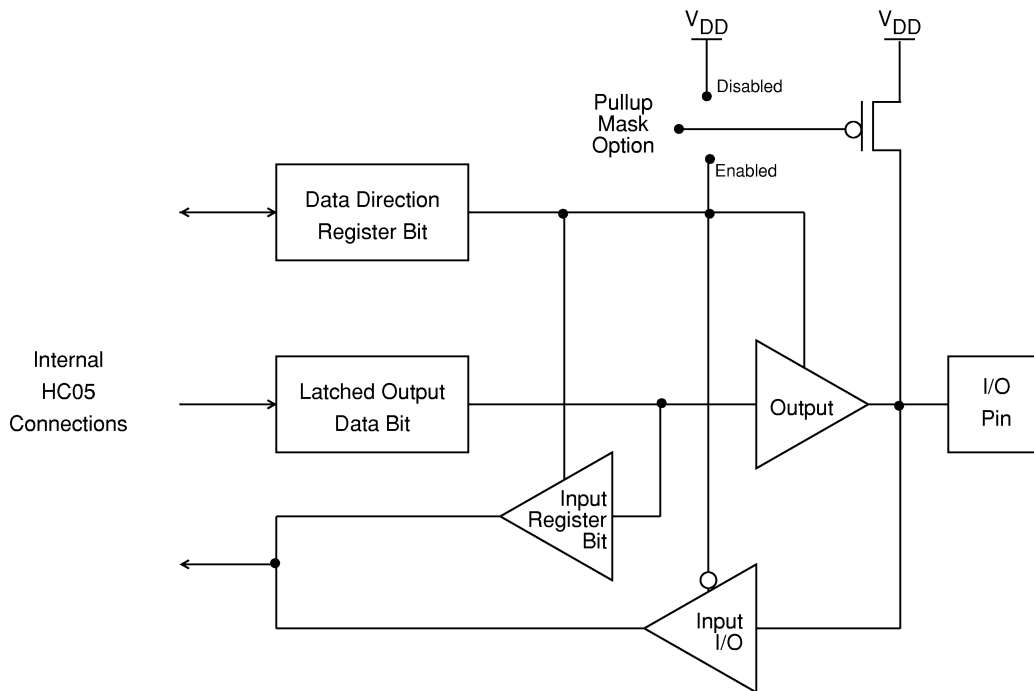


Figure 7-1: Port I/O Circuitry

SECTION 8

PULSE WIDTH MODULATOR (PWM)

8.1 PWM INTRODUCTION

The pulse width modulator (PWM) system has two 8-bit channels (PWM0 and PWM1). PWM 1 is delayed by two PWM clock cycles. The PWM has a programmable period of $256 \times T$, where T can be E/2, E/4, and E/8 for an output frequency of 4 KHz, 2 KHz, and 1 KHz respectively with E= 2 MHz. E is the internal bus frequency fixed to half of the external oscillator frequency.

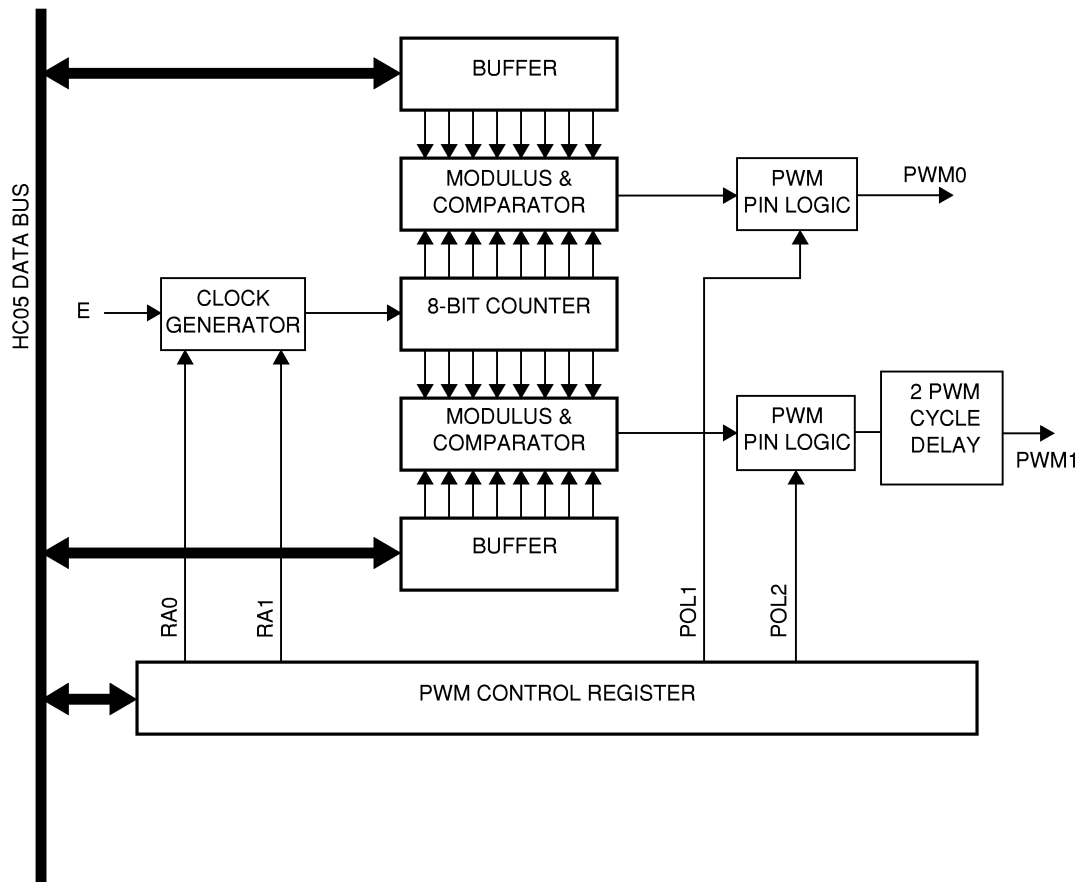


Figure 8-1: PWM Block Diagram

8.2 FUNCTIONAL DESCRIPTION

The PWM is capable of generating signals from 0% to 100% duty cycle. A \$00 in the PWM data register yields an “OFF” output (0%), but an \$FF yields a duty of 255/256 when the Polarity Bit is 1. To achieve the 100% duty (“ON” output), the polarity control bit is set to 0 for that channel (for example, PWM0 or PWM1) while the data register contains \$00.

When not in use, the PWM system can be shut off to save power by clearing the clock rate select bits RA0 and RA1 in PWCR.

Writes to the PWM can be performed at any time without affecting the output signal until the two data latches and the control latch have been written to. Updates on the outputs, PWM0 and PWM1, occur at the end of the PWM period. At this time, the new value is loaded into the PWM data and control registers. If a write to the registers is performed during the off state, the data gets transferred directly to the PWM registers. All registers are updated after the PWM data register 1 is written to and the end of a PWM cycle occurs.

The PWM output can have an active high or an active low pulse under software control.

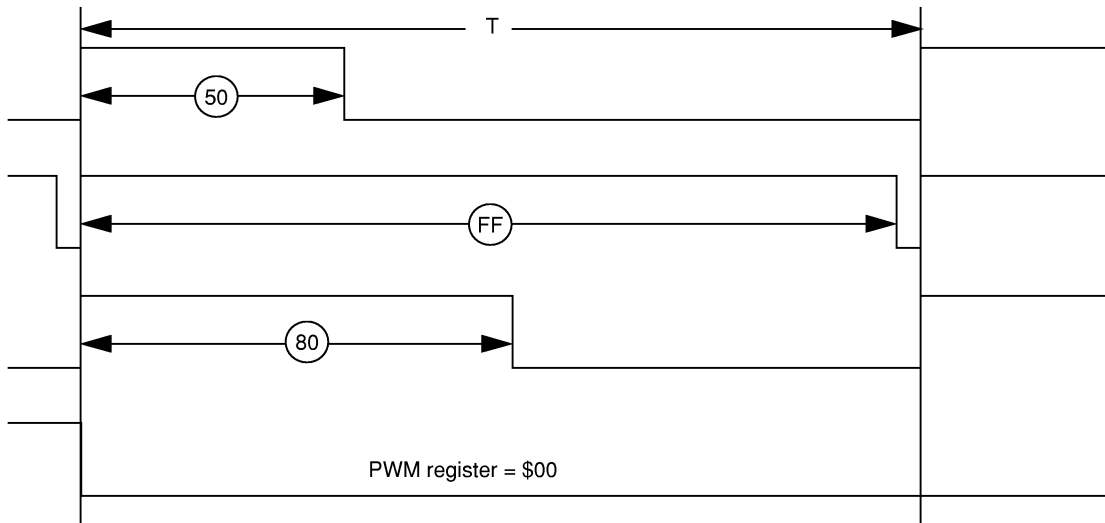


Figure 8-2: PWM Waveforms (POL = 1)

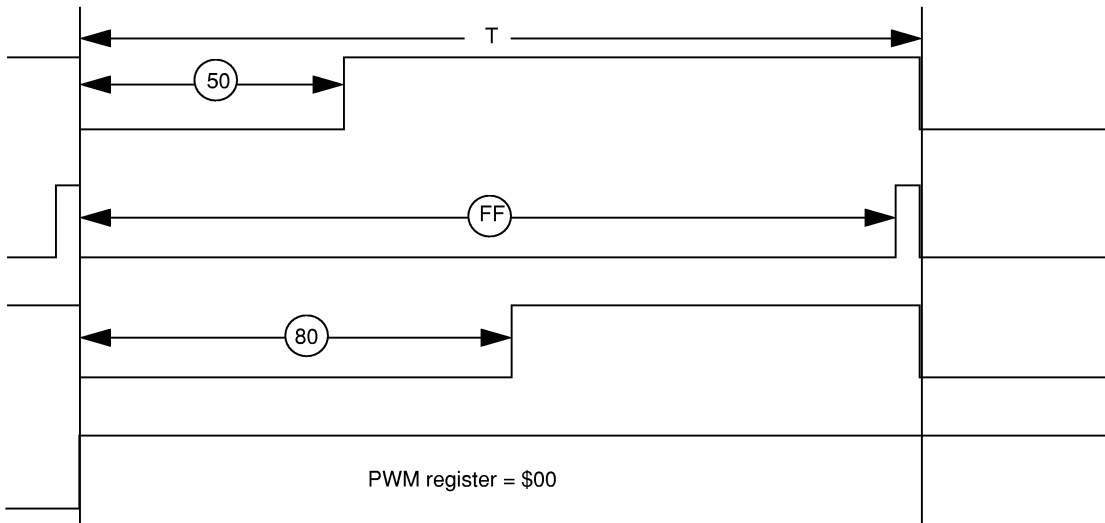


Figure 8-3: PWM Waveforms (POL = 0)

8.3 REGISTERS

Associated with the PWM system, there are two PWM data registers and a control register. The following registers can be written to and read at any time. The data is not latched until PWM Data Register 1 (address location = \$10) has been written to and the end of a cycle has been reached.

8.3.1 PWM CONTROL \$0E.

| | | | | | | | | |
|--------|----|----|----|----|------|------|-----|-----|
| \$0E | -- | -- | -- | -- | POL1 | POL0 | RA1 | RA0 |
| RESET: | -- | -- | -- | -- | 1 | 1 | 0 | 0 |

Figure 8-4: PWM Control Register

8.3.2 POL1 - PWM1 Polarity

When set, this bit makes the PWM1 pulse active high, and when low, the pulse is active low. Refer to **Figure 8-2: PWM Waveforms (POL = 1)** and **Figure 8-3: PWM Waveforms (POL = 0)**.

8.3.3 POL0 - PWM0 Polarity

When set, this bit makes the PWM0 pulse active high, and when low, the pulse is active low. Refer to **Figure 8-2: PWM Waveforms (POL = 1)** and **Figure 8-3: PWM Waveforms (POL = 0)**.

RA0, RA1 - PWM A Clock Rate bits

These bits select the input clock rate and determine the period.

Table 8-1: PWM Clock Rate

| RA1: RA0 | PWM Input Clock |
|----------|-----------------|
| 0 0 | OFF |
| 0 1 | E/2 |
| 1 0 | E/4 |
| 1 1 | E/8 |

8.3.4 PWM DATA REGISTERS

The PWM system has two 8-bit data registers that hold the duty cycle for each PWM output. PWM Data 0 and PWM Data 1 are the data registers located at addresses \$0F and \$10 respectively. These registers are unaffected by RESET.

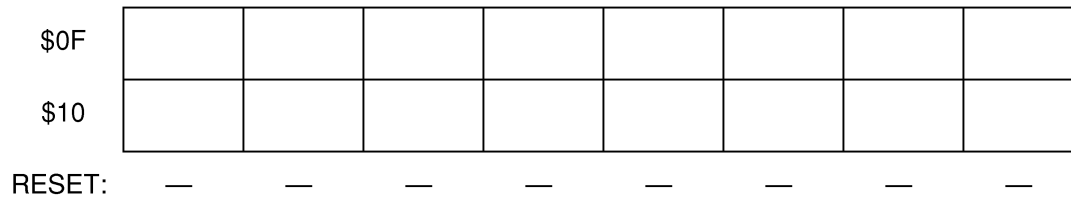


Figure 8-5: PWM Data Registers

A delay of 2 PWM clock counts exists between the first transition of the PWM0 signal to the first transition of the PWM1 signal. Refer to **Figure 8-1: PWM Block Diagram**.

8.4 PWM DURING WAIT MODE

The PWM continues normal operation during WAIT mode. To decrease power consumption during WAIT, it is recommended that the rate select bits in both PWM Control Registers are cleared if the PWM D/A converter is not being used.

8.5 PWM DURING STOP MODE

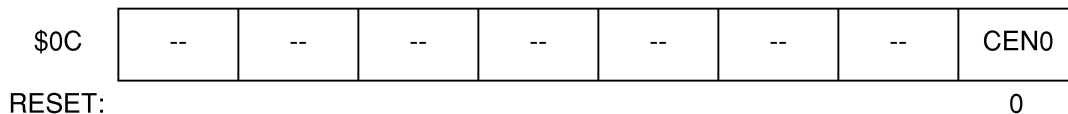
In STOP mode the oscillator is stopped causing the PWM to cease function. Any signal in process is aborted in whatever phase the signal happens to be in.

8.6 PWM DURING RESET

Upon RESET, the RA0 and RA1 bits in PWM CONTROL are cleared. This in effect disables the PWM system and sets the outputs driving low. The user should write to the data registers prior to enabling the PWM system (for example, prior to setting RA1 or RA0). This will avoid an erroneous duty cycle from being driven.

SECTION 9**COMPARATOR**

Port B shares its pins with CMP+/PB6, CMP-/PB5 and CMPO/PB7 of the comparator. This circuitry is used for the comparison of analog signals with a digital output. The comparator circuitry may be powered up by setting the CEN0 bit in the Comparator Control Register. This register is located at \$000C and is cleared by reset. The state of the comparator output pin upon RESET is logic 0. For further electrical information, see **SECTION 10 ELECTRICAL SPECIFICATIONS**.

9.1 COMPARATOR CONTROL REGISTER**Figure 9-1: Comparator Control Register****CEN0 - Comparator Enable**

This bit powers up the voltage comparator on PB7-PB5 when set. 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.

9.2 READING COMPARATOR OUTPUT

The state of the output pin, CMPO/PB7, is internally readable. From **Figure 3-2: I/O Registers** Port B Data is located at address location \$01. A read of Port B bit 7 reads the data latch. When CEN0 is set, this read gives the current state of the comparator output. Port B bit 5 and 6 are not internally readable. Writes to PB7/COMPO when the CEN0 bit is set have no effect to PB7. Writes to PB7/COMPO when the CEN0 bit is cleared writes to PB7 data latch.

9.3 COMPARATOR DURING WAIT

The comparator operates normally in WAIT. If the user wishes to save power during the WAIT Mode, the CEN0 bit should be cleared before the WAIT instruction. For further information on WAIT Mode, see **4.6 LOW-POWER MODES**.

9.4 COMPARATOR DURING STOP

STOP clears the CEN0 bit. This in effect disables the comparator. If the STOP disable Mask option is used, all STOP instructions become a NOP and STOP mode never occurs. For further information on STOP mode, see **4.6 LOW-POWER MODES**.

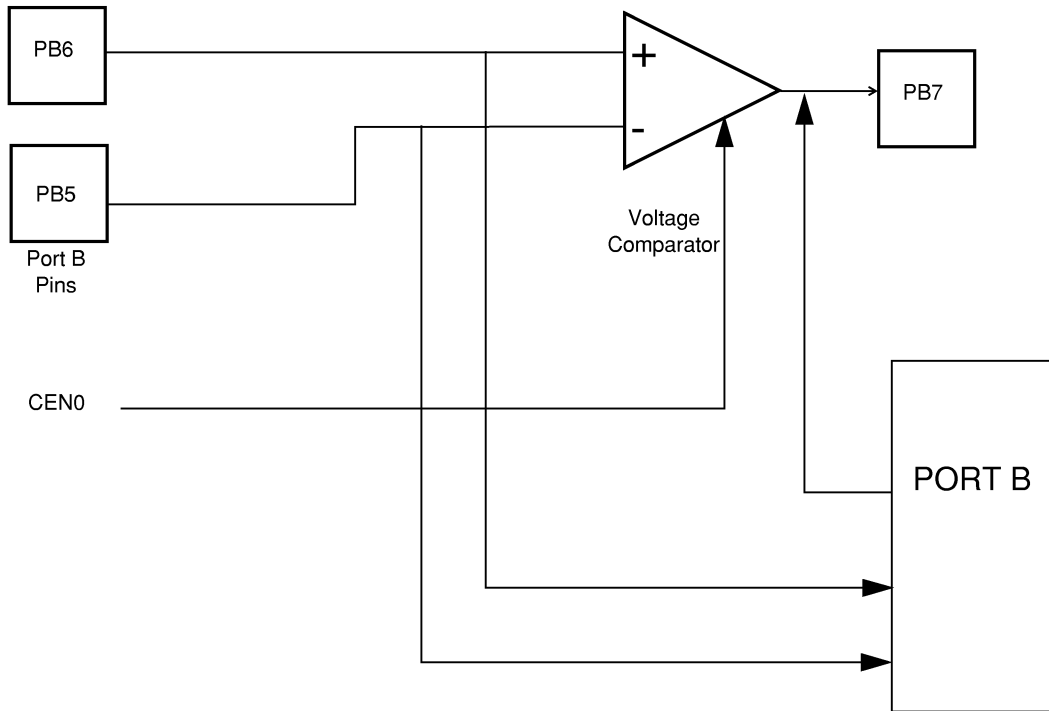


Figure 9-2: Port B Comparator Circuitry

SECTION 10**ELECTRICAL SPECIFICATIONS****10.1 MAXIMUM RATINGS**(Voltages referenced to V_{SS})

| Rating | Symbol | Value | Unit |
|---|-----------|--|--------------------|
| Supply Voltage | V_{DD} | -0.5 to 7.0 | V |
| Input Voltage | V_{IN} | $V_{SS} - 0.5$ to $V_{DD} + 0.5$ | V |
| Self-Check Mode (\overline{IRQ} Pin Only) | V_{IN} | $V_{SS} - 0.5$ to $2 \times V_{DD} + 0.5$ | V |
| Current Drain Per Pin Excluding V_{DD} and V_{SS} | I | 25 | mA |
| Operating Temperature Range MC68HC05P5P, DW MC68HC05P5CP, CDW | T_A | T_L to T_H 0 to +70 -40 to +85 | $^{\circ}\text{C}$ |
| Storage Temperature Range | T_{STG} | -65 to +150 | $^{\circ}\text{C}$ |

NOTES:

1. P = Plastic dual-in-line package (PDIP)
2. C = Extended temperature range (-40 to +85 $^{\circ}\text{C}$)
3. DW = Small Outline Integrated Circuit (SOIC)

Maximum ratings are the extreme limits the device can be exposed to without causing permanent damage to the chip. The device is **NOT** intended to operate at these conditions.

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}).

10.2 THERMAL CHARACTERISTICS

| Characteristic | Symbol | Value | Unit |
|-------------------------------|---------------|-------|-----------------------------|
| Thermal Resistance Plastic | θ_{JA} | 60 | $^{\circ}\text{C}/\text{W}$ |
| SOIC | θ_{JA} | 60 | $^{\circ}\text{C}/\text{W}$ |

10.3 DC ELECTRICAL CHARACTERISTICS (5.0 Vdc)

($V_{DD} = 5.0 \text{ Vdc} \pm 10\%$, $V_{SS} = 0 \text{ Vdc}$, $T_A = -40^\circ\text{C}$ to $+85^\circ\text{C}$, unless otherwise noted)

| Characteristic | Symbol | Min | Typ | Max | Unit |
|--|--|---------------------|------------------------|-------------------------|--|
| Output Voltage $I_{Load} = 10.0 \mu\text{A}$ $I_{Load} = -10.0 \mu\text{A}$ | V_{OL} V_{OH} | — $V_{DD}-0.1$ | — — | 0.1 — | V |
| Output High Voltage ($I_{Load} = -0.8 \text{ mA}$) PA0-7, PB7, PC0-6, PWM0-1, TCMP | V_{OH} | $V_{DD}-0.8$ | — | — | V |
| Output Low Voltage ($I_{Load} = 1.6 \text{ mA}$) PA0-7, PB7, PC0-6, PWM0-1, TCMP | V_{OL} | — | — | 0.4 | V |
| Input High Voltage PA0-7, PC0-6, PB5-6, TCAP/PD7, \overline{IRQ} , \overline{RESET} , OSC1 | V_{IH} | $0.7 \times V_{DD}$ | — | V_{DD} | V |
| Input Low Voltage PA0-7, PC0-6, PB5-6, OSC1, \overline{IRQ} | V_{IL} | V_{SS} | — | $0.2 \times V_{DD}$ | V |
| Input Low Voltage \overline{RESET} , TCAP/PD7 | V_{IL} | V_{SS} | — | $0.3 \times V_{DD}$ | V |
| Supply Current (see Notes) Run Wait Stop (Comparator Disabled) 25°C 0°C to +70°C (STANDARD) | I_{DD} I_{DD} I_{DD} I_{DD} | — — — — | 3.5 1.6 2.0 — | 7.0 4.0 50 140 | mA mA μA μA |
| Input Current \overline{RESET} , \overline{IRQ} , OSC1 | I_{IN} | — | — | ± 1 | μA |
| Capacitance Ports (as Input or Output) \overline{RESET} , \overline{IRQ} | C_{OUT} C_{IN} | — — | — — | 12 8 | pF pF |
| Input Current (Without Port Pullups) PB5-6, TCAP/PD7, PA0-7, PC0-6 | I_{IN} | — | — | ± 1 | μA |
| Input Current with Port Pullups PA0-7, PC0-6 | I_{IN} | 0.2 | 0.3 | 0.5 | ma |

NOTES:

- All values shown reflect average measurements.
- Typical values at midpoint of voltage range, 25°C only.
- Wait I_{DD} : Only timer system active.
- Run (Operating) I_{DD} , Wait I_{DD} : Measured using external square wave clock source ($f_{osc} = 4.2 \text{ MHz}$), PA0-7 and PC0-6 = V_{DD} , all other inputs 0.2 V from rail; no dc loads, less than 50 pF on all outputs, $C_L = 20 \text{ pF}$ on OSC2.
- Wait, Stop I_{DD} : All ports configured as inputs, PA0-7 and PC0-6 = V_{DD} , all other inputs 0.2 V from rail.
- Stop I_{DD} measured with OSC1 = V_{SS} .
- Wait I_{DD} is affected linearly by the OSC2 capacitance.

10.4 DC ELECTRICAL CHARACTERISTICS (3.0 Vdc)

($V_{DD} = 3.0 \text{ Vdc} \pm 10\%$, $V_{SS} = 0 \text{ Vdc}$, $T_A = -40^\circ\text{C}$ to $+85^\circ\text{C}$, unless otherwise noted)

| Characteristic | Symbol | Min | Typ | Max | Unit |
|--|--|---------------------|----------------------|------------------------|--|
| Output Voltage $I_{Load} = 10.0 \mu\text{A}$ $I_{Load} = -10.0 \mu\text{A}$ | V_{OL} V_{OH} | — $V_{DD}-0.1$ | — — | 0.1 — | V |
| Output High Voltage ($I_{Load} = -0.2 \text{ mA}$) PA0-7, PB7, PC0-6, PWM0-1, TCMP | V_{OH} | $V_{DD}-0.3$ | — | — | V |
| Output Low Voltage ($I_{Load} = 0.4 \text{ mA}$) PA0-7, PB7, PC0-6, PWM0-1, TCMP | V_{OL} | — | — | 0.3 | V |
| Input High Voltage PA0-7, PC0-6, PB5-6, TCAP/PD7, \overline{IRQ} , \overline{RESET} , OSC1 | V_{IH} | $0.7 \times V_{DD}$ | — | V_{DD} | V |
| Input Low Voltage PA0-7, PC0-6, PB5-6, OSC1, \overline{IRQ} | V_{IL} | V_{SS} | — | $0.2 \times V_{DD}$ | V |
| Input Low Voltage \overline{RESET} , TCAP/PD7 | V_{IL} | V_{SS} | — | $0.3 \times V_{DD}$ | V |
| Supply Current (see Notes) Run Wait Stop (Comparator Disabled) 25°C 0°C to +70°C (STANDARD) | I_{DD} I_{DD} I_{DD} I_{DD} | — — — — | 1.0 0.5 — — | 2.5 1.4 30 80 | mA mA μA μA |
| Input Current \overline{RESET} , \overline{IRQ} , OSC1 | I_{IN} | — | — | ± 1 | μA |
| Capacitance Ports (as Input or Output) \overline{RESET} , \overline{IRQ} | C_{OUT} C_{IN} | — — | — — | 12 8 | pF pF |
| Input Current (Without Port Pullups) PB5-6, TCAP/PD7, PA0-7, PC0-6 | I_{IN} | — | — | ± 1 | μA |
| Input Current with Port Pullups PA0-7, PC0-6 | I_{IN} | 60 | 100 | 160 | μA |

NOTES:

- All values shown reflect average measurements.
- Typical values at midpoint of voltage range, 25°C only.
- Wait I_{DD} : Only timer system active.
- Run (Operating) I_{DD} , Wait I_{DD} : Measured using external square wave clock source ($f_{osc} = 2.0 \text{ MHz}$), PA0-7 and PC0-6 = V_{DD} , all other inputs 0.2 V from rail; no dc loads, less than 50 pF on all outputs, $C_L = 20 \text{ pF}$ on OSC2.
- Wait, Stop I_{DD} : All ports configured as inputs, PA0-7 and PC0-6 = V_{DD} , all other inputs 0.2 V from rail.
- Stop I_{DD} measured with OSC1 = V_{SS} .
- Wait I_{DD} is affected linearly by the OSC2 capacitance.

10.5 CONTROL TIMING (5.0 Vdc)

($V_{DD} = 5.0 \text{ Vdc} \pm 10\%$, $V_{SS} = 0 \text{ Vdc}$, $T_A = -40^\circ\text{C}$ to $+85^\circ\text{C}$, unless otherwise noted)

| Characteristic | Symbol | Min | Max | Unit |
|---|------------------|-----|-----|-----------|
| Frequency of Operation | | | | |
| Crystal | f_{OSC} | — | 4.2 | MHz |
| External Clock | f_{OSC} | dc | 4.2 | MHz |
| Internal Operating Frequency ($f_{OSC} \div 2$) | | | | |
| Crystal | f_{OP} | — | 2.1 | MHz |
| External Clock | f_{OP} | dc | 2.1 | MHz |
| Cycle Time | t_{CYC} | 476 | — | ns |
| Crystal Oscillator Start-up Time | t_{OXOV} | — | 100 | ms |
| Stop Recovery Start-up Time (Crystal Oscillator) | t_{ILCH} | — | 100 | ms |
| $\overline{\text{RESET}}$ 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 $19 t_{CYC}$.

10.6 CONTROL TIMING (3.0 Vdc)

($V_{DD} = 3.0 \text{ Vdc} \pm 10\%$, $V_{SS} = 0 \text{ Vdc}$, $T_A = -40^\circ\text{C}$ to $+85^\circ\text{C}$, unless otherwise noted)

| Characteristic | Symbol | Min | Max | Unit |
|---|------------------|------|-----|-----------|
| Frequency of Operation | | | | |
| Crystal | f_{OSC} | — | 2.0 | MHz |
| External Clock | f_{OSC} | dc | 2.0 | MHz |
| Internal Operating Frequency ($f_{OSC} \div 2$) | | | | |
| Crystal | f_{OP} | — | 1.0 | MHz |
| External Clock | f_{OP} | dc | 1.0 | MHz |
| Cycle Time | t_{CYC} | 1000 | — | ns |
| Crystal Oscillator Start-up Time | t_{OXOV} | — | 100 | ms |
| Stop Recovery Start-up Time (Crystal Oscillator) | t_{ILCH} | — | 100 | ms |
| $\overline{\text{RESET}}$ Pulse Width | t_{RL} | 1.5 | — | t_{CYC} |
| Interrupt Pulse Width Low (Edge-Triggered) | t_{ILIH} | 250 | — | ns |
| Interrupt Pulse Period | t_{ILIL} | * | — | t_{CYC} |
| OSC1 Pulse Width | t_{OH}, t_{OL} | 200 | — | 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 $19 t_{CYC}$.

10.7 COMPARATOR

($V_{DD} = 5.0 \text{ Vdc}$, $V_{SS} = 0 \text{ Vdc}$, $T_A = -40^\circ\text{C}$ to $+85^\circ\text{C}$, 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} | $V_{SS} - 0.5$ | $V_{SS} + 2.5$ | V |
| Offset ⁴ | V_{OFF} | — | ± 15 | mV |
| Comparator Enable Time ⁵ | t_{CEN} | -- | 100 | ms |
| Input Current $V_{SS} \leq V_{IN} \leq V_{DD}$ | I_{IN} | — | 1 | μA |
| Input Current $V_{IN} \leq V_{SS}$ | I_{IN} | — | 4.0 | mA |
| Gain Bandwidth Product | | — | 10 | kHz |
| Slew Rate | SR | 5 | — | $\text{V}/\mu\text{s}$ |
| Saturation Voltage ($I_{OUT} = -5\text{mA}$) | V_{SAT} | — | 0.5 | V |

NOTES:

1. Comparator is guaranteed to function over the specified input voltage range with no erroneous outputs.
2. Signal propagation time through the comparator 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 CEN0 bit until the comparator is fully functional.

10.8 PWM TIMING

| Characteristic | Symbol | Min | Max | Unit |
|----------------|-----------|-----|-----|------|
| PWM Rise Time | t_{PWR} | 15 | 35 | ns |
| PWM Fall Time | t_{PWR} | 15 | 35 | ns |

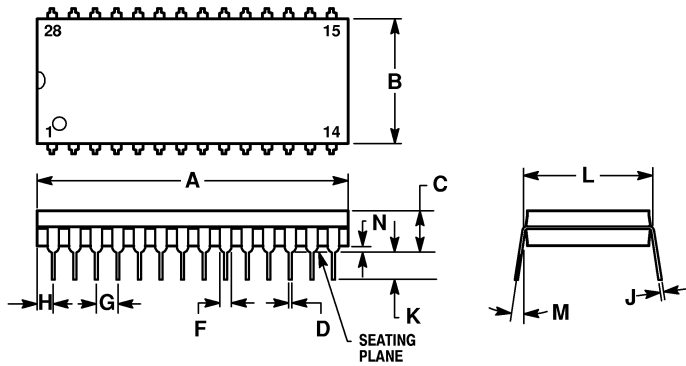
*See Figure 8-2: PWM Waveforms (POL = 1).

SECTION 11

MECHANICAL SPECIFICATIONS

This section describes the dimensions of the dual in-line package (DIP) and small outline integrated circuit (SOIC) MCU packages.

11.1 28-PIN PLASTIC DUAL IN-LINE PACKAGE (CASE 710-02)

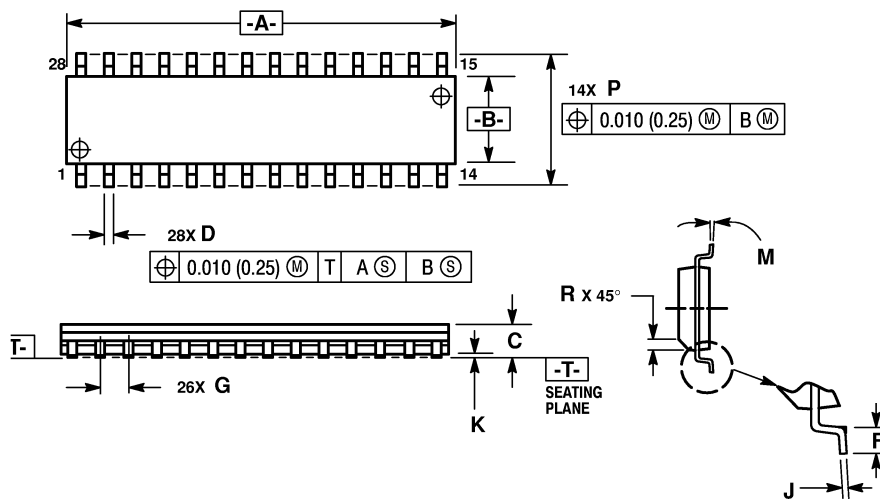


NOTES:

1. POSITIONAL TOLERANCE OF LEADS (D), SHALL BE WITHIN 0.25mm (0.010) AT MAXIMUM MATERIAL CONDITION, IN RELATION TO SEATING PLANE AND EACH OTHER.
2. DIMENSION L TO CENTER OF LEADS WHEN FORMED PARALLEL.
3. DIMENSION B DOES NOT INCLUDE MOLD FLASH.

| DIM | MILLIMETERS | | INCHES | |
|-----|-------------|-------|-----------|-------|
| | MIN | MAX | MIN | MAX |
| A | 36.45 | 37.21 | 1.435 | 1.465 |
| B | 13.72 | 14.22 | 0.540 | 0.560 |
| C | 3.94 | 5.08 | 0.155 | 0.200 |
| D | 0.36 | 0.56 | 0.014 | 0.022 |
| F | 1.02 | 1.52 | 0.040 | 0.060 |
| G | 2.54 BSC | | 0.100 BSC | |
| H | 1.65 | 2.16 | 0.065 | 0.085 |
| J | 0.20 | 0.38 | 0.008 | 0.015 |
| K | 2.92 | 3.43 | 0.115 | 0.135 |
| L | 15.24 BSC | | 0.600 BSC | |
| M | 0° | 15° | 0° | 15° |
| N | 0.51 | 1.02 | 0.020 | 0.040 |

11.2 28-PIN SMALL OUTLINE INTEGRATED CIRCUIT PACKAGE (CASE 751F-04)



NOTES:

1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: MILLIMETER.
3. DIMENSION A AND B DO NOT INCLUDE MOLD PROTRUSION.
4. MAXIMUM MOLD PROTRUSION 0.15 (0.006) PER SIDE.
5. DIMENSION D DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL BE 0.13 (0.005) TOTAL IN EXCESS OF D DIMENSION AT MAXIMUM MATERIAL CONDITION.

| DIM | MILLIMETERS | | INCHES | |
|-----|-------------|-------|-----------|-------|
| | MIN | MAX | MIN | MAX |
| A | 17.80 | 18.05 | 0.701 | 0.711 |
| B | 7.40 | 7.60 | 0.292 | 0.299 |
| C | 2.35 | 2.65 | 0.093 | 0.104 |
| D | 0.35 | 0.49 | 0.014 | 0.019 |
| F | 0.41 | 0.90 | 0.016 | 0.035 |
| G | 1.27 BSC | | 0.050 BSC | |
| J | 0.23 | 0.32 | 0.009 | 0.013 |
| K | 0.13 | 0.29 | 0.005 | 0.011 |
| M | 0° | 8° | 0° | 8° |
| P | 10.05 | 10.55 | 0.395 | 0.415 |
| R | 0.25 | 0.75 | 0.010 | 0.029 |

THIS PAGE INTENTIONALLY LEFT BLANK

SECTION 12

ORDERING INFORMATION

12.1 INTRODUCTION

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

12.2 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 **12.3 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 lowercase letters. Then press the return key to start the BBS software.

12.3 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.

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.

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.

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

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 filename of the source code.

12.4 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.

12.5 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 ten 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 ten 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.

12.6 MC ORDER NUMBERS

Table 12-1: Order Numbers shows the order numbers for the available package types.

Table 12-1: Order Numbers

| Package Type | Operating Temperature Range | MC Order Number |
|--|-----------------------------|-----------------|
| 28-Pin Plastic Dual In-Line Package (DIP) | 0 C to +70 C | MC68HC05P5P |
| 28-Pin Plastic Dual In-Line Package (DIP) | -40 C to +85 C | MC68HC05P5CP |
| 28-Pin Small Outline Integrated Circuit Package (SOIC) | 0 C to +70 C | MC68HC05P5DW |
| 28-Pin Small Outline Integrated Circuit Package (SOIC) | -40 C to +85 C | MC68HC05P5CDW |

THIS PAGE INTENTIONALLY LEFT BLANK

Freescale Semiconductor, Inc.

Home Page:

www.freescale.com

email:

support@freescale.com

USA/Europe or Locations Not Listed:

Freescale Semiconductor
Technical Information Center, CH370

1300 N. Alma School Road
Chandler, Arizona 85224
(800) 521-6274

480-768-2130

support@freescale.com

Europe, Middle East, and Africa:

Freescale Halbleiter Deutschland GmbH
Technical Information Center

Schatzbogen 7

81829 Muenchen, Germany

+44 1296 380 456 (English)

+46 8 52200080 (English)

+49 89 92103 559 (German)

+33 1 69 35 48 48 (French)

support@freescale.com

Japan:

Freescale Semiconductor Japan Ltd.

Headquarters

ARCO Tower 15F

1-8-1, Shimo-Meguro, Meguro-ku

Tokyo 153-0064, Japan

0120 191014

+81 2666 8080

support.japan@freescale.com

Asia/Pacific:

Freescale Semiconductor Hong Kong Ltd.

Technical Information Center

2 Dai King Street

Tai Po Industrial Estate,

Tai Po, N.T., Hong Kong

+800 2666 8080

support.asia@freescale.com

For Literature Requests Only:

Freescale Semiconductor

Literature Distribution Center

P.O. Box 5405

Denver, Colorado 80217

(800) 441-2447

303-675-2140

Fax: 303-675-2150

LDCForFreescaleSemiconductor@hibbertgroup.com

RoHS-compliant and/or Pb- free versions of Freescale products have the functionality and electrical characteristics of their non-RoHS-compliant and/or non-Pb- free counterparts. For further information, see <http://www.freescale.com> or contact your Freescale sales representative.

For information on Freescale's Environmental Products program, go to <http://www.freescale.com/epp>.

Information in this document is provided solely to enable system and software implementers to use Freescale Semiconductor products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document. Freescale Semiconductor reserves the right to make changes without further notice to any products herein. Freescale Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale Semiconductor 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 which may be provided in Freescale Semiconductor data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts. Freescale Semiconductor does not convey any license under its patent rights nor the rights of others. Freescale Semiconductor 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 Freescale Semiconductor product could create a situation where personal injury or death may occur. Should Buyer purchase or use Freescale Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold Freescale Semiconductor 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 Freescale Semiconductor was negligent regarding the design or manufacture of the part.



HC05P5GRS/D

**For More Information On This Product,
Go to: www.freescale.com**