68HC05P18

SPECIFICATION

REV 0.5

(General Release)

CSIC MCU Design Center
Austin, Texas

The MC68HC05P18 is an microcontroller unit (MCU) device in a 28-pin DIP or SOIC package with the HC05 CPU core, a 16-bit timer including an output compare and input capture, an 8-bit A/D converter with a 4 channel input multiplexer, a serial communications port (SIOP), a COP watchdog timer, and 21 general purpose I/O port lines. The 16 K byte memory map has 8064 bytes of user ROM, 192 bytes of RAM and 128 bytes of EEPRROM.

TABLE OF CONTENTS

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF FIGURES

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

SECTION 1 INTRODUCTION

The Motorola MC68HC05P18 is a low-cost microcontroller with a four-channel 8-bit A/D Converter, a 16-bit Timer with Output Compare and Input Capture, a serial communications port (SIOP), and a Computer Operating Properly (COP) Watchdog Timer. The HC05 CPU core contains 192 bytes of RAM, 8064 bytes of user ROM, 128 bytes of EEPROM, and 21 I/O pins (20 bidirectional, 1 input-only). This device is available in a 28-pin DIP or SOIC package. A functional block diagram of the MC68HC05P18 is shown in **Figure 1-1: MC68HC05P18 Block Diagram**.

1.1 FEATURES

- Low cost, HC05 Core running at 2 MHz bus speed
- 28-pin DIP or SOIC package
- On-Chip Crystal/Ceramic Resonator
- 8064 Bytes of User ROM (including 48 Bytes of Page Zero ROM and 16 Bytes of User Vectors)
- 192 Bytes of On-Chip RAM
- 128 Bytes of EEPROM
- Low Voltage Reset
- Four Channel 8-bit A/D Converter
- SIOP Serial Communications Port
- COP Watchdog Timer with active pull down on RESET
- 16-bit Timer with Output Compare and Input Capture
- Edge- and Level-Sensitive Interrupt or Edge-Sensitive Only (Mask Option)
- 20 Bidirectional I/O Lines and 1 Input-only line
- Individually Mask Selectable Pull-ups/Interrupts on Port A pins
- High Current Sink and Source on two I/O Pins (PC0 and PC1)
- Power Saving STOP and WAIT Mode Instructions and STOP conversion to HALT Mode (Mask Option)
- Mask Option for Clock Output pin

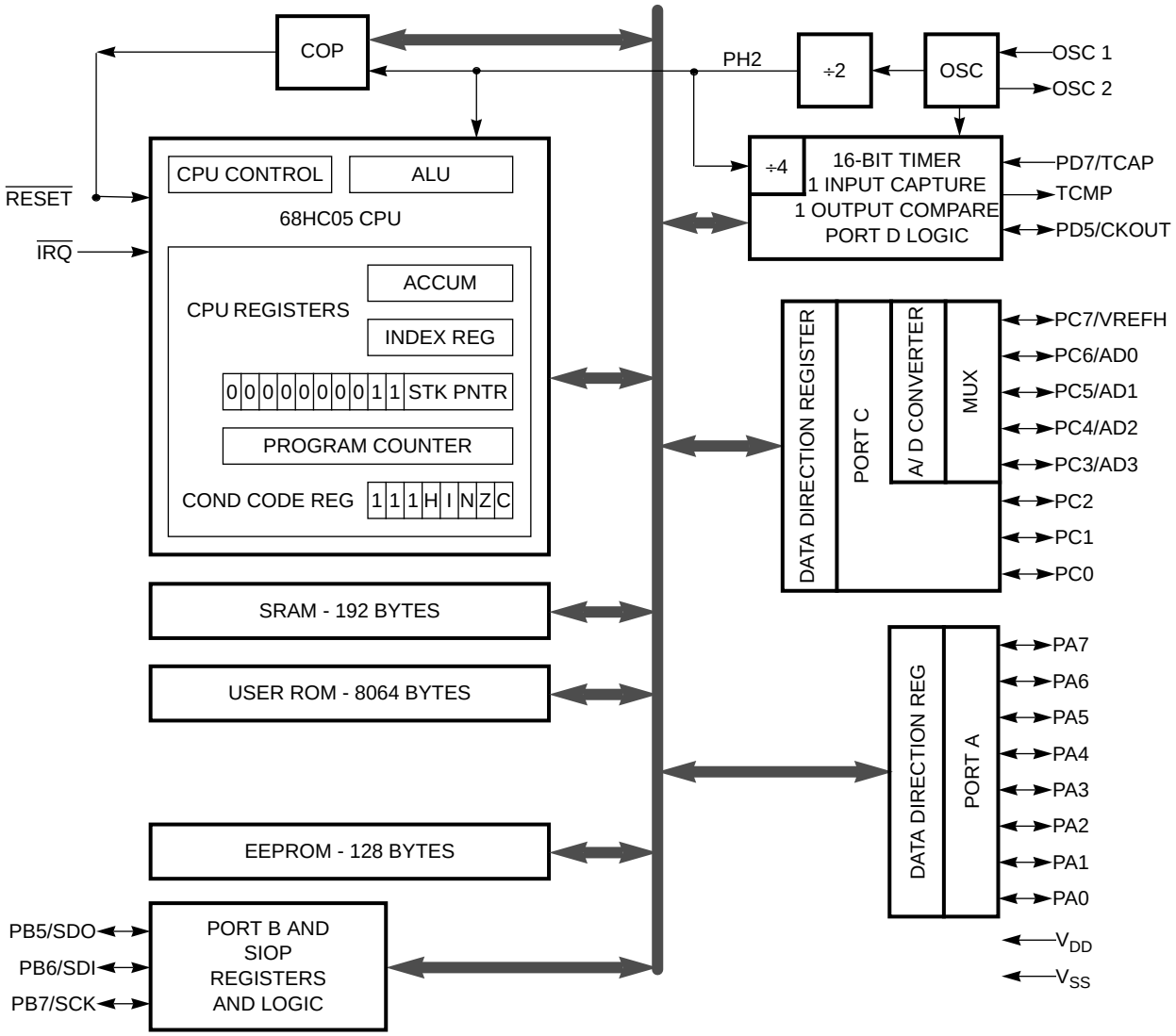


Figure 1-1: MC68HC05P18 Block Diagram

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

1.2 MASK OPTIONS

There are eight Mask Options on the MC68HC05P18. The Mask Options are as follows:

1. $\overline{\text{IRQ}}$ is Edge- and Level-Sensitive or Edge-Sensitive Only.
2. SIOP Most-Significant-Bit First or LSB First.
3. SIOP clock rate set to OSC divided by 2, 4, 8, or 16.
4. COP Watchdog Timer Enabled or Disabled.

5. Stop Instruction Enabled or converted to Halt Mode.
6. Option to enable Clock Output pin to replace PD5.
7. Option to Individually Enable Pull-ups/Interrupts on each of the eight Port A pins.
8. LVR Reset Enabled or Disabled

1.3 FUNCTIONAL PIN DESCRIPTION

The following paragraphs describe the functionality of each pin on the MC68HC05P18 package. Pins connected to subsystems described in other chapters provide a reference to the chapter instead of a detailed functional description.The pinout is shown in **Figure 1-2: User Mode Pinout**.

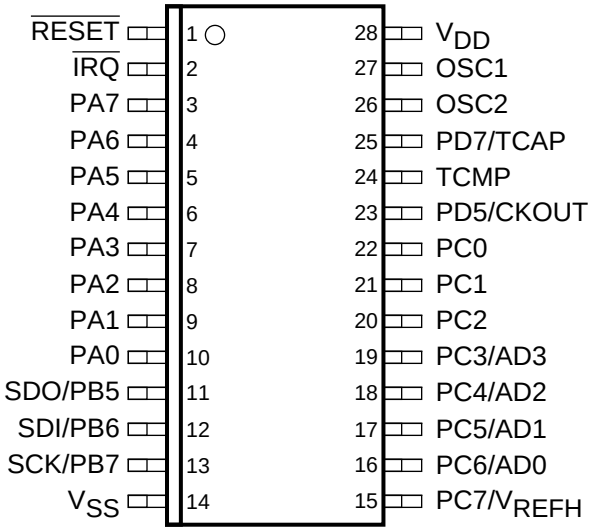


Figure 1-2: User Mode Pinout

1.3.1 V_{DD} AND V_{SS}

Power is supplied to the MCU through V_{DD} and V_{SS}. V_{DD} is connected to a regulated +5 volt supply and V_{SS} is connected to ground.

Very fast signal transitions occur on the MCU pins. The short rise and fall times place very high short-duration current demands on the power supply. To prevent noise problems, take special care to provide good power supply bypassing at the MCU. Use bypass capacitors with good high-frequency characteristics, and position them as close to the MCU as possible. Bypassing requirements vary, depending on how heavily the MCU pins are loaded.

1.3.2 OSC1, OSC2

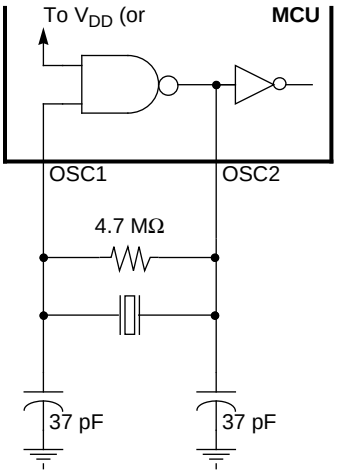
The OSC1 and OSC2 pins are the control connections for the on-chip oscillator. The OSC1 and OSC2 pins can accept the following:

1. A crystal or ceramic resonator, as shown in **Figure 1-3: Oscillator Connections(a)**
2. An external clock signal, as shown in **Figure 1-3: Oscillator Connections(b)**

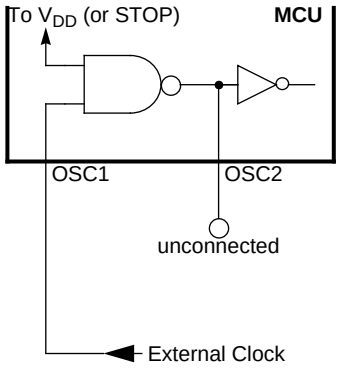
The frequency, f_{osc}, of the oscillator or external clock source is divided by two to produce the internal PH2 bus clock operating frequency, f_{OP}. The oscillator cannot be turned off by software if the Stop to Halt conversion is enabled via Mask Option.

1.3.2.1 Crystal

The circuit in **Figure 1-3: Oscillator Connections(a)** shows a typical oscillator circuit for an AT-cut, parallel resonant crystal. Follow the crystal manufacturer's recommendations, as the crystal parameters determine the external component values required to provide maximum stability and reliable start-up. The load capacitance values used in the oscillator circuit design should include all stray capacitances. Mount the crystal and components as close as possible to the pins for start-up stabilization and to minimize output distortion.



(a) Crystal or Ceramic Resonator Connections



(b) External Clock Source Connections

Figure 1-3: Oscillator Connections

1.3.2.2 Ceramic Resonator

In cost-sensitive applications, use a ceramic resonator in place of a crystal. Use the circuit in **Figure 1-3: Oscillator Connections(a)** for a ceramic resonator and follow the resonator manufacturer's recommendations, as the resonator parameters determine the external component values required for maximum stability and reliable starting. The load capacitance values used in the oscillator circuit design should include all stray capacitances. Mount the resonator and components as close as possible to the pins for start-up stabilization and to minimize output distortion.

1.3.2.3 External Clock

An external clock from another CMOS-compatible device can be connected to the OSC1 input, with the OSC2 input not connected, as shown in **Figure 1-3: Oscillator Connections(b):**

1.3.3 RESET

Driving this input low resets the MCU to a known start-up state. As an output pin, the $\overline{\text{RESET}}$ pin indicates that an internal MCU reset has occurred. The $\overline{\text{RESET}}$ pin contains an internal Schmitt trigger to improve its noise immunity. Refer to **SECTION 3: RESETS**.

1.3.4 PA0-PA7

These eight I/O pins comprise Port A. The state of any pin is software programmable and all Port A lines are configured as inputs during power-on or reset. Eight Mask Options can be chosen to enable pull-ups and interrupts (active low) on Port A pins. Refer to **SECTION 6: INPUT/OUTPUT PORTS** and **SECTION 4: INTERRUPTS**.

1.3.5 PB5/SDO, PB6/SDI, PB7/SCK

These three I/O pins comprise Port B and are shared with the SIOP communications subsystem. The state of any pin is software programmable and all Port B lines are configured as inputs during power-on or reset. Refer to **SECTION 6: INPUT/OUTPUT PORTS** and **SECTION 9: SERIAL INPUT / OUTPUT PORTS**.

1.3.6 PC0-PC2, PC3/AD3, PC4/AD2, PC5/AD1, PC6/AD0, PC7/VREFH

These eight I/O pins comprise Port C and are shared with the A/D Converter subsystem. The state of any pin is software programmable and all Port C lines are configured as inputs during power-on or reset. Port pins PC0 and PC1 are capable of sourcing and sinking high currents. Refer to **SECTION 6: INPUT/OUTPUT PORTS** and **SECTION 7: ANALOG TO DIGITAL CONVERTER**.

1.3.7 PD5/CKOUT, PD7/TCAP

These two I/O pins comprise Port D and one of them is shared with the 16-bit timer subsystem. The state of PD5/CKOUT is software programmable and is configured as an input during power-on or reset. PD7 is always an input; it may be read at any time, regardless of the mode of operation the 16-bit timer may be in. Refer to **SECTION 6: INPUT/OUTPUT PORTS** and **SECTION 8: 16-BIT TIMER**. There is a mask option to turn the PD5/CKOUT pin into a Clock Output. Clock Output is a buffered OSC2 signal with a CMOS output driver. The Clock Output or the Port D function must be chosen with the mask option and is not alterable in software.

1.3.8 TCMP

This pin is the output from the 16-bit timer's Output Compare function. It is low after reset. Refer to **SECTION 8: 16-BIT TIMER**.

1.3.9 $\overline{\text{IRQ}}$ (MASKABLE INTERRUPT REQUEST)

This input pin drives the asynchronous interrupt function of the MCU. The MCU completes the current instruction being executed before it responds to the $\overline{\text{IRQ}}$ interrupt request. When $\overline{\text{IRQ}}$ is driven low, the event is latched internally to signify an interrupt has been requested. When the MCU completes its current instruction, the interrupt latch is tested. If the interrupt latch is set, and the interrupt mask bit (I-bit) in the Condition Code Register is clear, the MCU begins the interrupt sequence.

Depending on the Mask Option selected, the $\overline{\text{IRQ}}$ pin triggers this interrupt on either a negative going edge at the $\overline{\text{IRQ}}$ pin and/or while the $\overline{\text{IRQ}}$ pin is held in the low state. In either case, the $\overline{\text{IRQ}}$ pin must be held low for at least one t_{ILIH} time period. If the Edge- and Level-Sensitive Mask Option is selected, the $\overline{\text{IRQ}}$ input requires an external resistor connected to V_{DD} for a wired-OR operation. If the $\overline{\text{IRQ}}$ pin is not used, it must be tied to the V_{DD} supply. The $\overline{\text{IRQ}}$ pin contains an internal Schmitt trigger as part of its input circuitry to improve noise immunity. Refer to **SECTION 4: INTERRUPTS**.

1.4 CPU CORE

The MC68HC05P18 uses a standard 68HC05 series CPU core. A description of the instruction set can be found in *MC68HC05P4 HCMOS Technical Data* (MC68HC05P4/D).

SECTION 2 OPERATING MODES

The MC68HC05P18 has one user mode of operation.

2.1 USER MODE

The User 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 and are not available externally. User Mode is entered on the rising edge of $\overline{\text{RESET}}$ if the $\overline{\text{IRQ}}$ pin is within the normal operating voltage range. In the User Mode, there is an 8-bit I/O port, a second 8-bit I/O port shared with the A/D subsystem, one 3-bit I/O port shared with the SIOP, and a 2-bit I/O port shared with the 16-bit timer subsystem.

2.2 LOW-POWER MODES

The MC68HC05P18 is capable of running in a low-power mode in each of its configurations. The WAIT and STOP instructions provide three modes that reduce the power required for the MCU by stopping various internal clocks and/or the on-chip oscillator. The STOP and WAIT instructions are not normally used if the COP Watchdog Timer is enabled. The Stop conversion Mask Option is used to modify the behavior of the STOP instruction from Stop Mode to Halt Mode. The flow of the Stop, Halt, and Wait modes is shown in **Figure 2-1: STOP/WAIT Flowcharts**.

2.2.1 STOP INSTRUCTION

The STOP instruction can result in one of two modes of operation depending on the Mask Option. If the Stop conversion to Halt Mask Option is not chosen, the STOP instruction behaves like a normal STOP instruction in the MC68HC05P18 family and place the MCU in the Stop Mode. If the Stop conversion to Halt option is chosen, the STOP instruction behaves like a WAIT instruction (with the exception of a brief delay at startup) and place the MCU in the Halt Mode.

2.2.1.1 Stop Mode

Execution of the STOP instruction without conversion to Halt places the MCU in its lowest power consumption mode. In the Stop Mode the internal oscillator is turned off, halting *all* internal processing, including the COP watchdog timer.The RC oscillator that feeds the EEPROM and the A/D converter is also stopped. Execution of the STOP instruction automatically clears the I-bit in the Condition Code Register so that the $\overline{\text{IRQ}}$ external interrupt is enabled. All other registers and memory remain unaltered. All input/output lines remain unchanged.

The MCU can be brought out of the Stop Mode only by an $\overline{\text{IRQ}}$ (or Port A, if selected as a mask option) external interrupt or an externally generated RESET. When exiting the

Stop Mode the internal oscillator resumes after a 4064 PH2 clock cycle oscillator stabilization delay.

NOTE: Execution of the STOP instruction without conversion to Halt (via Mask Option) causes the oscillator to stop, and therefore disable the COP watchdog timer. If the COP watchdog timer is to be used, the Stop Mode should be changed to the Halt Mode by selecting the appropriate Mask Option.

2.2.1.2 Halt Mode

Execution of the STOP instruction with the conversion to Halt places the MCU in this low-power mode. Halt Mode consumes the same amount of power as Wait Mode (both Halt and Wait Modes consume more power than Stop Mode).

In Halt Mode the PH2 clock is halted, suspending all processor and internal bus activity. Internal timer clocks remain active, permitting interrupts to be generated from the 16-bit timer or a reset to be generated from the COP watchdog timer. Execution of the STOP instruction automatically clears the I-bit in the Condition Code Register enabling the IRQ external interrupt. All other registers, memory, and input/output lines remain in their previous states.

If the 16-bit timer interrupt is enabled it causes the processor to exit the Halt Mode and resume normal operation. The Halt Mode can also be exited when an IRQ external interrupt or external RESET occurs. When exiting the Halt Mode the PH2 clock resumes after a delay of one to 4064 PH2 clock cycles. This varied delay time is the result of the Halt Mode exit circuitry testing the oscillator stabilization delay timer (a feature of the Stop Mode), which has been free-running (a feature of the Wait Mode).

NOTE: The Halt Mode is not intended for normal use. This feature is provided to keep the COP watchdog timer active in the event a STOP instruction is inadvertently executed.

2.2.2 WAIT INSTRUCTION

The WAIT instruction places the MCU in a low-power mode, which consumes more power than the STOP mode. In Wait mode, the PH2 clock is halted, suspending all processor and internal bus activity. Internal timer clocks remain active, permitting interrupts to be generated from the 16-bit timer and reset to be generated from the COP watchdog timer. Execution of the WAIT instruction automatically clears the I-bit in the Condition Code Register enabling the IRQ external interrupt. All other registers, memory, and input/output lines remain in their previous state.

If the 16-bit timer interrupt is enabled, it causes the processor to exit the Wait Mode and resume normal operation. The 16-bit timer may be used to generate a periodic exit from

the Wait Mode. The Wait Mode may also be exited when an $\overline{\text{IRQ}}$ or $\overline{\text{RESET}}$ occurs. Note that if Port A interrupts (if selected as a Mask Option), it also exits Wait Mode.

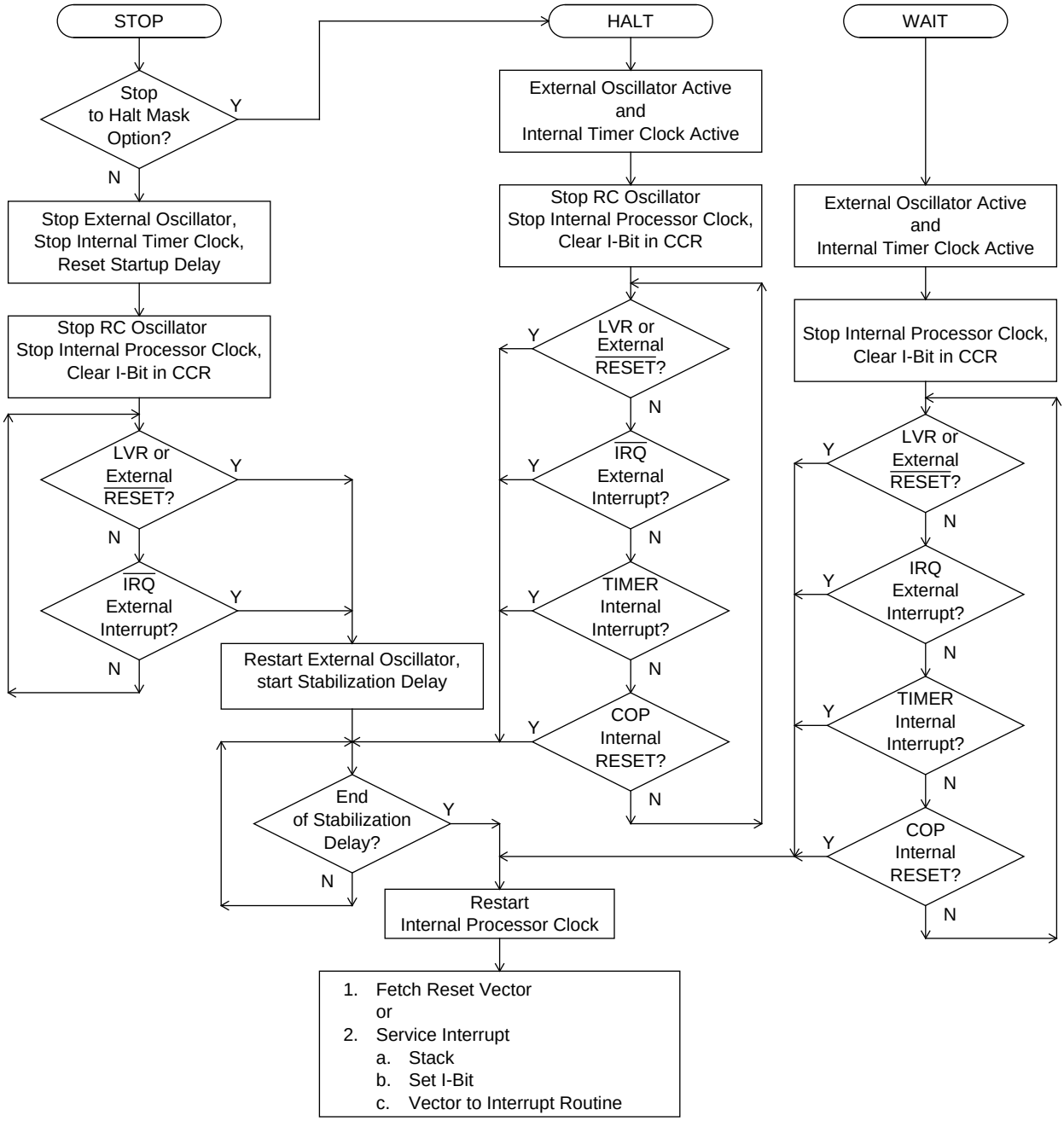


Figure 2-1: STOP/WAIT Flowcharts

2.3 COP WATCHDOG TIMER CONSIDERATIONS

The COP watchdog timer is active in the User Mode of operation when selected by Mask Option. Executing the STOP instruction without conversion to Halt (via Mask Option) causes the COP to be disabled. Therefore, it is recommended that the STOP instruction be modified to produce Halt Mode (via Mask Option) if the COP watchdog timer is enabled.

Furthermore, it is recommended that the COP watchdog timer be disabled for applications that use the Halt or Wait Modes for time periods that exceed the COP time-out period.

COP watchdog timer interactions are summarized in **Table 2-1: COP Watchdog Timer Recommendations**.

Table 2-1: COP Watchdog Timer Recommendations

IF the following conditions exist:		**THEN the COP Watchdog Timer** should be:
STOP Instruction Mode	**WAIT Period**	
Halt Mode selected via Mask Option	WAIT period *less than* COP time-out	Enable or disable COP via Mask Option
Halt Mode selected via Mask Option	WAIT period *more than* COP time-out	Disable COP via Mask Option
Stop Mode selected via Mask Option	any length WAIT period	Disable COP via Mask Option

SECTION 3 RESETS

The MCU can be reset from four sources: one external input and three internal reset conditions. The $\overline{\text{RESET}}$ pin is an input with a Schmitt trigger as shown in **Figure 3-1: Reset Block Diagram**. The CPU and all peripheral modules are reset by the RST signal, which is the logical OR of internal reset functions and is clocked by PH2.

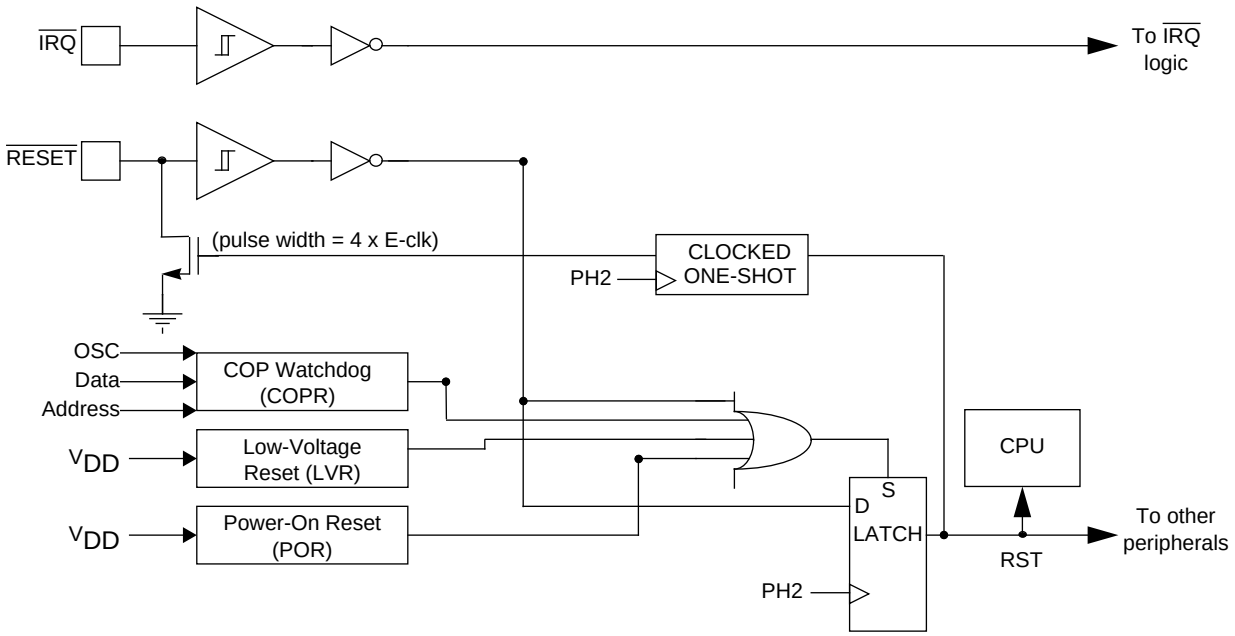


Figure 3-1: Reset Block Diagram

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

The $\overline{\text{RESET}}$ input is the only external reset and is connected to an internal Schmitt trigger. The external reset occurs whenever the $\overline{\text{RESET}}$ input is driven below the lower threshold and remains in reset until the $\overline{\text{RESET}}$ pin rises above the upper threshold. The upper and lower thresholds are given in **SECTION 10: ELECTRICAL SPECIFICATIONS**.

3.2 INTERNAL RESETS

The three internally generated resets are the initial Power-On Reset (POR), the COP watchdog timer, and Low Voltage Reset (LVR) functions.

3.2.1 POWER-ON RESET (POR)

The internal POR is generated at power-up to allow the clock oscillator to stabilize. The POR 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 PH2 clock cycle oscillator stabilization delay after the oscillator becomes active.

The POR generates the RST signal and reset the MCU. The POR also pulls the $\overline{\text{RESET}}$ pin low at the same time, allowing external devices to be reset with the MCU. If any other reset function is active at the end of this 4064 PH2 clock cycle delay, the RST signal remains active until the other reset condition(s) end.

3.2.2 COMPUTER OPERATING PROPERLY (COP) RESET

When the COP watchdog timer is enabled (by Mask Option), the internal COP reset is generated automatically by a time-out of the COP watchdog timer. This timer is implemented with an 18-stage ripple counter that provides a time-out period of 65.5 ms when a 4 MHz oscillator is used. The COP watchdog counter is cleared by writing a logical zero to bit zero at location $3FF0.

The COP register is shared with the MSB of an unimplemented User Interrupt Vector, as shown in **Figure 3-2: COP Register (COPR)**. Reading this location returns the MSB of the unimplemented User Interrupt Vector. Writing to this location clears the COP watchdog timer.

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

UNIMPLEMENTED RESERVED FOR TEST

Figure 3-2: COP Register (COPR)

3.2.3 LOW VOLTAGE RESET (LVR)

The internal LVR reset is generated when the supply voltage to the V_{DD} pin falls below a nominal 3.80 Vdc. The LVR threshold is not intended to be an accurate and stable trip point, but is intended to assure that the CPU is held in reset when the V_{DD} supply voltage is below reasonable operating limits. If the LVR is tripped for a short time, the LVR reset signal will last at least two cycles of the CPU bus clock, PH2. A Mask Option is provided to disable the LVR.

The LVR generates the RST signal, which resets the CPU and other peripherals. If any other reset function is active at the end of the LVR reset signal, the RST signal remains in the reset condition until the other reset condition(s) end.

SECTION 4 INTERRUPTS

The MCU can be interrupted six different ways:

1. Nonmaskable Software Interrupt Instruction (SWI)
2. External Asynchronous Interrupt ($\overline{\text{IRQ}}$)
3. Input Capture Interrupt (TIMER)
4. Output Compare Interrupt (TIMER)
5. Timer Overflow Interrupt (TIMER)
6. Port A Interrupt (if selected as a Mask Option).

Interrupts cause the processor to save the 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 completed.

When the current instruction is completed, the processor checks all pending hardware interrupts. If interrupts are not masked (I-bit in the Condition Code Register is clear), and the corresponding interrupt enable bit is set, the processor proceeds with interrupt processing. Otherwise, the next instruction is fetched and executed. 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 puts the register contents on the stack, sets the I-bit in the CCR, and fetches the address of the corresponding interrupt service routine from the vector table at locations $3FF0 through $3FFF. If more than one interrupt is pending when the interrupt vector is fetched, the interrupt with the highest vector location (shown in **Table 4-1: Vector Addresses for Interrupts and Reset**) is serviced first.

Table 4-1: Vector Addresses for Interrupts and Reset

Register	Flag Name	Interrupts	CPU Interrupt	Vector Address
N/A	N/A	Reset	RESET	$3FFE-$3FFF
N/A	N/A	Software	SWI	$3FFC-$3FFD
N/A	N/A	External Interrupt	IRQ	$3FFA-$3FFB
TSR	ICF	Timer Input Capture	TIMER	$3FF8-$3FF9
TSR	OCF	Timer Output Compare	TIMER	$3FF8-$3FF9
TSR	TOF	Timer Overflow	TIMER	$3FF8-$3FF9
N/A	N/A	unimplemented	N/A	$3FF6-$3FF7
N/A	N/A	unimplemented	N/A	$3FF4-$3FF5
N/A	N/A	unimplemented	N/A	$3FF2-$3FF3
N/A	N/A	unimplemented	N/A	$3FF0-$3FF1

An RTI instruction is used to signify when the interrupt software service routine is completed. The RTI instruction causes the CPU state 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: Interrupt Processing Flowchart** shows the sequence of events that occur during interrupt processing.

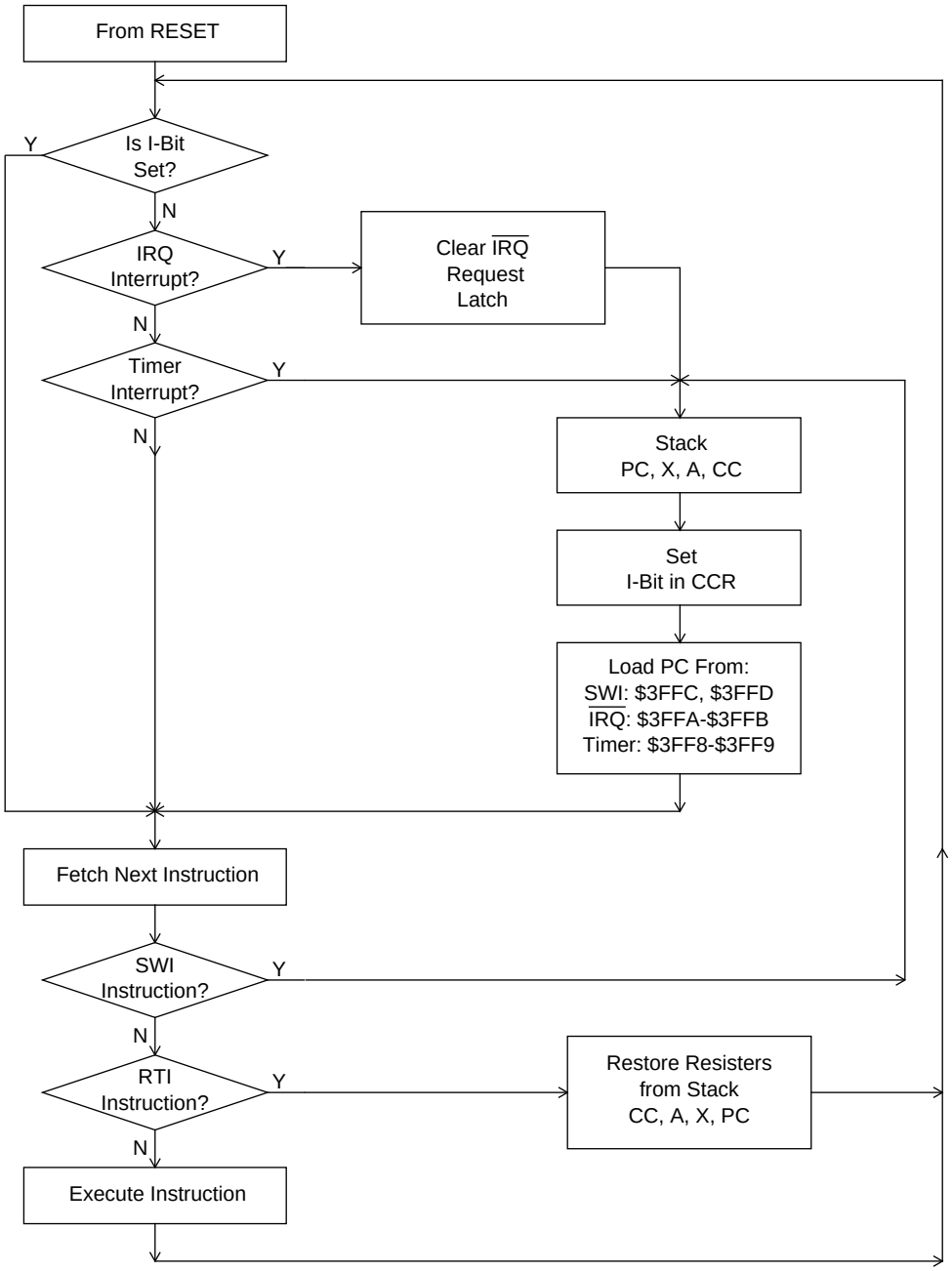


Figure 4-1: Interrupt Processing Flowchart

4.1 INTERRUPT TYPES

The interrupts fall into three categories: reset, software, and hardware.

4.1.1 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: Interrupt Processing Flowchart**. A low level input on the $\overline{\text{RESET}}$ pin or internally generated RST signal causes the program to vector to its starting address, which is specified by the contents of memory locations $3FFE and $3FFF. 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 previously described in **SECTION 3: RESETS**.

4.1.2 SOFTWARE INTERRUPT (SWI)

The SWI is an executable instruction. It is also a non-maskable interrupt since it is executed regardless of the state of the I-bit in the CCR. As with any instruction, interrupts pending during the previous instruction are serviced before the SWI opcode is fetched. The interrupt service routine address for the SWI instruction is specified by the contents of memory locations $3FFC and $3FFD.

4.1.3 HARDWARE INTERRUPTS

All hardware interrupts 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 four hardware interrupts, which are explained in the following sections.

4.1.3.1 External Interrupt ($\overline{\text{IRQ}}$)

The $\overline{\text{IRQ}}$ pin drives an asynchronous interrupt to the CPU. An edge detector flip-flop is latched on the falling edge of $\overline{\text{IRQ}}$. If either the output from the internal edge detector flip-flop or the level on the $\overline{\text{IRQ}}$ pin is low, a request is synchronized to the CPU to generate the IRQ interrupt. If the Edge-Sensitive Only Mask Option is selected, the output of the internal edge detector flip-flop is sampled and the input level on the $\overline{\text{IRQ}}$ pin is ignored. If Port A interrupts are selected as a Mask Option, a Port A interrupt uses the same vector. The interrupt service routine address is specified by the contents of memory locations $3FFA and $3FFB.

NOTE: The internal interrupt latch is cleared 9 PH2 clock cycles after the interrupt is recognized (after location $3FFA is read). Therefore, another external interrupt pulse could be latched during the IRQ service routine.

NOTE: When the Edge- and Level-Sensitive Mask Option is selected, the voltage applied to the $\overline{\text{IRQ}}$ pin must return to the high state before the RTI instruction in the interrupt service routine is executed.

4.1.3.2 Input Capture Interrupt

The input capture interrupt is generated by the 16-bit timer as described in **SECTION 8: 16-BIT TIMER**. The input capture interrupt flag is located in register TSR and its corresponding enable bit can be found in register TCR. The I-bit in the CCR must be clear in order for the input capture interrupt to be enabled. The interrupt service routine address is specified by the contents of memory locations $3FF8 and $3FF9.

4.1.3.3 Output Compare Interrupt

The output compare interrupt is generated by the 16-bit timer as described in **SECTION 8: 16-BIT TIMER.** The output compare interrupt flag is located in register TSR and its corresponding enable bit can be found in register TCR. The I-bit in the CCR must be clear in order for the output compare interrupt to be enabled. The interrupt service routine address is specified by the contents of memory locations $3FF8 and $3FF9.

4.1.3.4 Timer Overflow Interrupt

The timer overflow interrupt is generated by the 16-bit timer as described in **SECTION 8: 16-BIT TIMER**. The timer overflow interrupt flag is located in register TSR and its corresponding enable bit can be found in register TCR. The I-bit in the CCR must be clear in order for the timer overflow interrupt to be enabled. This internal interrupt will vector to the interrupt service routine located at the address specified by the contents of memory locations $3FF8 and $3FF9.

SECTION 5 MEMORY

The MC68HC05P18 utilizes 14 address lines to access an internal memory space covering 16K bytes. This memory space is divided into I/O, RAM, EEPROM, and ROM areas.

5.1 USER MODE MEMORY MAP

When the MC68HC05P18 is in the User Mode the 32 bytes of I/O, 192 bytes of RAM, 128 bytes of EEPROM, 8000 bytes of User ROM, 48 bytes of User Page Zero ROM, and 16 bytes of User Vectors ROM are all active, as shown in **Figure 5-1: MC68HC05P18 User Mode Memory Map**.

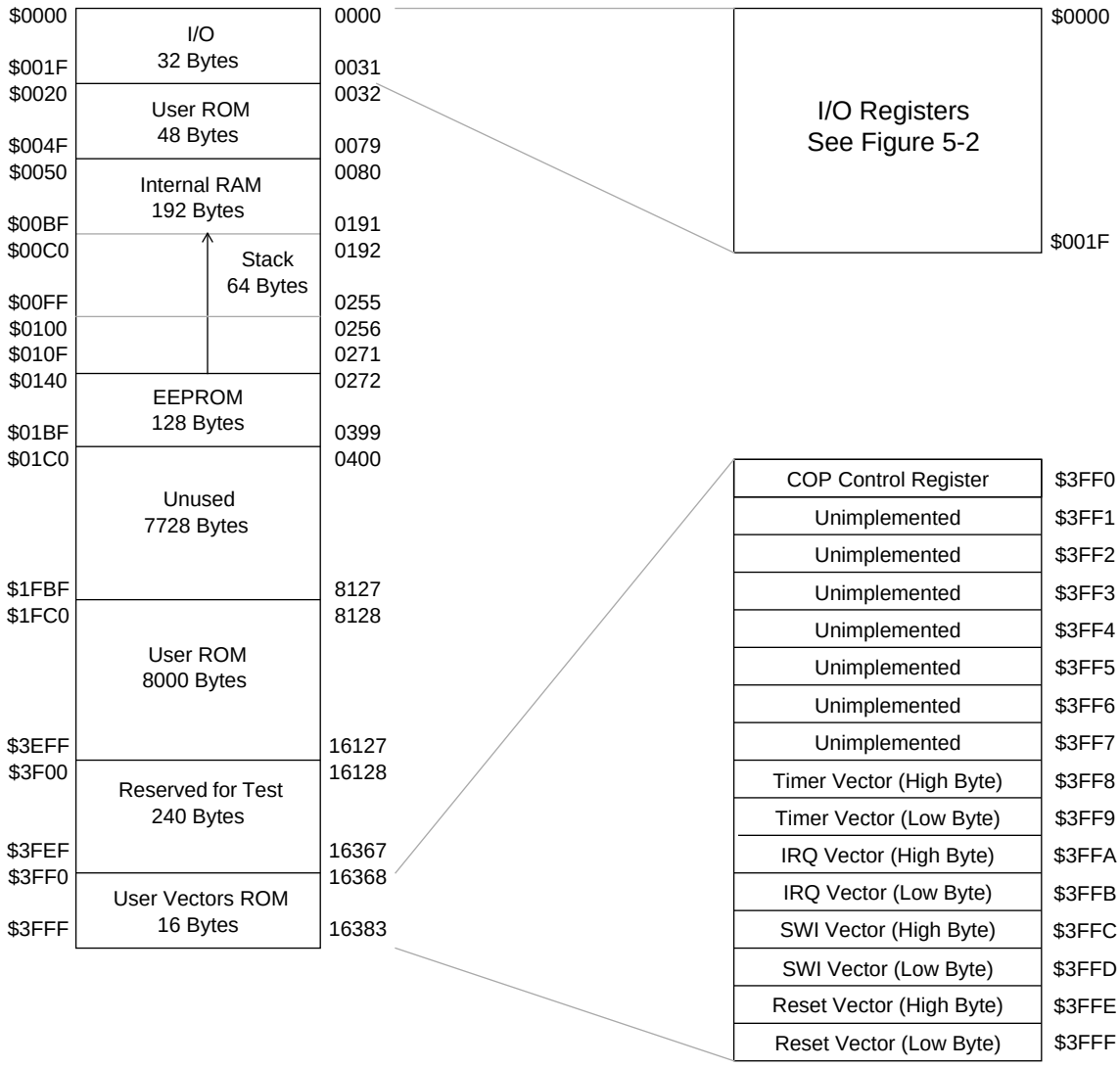


Figure 5-1: MC68HC05P18 User Mode Memory Map

5.2 I/O AND CONTROL REGISTERS

Figure 5-2: MC68HC05P18 I/O and Control Registers Memory Map through **Figure 5-4: MC68HC05P18 I/O and Control Registers $0010-$001F** briefly describe the I/O and Control Registers at locations $0000-$001F. Reading unimplemented bits returns unknown states, and writing unimplemented bits is ignored.

Port A Data Register	$0000
Port B Data Register	$0001
Port C Data Register	$0002
Port D Data Register	$0003
Port A Data Direction Register	$0004
Port B Data Direction Register	$0005
Port C Data Direction Register	$0006
Port D Data Direction Register	$0007
Unimplemented	$0008
Unimplemented	$0009
SIOP Control Register	$000A
SIOP Status Register	$000B
SIOP Data Register	$000C
Unimplemented	$000D
Unimplemented	$000E
Unimplemented	$000F
Unimplemented	$0010
Unimplemented	$0011
Timer Control Register	$0012
Timer Status Register	$0013
Input Capture MSB	$0014
Input Capture LSB	$0015
Output Compare MSB	$0016
Output Compare LSB	$0017
Timer MSB	$0018
Timer LSB	$0019
Alternate Counter MSB	$001A
Alternate Counter LSB	$001B
EEPROM Programming Register	$001C
A/D Converter Data Register	$001D
A/D Converter Control & Status Reg	$001E
Reserved for Test	$001F

Figure 5-2: MC68HC05P18 I/O and Control Registers Memory Map

ADDR	REGISTER	READ / WRITE	7	6	5	4	3	2	1	0
$0000	PORT A DATA PORTA	R	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
		W								
$0001	PORT B DATA PORTB	R	PB7	PB6	PB5	0	0	0	0	0
		W								
$0002	PORT C DATA PORTC	R	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
		W								
$0003	PORT D DATA PORTD	R	PD7	0	PD5	1	0	0	0	0
		W								
$0004	PORT A DATA DIRECTION DDRA	R	DDRA7	DDRA6	DDRA5	DDRA4	DDRA3	DDRA2	DDRA1	DDRA0
		W								
$0005	PORT B DATA DIRECTION DDRB	R	DDRB7	DDRB6	DDRB5	1	1	1	1	1
		W								
$0006	PORT C DATA DIRECTION DDRC	R	DDRC7	DDRC6	DDRC5	DDRC4	DDRC3	DDRC2	DDRC1	DDRC0
		W								
$0007	PORT D DATA DIRECTION DDRD	R	0	0	DDRD5	0	0	0	0	0
		W								
$0008	UNIMPLEMENTED	R								
		W								
$0009	UNIMPLEMENTED	R								
		W								
$000A	SIOP CONTROL REGISTER SCR	R	0	SPE	0	MSTR	0	0	0	0
		W								
$000B	SIOP STATUS REGISTER SSR	R	SPIF	DCOL	0	0	0	0	0	0
		W								
$000C	SIOP DATA REGISTER SDR	R	SDR7	SDR6	SDR5	SDR4	SDR3	SDR2	SDR1	SDR0
		W								
$000D	RESERVED FOR TEST	R								
		W								
$000E	UNIMPLEMENTED	R								
		W								
$000F	UNIMPLEMENTED	R								
		W								

UNIMPLEMENTED RESERVED FOR TEST

Figure 5-3: MC68HC05P18 I/O and Control Registers $0000-$000F

ADDR	REGISTER	READ / WRITE	7	6	5	4	3	2	1	0
$0010	UNIMPLEMENTED	R								
		W								
$0011	UNIMPLEMENTED	R								
		W								
$0012	TIMER CONTROL REGISTER TCR	R	ICIE	OCIE	TOIE	0	0	0	IEDG	OLVL
		W								
$0013	TIMER STATUS REGISTER TSR	R	ICF	OCF	TOF	0	0	0	0	0
		W								
$0014	INPUT CAPTURE MSB ICRH	R	ICRH7	ICRH6	ICRH5	ICRH4	ICRH3	ICRH2	ICRH1	ICRH0
		W								
$0015	INPUT CAPTURE LSB ICRL	R	ICRL7	ICRL6	ICRL5	ICRL4	ICRL3	ICRL2	ICRL1	ICRL0
		W								
$0016	OUTPUT COMPARE MSB OCRH	R	OCRH7	OCRH6	OCRH5	OCRH4	OCRH3	OCRH2	OCRH1	OCRH0
		W								
$0017	OUTPUT COMPARE LSB OCRL	R	OCRL7	OCRL6	OCRL5	OCRL4	OCRL3	OCRL2	OCRL1	OCRL0
		W								
$0018	TIMER MSB TMRH	R	TMRH7	TMRH6	TMRH5	TMRH4	TMRH3	TMRH2	TMRH1	TMRH0
		W								
$0019	TIMER LSB TMRL	R	TMRL7	TMRL6	TMRL5	TMRL4	TMRL3	TMRL2	TMRL1	TMRL0
		W								
$001A	ALTERNATE COUNTER MSB ACRH	R	ACRH7	ACRH6	ACRH5	ACRH4	ACRH3	ACRH2	ACRH1	ACRH0
		W								
$001B	ALTERNATE COUNTER LSB ACRL	R	ACRL7	ACRL6	ACRL5	ACRL4	ACRL3	ACRL2	ACRL1	ACRL0
		W								
$001C	EEPROM PROGRAMMING REGISTER	R	0	CPEN	0	ER1	ER0	LATCH	EERC	EEPGM
		W								
$001D	A/D CONVERSION DATA ADC	R	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0
		W								
$001E	A/D STATUS AND CONTROL ADSC	R	CC	Reserved	ADON	0	0	CH2	CH1	CH0
		W								
$001F	RESERVED FOR TEST	R								
		W								

UNIMPLEMENTED RESERVED FOR TEST

Figure 5-4: MC68HC05P18 I/O and Control Registers $0010-$001F

5.3 RAM

The user RAM consists of 192 bytes (including the stack) at locations $0050 through $010F. The stack begins at address $00FF. The stack pointer can access 64 bytes of RAM from $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.

5.4 ROM

There are 8064 bytes of user ROM available, consisting of 8000 bytes at locations $1FC0 through $3EFF, 48 bytes in page zero locations $0020 through $004F, and 16 additional bytes for user vectors at locations $3FF0 through $3FFF.

NOTE: Address space $3F00 through $3FEF is reserved for test code. Unlike other M68HC05 devices, the MC68HC05P18 does not contain self-check code.

5.5 EEPROM

The EEPROM is located at address $0140 and consists of 128 bytes. Programming the EEPROM can be done by the user on a single byte basis by manipulating the Programming Register, located at address $001C.

5.5.1 PROGRAMMING REGISTER $1C

The contents and use of the programming register are discussed below.

		7	6	5	4	3	2	1	0
EEPROG $001C	R	0	CPEN	0	ER1	ER0	LATCH	EERC	EEPGM
	W								
reset ⇒		0	0	0	0	0	0	0	0

5.5.1.1 CPEN - Charge Pump Enable

When set, CPEN enables the charge pump that produces the internal EEPROM programming voltage. This bit should be set concurrently with the LATCH bit. The programming voltage will not be available until EEPGM is set. The charge pump should be disabled when not in use. CPEN is readable and writable and is cleared by reset.

5.5.1.2 ER1:ER0 - Erase Select Bits

ER1 and ER0 form a 2-bit field that is used to select one of three erase modes: byte, block, or bulk. **Table 5-1: Erase Mode Select** shows the modes selected for each bit configuration. These bits are readable and writable and are cleared by reset.

In byte erase mode, only the selected byte is erased. In block mode, a 32-byte block of EEPROM is erased. The EEPROM memory space is divided into four 32-byte blocks ($140-$15F, $160-$17F, $180-$19F, $1A0-$1BF), and doing a block erase to any address within a block erases the entire block. In bulk erase mode, the entire 128 byte EEPROM section is erased.

Table 5-1: Erase Mode Select

ER1	ER0	Mode
0	0	Program (no Erase)
0	1	Byte Erase
1	0	Block Erase
1	1	Bulk Erase

5.5.1.3 LATCH

When set, LATCH configures the EEPROM address and data bus for programming. When LATCH is set, writes to the EEPROM array cause the data bus and the address bus to be latched. This bit is readable and writable, but reads from the array are inhibited if the LATCH bit is set and a write to the EEPROM space has taken place. When clear, address and data buses are configured for normal operation. Reset clears this bit.

5.5.1.4 EERC - EEPROM RC Oscillator Control

When this bit is set, the EEPROM section uses the internal RC oscillator instead of the CPU clock. The RC oscillator is shared with the A/D converter, so this bit should be set by the user when the internal bus frequency is below 1.5 MHz to guarantee reliable operation of the EEPROM or A/D converter. After setting the EERC bit, delay a time t_{RCON} to allow the RC oscillator to stabilize. This bit is readable and writable. The EERC bit is cleared by Reset. The RC oscillator is disabled while the MCU is in STOP mode.

5.5.1.5 EEPGM - EEPROM Programming Power Enable

EEPGM must be written to enable (or disable) the EEPGM function. When set, EEPGM turns on the charge pump and enables the programming (or erasing) power to the EEPROM array. When clear, this power is switched off. This enables pulsing of the programming voltage to be controlled internally. This bit can be read at any time, but can only be written to if LATCH = 1. If LATCH is not set, EEPGM cannot be set. LATCH and EEPGM cannot both be set with one write if LATCH is cleared. EEPGM is automatically cleared when LATCH is cleared. Reset clears this bit.

5.5.2 PROGRAMMING/ERASING PROCEDURES

To program a byte of EEPROM, set LATCH = CPEN = 1, set ER1 = ER0 = 0, write data to the desired address and then set EEPGM for a time t_{EPGM}.

NOTE: Any bit should be erased before it is programmed. However, if write/ erase cycling is a concern, a procedure can be followed to minimize the cycling of each bit in each EEPROM byte. Here is the procedure:

If $PB \bullet \overline{EB} = 0$, program the new data over the existing data without erasing it first

If $PB \bullet \overline{EB} \neq 0$, erase byte before programming

Where PB = Byte data to be programmed and EB = Existing EEPROM byte data.

To erase a **byte** of EEPROM set LATCH = 1, CPEN = 1, ER1 = 0 and ER0 = 1, write to the address to be erased, and set EEPGM for a time t_{EBYT}.

To erase a **block** of EEPROM set LATCH = 1, CPEN = 1, ER1 = 1 and ER0 = 0, write to any address in the block, and set EEPGM for a time t_{EBLOCK}.

For a **bulk** erase set LATCH = 1, CPEN = 1, ER1 = 1, and ER0 = 1, write to any address in the array, and set EEPGM for a time t_{EBULK}.

To terminate the programming or erase sequence, clear EEPGM, delay for a time t_{FPV} to allow the program voltage to fall, and then clear LATCH and CPEN to free up the buses. Following each erase or programming sequence, clear all programming control bits.

SECTION 6 INPUT/OUTPUT PORTS

In the User Mode, there are 20 bidirectional I/O lines arranged as two 8-bit I/O ports (Ports A and C), one 3-bit I/O port (Port B), and one 1-bit I/O port (Port D). These ports are programmable as either inputs or outputs under software control of the data direction registers (DDRs). There is also an input-only pin associated with Port D.

6.1 PORT A

Port A is an 8-bit bidirectional port which can share its pins with the IRQ interrupt system, as shown in **Figure 6-1: Port A I/O Circuitry**. Each Port A pin is controlled by the corresponding bits in a data direction register and a data register. The Port A Data Register is located at address $0000. The Port A Data Direction Register (DDRA) is located at address $0004. Reset clears the DDRA thereby initializing Port A as an input port. The Port A Data Register is unaffected by reset.

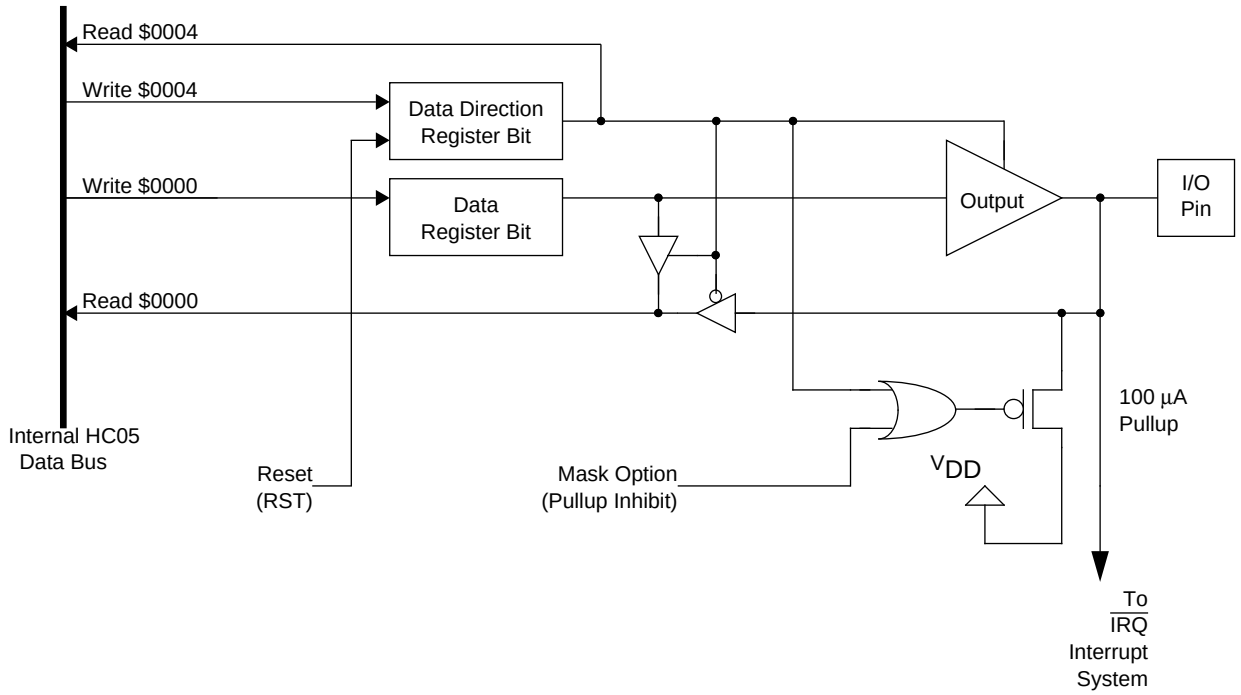


Figure 6-1: Port A I/O Circuitry

6.2 PORT B

Port B is a 3-bit bidirectional port that can share pins PB5-PB7 with the SIOP communications subsystem. The Port B data register is located at address $0001 and its Data Direction Register (DDR) is located at address $0005. Reset does not affect the data registers, but clears the DDRs, thereby setting all of the port pins to input mode. Writing

a one to a DDR bit sets the corresponding port pin to output mode (see **Figure 6-2: Port B I/O Circuitry**).

Port B may be used for general I/O applications when the SIOP subsystem is disabled. The SPE bit in register SPCR is used to enable/disable the SIOP subsystem. When the SIOP subsystem is enabled Port B registers are still accessible to software. Writing to either of the Port B registers while a data transfer is underway could corrupt the data. See **SECTION 9: SERIAL INPUT / OUTPUT PORT** for a discussion of the SIOP subsystem.

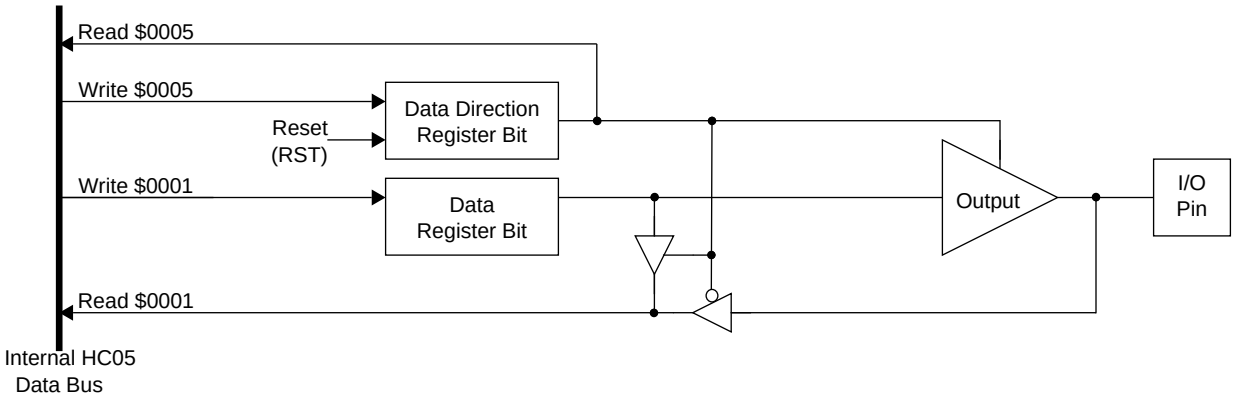


Figure 6-2: Port B I/O Circuitry

6.3 PORT C

Port C is an 8-bit bidirectional port that can share pins PC3-PC7 with the A/D subsystem. The Port C data register is located at address $0002 and its Data Direction Register (DDR) is located at address $0006. Reset does not affect the data registers, but clears the DDRs, thereby setting all of the port pins to input mode. Writing a one to a DDR bit sets the corresponding port pin to output mode (see **Figure 6-3: Port C I/O Circuitry**). Two Port C pins, PC0 and PC1, can source and sink a higher current than a typical I/O pin. See **SECTION 10: ELECTRICAL SPECIFICATIONS** regarding current specifications.

Port C may be used for general I/O applications when the A/D subsystem is disabled. The ADON bit in register ADSC is used to enable/disable the A/D subsystem. Care must be exercised when using pins PC0-PC2 while the A/D subsystem is enabled. Accidental changes to bits that affect pins PC3-PC7 in the data or DDR registers will produce unpredictable results in the A/D subsystem. See **SECTION 7: ANALOG TO DIGITAL CONVERTER**.

.

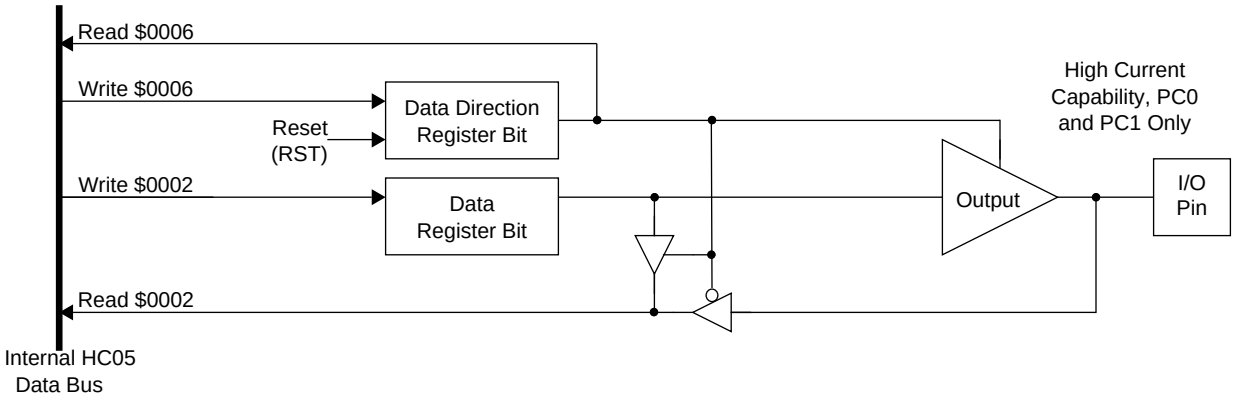


Figure 6-3: Port C I/O Circuitry

6.4 PORT D

Port D is a 2-bit port with one bidirectional pin (PD5/CKOUT) and one input-only pin (PD7). Pin PD7 is shared with the 16-bit timer. There is a Mask Option to have PD5 replaced with the clock output. The Port D data register is located at address $0003 and its Data Direction Register (DDR) is located at address $0007. Reset does not affect the data registers, but clears the DDRs, thereby setting PD5/CKOUT to input mode. Writing a one to DDR bit 5 sets PD5/CKOUT to output mode (see **Figure 6-4: Port D I/O Circuitry**).

Port D may be used for general I/O applications regardless of the state of the 16-bit timer. Since PD7 is an input-only line, its state can be read from the Port D data register at any time.

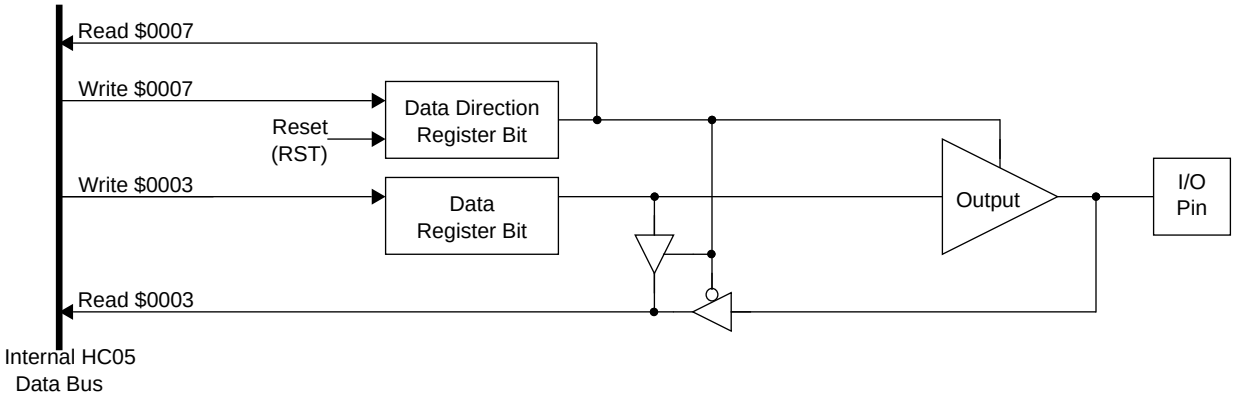


Figure 6-4: Port D I/O Circuitry

6.5 I/O PORT PROGRAMMING

Each pin on Ports A-D (except pin 7 of Port D) may be programmed as an input or an output under software control as shown in **Table 6-1: Port A I/O Pin Functions**, **Table 6-2: Port B I/O Pin Functions**, **Table 6-3: Port C I/O Pin Functions** and **Table 6-4: Port D I/O Pin Functions**. The direction of a pin is determined by the state of its corresponding bit in the associated port Data Direction Register (DDR). A 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.

Table 6-1: Port A I/O Pin Functions

DDRA	I/O Pin Mode	Accesses to DDRA @ $0004	Accesses to Data Register @ $0000	
		Read/Write	Read	Write
0 1	IN, Hi-Z OUT	DDRA0-7 DDRA0-7	I/O Pin PA0-7	* PA0-7

* Does not affect input, but stored to Data Register

Table 6-2: Port B I/O Pin Functions

DDRB	I/O Pin Mode	Accesses to DDRB @ $0005	Accesses to Data Register @ $0001	
		Read/Write	Read	Write
0 1	IN, Hi-Z OUT	DDRB5-7 DDRB5-7	I/O Pin PB5-7	* PB5-7

* Does not affect input, but stored to Data Register

Table 6-3: Port C I/O Pin Functions

DDRC	I/O Pin Mode	Accesses to DDRC @ $0006	Accesses to Data Register @ $0002	
		Read/Write	Read	Write
0 1	IN, Hi-Z OUT	DDRC0-7 DDRC0-7	I/O Pin PC0-7	* PC0-7

*Does not affect input, but stored to Data Register

Table 6-4: Port D I/O Pin Functions

DDRD	I/O Pin Mode	Accesses to DDRD @ $0007	Accesses to Data Register @ $0003	
		Read/Write	Read	Write
0	IN, Hi-Z	DDRD5	I/O Pin	*
1	OUT	DDRD5	PD5/CKOUT	PD5/CKOUT

***** Does not affect input, but stored to Data Register
* PD7 is input-only.

NOTE: To avoid generating a glitch on an I/O port pin, data should be written to the I/O port data register before writing a logical one to the corresponding Data Direction Register.

THIS PAGE INTENTIONALLY LEFT BLANK

SECTION 7 ANALOG TO DIGITAL CONVERTER

The MC68HC05P18 includes a 4-channel, multiplexed input, 8-bit, successive approximation A/D converter. The A/D subsystem shares its inputs with Port C pins PC3-PC7.

7.1 ANALOG SECTION

The following paragraphs describe the operation and performance of analog modules within the analog subsystem.

7.1.1 RATIOMETRIC CONVERSION

The A/D converter is ratiometric, with pin V_{REFH} supplying the high reference voltage. Applying an input voltage equal to V_{REFH} produces a conversion result of $FF (full scale). Applying an input voltage equal to V_{SS} produces a conversion result of $00. An input voltage greater than V_{REFH} converts to $FF with no overflow indication. For ratiometric conversions, V_{REFH} should be at the same potential as the supply voltage being used by the analog signal being measured and be referenced to V_{SS}.

7.1.2 V_{REFH}

The reference supply for the A/D converter shares pin PC7 with Port C. The low reference is tied to the V_{SS} pin internally. V_{REFH} can be any voltage between V_{SS} and V_{DD}; however, the accuracy of conversions is tested and guaranteed only for $V_{REFH} = V_{DD}$.

7.1.3 ACCURACY AND PRECISION

The 8-bit conversion result is accurate to within ± 1 1/2 LSB, including quantization; however, the accuracy of conversions is tested and guaranteed only with external oscillator operation.

7.2 CONVERSION PROCESS

The A/D reference inputs are applied to a precision digital-to-analog converter. Control logic drives the D/A and the analog output is successively compared to the selected analog input that was sampled at the beginning of the conversion cycle. The conversion process is monotonic and has no missing codes.

7.3 DIGITAL SECTION

The following paragraphs describe the operation and performance of digital modules within the analog subsystem.

7.3.1 CONVERSION TIMES

Each input conversion requires 32 PH2 clock cycles, which must be at a frequency equal to or greater than 1 MHz.

7.3.2 INTERNAL VS. EXTERNAL OSCILLATOR

If the MCU PH2 clock frequency is less than 1 MHz (2 MHz external oscillator), the internal RC oscillator (approximately 1.5 MHz) must be used for the A/D converter clock. The internal RC clock is selected by setting the EERC bit in the EEPROG register. Note that the RC oscillator is shared with the EEPROM module. The RC oscillator is disabled while the MCU is in STOP mode.

When the internal RC oscillator is being used these limitations apply:

Since the internal RC oscillator is running asynchronously with respect to the PH2 clock, the Conversion Complete (CC) bit in register ADSC must be used to determine when a conversion sequence has been completed.

1. Electrical noise slightly degrades the accuracy of the A/D converter. The A/D converter is synchronized to read voltages during the quiet period of the clock driving it. Since the internal and external clocks are not synchronized, the A/D converter occasionally measures an input when the external clock is making a transition.
2. If the PH2 clock is 1 MHz or greater (for example, external oscillator 2 MHz or greater), the internal RC oscillator should be turned off and the external oscillator used as the conversion clock.

7.3.3 MULTI-CHANNEL OPERATION

An input multiplexer allows the A/D converter to select from one of four external analog signals. Port C pins PC3 thru PC6 are shared with the inputs to the multiplexer.

7.4 A/D STATUS AND CONTROL REGISTER (ADSC)

The ADSC register reports the completion of A/D conversion and provides control over oscillator selection, analog subsystem power, and input channel selection. See **Figure 7-1: A/D Status and Control Register (ADSC)**.

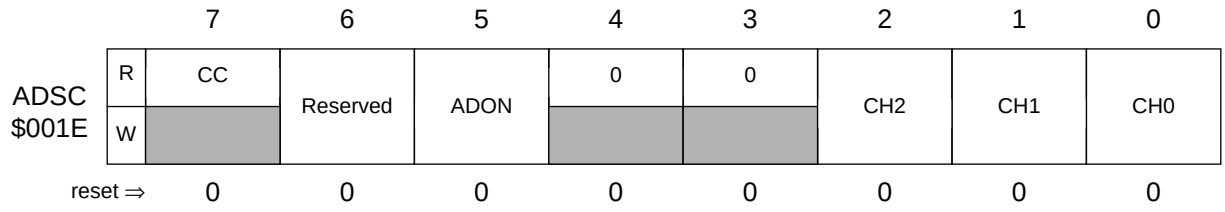


Figure 7-1: A/D Status and Control Register (ADSC)

7.4.1 CC - CONVERSION COMPLETE

This read-only status bit is set when a conversion sequence has completed and data is ready to be read from the ADC register. CC is cleared when a channel is selected for conversion, when data is read from the ADC register, or when the A/D subsystem is turned off. Once a conversion is started, conversions of the selected channel continue every 32 PH2 clock cycles until the ADSC register is written to again. During continuous conversion operation, the ADC register is updated with new data, and the CC bit is set, every 32 PH2 clock cycles. Also, data from the previous conversion is overwritten regardless of the state of the CC bit.

7.4.2 Reserved

This bit is not currently used. It can be read or written, but does not control anything.

7.4.3 ADON - A/D SUBSYSTEM ON

When the A/D subsystem is turned on (ADON = 1), it requires a time t_{ADON} to stabilize before accurate conversion results can be attained.

7.4.4 CH2-CH0 - CHANNEL SELECT BITS

CH2, CH1, and CH0 form a 3-bit field that is used to select an input to the A/D converter. Channels 0-3 correspond to Port C input pins PC6-PC3. Channels 4-6 are used for reference measurements. In User Mode, channel 7 is reserved. If a conversion is attempted with channel 7 selected, the result is $00. **Table 7-1: A/D Multiplexer Input Channel Assignments** lists the inputs selected by bits CH0-CH3.

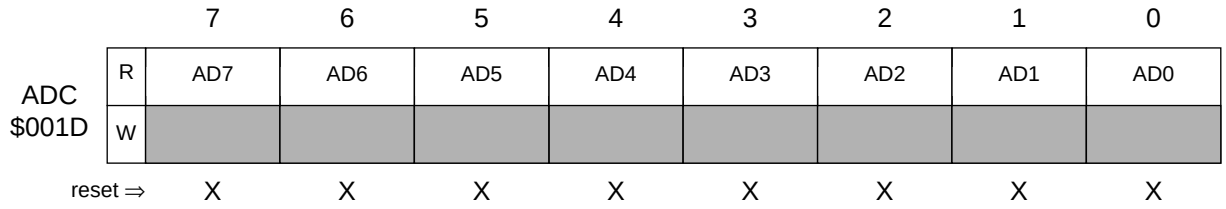
If the ADON bit is set, and an input from channels 0-4 is selected, the corresponding Port C pin's DDR bit is cleared (making that Port C pin an input). If the Port C data register is read while the A/D is on, and one of the shared input channels is selected using bit CH0-CH2, the corresponding Port C pin reads as a logic zero. The remaining Port C pins read normally. To digitally read a Port C pin, the A/D subsystem must be disabled (ADON = 0), or input channel 5-7 must be selected.

Table 7-1: A/D Multiplexer Input Channel Assignments

Channel	Signal
0	AD0 PORT C BIT 6
1	AD1 PORT C BIT 5
2	AD2 PORT C BIT 4
3	AD3 PORT C BIT 3
4	V_{REFH} PORT C BIT 7
5	$(V_{REFH} + V_{SS})/2$
6	V_{SS}
7	Reserved for Testing

7.5 A/D CONVERSION DATA REGISTER (ADC)

This register contains the output of the A/D converter. See **Figure 7-2: A/D Conversion Value Data Register (ADC)**.



		7	6	5	4	3	2	1	0
ADC $001D	R	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0
	W								
reset ⇒		X	X	X	X	X	X	X	X

Figure 7-2: A/D Conversion Value Data Register (ADC)

7.6 A/D SUBSYSTEM OPERATION DURING WAIT/HALT MODES

The A/D subsystem continues normal operation during Wait and Halt Modes. To decrease power consumption during Wait or Halt, the ADON bit in the ADSC register and the EERC bit in the EEPROG register should be cleared if the A/D subsystem is not being used.

7.7 A/D SUBSYSTEM OPERATION DURING STOP MODE

When the Stop Mode is enabled, execution of the STOP instruction terminates all A/D subsystem functions. Any pending conversion is aborted. When the oscillator resumes operation upon leaving the Stop Mode, a finite amount of time passes before the A/D subsystem stabilizes sufficiently to provide conversions at its rated accuracy. The delays built into the MC68HC05P18 when coming out of Stop Mode are sufficient for this purpose. No explicit delays need to be added to the application software.

SECTION 8 16-BIT TIMER

The MC68HC05P18 MCU contains a single 16-bit programmable timer with an Input Capture function and an Output Compare function. The 16-bit timer is driven by the output of a fixed divide-by-four prescaler operating from the PH2 clock. The 16-bit timer may be used for many applications including input waveform measurement while simultaneously generating an output waveform. Pulse widths can vary from microseconds to seconds depending on the oscillator frequency selected. The 16-bit timer is also capable of generating periodic interrupts. See **Figure 8-1: 16-Bit Timer Block Diagram**.

Because the timer has a 16-bit architecture, each function is represented by two registers. Each register pair contains the high and low byte of that function. 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 Condition Code Register (CCR) should be set while manipulating both the high and low byte registers of a specific timer function. This prevents interrupts from occurring between the time that the high and low bytes are accessed.

8.1 TIMER

The key element of the programmable timer is a 16-bit free-running counter, or Timer Registers, preceded by a prescaler, which divides the PH2 clock by four. The prescaler gives the timer a resolution of 2.0 microseconds when a 4 MHz crystal is used. The counter is incremented to increasing values during the low portion of the PH2 clock cycle.

The double byte free-running counter can be read from either of two locations: the Timer Registers (TMRH, TMRL) or the Alternate Counter Registers (ACRH, ACRL). Both locations will contain identical values. A read sequence containing only a read of the LSB of the counter (TMRL / ACRL) returns the count value at the time of the read. If a read of the counter accesses the MSB first (TMRH / ACRH), it causes the LSB (TMRL / ACRL) to be transferred to a buffer. This buffer value remains fixed after the first MSB byte read even if the MSB is read several times. The buffer is accessed when reading the counter LSB (TMRL / ACRL), and thus completes a read sequence of the total counter value. When reading either the Timer or Alternate Counter registers, if the MSB is read, the LSB must also be read to complete the read sequence. See **Figure 8-2: Timer Registers (TMRH / TMRL)** and **Figure 8-3: Alternate Counter Registers (ACRH / ACRL)**.

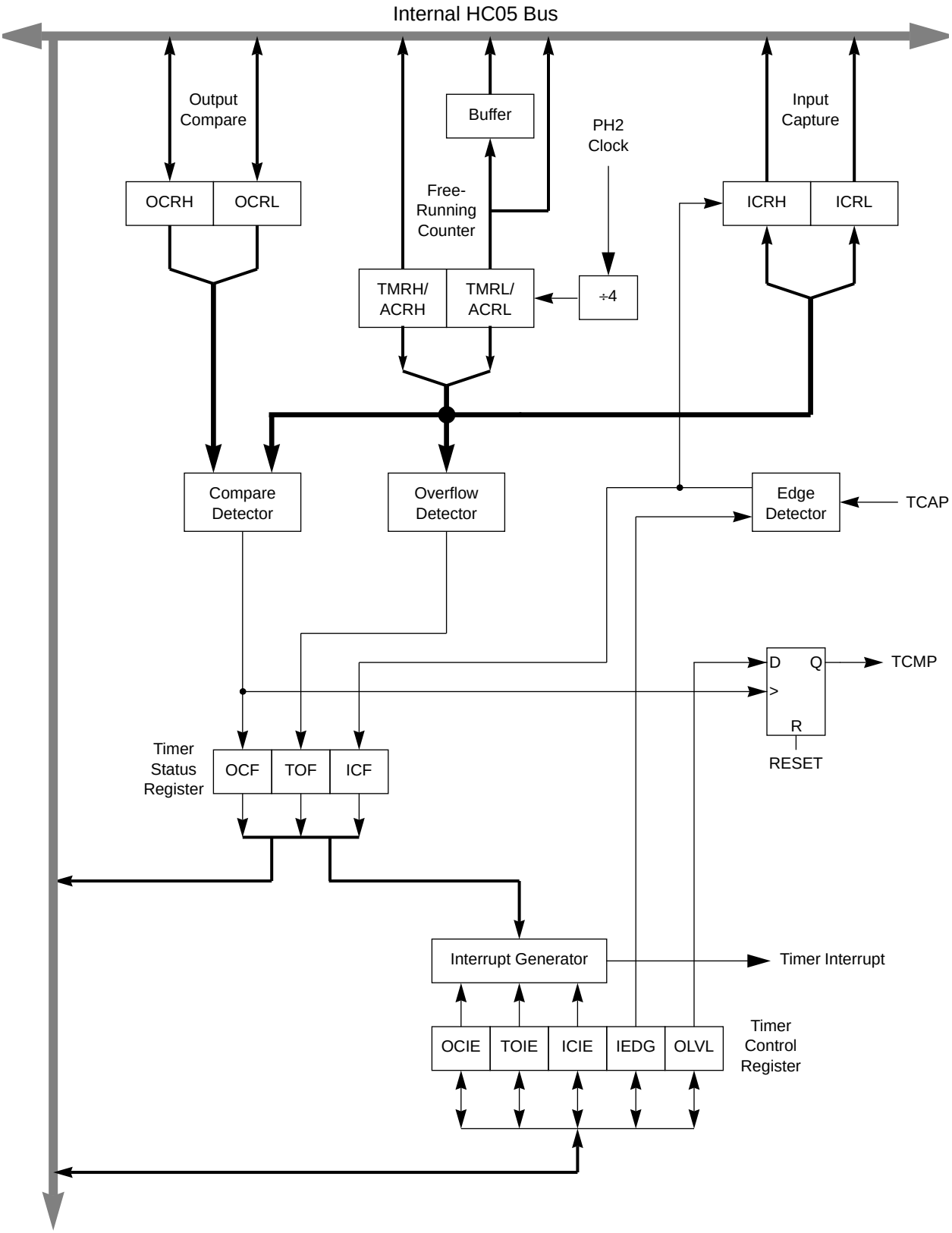


Figure 8-1: 16-Bit Timer Block Diagram

		7	6	5	4	3	2	1	0
TMRH $0018	R	TMRH7	TMRH6	TMRH5	TMRH4	TMRH3	TMRH2	TMRH1	TMRH0
	W								
reset ⇒		1	1	1	1	1	1	1	1

		7	6	5	4	3	2	1	0
TMRL $0019	R	TMRL7	TMRL6	TMRL5	TMRL4	TMRL3	TMRL2	TMRL1	TMRL0
	W								
reset ⇒		1	1	1	1	1	1	0	0

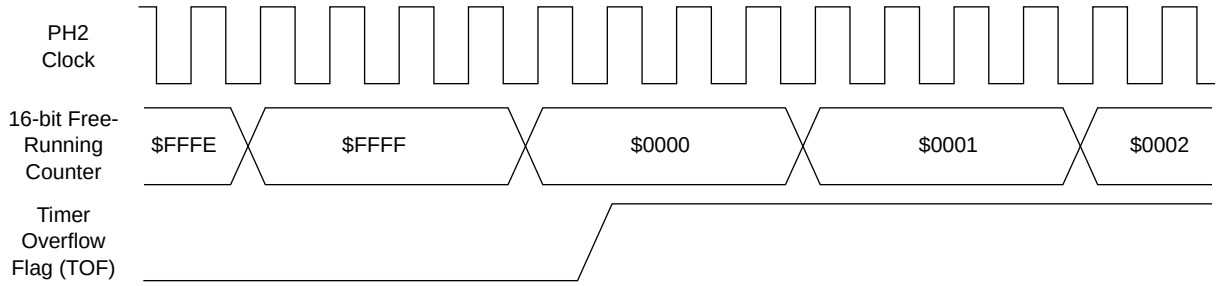
Figure 8-2: Timer Registers (TMRH / TMRL)

		7	6	5	4	3	2	1	0
ACRH $001A	R	ACRH7	ACRH6	ACRH5	ACRH4	ACRH3	ACRH2	ACRH1	ACRH0
	W								
reset ⇒		1	1	1	1	1	1	1	1

		7	6	5	4	3	2	1	0
ACRL $001B	R	ACRL7	ACRL6	ACRL5	ACRL4	ACRL3	ACRL2	ACRL1	ACRL0
	W								
reset ⇒		1	1	1	1	1	1	0	0

Figure 8-3: Alternate Counter Registers (ACRH / ACRL)

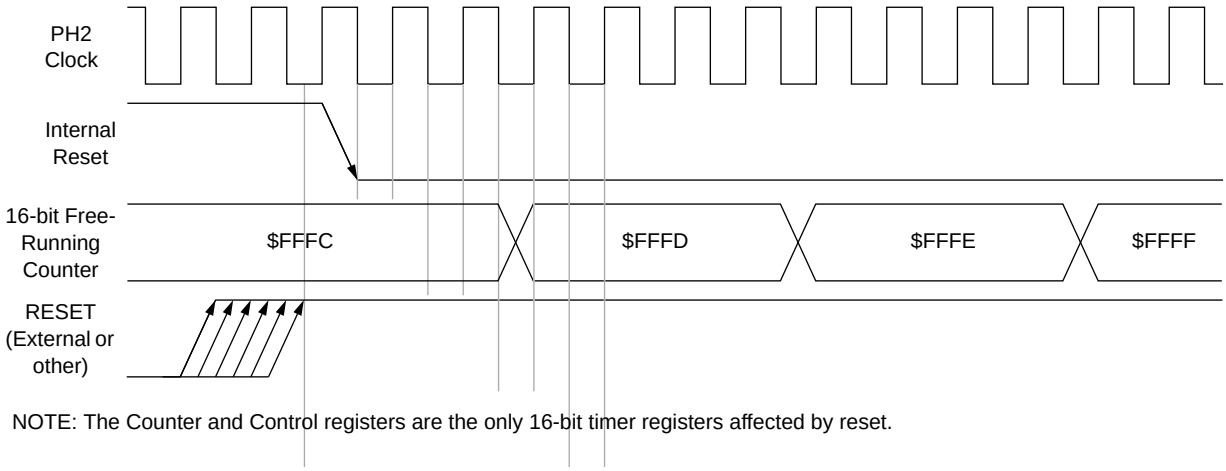
The Timer Registers and Alternate Counter Registers can be read at any time without affecting their values. However, the Alternate Counter Registers differ from the Timer Registers in one respect: a read of the Timer register MSB can clear the Timer Overflow Flag (TOF). Therefore, the Alternate Counter Registers can be read at any time without the possibility of missing timer overflow interrupts due to clearing of the TOF. See **Figure 8-4: State Timing Diagram for Timer Overflow**.



NOTE: The TOF bit is set at timer state T11 (transition of counter from $FFFF to $0000). It is cleared by reading the timer status register (TSR) during the high portion of the PH2 clock followed by reading the LSB of the counter register pair (TCRL).

Figure 8-4: State Timing Diagram for Timer Overflow

The free-running counter is initialized to $FFFC during reset and is a read-only register. During power-on-reset (POR), the counter is initialized to $FFFC and begins counting after the oscillator start-up delay. Since the counter is sixteen bits preceded by a fixed divide-by-four prescaler, the value in the counter repeats every 262,144 PH2 clock cycles (524,288 oscillator cycles). When the free-running counter rolls over from $FFFF to $0000, the Timer Overflow Flag bit (TOF) in register TSR is set. An interrupt can also be enabled when counter rollover occurs by setting the Timer Overflow Interrupt Enable bit (TOIE) in register TCR. See **Figure 8-5: State Timing Diagram for Timer Reset**.



NOTE: The Counter and Control registers are the only 16-bit timer registers affected by reset.

Figure 8-5: State Timing Diagram for Timer Reset

8.2 OUTPUT COMPARE

The Output Compare function may be used to generate an output waveform and/or as an elapsed time indicator. All of the bits in the Output Compare register pair OCRH / OCRL are readable and writable and are not altered by the 16-bit timer's control logic. Reset does not affect the contents of these registers. If the output compare function is not utilized its registers may be used for data storage. See **Figure 8-6: Output Compare Registers (OCRH / OCRL)**.

		7	6	5	4	3	2	1	0
OCRH $0016	R	OCRH7	OCRH6	OCRH5	OCRH4	OCRH3	OCRH2	OCRH1	OCRH0
	W								
OCRL $0017	R	OCRL7	OCRL6	OCRL5	OCRL4	OCRL3	OCRL2	OCRL1	OCRL0
	W								
reset ⇒		X	X	X	X	X	X	X	X

Figure 8-6: Output Compare Registers (OCRH / OCRL)

The contents of the output compare registers are compared with the contents of the free-running counter once every four PH2 clock cycles. If a match is found, the Output Compare Flag bit (OCF) is set and the Output Level bit (OLVL) is clocked to the output latch. The values in the output compare registers and output level bit should be changed after each successful comparison to control an output waveform, or to establish a new elapsed time-out. An interrupt can also accompany a successful output compare if the Output Compare Interrupt Enable bit (OCIE) is set.

After a CPU write cycle to the MSB of the output compare register pair (OCRH), the output compare function is inhibited until the LSB (OCRL) is written. Both bytes must be written if the MSB is written. A write made only to the LSB will not inhibit the compare function. The free-running counter increments every four PH2 clock cycles. The minimum time required to update the output compare registers is a function of software rather than hardware.

The output compare Output Level bit (OLVL) will be clocked to its output latch regardless of the state of the Output Compare Flag bit (OCF). A valid output compare must occur before the OLVL bit is clocked to its output latch (TCMP).

Since neither the Output Compare Flag (OCF) nor the output compare registers are affected by reset, care must be exercised when initializing the output compare function. The following procedure is recommended:

1. Block interrupts by setting the I-bit in the Condition Code Register (CCR).
2. Write the MSB of the output compare register pair (OCRH) to inhibit further compares until the LSB is written.
3. Read the Timer Status Register (TSR) to arm the Output Compare Flag (OCF).
4. Write the LSB of the output compare register pair (OCRL) to enable the output compare function and to clear its flag (and interrupt).
5. Unblock interrupts by clearing the I-bit in the CCR.

This procedure prevents the Output Compare Flag bit (OCF) from being set between the time it is read and the time the output compare registers are updated. A software example is shown in **Figure 8-7: Output Compare Software Initialization Example**.

```
9B          SEI                 BLOCK INTERRUPTS
.   .   .   .   .
.   .   .   .   .
B6  XX  LDA DATAH HI BYTE FOR COMPARE
BE  XX  LDX DATAL LO BYTE FOR COMPARE
B7  16  STA OCRH  INHIBIT OUTPUT COMPARE
B6  13  LDA TSR   ARM OCF BIT TO CLEAR
BF  17  STX OCRL  READY FOR NEXT COMPARE
.   .   .   .   .
```

Figure 8-7: Output Compare Software Initialization Example

8.3 INPUT CAPTURE

Two 8-bit read-only registers (ICRH, ICRL) make up the 16-bit input capture. They are used to latch the value of the free-running counter after a defined transition is sensed by the input capture edge detector (*Note:* the input capture edge detector contains a Schmitt trigger to improve noise immunity). The edge that triggers the counter transfer is defined by the input edge bit (IEDG) in register TCR. Reset does not affect the contents of the input capture registers. See **Figure 8-8: Input Capture Registers (ICRH / ICRL)**.

		7	6	5	4	3	2	1	0
ICRH $0014	R	ICRH7	ICRH6	ICRH5	ICRH4	ICRH3	ICRH2	ICRH1	ICRH0
	W								
ICRL $0015	R	ICRL7	ICRL6	ICRL5	ICRL4	ICRL3	ICRL2	ICRL1	ICRL0
	W								
reset ⇒		X	X	X	X	X	X	X	X

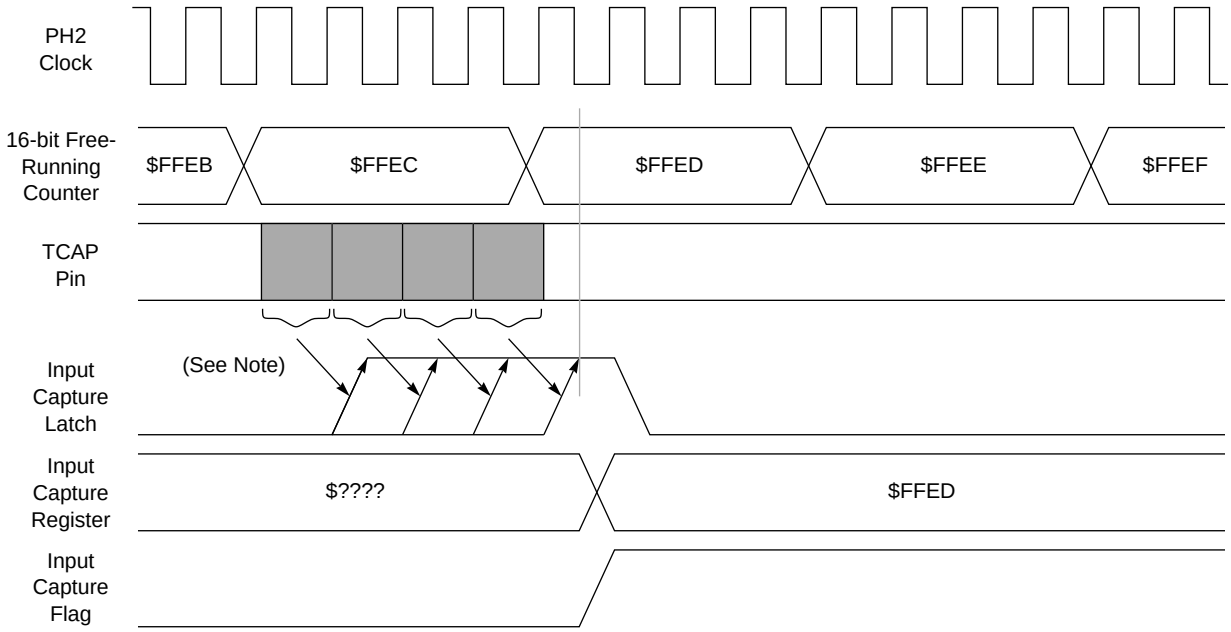
Figure 8-8: Input Capture Registers (ICRH / ICRL)

The result obtained by an input capture will be one more than the value of the free-running counter on the rising edge of the PH2 clock preceding the external transition (see **Figure 8-9: State Timing Diagram for Input Capture**). This delay is required for internal synchronization. Resolution is affected by the prescaler, allowing the free-running counter to increment once every four PH2 clock cycles.

The contents of the free-running counter are transferred to the input capture registers on each proper signal transition regardless of the state of the Input Capture Flag bit (ICF) in register TSR. The input capture registers always contain the free-running counter value which corresponds to the most recent input capture.

After a read of the MSB of the input capture register pair (ICRH), counter transfers are inhibited until the LSB of the register pair (ICRL) is also read. This characteristic forces the minimum pulse period attainable to be determined by the time required to execute an input capture software routine in an application.

Reading the LSB of the input capture register pair (ICRL) does not inhibit transfer of the free-running counter. Again, minimum pulse periods are ones which allow software to read the LSB of the register pair (ICRL) and perform needed operations. There is no conflict between reading the LSB (ICRL) and the free-running counter transfer since they occur on opposite edges of the PH2 clock.



NOTE: If the input edge occurs in the shaded area from one T10 timer state to the other T10 timer state the Input Capture flag is set during the next T11 timer state.

Figure 8-9: State Timing Diagram for Input Capture

8.4 TIMER CONTROL REGISTER (TCR)

The timer control (TCR) shown in **Figure 8-10: Timer Control Register (TCR)** and free-running counter (TMRH, TMRL, ACRH, ACRL) registers are the only registers of the 16-bit timer affected by reset. The Output Compare port (TCMP) is forced low after reset and remains low until OLVL is set and a valid Output Compare occurs.

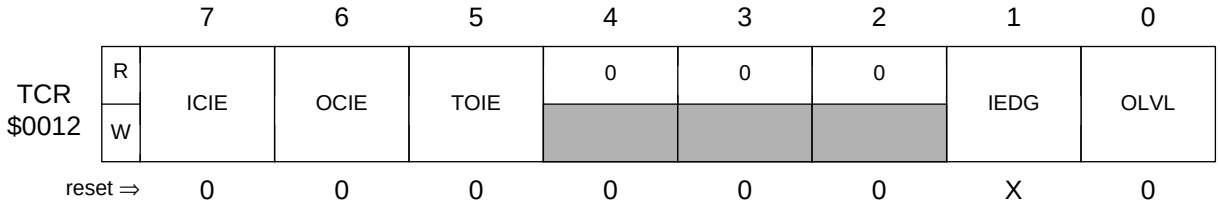


Figure 8-10: Timer Control Register (TCR)

8.4.1 ICIE - INPUT CAPTURE INTERRUPT ENABLE

Bit 7, when set, enables input capture interrupts to the CPU. The interrupt will occur at the same time bit 7 (ICF) in the TSR register is set.

8.4.2 OCIE - OUTPUT COMPARE INTERRUPT ENABLE

Bit 6, when set, enables output compare interrupts to the CPU. The interrupt will occur at the same time bit 6 (OCF) in the TSR register is set.

8.4.3 TOIE - TIMER OVERFLOW INTERRUPT ENABLE

Bit 5, when set, enables timer overflow (rollover) interrupts to the CPU. The interrupt will occur at the same time bit 5 (TOF) in the TSR register is set.

8.4.4 IEDG - INPUT CAPTURE EDGE SELECT

Bit 1 selects which edge of the input capture signal will trigger a transfer of the contents of the free-running counter registers to the input capture registers. Clearing this bit will select the falling edge, setting it selects the rising edge.

8.4.5 OLVL - OUTPUT COMPARE OUTPUT LEVEL SELECT

Bit 0 selects the output level (high or low) that is clocked into the output compare output latch at the next successful output compare.

8.5 TIMER STATUS REGISTER (TSR)

Reading the Timer Status Register (TSR) satisfies the first condition required to clear status flags and interrupts. See **Figure 8-11: Timer Status Register (TSR)**. The only remaining step is to read (or write) the register associated with the active status flag (and/or interrupt). This method does not present any problems for input capture or output compare functions.

However, a problem can occur when using a timer interrupt function and reading the free-running counter at random times to, for example, measure an elapsed time. If the proper precautions are not designed into the application software, a timer interrupt flag (TOF) could unintentionally be cleared if:

1. The TSR is read when bit 5 (TOF) is set, and
2. The LSB of the free-running counter is read, but not for the purpose of servicing the flag or interrupt.

The alternate counter registers (ACRH, ACRL) contain the same values as the timer registers (TMRH, TMRL). Registers ACRH and ACRL can be read at any time without affecting the Timer Overflow Flag (TOF) or interrupt.

		7	6	5	4	3	2	1	0
TSR $0013	R	ICF	OCF	TOF	0	0	0	0	0
	W								
reset ⇒		X	X	X	0	0	0	0	0

Figure 8-11: Timer Status Register (TSR)

8.5.1 ICF - INPUT CAPTURE FLAG

Bit 7 is set when the edge specified by IEDG in register TCR has been sensed by the input capture edge detector fed by pin TCAP. This flag, and the input capture interrupt, can be cleared by reading register TSR followed by reading the LSB of the input capture register pair (ICRL).

8.5.2 OCF - OUTPUT COMPARE FLAG

Bit 6 is set when the contents of the output compare registers match the contents of the free-running counter. This flag, and the output compare interrupt, can be cleared by reading register TSR followed by writing the LSB of the output compare register pair (OCRL).

8.5.3 TOF - TIMER OVERFLOW FLAG

Bit 5 is set by a rollover of the free-running counter from $FFFF to $0000. This flag, and the timer overflow interrupt, can be cleared by reading register TSR followed by reading the LSB of the timer register pair (TMRL).

8.6 TIMER OPERATION DURING WAIT/HALT MODES

During Wait and Halt Modes the 16-bit timer continues to operate normally and may generate an interrupt to trigger the MCU out of the Wait or Halt Mode.

8.7 TIMER OPERATION DURING STOP MODE

When the MCU enters the Stop Mode the free-running counter stops counting (the PH2 clock is stopped). It remains at that particular count value until the Stop Mode is exited by applying a low signal to the IRQ pin, at which time the counter resumes from its stopped value as if nothing had happened. If Stop Mode is exited via an external RESET (logic low applied to the RESET pin) the counter is forced to $FFFC.

If a valid input capture edge occurs at the TCAP pin during Stop Mode, the input capture detect circuitry is armed. This action does not set any flags or “wake up” the MCU, but when the MCU does “wake up” there will be an active input capture flag (and data) from the first valid edge. If the Stop Mode is exited by an external RESET, no input capture flag or data will be present even if a valid input capture edge was detected during the Stop Mode.

SECTION 9 SERIAL INPUT / OUTPUT PORTS

The simple synchronous Serial I/O Port (SIOP) subsystem is designed to provide efficient serial communications between peripheral devices or other MCUs. The SIOP is implemented as a three wire Master/Slave system with Serial Clock (SCK), Serial Data Input (SDI), and Serial Data Output (SDO). A block diagram of the SIOP is shown in **Figure 9-1: SIOP Block Diagram**.

The SIOP subsystem shares its input/output pins with Port B. When the SIOP is enabled (SPE bit set in register SCR) Port B DDR and data registers are modified by the SIOP. Although Port B DDR and data registers can be altered by application software, these actions could affect the transmitted or received data.

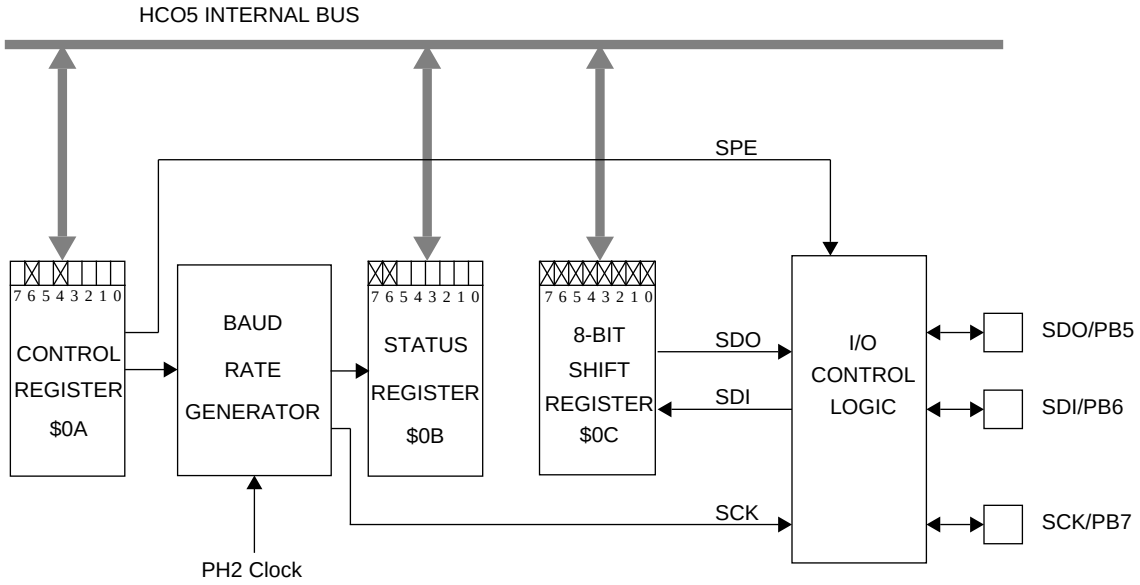


Figure 9-1: SIOP Block Diagram

9.1 SIOP SIGNAL FORMAT

The SIOP subsystem is software configurable for Master or Slave operation. There are no external mode selection inputs available (for example, slave select pin).

9.1.1 SERIAL CLOCK (SCK)

The state of the SCK output normally remains a logic one during idle periods between data transfers. The first falling edge of SCK signals the beginning of a data transfer. At this time the first bit of received data is accepted at the SDI pin and the first bit of transmitted data is presented at the SDO pin (see **Figure 9-2: SIOP Timing Diagram**). Data is captured at the SDI pin on the rising edge of SCK, and the first bit of transmitted data is presented at the SDO pin. The transfer is terminated upon the eighth rising edge of SCK.

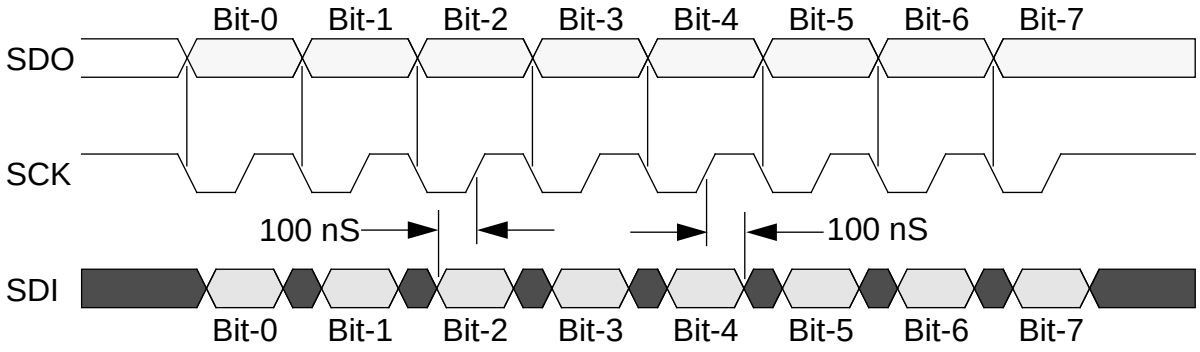


Figure 9-2: SIOP Timing Diagram

The Master and Slave modes of operation differ only by the sourcing of SCK. In Master mode, SCK is driven from an internal source within the MCU. In Slave mode, SCK is driven from a source external to the MCU. The SCK frequency is Mask Option selectable. Available rates are OSC divided by 2, 4, 8, or 16. Note that OSC divided by 2 is four times faster than the standard rate available on the 68HC05P6. Refer to **1.2 MASK OPTIONS** for a description of available Mask Options.

9.1.2 SERIAL DATA INPUT (SDI)

The SDI pin becomes an input as soon as the SIOP subsystem is enabled. New data is presented to the SDI pin on the falling edge of SCK. Valid data must be present at least 100 nanoseconds before the rising edge of SCK and remain valid for 100 nanoseconds after the rising edge of SCK. See **Figure 9-2: SIOP Timing Diagram**.

9.1.3 SERIAL DATA OUTPUT (SDO)

The SDO pin becomes an output as soon as the SIOP subsystem is enabled. Prior to enabling the SIOP, PB5 can be initialized to determine the beginning state. While the SIOP is enabled, PB5 cannot be used as a standard output since that pin is connected to the last stage of the SIOP serial shift register. A Mask Option is included to allow the data to be transmitted in either MSB first format or the LSB format.

On the first falling edge of SCK the first data bit will be shifted out to the SDO pin. The remaining data bits will be shifted out to the SDI pin on subsequent falling edges of SCK. The SDO pin will present valid data at least 100 nanoseconds before the rising edge of the SCK and remain valid for 100 nanoseconds after the rising edge of SCK. See **Figure 9-2: SIOP Timing Diagram**.

9.2 SIOP REGISTERS

The SIOP is programmed and controlled by the SIOP Control Register (SCR) located at address $000A, the SIOP Status Register (SSR) located at address $000B, and the SIOP Data Register (SDR) located at address $000C.

9.2.1 SIOP CONTROL REGISTER (SCR)

This register is located at address $000A and contains two bits. **Figure 9-3: SIOP Control Register (SCR)** shows the position of each bit in the register and indicates the value of each bit after reset.

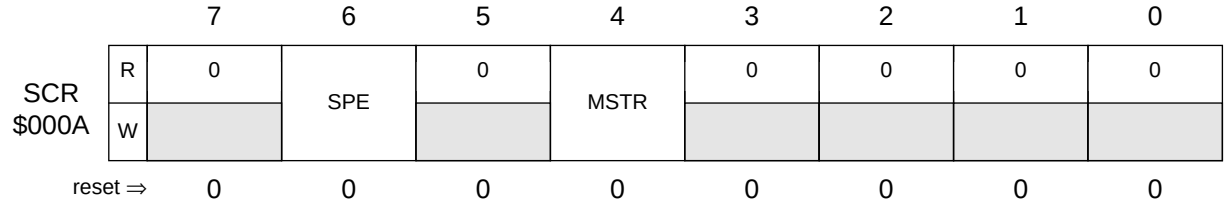


Figure 9-3: SIOP Control Register (SCR)

9.2.1.1 SPE - Serial Peripheral Enable

When set, the SPE bit enables the SIOP subsystem such that SDO/PB5 is the Serial Data Output, SDI/PB6 is the Serial Data Input, and SCK/PB7 is a Serial Clock input in the Slave mode or a Serial Clock output in the Master mode. Port B DDR and data registers can be manipulated as usual (except for PB5); however these actions could affect the transmitted or received data.

The SPE bit is readable and writeable at any time. Clearing the SPE bit while a transmission is in progress will 1) abort the transmission 2) reset the serial bit counter and 3) convert the Port B/SIOP port to a general-purpose I/O port. Reset clears the SPE bit.

9.2.1.2 MSTR - Master Mode Select

When set, the MSTR bit configures the Serial I/O Port for Master mode. A transfer is initiated by writing to the SDR. Also, the SCK pin becomes an output providing a synchronous data clock dependent upon the oscillator frequency. When the device is in Slave mode, the SDO and SDI pins do not change function. These pins behave exactly the same in both the Master and Slave modes.

The MSTR bit is readable and writeable at any time regardless of the state of the SPE bit. Clearing the MSTR bit will abort any transfers that may have been in progress. Reset clears the MSTR bit, placing the SIOP subsystem in Slave mode.

9.2.2 SIOP STATUS REGISTER (SSR)

This register is located at address $000B and contains two bits. **Figure 9-4: SIOP Status Register** shows the position of each bit in the register and indicates the value of each bit after reset.

		7	6	5	4	3	2	1	0
SSR $000B	R	SPIF	DCOL	0	0	0	0	0	0
	W								
reset ⇒		0	0	0	0	0	0	0	0

Figure 9-4: SIOP Status Register

9.2.2.1 SPIF - Serial Port Interface Flag

SPIF is a read-only status bit that is set on the last rising edge of SCK and indicates that a data transfer has been completed. It has no effect on any future data transfers and can be ignored. The SPIF bit is cleared by reading the SSR followed by a read or write of the SDR. If the SPIF is cleared before the last rising edge of SCK it will be set again on the last rising edge of SCK. Reset clears the SPIF bit.

9.2.2.2 DCOL - Data Collision

DCOL is a read-only status bit, which indicates that an illegal access of the SDR has occurred. The DCOL bit will be set when reading or writing the SDR after the first falling edge of SCK and before SPIF is set. Reading or writing the SDR during this time will result in invalid data being transmitted or received.

The DCOL bit is cleared by reading the SSR (when the SPIF bit is set) followed by a read or write of the SDR. If the last part of the clearing sequence is done after another transfer has started, the DCOL bit will be set again. Reset clears the DCOL bit.

9.2.3 SIOP DATA REGISTER (SDR)

This register is located at address $000C and serves as both the transmit and receive data register. Writing to this register will initiate a message transmission if the SIOP is in master mode. The SIOP subsystem is not double buffered and any write to this register will destroy the previous contents. The SDR can be read at any time; however, if a transfer is in progress, the results may be ambiguous and the DCOL bit will be set. Writing to the SDR while a transfer is in progress can cause invalid data to be transmitted and/or received. **Figure 9-5: Serial Port Data Register** shows the position of each bit in the register. This register is not affected by reset.

		7	6	5	4	3	2	1	0
SDR $000C	R	SD7	SD6	SD5	SD4	SD3	SD2	SD1	SD0
	W								
reset ⇒		X	X	X	X	X	X	X	X

Figure 9-5: Serial Port Data Register

SECTION 10 ELECTRICAL SPECIFICATIONS

10.1 MAXIMUM RATINGS

(Voltages referenced to V_{SS})

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
Current Drain Per Pin Excluding V_{DD} and V_{SS}	I	25	mA
Operating Temperature Range MC68HC05P18 (Standard) MC68HC05P18 (Extended) MC68HC05P18 (Automotive)	T_A	T_L to T_H 0 to +70 -40 to +85 -40 to +125	°C
Storage Temperature Range	T_{STG}	-65 to +150	°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 $V_{SS} \le (V_{IN}$ or $V_{OUT}) \le V_{DD}$. Reliability of operation is enhanced if unused inputs are connected to an appropriate logic voltage level (for example, either V_{SS} or V_{DD}).

10.2 THERMAL CHARACTERISTICS

Characteristic	Symbol	Value	Unit
Thermal Resistance Plastic SOIC	θ_{JA}	 60 60	°C/W

10.3 DC ELECTRICAL CHARACTERISTICS

(V_{DD} = 5.0 Vdc ±10%, V_{SS} = 0 Vdc, T_A = -40°C to +125°C, unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Output Voltage I_{Load} = 10.0 µA	V_{OL} V_{OH}	— V_{DD} - 0.1	— —	0.1 —	V
Output High Voltage (I_{Load} = -0.8 mA) PA0-7, PB5-7,PC0-7, PD5/CKOUT,	V_{OH}	V_{DD} - 0.8	—	—	V
Output Low Voltage (I_{Load} = 1.6 mA) PA0-7, PB5-7, PC0-7, PD5/CKOUT,	V_{OL}	—	—	0.4	V
Input High Voltage PA0-7, PB5-7, PC0-7, PD5/CKOUT, TCAP/PD7, $\overline{IRQ}$, $\overline{RESET}$, OSC1	V_{IH}	$0.7 \times V_{DD}$	—	V_{DD}	V
Input Low Voltage PA0-7, PB5-7, PC0-7, PD5, TCAP/ PD7, $\overline{IRQ}$, $\overline{RESET}$, OSC1	V_{IL}	V_{SS}	—	$0.2 \times V_{DD}$	V
Supply Current (see Notes)					
Run	I_{DD}	—	7(Note 8)	TBD	mA
Wait	I_{DD}	—	TBD	TBD	mA
Stop	I_{DD}	—	TBD	TBD	mA
25°C	I_{DD}	—	TBD	TBD	µA
0°C to +70°C (Standard)	I_{DD}	—	TBD	TBD	µA
-40°C to +85°C (Extended)	I_{DD}	—	TBD	TBD	µA
-40°C to +125°C (Automotive)	I_{DD}	—	TBD	TBD	µA
I/O Ports Hi-Z Leakage Current PA0-7, PB5-7, PC0-7, PD5/CKOUT, TCAP/PD7	I_{IL}	—	—	±10	µA
I/O Ports Switch Resistance (Pullup Enabled PA0-7)	R_{PTA}	62	—	102	k
A/D Ports Hi-Z Leakage Current PC3-7	I_{IL}	—	—	±1	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
EEPROM Program/Erase Time Byte Block Bulk				 10 30 100	 ms ms ms
Low Voltage Reset Voltage			3.8		V

NOTES:

1. All values shown reflect average measurements.
2. Typical values at midpoint of voltage range, 25°C only.

3. Wait I_{DD}: Only timer system active.
4. Run (Operating) I_{DD}, Wait I_{DD}: Measured using external square wave clock source (f_{osc} = 4.2 MHz), all inputs 0.2 V from rail; 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} measured with OSC1 = V_{SS}.
7. Wait I_{DD} is affected linearly by the OSC2 capacitance.
8. Pre-silicon estimate.

10.4 ACTIVE RESET CHARACTERISTICS

(V_{DD} =4.5 Vdc, V_{SS} = 0 Vdc, T_A = 125°C)

Table 10-1: Active Reset Characteristics

Rise Time	Fall Time	Pulse Width	C_{LOAD}	Pull-Up
0.5 us	13 ns	2.4 us	50 pF	10K
1.0 us	20 ns	2.7 us	100 pF	10K
2.5 us	42 ns	3.7 us	250 pF	10K

10.5 A/D CONVERTER CHARACTERISTICS

(V_{DD} = 5.0 Vdc ±10%, V_{SS} = 0 Vdc, T_A = -40°C to +125°C, unless otherwise noted)

Characteristic	Min	Max	Unit	Comments
Resolution	8	8	Bits	
Absolute Accuracy ($V_{DD} \geq V_{REFH} > 4.5$)	—	±1 1/2	LSB	Including quantization
Conversion Range V_{REFH}	V_{SS} V_{SS}	V_{REFH} V_{DD}	V V	A/D accuracy may decrease proportionately as V_{REFH} is reduced below 4.5.
Input Leakage AD0, AD1, AD2, AD3 V_{REFH}	 — —	 ±1 ±1	 µA µA	
Conversion Time (Includes Sampling Time)	32	32	t_{AD}*	
Monotonicity	Inherent (Within Total Error)			
Zero Input Reading	00	01	Hex	V_{IN} = 0V
Full-scale Reading	FE	FF	Hex	V_{IN} = V_{REFH}
Sample Time	12	12	t_{AD}*	
Input Capacitance	—	12	pF	
Analog Input Voltage	V_{SS}	V_{REFH}	V	

*t_{AD} = t_{CYC} if clock source equals MCU.

10.6 SIOP TIMING

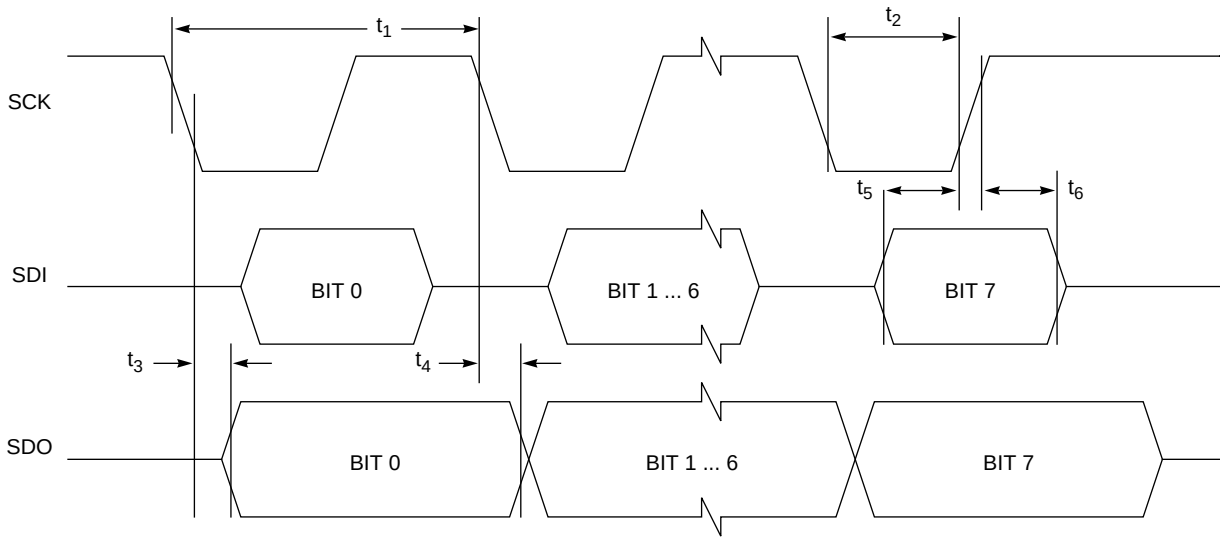


Figure 10-1: SIOP Timing Diagram

(V_{DD} = 5.0 Vdc ±10%, V_{SS} = 0 Vdc, T_A = -40°C to +125°C, unless otherwise noted)

Number	Characteristic	Symbol	Minimum	Maximum	Unit
	Operating Frequency Master Slave	 $f_{OP(M)}$ $f_{OP(S)}$	 1 dc	 1 1	 f_{OP} f_{OP}
1	Cycle Time Master Slave	 $t_{CYC(M)}$ $t_{CYC(S)}$	 4.0 —	 4.0 4.0	 t_{CYC} t_{CYC}
2	SCK Low Time	t_{CYC}	932	—	ns
3	SDO Data Valid Time	t_V	—	200	ns
4	SDO Hold Time	t_{HO}	0	—	ns
5	SDI Setup Time	t_S	100	—	ns
6	SDI Hold Time	t_H	100	—	ns

Note: $f_{OP} = f_{OSC} \div 2$ = 2.1 MHz max; $t_{CYC} = 1 \div f_{OP}$

10.6.1 OSC Out Timing

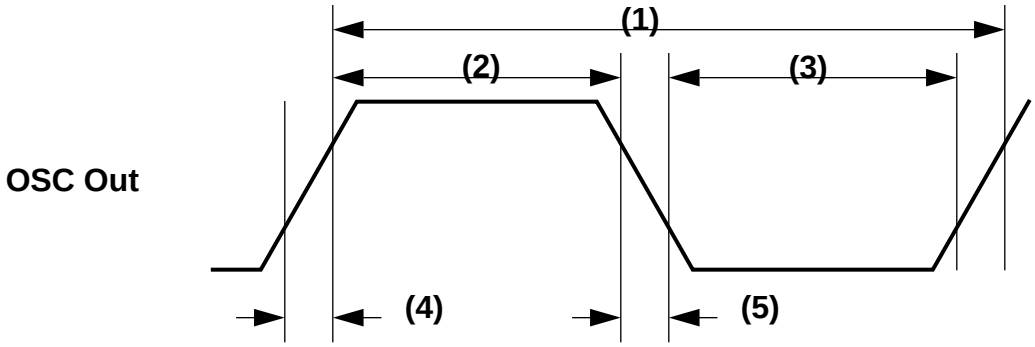


Figure 10-2: OSC Out Timing

Characteristic	Symbol	Min	Max	Unit
Cycle Time	1	476		ns
Rise Time	4	3.5	12	ns
Fall Time	5	7.5	27.5	ns
Pulse Width	2,3	200	-	ns

NOTE: All timing is shown with respect to 20% and 70% V_{DD}. Maximum rise and fall times assume 44% duty cycle. Minimum rise and fall times assume 55% duty cycle.

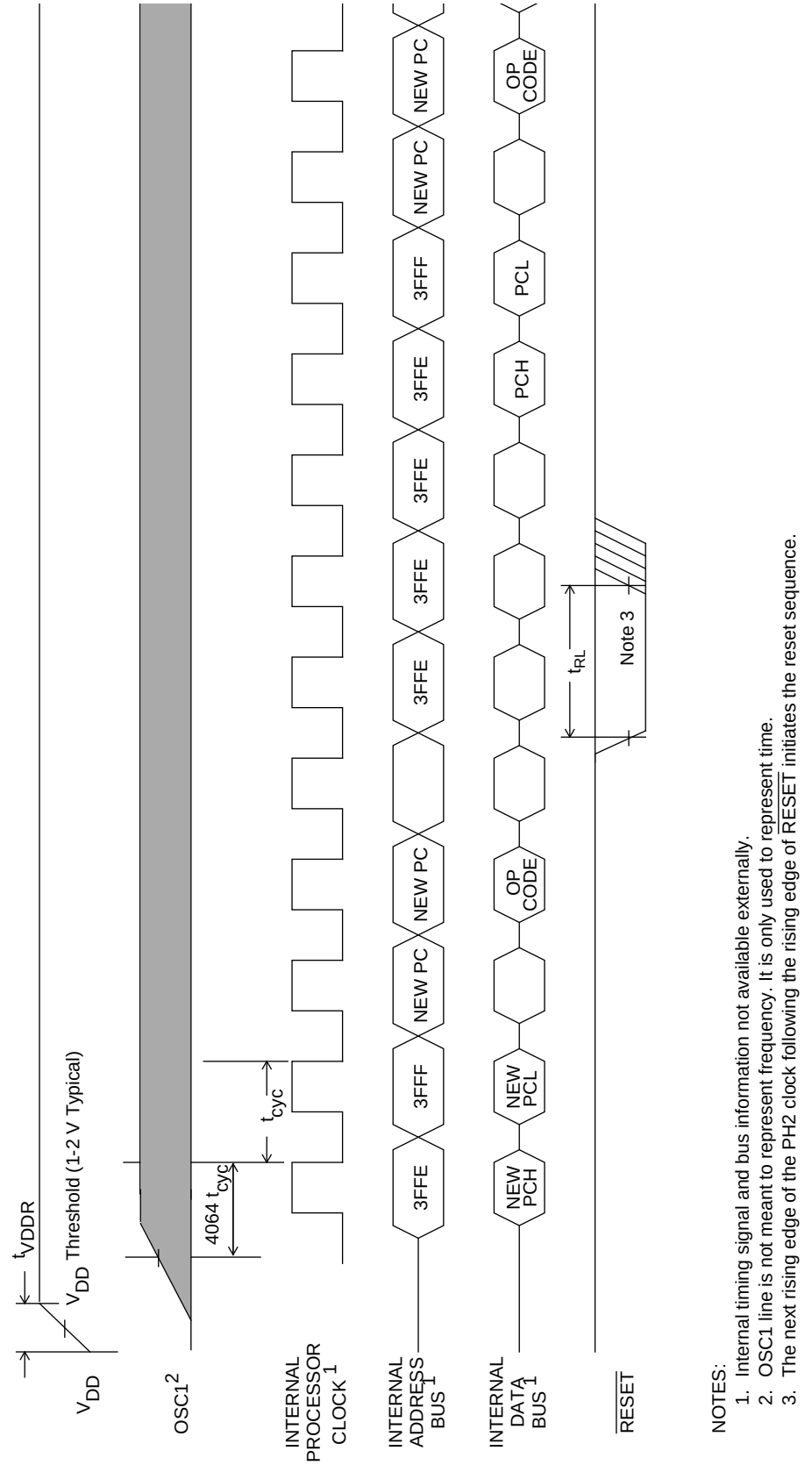
10.7 CONTROL TIMING

(V_{DD} = 5.0 Vdc ±10%, V_{SS} = 0 Vdc, T_A = -40°C to +125°C, unless otherwise noted)

Characteristic	Symbol	Min	Max	Unit
Frequency of Operation Crystal Option External Clock Option	 f_{osc} f_{OSC}	 — dc	 4.2 4.2	 MHz MHz
Internal Operating Frequency Crystal (f_{osc} ÷ 2) External Clock (f_{osc} ÷ 2)	 f_{OP} f_{OP}	 — dc	 2.1 2.1	 MHz MHz
Cycle Time	t_{CYC}	476	—	ns
Crystal Oscillator Start-up Time	f_{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}	200	—	ns
A/D On Current Stabilization Time	t_{ADON}	—	100	µs
RC Oscillator Stabilization Time (A/D)	t_{RCON}	—	5.0	µs

* The minimum period T_{ILIL} should not be less than the number of cycles it takes to execute the interrupt service routine plus 19 t_{CYC}.

CONTROL TIMING (CONTINUED)



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 PH2 clock following the rising edge of RESET initiates the reset sequence.

Figure 10-3: Power-On Reset and External Reset Timing Diagram

THIS PAGE INTENTIONALLY LEFT BLANK

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

HC05P18GRS/D