HC05PV8GRS/D
Rev. 1.4

68HC(8)05PV8

General Release Specification
Rev. 1.4

July 10, 1998

CSIC System Design Group
Munich

MOTOROLA

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

List of Sections

List of Sections

Table of Contents

Section 3. CPU and Instruction Set

Section 4. Interrupts

Section 5. Resets

Section 6. Operating Modes

Section 7. Input/Output Ports

Section 8. Core Timer

Section 9. 16-Bit Programmable Timer

Section 10. Analog to Digital Converter

Section 11. Pulse Width Modulator

Section 12. Voltage Regulator

Section 13. EEPROM

Section 14. Program EEPROM

Section 15. Fast Parallel Interface

Section 16. Electrical Specifications

List of Figures

List of Figures

List of Figures

List of Tables

List of Tables

Section 1. General Description

1.1 Contents

1.2 Introduction

The MC68HC(8)05PV8 microcomputer is a member of Motorola's 68HC05 family, designed for low-cost and single-chip systems in automotive applications. It combines an HC05 core with a shell of high-voltage peripherals.

The ROM (MC68HC05PV8) version of the 8 bit microcomputer unit (MCU) contains the HC05 CPU with integrated voltage regulator, RAM, ROM, EEPROM, core timer, COP watchdog, power-on reset, 16-bit programmable timer, PWM generator, standard parallel I/O, and special I/O for the automotive voltage range, including relay driver and contact

monitors. Bootloader and test modes are supported. The package is 28-pin SOIC for the ROM and development version.

In the flash-like development version (MC68HC805PV8), the ROM is replaced by a program EEPROM.

The MC68HC(8)05PV8 is fabricated in a low-cost double-layer poly, single-layer metal, 40V, 1.2µm CMOS technology.

1.3 Features

Features of the MC68HC(8)05PV8 include:

- HC05 Core
- 28 Pin SOIC Package
- Program EEPROM or ROM
  - MC68HC805PV8: 7936 Bytes of Program EEPROM + 240 Bytes of Monitor ROM + 16 Bytes User Vectors
  - MC68HC05PV8: 7936 Bytes of ROM + 240 Bytes of Monitor ROM + 16 Bytes User Vectors
- 192 Bytes of RAM Including Stack
- 128 Bytes of Data EEPROM
- On-Chip 5V (±5%) Voltage Regulator Including Power-On Reset, with 20mA Supply for External Devices. VSUP Range is 6V to 16V. Limited Operation Above and Below That Range. Breakdown Voltage above 40V.
- On-Chip Oscillator with External Resonator. Internal bus Frequency in Run and Wait Mode is $f_{OSC} \div 2$.
- Multipurpose Core Timer, Real Time Interrupt (RTI), (Window) COP Watchdog Timer
- 16-Bit Timer With Two Input Captures and Two Output Compares
- 1 Channel High-Speed PWM With Adjustable Frame Frequency
- 8 bit 6 Channel A/D Converter

- Port A: 8 Channel 5V I/O, with Pull-Ups, Shared with A/D Converter
- Port B: 5 Channel 5V I/O Shared with Timer and PWM
- Port C: 7 channel 40V I/O
  - 5 Channel 10mA Contact Monitor, 1 for a Switch to Ground, 1 for a Switch to Battery and 3 of Universal Type. Contact Monitoring Requires a 1KΩ External Resistor. Contact Monitor Pins May Alternatively be Configured as High-Voltage I/O Relative to VSUP. Pins are Shared with Timer and PWM.
  - 2 Channel 2Ω LS Relay Driver. The Pins are Shared with the PWM.
- Break-Down Voltage of High-Voltage Pins is Greater Than 40V.
- High-Voltage Interrupt/Reset (HVI/HVR) and Low-Voltage Reset (LVR).
- –40°C to 125°C Junction Temperature.
- Operational Amplifier, Connected to PA4–6
- Keyboard Wake-Up Interrupt on Port A and PC4–0
- ISO9141 Compatible Transceiver on Port C4

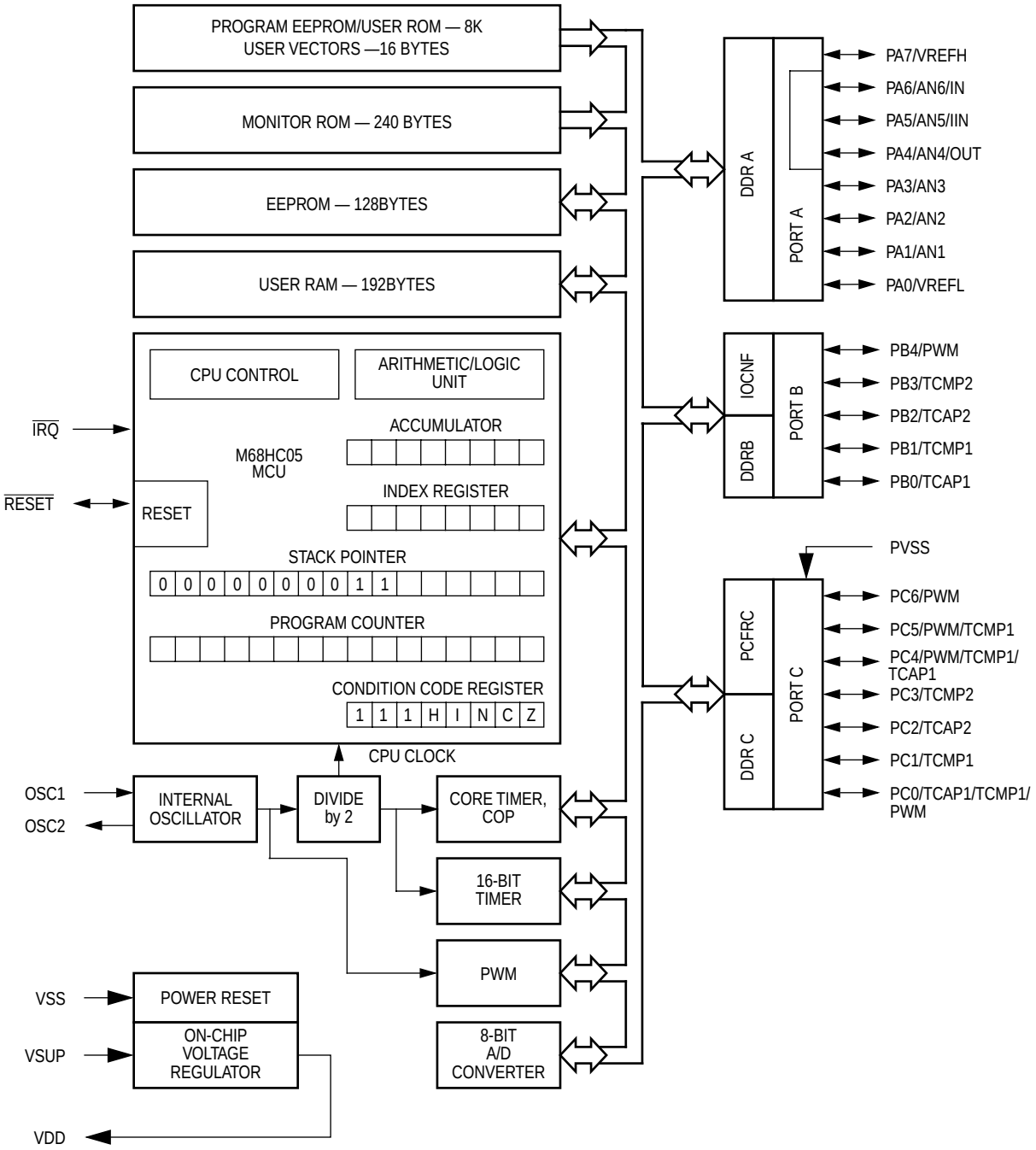


Figure 1-1 MC68HC(8)05PV8 Block Diagram

1.4 Mask Options

There are five mask options on the MC68HC(8)05PV8:

- STOP Instruction (enable/disable)
- COP Watchdog Timer (enable/disable)
- Clock Monitor (enable/disable)
- High Temperature Reset (enable/disable)
- High Voltage Reset (enable/disable)

1.5 Pin Assignments

Figure 1-2 shows the 28-pin SOIC pin assignments.

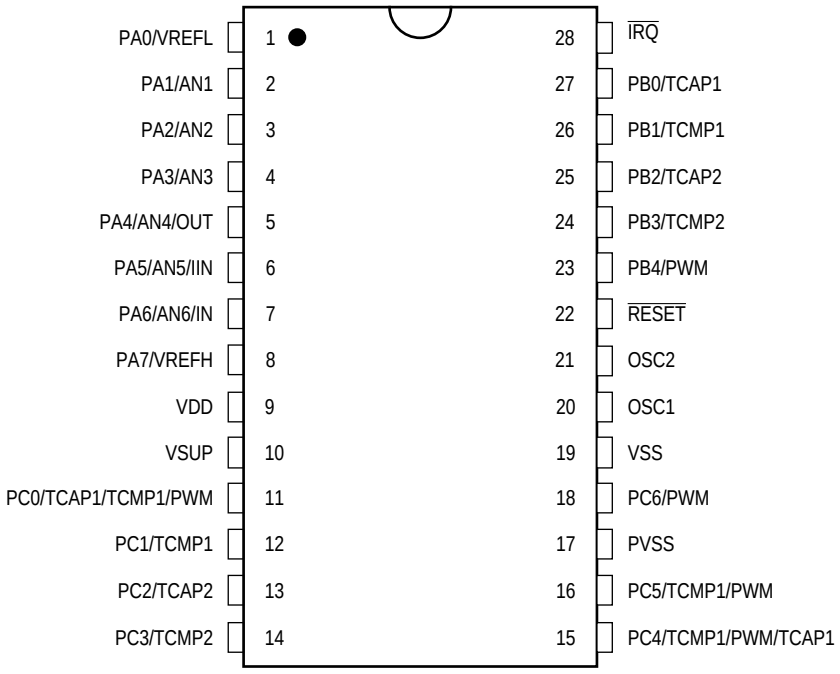


Figure 1-2 MC68HC(8)05PV8 Pin Assignments

NOTE: *This pin-out is for reference purposes only and subject to change without notice*

1.6 Functional Pin Descriptions

The following paragraphs give a description of the general function of each pin.

1.6.1 VSUP, VSS and PVSS

The microcontroller is operated from a single power supply. VSUP is connected to the positive supply, VSS to ground. The on-chip voltage regulator uses VSUP to derive the VDD supply for the MCU and external components. PVSS is a separate ground for the relay drivers.

1.6.2 VDD

This pin is driven by the on-chip voltage regulator. It can be used to provide a regulated voltage to external devices. A capacitor must be attached to this pin in order to stabilize the regulator. The regulator is short-circuit protected.

1.6.3 OSC1, OSC2

The OSC1 and OSC2 pins are the connections for the on-chip oscillator. A crystal connected across these pins or an external signal connected to OSC1 provides the oscillator clock. The frequency, f_{OSC}, of the oscillator or external clock source is divided by two to produce the internal operating frequency, f_{OP}.

1.6.4 $\overline{\text{RESET}}$

This pin can be used as an input to reset the MCU to a known start-up state by pulling it to the low state. The $\overline{\text{RESET}}$ pin contains an internal Schmitt trigger to improve its noise immunity as an input. The $\overline{\text{RESET}}$ pin has an internal pull-down device that pulls the $\overline{\text{RESET}}$ pin low when there is an internal COP watchdog reset, power-on reset (POR), illegal address reset, internal high voltage or an internal low voltage reset. Refer to **Section 5. Resets**.

1.6.5 $\overline{\text{IRQ}}$

The interrupt triggering sensitivity of this pin can be programmed as rising/falling edge sensitive or high/low level sensitive.The $\overline{\text{IRQ}}$ pin contains an internal Schmitt trigger as part of its input to improve noise immunity. See **Section 4. Interrupts** for more details on the interrupts.

1.6.6 PA0–PA7/VREFH, VREFL, AN1–6, IN, IIN, OUT

These eight I/O lines 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. The eight I/O lines are shared with the A/D converter function (see **Section 10. Analog to Digital Converter**). The internal operational amplifier is connected to PA4/OUT (output), PA5/IIN (inverting input) and PA6/IN (input) (see **7.4.6 Operational Amplifier**).

See **Section 7. Input/Output Ports** for more details on the I/O ports.

1.6.7 PB0–PB4/TCMP0, TCMP1, TCAP0, TCAP1, PWM

These five I/O lines comprise port B. The state of any pin is software programmable and all port B lines are configured as inputs during power-on or reset. The port pins PB0–PB3 are shared with the 16-bit timer (TCAP0–1, TCMP0–1). See **Section 9. 16-Bit Programmable Timer** for more details on the operation of the 16-bit timer. Pin PB4 is shared with the PWM system (see **Section 11. Pulse Width Modulator**).

See **Section 7. Input/Output Ports** for more details on the I/O ports.

1.6.8 PTC0–PTC6/TCMP1, TCMP2, TCAP1, TCAP2, PWM

These seven high voltage I/O lines comprise port C. The state of any pin is software programmable and all port C lines are configured as inputs during power-on or reset. The port pins PC0–PC5 are shared with the 16-bit timer (TCAP1–2, TCMP1–2). See **Section 9. 16-Bit Programmable Timer** for more details on the operation of the 16-Bit Timer. Pins PC0, PC4–6 are shared with the PWM system. PC5–6 are intended to drive relays.

See **Section 7. Input/Output Ports** for more details on the I/O ports.

Section 2. Memory

2.1 Contents

2.2 Introduction

The MC68HC(8)05PV8 has a 16K byte memory map consisting of registers (for I/O, control and status), user RAM, user ROM (or program EEPROM), EEPROM, Monitor ROM, and reset and interrupt vectors as shown in **Figure 2-1**.

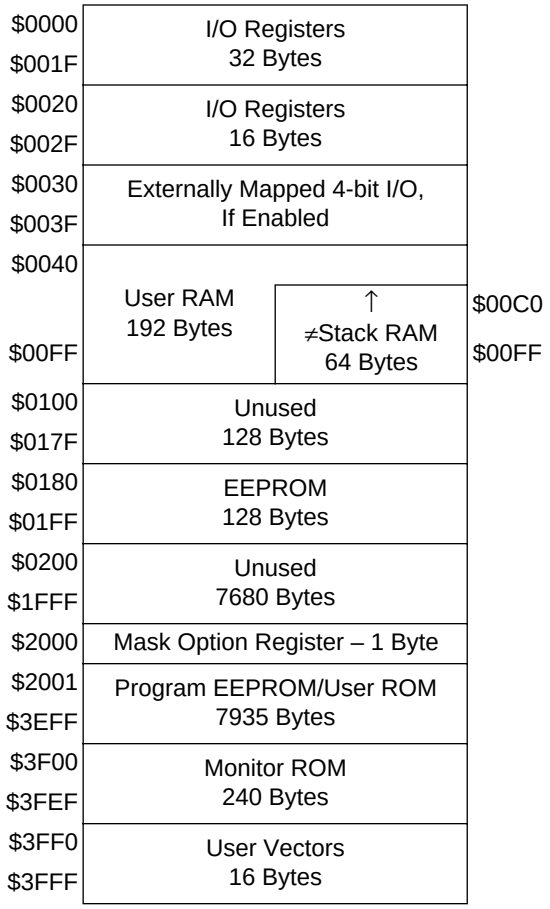


Figure 2-1 MC68HC(8)05PV8 Memory Map

2.3 Registers

The I/O and control registers reside in locations $0000–$002F. The overall organization of these registers is shown in **Figure 2-2**. The bit assignments for each register are shown in **Figure 2-3**, **Figure 2-4**, and **Figure 2-4**.

Addr	Register Name
$0000	Port A data register
$0001	Port B data register
$0002	Port C data register
$0003	Unused
$0004	Port A data direction register
$0005	Port B data direction register
$0006	Port C data direction register
$0007	Unused
$0008	Core timer control/status (CTCSR)
$0009	Core timer counter (CTCR)
$000A	System control register
$000B	Unused
$000C	EEPROM programming register
$000D	Program EEPROM programming register(1)
$000E	A/D data
$000F	A/D status/control
$0010	Timer capture 1 high
$0011	Timer capture 1 low
$0012	Timer compare 1 high
$0013	Timer compare 1 low
$0014	Timer capture 2 high
$0015	Timer capture 2 low
$0016	Timer compare 2 high
$0017	Timer compare 2 low
$0018	Timer counter high
$0019	Timer counter low
$001A	Timer alternate counter high

Figure 2-2 I/O Register Summary

Addr	Register Name
$001B	Timer alternate counter low
$001C	Timer control 1
$001D	Timer control 2
$001E	Timer status
$001F	TEST
$0020	Port A configuration register
$0021	I/O configuration register
$0022	Port C configuration register 0
$0023	Unused
$0024	Port A interrupt status
$0025	Unused
$0026	Port C configuration register 1
$0027	Port C status register
$0028	Interrupt control register
$0029	Interrupt status register
$002A	Reset status register
$002B	Unused
$002C	PWM period
$002D	PWM control
$002E	PWM data
$002F	MFTEST

Figure 2-2 I/O Register Summary

1. Implemented in MC68HC805PV8 only; unused in MC68HC05PV8

Addr	Register	R/W	Bit 7	6	5	4	3	2	1	Bit 0
$0000	Port A Data	R W	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
$0001	Port B Data	R W	0	0	TCAP1	PB4	PB3	PB2	PB1	PB0
$0002	Port C Data	R W	0	PC6	PC5	PC4	PC3	PC2	PC1	PC0
$0003	Unused	R W								
$0004	Port A Data Direction	R W	DDRA7	DDRA6	DDRA5	DDRA4	DDRA3	DDRA2	DDRA1	DDRA0
$0005	Port B Data Direction	R W	0	0	0	DDRB4	DDRB3	DDRB2	DDRB1	DDRB0
$0006	Port C Data Direction	R W	0	0	0	DDRC4	DDRC3	DDRC2	DDRC1	DDRC0
$0007	Unused	R W								
$0008	CTSCR	R W	TOF	RTIF	TOFE	RTIE	0 RTOF	0 RTIF	RT1	RT0
$0009	CTCR	R W	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
$000A	System Control	R W	POR	INTP	INTN	INTE	WCOP*	WCP	FPIE	FPICLK
$000B	Unused	R W								
$000C	EEPROG	R W	0	0	0	EEOSC	EER1	EER0	EELAT	EEPGM
$000D	Program EEPROM Control	R W				RCON	BULK	EEPERA	EEPLAT	EEPPGM
$000E	A/D Data	R W	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
$000F	A/D Status/Control	R W	COCO	ADRC	ADON	ADTEST	CH3	CH2	CH1	CH0

Figure 2-3 I/O Registers $0000–$000F

NOTE: *WCOP Bit is write once*

Addr	Register	R/W	Bit 7	6	5	4	3	2	1	Bit 0
$0010	Timer Input Capture1 High	R	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8
		W								
$0011	Timer Input Capture1 Low	R	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
		W								
$0012	Timer Output Compare1 High	R	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8
		W								
$0013	Timer Output Compare1 Low	R	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
		W								
$0014	Timer Input Capture2 High	R	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8
		W								
$0015	Timer Input Capture2 Low	R	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
		W								
$0016	Timer Output Compare2 High	R	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8
		W								
$0017	Timer Output Compare2 Low	R	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
		W								
$0018	Timer Counter High	R	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8
		W								
$0019	Timer Counter Low	R	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
		W								
$001A	Timer Alternate Counter High	R	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8
		W								
$001B	Timer Alternate Counter Low	R	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
		W								
$001C	Timer Control1	R	ICI1E	ICI2E	OCI1E	TOIE	OCI2E			TOFF
		W								
$001D	Timer Control2	R	IEDGE1	IEDGE2	CLK21	0	OLVL1	CLK12	0	OLVL2
		W				FOLV1			FOLV2	
$001E	Timer Status	R	IC1F	IC2F	OC1F	TOF	OC2F	SI1	SI2	0
		W								
$001F	TEST	R	0	0	0	0	0	0	0	0
		W	–	–	–	–	–	–	–	–

Figure 2-4 I/O Registers $0010–$001F

Addr	Register	R/W	Bit 7	6	5	4	3	2	1	Bit 0
$0020	Port A Configuration	R W	VRHEN	PUHEN	EDGEH	PAHIE	PULEN	EDGEL	PALIE	VRLEN
$0021	I/O Configuration	R W	TXOR	OPAMP	0	PB4PW	PB3OC	PB2IC	PB1OC	PB0IC
$0022	Port C Configuration 0	R W	ISOMODE	PC6PW	PWMS1	PWMS0	PC3OC	TS2	TS1	TS0
$0023	Unused	R W								
$0024	Port A Interrupt Status	R W	PAIF7	PAIF6	PAIF5	PAIF4	PAIF3	PAIF2	PAIF1	PAIF0
$0025	Unused	R W								
$0026	Port C Configuration 1	R W	CSIE	SCIE6	SCIE5	PC4CS	PC3CS	PC2CS	PC1CS	PC0CS
$0027	Port C Status	R W	CSIF	SCIF6	SCIF5	CSD4	CSD3	CSD2	CSD1	CSD0
$0028	Interrupt Control Register	R W	0	0	0	0	0	HTIE	HVIE	LVIE
$0029	Interrupt Status Register	R W	RCON	PC4CL	0	0	0	HTIF	HVIF	LVIF
$002A	Reset Status Register	R W	PINR	STOPR	COPR	ILINR	CMR	HTR	HVR	LVR
$002B	Unused	R W								
$002C	PWM Period	R W	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
$002D	PWM Control	R W	PWMON	POL	0	CYCLE	PRA3	PRA2	PRA1	PRA0
$002E	PWM Data	R W	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
$002F	MFTEST	R W	HVTOFF	0 –	0 –	VSCAL	LSOFF	VT2	VT1	VT0

Figure 2-5 I/O Registers $0020–$002F

2.4 RAM

The user RAM consists of 192 bytes ranging from $0040 to $00FF. The stack begins at address $00FF. The stack pointer can access 64 bytes of RAM in the range $00FF to $00C0.

The stack is located in the RAM address space. Data written to addresses within the stack address range could be overwritten during stack activity.

2.5 Monitor ROM

The monitor ROM ranges from $3F00 to $3FEF. The vectors for the bootloader are located from $3FE0 to $3FEF.

2.6 Program EEPROM

The program EEPROM holds 7952 bytes in total. The mask option register is located at address $2000. The 7935 bytes of the program EEPROM are located from $2001 to $3EFF, plus 16 bytes of user vectors from $3FF0 to $3FFF. The user programs the EEPROM on a 4 byte erase basis by manipulating the programming register located at address $000D. Refer to **Section 14. Program EEPROM** for details.

This EEPROM is replaced by an 8K ROM in the MC68HC05PV8, ranging from $2000 to $3EFF and $3FF0 to $3FFF. Mask options are controlled by the contents of location $2000. Refer to **Section 14. Program EEPROM** for coding details.

2.7 EEPROM

The 128 bytes of EEPROM are located from $0180 to $01FF. The user programs the EEPROM on a single-byte basis by manipulating the programming register, located at address $000C. Refer to **Section 13. EEPROM** for programming details.

Section 3. CPU and Instruction Set

3.1 Contents

3.2 CPU Registers

Figure 3-1 shows the five CPU registers. CPU registers are not part of the memory map.

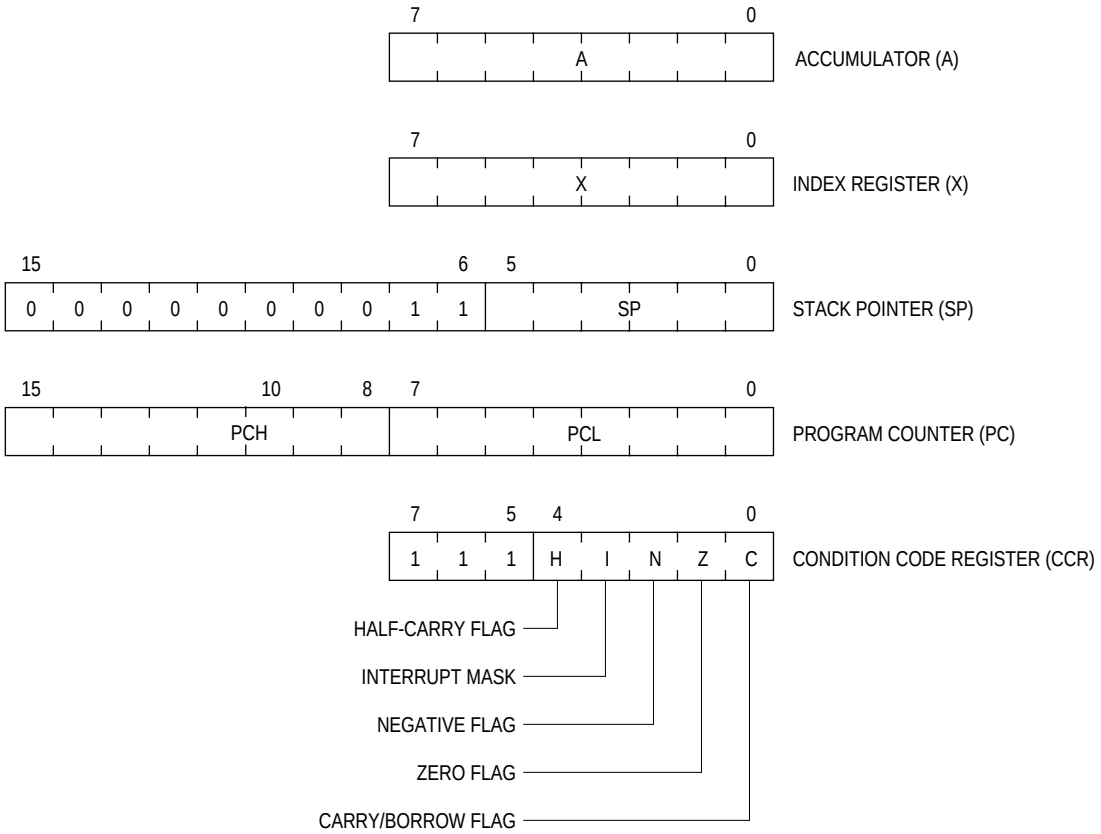


Figure 3-1 Programming Model

3.2.1 Accumulator

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

	Bit 7	6	5	4	3	2	1	Bit 0
Reset:	Unaffected by reset							

Figure 3-2 Accumulator

3.2.2 Index Register

In the indexed addressing modes, the CPU uses the byte in the index register to determine the conditional address of the operand.

	Bit 7	6	5	4	3	2	1	Bit 0
Reset:	Unaffected by reset							

Figure 3-3 Index Register

The 8-bit index register can also serve as a temporary data storage location.

3.2.3 Stack Pointer

The stack pointer is a 16-bit register that contains the address of the next location on the stack. During a reset or after the reset stack pointer (RSP) instruction, the stack pointer is preset to $00FF. The address in the stack pointer decrements as data is pushed onto the stack and increments as data is pulled from the stack.

	Bit 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	Bit 0
	0	0	0	0	0	0	0	0	1	1						
Reset	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1

Figure 3-4 Stack Pointer

The ten most significant bits of the stack pointer are permanently fixed at 000000011, so the stack pointer produces addresses from $00C0 to $00FF. If subroutines and interrupts use more than 64 stack locations, the stack pointer wraps around to address $00FF and begins writing over the previously stored data. A subroutine uses two stack locations. An interrupt uses five locations.

3.2.4 Program Counter

The program counter is a 16-bit register that contains the address of the next instruction or operand to be fetched. The two most significant bits of the program counter are ignored internally.

Normally, the address in the program counter automatically increments to the next sequential memory location every time an instruction or operand is fetched. Jump, branch, and interrupt operations load the program counter with an address other than that of the next sequential location.

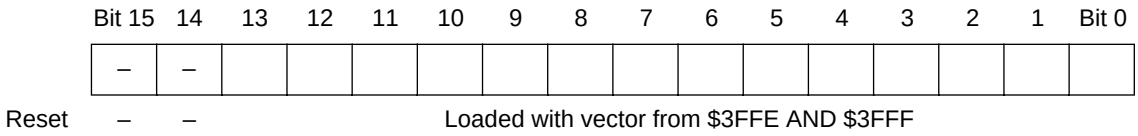


Figure 3-5 Program Counter

3.2.5 Condition Code Register

The condition code register is an 8-bit register whose three most significant bits are permanently fixed at 111. The condition code register contains the interrupt mask and four flags that indicate the results of the instruction just executed. The following paragraphs describe the functions of the condition code register.

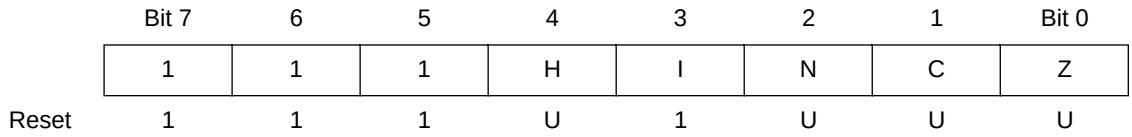


Figure 3-6 Condition Code Register

Half-Carry Flag

The CPU sets the half-carry flag when a carry occurs between bits 3 and 4 of the accumulator during an ADD or ADC operation. The half-carry flag is required for binary-coded decimal (BCD) arithmetic operations.

Interrupt Mask

Setting the interrupt mask disables interrupts. If an interrupt request occurs while the interrupt mask is logic zero, the CPU saves the CPU registers on the stack, sets the interrupt mask, and then fetches the interrupt vector. If an interrupt request occurs while the interrupt mask is set, the interrupt request is latched. Normally, the CPU processes the latched interrupt as soon as the interrupt mask is cleared again.

A return from interrupt (RTI) instruction pulls the CPU registers from the stack, restoring the interrupt mask to its cleared state. After any reset, the interrupt mask is set and can be cleared only by a software instruction.

Negative Flag

The CPU sets the negative flag when an arithmetic operation, logical operation, or data manipulation produces a negative result.

Zero Flag

The CPU sets the zero flag when an arithmetic operation, logical operation, or data manipulation produces a result of $00.

Carry/Borrow Flag

The CPU sets the carry/borrow flag when an addition operation produces a carry out of bit 7 of the accumulator or when a subtraction operation requires a borrow. Some logical operations and data manipulation instructions also clear or set the carry/borrow flag.

3.3 Arithmetic/Logic Unit (ALU)

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

The binary arithmetic circuits decode instructions and set up the ALU for the selected operation. Most binary arithmetic is based on the addition algorithm, carrying out subtraction as negative addition. Multiplication is not performed as a discrete operation but as a chain of addition and shift

operations within the ALU. The multiply instruction (MUL) requires 11 internal clock cycles to complete this chain of operations.

3.4 Instruction Set Overview

The MCU instruction set has 62 instructions and uses eight addressing modes. The instructions include all those of the M146805 CMOS Family plus one more: the unsigned multiply (MUL) instruction. The MUL instruction allows unsigned multiplication of the contents of the accumulator (A) and the index register (X). The high-order product is stored in the index register, and the low-order product is stored in the accumulator.

3.5 Addressing Modes

The CPU uses eight addressing modes for flexibility in accessing data. The addressing modes provide eight different ways for the CPU to find the data required to execute an instruction. The eight addressing modes are:

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

3.5.1 Inherent

Inherent instructions are those that have no operand, such as return from interrupt (RTI) and stop (STOP). Some of the inherent instructions act on data in the CPU registers, such as set carry flag (SEC) and

increment accumulator (INCA). Inherent instructions require no operand address and are one byte long.

3.5.2 Immediate

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

3.5.3 Direct

Direct instructions can access any of the first 256 memory locations with two bytes. The first byte is the opcode, and the second is the low byte of the operand address. In direct addressing, the CPU automatically uses $00 as the high byte of the operand address.

3.5.4 Extended

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

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

3.5.5 Indexed, No Offset

Indexed instructions with no offset are 1-byte instructions that can access data with variable addresses within the first 256 memory locations. The index register contains the low byte of the effective address of the operand. The CPU automatically uses $00 as the high byte, so these instructions can address locations $0000–$00FF.

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

3.5.6 Indexed, 8-Bit Offset

Indexed, 8-bit offset instructions are 2-byte instructions that can access data with variable addresses within the first 511 memory locations. The CPU adds the unsigned byte in the index register to the unsigned byte following the opcode. The sum is the effective address of the operand. These instructions can access locations $0000–$01FE.

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

3.5.7 Indexed,16-Bit Offset

Indexed, 16-bit offset instructions are 3-byte instructions that can access data with variable addresses at any location in memory. The CPU adds the unsigned byte in the index register to the two unsigned bytes following the opcode. The sum is the effective address of the operand. The first byte after the opcode is the high byte of the 16-bit offset; the second byte is the low byte of the offset.

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

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

3.5.8 Relative

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

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

3.6 Instruction Types

The MCU instructions fall into the following five categories:

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

3.6.1 Register/Memory Instructions

These instructions operate on CPU registers and memory locations. Most of them use two operands. One operand is in either the accumulator or the index register. The CPU finds the other operand in memory.

Table 3-1 Register/Memory Instructions

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

3.6.2 Read-Modify-Write Instructions

These instructions read a memory location or a register, modify its contents, and write the modified value back to the memory location or to the register.

NOTE: *Do not use read-modify-write operations on write-only registers.*

Table 3-2 Read-Modify-Write Instructions

Instruction	Mnemonic
Arithmetic Shift Left (Same as LSL)	ASL
Arithmetic Shift Right	ASR
Bit Clear	BCLR(1)
Bit Set	BSET(1)
Clear Register	CLR
Complement (One's Complement)	COM
Decrement	DEC
Increment	INC
Logical Shift Left (Same as ASL)	LSL
Logical Shift Right	LSR
Negate (Two's Complement)	NEG
Rotate Left through Carry Bit	ROL
Rotate Right through Carry Bit	ROR
Test for Negative or Zero	TST(2)

1. Unlike other read-modify-write instructions, BCLR and BSET use only direct addressing.
2. TST is an exception to the read-modify-write sequence because it does not write a replacement value.

3.6.3 Jump/Branch Instructions

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

The BRCLR and BRSET instructions cause a branch based on the state of any readable bit in the first 256 memory locations. These 3-byte instructions use a combination of direct addressing and relative addressing. The direct address of the byte to be tested is in the byte following the opcode. The third byte is the signed offset byte. The CPU finds the effective branch destination by adding the third byte to the program counter if the specified bit tests true. The bit to be tested and its condition (set or clear) is part of the opcode. The span of branching is from –128 to +127 from the address of the next location after the branch instruction. The CPU also transfers the tested bit to the carry/borrow bit of the condition code register.

Table 3-3 Jump and Branch Instructions

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

3.6.4 Bit Manipulation Instructions

The CPU can set or clear any writable bit in the first 256 bytes of memory, which includes I/O registers and on-chip RAM locations. The CPU can also test and branch based on the state of any bit in any of the first 256 memory locations.

Table 3-4 Bit Manipulation Instructions

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

3.6.5 Control Instructions

These instructions act on CPU registers and control CPU operation during program execution.

Table 3-5 Control Instructions

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

3.7 Instruction Set Summary

Table 3-6 Instruction Set Summary

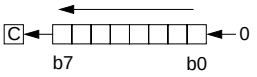
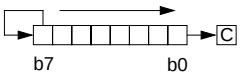
Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			H	I	N	Z	C				
ADC #*opr* ADC *opr* ADC *opr* ADC *opr*,X ADC *opr*,X ADC ,X	Add with Carry	A ← (A) + (M) + (C)	↕	—	↕	↕	↕	IMM DIR EXT IX2 IX1 IX	A9 B9 C9 D9 E9 F9	ii dd hh ll ee ff ff	2 3 4 5 4 3
ADD #*opr* ADD *opr* ADD *opr* ADD *opr*,X ADD *opr*,X ADD ,X	Add without Carry	A ← (A) + (M)	↕	—	↕	↕	↕	IMM DIR EXT IX2 IX1 IX	AB BB CB DB EB FB	ii dd hh ll ee ff ff	2 3 4 5 4 3
AND #*opr* AND *opr* AN*D opr* AND *opr*,X AND *opr*,X AND ,X	Logical AND	A ← (A) ∧ (M)	—	—	↕	↕	—	IMM DIR EXT IX2 IX1 IX	A4 B4 C4 D4 E4 F4	ii dd hh ll ee ff ff	2 3 4 5 4 3
ASL *opr* ASLA ASLX ASL *opr*,X ASL ,X	Arithmetic Shift Left (Same as LSL)		—	—	↕	↕	↕	DIR INH INH IX1 IX	38 48 58 68 78	dd ff 	5 3 3 6 5
ASR *opr* ASRA ASRX ASR *opr*,X ASR ,X	Arithmetic Shift Right		—	—	↕	↕	↕	DIR INH INH IX1 IX	37 47 57 67 77	dd ff 	5 3 3 6 5
BCC *rel*	Branch if Carry Bit Clear	PC ← (PC) + 2 + *rel* ? C = 0	—	—	—	—	—	REL	24	rr	3
BCLR *n opr*	Clear Bit n	Mn ← 0	—	—	—	—	—	DIR (b0) DIR (b1) DIR (b2) DIR (b3) DIR (b4) DIR (b5) DIR (b6) DIR (b7)	11 13 15 17 19 1B 1D 1F	dd dd dd dd dd dd dd dd	5 5 5 5 5 5 5 5
BCS *rel*	Branch if Carry Bit Set (Same as BLO)	PC ← (PC) + 2 + *rel* ? C = 1	—	—	—	—	—	REL	25	rr	3
BEQ *rel*	Branch if Equal	PC ← (PC) + 2 + *rel* ? Z = 1	—	—	—	—	—	REL	27	rr	3
BHCC *rel*	Branch if Half-Carry Bit Clear	PC ← (PC) + 2 + *rel* ? H = 0	—	—	—	—	—	REL	28	rr	3
BHCS *rel*	Branch if Half-Carry Bit Set	PC ← (PC) + 2 + *rel* ? H = 1	—	—	—	—	—	REL	29	rr	3
BHI *rel*	Branch if Higher	PC ← (PC) + 2 + *rel* ? C ∨ Z = 0	—	—	—	—	—	REL	22	rr	3
BHS *rel*	Branch if Higher or Same	PC ← (PC) + 2 + *rel* ? C = 0	—	—	—	—	—	REL	24	rr	3

Table 3-6 Instruction Set Summary (Continued)

Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			H	I	N	Z	C				
BIH *rel*	Branch if IRQ Pin High	PC ← (PC) + 2 + *rel* ? IRQ = 1	—	—	—	—	—	REL	2F	rr	3
BIL *rel*	Branch if IRQ Pin Low	PC ← (PC) + 2 + *rel* ? IRQ = 0	—	—	—	—	—	REL	2E	rr	3
BIT #*opr* BIT *opr* BIT *opr* BIT *opr*,X BIT *opr*,X BIT ,X	Bit Test Accumulator with Memory Byte	(A) ∧ (M)	—	—	↕◊	↕	—	IMM DIR EXT IX2 IX1 IX	A5 B5 C5 D5 E5 F5	ii dd hh ll ee ff ff	2 3 4 5 4 3
BLO *rel*	Branch if Lower (Same as BCS)	PC ← (PC) + 2 + *rel* ? C = 1	—	—	—	—	—	REL	25	rr	3
BLS *rel*	Branch if Lower or Same	PC ← (PC) + 2 + *rel* ? C ∨ Z = 1	—	—	—	—	—	REL	23	rr	3
BMC *rel*	Branch if Interrupt Mask Clear	PC ← (PC) + 2 + *rel* ? I = 0	—	—	—	—	—	REL	2C	rr	3
BMI *rel*	Branch if Minus	PC ← (PC) + 2 + *rel* ? N = 1	—	—	—	—	—	REL	2B	rr	3
BMS *rel*	Branch if Interrupt Mask Set	PC ← (PC) + 2 + *rel* ? I = 1	—	—	—	—	—	REL	2D	rr	3
BNE *rel*	Branch if Not Equal	PC ← (PC) + 2 + *rel* ? Z = 0	—	—	—	—	—	REL	26	rr	3
BPL *rel*	Branch if Plus	PC ← (PC) + 2 + *rel* ? N = 0	—	—	—	—	—	REL	2A	rr	3
BRA *rel*	Branch Always	PC ← (PC) + 2 + *rel* ? 1 = 1	—	—	—	—	—	REL	20	rr	3
BRCLR *n opr rel*	Branch if Bit n Clear	PC ← (PC) + 2 + *rel* ? Mn = 0	—	—	—	—	↕◊	DIR (b0) DIR (b1) DIR (b2) DIR (b3) DIR (b4) DIR (b5) DIR (b6) DIR (b7)	01 03 05 07 09 0B 0D 0F	dd rr dd rr dd rr dd rr dd rr dd rr dd rr dd rr	5 5 5 5 5 5 5 5
BRN *rel*	Branch Never	PC ← (PC) + 2 + *rel* ? 1 = 0	—	—	—	—	—	REL	21	rr	3
BRSET *n opr rel*	Branch if Bit n Set	PC ← (PC) + 2 + *rel* ? Mn = 1	—	—	—	—	◊↕	DIR (b0) DIR (b1) DIR (b2) DIR (b3) DIR (b4) DIR (b5) DIR (b6) DIR (b7)	00 02 04 06 08 0A 0C 0E	dd rr dd rr dd rr dd rr dd rr dd rr dd rr dd rr	5 5 5 5 5 5 5 5
BSET *n opr*	Set Bit n	Mn ← 1	—	—	—	—	—	DIR (b0) DIR (b1) DIR (b2) DIR (b3) DIR (b4) DIR (b5) DIR (b6) DIR (b7)	10 12 14 16 18 1A 1C 1E	dd dd dd dd dd dd dd dd	5 5 5 5 5 5 5 5
BSR *rel*	Branch to Subroutine	PC ← (PC) + 2; push (PCL) SP ← (SP) – 1; push (PCH) SP ← (SP) – 1 PC ← (PC) + *rel*	—	—	—	—	—	REL	AD	rr	6
CLC	Clear Carry Bit	C ← 0	—	—	—	—	0	INH	98		2
CLI	Clear Interrupt Mask	I ← 0	—	0	—	—	—	INH	9A		2

Table 3-6 Instruction Set Summary (Continued)

Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			H	I	N	Z	C				
CLR *opr* CLRA CLRX CLR *opr*,X CLR ,X	Clear Byte	M ← $00 A ← $00 X ← $00 M ← $00 M ← $00	—	—	0	1	—	DIR INH INH IX1 IX	3F 4F 5F 6F 7F	dd ff 	5 3 3 6 5
CMP #*opr* CMP *opr* CMP *opr* CMP *opr*,X CMP *opr*,X CMP ,X	Compare Accumulator with Memory Byte	(A) – (M)	—	—	↕◊	↕	↕	IMM DIR EXT IX2 IX1 IX	A1 B1 C1 D1 E1 F1	ii dd hh ll ee ff ff 	2 3 4 5 4 3
COM *opr* COMA COMX COM *opr*,X COM ,X	Complement Byte (One's Complement)	M ← ($\overline{M}$) = $FF – (M) A ← ($\overline{A}$) = $FF – (A) X ← ($\overline{X}$) = $FF – (X) M ← ($\overline{M}$) = $FF – (M) M ← ($\overline{M}$) = $FF – (M)	—	—	↕◊	↕◊	1	DIR INH INH IX1 IX	33 43 53 63 73	dd ff 	5 3 3 6 5
CPX #*opr* CPX *opr* CPX *opr* CPX *opr*,X CPX *opr*,X CPX ,X	Compare Index Register with Memory Byte	(X) – (M)	—	—	↕◊	◊↕	◊↕	IMM DIR EXT IX2 IX1 IX	A3 B3 C3 D3 E3 F3	ii dd hh ll ee ff ff 	2 3 4 5 4 3
DEC *opr* DECA DECX DEC *opr*,X DEC ,X	Decrement Byte	M ← (M) – 1 A ← (A) – 1 X ← (X) – 1 M ← (M) – 1 M ← (M) – 1	—	—	↕◊	↕◊	—	DIR INH INH IX1 IX	3A 4A 5A 6A 7A	dd ff 	5 3 3 6 5
EOR #*opr* EOR *opr* EOR *opr* EOR *opr*,X EOR *opr*,X EOR ,X	EXCLUSIVE OR Accumulator with Memory Byte	A ← (A) ⊕ (M)	—	—	↕◊	↕	—	IMM DIR EXT IX2 IX1 IX	A8 B8 C8 D8 E8 F8	ii dd hh ll ee ff ff 	2 3 4 5 4 3
INC *opr* INCA INCX INC *opr*,X INC ,X	Increment Byte	M ← (M) + 1 A ← (A) + 1 X ← (X) + 1 M ← (M) + 1 M ← (M) + 1	—	—	↕◊	↕◊	—	DIR INH INH IX1 IX	3C 4C 5C 6C 7C	dd ff 	5 3 3 6 5
JMP *opr* JMP *opr* JMP *opr*,X JMP *opr*,X JMP ,X	Unconditional Jump	PC ← Jump Address	—	—	—	—	—	DIR EXT IX2 IX1 IX	BC CC DC EC FC	dd hh ll ee ff ff 	2 3 4 3 2

Table 3-6 Instruction Set Summary (Continued)

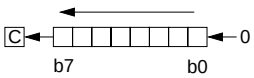
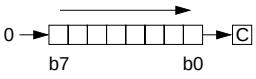
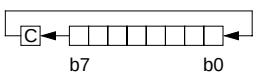
Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			H	I	N	Z	C				
JSR *opr* JSR *opr* JSR *opr*,X JSR *opr*,X JSR ,X	Jump to Subroutine	PC ← (PC) + n (n = 1, 2, or 3) Push (PCL); SP ← (SP) – 1 Push (PCH); SP ← (SP) – 1 PC ← Effective Address	—	—	—	—	—	DIR EXT IX2 IX1 IX	BD CD DD ED FD	dd hh ll ee ff ff	5 6 7 6 5
LDA #*opr* LDA *opr* LDA *opr* LDA *opr*,X LDA *opr*,X LDA ,X	Load Accumulator with Memory Byte	A ← (M)	—	—	↕◊	↕	—	IMM DIR EXT IX2 IX1 IX	A6 B6 C6 D6 E6 F6	ii dd hh ll ee ff ff	2 3 4 5 4 3
LDX #*opr* LDX *opr* LDX *opr* LDX *opr*,X LDX *opr*,X LDX ,X	Load Index Register with Memory Byte	X ← (M)	—	—	↕◊	↕◊	—	IMM DIR EXT IX2 IX1 IX	AE BE CE DE EE FE	ii dd hh ll ee ff ff	2 3 4 5 4 3
LSL *opr* LSLA LSLX LSL *opr*,X LSL ,X	Logical Shift Left (Same as ASL)		—	—	↕◊	↕	↕	DIR INH INH IX1 IX	38 48 58 68 78	dd ff 	5 3 3 6 5
LSR *opr* LSRA LSRX LSR *opr*,X LSR ,X	Logical Shift Right		—	—	0	↕	↕	DIR INH INH IX1 IX	34 44 54 64 74	dd ff 	5 3 3 6 5
MUL	Unsigned Multiply	X : A ← (X) × (A)	0	—	—	—	0	INH	42		11
NEG *opr* NEGA NEGX NEG *opr*,X NEG ,X	Negate Byte (Two's Complement)	M ← –(M) = $00 – (M) A ← –(A) = $00 – (A) X ← –(X) = $00 – (X) M ← –(M) = $00 – (M) M ← –(M) = $00 – (M)	—	—	↕◊	↕	↕	DIR INH INH IX1 IX	30 40 50 60 70	dd ff 	5 3 3 6 5
NOP	No Operation		—	—	—	—	—	INH	9D		2
ORA #*opr* ORA *opr* ORA *opr* ORA *opr*,X ORA *opr*,X ORA ,X	Logical OR Accumulator with Memory	A ← (A) ∨ (M)	—	—	↕◊	↕	—	IMM DIR EXT IX2 IX1 IX	AA BA CA DA EA FA	ii dd hh ll ee ff ff	2 3 4 5 4 3
ROL *opr* ROLA ROLX ROL *opr*,X ROL ,X	Rotate Byte Left through Carry Bit		—	—	↕◊	↕	↕	DIR INH INH IX1 IX	39 49 59 69 79	dd ff 	5 3 3 6 5

Table 3-6 Instruction Set Summary (Continued)

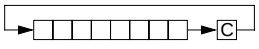
Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			H	I	N	Z	C				
ROR *opr* RORA RORX ROR *opr*,X ROR ,X	Rotate Byte Right through Carry Bit		—	—	↕◊	↕	↕	DIR INH INH IX1 IX	36 46 56 66 76	dd ff 	5 3 3 6 5
RSP	Reset Stack Pointer	SP ← $00FF	—	—	—	—	—	INH	9C		2
RTI	Return from Interrupt	SP ← (SP) + 1; Pull (CCR) SP ← (SP) + 1; Pull (A) SP ← (SP) + 1; Pull (X) SP ← (SP) + 1; Pull (PCH) SP ← (SP) + 1; Pull (PCL)	↕◊	↕	↕	↕	↕	INH	80		9
RTS	Return from Subroutine	SP ← (SP) + 1; Pull (PCH) SP ← (SP) + 1; Pull (PCL)	—	—	—	—	—	INH	81		6
SBC #*opr* SBC *opr* SBC *opr* SBC *opr*,X SBC *opr*,X SBC ,X	Subtract Memory Byte and Carry Bit from Accumulator	A ← (A) – (M) – (C)	—	—	◊↕	↕	↕	IMM DIR EXT IX2 IX1 IX	A2 B2 C2 D2 E2 F2	ii dd hh ll ee ff ff	2 3 4 5 4 3
SEC	Set Carry Bit	C ← 1	—	—	—	—	1	INH	99		2
SEI	Set Interrupt Mask	I ← 1	—	1	—	—	—	INH	9B		2
STA *opr* STA *opr* STA *opr*,X STA *opr*,X STA ,X	Store Accumulator in Memory	M ← (A)	—	—	↕◊	↕	—	DIR EXT IX2 IX1 IX	B7 C7 D7 E7 F7	dd hh ll ee ff ff	4 5 6 5 4
STOP	Stop Oscillator and Enable IRQ Pin		—	0	—	—	—	INH	8E		2
STX *opr* STX *opr* STX *opr*,X STX *opr*,X STX ,X	Store Index Register In Memory	M ← (X)	—	—	↕◊	↕	—	DIR EXT IX2 IX1 IX	BF CF DF EF FF	dd hh ll ee ff ff	4 5 6 5 4
SUB #*opr* SUB *opr* SUB *opr* SUB *opr*,X SUB *opr*,X SUB ,X	Subtract Memory Byte from Accumulator	A ← (A) – (M)	—	—	↕	↕	↕	IMM DIR EXT IX2 IX1 IX	A0 B0 C0 D0 E0 F0	ii dd hh ll ee ff ff	2 3 4 5 4 3
SWI	Software Interrupt	PC ← (PC) + 1; Push (PCL) SP ← (SP) – 1; Push (PCH) SP ← (SP) – 1; Push (X) SP ← (SP) – 1; Push (A) SP ← (SP) – 1; Push (CCR) SP ← (SP) – 1; I ← 1 PCH ← Interrupt Vector High Byte PCL ← Interrupt Vector Low Byte	—	1	—	—	—	INH	83		10
TAX	Transfer Accumulator to Index Register	X ← (A)	—	—	—	—	—	INH	97		2

Table 3-6 Instruction Set Summary (Continued)

Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			H	I	N	Z	C				
TST *opr* TSTA TSTX TST *opr*,X TST ,X	Test Memory Byte for Negative or Zero	(M) – $00	—	—	↕	↕	—	DIR INH INH IX1 IX	3D 4D 5D 6D 7D	dd ff 	4 3 3 5 4
TXA	Transfer Index Register to Accumulator	A ← (X)	—	—	—	—	—	INH	9F		2
WAIT	Stop CPU Clock and Enable Interrupts		—	0 ◊	—	—	—	INH	8F		2

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

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

Table 3-7 . Opcode Map

	Bit Manipulation		Branch	Read-Modify-Write					Control		Register/Memory						
	DIR	DIR	REL	DIR	INH	INH	IX1	IX	INH	INH	IMM	DIR	EXT	IX2	IX1	IX	
MSB / LSB	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	MSB / LSB
0	5 BRSET0 3 DIR	5 BSET0 2 DIR	3 BRA 2 REL	5 NEG 2 DIR	3 NEGA 1 INH	3 NEGX 1 INH	6 NEG 2 IX1	5 NEG 1 IX	9 RTI 1 INH		2 SUB 2 IMM	3 SUB 2 DIR	4 SUB 3 EXT	5 SUB 3 IX2	4 SUB 2 IX1	3 SUB 1 IX	0
1	5 BRCLR0 3 DIR	5 BCLR0 2 DIR	3 BRN 2 REL						6 RTS 1 INH		2 CMP 2 IMM	3 CMP 2 DIR	4 CMP 3 EXT	5 CMP 3 IX2	4 CMP 2 IX1	3 CMP 1 IX	1
2	5 BRSET1 3 DIR	5 BSET1 2 DIR	3 BHI 2 REL		11 MUL 1 INH						2 SBC 2 IMM	3 SBC 2 DIR	4 SBC 3 EXT	5 SBC 3 IX2	4 SBC 2 IX1	3 SBC 1 IX	2
3	5 BRCLR1 3 DIR	5 BCLR1 2 DIR	3 BLS 2 REL	5 COM 2 DIR	3 COMA 1 INH	3 COMX 1 INH	6 COM 2 IX1	5 COM 1 IX	10 SWI 1 INH		2 CPX 2 IMM	3 CPX 2 DIR	4 CPX 3 EXT	5 CPX 3 IX2	4 CPX 2 IX1	3 CPX 1 IX	3
4	5 BRSET2 3 DIR	5 BSET2 2 DIR	3 BCC 2 REL	5 LSR 2 DIR	3 LSRA 1 INH	3 LSRX 1 INH	6 LSR 2 IX1	5 LSR 1 IX			2 AND 2 IMM	3 AND 2 DIR	4 AND 3 EXT	5 AND 3 IX2	4 AND 2 IX1	3 AND 1 IX	4
5	5 BRCLR2 3 DIR	5 BCLR2 2 DIR	3 BCS/BLO 2 REL								2 BIT 2 IMM	3 BIT 2 DIR	4 BIT 3 EXT	5 BIT 3 IX2	4 BIT 2 IX1	3 BIT 1 IX	5
6	5 BRSET3 3 DIR	5 BSET3 2 DIR	3 BNE 2 REL	5 ROR 2 DIR	3 RORA 1 INH	3 RORX 1 INH	6 ROR 2 IX1	5 ROR 1 IX			2 LDA 2 IMM	3 LDA 2 DIR	4 LDA 3 EXT	5 LDA 3 IX2	4 LDA 2 IX1	3 LDA 1 IX	6
7	5 BRCLR3 3 DIR	5 BCLR3 2 DIR	3 BEQ 2 REL	5 ASR 2 DIR	3 ASRA 1 INH	3 ASRX 1 INH	6 ASR 2 IX1	5 ASR 1 IX		2 TAX 1 INH		4 STA 2 DIR	5 STA 3 EXT	6 STA 3 IX2	5 STA 2 IX1	4 STA 1 IX	7
8	5 BRSET4 3 DIR	5 BSET4 2 DIR	3 BHCC 2 REL	5 ASL/LSL 2 DIR	3 ASLA/LSLA 1 INH	3 ASLX/LSLX 1 INH	6 ASL/LSL 2 IX1	5 ASL/LSL 1 IX		2 CLC 1 INH	2 EOR 2 IMM	3 EOR 2 DIR	4 EOR 3 EXT	5 EOR 3 IX2	4 EOR 2 IX1	3 EOR 1 IX	8
9	5 BRCLR4 3 DIR	5 BCLR4 2 DIR	3 BHCS 2 REL	5 ROL 2 DIR	3 ROLA 1 INH	3 ROLX 1 INH	6 ROL 2 IX1	5 ROL 1 IX		2 SEC 1 INH	2 ADC 2 IMM	3 ADC 2 DIR	4 ADC 3 EXT	5 ADC 3 IX2	4 ADC 2 IX1	3 ADC 1 IX	9
A	5 BRSET5 3 DIR	5 BSET5 2 DIR	3 BPL 2 REL	5 DEC 2 DIR	3 DECA 1 INH	3 DECX 1 INH	6 DEC 2 IX1	5 DEC 1 IX		2 CLI 1 INH	2 ORA 2 IMM	3 ORA 2 DIR	4 ORA 3 EXT	5 ORA 3 IX2	4 ORA 2 IX1	3 ORA 1 IX	A
B	5 BRCLR5 3 DIR	5 BCLR5 2 DIR	3 BMI 2 REL							2 SEI 1 INH	2 ADD 2 IMM	3 ADD 2 DIR	4 ADD 3 EXT	5 ADD 3 IX2	4 ADD 2 IX1	3 ADD 1 IX	B
C	5 BRSET6 3 DIR	5 BSET6 2 DIR	3 BMC 2 REL	5 INC 2 DIR	3 INCA 1 INH	3 INCX 1 INH	6 INC 2 IX1	5 INC 1 IX		2 RSP 1 INH		2 JMP 2 DIR	3 JMP 3 EXT	4 JMP 3 IX2	3 JMP 2 IX1	2 JMP 1 IX	C
D	5 BRCLR6 3 DIR	5 BCLR6 2 DIR	3 BMS 2 REL	4 TST 2 DIR	3 TSTA 1 INH	3 TSTX 1 INH	5 TST 2 IX1	4 TST 1 IX		2 NOP 1 INH	6 BSR 2 REL	5 JSR 2 DIR	6 JSR 3 EXT	7 JSR 3 IX2	6 JSR 2 IX1	5 JSR 1 IX	D
E	5 BRSET7 3 DIR	5 BSET7 2 DIR	3 BIL 2 REL						2 STOP 1 INH		2 LDX 2 IMM	3 LDX 2 DIR	4 LDX 3 EXT	5 LDX 3 IX2	4 LDX 2 IX1	3 LDX 1 IX	E
F	5 BRCLR7 3 DIR	5 BCLR7 2 DIR	3 BIH 2 REL	5 CLR 2 DIR	3 CLRA 1 INH	3 CLRX 1 INH	6 CLR 2 IX1	5 CLR 1 IX	2 WAIT 1 INH	2 TXA 1 INH		4 STX 2 DIR	5 STX 3 EXT	6 STX 3 IX2	5 STX 2 IX1	4 STX 1 IX	F

INH = InherentREL = Relative
IMM = ImmediateIX = Indexed, No Offset
DIR = DirectIX1 = Indexed, 8-Bit Offset
EXT = ExtendedIX2 = Indexed, 16-Bit Offset

LSB of Opcode in Hexadecimal

MSB / LSB	0
0	5 BRSET0 3 DIR

MSB of Opcode in Hexadecimal
Number of Cycles
Opcode Mnemonic
Number of Bytes/Addressing Mode

Section 4. Interrupts

4.1 Contents

4.2 Introduction

The MCU can be interrupted in different ways:

1. Nonmaskable Software Interrupt Instruction (SWI)
2. External Asynchronous Interrupt (IRQ)
3. External Asynchronous Interrupt on Port A
4. External Asynchronous Interrupt on Port C
5. Internal 8-bit Timer Interrupt (CTIMER)
6. Internal 16-bit Timer1 Interrupt (TIMER)
7. Low Voltage Interrupt
8. Port C5 & C6 Short Circuit Interrupt
9. High Voltage Interrupt
10. High Temperature Interrupt

4.3 CPU Interrupt Processing

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

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

When an interrupt is to be processed, the CPU fetches the address of the appropriate interrupt software service routine from the vector table at locations $3FF0 through $3FFF as defined in **Table 4-1**.

Table 4-1 Reset/Interrupt Vector Addresses

Function	Source	Local Mask	Global Mask	Priority (1 = Highest)	Vector Address
Reset	Power-On Logic	None	None	1	$3FFE–$3FFF
	$\overline{\text{RESET}}$ Pin				
	COP Watchdog				
Software Interrupt (SWI)	User Code	None	None	Same Priority As Instruction	$3FFC–$3FFD
External Interrupt	$\overline{\text{IRQ}}$ Pin	INTE Bit	I-Bit	2	$3FFA–$3FFB
Core Timer Interrupts	RTIF	RTIE Bit	I-Bit	3	$3FF8–$3FF9
	TOF	TOFE Bit			
16-Bit Timer Interrupts	ICF Bits	ICIE Bits	I-Bit	4	$3FF6–$3FF7
	OCF Bits	OCIE Bits			
	TOF Bit	TOIE Bit			
Voltage, Temperature and Port C Short circuit Interrupts	HTI Bit	HTIM Bit	I-Bit	5	$3FF4–$3FF5
	HVI Bit	HVIM Bit			
	LVI Bit	LVIM Bit			
	SCIF6	SCIE6			
	SCIF5	SCIE5			
Port A High Nibble Interrupt	Port A4–7	PAHIE Bit	I-Bit	6	$3FF2–$3FF3
Port A Low Nibble Interrupt	Port A0–3	PALIE Bit			
Port C Current Sense/HV Inputs	CSIF	CSIE	I-Bit	7	$3FF0–$3FF1

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

Latency = (Longest instruction execution time + 10) x t_{CYC}

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

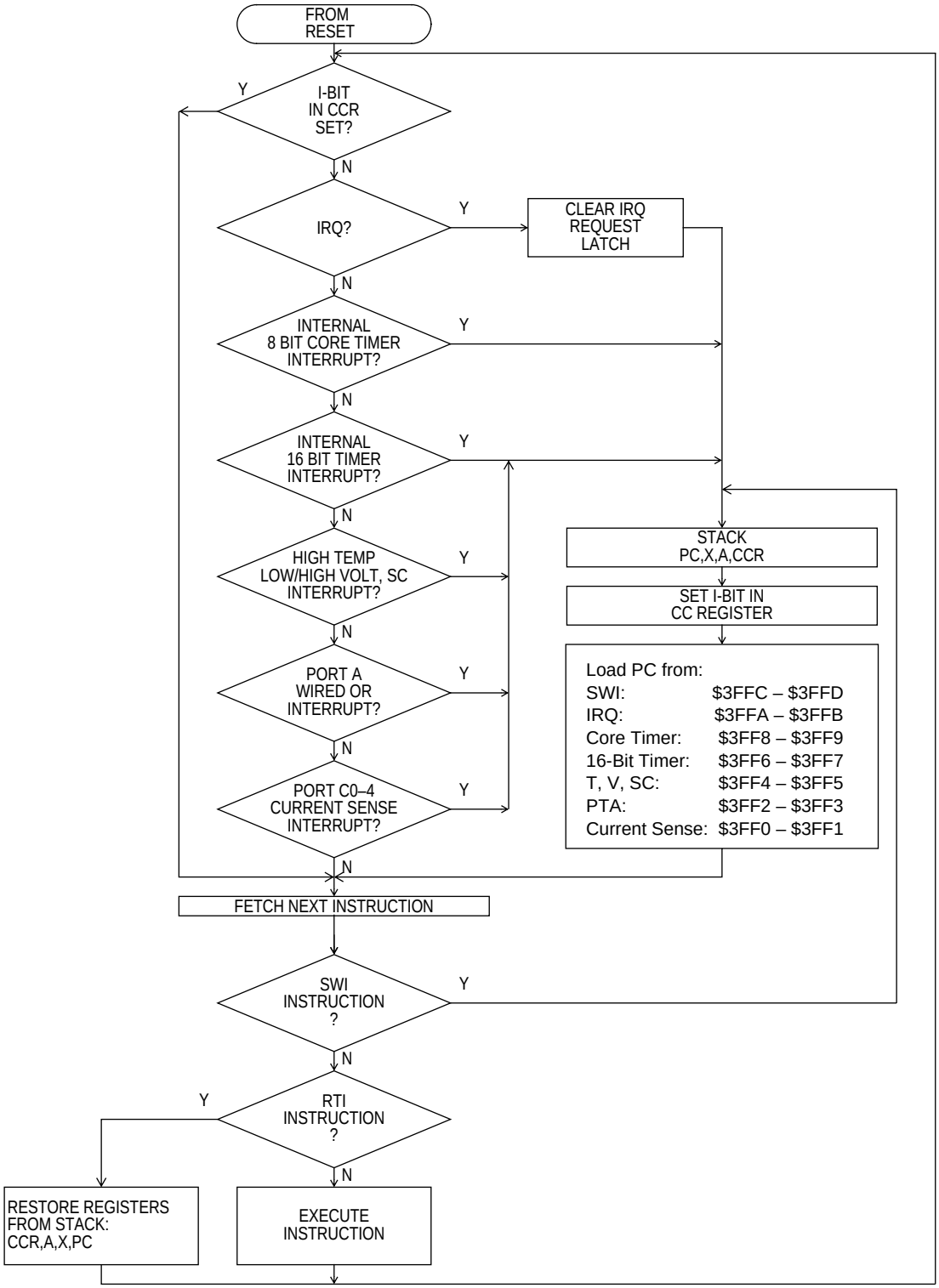


Figure 4-1 Interrupt Processing Flowchart

4.4 Reset Interrupt Sequence

The reset function is not in the strictest sense an interrupt; however, it is acted upon in a similar manner as shown in **Figure 4-1**. A low level input on the $\overline{\text{RESET}}$ 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 described in **Section 5. Resets**.

4.5 Software Interrupt (SWI)

The SWI is an executable instruction and a non-maskable interrupt since it is executed regardless of the state of the I-bit in the CCR. If the I-bit is zero (interrupts enabled), the SWI instruction executes after interrupts which were pending before the SWI was fetched, or before interrupts generated after the SWI was fetched. The interrupt service routine address is specified by the contents of memory locations $3FFC and $3FFD.

4.6 Hardware Interrupts

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

4.7 External Interrupt (IRQ)

If the interrupt mask bit (I-bit) of the CCR has been cleared and the interrupt enable bit is set (INTE bit) and the signal of the external interrupt pin ($\overline{\text{IRQ}}$) satisfies the condition selected by the option control bits (INTP and INTN), then the external interrupt is recognized. INTE, INTP and INTN are all bits contained in the system control register located at $000A. When the interrupt is recognized, the current state of

the CPU is pushed onto the stack and the I-bit is set. This masks further interrupts until the present one is serviced. The interrupt service routine address is specified by the contents of memory locations $3FFA and $3FFB.

$000A	Bit 7	6	5	4	3	2	1	Bit 0
Read:	POR	INTP	INTN	INTE	WCOP	WCP	FPIE	FPICLK
Write:								
Reset:	0	0	0	0	0	0	0	0

Figure 4-2 System Control Register (SYSCTRL)

INTP, INTN – External interrupt sensitivity options

These two bits allow the user to select which edge of the $\overline{IRQ}$ pin is sensitive as shown in **Table 4-1**. Both bits can be written only while the I-bit is set, and are cleared by power-on or external reset. Therefore the device is initialized with negative edge and low level sensitivity.

Table 4-2 $\overline{IRQ}$ sensitivity

INTP	INTN	IRQ sensitivity
0	0	Negative edge and low level sensitive
0	1	Negative edge only
1	0	Positive edge only
1	1	Positive and negative edge sensitive

INTE – External interrupt enable

1 = External interrupt function ($\overline{IRQ}$) enabled.
0 = External interrupt function ($\overline{IRQ}$) disabled.

The INTE bit can be written to only while the I-bit is set, and is set by power-on or external reset, thus enabling the external interrupt function.

Table 4-1 describes the various triggering options available for the IRQ pin, however it is important to re-emphasize here that in order to avoid any conflict and spurious interrupt, it is only possible to change the external interrupt options while the I-bit is set. Any attempt to change the external interrupt option while the I-bit is clear will be unsuccessful. If an external interrupt is pending, it will automatically be cleared when selecting a different interrupt option.

NOTE: *If the external interrupt function is disabled by the INTE bit and an external interrupt is sensed by the edge detection circuitry, then the interrupt request is latched and the interrupt stays pending until the INTE bit is set. The external latch of the external interrupt is cleared in the first part of the service routine (except for the low level interrupt which is not latched); therefore only one external interrupt pulse can be latched during t_{ILIL} and serviced as soon as the I-bit is cleared.*

4.8 8-Bit Timer Interrupt

This timer can create two types of interrupts. A timer overflow interrupt occurs whenever the 8 bit timer rolls over from $FF to $00 and the enable bit TOFE is set. A real time interrupt occurs whenever the programmed time elapses and the enable bit RTIE is set. This interrupt vector to the interrupt service routine located at the address specified by the contents of memory location $3FF8 and $3FF9.

For details see **Section 8. Core Timer**.

4.8.1 16-Bit Timer Interrupt

There are five different timer interrupt flags that cause a 16-bit 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 register1 (TCR1). Any of these interrupts vectors to the same interrupt service routine, located at the address specified by the contents of memory location $3FF6 and $3FF7.

For details see **Section 9. 16-Bit Programmable Timer**.

4.9 Ambient Exception Interrupts

There are three different interrupt flags that cause an environmental exception interrupt whenever they are set and enabled. The interrupt flags are in the reset/interrupt status register (INTSR), and the enable bits are in the interrupt control register (INTCR). Any of these interrupts vectors to the same interrupt service routine, located at the address specified by the contents of memory location $3FF4 and $3FF5.

$0028	Bit 7	6	5	4	3	2	1	Bit 0
Read:	0	0	0	0	0	HTIE	HVIE	LVIE
Write:								
Reset:	0	0	0	0	0	0	0	0

Figure 4-3 Interrupt Control Register (INTCR)

$0029	Bit 7	6	5	4	3	2	1	Bit 0
Read:	RCON	PC4CL	0	0	0	HTIF	HVIF	LVIF
Write:								
Reset:	?	0	0	0	0	?	0	0

Figure 4-4 Interrupt Status Register (INTSR)

4.10 High Temperature Interrupt

HTIF – High Temperature Interrupt Flag

This bit is set if the die temperature is higher than the upper trip point and cleared again if the die temperature falls below the lower trip point of the HTI.

1 = The die temperature is higher than T_{HTION}

0 = The die temperature is lower than T_{HTIOFF}

HTIE – High Temperature Interrupt Enable

This bit enables/disables the high temperature interrupt. Once this interrupt is acknowledged, the enable bit should be cleared and the high temperature interrupt flag should be monitored until the bit is cleared.

1 = High temperature interrupt enabled
0 = High temperature interrupt disabled

4.10.1 High Voltage Interrupt

HVIF – High Voltage Interrupt Flag

This bit is set if the supply voltage V_{SUP} is higher than the upper trip point and cleared again if the voltage falls below the lower trip point of the HVI.

1 = The supply voltage is higher than V_{HVION}
0 = The supply voltage is lower than V_{HIOFF}

HVIE – High Voltage Interrupt Enable

This bit enables/disables the high voltage interrupt. Once this interrupt is acknowledged, the enable bit should be cleared and the high voltage interrupt flag should be monitored until the bit is cleared.

1 = High voltage interrupt enabled
0 = High voltage interrupt disabled

4.10.2 Low Voltage Interrupt

LVIF – Low Voltage Interrupt Flag

This bit is set if the supply voltage V_{SUP} is lower than the lower trip point and cleared again if the voltage rises above the upper trip point of the LVI.

1 = The supply voltage is lower than V_{LVION}
0 = The supply voltage is higher than $V_{LVIOFFf}$

LVIE – Low Voltage Interrupt Enable

This bit enables/disables the low voltage interrupt. Once this interrupt is acknowledged, the enable bit should be cleared and the low voltage interrupt flag should be monitored until the bit is cleared.

1 = Low voltage interrupt enabled
0 = Low voltage interrupt disabled

4.10.3 Power Driver Short CIrcuit Interrupt

There are two different interrupt flags that cause a power driver short circuit interrupt whenever they are set and enabled. The interrupt flags are located in the port C status register, and the enable bits are located in the port C configuration register 1. Any of these interrupts vector to the same interrupt service routine, located at the address specified by the contents of memory location $3FF4 and $3FF5.

For details see **7.6 Port C (High Voltage Port)**.

4.11 Keyboard Interrupts

When configured as input pins, PA0–7 provide a wired-OR keyboard interrupt facility and generate an interrupt provided the interrupt enable bits (PALIE or PAHIE) in the port A configuration register are set.

The interrupt vector for this interrupt is located at $3FF2 and $3FF3. Further information on the keyboard interrupt facility can be found in **7.4 Port A**.

4.12 Port C Current Sense Interrupt

There is an interrupt flag that causes a current sense interrupt whenever it is set and enabled. This interrupt flag is a wired-OR of the active current sense inputs. The interrupt flag is located in the port C status register, and the enable bit is located in the port C configuration register 1. This interrupt vectors to the memory location $3FF0 and $3FF1. Whenever a PCxCS bit is set, but the corresponding pin is not configured as an output, the signal for the corresponding CSDx bit, and

therefore for the current sense interrupt, is derived from the high-voltage input circuit.

For details see **7.6 Port C (High Voltage Port)**.

4.13 STOP and WAIT Modes

All modules that are capable of generating interrupts in STOP or WAIT mode can only do so when configured properly. The I-bit is automatically cleared when STOP or WAIT mode is entered. Environmental exception interrupts and interrupts detected on port A and port C are recognized in STOP or WAIT modes.

Section 5. Resets

5.1 Contents

5.2 Introduction

The MCU can be reset from nine sources: one external input and eight internal restart conditions. The RESET pin is an input with a Schmitt trigger. All the internal peripheral modules are reset by the internal reset signal (RST). Refer to **Figure 5-2** for reset timing details.

5.3 Reset status register (RSR)

This register contains eight flags that show the source of the last reset. A power-on reset sets the POR bit in the system control register and clears all other bits in the reset status register. All bits can be cleared by writing a one to the corresponding bit. Uncleared bits remain set as long as they are not cleared by a power-on reset or by software.

$002A	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	PINR	STOPR	COPR	ILINR	CMR	HTR	HVR	LVR
POR:	0	0	0	0	0	0	0	0

Figure 5-1 Reset Status Register (RSR)

PINR – External Reset Bit

1 = Last reset caused by external reset pin ($\overline{\text{RESET}}$)
0 = No pin reset since PINR was cleared by software or POR

STOPR – Illegal STOP Instruction Reset Bit

Indicates the last reset was caused by a disabled STOP instruction.

1 = Last reset caused by a disabled STOP instruction
0 = No illegal STOP instruction since STOPR was cleared by software or POR

COPR – COP (Computer Operating Properly) Reset Bit

1 = Last reset caused by COP
0 = No COP reset since COPR was cleared by software or POR

ILINR – Illegal Instruction Reset Bit

1 = Last reset caused by an instruction fetch from an illegal address
0 = No illegal instruction fetch reset since ILINR was cleared by software or POR

CMR – Clock Monitor Reset Bit

1 = Last reset caused by the clock monitor due to a failure on system clock or system clock is back. Refer to RCON status bit in the interrupt status register
0 = No clock monitor reset since CMR was cleared by software or POR

HTR – High Temperature Reset Bit

1 = Last reset caused by high temperature detect circuitry

0 = No high temperature reset since HTR was cleared by software or POR

HVR – High Voltage Reset Bit

1 = Last reset caused by high voltage detect circuitry

0 = No high voltage reset since HVR is cleared by software or POR

LVR – Low Voltage Reset Bit

1 = Last reset caused by low voltage detect circuitry

0 = No low voltage reset since LVR was cleared by software or POR

Note: If the cause of an environmental reset only lasts for a short time and if there is an external capacitor on the RESET pin, the corresponding bit in the reset status register may be set without occurrence of a reset.

5.4 External Reset ($\overline{\text{RESET}}$)

The $\overline{\text{RESET}}$ pin is the only external source of a reset. This pin is connected to a Schmitt trigger input gate to provide an upper and lower threshold voltage separated by a minimum amount of hysteresis. This external reset occurs whenever the $\overline{\text{RESET}}$ pin is pulled below the lower threshold and remains in reset until the $\overline{\text{RESET}}$ pin rises above the upper threshold. This active low input generates the RST signal and resets the CPU and peripherals.

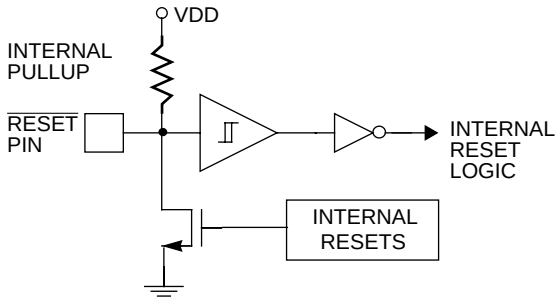
Activation of the RST signal is generally referred to as reset of the device, unless otherwise specified.

The $\overline{\text{RESET}}$ pin can also act as an open drain output. It is pulled to a low state by an internal pull-down that is activated by any reset source. This $\overline{\text{RESET}}$ pull-down device is asserted until the internal reset source is deasserted and the reset is internally recognized.

5.5 Internal Resets

The eight internally generated resets are the initial power-on reset function, the COP watchdog timer reset, the illegal address detector, clock-monitor, the high temperature reset, high voltage reset, low-voltage reset, and the disabled STOP instruction.

When forcing $\overline{\text{RESET}}$ externally to V_{DD}, all temperature, voltage and clock-monitor dependent reset sources are disabled. In this case, the internal pull-down device tries to pull down the pin until the next recognized internal reset, which leads to some power-consumption.

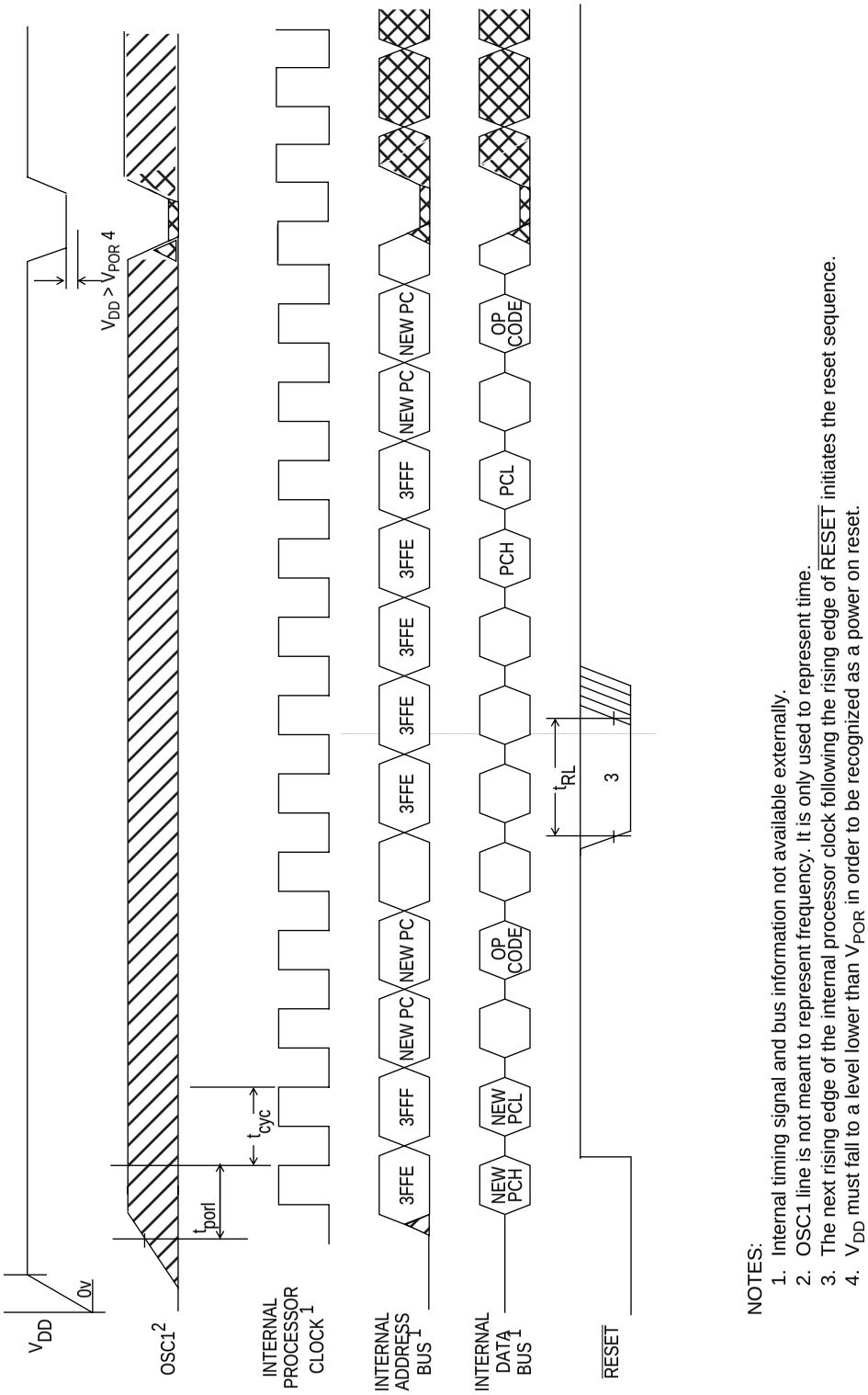


5.6 Power-On Reset (POR)

The internal POR is generated on power-up to allow the clock oscillator to stabilize. The POR is strictly for power turn-on conditions and is not able to detect a drop in the power supply voltage (brown-out). There is an oscillator stabilization delay of t_{PORL} after the oscillator becomes active. See **Figure 5-2** for details. T_{PORL} is 4064 internal processor clock cycles.

The POR generates the RST signal which resets the CPU. If any other reset function is active at the end of this t_{PORL} delay, the RST signal remains in the reset condition until the other reset condition(s) ends.

POR activates the $\overline{\text{RESET}}$ pin pull-down device connected to the pin. VDD must drop below V_{POR} in order for the internal POR circuit to detect the next rise of V_{DD}.



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

Figure 5-2 RESET and POR Timing Diagram

5.7 Computer Operating Properly Reset (COPR)

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

The COP reset function is enabled or disabled by the MOR[COPE] bit and is verified during production testing.

The COP watchdog reset activates the internal pull-down device connected to the $\overline{\text{RESET}}$ pin.

The window COP function can be enabled via the WCOP bit in the system control register. This bit is a write once bit, e.g. the WCOP feature stays enabled until the next system reset. In case of WCOP bit enabled, the COP timer is only reset when the write to the COPEN bit in the mask option register occurs in the second half of the COP watchdog time. A write in the first half causes a system reset with the COPR bit set. The phase of the COP timer can be monitored via the WCP (window COP phase) in the system control register. A 0 indicates that writing to the MOR bit causes system reset. A 1 indicates that writing to the MOR bit causes a reset of the COP timer cycle.

5.7.1 Resetting the COP

A COP reset is prevented by writing a 0 to the COPR bit. This action resets the counter and begin the time-out period again. The COPR bit is bit 0 of address $3FF0. A read of address $3FF0 returns user data programmed at that location.

5.7.2 COP During WAIT Mode

The COP continues to operate normally during WAIT mode. The system should be configured to pull the device out of WAIT mode periodically and reset the COP by writing to the COPR bit to prevent a COP reset.

5.7.3 COP During STOP Mode

When the STOP enable mask option is selected, STOP mode disables the oscillator circuit and thereby turns the clock off for the entire device. The COP counter is reset when STOP mode is entered. If a reset is used to exit STOP mode, the COP counter is held in reset during the 4064 cycles of start up delay. If any operable interrupt is used to exit STOP mode, the COP counter is not reset during the 4064 cycle start-up delay and has the number of cycles already counted when control is returned to the program.

5.7.4 COP Watchdog Timer Considerations

The COP watchdog timer is active in user mode if enabled by the COPEN bit in the mask option register. If the COP watchdog timer is selected, any execution of the STOP instruction (either intentional or inadvertent due to the CPU being disturbed) causes the oscillator to halt and prevent the COP watchdog timer from timing out. Therefore, it is recommended that the STOP instruction should be disabled if the COP watchdog timer is enabled.

If the COP watchdog timer is selected, the COP resets the MCU when it times out. Therefore, it is recommended that the COP watchdog be disabled for a system that must use the WAIT mode for periods longer than the COP time-out period.

5.7.5 COP Register

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

$3FF0	Bit 7	6	5	4	3	2	1	Bit 0
Read:								
Write:								COPR
Reset:								

Figure 5-3 COP Watchdog Timer Location Register (COPR)

5.8 Illegal Address Reset

An illegal address reset is generated when the CPU attempts to fetch an instruction from either unimplemented address space ($0100 to $017F, $0200 to $1FFF) monitor ROM ($3F00 to $3FEF) or I/O address space ($0000 to $003F).

The illegal address reset activates the internal pull-down device connected to the RESET pin.

5.9 Disabled STOP Instruction Reset

When the mask option is selected to disable the STOP instruction, execution of a STOP instruction results in an internal reset. This activates the internal pull-down device connected to the $\overline{\text{RESET}}$ pin.

5.10 High Temperature Reset

The internal high temperature (HTR) reset is generated when the die temperature rises above the high temperature threshold T_{HTON}. This condition remains active until the temperature falls below the threshold T_{HTOFF}.

This reset can be disabled by using a mask option.

5.11 High Voltage Reset

The internal high voltage (HVR) reset is generated when the supply voltage V_{SUP} rises above the high voltage reset threshold V_{HVRON}. This condition remains active until the supply voltage falls below the threshold V_{HVROFF}.

This reset can be disabled by using a mask option.

5.12 Low Voltage Reset

The internal low voltage (LVR) reset is generated when the supply voltage V_{DD} falls below the low voltage threshold V_{LVRON}. This condition remains active until the voltage rises above the threshold V_{LVROFF} or a proper power-on sequence occurs.

5.13 Operation in STOP and WAIT Mode

If enabled, all reset sources remain active during STOP and WAIT. Any reset source can bring the MCU out of STOP or WAIT modes.

Since no instructions are executed in WAIT or STOP mode the illegal address reset and the stop disabled reset can not become active in STOP or WAIT mode.

Since the core timer is not active in STOP mode, the COP reset can not become active in STOP mode.

5.14 Clock Monitor Reset (CMR)

The clock monitor reset is based on an internal RC time delay. If no MCU clock edges are detected within this RC time delay, the clock monitor can optionally generate a system reset. The system clock is then automatically switched to an on-chip RC oscillator. The clock monitor function is enabled via a mask option bit. Clock monitor is used as a backup for the COP system. Because the COP needs a clock to function

it is disabled when the clock stops. Therefore, the clock monitor system can detect clock failures not detected by the COP system.

Semiconductor wafer processing causes variations of the RC timeout values between individual devices. A processor clock frequency below 10 KHz is detected as a clock monitor error. A processor clock frequency of 400 KHz or more prevents clock monitor errors. Using the clock monitor when the processor clock is below 400 KHz is not recommended.

The oscillator used for deriving the system clock can be determined by the RCON Bit in the interrupt status register.

$0029	Bit 7	6	5	4	3	2	1	Bit 0
Read:	RCON	PC4CL	0	0	0	HTIF	HVIF	LVIF
Write:								
Reset:	U	0	0	0	0	0	0	0

Figure 5-4 Interrupt Status Register (INTSR)

Section 6. Operating Modes

6.1 Contents

6.2 Introduction

The normal operating mode of the MC68HC(8)05PV8 is user (or single chip) mode. There is also a monitor mode, primarily for programming and evaluation purposes. In addition to these modes, there are two low power modes which may be entered and exited at will from user mode: STOP and WAIT. **Table 6-1** shows the conditions required to enter the modes of operation on the rising edge of $\overline{\text{RESET}}$, where VTST = 2 x VDD.

Table 6-1 Operating Mode Entry Conditions

IRQ	PB0	Mode
VSS to VDD	VSS to VDD	User
VTST	VDD	Monitor

6.3 User mode

Intended mode of operation for executing user firmware.

6.4 Monitor Mode

Used for programming the on-chip EEPROM.

6.5 Low Power Modes

The MC68HC(8)05PV8 is capable of running in one of several low-power operational modes. The WAIT and STOP instructions provide two modes that reduce the power required for the MCU by stopping various internal clocks and/or the on-chip oscillator. The flows of the STOP and WAIT modes are shown in **Figure 6-2**.

6.5.1 STOP Mode

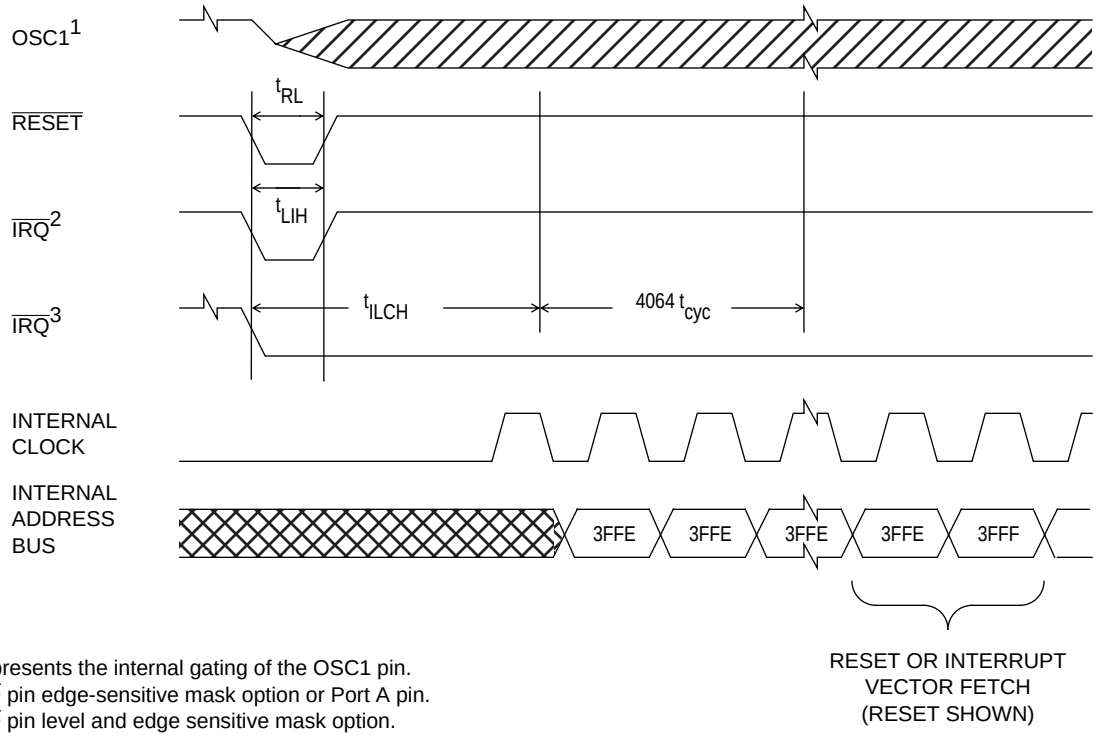
The STOP instruction places the MCU in its lowest power consumption mode. In STOP mode, the internal oscillator is turned off, halting all internal processing, including timer (and COP watchdog timer) operation.

During STOP mode, the core timer interrupt flags and interrupt enable bits of the CTCSR register are cleared by internal hardware to remove any pending timer interrupt requests and to disable any further timer interrupts. The timer pre-scaler is also cleared. The I bit in the CCR is cleared to enable external interrupts. All other registers, including the remaining bits in the CTCSR, 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 $\overline{\text{RESET}}$.

The STOP instruction can be disabled by a mask option. When disabled, the STOP instruction causes a system reset.

6.5.2 STOP Recovery

The processor can be brought out of the STOP mode only by an external interrupt, an environmental exception interrupt or $\overline{\text{RESET}}$. See **Figure 6-1**.



Notes:
1. Represents the internal gating of the OSC1 pin.
2. $\overline{\text{IRQ}}$ pin edge-sensitive mask option or Port A pin.
3. $\overline{\text{IRQ}}$ pin level and edge sensitive mask option.

Figure 6-1 Stop Recovery Timing Diagram

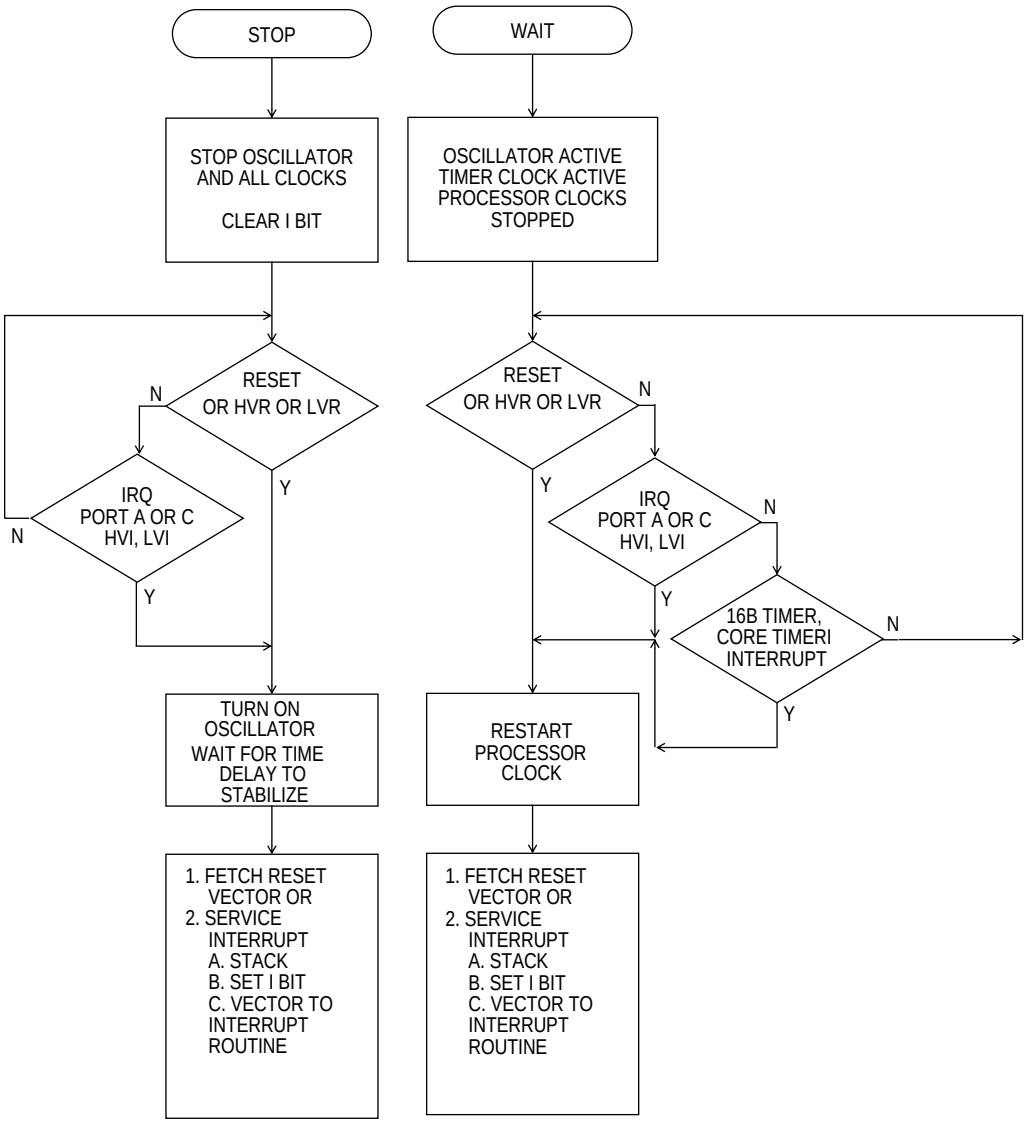


Figure 6-2 STOP and WAIT Flowcharts

6.6 WAIT Mode

The WAIT instruction places the MCU in a low-power consumption mode. All CPU action is suspended, but the core timer, the 16-bit timer (controlled by TOFF bit) and the PWM will or can remain active. An interrupt, if enabled, from the core timer or any peripheral still active in WAIT mode causes the MCU to exit WAIT mode.

During 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 core timer may be enabled to allow a periodic exit from the WAIT mode.

WAIT mode consumes more power than STOP mode.

Section 7. Input/Output Ports

7.1 Contents

7.2 Introduction

In single chip mode there are 20 lines arranged as one 8-bit I/O port (port A), one 5-bit I/O port (port B), and one 7-bit high-voltage I/O port (port C). The I/O ports are programmable as either inputs or outputs under software control of the data direction registers (see **7.3 General Input/Output Programming**).

Port A is shared with A/D channels. Ports B and C are shared with timer and PWM channels. Port C comprises 5 lines with current sensors and 2 lines with relay drivers.

7.3 General Input/Output Programming

Bidirectional port lines may be programmed as an input or an 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 I/O port pin is configured as an output if its corresponding DDR bit is set to a logic one. A pin is configured as an input if its corresponding DDR bit is cleared to a logical zero (see **Table 7-1** and **Figure 7-1**).

At power-on or reset, all DDRs are cleared, thus configuring all port 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.

Table 7-1 I/O Pin Functions

R/W[1]	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 to the I/O pin.
1	0	The state of the I/O pin is read.
1	1	The I/O pin is in output mode. The output data latch is read.

1. R/W is an internal signal

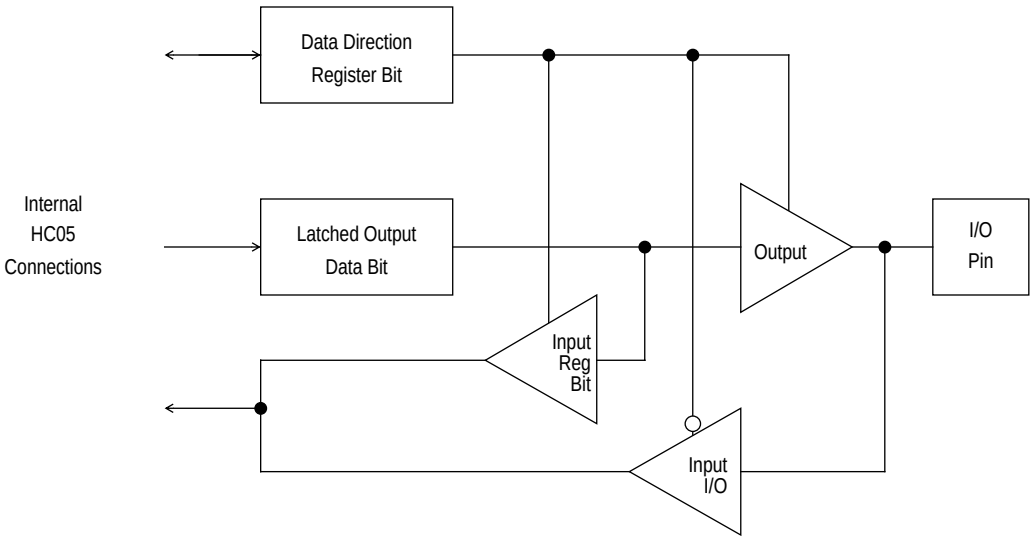


Figure 7-1 Port I/O Circuitry

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

NOTE: *If the I/O pin is an input and a read-modify-write (RMW) instruction is executed, the I/O pin will be read into the HC05 CPU and the computed result will then be written to the data latch.*

7.4 Port A

Port A is an 8-bit bidirectional port (PA0–7) with interrupt capability, shared with the A/D converter (AN1–6, VREFL, VREFH).

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

When the A/D converter is turned on, one of the channels AN1–6 may be selected through the A/D status and control register for conversion.

The input lines of port A include software programmable pull-up resistors.

7.4.1 Port A Keyboard Interrupt

The keyboard interrupt consists of 8 individual edge-sensitive interrupts with 8 interrupt flags. The keyboard interrupt is generated by a logical OR function of the 8 interrupt flags. The interrupt inputs are connected to PA0–7. All interrupts are maskable. If the interrupt mask bit (I bit) in the condition code register is set, all interrupts are disabled.

The interrupts are split in two groups of four lines each (PA0–3 and PA4–7). All interrupts of one group can be simultaneously masked by the corresponding PAIE bits in the port A configuration register. The trigger edges of the interrupt lines are selectable for each group with the EDGE bits in the port A configuration register.

The port A interrupt status register indicates which interrupt request is pending.

7.4.2 Port A Pull-up Resistors

The PA0–7 input lines have internal pull-up resistors. The port A lines form two groups with four lines each (PA0–3 and PA4–7). All pull-up´s of one group can be switched on with the PULEN or PUHEN bits of the port A configuration register by resetting the bit to 0. They are disabled

- when the enable bit is set to 1
- when a line is configured as output.

The pull-up resistor should be disabled when a line is used as A/D input or A/D reference channel.

7.4.3 Port A Voltage Reference for A/D Converter

The lines PA0 and PA7 can be connected to the reference inputs for the A/D converter (VREFL and VREFH). In order to connect the reference inputs, the corresponding VRHEN or VRLEN bits of the port A configuration register have to be set. In addition, the corresponding lines (PA0 or PA7) must be configured as inputs.

7.4.4 Port A Configuration Register

$0020	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	VRHEN	PUHEN	EDGEH	PAHIE	PULEN	EDGEL	PALIE	VRLEN
Reset:	0	0	0	0	0	0	0	0

Figure 7-2 Port A Configuration Register (PACFG)

VRHEN – Enable A/D High Reference Channel

Those bits connect the PA7 pin with the A/D high reference channel.

1 = A/D high reference channel connected to external VREFH.

0 = A/D high reference channel connected to internal voltage supply.

PUHEN – PA4–7 Pull-Up Resistor Enable Higher Nibble

This bit disables/enables the pull-up resistors of the PA4–7 pins.

1 = PA4–7 pull-up resistors disabled

0 = PA4–7 pull-up resistors enabled

EDGEH – PA4–7 Interrupt Edge Higher Nibble

This bit selects the trigger edges of the interrupt lines PA4–7.

1 = Rising edge sensitive

0 = Falling edge sensitive

PAHIE – PA4–7 Interrupt Enable Higher Nibble

This bit disables/enables the PA4–7 pins as an interrupt group.

1 = PA4–7 interrupt enabled

0 = PA4–7 interrupt disabled

PULEN – PA0–3 Pull-Up Resistor Enable Lower Nibble

This bits disables/enables the pull-up resistors of the PA0–3 pins.

1 = PA0–3 pull-up resistors disabled

0 = PA0–3 pull-up resistors enabled

EDGEL – PA0–3 Interrupt Edge Lower Nibble

This bit selects the trigger edges of the interrupt lines PA0–3.

1 = Rising edge sensitive

0 = Falling edge sensitive

PALIE – PA0–3 Interrupt Enable Lower Nibble

This bit disables/enables the PA0–3 pins as interrupt group.

1 = PA0–3 interrupt enabled

0 = PA0–3 interrupt disabled

VRLEN – Enable A/D Low Reference Channel

This bit connects the PA0 pin with the A/D low reference channel.

1 = A/D low reference channel connected to external VREFL.

0 = A/D low reference channel connected to internal ground.

7.4.5 Port A Interrupt Status Register

$0024	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	PAIF7	PAIF6	PAIF5	PAIF4	PAIF3	PAIF2	PAIF1	PAIF0
Reset:	0	0	0	0	0	0	0	0

Figure 7-3 Port A Interrupt Status Register (PAISR)

PAIF0–7 – Port A Interrupt Flags

These flags indicate which of the port A interrupt requests is pending. The 8 interrupt flags can be reset individually if a 1 is written to the bit position.

1 = Flag set when corresponding transition is sensed (if interrupt enabled). Writing a 1 clears the flag

0 = No interrupt. Writing a 0 has no effect

7.4.6 Operational Amplifier

Pins PA4–6 are connected to an operational amplifier. The operational amplifier is intended for amplifying small signals over VSS to increase the resolution of the A/D converter. The output stage of this operational amplifier is asymmetrical and thus optimized for driving loads to VSS while keeping the quiescent current low. The output of the operational amplifier is connected to channel 4 of the A/D converter. The amplifier is enabled by the I/O configuration register Bit6. As long as IOCFG Bit6 is 0, the presence of the operational amplifier is without any effect. If the opamp is enabled, first ensure that the PA4 is switched to input mode.

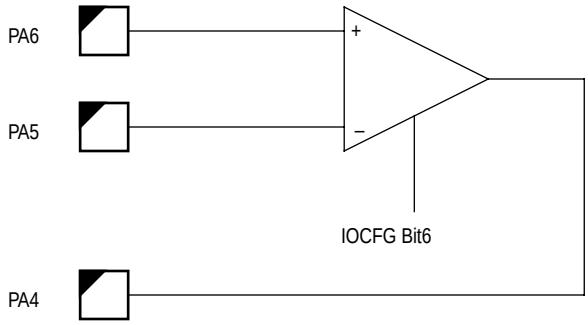


Figure 7-4 Operational Amplifier

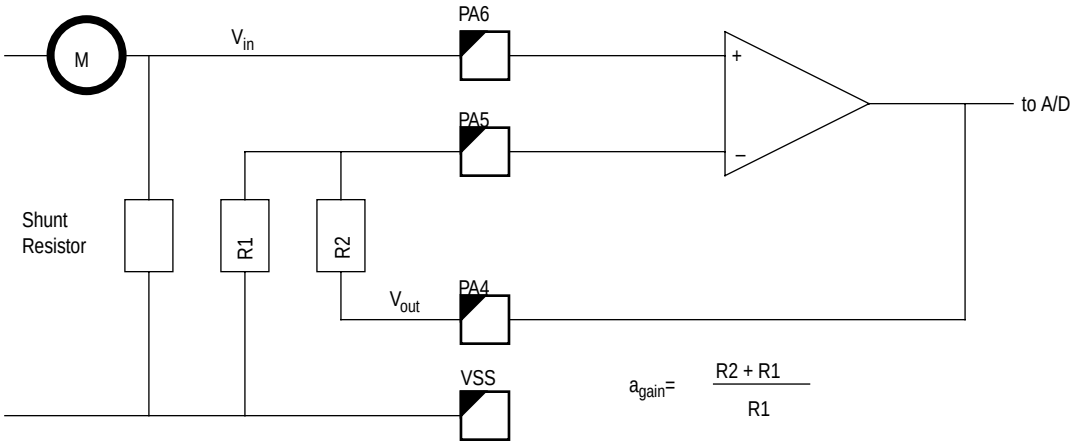


Figure 7-5 Typical application: positive Vgain amplifier

- Keep V_{in} limited between V_{SS} and V_{DD}
- For precise measurements, R1 + R2 should be in the range of 50kΩ and the V_{out} should not reach V_{DD}
- External loads should be connected to ground, due to small current sinking capability.
- In case of V_{in} x α_{gain} >= V_{DD} (i.e. the output of the operational amplifier cannot follow the input anymore) channel 6 (input) should be converted to read the input voltage V_{in} directly.

7.5 Port B

Port B is a 5-bit bidirectional port, shared with timer and PWM channels (TCAP, TCMP, PWM). A XOR function is provided for one timer capture channel.

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

7.5.1 Port B Timer Channels and XOR Function

The port pins PB0–PB3 are shared with the 16-bit timer channels (TCAP1–2, TCMP1–2). The timer capture channel TCAP1 can be driven by the XOR of two channels if TXOR bit in the I/O Configuration Register is set (see **Figure 7-6**).

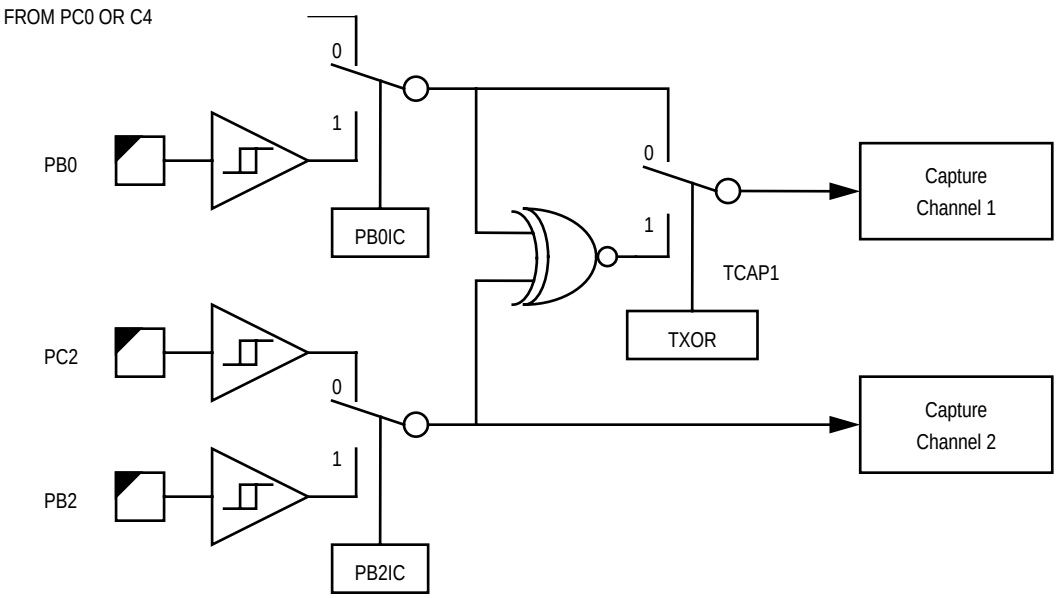


Figure 7-6 Mapping Ports to Timer Capture Channels

7.5.2 Port B PWM Channel

The port pin PB4 is shared with the PWM channel. In order to connect this pin to the PWM channel, the corresponding bit PWM4 of the I/O configuration register must be set.

7.5.3 I/O Configuration Register

$0021	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	TXOR	OPAMP		PB4PW	PB3OC	PB2IC	PB1OC	PB0IC
Reset:	0	0	0	0	0	0	0	0

Figure 7-7 I/O Configuration Register (IOCFG)

TXOR – Timer EXOR Enable

This bit enables the EXOR of the TCAP0 channel
1 = EXOR enabled
0 = EXOR disabled

OPAMP – Enable Operational Amplifier

This bit enables the operational amplifier on PA6
1 = Opamp enabled
0 = Opamp disabled

PB4PW – PB4 PWM Enable

This bit enables the PB4 pin as PWM output.
1 = PB4 PWM enabled. PBDD4 bit must be set in order to drive the output
0 = PB4 PWM disabled

PB3OC – PB3 Output Compare Enable

This bit enables the PB3 pin for output compare channel 2.
1 = PB3 output compare channel 2 enabled. PBDD3 bit must be set in order to drive the output
0 = PB3 output compare channel 2 disabled

PB2IC – PB2 Input Capture Enable

This bit enables the PB2 pin to drive the input capture channel 2.
1 = PB2 drives the input capture channel 2
0 = PC2 drives the input capture channel 2

PB1OC – PB1 Output Compare Enable

This bit enables the PB1 pin for output compare channel 1.
1 = PB1 output compare channel 1 enabled. PBDD1 bit must be set in order to drive the output
0 = PB1 output compare channel 1disabled

PB0IC – PB0 Input Capture Enable

This bit enables the PB0 pin to drive the input capture channel 1.
1 = PB0 drives the input capture channel 1
0 = PC0 or PC4 drives the input capture channel 1

7.6 Port C (High Voltage Port)

Port C is a 7-bit multifunctional and bidirectional port (PC0–6) with high voltage capability. The port is shared with timer and PWM channels (TCAP, TCMP, PWM) and provides a special current sense feature with interrupt capability.

In addition, port C comprises a low ohmic two channel relay driver with internal Zener diode turn-off.

The port C data register is at $0002 and the data direction register (DDR) is at $0006. Reset does not affect the data registers, but clears the data direction registers, thereby returning the PC0–4 to inputs, PC5 and PC6 are switched to the off state. Writing a one to a DDR bit sets the corresponding port bit to output mode.

The port C pins PC5–6 are open drain outputs only without internal pull-ups.

The voltage levels of PC0–4 I/O signals are related to the V_{SUP} and V_{SS} levels respectively. PC5–6 have an additional power supply pin for VSS (PVSS) to which the relay drivers relate.

7.6.1 Port C Timer Channels

The port pins PC0–5 are shared with the 16-bit timer channels (TCAP1–2, TCMP1–2).

7.6.2 Port C PWM Channel

The port pins PC0, 4–6 are shared with the PWM channel. In order to connect those pins, please refer to **7.6.6 Port C Configuration Register 0** for details.

7.6.3 Port C Current Sense Circuitry

The port C pins PC0–4 have a special current sense circuit (see **Figure 7-8**, **Figure 7-9**, **Figure 7-10**). This feature allows, for example, the monitoring of mechanical contacts in automotive applications (switch monitor).

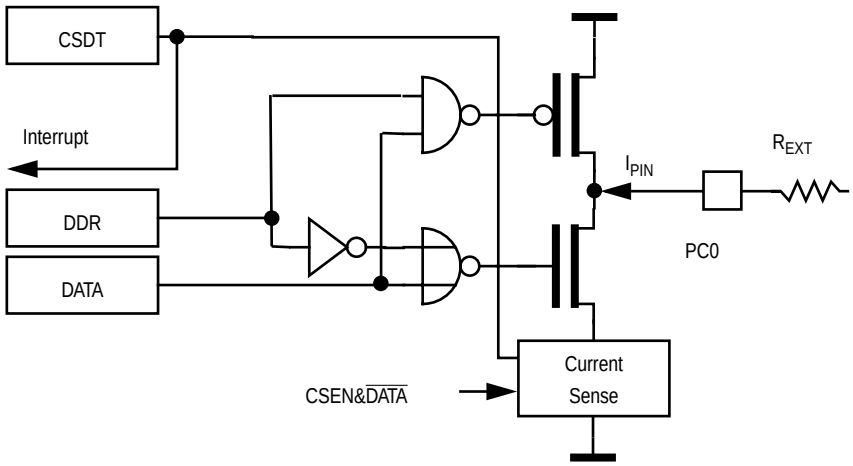


Figure 7-8 PC0 Current Sense Circuitry

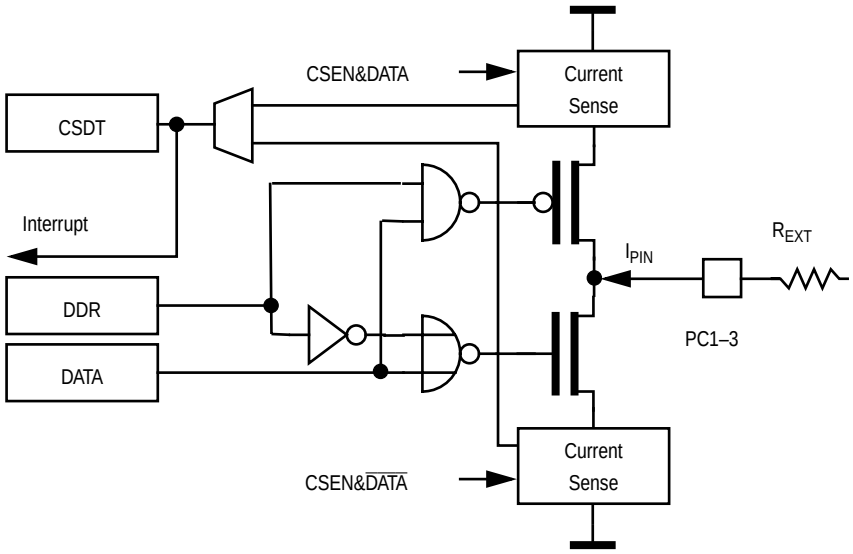


Figure 7-9 PC1–3 Current Sense Circuitry

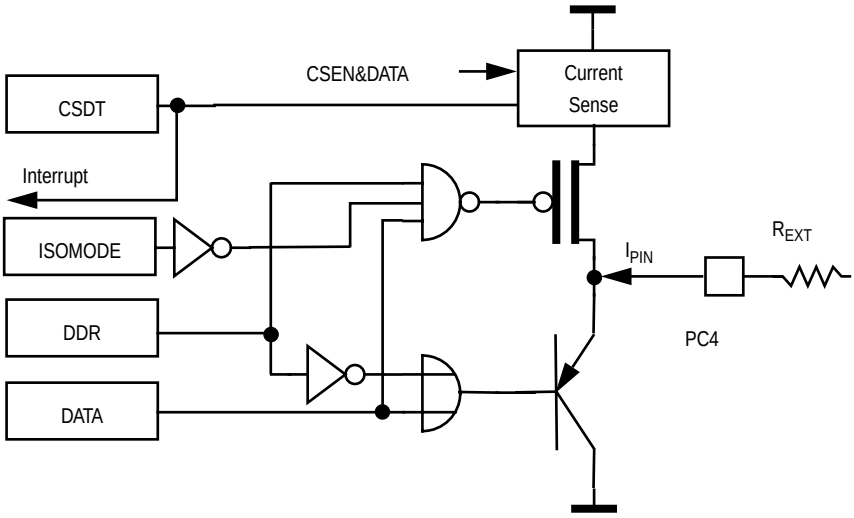


Figure 7-10 .PC4 Current Sense Circuitry

Port pin PC0 comprises a circuit that senses the pin current I_{PIN} to VSUP. PC4 has a different circuit, which senses the pin current I_{PIN} to VSS. PC1, PC2 and PC3 have a universal one, which senses the current either to VSS or to VSUP, depending on the state of the corresponding data register bit.

The current sense circuitry is enabled by setting the corresponding bits PC4CS, PC3CS, PC2CS, PC1CS or PC0CS of the port C configuration register to 1. In addition, the pin has to be configured as an output by setting the corresponding DDR bit to 1 and the data bit to 0 (for I_{PIN} to VSUP, e.g. external switch to VSUP) or to 1 (for I_{PIN} to VSS, e.g. external switch to VSS). If the pin current I_{PIN} exceeds a specified value, the current sense circuitry interprets this as a logical 1. The principal sense characteristic is given in **Figure 7-11**. The result of this sense operation is given by the bits CSD4, CSD3, CSD2, CSD1 and CSD0 of the port C status register.

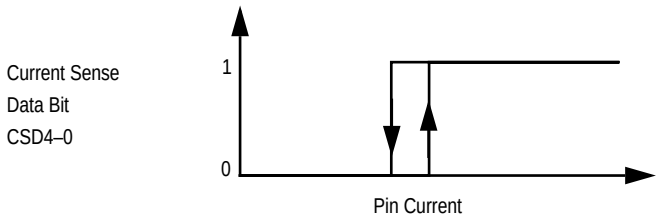


Figure 7-11 Principal Characteristic of the Current Sense Circuitry

A current sense interrupt is generated if the status of any sense circuitry changes. The interrupt trigger occurs on both edges of the status change and sets the corresponding CSIF flag in the port C status register.

The interrupt can be masked by the CSIE bit of the port C configuration register.

When setting PCXCS, but clearing the corresponding DDR bit, inverse input voltage can be read at the CSD bits, and those signals will be used to generate current sense interrupts, if enabled.

The maximum current of the pins PC0–4 has to be limited by an external resistor because

- the lines are low ohmic,
- internal substrate current injection may occur if the pin voltage is out of the supply voltage range.

7.6.4 Port C ISO9141 Interface

To use Port C4 as an ISO9141 physical interface, Port C4 should be always programmed as an output. This automatically enables the biasing circuit for the ISO9141 driver. Furthermore, the ISOMODE bit in the Port C Configuration Register 0 has to be set. This driver encorporates an overcurrent limitation circuit. Because of excessive power dissipation the software should take care to switch off the driver as soon as possible whenever a short-circuit occurs. To detect such a condition the PC4CL (Bit 6) in the Interrupt Status Register should be polled.

$0029	Bit 7	6	5	4	3	2	1	Bit 0
Read:	RCON	PC4CL	0	0	0	HTIF	HVIF	LVIF
Write:								
Reset:	?	0	0	0	0	?	0	0

Figure 7-12 Interrupt Status Register (INTSR)

PC4CL - Port C4 in current limit mode
1 = current on PC4 exceeds limit
0 = current on PC4 below limit

If the timer input capture 1 is configured to Port C4, the state of the PC4 pin is transfered to the timer module input capture.

7.6.5 Port C Relay Driver

The port C pins PC5–6 comprise of two relay driver channels which are shared with the PWM function. The channels can either be controlled directly by the data register or are linked to the PWM function (see **7.6.2 Port C PWM Channel**).

The relay driver channels are open-drain outputs with an internal Zener diode. The diode clamps the maximum output voltage and limits the turn-off time of inductive loads (see **Figure 7-13**).

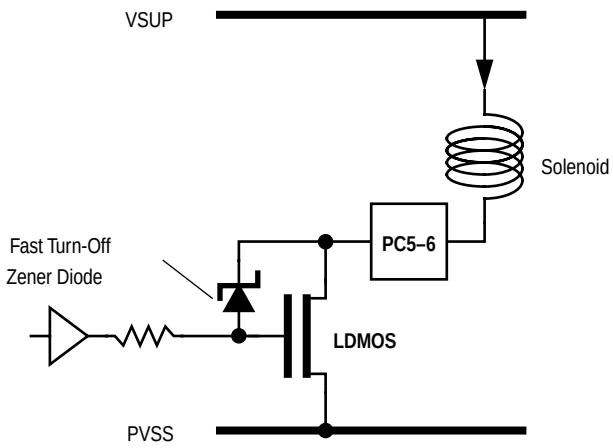


Figure 7-13 Principle of Port C Relay Driver

A permanent external pin voltage above the minimum Zener break-down can destroy the driver.

The relay drivers have a short circuit protection feature. Whenever the drain current of the LDMOS transistor rises above a fixed value, the output is automatically switched off (i.e. the LDMOS is in the high impedance state) and the corresponding short circuit flag is set (SCIF5 or SCIF6). If the SCIE5/6 bits are enabled, an interrupt occurs. As long as the SCIF5/6 bits are set, the output cannot be switched on. These bits are cleared by writing a logical 1 to the corresponding bit location. The outputs are also protected by a common over temperature detection. See **Figure 7-14** for details.

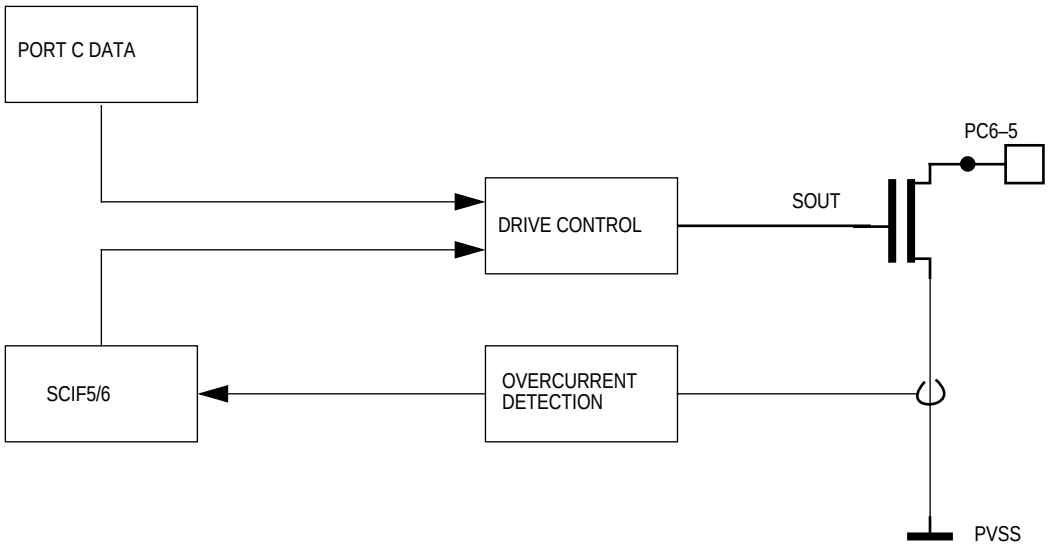


Figure 7-14 Short Circuit Diagnostic of Port C Relay Driver

7.6.6 Port C Configuration Register 0

$0022	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	ISOMODE	PC6PW	PWMS1	PWMS0	PC3OC	TS2	TS1	TS0
Reset:	0	0	0	0	0	0	0	0

Figure 7-15 Port C Configuration Register 0 (PCCFG0)

ISOMODE – Driver Mode of PC4

This bit selects the driver mode of PC4.

1 = ISO9141 compatible output (low side driver only)

0 = PC4 is a push-pull output

PC6PW – PC6 PWM Enable

This bit enables the PC6 pin as PWM output.

1 = PC6 PWM enabled.

0 = PC6 PWM disabled

PWMS1, PWMS0 – PWM Select Bits

These bits select the output pin for the PWM on PC0, PC4 or PC5.

Table 7-2 PWM Select

PWMS1	PWMS0	PWM Output at Port C
0	0	none
0	1	PC0
1	0	PC4
1	1	PC5

PC3OC – PC3 Output Compare Enable

This bit enables the PC3 pin for output compare channel 2.

1 = PC3 output compare channel 2 enabled. PC3 DDR bit must be set in order to drive the output

0 = PC3 output compare channel 2 disabled

TS2, TS1, TS0 – Timer Channel 1 Select Bits

These bits select the input and output pins for the timer channel 1.

Table 7-3 Timer Channel 1 Select

TS2	TS1	TS0	Output Compare at PC	Input Capture at PC
0	0	0	none, Bit I/O	PC0
0	0	1	none, Bit I/O	PC4
0	1	0	PC0	PC0
0	1	1	PC0	PC4
1	0	0	PC1	PC0
1	0	1	PC1	PC4
1	1	0	PC4	PC4
1	1	1	PC5	PC4

NOTE: *If PC0, PC1, PC4 and PC5 are neither switched to PWM nor to timer output compare, the output states of these pins follow the states of their data register bits.*

To enable either PWM or output compare function the corresponding DDR bit must be set to 1.

If PWM and timer output compare functions are routed to the same pin, PC0 and PC4 would be connected to the output compare signal, PC5 would be connected to the PWM signal.

For using the input capture be sure that the PB0IC bit in the I/O configuration register is set to 0, and the corresponding pin PC0 or PC4 is switched to input mode. PC4 may also be in the ISO9141 compatible mode.

For using the current sense function, it is not recommended to route any special signal to the corresponding pins.

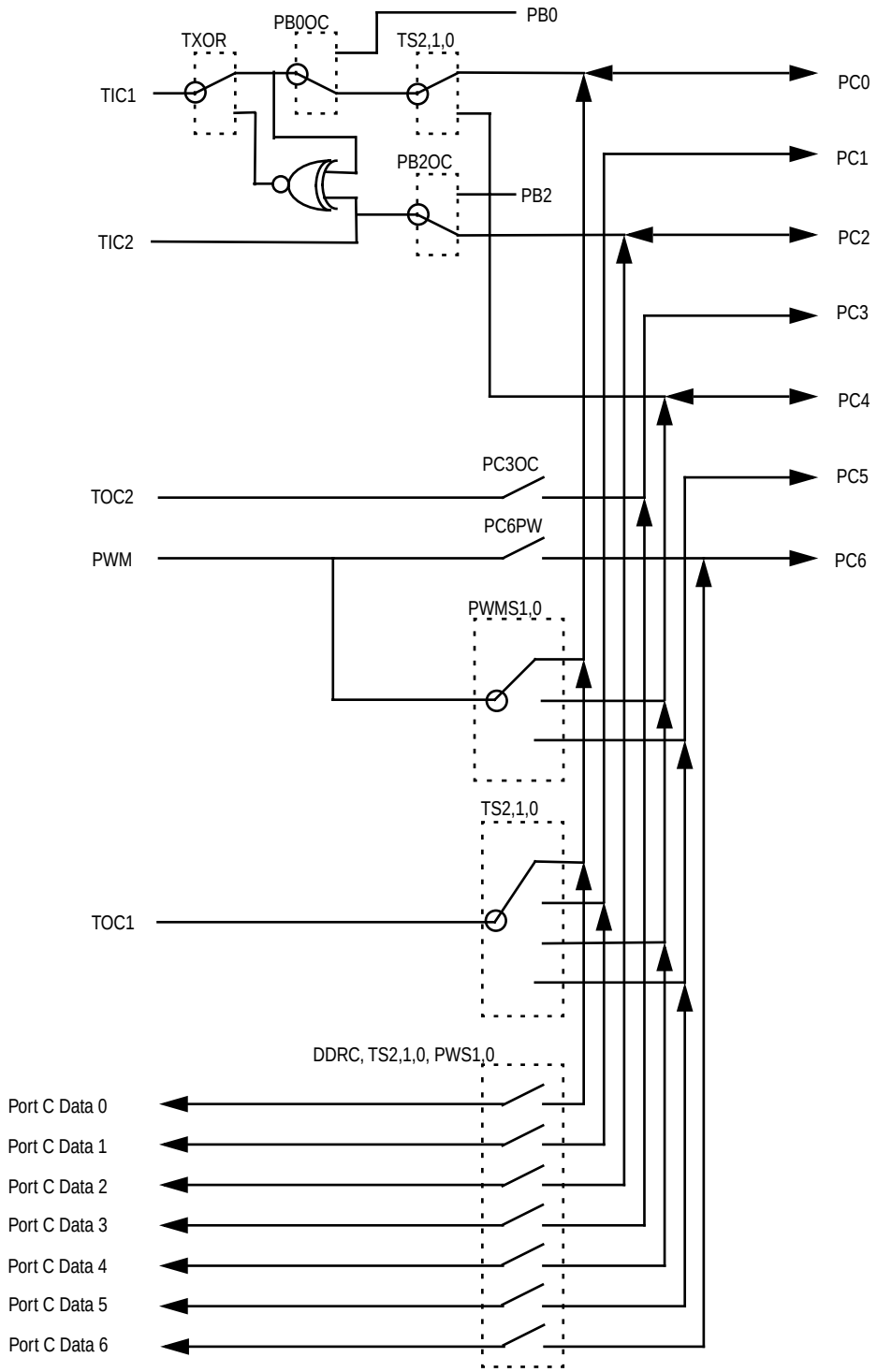


Figure 7-16 Port C Special Signal Routing

7.6.7 Port C Configuration Register 1

$0026	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	CSIE	SCIE6	SCIE5	PC4CS	PC3CS	PC2CS	PC1CS	PC0CS
Reset:	0	0	0	0	0	0	0	0

Figure 7-17 Port C Configuration Register 1 (PCCFG1)

CSIE – Port C Current Sense Interrupt Enable

This bit enables current sense interrupt of the lines PC4–0.

1 = Port C current sense interrupt enabled
0 = Port C current sense interrupt disabled

SCIE6 – Relay Driver Short Circuit Interrupt Enable

This bit enables short circuit interrupt of the relay driver PC6.

1 = Relay driver short circuit interrupt enabled
0 = Relay driver short circuit interrupt disabled

SCIE5 – Relay Driver Short Circuit Interrupt Enable

This bit enables short circuit interrupt of the relay driver PC5.

1 = Relay driver short circuit interrupt enabled
0 = Relay driver short circuit interrupt disabled

PC4CS – PC4 Current Sense Enable

This bit enables the PC4 current sense circuitry.

1 = PC4 current sense circuitry enabled
0 = PC4 current sense circuitry disabled

PC3CS – PC3 Current Sense Enable

This bit enables the PC3 current sense circuitry.

1 = PC3 current sense circuitry enabled
0 = PC3 current sense circuitry disabled

PC2CS – PC2 Current Sense Enable

This bit enables the PC2 current sense circuitry.

1 = PC2 current sense circuitry enabled
0 = PC2 current sense circuitry disabled

PC1CS – PC1 Current Sense Enable

This bit enables the PC1 current sense circuitry.
1 = PC1 current sense circuitry enabled
0 = PC1 current sense circuitry disabled

PC0CS – PC0 Current Sense Enable

This bit enables the PC0 current sense circuitry.
1 = PC0 current sense circuitry enabled
0 = PC0 current sense circuitry disabled

7.6.8 Port C Status Register

$0027	Bit 7	6	5	4	3	2	1	Bit 0
Read:	CSIF	SCIF6	SCIF5	CSD4	CSD3	CSD2	CSD1	CSD0
Write:								
Reset:	0	0	0	0	0	0	0	0

Figure 7-18 Port C Status Register (PCSTR)

CSIF – Port C Current Sense Interrupt Flag

This flag indicates that a current sense transition has occurred and an interrupt request is pending. The flag can be cleared by writing a 1 to it.
1 = Flag set when a transition is sensed by the current sense circuitry
0 = No interrupt

SCIF6 – Relay Driver Short Circuit Interrupt Flag

This flag indicates a short circuit on PC6 is active and an interrupt request is pending.
1 = Short circuit at the PC6 pin; PC6 is switched to high impedance
0 = No short circuit at the PC6 pin

SCIF5 – Relay Driver Short Circuit Interrupt Flag

This flag indicates a short circuit on PC5 is active and an interrupt request is pending.

1 = Short circuit at the PC5 pin; PC5 is switched to high impedance
0 = No short circuit at the PC5 pin

CSD4 – PC4 Current Sense Data

This data bit represents the result of the PC4 current sense circuitry.

1 = High current flow sensed (see **Figure 7-11**), or input PC4 is 0.
0 = Low current flow sensed

CSD3 – PC3 Current Sense Data

This data bit represents the result of the PC3 current sense circuitry.

1 = High current flow sensed (see **Figure 7-11**), or input PC3 is 0.
0 = Low current flow sensed

CSD2 – PC2 Current Sense Data

This data bit represents the result of the PC2 current sense circuitry.

1 = High current flow sensed (see **Figure 7-11**), or input PC2 is 0.
0 = Low current flow sensed

CSD1 – PC1 Current Sense Data

This data bit represents the result of the PC1 current sense circuitry.

1 = High current flow sensed (see **Figure 7-11**), or input PC1 is 0.
0 = Low current flow sensed

CSD0 – PC0 Current Sense Data

This data bit represents the result of the PC0 current sense circuitry.

1 = High current flow sensed (see **Figure 7-11**), or input PC0 is 0.
0 = Low current flow sensed

7.6.9 MFTEST Register

$002F	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	HVTOFF			VSCAL	LSOFF	VT2	VT1	VT0
Reset:	0	0	0	0	0	0	0	0

Figure 7-19 MFTEST Register (MFTEST)

HVTOFF - Disable of Port C Inputs

This data bit controls the operation of the Port C Inputs

1 = Port C Inputs (PC0 - PC4) disabled

0 = Port C Inputs enabled

VSCAL – Disable of V_{SUP} Scaler Circuit

This data bit controls the operation of the V_{SUP} scaler circuit

1 = V_{SUP} scaler disabled, this mode saves power consumption

0 = V_{SUP} scaler enabled, V_{SUP} can be measured by the A/D converter

LSOFF – Low Side Drivers Off

This data bit controls the operation of PC5-6 and the temperature sensor block

1 = PC5-6 and temperature block disabled to minimize power consumption

0 = PC5-6 and and temperature block enabled

VT2, VT1, VT0 – Voltage Regulator Trimming Bits

Refer to **12.5 Trimming the Voltage Regulator**.

Section 8. Core Timer

8.1 Contents

8.2 Introduction

The core timer for this device is a 15-stage multi-functional ripple counter. The features include timer over flow, power-on reset (POR), real time interrupt (RTI), and COP watchdog timer.

As seen in **Figure 8-1**, the timer is driven by the output of the clock select circuit followed by a fixed divide by four pre-scaler. This signal drives an 8-bit ripple counter. The value of this 8-bit ripple counter can be read by the CPU at any time by accessing the timer counter register (TCR) at address $09. A timer overflow function is implemented on the last stage of this counter, giving a possible interrupt at the rate of f_{op}/1024. Two additional stages produce the POR function after 4064 clks (if selected). The timer counter bypass circuitry (available only in test mode) is at this point in the timer chain. This circuit is followed by two more stages, with the resulting clock (f_{op}/16384) driving the real time interrupt circuit. The RTI circuit consists of three divider stages with a 1 of 4 selector. The output of the RTI circuit is further divided by eight to drive the mask optional COP watchdog timer circuit. The RTI rate selector bits, and the

RTI and TOF enable bits and flags are located in the timer status and control register at location $08.

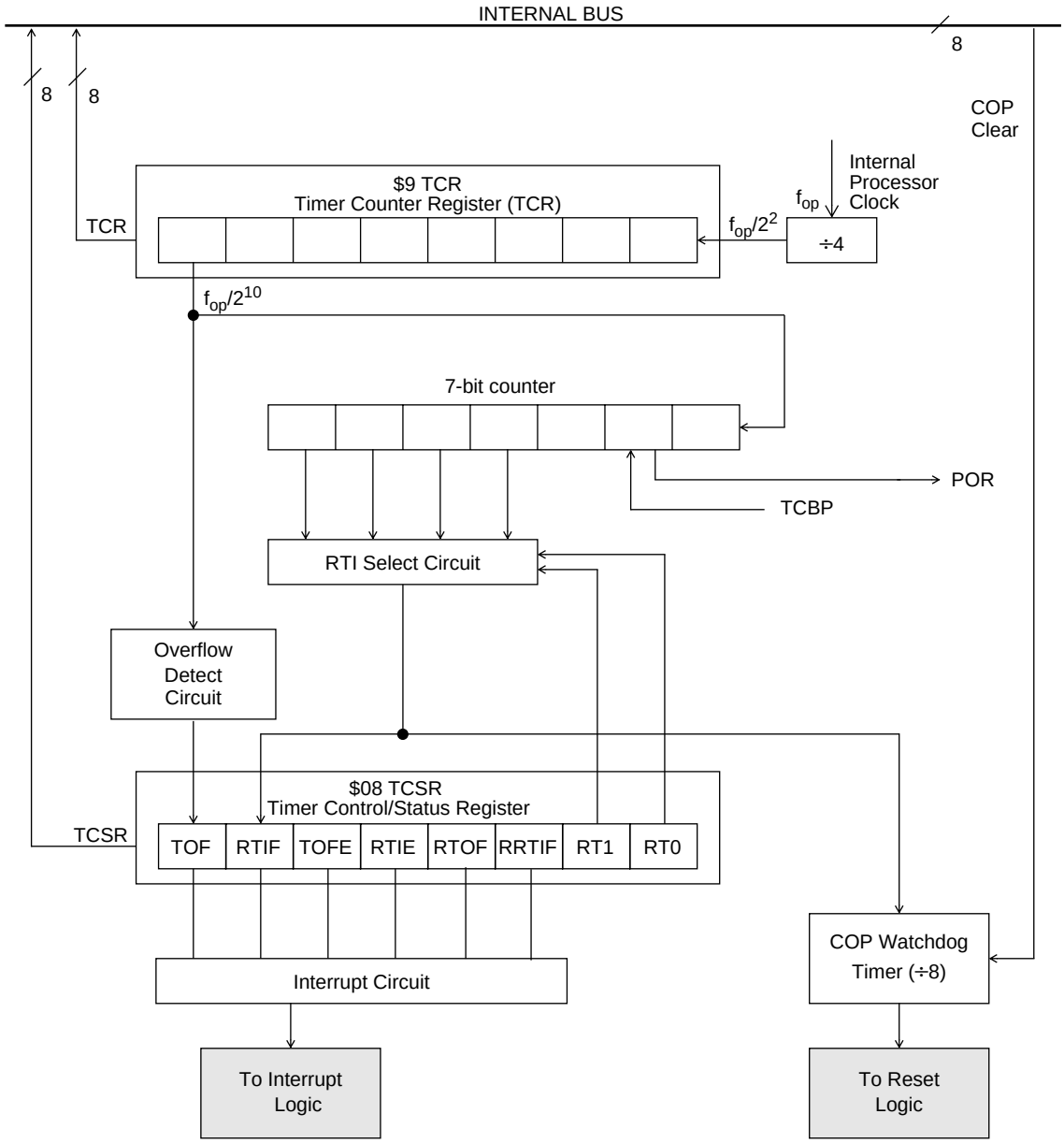


Figure 8-1 Core Timer Block Diagram

8.3 Registers

8.3.1 Core Timer Status & Control Register (CTSCR)

The CTSCR contains the timer interrupt flag, the timer interrupt enable bits, and the real time interrupt rate select bits. **Figure 8-2** shows the value of each bit in the CTSCR when coming out of reset.

$0008	Bit 7	6	5	4	3	2	1	Bit 0
Read:	TOF	RTIF	TOFE	RTIE	0	0	RT1	RT0
Write:					RTOF	RRTIF		
Reset:	0	0	0	0	0	0	1	1

Figure 8-2 Core Timer Status and Control Register (CTSCR)

TOF – Timer Over Flow

TOF is a read-only status bit and is set when the 8-bit ripple counter rolls over from $FF to $00. A CPU interrupt request will be generated if TOFE is set. Reset clears TOF.

RTIF – Real Time Interrupt Flag

The real time interrupt circuit consists of a three stage divider and a 1 of 4 selector. The clock frequency that drives the RTI circuit is $f_{op}/2^{13}$ (or $f_{op}/8192$) with three additional divider stages giving a maximum interrupt period of about 250ms at a crystal frequency of 1 MHz. RTIF is a read-only status bit and is set when the output of the chosen (1 of 4 selection) stage goes active. A CPU interrupt request will be generated if RTIE is set. Reset clears RTIF.

TOFE – Timer Over Flow Enable

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

RTIE – Real Time Interrupt Enable

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

RTOF – Reset TOF

This bit always reads 0. Setting this bit clears the timer overflow flag (TOF). Clearing this bit has no effect.

RRTIF – Reset RTIF

This bit always reads 0. Setting this bit clears the real time interrupt flag (RTIF). Clearing this bit has no effect.

RT1, RT0 – Real Time Interrupt Rate Select

These two bits select one of four taps from the real time interrupt circuit. **Table 8-1** shows the available interrupt rates with several f_{op} values. Reset sets these RT0 and RT1, selecting the lowest periodic rate and therefore the maximum time in which to alter these bits if necessary. Care should be taken when altering RT0 and RT1 if the time-out period is imminent or uncertain. If the selected tap is modified during a cycle in which the counter is switching an RTIF could be missed or an additional one could be generated. To avoid problems the COP should be cleared before changing RTI taps.

Table 8-1 RTI Rates

	RTI Rates at Bus Frequency f_{OP} specified:				
RT1:RT0	**500 kHz**	**1.000 MHz**	**2.000 MHz**	**2.4576 MHz**	**RATIO**
00	32.768ms	16.384ms	8.192ms	6.667ms	$2^{14}/f_{op}$
01	65.536ms	32.768ms	16.384ms	13.333ms	$2^{15}/f_{op}$
10	131.072ms	65.536ms	32.768ms	26.667ms	$2^{16}/f_{op}$
11	262.144ms	131.072ms	65.536ms	53.333ms	$2^{17}/f_{op}$

8.3.2 Computer Operating Properly (COP) Watchdog Reset

The COP watchdog timer function is implemented on this device by using the output of the RTI circuit and further dividing it by eight. The minimum COP reset rates are listed in **Table 8-2**. If the COP circuit times out, an internal reset is generated and the normal reset vector is fetched. A COP time-out is prevented by clearing bit 0 of address $3FF0. When the COP is cleared, only the final divide by eight stage (output of the RTI) is cleared.

Table 8-2 Minimum COP Reset Times

	Minimum COP Reset Bus Frequency at f_{OP} specified:				
RT1:RT0	**500 kHz**	**1.000 MHz**	**2.000 MHz**	**2.4576 MHz**	**RATIO**
00	229.376ms	114.689ms	57.344ms	46.666ms	$7*2^{14}/f_{op}$
01	458.752ms	229.376ms	114.689ms	93.333ms	$7*2^{15}/f_{op}$
10	917.504ms	458.752ms	229.376ms	186.666ms	$7*2^{16}/f_{op}$
11	1835.000ms	917.504ms	458.752ms	373.333ms	$7*2^{17}/f_{op}$

8.3.3 Core Timer Counter Register (CTCR)

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

$0009	Bit 7	6	5	4	3	2	1	Bit 0
Read:	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
Write:								
Reset:	0	0	0	0	0	0	0	0

Figure 8-3 Core Timer Counter Register (CTCR)

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

8.4 Core Timer During WAIT

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

8.5 Core Timer During STOP

The timer and the interrupt mask and enable flags are cleared when going into STOP mode. When STOP is exited by an external interrupt or an external reset the internal oscillator will restart, followed by an internal processor stabilization delay (t_{PORL}). The timer is then cleared and the operation resumes.

Section 9. 16-Bit Programmable Timer

9.1 Contents

9.2 Introduction

The MC68HC(8)05PV8 has one 16-bit timer with two channels. The timer consists of a 16-bit free running counter driven by a fixed divide-by-four pre-scaler. 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. The output compare is improved so that it is now possible to link the two output compares to one output in order to generate pulses as short as E/4. Refer to **Figure 9-1** for a timer block diagram.

Because the timer has a 16-bit architecture each specific functional segment 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.

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

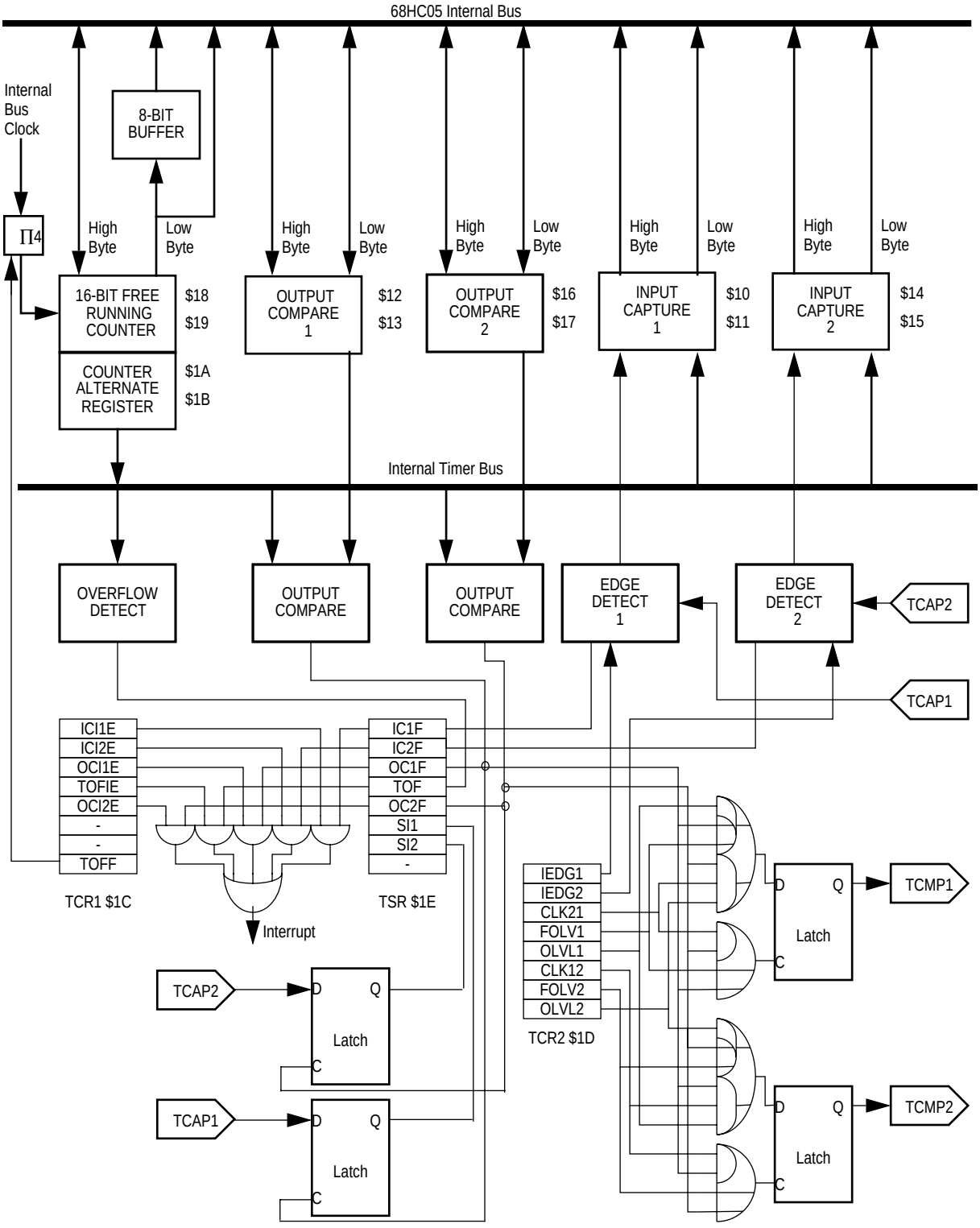


Figure 9-1 Timer Block Diagram

9.3 Registers

9.3.1 Counter

The key element in the programmable timer is a 16-bit free-running counter or counter register, preceded by a pre-scaler that divides the internal processor clock by four. The pre-scaler 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 ($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.

9.3.2 Output Compare Registers

There are two output compare registers: Output compare register 1 and output compare register 2. Output compare registers can be used for several purposes such as controlling an output waveform or indicating when a period of time has elapsed. All bits are readable and writeable and are not altered by the timer hardware or reset. If the compare function is not needed the two bytes of the output compare register can be used as storage locations.

9.3.2.1 Output Compare Register 1

The 16-bit output compare register 1 is made up of two 8-bit registers at locations $12 (MSB) and $13 (LSB). The output compare register contents are compared with the contents of the free-running counter once every four internal processor clock cycles. If a match is found, the output compare flag OC1F (bit 5 of the timer status register ($1E)) is set and the corresponding output level OLVL1 bit is clocked to TCMP1 output.

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 (OCI1E) is set.

After a processor write cycle to the output compare register 1 containing the MSB ($12), the output compare function is inhibited until the LSB ($13) is also written. The user must write both bytes (locations) if the MSB is written first. A write made only to the LSB ($13) will not inhibit the compare function. The free-running counter is updated every four internal bus clock cycles. The minimum time required to update the output compare register is a function of the program rather than the internal hardware.

The processor can write to either byte of the output compare register 1 without affecting the other byte. The output level (OLVL1) bit is clocked to the output level register regardless of whether the output compare flag (OC1F) is set or clear.

Because the output compare flag OC1F and the output compare register 1 are undetermined at power-on, and are not affected by external reset, care must be exercised when initializing the output compare function. The following procedure is recommended.

Write the high byte to the compare register 1 to inhibit further compares until the low byte is written.

Read the status register to arm the OC1F if it is already set.

Write the output compare register 1 low byte to enable the output compare 1 function with the flag clear.

The purpose of this procedure is to prevent the OC1F bit from being set between the time it is read and the write to the corresponding output compare register.

9.3.2.2 Output Compare Register 2

The 16-bit output compare register 2 is made up of two 8-bit registers at locations $16 (MSB) and $17 (LSB). The output compare register contents are compared with the contents of the free-running counter once every four internal processor clock cycles. If a match is found, the output compare flag OC2F (bit 3 of the timer status register ($1E)) is set and the corresponding output level OLVL2 bit is clocked to TCMP2 output.

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 (OCI2E) is set.

After a processor write cycle to the output compare register 2 containing the MSB ($16), the output compare function is inhibited until the LSB ($17) is also written. The user must write both bytes (locations) if the MSB is written first. A write made only to the LSB ($17) will not inhibit the compare function. The free-running counter is updated every four internal bus clock cycles. The minimum time required to update the output compare register is a function of the program rather than the internal hardware.

The processor can write to either byte of the output compare register 2 without affecting the other byte. The output level (OLVL2) bit is clocked to the output level register regardless of whether the output compare flag (OC2F) is set or clear.

Because the output compare flag OC2F and the output compare register 2 are undetermined at power-on, and are not affected by external reset, care must be exercised when initializing the output compare function. A procedure as recommended for compare register 1 should be followed.

9.3.3 Input Capture Registers

There are two identical input capture registers: Input capture register 1 and input capture register 2. The two following sections describe these two registers.

9.3.3.1 Input Capture Register 1

Two 8-bit registers, which make up the 16-bit input capture register 1, 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 on the TCAP1 pin. The level transition which triggers the counter transfer is defined by the corresponding input edge bit (IEDG1). Reset does not affect the contents of the input capture register except when exiting stop mode.

IEDG1 – Capture on Negative/Positive Edge
- 1 = Capture on positive edge
- 0 = Capture on negative edge

An interrupt can also accompany a capture provided the corresponding interrupt enable bit, ICI1E, is set.

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

The free-running counter contents are transferred to the input capture register on each proper signal transition regardless of whether the input capture flag (IC1F) is set or clear. The input capture register always contains the free-running counter value that corresponds to the most recent input capture.

After a read of the input capture register most significant byte ($10), the counter transfer is inhibited until the least significant byte ($11) 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 ($11) does not inhibit the free-running counter transfer since they occur on opposite edges of the internal bus clock.

9.3.3.2 Input Capture Register 2

Two 8-bit registers, which make up the 16-bit input capture register 2, 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 on the TCAP2 pin. The level transition which triggers the counter transfer is defined by the corresponding input edge bit (IEDG2). Reset does not affect the contents of the input capture register except when exiting stop mode.

IEDG2 – Capture on Negative/Positive Edge
- 1 = Capture on positive edge
- 0 = Capture on negative edge

An interrupt can also accompany a capture provided the corresponding interrupt enable bit, ICI2E, is set.

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

The free-running counter contents are transferred to the input capture register on each proper signal transition regardless of whether the input capture flag (IC2F) is set or clear. The input capture register always contains the free-running counter value that corresponds to the most recent input capture.

After a read of the input capture register most significant byte ($14), the counter transfer is inhibited until the least significant byte ($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.

9.3.4 Timer Control Register 1

$001C	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	ICI1E	ICI2E	OCI1E	TOIE	OCI2E			TOFF
Reset:	0	0	0	0	0	U	U	0

Figure 9-2 Timer Control Register 1 (TCR1)

ICI1E – Input Capture 1 Interrupt Enable
1 = Interrupt enabled
0 = Interrupt disabled

ICI2E – Input Capture 2 Interrupt Enable
1 = Interrupt enabled
0 = Interrupt disabled

OCI1E – Output Compare 1 Interrupt Enable
1 = Interrupt enabled
0 = Interrupt disabled

TOIE – Timer Overflow Interrupt Enable
1 = Interrupt enabled
0 = Interrupt disabled

OCI2E – Output Compare 2 Interrupt Enable

1 = Interrupt enabled
0 = Interrupt disabled

TOFF – Shut Off Timer

1 = Timer is disabled. This can be used to save power if timer is not used
0 = Timer is enabled

9.3.5 Timer Control Register 2

$001D	Bit 7	6	5	4	3	2	1	Bit 0
Read:	IEDG1	IEDG2	CLK21	0	OLVL1	CLK12	0	OLVL2
Write:				FOLV1			FOLV2	
Reset:	U	U	0	0	U	0	0	U

Figure 9-3 Timer Control Register 2 (TCR2)

IEDG1 – Input Edge

Value of input edge determines which level transition on TCAP1 pin will trigger free running counter transfer to the input capture register 1.

1 = Positive edge
0 = Negative edge

IEDG2 – Input Edge

Value of input edge determines which level transition on TCAP2 pin will trigger free running counter transfer to the input capture register 2.

1 = Positive edge
0 = Negative edge

CLK21 – Output Compare 2 clocks 1 output latch

If this bit is set to 1, a successful compare of compare register 2 loads the OLVL2 bit to the 1 output latch. This feature can be used to get output pulses as short as E/4 while using only one interrupt.

FOLV1 – Force Output Level 1

The FOLV1 bit always reads as zero. Writing a zero at this position has no effect. Writing a one at this position will force the OLVL1 bit to the corresponding output level latch, thus appearing at pin TCMP1. Note that the force output compare 1 does not affect the OCF1 bit of the status register.

OLVL1 – Output Level 1

Value of output level is clocked into output level register by the next successful output compare 1 and will appear on the TCMP1 pins.

1 = High output
0 = Low output

CLK12 – Output Compare 1 clocks 2 output latch

If this bit is set to 1, a successful compare of compare register 1 loads the OLVL1 bit to the 2 output latch. This feature can be used to get output pulses as short a E/4 while using only one interrupt.

FOLV2 – Force Output Level 2

The FOLV2 bit always reads as zero. Writing a zero at this position has no effect. Writing a one at this position will force the OLVL2 bit to the corresponding output level latch thus appearing at pin TCMP2. Note that the force output compare 2 does not affect the OCF2 bit of the status register.

OLVL2 – Output Level 2

Value of output level is clocked into output level register by the next successful output compare 2, and will appear on the TCMP2 pin.

1 = High output
0 = Low output

9.3.6 Timer Status Register

The timer status register is a read-only register containing timer status flags.

$001E	Bit 7	6	5	4	3	2	1	Bit 0
Read:	IC1F	IC2F	OC1F	TOF	OC2F	SI1	SI2	0
Write:								
Reset:	U	U	U	U	U	U	U	0

Figure 9-4 Timer Status Register 1 (TSR)

IC1F – Input Capture 1 Flag

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

0 = Flag cleared when TSR and input capture 1 registers low byte is accessed

IC2F – Input Capture 2 Flag

1 = Flag set when selected polarity edge is sensed by input capture 2 edge detector

0 = Flag cleared when TSR and input capture 2 registers low byte is accessed

OC1F – Output Compare 1 Flag

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

0 = Flag cleared when TSR and output compare register 1 low byte 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 are accessed

OC2F – Output Compare 2 Flag

1 = Flag set when output compare register 2 contents match the free-running counter contents

0 = Flag cleared when TSR and output compare register 2 low byte are accessed

SI1 – Sample Input 1

1 = Bit set when input capture 1 input is sampled high while output compare register 1 matches the free running counter

0 = Bit cleared when input capture 1 input is sampled low while output compare register 1 matches the free running counter

SI2 – Sample Input 2

1 = Bit set when input capture 2 input is sampled high while output compare register 2 matches the free running counter

0 = Bit cleared when input capture 2 input is sampled low while output compare register 2 matches the free running counter

Accessing the timer status registers satisfies the first condition required to clear status bits. The remaining step is to access the registers 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, and
2. The LSB of the free-running counter is read but not for the purpose of servicing the flag

The counter alternate register contains the same value as the free-running counter; therefore this alternate register can be read at any time without affecting the timer overflow flag in the timer status register.

9.4 Timer During WAIT Mode

The CPU clock halts during WAIT mode but the timer keeps on running. If any reset is used to exit WAIT mode the counters are forced to $FFFC. If interrupts are enabled a timer interrupt will cause the processor to exit WAIT mode.

9.5 Timer During STOP Mode

In STOP mode the timer stops counting and holds the last count value if STOP is exited by an interrupt. If any reset is used the counters are forced to $FFFC.

Note: During STOP, if at least one valid input capture edge occurs at the TCAP pins, the input capture detect circuit is armed. This does not set any timer flags nor 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 STOP mode. If any reset is used to exit STOP mode then no input capture flag or data remains even if a valid input capture edge occurred.

Section 10. Analog to Digital Converter

10.1 Contents

10.2 Introduction

The analog to digital converter system consists of a single 8-bit successive approximation converter and a channel multiplexer. There is one 8-bit result data register and one 8-bit status/control register.

The reference supply can be switched by software either to the internal VDD and VSS supplies or to external pins individually.

An internal RC type oscillator is activated by the ADRC bit in the A/D status and control register (ADSCR). This RC oscillator is used to provide a sufficiently high clock rate to the A/D when the bus speed is too low for the A/D to be accurate.

Additionally, the ADON bit allows the user to save power by disconnecting the A/D when not in use. This is particularly useful to reduce current consumption (typically by 100μA) when going into WAIT mode.

The A/D is ratiometric to the internal reference voltages VREFH and VREFL which can be derived from either VDD/VSS or external pins. An input voltage equal to or greater than VREFH converts to $FF (full scale) with no overflow indication (if greater). An input voltage equal to VREFL converts to $00. For ratiometric conversions, the source of each analog input should use VREFH as the supply voltage and be referenced to VREFL.

10.3 A/D Principle

The A/D reference inputs are applied to a precision internal digital to analog converter. Control logic drives this D/A and the analog output is successively compared to the selected analog input which was sampled at the beginning of the conversion time. The conversion is monotonic with no missing codes.

The 8-bit conversions are accurate to within ±1.5 LSB including quantization.

10.4 A/D Operation

The A/D is an 8-bit successive approximation register (SAR) type A/D converter with continuous conversion per given channel. The result of a conversion is loaded into the read-only result data register and a conversion complete flag COCO is set in the A/D status/control register.

Any write to the A/D status/control register will abort the current conversion, reset the conversion complete flag and start a new conversion on the selected channel.

At power-on or external reset both the ADRC and ADON bits are cleared. Thus the A/D is disabled.

Each conversion takes 32 clock cycles which must be at a frequency equal to or greater than 1 MHz.

A multiplexer allows the single A/D converter to select one of six external analog signals two internal signal sources and three internal reference sources.

10.5 Internal and Master Oscillator

If the MCU bus (E clock) frequency is less than 1.0 MHz, an internal RC oscillator (nominally 1.5 MHz) must be used for the A/D conversion clock. This selection is made by setting the ADRC bit in the A/D status and control register to 1.

When the internal RC oscillator is being used as the conversion clock three limitations apply:

1. The conversion complete flag (COCO) must be used to determine when a conversion sequence has been completed, due to the frequency tolerance of the RC oscillator and its asynchronism with regard to the MCU bus clock.
2. The conversion process runs at the nominal 1.5 MHz rate but the conversion results must be transferred to the MCU result registers synchronously with the MCU bus clock so conversion time is limited to a maximum of one channel per bus cycle.

3. If the system clock is running faster than the RC oscillator, the RC oscillator should be turned off, and the system clock used as the conversion clock.

10.6 A/D Registers

10.6.1 A/D Status and Control Register (ADSCR)

The following paragraphs describe the function of the A/D status and control register.

$000F	Bit 7	6	5	4	3	2	1	Bit 0
Read:	COCO	ADRC	ADON	ADTEST	CH3	CH2	CH1	CH0
Write:								
Reset:	U	U	U	U	U	U	U	U

Figure 10-1 A/D Status and Control Register (ADSCR)

COCO – Conversion Complete

This read-only status bit is set when a conversion is completed, indicating that the A/D data register contains valid results. This bit is cleared whenever the A/D status and control register is written and a new conversion automatically started, or whenever the A/D register is read. Once a conversion has been started by writing to the A/D status and control register, conversions of the selected channel will continue every 32 cycles until the A/D status and control register is written again. In this continuous conversion mode, the A/D data register will be filled with new data, and the COCO bit set, every 32 cycles. Data from the previous conversion will be overwritten regardless of the state of the COCO bit prior to writing.

ADRC – RC Oscillator On

When ADRC is set, the A/D section runs on the internal RC oscillator instead of the CPU clock. The RC oscillator requires a time t_{RCON} to stabilize and results can be inaccurate during this time. See **10.5 Internal and Master Oscillator**.

ADON – A/D On

When the A/D is turned on (ADON = 1), it requires a time t_{ADON} for the current sources to stabilize, and results can be inaccurate during this time. This bit turns on the charge pump.

ADTEST

This bit is for test purposes only. Write only 0.

Table 10-2. A/D Clock Selection

ADRC	ADON	Comments
0	0	RC oscillator off, A/D converter off.
0	1	RC oscillator off, A/D converter on.
1	0	RC oscillator on, A/D converter off. Gives time for the RC osc to stabilize.
1	1	RC oscillator on, A/D converter on. A/D using RC osc clocks

CH3:0 – Channel Select Bit

CH3, CH2, CH1 and CH0 form a four bit field which is used to select one of sixteen A/D channels. Channels 8–15 are used for internal reference points. The following table shows the signals selected by the channel select field.

Table 10-1 A/D Channel Assignments

CH3	CH2	CH1	CH0	Channel	Signal
0	0	0	0	0	T_J
0	0	0	1	1	PA1
0	0	1	0	2	PA2
0	0	1	1	3	PA3
0	1	0	0	4	PA4
0	1	0	1	5	PA5
0	1	0	0	6	PA6
0	1	1	1	7	V_{SUP} / α [100mV/bit]

Table 10-1 A/D Channel Assignments

CH3	CH2	CH1	CH0	Channel	Signal
1	0	0	0	8	V_{REFH}
1	0	0	1	9	$(V_{REFH}+V_{REFL})/2$
1	0	1	0	10	V_{REFL}
1	0	1	1	11	V_{REFL}
1	1	X	X	12-15	V_{REFL}

NOTE: *Channel 0 and 7–15 convert internal signals which cannot be accessed externally.*

10.6.2 A/D Data Register

One 8-bit result register is provided. This register is updated each time COCO is set.

$000E	Bit 7	6	5	4	3	2	1	Bit 0
Read:	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
Write:								
Reset:	U	U	U	U	U	U	U	U

Figure 10-3 A/D Data Register (ADDR)

10.7 A/D During WAIT Mode

The A/D converter continues normal operation during WAIT mode. To decrease power consumption during WAIT it is recommended that both the ADON and ADRC bits in the A/D status and control registers be cleared if the A/D converter is not being used. If the A/D converter is in use and the system clock rate is above 1.0 MHz it is recommended that the ADRC bit be cleared.

As the A/D converter continues to function normally in WAIT mode the COCO bit is not cleared.

10.8 A/D During STOP Mode

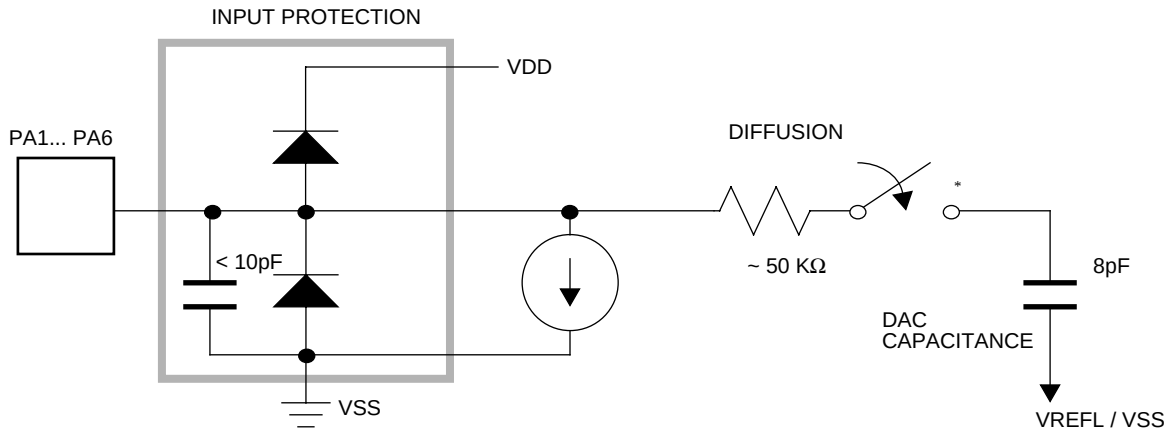
In STOP mode the comparator and charge pump are turned off and the A/D ceases to function. Any pending conversion is aborted. When the clocks begin oscillation upon leaving the STOP mode, a finite amount of time passes before the A/D circuits stabilize enough to provide conversions to the specified accuracy. Normally the delays built into the device when coming out of STOP mode are sufficient for this purpose therefore no explicit delays need to be built into the software.

Although the comparator and charge pump are disabled in STOP mode the A/D data and status/control registers are not modified. Disabling the A/D prior to entering STOP mode will not effect the STOP mode current consumption.

10.9 Analog Input

The external analog voltage value to be converted by the A/D converter is sampled on an internal capacitor through a resistive path provided by input-selection switches and a sampling aperture time switch. Sampling time is limited to 12 bus clock cycles. After sampling, the analog value is stored on a capacitor and held until the end of conversion. During this hold time, the analog input is disconnected from the internal A/D system and the external voltage source sees a high impedance input.

The equivalent analog input during sampling is a RC low-pass filter with resistance around 50 kΩ and a capacitance of around 8pF. (It should be noted that these are typical values measured at room temperature).



* THIS ANALOG SWITCH IS CLOSED ONLY DURING THE 12-CYCLE SAMPLE TIME

Figure 10-4 Electrical Model of an A/D Input Pin

Be sure that pins you want to use as analog inputs are configured as inputs with their appropriate pull-up resistors disabled (enabled after reset).

10.10 Conversion Accuracy Definitions

This section explains the terminology used to specify the analog characteristics of the A/D converter.

10.10.1 Transfer Curve

The ideal transfer curve can be thought of as a staircase of uniform step size with perfect positioning of the endpoints. **Figure 10-5** shows the ideal transfer curve of an 8-bit A/D converter.

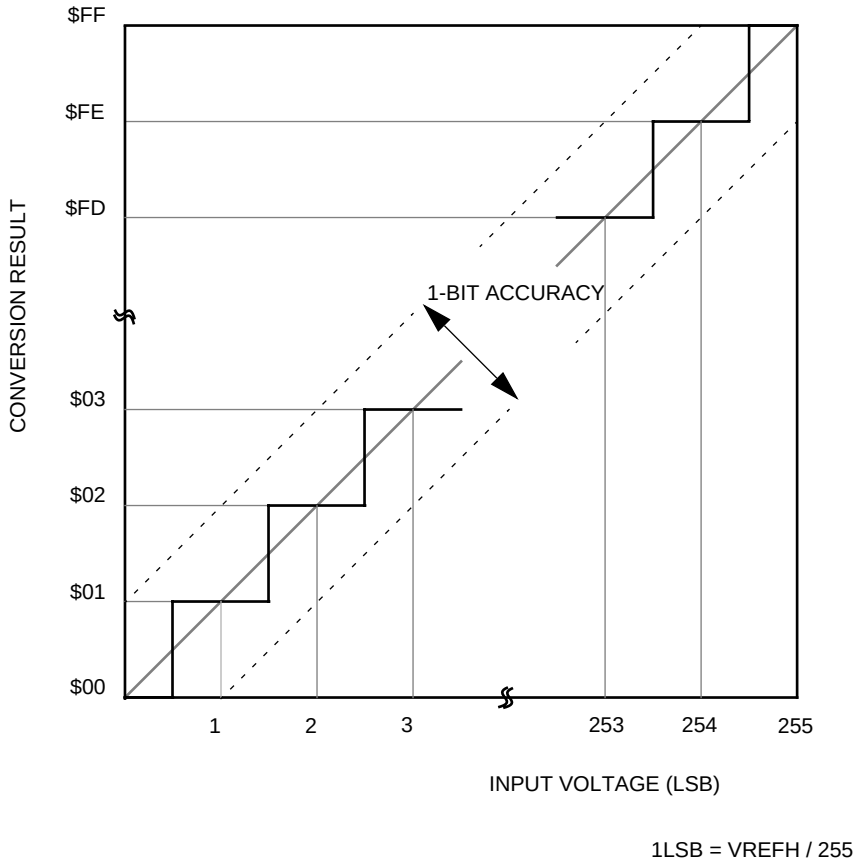


Figure 10-5 Transfer Curve of an Ideal 8-Bit A/D Converter

10.10.2 Monotonicity

The characteristic of the transfer function whereby increasing the input signal results in the output never decreasing.

10.10.3 Quantization Error

Also known as digitization error or uncertainty. It is the inherent error involved in digitizing an analog signal due to the finite number of steps at the digital output versus the infinite number of values at the analog input.

10.10.4 Offset Error

The offset error is the DC shift of the entire transfer curve of an ideal converter.

10.10.5 Gain Scale Error

The gain error is an error in the input to output transfer ratio. Gain error causes an error in the slope of the transfer curve.

10.10.6 Differential Linearity Error

The differential linearity error is the difference between actual analog voltage change and the ideal (1LSB) voltage change at any code change.

10.10.7 Integral Linearity Error

The differential linearity error is the departure from the best fitting line through all A/D code changes.

10.10.8 Total Unadjusted Error

The total unadjustable error is the maximum error that occurs without adjusting offset and gain errors. This error is a combination of offset, scale and integral linearity errors.

Section 11. Pulse Width Modulator

11.1 Contents

11.2 Introduction

The pulse width modulator (PWM) system has one channel. The PWM has a programmable period of PWMPRxT = PWMPR / f_{PWM}, where PWMPR is a programmable period (1... 256) and T = 1 / f_{PWM} can be $1/f_{OSC}$, $1.5/f_{OSC}$, $2/f_{OSC}$, $3/f_{OSC}$ and so on. f_{OSC} is the oscillator frequency.

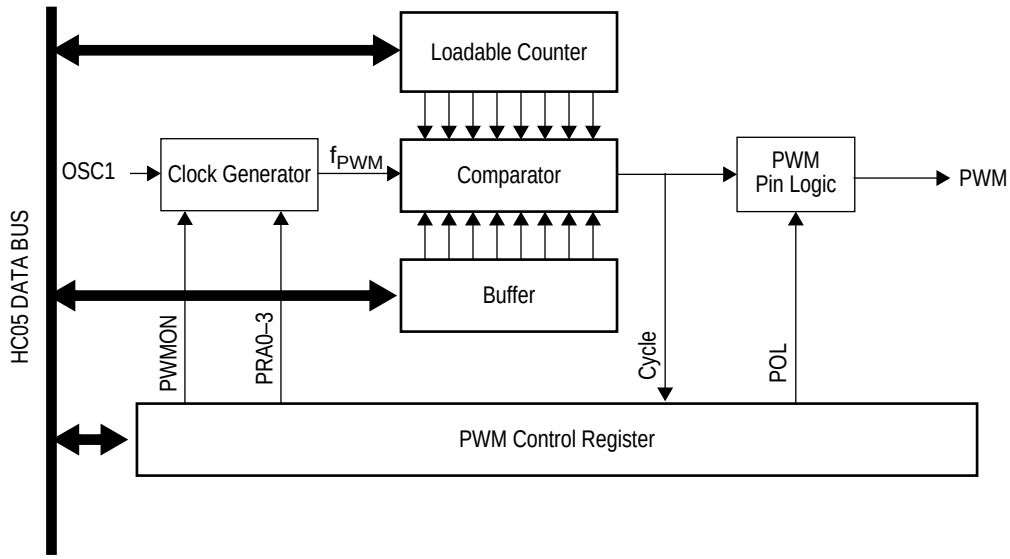


Figure 11-1 PWM Block Diagram

11.3 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 (assuming the PWM period register is set to $FF). To achieve the 100% duty (ON output), the polarity control bit is set while the data register contains $00. When not in use the PWM system can be shut off to save power by clearing the PWMON bit in the PWM control register. The PWM starts conversion immediately after setting PWMON. The PWM output can have an active high or an active low pulse under software control.

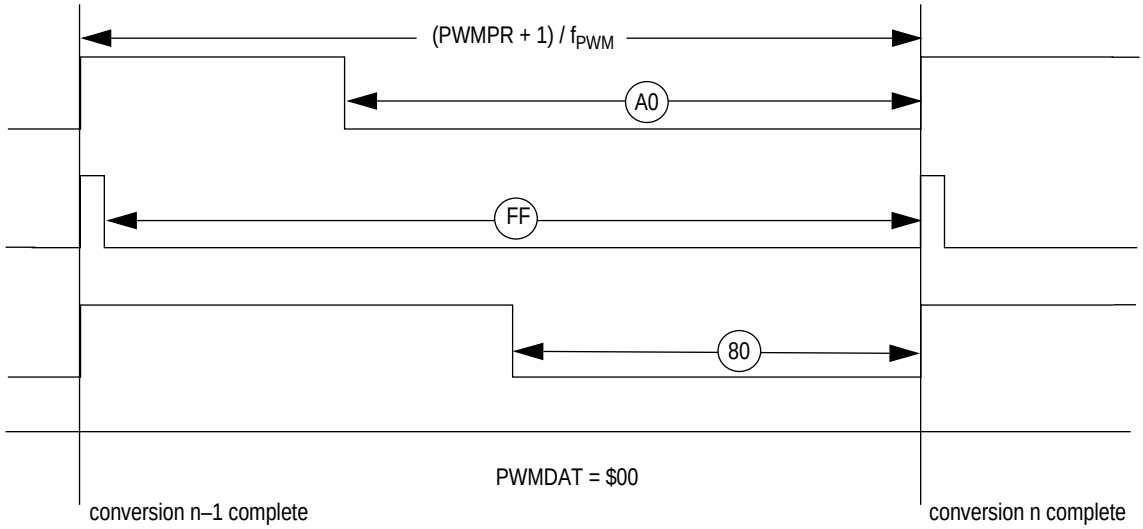


Figure 11-2 PWM Waveforms (POL = 0, active low), PWMPR = $FF

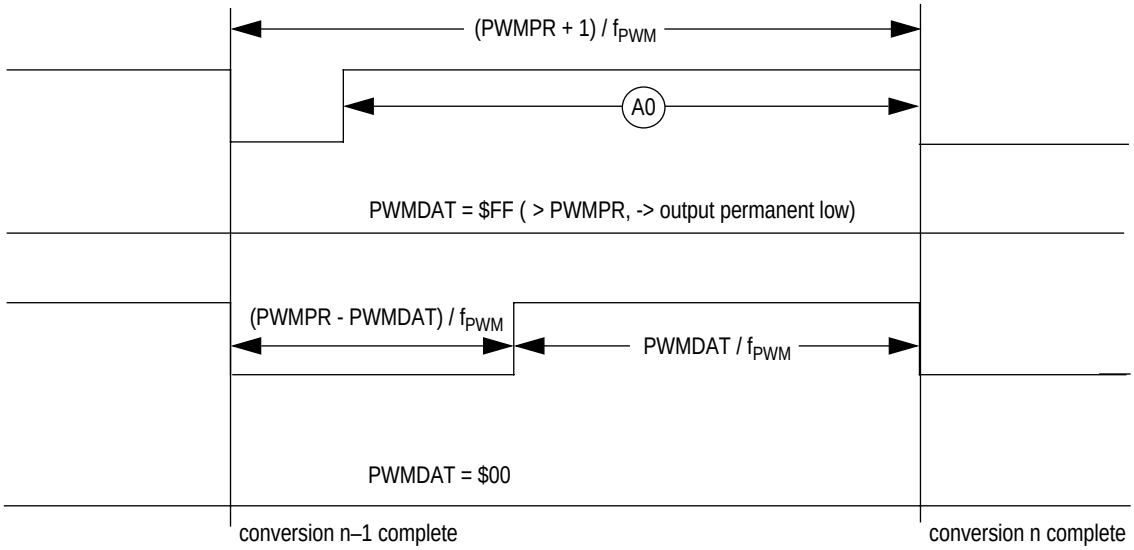


Figure 11-3 PWM Waveforms (POL = 1, active high), PWMPR = $CF

11.4 Registers

Associated with the PWM system, there are a PWM data register, a PWM period register and a PWM control register. These registers can be written to and read at any time. Writing to the data or the period register takes effect when the whole PWM system is started by switching on the PWMON bit or when a conversion cycle is complete. After reset the user should write to the prescaler bits prior to enabling the PWM system. This prevents an erroneous duty cycle from being driven.

11.4.1 PWM Control Register

$002D	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	PWMON	POL		CYCLE	PRA3	PRA2	PRA1	PRA0
Reset:	0	0	0	0	0	0	0	0

Figure 11-4 PWM Control Register (PWMCR)

PWMON – PWM Module On

1 = PWM module operating
0 = PWM module stopped

POL – PWM Polarity

When set, this bit makes the active PWM pulse high. When cleared, the output is active low (e.g. $00 in the data register yields an all high signal for POLA = 0). The programmed polarity bit is copied into a shadow polarity bit when the PWM data register is written. At the end of the current conversion, the shadow polarity bit takes effect.

1 = PWM polarity active high
0 = PWM polarity active low

CYCLE – PWM Cycle Completed

This bit indicates the completion (reload of PWM data and period) of a PWM cycle. This flag is cleared by writing a 1 to the bit position.

1 = PWM registers were reloaded after last flag clear
0 = PWM registers were not reloaded after last flag clear

PRA3, PRA2, PRA1, PRA0 – PWM Clock Rate Bits

These bits select the input clock rate f_{PWM}. For exact values see **Table 11-1**.

The PWM clock rate bits are not latched until the end of conversion. They affect the PWM output immediately. For proper operation these control bits must not be changed during conversion.

Table 11-1 PWM Clock Rate

PRA3:PRA0	f_{PWM}	PRA3:PRA0	f_{PWM}
0000	f_{osc}	1000	$f_{osc}/16$
0001	$f_{osc}/1.5$	1001	$f_{osc}/24$
0010	$f_{osc}/2$	1010	$f_{osc}/32$
0011	$f_{osc}/3$	1011	$f_{osc}/48$
0100	$f_{osc}/4$	1100	$f_{osc}/64$
0101	$f_{osc}/6$	1101	$f_{osc}/96$
0110	$f_{osc}/8$	1110	$f_{osc}/128$
0111	$f_{osc}/12$	1111	$f_{osc}/192$

11.4.2 PWM Data Register

The PWM system has an 8-bit data register that holds the duty cycle for the PWM output. This register can be changed at any time. When the PWMDAT register is updated, the programmed value, as well as the POL bit, take effect in the following conversion cycle. Note that if the contents of PWMDAT are higher than the contents of PWMPR the output will be permanently switched to the passive state (i.e. the same result as PWMDAT = $00).

$002E	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
Reset:	0	0	0	0	0	0	0	0

Figure 11-5 PWM Data Register (PWMDAT)

11.4.3 PWM Period Register

The PWM system has an 8-bit period register that holds the PWM period. The frame frequency of the PWM system is defined as

$f_{frame}=f_{PWM}/(PWMPR + 1)$.

This register can be written at any time. The period of the output changes after the current cycle.

$002C	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
Reset:	1	1	1	1	1	1	1	1

Figure 11-6 PWM Period Register (PWMPR)

11.5 PWM During WAIT Mode

The PWM continues normal operation during WAIT mode. To decrease power consumption during WAIT it is recommended to shut off the PWM by clearing the PWMON bit if the PWM system is not used.

11.6 PWM During STOP Mode

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

11.7 PWM During Reset

Upon reset the PWMON and PRA3–0 bits in the PWM control register are cleared, the data register is written with $00 and the polarity bit is reset. This in effect disables the PWM system and sets the output driving high. The user should write to the data register, the period register, the polarity bit and the clock rate bits prior to enabling the PWM system (i.e. prior to setting PWMON). This prevents an erroneous duty cycle from being driven.

11.8 Frame Frequency Examples

Table 11-2 Frame Frequency for f_{OSC} = 4.2MHz

PRA3–PRA0	PWMPR = $10	PWMPR = $40	PWMPR= $C7	PWMPR = $FF
0000	247KHz	64.5KHz	21KHz	16.4KHz
0001	165KHz	43KHz	14KHz	10.9KHz
0010	123KHz	32.3KHz	10.5KHz	8.2KHz
0111	20.6KHz	5.38KHz	1.75KHz	1.37KHz

Table 11-3 Frame Frequency for f_{OSC} = 2MHz

PRA3–PRA0	PWMPR = $10	PWMPR = $40	PWMPR= $C7	PWMPR = $FF
0000	118KHz	30.8KHz	10KHz	7.81KHz
0001	78.4KHz	20.5KHz	6.67KHz	5.21KHz
0010	58.8KHz	15.4KHz	5KHz	3.91KHz
0111	9.8KHz	1.28KHz	833Hz	651Hz

Section 12. Voltage Regulator

12.1 Contents

12.2 Introduction

The MC68HC(8)05PV8 contains a low-power, low-drop CMOS on-chip fixed voltage regulator to provide internal power to the MCU from an external DC source.

12.3 Internal Power Supply

The on-chip voltage regulation and power supply control circuitry is comprised of two elements: the regulator and the low voltage reset (LVR) circuitry.

12.4 5V Regulator

The 5V regulator accepts an unregulated input supply and provides a regulated 5V supply to all the digital sections of the device. The output of this regulator is also connected to the VDD pin to allow for decoupling and to provide an external power source.

The voltage regulator handles the generation of low voltage resets.

For details refer to **5.12 Low Voltage Reset**.

Any loss of V_{DD} sufficient to trigger an LVR causes the device to be reset. The device remains in the reset state for the duration of the LVR condition or until the internal V_{DD} drops below the functional level of the device, at which point reset no longer has meaning. If the drop in V_{DD} that triggers an LVR is transient, then an internal RST is asserted for a minimum 4064 cycles of the CPU bus clock, PH2 (the POR delay).

12.5 Trimming the Voltage Regulator

The output of the voltage regulator can be trimmed to reach a higher accuracy. This is performed by setting the VT2, VT1 and VT0 bits in the MFTEST register

$002F	Bit 7	6	5	4	3	2	1	Bit 0
Read:	HVTOFF	0	0	VSCAL	LSOFF	VT2	VT1	VT0
Write:		–	–					
Reset:	0	0	0	0	0	0	0	0

Figure 12-1 MFTEST Register (MFTEST)

Table 12-1 illustrates the effect of the trimming bits to V_{DD} in increase or decrease of the output voltage by trimming steps (typically 40mV).

Table 12-1 Trimming Effect

VT2	VT1	VT0	Effect
0	0	0	± 0
0	0	1	–1
0	1	0	–2
0	1	1	–3
1	0	0	+4
1	0	1	+3
1	1	0	+2
1	1	1	+1

Section 13. EEPROM

13.1 Contents

13.2 Introduction

The EEPROM on this device is 128 bytes and is located from address $0180 to $01FF. The user programs the EEPROM on a single-byte basis by manipulating the EEPROM control register (EEPCR).

An erased byte reads as $FF and any programmed bit reads as 0.

13.3 EEPROM Control Register (EEPCR)

$000C	Bit 7	6	5	4	3	2	1	Bit 0
Read:	0	0	0	EEOSC	EER1	EER0	EELAT	EEPGM
Write:								
Reset:	0	0	0	0	0	0	0	0

Figure 13-1 EEPROM Control Register (EEPCR)

EEOSC – EEPROM RC Oscillator Control

When this bit is set, the EEPROM section uses the internal RC oscillator instead of the CPU clock. The user must wait a time t_{RCON} after setting the EEOSC bit to allow the RC oscillator to stabilize. EEOSC is readable and writable. It should be set by the user when the internal bus frequency falls below 1.5 MHz. Reset clears this bit.

EER1, EER0 – Erase Select Bits

EER1 and EER0 form a 2-bit field that is used to select one of three erase modes: byte, block, or bulk erase. **Table 13-1** 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 128-byte block of EEPROM is erased. The EEPROM memory space is divided into two 64-byte blocks ($0180–$01BF, $01C0–$01FF) and performing a block erase on any address within a block will erase the entire block. In bulk erase mode, the entire 128 byte EEPROM section is erased.

A block protect function applies on block2 of the EEPROM memory space. See **13.4 EEPROM Options Register (EEOPR)** for more details.

Table 13-1 Erase Mode Select

EER1	EER0	MODE
0	0	No erase
0	1	Byte erase
1	0	Block erase (block1 or block2)

Table 13-1 Erase Mode Select

EER1	EER0	MODE
1	1	Bulk erase (block1 & block2)

EELAT – EEPROM Programming Latch

The EELAT bit is the EEPROM programming latch enable. When EELAT is at 0, the EER1, EER0 and EEPGM bits are reset to zero. When the EELAT bit is clear, data can be read from the EEPROM. When set, this bit allows the address and data to be latched into the EEPROM for further programming or erase operation. Address and data can only be latched when the EEPGM bit is at 0. STOP, reset and power-on reset reset the EELAT bit.

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 allows pulsing of the programming voltage to be controlled internally. This bit can be read at any time, but can only be written to if EELAT = 1. If EELAT is not set, then EEPGM cannot be set. This bit is cleared by reset or when EELAT = 0.

13.4 EEPROM Options Register (EEOPR)

This register contains the secure and protect functions for the EEPROM and allows the user to select options in a non-volatile manner. The contents of the EEOPR register are loaded into data latches with each power-on or external reset. The register is implemented in EEPROM, therefore reset has no effect on the individual bits.

$0180	Bit 7	6	5	4	3	2	1	Bit 0
Read:							EEPRT	
Write:								
Reset:	U	U	U	U	U	U	U	U

Figure 13-2 EEPROM Options Register (EEOPR)

EEPRT – EEPROM Protect Bit

In order to achieve a higher degree of protection, the EEPROM is split into two 64-byte blocks. Block 1 cannot be protected. Block 2 is protected by the EEPRT bit of the options register. When this bit is set from 0 to 1 (erased) the protection remains until the next power-on or external reset. EEPRT can only be written to 0 when the ELAT bit in the EEPROM control register is set.

1 = Block 2 of the EEPROM array is not protected; all 128 bytes of EEPROM can be accessed for any read, erase or programming operations

0 = Block 2 of the EEPROM array is protected; any attempt to erase or program a location will be unsuccessful

13.5 EEPROM READ, ERASE and Programming Procedures

13.5.1 READ Procedure

To read data from EEPROM the EELAT bit must be clear. EEPGM, EER1 and EER0 are forced to zero. The EEPROM is read as if it were a normal ROM. The charge pump generator is off since EEPGM is zero. If a read is performed while ELAT is set, data will be read as $FF.

13.5.2 ERASE Procedure

There are three types of ERASE operation mode (see **Table 13-1 Erase Mode Select**), byte erase, block erase or bulk erase.

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

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

For a **bulk** erase set EELAT = 1, ER1 = 1, and ER0 = 1, write to an address in the array with A0 or A1 = 1, and set EEPGM for a time t_{EBULK}.

13.5.3 Programming Procedure

To program the content of EEPROM, set EELAT bits, write data to the desired address and set the EEPGM bit. After the required programming delay t_{EEPGM}, EELAT must be cleared. This also resets EEPGM. During a programming operation, any access of EEPROM will return $FF. To program a second byte, EELAT must be cleared before it is set, otherwise the programming will have no effect.

13.6 Operation in STOP and WAIT Modes

The RC oscillator for the EEPROM is automatically disabled when entering STOP mode. The user may want to ensure that the RC oscillator is disabled before entering WAIT mode to help conserve power.

Section 14. Program EEPROM

14.1 Contents

14.2 Introduction

The Program EEPROM on the MC68HC805PV8 is 7936 bytes and is located from address $2000 to $3EFF. It also holds 16 bytes of user vectors ranging from $3FF0 to $3FFF. Programming circuitry embedded in the EEPROM block allows a group of up to four different bytes to be written or erased simultaneously. These four bytes must be located in the set of addresses which differ only in the two least significant bits. An internal charge pump is provided, avoiding the necessity to supply a high voltage for erase and programming functions. In order to achieve a higher degree of security for stored data, there is no capability for bulk or row erase in single chip mode.

14.3 Programming Register

Three bits of the program EEPROM programming register have been provided in order to control the EEPROM operations.

$000D	Bit 7	6	5	4	3	2	1	Bit 0
Read:				RCON	BULK	ERAB	LATB	PGMB
Write:								
Reset:	–	–	–	1	0	1	1	1

Figure 14-1 Program EEPROM Control Register (PEECR)

RCON – RC Oscillator On

This bit determines the state of the RC oscillator. This oscillator should be switched on when the device is operated below 1MHz bus clock. On higher bus speeds, this bit can be switched off to reduce power consumption

1 = RC oscillator switched on
0 = RC oscillator switched off

BULK – Bulk Erase Enable

This bit determines the selection of 4-byte or bulk erase mode. For programming the array, this bit must be cleared.

1 = Bulk erase mode selected
0 = 4-byte erase mode selected

ERAB – Write/Erase Mode Selection

The status of this bit is latched on the first store to EEPROM following the clearing of the LATB bit.

1 = EEPROM write mode
0 = EEPROM erase mode

LATB – Programming Latch Enable

When cleared, this bit allows data and address to be latched into the corresponding EEPROM flip-flops during the first store access to the same EEPROM address. Any subsequent EEPROM store instruction modifies the data register defined by address bits 0 and 1. For normal access to the EEPROM, this bit must be set in order to force the EEPROM address latch to the transparent mode. This bit also

controls the activation of the charge pump. The charge pump is not affected by WAIT mode, thus it is possible to wait the t_{ERA} erase time or t_{PROG} programming time in WAIT mode. The EEPROM is set to read mode when entering STOP mode.

1 = EEPROM read state
0 = Activate charge pump; address and data may be latched for EEPROM write.

PGMB – Programming enable

When cleared, this bit allows programming of the EEPROM. It can only be cleared if the LATB is already cleared and at least one EEPROM write has occurred. This bit must be set when changing the address and data for programming new data. It is automatically set when LATB is set.

1 = EEPROM programming is inhibited
0 = EEPROM programming is enabled

14.4 EEPROM Protection Mechanism

In order to achieve a higher degree of protection, inadvertent programming of the EEPROM can be avoided by use of the EEPRT bit of the options register. As long as this bit is not active (= 0), the whole array, except the first 4 bytes, can be erased or programmed. As soon as the EEPRT bit is active (= 1), the EEPROM is protected and becomes a read-only memory in single chip mode. Note that programming cannot be done by software executed from this EEPROM array!

Any attempt to erase or program a location in single-chip mode will then be unsuccessful. Then the EEPROM can be programmed only in bootloader mode. If the EEPRT bit is then cleared (not protected), the EEPROM will stay protected until the next power-on or external reset.

14.5 Options Register

The options register (OPTR), which also contains the protect function for the Program EEPROM in the MC68HC805PV8 version, is located at $2000 and allows the user to select options in a non-volatile manner. The contents of the OPTR register are loaded into data latches with each reset.

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
OPTR	R		HVRE	HTRE	STOPR		CME	EEPRT	COPD
$2000	W								
RESET		NA	NA	NA	NA	NA	NA	NA	NA

Figure 14-2 Options Register

EEPRT – Program EEPROM Protect (only MC68HC805PV8)

The EEPRT bit allows the Program EEPROM ($2004–$3EFF, $3FF0–$3FFF) to be protected. If the EEPRT bit is in the erased state (logic 0), the EEPROM is not protected and can be used as a regular byte erasable EEPROM. As soon as the EEPRT bit is programmed to 1, the EEPROM is hardware protected. The EEPROM can still be read, but any attempt to erase or program will be unsuccessful. When this bit is cleared, protection remains until the next power-on or external reset occurs. In single chip mode, addresses $2000–$2003 are always write protected.

1 = EEPROM protected
0 = EEPROM erasable and writable

COPD – COP (Computer Operating Properly) Reset Disabled

The COPD bit allows the COP (computer operating properly timer) to be disabled. If the COPD bit is in an erased state (logic 0), the COP is enabled. Programming this bit (logic 1) disables the COP. Changes to this bit do not take effect until the next power-on or external reset occurs.

1 = COP disabled
0 = COP enabled

CME – Clock Monitor Enable

The CME bit enables a watchdog for the oscillator circuit. When the frequency drops below a threshold (due to a brown-out or a defective element), when enabled, the clock monitor will reset the MCU and switch to an internal RC oscillator.

1 = Clock monitor enabled
0 = Clock monitor disabled

STOPR – STOP Reset

When enabled, the MCU will be reset when a STOP instruction is to be executed.

1 = STOP instruction causes reset
0 = STOP instruction executes normally

HTRE – High Temperature Reset Enable

The HTRE bit allows the high temperature reset to be enabled. If the HTRE bit is in erased state (logic 0), the HTR is disabled. Programming this bit (logic 1) enables the HTR. Changes to this bit do not take effect until the next power-on or external reset occurs. See **Section 5. Resets** for details.

1 = HTR enabled
0 = HTR disabled

HVRE – High Voltage Reset Enable

The HVRE bit allows the high voltage reset to be enabled. If the HVRE bit is in erased state (logic 0), the HVR is disabled. Programming this bit (logic 1) enables the HVR. Changes to this bit do not take effect until the next power-on or external reset occurs. See **Section 5. Resets** for details.

1 = HVR enabled
0 = HVR disabled

Section 15. Fast Parallel Interface

15.1 Contents

15.2 Introduction

The MC68HC(8)05PV8 includes a fast parallel interface to access external peripheral components as fast as internal ones. The external address space ranges from $0030 to $003F and all 68HC05 instructions can be applied to this memory. Since the data path is only 4-bits wide either the lines PA7–PA4 or the corresponding data bits in the port A data register are read depending on the state of the DDRA7–DDRA4 bits.

15.3 Description

If this interface is enabled by setting the FPIE bit in the system control register PA0–3 and PB0–3 lines provide a 4 bit address, multiplexed with 4 bit wide data and timing control lines.

The interface uses the lower port A lines (PA0–3) to provide a 4 bit address multiplexed with 4 bit wide data. The timing is controlled by port B lines.

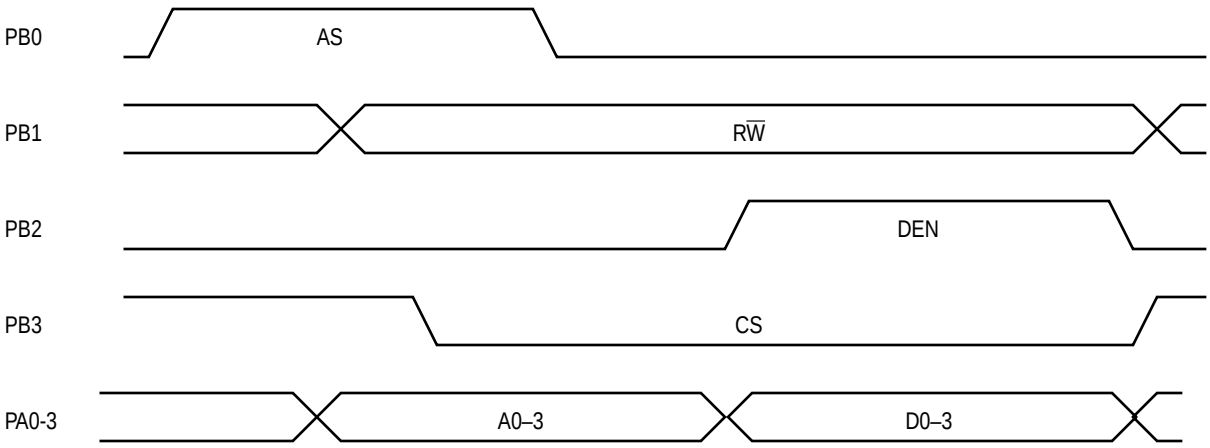


Figure 15-1 Basic Fast Peripheral Interface Timing

The basic timing as shown in **Figure 15-1** is similar to the timing used on the HC11 parts in expanded multiplex mode. At the falling edge of the address strobe signal (AS/PB0) the addresses on PA0–3, the read/write signal ($R\overline{W}$/PB1) and the chip select ($\overline{CS}$/PB3) signal are valid. A high DEN/PB2 signal indicates that data are driven on the bus in CPU write cycles or that the peripheral IC can drive data in read cycles. Whenever the FPICLK bit in the system control register is set the signals become only active when the range from $0030–$003F is addressed by the CPU thus significantly reducing electromagnetic noise.

When using the A/D converter in conjunction with the fast peripheral interface the VRLEN bit of port A configuration register must be cleared. See **7.4.4 Port A Configuration Register**.

15.3.1 System Control Register

The following paragraphs describe the FPIE and FPICLK bit function of the system control register.

$000A	Bit 7	6	5	4	3	2	1	Bit 0
Read:	POR	INTP	INTN	INTE	WCOP	WCP	FPIE	FPICLK
Write:								
Reset:	U	0	0	1	0	0	0	0

Figure 15-2 System Control Register (SYSCR)

FPIE – Fast Peripheral Interface Enable

If this bit is set the fast peripheral interface is enabled. PA0–3 and PB0–3 are no longer available as I/Os.

1 = Fast peripheral interface enabled
0 = Fast peripheral interface disabled

FPICLK – Fast Peripheral Clock

If this bit is set, the FPI clocks are free running

1 = AS and DEN only become active when CPU accesses $0030–$003F
0 = AS and DEN always active

Section 16. Electrical Specifications

16.1 Contents

16.2 Maximum Ratings

(Voltages referenced to V_{SS})

Rating	Symbol	Value	Unit
Supply Voltage	V_{SUP}	–0.3 to +40.0	V
Supply Voltage without using the Voltage Regulator ($V_{SUP} = V_{DD}$)	V_{DD}	–0.3 to +7.0	V
Input Voltage (PA0–7, PB0–4, OSC1)	LV_{IN1}	V_{SS} –0.3 to V_{DD} +0.3	V
Input Voltage (IRQ, RESET)	LV_{IN2}	V_{SS} –0.3 to 12	V
Input Voltage (PC0–3)	HV_{IN1}	V_{SS} –0.3 to V_{SUP} +0.3	V
Input Voltage (PC4)	HV_{IN2}	V_{SS} –5 to V_{SUP} +0.3	V
Applied Voltage (PC5/6)	HV_{IN3}	≤ 40	V
Applied Voltage (PVSS)	HV_{IN4}	V_{SS} to V_{DD}	V
Current Drain per Pin (all I/O, except PC4–6)	I_{OUT1}	25	mA
Current Drain per Pin (VSUP, VDD, PVSS, PC4)	I_{OUT2}	110	mA
Current Drain per Pin (PC5/6, PVSS)	I_{OUT3}	700	mA
Operating Junction Temperature Range	T_J	–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 LV_{IN} and LV_{OUT} be constrained to the range $V_{SS} \le (LV_{IN}$ or $LV_{OUT}) \le V_{DD}$. Reliability of operation could be affected if unused inputs are not connected to an appropriate logic voltage level (e.g., either V_{SS} or V_{DD}, or V_{SS} for the high voltage pins).

16.3 Thermal Characteristics

Characteristic	Symbol	Value	Unit
Thermal Resistance SOIC28	θ_{JA}	60	°C/W

16.4 Program and Data EEPROM Characteristics

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

Rating	Symbol	Min	Max	Unit
Write/Erase Cycles Program EEPROM @ 10ms write time, T_J = +125°C		–	100	Cycles
Write/Erase Cycles Data EEPROM @ 10ms write time, T_J = +125°C		–	10,000	Cycles
Data Retention EEPROMs		–	10	Years
Program EEPROM Programming Time per 4 Bytes	t_{PEEPGM}	5	10	ms
Program EEPROM Erase Time per 4 Bytes	t_{PEBYT}	5	10	ms
Program EEPROM Bulk Erase Time	t_{PEBULK}	400	–	ms
Data EEPROM Programming Time per Byte	t_{EEPGM}	5	10	ms
Data EEPROM Erase Time per Byte	t_{EBYT}	5	10	ms
Data EEPROM Erase Time per Block	t_{EBLOCK}	5	10	ms
Data EEPROM Bulk Erase Time	t_{EBULK}	5	10	ms
RC Oscillators Stabilization Time (Program & Data EEPROM)	t_{RCON}	–	5	t_{CYC}

16.5 Supply Current

(6V ≤ V_{SUP} ≤ 16V, device untrimmed, V_{SS} = 0Vdc, T_J = –40°C to +125°C, unless otherwise noted)

Characteristic	Symbol	Typ	Max	Unit	Comment
Full circuit in Run mode TIMER, A/D, PWM, COP on	I_{SUP1}	4.4	9	mA	See note 2,4
Full circuit in Wait mode TIMER, A/D, PWM, COP on TIMER, A/D, PWM, COP off	 I_{SUP2} I_{SUP3}	 1.95 1.45	 – –	 mA mA	See notes 2, 3 & 4
Full circuit in Stop mode Port C, Op Amp, Power Supply Monitor, Temperature Sensor disabled	I_{SUP4}	485	650	µA	See note 5
Down Scaler Biasing Current	I_{SUP5}	100	–	µA	See note 6
Low Side Driver Biasing Current	I_{SUP6}	280	–	µA	See notes 7, 8, 13
Current Sense Circuitry Internal Reference Biasing Current	I_{SUP7}	600	–	µA	See notes 9 & 10
Current Sense Circuitry to V_{BAT} Biasing Current per Output	I_{SUP8B}	60	–	µA	
Current Sense Circuitry to Ground Biasing Current per Output	I_{SUP8G}	120	–	µA	See note 14
ISO9141 Driver Biasing Current On State	I_{SUP9}	280	–	µA	See note 11
ISO9141 Driver Biasing Current Off State	I_{SUP10}	35	–	µA	
Port C Input Biasing Current	I_{SUP11}	10	–	µA	See note 12

NOTES:

1. Typical values reflect average measurements at mid point of supply voltage range (**VSUP = 12V, VDD = 5V**) and **T_J = 25°C** (applies to all tables).
2. Run (Operating), Wait I_{SUP}: measured using external square wave clock source to OSC1 (F_{OSC} = 4.2 MHz), all inputs 0.2 Vdc from rail; no DC load, all programmable outputs are static, C_L = 20 pF on OSC2.
3. Wait, Stop I_{SUP}: all ports configured as inputs, LV_{IL} = 0.2Vdc, LV_{IH} = V_{DD}–0.2Vdc, HV_I = 0.2Vdc.
4. $I_{SUP1/2/3}$ are affected by the OSC2 capacitance.
5. Stop I_{SUP4} measured with OSC1 = PA0–7 = PB0–4 = IRQ = V_{SS}. $\overline{\text{RESET}}$ open.
6. The down scaler is automatically enabled after any reset and can be disabled by setting the VSCAL bit in the MFTEST register.
7. Low Side Drivers and Die Temperature Monitor can be disabled by setting LSOFF bit in the MFTEST register.
8. The Die Temperature Monitor is only disabled when the LSOFF bit is set and the Port C4 DDR bit is cleared as well.

9. There are two common reference blocks, one for contacts to Vbat and one for contacts to ground.
10. This current is proportional to V_{SUP}.
11. The ISO9141 driver can be disabled by clearing the PCDDR4 bit.
12. The Port C Inputs can be disabled by setting the HVTOFF bit in the MFTEST register.
13. Low Side Drivers must be switched off.
14. Comparators are automatically enabled with the corresponding output.

16.6 V_{DD} Referenced Pins Electrical Characteristics

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

Characteristic	Symbol	Min	Typ	Max	Unit	Comment
Output Low Voltage Port A, Port B	V_{OL1} V_{OL2}	– –	– –	0.1 0.4	V V	I_{LOAD} = 10µA I_{LOAD} = 1.6mA
Output Low Voltage RESET	V_{OL3}	–	–	1	V	I_{LOAD} = 1.6mA
Output High Voltage Port A, Port B	V_{OH1} V_{OH2}	V_{DD}– 0.1 V_{DD}– 0.8	– –	– –	V V	I_{LOAD} = –10µA I_{LOAD} =–0.8mA
Input High Voltage Port A, Port B, $\overline{IRQ}$, $\overline{RESET}$, OSC1	V_{IH}	0.7xV_{DD}	–	V_{DD}	V	
Input Low Voltage Port A, Port B, $\overline{IRQ}$, $\overline{RESET}$,OSC1	V_{IL}	V_{SS}	–	0.3xV_{DD}	V	
Schmitt Trigger Hysteresis Port A, Port B, $\overline{IRQ}$, $\overline{RESET}$	V_{HYS}	–	1	-	V	
Input Pull-up Current PA0–3	I_{IN1}	–	80	250	µA	$V_{IN} = V_{SS}$, see notes
Input Pull-up Current PA4–7	I_{IN2}	–	0.8	2.5	mA	
Input Pull-up Current PA0–3	I_{IN3}	–	50	250	µA	$V_{IN} = 0.7xV_{DD}$, see notes
Input Pull-up Current PA4–7	I_{IN4}	–	0.5	2.5	mA	
Internal Pull-up Resistor RESET	R_{RSTPU}	5	19	50	KΩ	
Input Current $\overline{IRQ}$, OSC1	I_{IN6}	–	–	1	µA	$V_{SS} \le V_{IN} \le V_{DD}$
I/O Ports Hi-Z Leakage Current Port A, Port B	I_{LEAK}	–1	–	1	µA	
Pin Capacitance Port A, Port B, $\overline{RESET}$, $\overline{IRQ}$	C_{PIN}	–	–	10	pF	Not tested
Oscillator Transconductance (I_{OSC2}/V_{OSC2})	g_M	0.9	–	–	mA/V	
Injection Current PA1–5	I_{INJ}	–	–	±5	mA	Not production tested. See also note 3.
Injection Current PB2–4	I_{INJ}	–	–	±1	mA	

NOTES:

(see next page)

1. The pull-up structures on Port A0–7 can be disabled by software, they are automatically enabled by each reset.

2. The pull-up structures on Port A consist of enabled PMOS devices. For input voltages near V_{SS} they act like a constant current source.

3. A simple protection can be built with a series resistor: $R > V_{MAX} / I_{INJ}$. The sum of currents during multiple injection should be limited below the maximum values for a single pin: $R > (V_{MAX} / I_{INJ}) \bullet$(number of pins).
   Positive injection current can raise the supply voltage (V_{DD}). Care must be taken in the application to ensure votages do not exceed the maximum ratings.
   Characterized on the HC08PV8.

16.7 Voltage Regulator

($6V \le V_{SUP} \le 16V$, device untrimmed, V_{SS} = 0Vdc, T_J = –40°C to +125°C, unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit	Comment
Output Voltage ($6V \le V_{SUP} \le 16V$)	V_{DD}	4.75	5.0	5.25	V	$I_{OUT} \le 20mA$
Output Voltage ($5.5V < V_{SUP} \le 40V$)	V_{DD}	4.5	5.0	5.5	V	$I_{OUT} \le 30mA$
Total Output Current	I_{OUT}	–	–	30	mA	See notes 1 & 2
Line Regulation ($6V \le V_{SUP} \le 16V$)	V_{LIR}	–	10	35	mV	I_{OUT} = 1mA
Load Regulation	V_{LOR}	–	50	100	mV	$1mA \le I_{OUT} \le 20mA$
Output Voltage Trimming Step	$V_{STEPTRIM}$	–	40	–	mV	See chapter 12
Low Voltage Reset Low Threshold	V_{LVRON}	4.15	4.40	4.65	V	See notes 3, 4 & **Figure 16-1**
Low Voltage Reset Hysteresis	V_{LVRH}	40	100	200	mV	

NOTES:

1. The current sourcing capability includes the current for the MCU core, for the ports and also for any external load.
2. Refer to the maximum power dissipation.
3. The Low Voltage Reset thresholds and hysteresis are measured relative to V_{DD} with VT2..VT0 cleared in the MFTEST register (POR condition, TRIM 0 configuration).
4. As the voltage regulator and the low voltage reset are using the same internal voltage reference, it is ensured that the low voltage reset will only occur when the voltage regulator is out of regulation.
5. The stability is ensured with a decoupling capacitor between VDD and VSS: $C_{OUT} \ge 10\mu F$ with $ESR \le 10\Omega$.

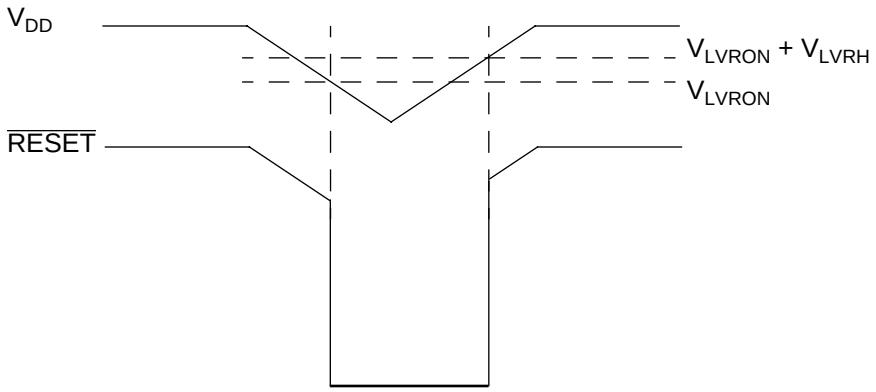


Figure 16-1 Low Voltage Reset waveform

16.8 Operational Amplifier

(device untrimmed, V_{SS} = 0Vdc, T_J = –40°C to +125°C, unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit	Comment
Input Offset Voltage	V_{IO}	–	1	20	mV	
Input Common Mode Voltage Range	V_{ICR}	V_{SS}	–	V_{DD} – 1.2	V	
Large Signal Gain	A_{VOL}	–	30	–	V/mV	
Output Voltage Swing	V_{OH}		V_{SS} .. V_{DD} – 0.2		V	R_{LOAD}= 50KΩ to VSS
Output Short Circuit Current to V_{SS}	I_{SCG}	–	5	–	mA	V_{ID} = 1V, V_O = V_{SS}, T_J = 25°C
Output Short Circuit Current to V_{DD}	I_{SCP}	–	50	–	µA	V_{ID} = –1V, V_O = V_{DD}, T_J = 25°C
Slew Rate	SR	–	1	–	V/µs	V_{IN} = 0.5V to 4.5V, R_{LOAD}= 50KΩ to VSS, C_{LOAD} = 25pF
Gain Bandwidth Product	GBW	–	1	–	MHz	f = 10KHz

16.9 Power Supply Monitor

16.9.1 V_{SUP} related Reset and Interrupts

(device untrimmed, V_{SS} = 0Vdc, T_J = –40°C to +125°C, unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit	Comment
High Voltage Reset On	V_{HVRON}	34.5	36	37.5	V	SEE **FIGURE 16-2**
High Voltage Reset Hysteresis	V_{HVRH}	–	1.5	–	V	
High Voltage Interrupt On	V_{HVION}	29	30.5	32	V	
High Voltage Interrupt Hysteresis	V_{HVIH}	–	1.5	–	V	
Low Voltage Interrupt On	V_{LVION}	6.5	7.5	8.5	V	
Low Voltage Interrupt Hysteresis	V_{LVIH}	–	0.6	–	V	

NOTE:

See chapter **16.7** for the Low Voltage Reset function.

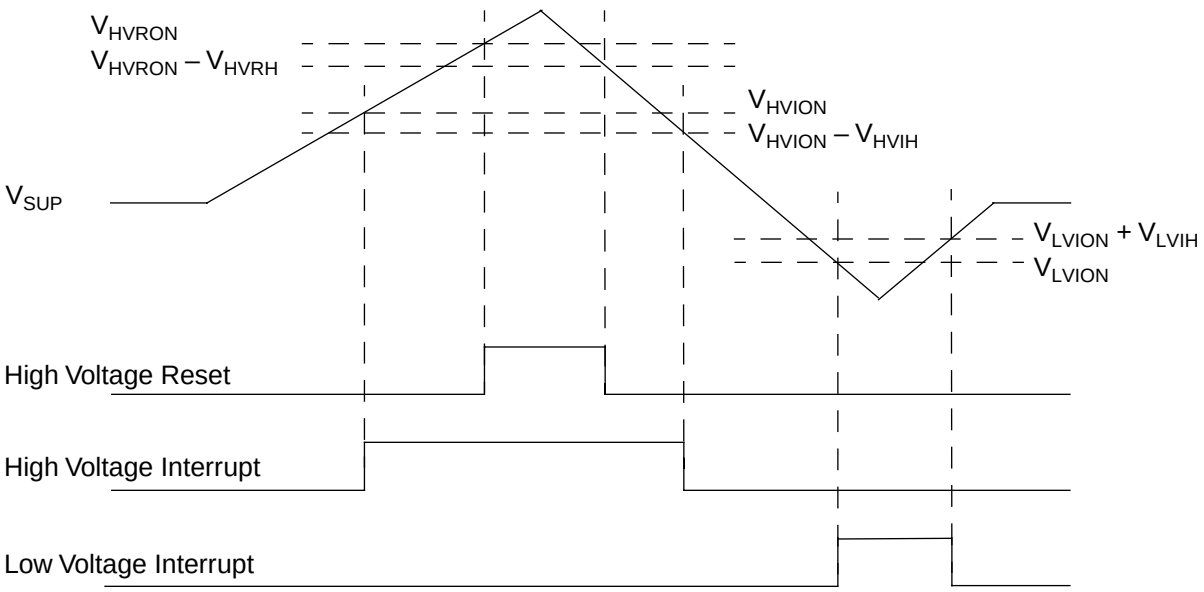


Figure 16-2 VSUP related Reset and Interrupts waveforms

16.10 Down Scaler

(device untrimmed, V_{SS} = 0Vdc, T_J = –40°C to +125°C, unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit	Comment
Voltage Ratio $\alpha = V_{SUP}/V_{AD7}$	α	4.85	5.1	5.35	–	$6V \le V_{SUP} \le 25.5V$, See note 1,2 and chapter 10

NOTE:

1. The Down Scaler output is internally clamped at 5.3V typical.
2. The Down Scaler can only be observed by the A/D. The errors of the A/D has to be taken into account.

16.11 Die Temperature Monitor

(device untrimmed, V_{SS} = 0Vdc, unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit	Comment
High Temperature Reset On	T_{HTRON}	–	150	–	°C	See notes
High Temperature Reset Hysteresis	T_{HTRH}	–	7	–	°C	
High Temperature Interrupt On	T_{HTION}	–	125	–	°C	
High Temperature Interrupt Hysteresis	T_{HTIH}	–	7	–	°C	
Temperature Sensor A/D Reading	N_{TSOUT}	–	142	–	-	T_J = 25°C
Temperature Sensor Output Sensitivity (A/D Reading)	S	–	0.45	–	1/°C	–40°C ≤ T_J ≤ +125°C

NOTE:

1. By design the High Temperature Reset threshold is guaranteed to be (typically 25°C) above the High Temperature Interrupt threshold.
2. Functionality of the device is not guaranteed for $T_J \ge 125$°C. See absolute maximum ratings.

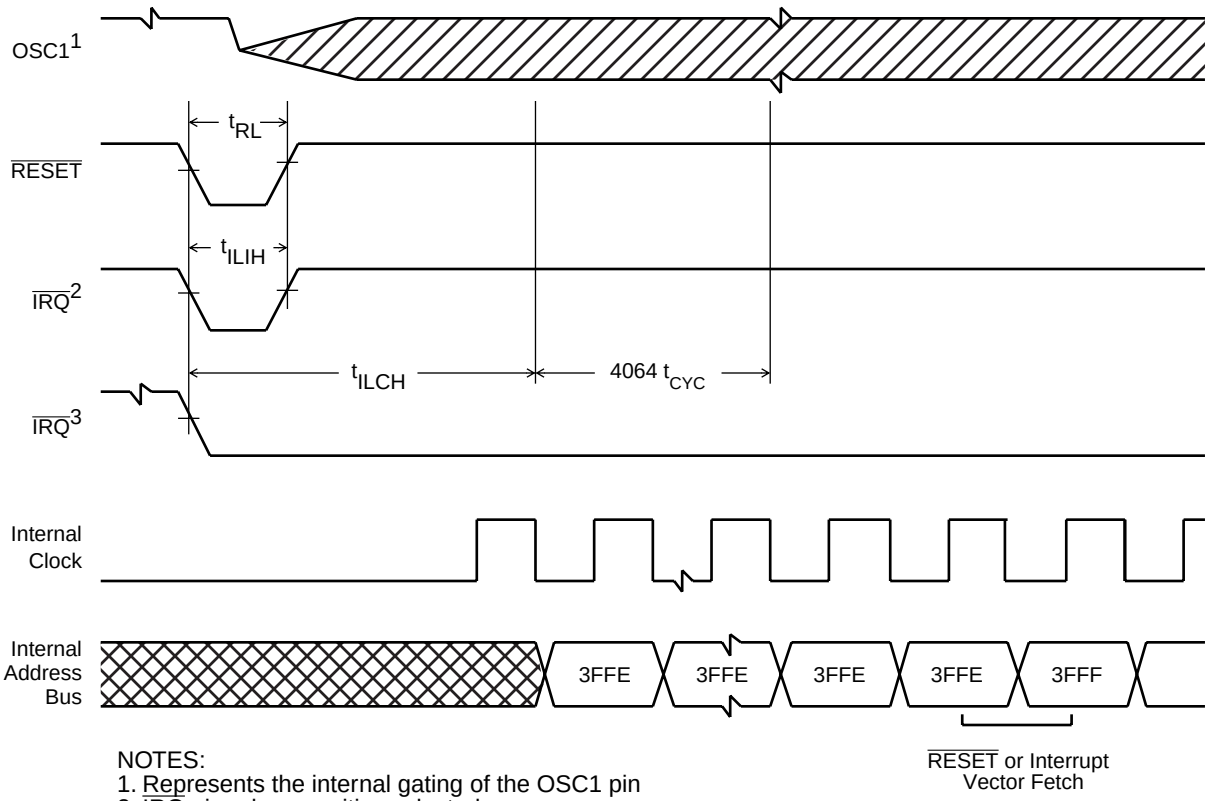
16.12 Control Timing

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

Characteristic	Symbol	Min	Max	Unit
Frequency of Operation Crystal Oscillator Option External Clock Source Oscillator Frequency With Enabled Clock Monitor	 f_{OSC} f_{OSC} f_{OSC}	 0.1 dc 0.4	 4.2 4.2 4.2	 MHz MHz MHz
Cycle Time ($2/f_{OSC}$)	t_{CYC}	476	–	ns
Frequency Detected As Clock Monitor Error	f_{OSC}	dc	10	KHz
Clock Monitor Backup-Oscillator Frequency	f_{OSC}	0.8	4.2	MHz
Crystal Oscillator Start-up Time	t_{OXON}	–	100	ms
Stop Recovery Start-up Time	t_{ILCH}	–	100	ms
$\overline{\text{RESET}}$ Pulse Width Low	t_{RL}	120	–	ns
Interrupt Pulse Width Low (Edge-Triggered)	t_{ILIH}	120	–	ns
Interrupt Pulse Period	t_{ILIL}	note 1	–	t_{CYC}
OSC1 Pulse Width	t_{OSC1}	90	–	ns
16 bit Timer Resolution (note 2) Input Capture Pulse Width Input Capture Period	 t_{RESL} t_{TH}, t_{TL} t_{TLTL}	 4.0 85 note 3	 – – –	 t_{CYC} ns t_{CYC}

NOTES:

1. The minimum period t_{ILIL} or t_{IHIH} should not be less than the number of cycles it takes to execute the interrupt service routine plus 19 t_{CYC}.
2. The 2-bit timer prescaler is the limiting factor in determining timer resolution.
3. The minimum period t_{TLTL} should not be less than the number of cycles it takes to execute the capture interrupt service routine plus 24 t_{CYC}.



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

Figure 16-3 Stop Recovery Timing Diagram

16.13 A/D Converter Characteristics

(V_{REFH} = V_{DD} = 5.0Vdc ±10%, V_{REFL} = 0Vdc, T_J = –40°C to +125°C, unless otherwise noted)

Characteristic	Symbol	Min	Max	Unit	Comment
Resolution	–	8		bit	
Absolute Accuracy	–	±1.5		LSB	Including quantization error
Conversion Range Voltage Reference High Level Voltage Reference Low Level	 V_{REFH} V_{REFL}	V_{REFL} V_{REFL} V_{SS}	V_{REFH} V_{DD} V_{REFH}	V V V	A/D accuracy may decrease proportionately as V_{REFH} is reduced below V_{DD}
Analog Input Voltage	–	V_{REFL}	V_{REFH}	V	Must be within V_{SS} and V_{DD}
Zero Input Reading	–	00	01	Hex	V_{IN} = V_{REFL}
Full-scale Reading	–	FE	FF	Hex	V_{IN} = V_{REFH}
Conversion Time (Including Sampling Time)	T_{CONV}	32		t_{AD}	See note
Sampling Time	T_{SAMP}	12		t_{AD}	
Power-up Time	–	–	100	µs	
A/D On Current Stabilization Time	t_{ADON}	–	100	µs	
RC Oscillator Stabilization Time	t_{RCON}	–	5	µs	
A/D Capacitance	C_{AD}	–	8	pF	Not tested

NOTE:

1. t_{AD} is either the bus clock period or the RC oscillator period (600ns typical).

16.14 Fast Peripheral Interface Timing

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

Characteristic	Symbol	Min	Max	Unit	Comment
DEN/AS Rise and Fall Time	t_R t_F	– –	25 25	ns ns	See ①
Pulse Width AS, DEN high	PW	210	–	ns	See ②
Address, $\overline{CS}$, R$\overline{W}$ setup time	t_{AS}	49	–	ns	See ③
Address, $\overline{CS}$, R$\overline{W}$ hold time	t_{AH}	22	–	ns	See ④
Read data setup time	t_{DSR}	100	–	ns	See ⑤
Read data hold time	t_{DHR}	50	–	ns	See ⑥
Write data setup time	t_{DSW}	30	–	ns	See ⑦
Write data hold time	t_{DHW}	30	–	ns	See ⑧

NOTES:

1. The first cycle denotes a read, the second a write cycle.
2. Unlike in the HC11 AS and DEN occur only when accessing the external memory if not enabled continuously.
3. OSC1/OSC2 input clock other than 50% duty cycle affect bus performance.
4. All timing is shown with respect to 20% V_{DD} and 70% V_{DD}, unless otherwise noted.

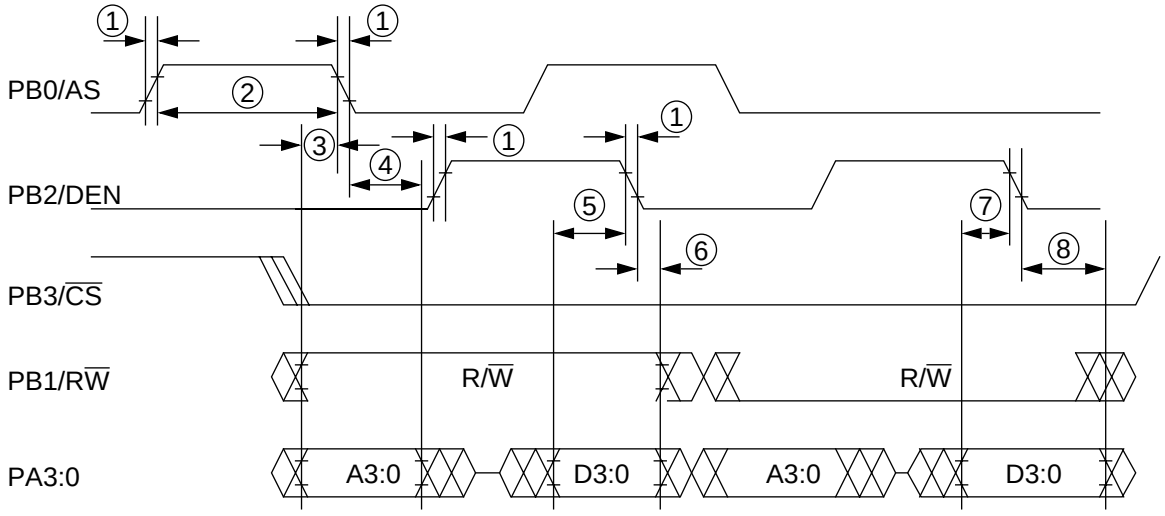


Figure 16-4 Timing definition

16.15 PORT C Characteristics

16.15.1 High Voltage Input/Output (PC0–4)

($6V \le V_{SUP} \le 16V$, device untrimmed, V_{SS} = 0Vdc, T_J = –40°C to +125°C, unless otherwise noted),

Characteristic	Symbol	Min	Typ	Max	Unit	Comment
Input Low Voltage	HV_{IL}	0	–	0.35 x V_{SUP}	V	
Input High Voltage	HV_{IH}	0.65 x V_{SUP}	–	V_{SUP}	V	
Input Hysteresis Voltage	V_{HYS}	0.1	0.1 x V_{SUP}	–	V	
Leakage Current	I_{LEAK}	–5	–	5	µA	Inputs disabled
Input Pull-Down Current	$I_{PULLDOWN}$	–	2.5	10	µA	Inputs enabled, $V_{IN} = V_{SUP}$
Output Low Voltage (PC0–3)	V_{OL}	–	–	0.2 x V_{SUP}	V	I_{LOAD} = 1mA
Output High Voltage (PC0–4)	V_{OH}	0.8 x V_{SUP}	–	–	V	I_{LOAD} = –1mA
Pin Capacitance	C_{OUT}	–	–	10	pF	Not tested

16.15.2 Current Sense Circuitry to Vbattery (PC0–3) and to Ground (PC1–4)

($9V \le V_{SUP} \le 16V$, device untrimmed, V_{SS} = 0Vdc, T_J = –40°C to +125°C, unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit	Comment
Effective Internal Input Resistance	R_{IN}	–	–	600	Ω	$\|I_{LOAD}\|$ = 5mA
Total Path Resistance for Low Threshold	R_{LT}	2.5	4.0	–	KΩ	
Total Path Resistance for High Threshold	R_{HT}	–	6.0	10.0	KΩ	
Total Path Resistance Hysteresis	R_{LT}/R_{HT}	–	0.75	–	–	
Injection Current	I_{INJ}	–	–	±5	mA	Not production tested. See also note 3 on page 171.

16.15.3 ISO9141 Driver (PC4)

(6V ≤ V_{SUP} ≤ 16V, device untrimmed, V_{SS} = 0Vdc, T_J = –40°C to +125°C, unless otherwise noted

Characteristic	Symbol	Min	Typ	Max	Unit	Comment
Output Falling Edge Slew Rate	SR_F	3	5	7	V/µs	$R_{Pull\text{-}up}$ = 510Ω, See note 2
Output Rising Edge Slew Rate	SR_R	3	5	7	V/µs	
Rise Fall Slew Rates Symmetry	ΔSR	–1	0	1	V/µs	
Output Low Voltage	V_{OL}	–	1	1.3	V	I_{LOAD} = 25mA
Leakage Current (driver switched recessive)	I_{LEAK}	–10		10	µA	-5V ≤ V_{IN} ≤ V_{SUP}
Current Limitation Threshold	I_{LIM}	40	55	–	mA	See note 3

NOTES:

1. The ISOMODE bit in PORTC CONFIG0 register must be set.
2. Calculated from 20% to 80% of the output swing.
3. PC4 is not short circuit protected to VSUP.

16.15.4 Low Side Driver (PC5/6, PVSS)

(6V ≤ V_{SUP} ≤ 16V, device untrimmed, Vss = 0 Vdc, T_J = –40°C to +125°C, unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit	Comments
Output Resistance	R_{DS_ON}	–	2	4	Ω	I_{LOAD} = 100mA
Leakage Current	I_{LEAK}	–10	–	10	µA	0V ≤ V_{IN} ≤ 16V
Positive Output Clamp Voltage	V_{CLAMP}	40	42.5	45	V	
Over Current Threshold Shutdown	$I_{SHUTDOWN}$	300	500	700	mA	

APPENDIX B
ELECTRICAL SPECIFICATION FOR CURRENT COMMUNICATION INTERFACE

B.1 Current Interface (PC5 or 6, PVSS)

($6V \leq V_{SUP} \leq 16V$, device untrimmed, Vss = 0 Vdc, $T_J = -40^oC$ to $+125^oC$, unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit	Comments
Output Current	I_{LIM2}	30	35	40	mA	See note 1

NOTE :
1. With an external serial resistor 82.6 Ω ±1%(typically) between PVSS and VSS.

How to reach us:

USA/EUROPE/Locations Not Listed: Motorola Literature Distribution; P.O. Box 5405; Denver, Colorado 80217.1-800-441-2447 or 303-675-2140

Mfax™: RMFAX0@email.sps.mot.com – TOUCHTONE 602-244-6609

INTERNET: http://www.mot.com/SPS/

JAPAN: Nippon Motorola Ltd.; Tatsumi-SPD-JLDC, 6F Seibu-Butsuryu-Center, 3-14-2 Tatsumi Koto-Ku, Tokyo 135, Japan. 81-3-3521-8315

ASIA/PACIFIC: Motorola Semiconductors H.K. Ltd.; 8B Tai Ping Industrial Park, 51 Ting Kok Road, Tai Po, N.T., Hong Kong. 852-26629298