

MC68HC705J1A
MC68HRC705J1A
MC68HSC705J1A
MC68HSR705J1A

HCMOS Microcontroller Units

TECHNICAL DATA

List of Sections

Table of Contents	5
Introduction	9
Pin Descriptions	13
Memory	21
Central Processor Unit.	31
Resets and Interrupts.	53
Low-Power Modes.	63
Parallel I/O Ports	71
Computer Operating Properly Module.	81
External Interrupt Module	85
Multifunction Timer Module	93
Specifications	101
Appendix A MC68HRC705J1A	119
Appendix B MC68HSC705J1A	123
Appendix C MC68HSR705J1A	129
Index	135
Literature Updates	141

List of Modules

All M68HC05 microcontroller units (MCUs) are customer-specified modular designs. To meet customer requirements, Motorola is constantly designing new modules and new versions of existing modules. The following table shows the version levels of the modules in the MC68HC705J1A MCU.

Module	Version
Central Processor Unit	HC05CPU
Computer Operating Properly	@COP0COPRT2
External Interrupt	@KBIELPA4HPD_A
Multifunction Timer	TIM15B1RTICLR_A

Table of Contents

List of Sections	List of Modules	4
Introduction	Contents	9
	Features	10
	Structure	11
	Package Types and Order Numbers	12
	Programmable Options	12
Pin Descriptions	Contents	13
	Pin Assignments	14
	Pin Functions	15
Memory	Contents	21
	Features	21
	Memory Map	22
	Input/Output Register Summary	23
	RAM	24
	EPROM/OTPROM	25
	Mask Option Register	27
	EPROM Programming Characteristics	30
Central Processor Unit	Contents	31
	Features	32
	Introduction	32
	CPU Control Unit	34

	Arithmetic/Logic Unit	34
	CPU Registers	34
	Instruction Set	38
Resets and Interrupts	Contents	53
	Introduction	53
	Resets	54
	Interrupts	56
Low-Power Modes	Contents	63
	Introduction	64
	Exiting Stop and Wait Modes	64
	Effects of Stop and Wait Modes	65
	Data-Retention Mode	68
	Timing	69
Parallel I/O Ports	Contents	71
	Introduction	71
	Port A.	73
	Port B.	76
	I/O Port Electrical Characteristics	79
Computer Operating Properly Module	Contents	81
	Features	81
	Introduction	81
	Operation.	82
	Interrupts	83
	COP Register	83
	Low-Power Modes.	84

External Interrupt Module	Contents	85
	Features	85
	Introduction	86
	Operation.	86
	IRQ Status and Control Register.	91
	Timing	92
Multifunction Timer Module	Contents	93
	Features	93
	Introduction	93
	Operation.	95
	Interrupts	96
	I/O Registers	96
	Low-Power Modes.	100
Specifications	Contents	101
	Maximum Ratings	102
	Operating Temperature Range	103
	Thermal Characteristics.	103
	Power Considerations	104
	5.0 V DC Electrical Characteristics	105
	3.3 V DC Electrical Characteristics	106
	Driver Characteristics	107
	Typical Supply Currents	109
	EPROM Programming Characteristics	111
	Control Timing	112
Mechanical Specifications	116	

Appendix A	Contents	119
MC68HRC705J1A	Introduction	119
	RC Oscillator Connections	119
	Typical Internal Operating Frequency for RC Oscillator Option	120
	Package Types and Order Numbers	122
Appendix B	Contents	123
MC68HSC705J1A	Introduction	124
	DC Electrical Characteristics	125
	Typical Supply Currents	126
	Package Types and Order Numbers	128
Appendix C	Contents	129
MC68HSR705J1A	Introduction	130
	RC Oscillator Connections (External Resistor)	130
	Typical Internal Operating Frequency at 25 °C for High-Speed RC Oscillator Option	131
	RC Oscillator Connections (No External Resistor)	132
	Typical Internal Operating Frequency Versus Temperature (No External Resistor)	133
	Package Types and Order Numbers	134
Index	135
Literature Updates	Literature Distribution Centers	141
	Mfax	142
	Motorola SPS World Marketing World Wide Web Server	142
	CSIC Microcontroller Division's Web Site	142

Contents

Features	10
Structure	11
Package Types and Order Numbers	12
Programmable Options	12

Features

- Peripheral Modules
 - 15-Stage Multifunction Timer
 - Computer Operating Properly (COP) Watchdog
- 14 Bidirectional Input/Output (I/O) Lines, Including:
 - 10 mA Sink Capability on Four I/O Pins
 - Mask Option Register (MOR) and Software Programmable Pulldowns on All I/O Pins
 - MOR Selectable Interrupt on Four I/O Pins, a Keyboard Scan Feature
- MOR Selectable Sensitivity on External Interrupt (Edge- and Level-Sensitive or Edge-Sensitive Only)
- On-Chip Oscillator with Connections for:
 - Crystal
 - Ceramic Resonator
 - Resistor-Capacitor (RC) Oscillator
 - External Clock
- 1240 Bytes of EPROM/OTPROM, Including Eight Bytes for User Vectors
- 64 Bytes of User RAM
- Memory-Mapped I/O Registers
- Fully Static Operation with No Minimum Clock Speed
- Power-Saving Stop, Halt, Wait, and Data-Retention Modes
- External Interrupt Mask Bit and Acknowledge Bit
- Illegal Address Reset
- Internal Steering Diode and Pullup Resistor from $\overline{\text{RESET}}$ Pin to V_{DD}

Structure

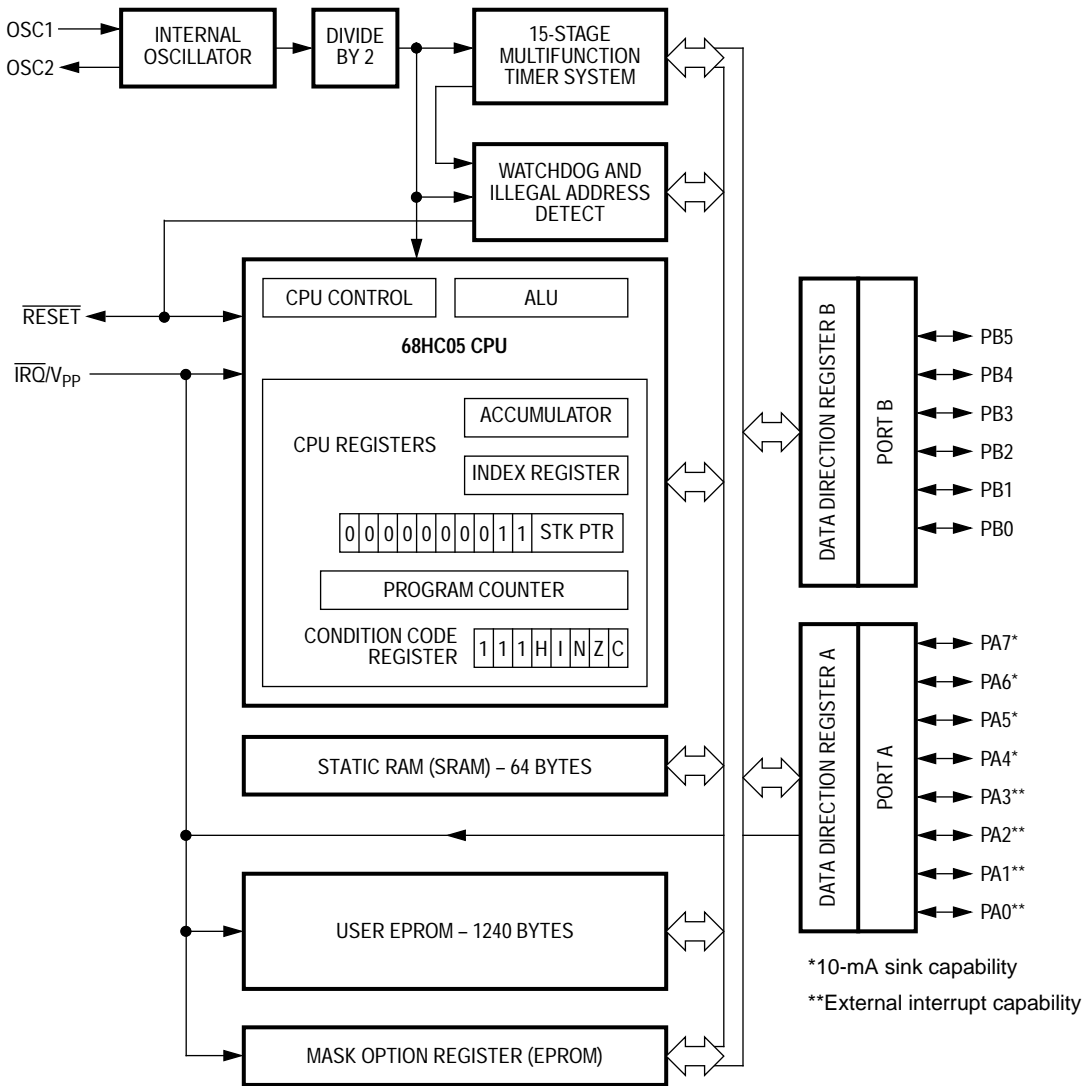


Figure 1. Block Diagram

Package Types and Order Numbers

Table 1. Order Numbers⁽¹⁾

Package Type	Case Outline	Pin Count	Operating Temperature	Order Number
PDIP	738-03	20	0 to 70 °C –40 to +85 °C –40 to + 105 °C	MC68HC705J1AP ⁽²⁾ MC68HC705J1AC ⁽³⁾ P MC68HC705J1AV ⁽⁴⁾ P
SOIC	751D-04	20	0 to 70 °C –40 to +85 °C –40 to + 105 °C	MC68HC705J1ADW ⁽⁵⁾ MC68HC705J1ACDW MC68HC705J1AVDW
Cerdip	732-03	20	0 to 70 °C –40 to +85 °C –40 to + 105 °C	MC68HC705J1AS ⁽⁶⁾ MC68HC705J1ACS MC68HC705J1AVS

1. Refer to [Appendix A](#), [Appendix B](#), and [Appendix C](#) for ordering information on optional high-speed and resistor-capacitor oscillator devices.
2. P = plastic dual in-line package (PDIP)
3. C = extended temperature range
4. V = automotive temperature range
5. DW = small outline integrated circuit (SOIC)
6. S = ceramic dual in-line package (Cerdip)

Programmable Options

The options in [Table 2](#) are programmable in the mask option register.

Table 2. Programmable Options

Feature	Option
COP Watchdog Timer	Enabled or Disabled
External Interrupt Triggering	Edge-Sensitive Only or Edge- and Level-Sensitive
Port A \overline{IRQ} Pin Interrupts	Enabled or Disabled
Port Pulldown Resistors	Enabled or Disabled
STOP Instruction Mode	Stop Mode or Halt Mode
Crystal Oscillator Internal Resistor	Enabled or Disabled
EPROM Security	Enabled or Disabled
Short Oscillator Delay Counter	Enabled or Disabled

Pin Descriptions

Contents

Pin Assignments	14
Pin Functions	15
VDD and VSS	15
OSC1 and OSC2	15
Crystal Oscillator	16
Ceramic Resonator Oscillator	17
RC Oscillator	19
External Clock	19
RESET	19
IRQ/VPP	20
PA0–PA7	20
PB0–PB5	20

Pin Assignments

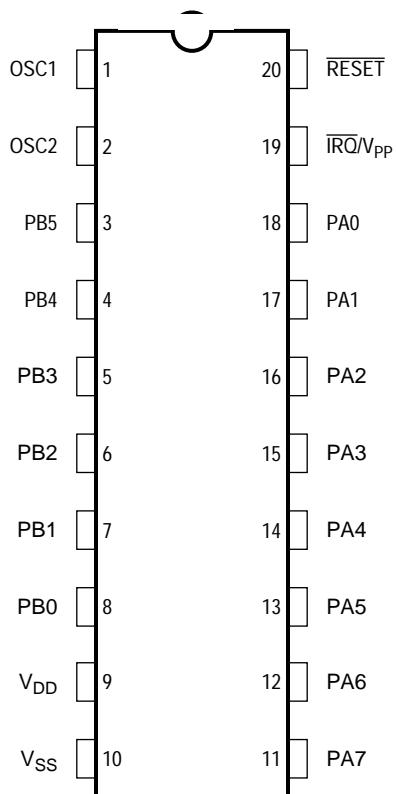


Figure 1. Pin Assignments

Pin Functions

V_{DD} and V_{SS}

V_{DD} and V_{SS} are the power supply and ground pins. The MCU operates from a single power supply.

Very fast signal transitions occur on the MCU pins, placing high, short-duration current demands on the power supply. To prevent noise problems, take special care as [Figure 2](#) shows, by placing the bypass capacitors as close as possible to the MCU. C2 is an optional bulk current bypass capacitor for use in applications that require the port pins to source high current levels.

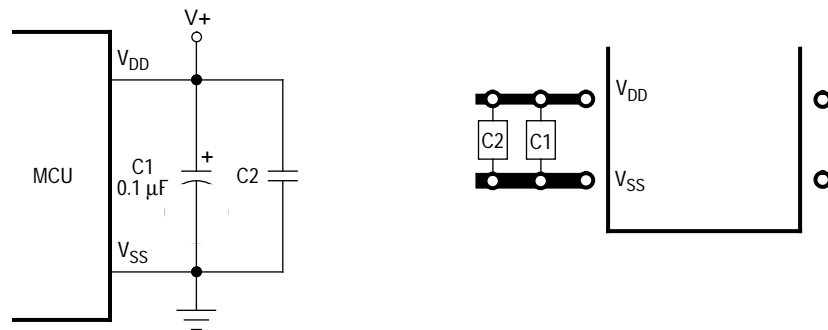


Figure 2. Bypassing Layout Recommendation

OSC1 and OSC2

The OSC1 and OSC2 pins are the connections for the on-chip oscillator. The oscillator can be driven by any of the following:

1. Crystal (See [Figure 3](#) and [Figure 4](#).)
2. Ceramic resonator (See [Figure 5](#) and [Figure 6](#).)
3. Resistor/capacitor (RC) oscillator (Refer to [Appendix A](#) and [Appendix C](#).)
4. External clock signal as shown in (See [Figure 7](#).)

The frequency, f_{OSC} , of the oscillator or external clock source is divided by two to produce the internal operating frequency, f_{OP} .

Crystal Oscillator

Figure 3 and **Figure 4** show a typical crystal oscillator circuit for an AT-cut, parallel resonant crystal. Follow the crystal supplier's recommendations, as the crystal parameters determine the external component values required to provide reliable startup and maximum stability. The load capacitance values used in the oscillator circuit design should include all stray layout capacitances.

To minimize output distortion, mount the crystal and capacitors as close as possible to the pins. An internal startup resistor of approximately 2 M Ω is provided between OSC1 and OSC2 for the crystal oscillator as a programmable mask option.

NOTE: Use an AT-cut crystal and not an AT-strip crystal because the MCU can overdrive an AT-strip crystal.

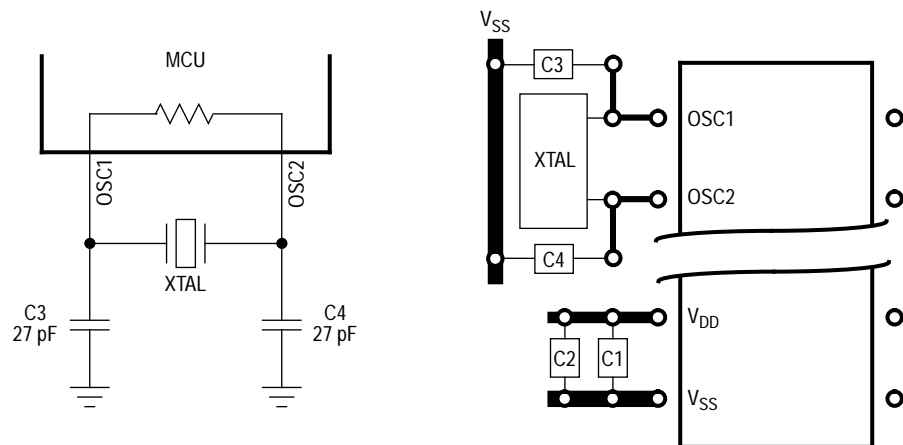


Figure 3. Crystal Connections with Oscillator Internal Resistor Mask Option

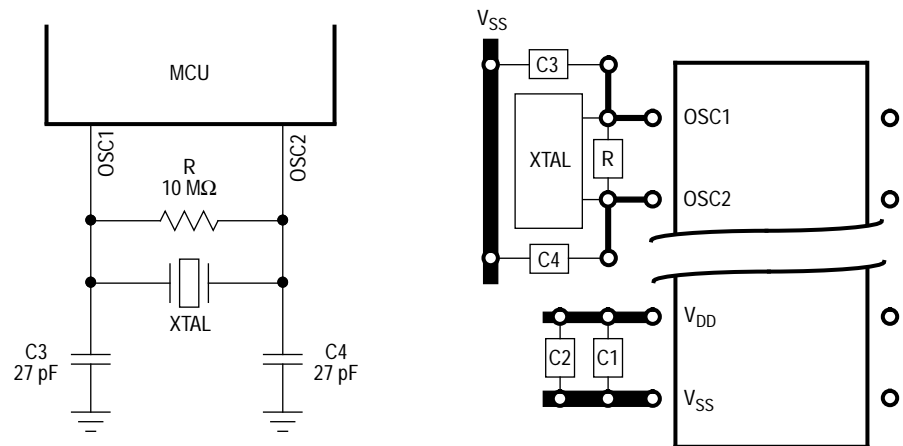


Figure 4. Crystal Connections without Oscillator Internal Resistor Mask Option

Ceramic Resonator Oscillator

To reduce cost, use a ceramic resonator instead of the crystal. The circuits shown in [Figure 5](#) and [Figure 6](#) show ceramic resonator circuits. Follow the resonator manufacturer's recommendations, as the resonator parameters determine the external component values required for maximum stability and reliable starting. The load capacitance values used in the oscillator circuit design should include all stray capacitances.

Mount the resonator and components as close as possible to the pins for startup stabilization and to minimize output distortion. An internal startup resistor of approximately 2 MΩ is provided between OSC1 and OSC2 as a programmable mask option.

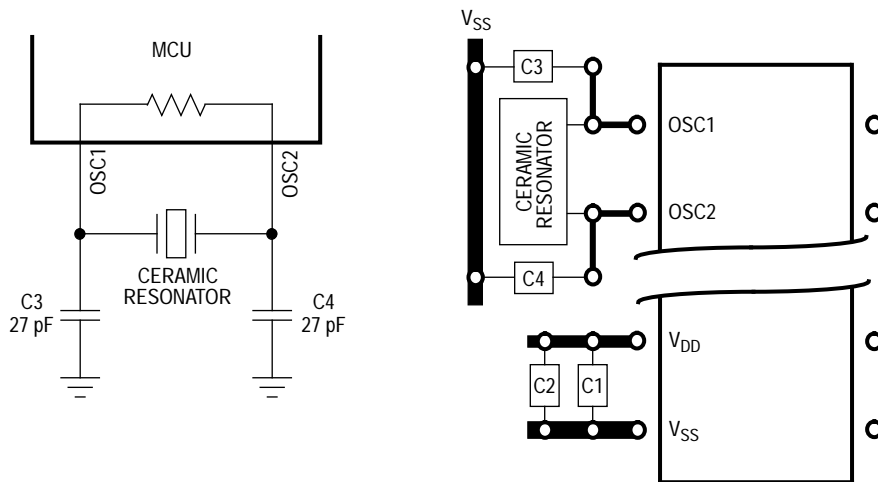


Figure 5. Ceramic Resonator Connections with Oscillator Internal Resistor Mask Option

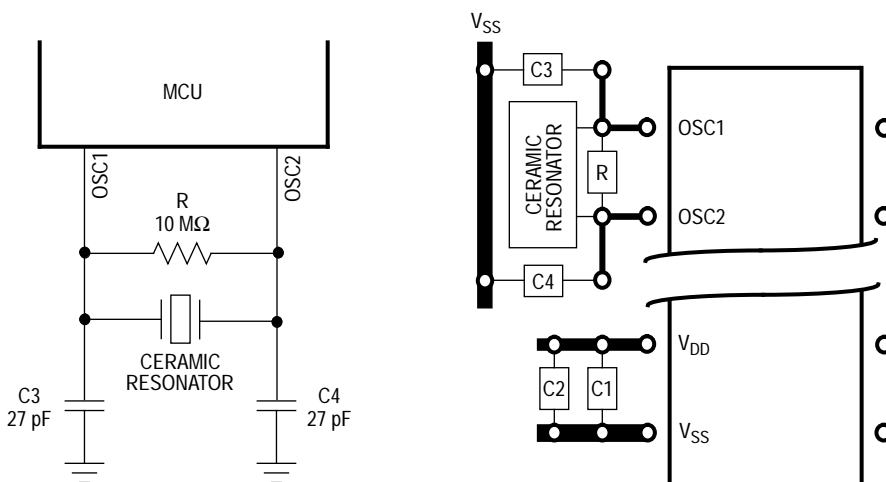


Figure 6. Ceramic Resonator Connections without Oscillator Internal Resistor Mask Option

RC Oscillator

Refer to [Appendix A](#) and [Appendix C](#).

External Clock

An external clock from another CMOS-compatible device can be connected to the OSC1 input, with the OSC2 input not connected, as shown in [Figure 7](#). This configuration is possible regardless of whether the crystal/ceramic resonator or the RC oscillator is enabled.

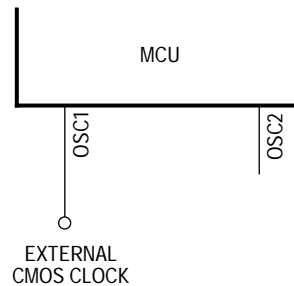


Figure 7. External Clock Connections

 $\overline{\text{RESET}}$

Applying a logic 0 to the $\overline{\text{RESET}}$ pin forces the MCU to a known startup state. An internal reset also pulls the $\overline{\text{RESET}}$ pin low. An internal resistor to V_{DD} pulls the $\overline{\text{RESET}}$ pin high. A steering diode between the $\overline{\text{RESET}}$ and V_{DD} pins discharges any $\overline{\text{RESET}}$ pin voltage when power is removed from the MCU. The $\overline{\text{RESET}}$ pin contains an internal Schmitt trigger to improve its noise immunity as an input. Refer to the [Resets and Interrupts](#) section for more information.

$\overline{\text{IRQ}}/V_{\text{PP}}$

The external interrupt/programming voltage pin ($\overline{\text{IRQ}}/V_{\text{PP}}$) drives the asynchronous IRQ interrupt function of the CPU. Additionally, it is used to program the user EPROM and mask option register. (See [Memory](#) and [External Interrupt Module](#).)

The LEVEL bit in the mask option register provides negative edge-sensitive triggering or both negative edge-sensitive and low level-sensitive triggering for the interrupt function.

If level-sensitive triggering is selected, the $\overline{\text{IRQ}}/V_{\text{PP}}$ input requires an external resistor to V_{DD} for wired-OR operation. If the $\overline{\text{IRQ}}/V_{\text{PP}}$ pin is not used, it must be tied to the V_{DD} supply.

The $\overline{\text{IRQ}}/V_{\text{PP}}$ pin contains an internal Schmitt trigger as part of its input to improve noise immunity. The voltage on this pin should not exceed V_{DD} except when the pin is being used for programming the EPROM.

NOTE: *The mask option register can enable the PA0–PA3 pins to function as external interrupt pins.*

PA0–PA7

These eight input/output (I/O) lines comprise port A, a general-purpose bidirectional I/O port. (See [External Interrupt Module](#) for information on PA0–PA3 external interrupts.)

PB0–PB5

These six I/O lines comprise port B, a general-purpose bidirectional I/O port.

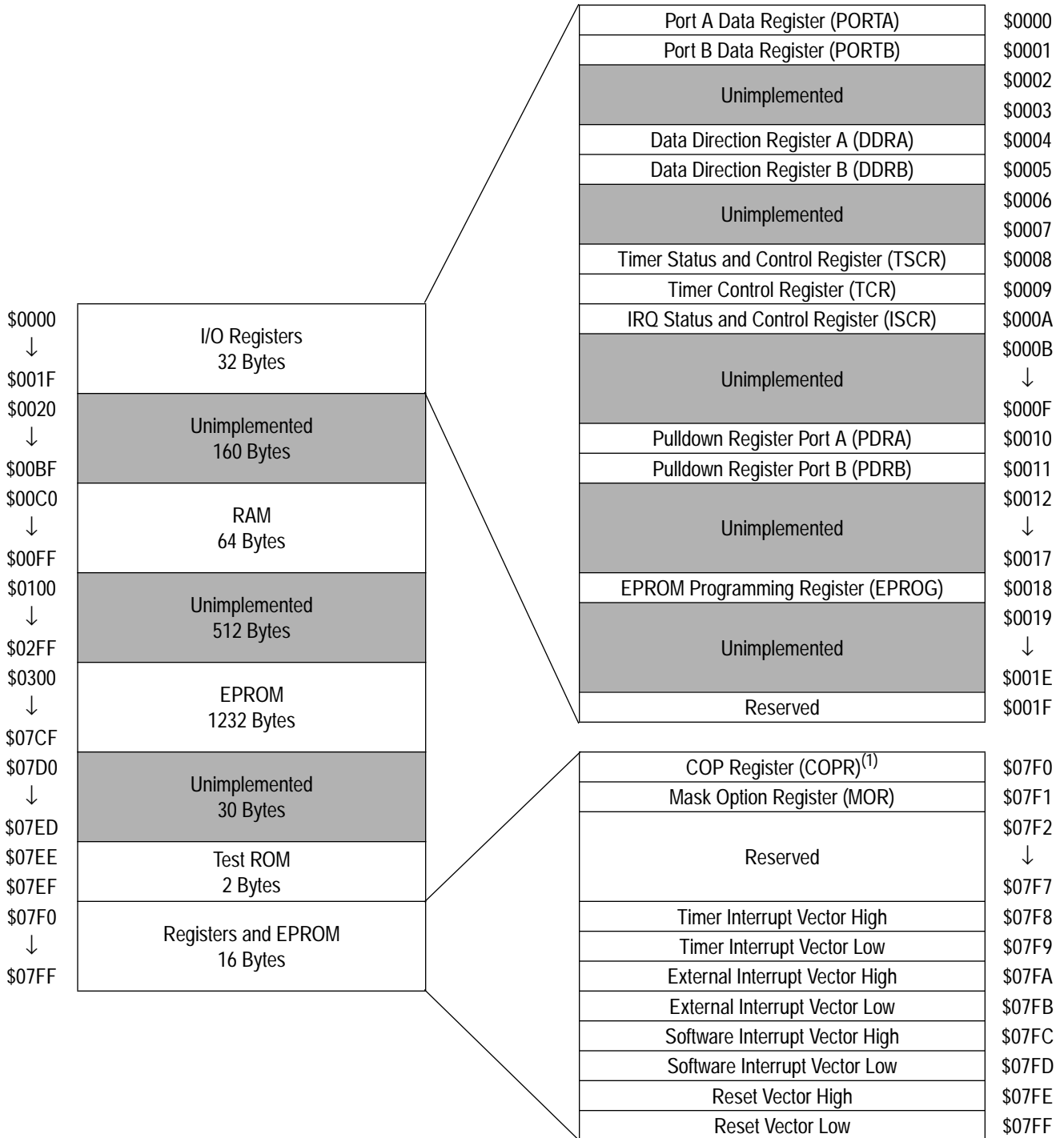
Contents

Features	21
Memory Map	22
Input/Output Register Summary	23
RAM	24
EPROM/OTPROM	25
EPROM/OTPROM Programming	25
EPROM Programming Register	26
EPROM Erasing	27
Mask Option Register	27
EPROM Programming Characteristics	30

Features

- 1232 Bytes of User EPROM, Plus Eight Bytes for User Vectors
- 64 Bytes of User RAM

Memory Map



⁽¹⁾Writing to bit 0 of \$07F0 clears the COP watchdog.

Figure 1. Memory Map

Input/Output Register Summary

Addr.	Register Name	Bit 7	6	5	4	3	2	1	Bit 0
\$0000	Port A Data Register (PORTA)	Read: PA7 Write: PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
		Reset: Unaffected by Reset							
\$0001	Port B Data Register (PORTB)	Read: 0 Write: 0	0	PB5	PB4	PB3	PB2	PB1	PB0
		Reset: Unaffected by Reset							
\$0002	Unimplemented								
\$0003	Unimplemented								
\$0004	Data Direction Register A (DDRA)	Read: DDRA7 Write: DDRA7	DDRA6	DDRA5	DDRA4	DDRA3	DDRA2	DDRA1	DDRA0
		Reset: 0 0 0 0 0 0 0 0							
\$0005	Data Direction Register B (DDRB)	Read: 0 Write: 0	0	DDRB5	DDRB4	DDRB3	DDRB2	DDRB1	DDRB0
		Reset: 0 0 0 0 0 0 0 0							
\$0006	Unimplemented								
\$0007	Unimplemented								
\$0008	Timer Status and Control Register (TSCR)	Read: TOF Write: TOFR	RTIF RTIFR	TOIE	RTIE	0 0	0 0	RT1	RT0
		Reset: 0 0 0 0 0 0 1 1							
\$0009	Timer Counter Register (TCR)	Read: Bit 7 Write: Bit 7	6	5	4	3	2	1	Bit 0
		Reset: 0 0 0 0 0 0 0 0							
\$000A	IRQ Status and Control Register (ISCR)	Read: IRQE Write: IRQE	0	0	0	IRQF R	0	0	0
		Reset: 1 0 0 0 0 0 0 0							
\$000B	Unimplemented								
↓									
\$000F	Unimplemented								
\$0010	Pulldown Register Port A (PDRA)	Read: PDIA7 Write: PDIA7	PDIA6	PDIA5	PDIA4	PDIA3	PDIA2	PDIA1	PDIA0
		Reset: 0 0 0 0 0 0 0 0							

= Unimplemented
 R = Reserved
 U = Unaffected

Figure 2. I/O Register Summary

Memory

Addr.	Register Name	Bit 7	6	5	4	3	2	1	Bit 0	
\$0011	Pulldown Register Port B (PDRB)	Read:								
		Write:			PDIB5	PDIB4	PDIB3	PDIB2	PDIB1	PDIB0
		Reset:	0	0	0	0	0	0	0	0
\$0012	Unimplemented									
↓										
\$0017	Unimplemented									
\$0018	EPROM Programming Register (EPROG)	Read:	0	0	0	0	0	ELAT	MPGM	EPGM
		Write:		R	R	R	R			
		Reset:	0	0	0	0	0	0	0	0
\$0019	Unimplemented									
↓										
\$001E	Unimplemented									
\$001F	Reserved	R	R	R	R	R	R	R	R	
\$07F0	COP Register (COPR)	Read:								
		Write:							COPC	
		Reset:	U	U	U	U	U	U	U	0
\$07F1	Mask Option Register (MOR)	Read:	SOSCD	EPMSEC	OSCREG	SWAIT	PDI	PIRQ	LEVEL	COPEN
		Write:								
		Reset:	0	0	0	0	0	0	0	0

= Unimplemented R = Reserved U = Unaffected

Figure 2. I/O Register Summary

RAM

The 64 addresses from \$00C0 to \$00FF serve as both the user RAM and the stack RAM. Before processing an interrupt, the CPU uses five bytes of the stack to save the contents of the CPU registers. During a subroutine call, the CPU uses two bytes of the stack to store the return address. The stack pointer decrements when the CPU stores a byte on the stack and increments when the CPU retrieves a byte from the stack.

NOTE: *Be careful when using nested subroutines or multiple interrupt levels. The CPU may overwrite data in the RAM during a subroutine or during the interrupt stacking operation.*

EPROM/OTPROM

An MCU with a quartz window has 1240 bytes of erasable, programmable ROM (EPROM). The quartz window allows EPROM erasure with ultraviolet light.

NOTE: *Keep the quartz window covered with an opaque material except when programming the MCU. Ambient light can affect MCU operation.*

In an MCU without the quartz window, the EPROM cannot be erased and serves as 1240 bytes of one-time programmable ROM (OTPROM).

The following addresses are user EPROM/OTPROM locations:

- \$0300–\$07CF
- \$07F8–\$07FF, used for user-defined interrupt and reset vectors

The COP register (COPR) is an EPROM/OTPROM location at address \$07F0.

The mask option register (MOR) is an EPROM/OTPROM location at address \$07F1.

EPROM/OTPROM Programming

The two ways to program the EPROM/OTPROM are:

- Manipulating the control bits in the EPROM programming register to program the EPROM/OTPROM on a byte-by-byte basis
- Programming the EPROM/OTPROM with the M68HC705J In-Circuit Simulator (M68HC705JICS) available from Motorola

EPROM Programming Register

The EPROM programming register (EPROG) contains the control bits for programming the EPROM/OTPROM.

Address: \$0018

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	0	0	0	0	0	ELAT	MPGM	EPGM
Write:		R	R	R	R			
Reset:	0	0	0	0	0	0	0	0

= Unimplemented
 R = Reserved

Figure 3. EPROM Programming Register (EPROG)

ELAT — EPROM Bus Latch Bit

This read/write bit latches the address and data buses for EPROM/OTPROM programming. Clearing the ELAT bit automatically clears the EPGM bit. EPROM/OTPROM data cannot be read while the ELAT bit is set. Reset clears the ELAT bit.

- 1 = Address and data buses configured for EPROM/OTPROM programming the EPROM
- 0 = Address and data buses configured for normal operation

MPGM — MOR Programming Bit

This read/write bit applies programming power from the \overline{IRQ}/V_{PP} pin to the mask option register. Reset clears MPGM.

- 1 = Programming voltage applied to MOR
- 0 = Programming voltage not applied to MOR

EPGM — EPROM Programming Bit

This read/write bit applies the voltage from the \overline{IRQ}/V_{PP} pin to the EPROM. To write the EPGM bit, the ELAT bit must be set already. Reset clears EPGM.

- 1 = Programming voltage (\overline{IRQ}/V_{PP} pin) applied to EPROM
- 0 = Programming voltage (\overline{IRQ}/V_{PP} pin) not applied to EPROM

NOTE: Writing logic 1s to both the ELAT and EPGM bits with a single instruction sets ELAT and clears EPGM. ELAT must be set first by a separate instruction.

Bits [7:3] — Reserved

Take the following steps to program a byte of EPROM/OTPROM:

1. Apply the programming voltage, V_{PP} , to the \overline{IRQ}/V_{PP} pin.
2. Set the ELAT bit.
3. Write to any EPROM/OTPROM address.
4. Set the EPGM bit and wait for a time, t_{EPGM} .
5. Clear the ELAT bit.

EPROM Erasing

The erased state of an EPROM bit is logic 0. Erase the EPROM by exposing it to 15 Ws/cm^2 of ultraviolet light with a wavelength of 2537 angstroms. Position the ultraviolet light source one inch from the EPROM. Do not use a shortwave filter.

Mask Option Register

The mask option register (MOR) is an EPROM/OTPROM byte that controls the following options:

- COP watchdog (enable or disable)
- External interrupt pin triggering (edge-sensitive only or edge- and level-sensitive)
- Port A external interrupts (enable or disable)
- Port pulldown resistors (enable or disable)
- STOP instruction (stop mode or halt mode)
- Crystal oscillator internal resistor (enable or disable)
- EPROM security (enable or disable)
- Short oscillator delay (enable or disable)

Take the following steps to program the mask option register:

1. Apply the programming voltage, V_{PP} , to the \overline{IRQ}/V_{PP} pin.
2. Write to the MOR.
3. Set the MPGM bit and wait for a time, t_{MPGM} .
4. Clear the MPGM bit.
5. Reset the MCU.

Address: \$07F1

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	SOSCD	EPMSEC	OSCREs	SWAIT	SWPDI	PIRQ	LEVEL	COPEN
Write:								
Reset:	0	0	0	0	0	0	0	0

Figure 4. Mask Option Register (MOR)

SOSCD — Short Oscillator Delay Bit

The SOSCD bit controls the oscillator stabilization counter. The normal stabilization delay following reset or exit from stop mode is $4064 t_{cyc}$. Setting SOSCD enables a short oscillator stabilization delay.

- 1 = Short oscillator delay enabled
- 0 = Short oscillator delay disabled

EPMSEC — EPROM Security Bit

The EPMSEC bit controls access to the EPROM/OTPROM.

- 1 = External access to EPROM/OTPROM denied
- 0 = External access to EPROM/OTPROM not denied

OSCREG — Oscillator Internal Resistor Bit

The OSCREG bit enables a 2-M Ω internal resistor in the oscillator circuit.

- 1 = Oscillator internal resistor enabled
- 0 = Oscillator internal resistor disabled

NOTE: *Program the OSCREG bit to logic 0 in devices using RC oscillators.*

SWAIT — Stop-to-Wait Conversion Bit

The SWAIT bit enables halt mode. When the SWAIT bit is set, the CPU interprets the STOP instruction as a WAIT instruction, and the MCU enters halt mode. Halt mode is the same as wait mode, except that an oscillator stabilization delay of 1 to 4064 t_{cyc} occurs after exiting halt mode.

- 1 = Halt mode enabled
- 0 = Halt mode not enabled

SWPDI — Software Pulldown Inhibit Bit

The SWPDI bit inhibits software control of the I/O port pulldown devices. The SWPDI bit overrides the pulldown inhibit bits in the port pulldown inhibit registers.

- 1 = Software pulldown control inhibited
- 0 = Software pulldown control not inhibited

PIRQ — Port A External Interrupt Bit

The PIRQ bit enables the PA0–PA3 pins to function as external interrupt pins.

- 1 = PA0–PA3 enabled as external interrupt pins
- 0 = PA0–PA3 not enabled as external interrupt pins

LEVEL — External Interrupt Sensitivity Bit

The LEVEL bit controls external interrupt triggering sensitivity.

- 1 = External interrupts triggered by active edges and active levels
- 0 = External interrupts triggered only by active edges

COPEN — COP Enable Bit

The COPEN bit enables the COP watchdog.

- 1 = COP watchdog enabled
- 0 = COP watchdog disabled

EPROM Programming Characteristics
Table 1. EPROM Programming Characteristics⁽¹⁾

Characteristic	Symbol	Min	Typ	Max	Unit
Programming Voltage $\overline{\text{IRQ}}/V_{\text{PP}}$	V_{PP}	16.0	16.5	17.0	V
Programming Current $\overline{\text{IRQ}}/V_{\text{PP}}$	I_{PP}	—	3.0	10.0	mA
Programming Time Per Array Byte	t_{EPGM}	4	—	—	ms
MOR	t_{MPGM}	4	—	—	ms

1. $V_{\text{DD}} = 5.0 \text{ Vdc} \pm 10\%$, $V_{\text{SS}} = 0 \text{ Vdc}$, $T_{\text{A}} = -40 \text{ }^{\circ}\text{C}$ to $+105 \text{ }^{\circ}\text{C}$

Central Processor Unit CPU

Contents

Features	32
Introduction	32
CPU Control Unit	34
Arithmetic/Logic Unit	34
CPU Registers	34
Accumulator	35
Index Register	35
Stack Pointer	35
Program Counter	36
Condition Code Register	36
Instruction Set	38
Addressing Modes	38
Inherent	38
Immediate	38
Direct	38
Extended	39
Indexed, No Offset	39
Indexed, 8-Bit Offset	39
Indexed, 16-Bit Offset	40
Relative	40
Instruction Types	41
Register/Memory Instructions	41
Read-Modify- Write Instructions	42
Jump/Branch Instructions	43
Bit Manipulation Instructions	45
Control Instructions	45
Instruction Set Summary	46

Features

- 2.1-MHz Bus Frequency
- 8-Bit Accumulator
- 8-Bit Index Register
- 11-Bit Program Counter
- 6-Bit Stack Pointer
- Condition Code Register with Five Status Flags
- 62 Instructions
- 8 Addressing Modes
- Power-Saving Stop, Wait, Halt, and Data-Retention Modes

Introduction

The central processor unit (CPU) consists of a CPU control unit, an arithmetic/logic unit (ALU), and five CPU registers. The CPU control unit fetches and decodes instructions. The ALU executes the instructions. The CPU registers contain data, addresses, and status bits that reflect the results of CPU operations.

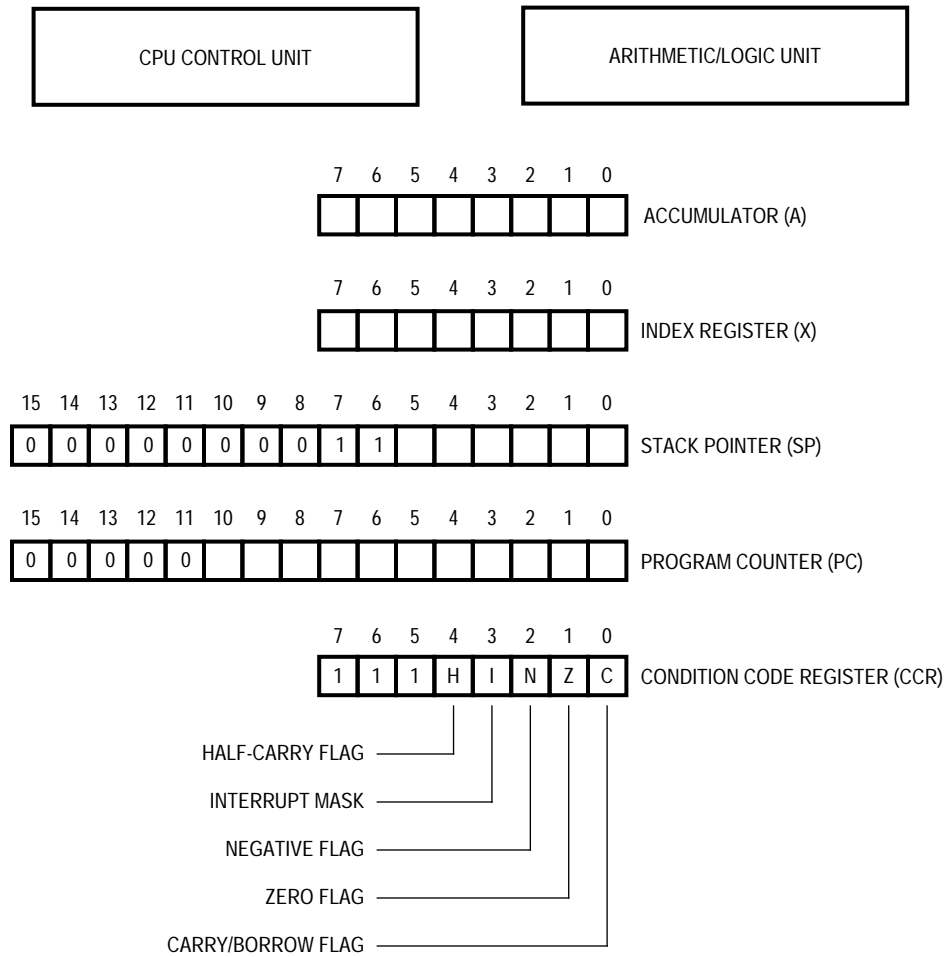


Figure 1. Programming Model

CPU Control Unit

The CPU control unit fetches and decodes instructions during program operation. The control unit selects the memory locations to read and write and coordinates the timing of all CPU operations.

Arithmetic/Logic Unit

The arithmetic/logic unit (ALU) performs the arithmetic, logic, and manipulation operations decoded from the instruction set by the CPU control unit. The ALU produces the results called for by the program and sets or clears status and control bits in the condition code register (CCR).

CPU Registers

The M68HC05 CPU contains five registers that control and monitor MCU operation:

- Accumulator
- Index register
- Stack pointer
- Program counter
- Condition code register

CPU registers are not memory mapped.

Accumulator

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



Figure 2. Accumulator (A)

Index Register

In the indexed addressing modes, the CPU uses the byte in the index register to determine the conditional address of the operand. The index register also can serve as a temporary storage location or a counter.



Figure 3. Index Register (X)

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 instruction (RSP), the stack pointer is preset to \$00FF. The address in the stack pointer decrements after a byte is stacked and increments before a byte is unstacked.

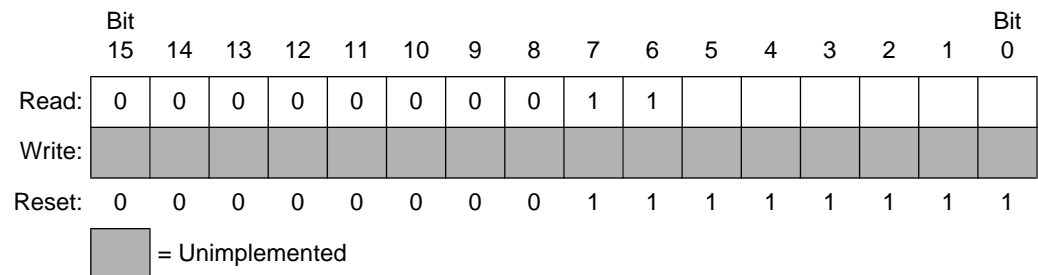


Figure 4. Stack Pointer (SP)

The 10 most significant bits of the stack pointer are permanently fixed at 0000000011, 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.

Program Counter

The program counter is a 16-bit register that contains the address of the next instruction or operand to be fetched. The five most significant bits of the program counter are ignored and appear as 00000.

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.

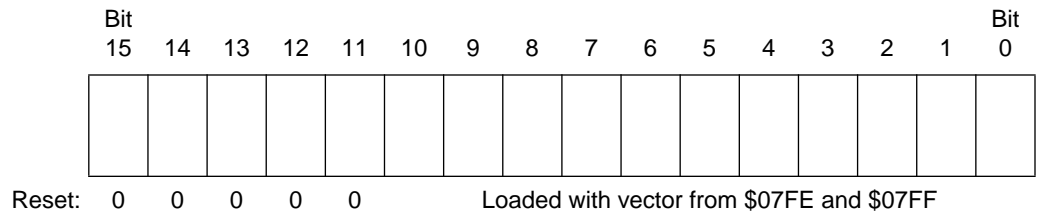


Figure 5. Program Counter (PC)

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.

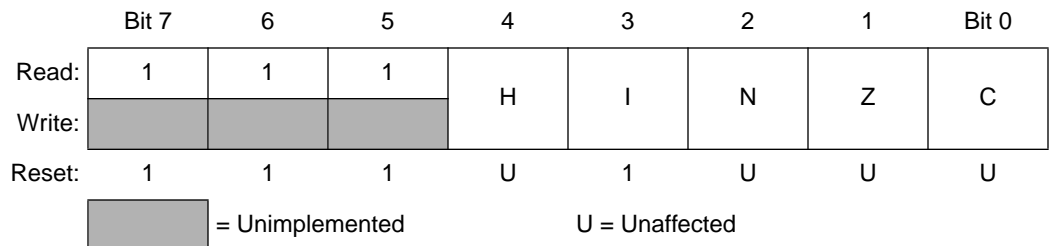


Figure 6. Condition Code Register (CCR)

H — 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.

I — Interrupt Mask

Setting the interrupt mask disables interrupts. If an interrupt request occurs while the interrupt mask is logic 0, 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 logic 1, the interrupt request is latched. Normally, the CPU processes the latched interrupt request as soon as the interrupt mask is cleared again.

A return from interrupt instruction (RTI) unstacks the CPU registers, 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.

N — Negative Flag

The CPU sets the negative flag when an ALU operation produces a negative result.

Z — Zero Flag

The CPU sets the zero flag when an ALU operation produces a result of \$00.

C — 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.

Instruction Set

The MCU instruction set has 62 instructions and uses eight addressing modes.

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

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.

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.

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.

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.

*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 input/output (I/O) location.

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

*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.

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.

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

*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 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

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 instructions on registers with write-only bits.*

Table 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.

*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.

NOTE: *Do not use BRCLR or BRSET instructions on registers with write-only bits.*

Table 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

*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 4. Bit Manipulation Instructions

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

NOTE: *Do not use bit manipulation instructions on registers with write-only bits.*

*Control
Instructions*

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

Table 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

Instruction Set
Summary

Table 6. Instruction Set Summary

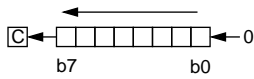
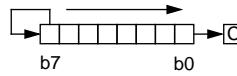
Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			H	I	N	Z	C				
ADC #opr ADC opr ADC opr ADC opr,X ADC opr,X ADC ,X	Add with Carry	$A \leftarrow (A) + (M) + (C)$	↓x	—	↓x	↓x	↓x	IMM DIR EXT IX2 IX1 IX	A9 B9 C9 D9 E9 F9	ii dd hh ll ee ff ff	2 3 4 4 4 3
ADD #opr ADD opr ADD opr ADD opr,X ADD opr,X ADD ,X	Add without Carry	$A \leftarrow (A) + (M)$	↓x	—	↓x	↓x	↓x	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 AND opr AND opr,X AND opr,X AND ,X	Logical AND	$A \leftarrow (A) \wedge (M)$	—	—	↓x	⊗	—	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)		—	—	↓x	⊗	↓x	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		—	—	↓x	↓x	↓x	DIR INH INH IX1 IX	37 47 57 67 77	dd ff	5 3 3 6 5
BCC rel	Branch if Carry Bit Clear	$PC \leftarrow (PC) + 2 + rel ? C = 0$	—	—	—	—	—	REL	24	rr	3
BCLR n opr	Clear Bit n	$M_n \leftarrow 0$	—	—	—	—	—	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)	$PC \leftarrow (PC) + 2 + rel ? C = 1$	—	—	—	—	—	REL	25	rr	3
BEQ rel	Branch if Equal	$PC \leftarrow (PC) + 2 + rel ? Z = 1$	—	—	—	—	—	REL	27	rr	3
BHCC rel	Branch if Half-Carry Bit Clear	$PC \leftarrow (PC) + 2 + rel ? H = 0$	—	—	—	—	—	REL	28	rr	3
BHCS rel	Branch if Half-Carry Bit Set	$PC \leftarrow (PC) + 2 + rel ? H = 1$	—	—	—	—	—	REL	29	rr	3

Table 6. Instruction Set Summary (Continued)

Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			H	I	N	Z	C				
BHI <i>rel</i>	Branch if Higher	$PC \leftarrow (PC) + 2 + rel ? C \vee Z = 0$	—	—	—	—	—	REL	22	rr	3
BHS <i>rel</i>	Branch if Higher or Same	$PC \leftarrow (PC) + 2 + rel ? C = 0$	—	—	—	—	—	REL	24	rr	3
BIH <i>rel</i>	Branch if IRQ Pin High	$PC \leftarrow (PC) + 2 + rel ? IRQ = 1$	—	—	—	—	—	REL	2F	rr	3
BIL <i>rel</i>	Branch if IRQ Pin Low	$PC \leftarrow (PC) + 2 + rel ? IRQ = 0$	—	—	—	—	—	REL	2E	rr	3
BIT # <i>opr</i> BIT <i>opr</i> BIT <i>opr</i> BIT <i>opr</i> ,X BIT <i>opr</i> ,X BIT ,X	Bit Test Accumulator with Memory Byte	(A) \wedge (M)	—	—	↕x	✕	—	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 <i>rel</i>	Branch if Lower (Same as BCS)	$PC \leftarrow (PC) + 2 + rel ? C = 1$	—	—	—	—	—	REL	25	rr	3
BLS <i>rel</i>	Branch if Lower or Same	$PC \leftarrow (PC) + 2 + rel ? C \vee Z = 1$	—	—	—	—	—	REL	23	rr	3
BMC <i>rel</i>	Branch if Interrupt Mask Clear	$PC \leftarrow (PC) + 2 + rel ? I = 0$	—	—	—	—	—	REL	2C	rr	3
BMI <i>rel</i>	Branch if Minus	$PC \leftarrow (PC) + 2 + rel ? N = 1$	—	—	—	—	—	REL	2B	rr	3
BMS <i>rel</i>	Branch if Interrupt Mask Set	$PC \leftarrow (PC) + 2 + rel ? I = 1$	—	—	—	—	—	REL	2D	rr	3
BNE <i>rel</i>	Branch if Not Equal	$PC \leftarrow (PC) + 2 + rel ? Z = 0$	—	—	—	—	—	REL	26	rr	3
BPL <i>rel</i>	Branch if Plus	$PC \leftarrow (PC) + 2 + rel ? N = 0$	—	—	—	—	—	REL	2A	rr	3
BRA <i>rel</i>	Branch Always	$PC \leftarrow (PC) + 2 + rel ? 1 = 1$	—	—	—	—	—	REL	20	rr	3
BRCLR <i>n opr rel</i>	Branch if Bit n Clear	$PC \leftarrow (PC) + 2 + rel ? Mn = 0$	—	—	—	—	↕x	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 <i>rel</i>	Branch Never	$PC \leftarrow (PC) + 2 + rel ? 1 = 0$	—	—	—	—	—	REL	21	rr	3
BRSET <i>n opr rel</i>	Branch if Bit n Set	$PC \leftarrow (PC) + 2 + rel ? Mn = 1$	—	—	—	—	✕	DIR (b0) DIR (b1) DIR (b2) DIR (b3) DIR (b4) DIR (b5) DIR (b6) DIR (b7)	00 02 04 06 08 0A 0C 0E	dd rr dd rr dd rr dd rr dd rr dd rr dd rr dd rr	5 5 5 5 5 5 5 5
BSET <i>n opr</i>	Set Bit n	$Mn \leftarrow 1$	—	—	—	—	—	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

Table 6. Instruction Set Summary (Continued)

Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			H	I	N	Z	C				
BSR <i>rel</i>	Branch to Subroutine	PC ← (PC) + 2; push (PCL) SP ← (SP) - 1; push (PCH) SP ← (SP) - 1 PC ← (PC) + <i>rel</i>	—	—	—	—	—	REL	AD	rr	6
CLC	Clear Carry Bit	C ← 0	—	—	—	—	0	INH	98		2
CLI	Clear Interrupt Mask	I ← 0	—	0	—	—	—	INH	9A		2
CLR <i>opr</i> CLRA CLR X CLR <i>opr,X</i> 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 # <i>opr</i> CMP <i>opr</i> CMP <i>opr</i> CMP <i>opr,X</i> CMP <i>opr,X</i> CMP ,X	Compare Accumulator with Memory Byte	(A) - (M)	—	—	↓x	↓x	↓	IMM DIR EXT IX2 IX1 IX1 IX	A1 B1 C1 D1 E1 F1	ii dd hh ll ee ff ff	2 3 4 5 4 3
COM <i>opr</i> COMA COM X COM <i>opr,X</i> COM ,X	Complement Byte (One's Complement)	M ← (\bar{M}) = \$FF - (M) A ← (\bar{A}) = \$FF - (A) X ← (\bar{X}) = \$FF - (X) M ← (\bar{M}) = \$FF - (M) M ← (\bar{M}) = \$FF - (M)	—	—	↓x	↓x	1	DIR INH INH IX1 IX	33 43 53 63 73	dd ff	5 3 3 6 5
CPX # <i>opr</i> CPX <i>opr</i> CPX <i>opr</i> CPX <i>opr,X</i> CPX <i>opr,X</i> CPX ,X	Compare Index Register with Memory Byte	(X) - (M)	—	—	↓x	↓x	↓x	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 <i>opr</i> DECA DEC X DEC <i>opr,X</i> DEC ,X	Decrement Byte	M ← (M) - 1 A ← (A) - 1 X ← (X) - 1 M ← (M) - 1 M ← (M) - 1	—	—	↓x	↓x	—	DIR INH INH IX1 IX	3A 4A 5A 6A 7A	dd ff	5 3 3 6 5
EOR # <i>opr</i> EOR <i>opr</i> EOR <i>opr</i> EOR <i>opr,X</i> EOR <i>opr,X</i> EOR ,X	EXCLUSIVE OR Accumulator with Memory Byte	A ← (A) ⊕ (M)	—	—	↓x	↓x	—	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 <i>opr</i> INCA INC X INC <i>opr,X</i> INC ,X	Increment Byte	M ← (M) + 1 A ← (A) + 1 X ← (X) + 1 M ← (M) + 1 M ← (M) + 1	—	—	↓x	↓x	—	DIR INH INH IX1 IX	3C 4C 5C 6C 7C	dd ff	5 3 3 6 5

Table 6. Instruction Set Summary (Continued)

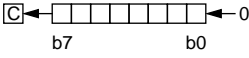
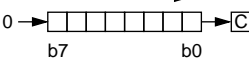
Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			H	I	N	Z	C				
JMP <i>opr</i> JMP <i>opr</i> JMP <i>opr,X</i> JMP <i>opr,X</i> JMP ,X	Unconditional Jump	PC ← Jump Address	—	—	—	—	—	DIR EXT IX2 IX1 IX	BC CC DC EC FC	dd hh ll ee ff ff ff	2 3 4 3 2
JSR <i>opr</i> JSR <i>opr</i> JSR <i>opr,X</i> JSR <i>opr,X</i> 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 ff	5 6 7 6 5
LDA # <i>opr</i> LDA <i>opr</i> LDA <i>opr</i> LDA <i>opr,X</i> LDA <i>opr,X</i> LDA ,X	Load Accumulator with Memory Byte	A ← (M)	—	—	↑x	x	—	IMM DIR EXT IX2 IX1 IX	A6 B6 C6 D6 E6 F6	ii dd hh ll ee ff ff ff	2 3 4 5 4 3
LDX # <i>opr</i> LDX <i>opr</i> LDX <i>opr</i> LDX <i>opr,X</i> LDX <i>opr,X</i> LDX ,X	Load Index Register with Memory Byte	X ← (M)	—	—	↑x	↑x	—	IMM DIR EXT IX2 IX1 IX	AE BE CE DE EE FE	ii dd hh ll ee ff ff ff	2 3 4 5 4 3
LSL <i>opr</i> LSLA LSLX LSL <i>opr,X</i> LSL ,X	Logical Shift Left (Same as ASL)		—	—	↑x	x	↑x	DIR INH INH IX1 IX	38 48 58 68 78	dd ff ff	5 3 3 6 5
LSR <i>opr</i> LSRA LSRX LSR <i>opr,X</i> LSR ,X	Logical Shift Right		—	—	0	↓	↓x	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 <i>opr</i> NEGA NEGX NEG <i>opr,X</i> 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)	—	—	↑x	x	x	DIR INH INH IX1 IX	30 40 50 60 70	dd ff	5 3 3 6 5
NOP	No Operation		—	—	—	—	—	INH	9D		2
ORA # <i>opr</i> ORA <i>opr</i> ORA <i>opr</i> ORA <i>opr,X</i> ORA <i>opr,X</i> ORA ,X	Logical OR Accumulator with Memory	A ← (A) ∨ (M)	—	—	↑x	x	—	IMM DIR EXT IX2 IX1 IX	AA BA CA DA EA FA	ii dd hh ll ee ff ff ff	2 3 4 5 4 3

Table 6. Instruction Set Summary (Continued)

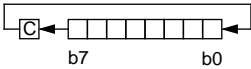
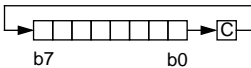
Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			H	I	N	Z	C				
ROL <i>opr</i> ROLA ROLX ROL <i>opr,X</i> ROL ,X	Rotate Byte Left through Carry Bit		—	—	↕x	x	↕x	DIR INH INH IX1 IX	39 49 59 69 79	dd ff	5 3 6 3 5
ROR <i>opr</i> RORA RORX ROR <i>opr,X</i> ROR ,X	Rotate Byte Right through Carry Bit		—	—	↕x	x	x	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)	↕x	x	x	↕x	x	INH	80		9
RTS	Return from Subroutine	SP ← (SP) + 1; Pull (PCH) SP ← (SP) + 1; Pull (PCL)	—	—	—	—	—	INH	81		6
SBC # <i>opr</i> SBC <i>opr</i> SBC <i>opr</i> SBC <i>opr,X</i> SBC <i>opr,X</i> SBC ,X	Subtract Memory Byte and Carry Bit from Accumulator	A ← (A) – (M) – (C)	—	—	x	↕x	x	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 <i>opr</i> STA <i>opr</i> STA <i>opr,X</i> STA <i>opr,X</i> STA ,X	Store Accumulator in Memory	M ← (A)	—	—	↕x	x	—	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 <i>opr</i> STX <i>opr</i> STX <i>opr,X</i> STX <i>opr,X</i> STX ,X	Store Index Register In Memory	M ← (X)	—	—	↕x	x	—	DIR EXT IX2 IX1 IX	BF CF DF EF FF	dd hh ll ee ff ff	4 5 6 5 4
SUB # <i>opr</i> SUB <i>opr</i> SUB <i>opr</i> SUB <i>opr,X</i> SUB <i>opr,X</i> SUB ,X	Subtract Memory Byte from Accumulator	A ← (A) – (M)	—	—	x	↕x	↕	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

Table 6. Instruction Set Summary (Continued)

Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			H	I	N	Z	C				
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
TST <i>opr</i> TSTA TSTX TST <i>opr</i> ,X TST ,X	Test Memory Byte for Negative or Zero	(M) – \$00	—	—	↓x	x	—	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		—	0x	—	—	—	INH	8F		2

- | | | | |
|-------|---|------------|--------------------------------------|
| A | Accumulator | <i>opr</i> | Operand (one or two bytes) |
| C | Carry/borrow flag | PC | Program counter |
| CCR | Condition code register | PCH | Program counter high byte |
| dd | Direct address of operand | PCL | Program counter low byte |
| dd rr | Direct address of operand and relative offset of branch instruction | REL | Relative addressing mode |
| DIR | Direct addressing mode | <i>rel</i> | Relative program counter offset byte |
| ee ff | High and low bytes of offset in indexed, 16-bit offset addressing | rr | Relative program counter offset byte |
| EXT | Extended addressing mode | SP | Stack pointer |
| ff | Offset byte in indexed, 8-bit offset addressing | X | Index register |
| H | Half-carry flag | Z | Zero flag |
| hh ll | High and low bytes of operand address in extended addressing | # | Immediate value |
| I | Interrupt mask | ^ | Logical AND |
| ii | Immediate operand byte | ∨ | Logical OR |
| IMM | Immediate addressing mode | ⊕ | Logical EXCLUSIVE OR |
| INH | Inherent addressing mode | () | Contents of |
| IX | Indexed, no offset addressing mode | -() | Negation (two's complement) |
| IX1 | Indexed, 8-bit offset addressing mode | ← | Loaded with |
| IX2 | Indexed, 16-bit offset addressing mode | ? | If |
| M | Memory location | : | Concatenated with |
| N | Negative flag | ↑ | Set or cleared |
| n | Any bit | — | Not affected |

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

INH = Inherent
 IMM = Immediate
 DIR = Direct
 EXT = Extended
 REL = Relative
 IX = Indexed, No Offset
 IX1 = Indexed, 8-Bit Offset
 IX2 = Indexed, 16-Bit Offset

LSB of Opcode in Hexadecimal

MSB	0
LSB	BRSET0 ⁵ ₃ DIR

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

Resets and Interrupts

Contents	Introduction	53
	Resets	54
	Power-On Reset	54
	External Reset	55
	COP Watchdog Reset	56
	Illegal Address Reset	56
	Interrupts	56
	Software Interrupt	56
	External Interrupt	56
	Timer Interrupts	59
	Real-Time Interrupt	59
	Timer Overflow Interrupt	59
	Interrupt Processing	59

Introduction Reset initializes the MCU by returning the program counter to a known address and by forcing control and status bits to known states.

 Interrupts temporarily change the sequence of program execution to respond to events that occur during processing.

Resets

A reset immediately stops the operation of the instruction being executed, initializes certain control and status bits, and loads the program counter with a user-defined reset vector address. The following sources can generate a reset:

- Power-on reset (POR) circuit
- $\overline{\text{RESET}}$ pin
- Computer operating properly (COP) watchdog
- Illegal address

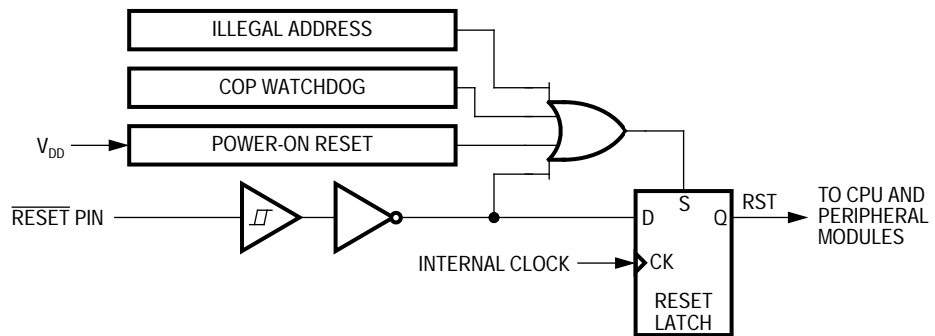


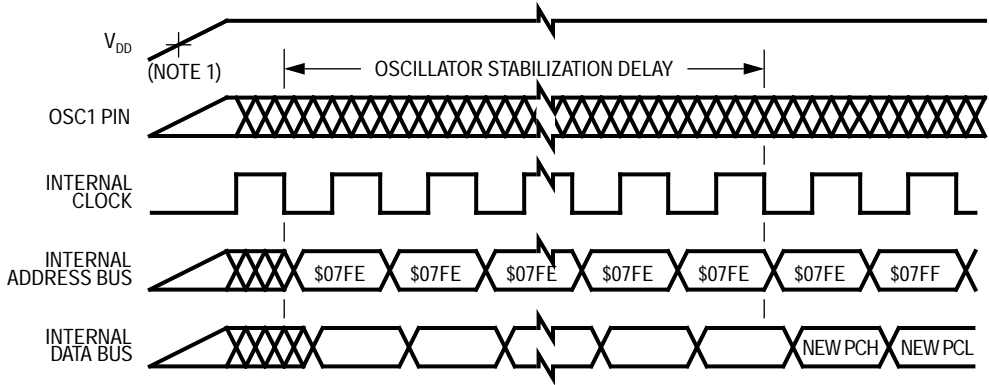
Figure 1. Reset Sources

Power-On Reset

A positive transition on the V_{DD} pin generates a power-on reset.

NOTE: *The power-on reset is strictly for power-up conditions and cannot be used to detect drops in power supply voltage.*

A $4064-t_{CYC}$ (internal clock cycle) delay after the oscillator becomes active allows the clock generator to stabilize. If any reset source is active at the end of this delay, the MCU remains in the reset condition until all reset sources are inactive.

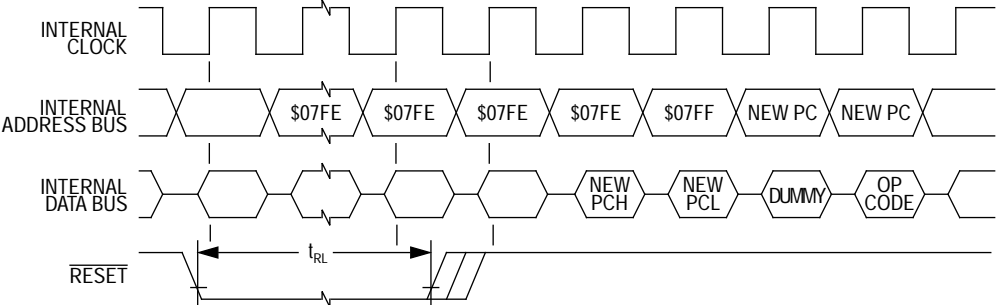


- NOTES:
 1. Power-on reset threshold is typically between 1 V and 2 V.
 2. Internal clock, internal address bus, and internal data bus are not available externally.

Figure 2. Power-On Reset Timing

External Reset

A logic 0 applied to the $\overline{\text{RESET}}$ pin for $1 \frac{1}{2} t_{\text{cyc}}$ generates an external reset. A Schmitt trigger senses the logic level at the $\overline{\text{RESET}}$ pin.



- NOTES:
 1. Internal clock, internal address bus, and internal data bus are not available externally.
 2. The next rising edge of the internal clock after the rising edge of $\overline{\text{RESET}}$ initiates the reset sequence.

Figure 3. External Reset Timing

Table 1. External Reset Timing

Characteristic	Symbol	Min	Max	Unit
$\overline{\text{RESET}}$ Pulse Width	t _{RL}	1.5	—	t _{cyc}

COP Watchdog Reset	A timeout of the COP watchdog generates a COP reset. The COP watchdog is part of a software error detection system and must be cleared periodically to start a new timeout period. To clear the COP watchdog and prevent a COP reset, write a logic 0 to bit 0 (COPC) of the COP register at location \$07F0.
Illegal Address Reset	An opcode fetch from an address not in RAM or EPROM generates a reset.

Interrupts

The following sources can generate interrupts:

- SWI instruction
- External interrupt pins
 - $\overline{\text{IRQ}}/V_{\text{PP}}$ pin
 - PA0–PA3
- Timer
 - Real-time interrupt flag (RTIF)
 - Timer overflow flag (TOF)

An interrupt temporarily stops the program sequence to process a particular event. An interrupt does not stop the operation of the instruction being executed, but takes effect when the current instruction completes its execution. Interrupt processing automatically saves the CPU registers on the stack and loads the program counter with a user-defined interrupt vector address.

Software Interrupt	The software interrupt (SWI) instruction causes a nonmaskable interrupt.
---------------------------	--

External Interrupt	An interrupt signal on the $\overline{\text{IRQ}}/V_{\text{PP}}$ pin latches an external interrupt request. When the CPU completes its current instruction, it tests the IRQ latch. If the IRQ latch is set, the CPU then tests the I bit in the condition
---------------------------	--

code register. If the I bit is clear, the CPU then begins the interrupt sequence.

The CPU clears the IRQ latch during interrupt processing, so that another interrupt signal on the $\overline{\text{IRQ}}/\text{V}_{\text{PP}}$ pin can latch another interrupt request during the interrupt service routine. As soon as the I bit is cleared during the return from interrupt, the CPU can recognize the new interrupt request. **Figure 4** shows the external interrupt logic.

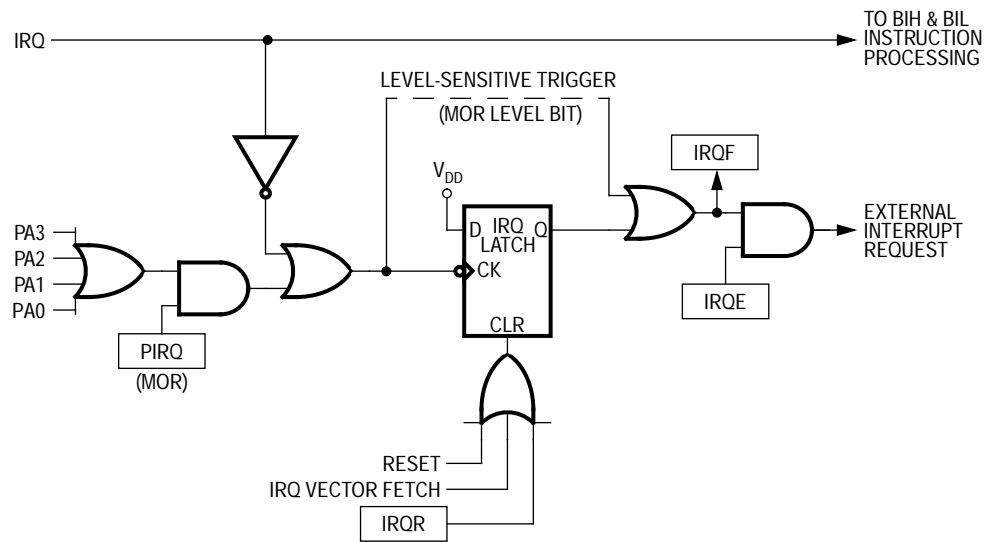


Figure 4. External Interrupt Logic

Setting the I bit in the condition code register disables external interrupts.

The port A external interrupt bit (PIRQ) in the mask option register enables pins PA0–PA3 to function as external interrupt pins.

The external interrupt sensitivity bit (LEVEL) in the mask option register controls interrupt triggering sensitivity of external interrupt pins. The $\overline{\text{IRQ}}/\text{V}_{\text{PP}}$ pin can be negative-edge triggered only or negative-edge and low-level triggered. Port A external interrupt pins can be positive-edge triggered only or both positive-edge and high-level triggered. The level-sensitive triggering option allows multiple external interrupt sources to be wire-ORed to an external interrupt pin. **Figure 5** shows the external interrupt timing.

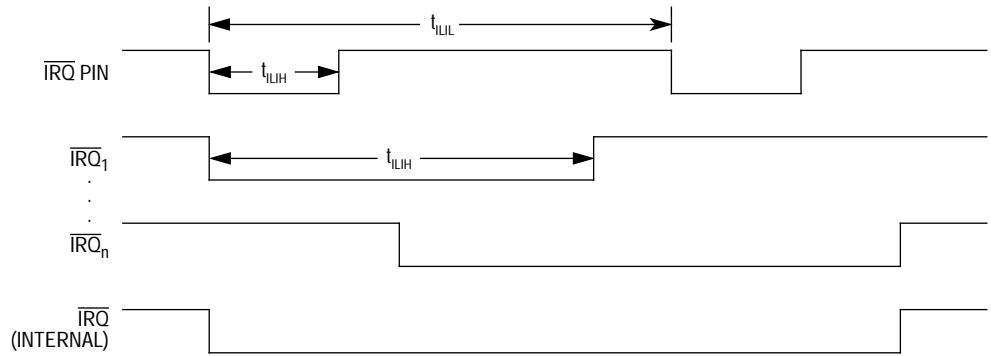


Figure 5. External Interrupt Timing

Table 2. External Interrupt Timing ($V_{DD} = 5.0\text{ Vdc}$)⁽¹⁾

Characteristic	Symbol	Min	Max	Unit
Interrupt Pulse Width Low (Edge-Triggered)	t_{LIH}	125	—	ns
Interrupt Pulse Period	t_{LIL}	Note ⁽²⁾	—	t_{cyc}

- $V_{DD} = 5.0\text{ Vdc} \pm 10\%$, $V_{SS} = 0\text{ Vdc}$, $T_A = -40\text{ }^\circ\text{C}$ to $+105\text{ }^\circ\text{C}$ unless otherwise noted.
- The minimum t_{LIL} should not be less than the number of interrupt service routine cycles plus $19 t_{cyc}$.

Table 3. External Interrupt Timing ($V_{DD} = 3.3\text{ Vdc}$)⁽¹⁾

Characteristic	Symbol	Min	Max	Unit
Interrupt Pulse Width Low (Edge-Triggered)	t_{LIH}	250	—	ns
Interrupt Pulse Period	t_{LIL}	Note ⁽²⁾	—	t_{cyc}

- $V_{DD} = 3.3\text{ Vdc} \pm 10\%$, $V_{SS} = 0\text{ Vdc}$, $T_A = -40\text{ }^\circ\text{C}$ to $+105\text{ }^\circ\text{C}$ unless otherwise noted.
- The minimum t_{LIL} should not be less than the number of interrupt service routine cycles plus $19 t_{cyc}$.

Timer Interrupts

The timer can generate the following interrupt requests:

- Real time
- Timer overflow

Setting the I bit in the condition code register disables timer interrupts.

Real-Time Interrupt

A real-time interrupt occurs if the real-time interrupt flag, RTIF, becomes set while the real-time interrupt enable bit, RTIE, is also set. RTIF and RTIE are in the timer status and control register.

Timer Overflow Interrupt

A timer overflow interrupt request occurs if the timer overflow flag, TOF, becomes set while the timer overflow interrupt enable bit, TOIE, is also set. TOF and TOIE are in the timer status and control register.

Interrupt Processing

The CPU takes the following actions to begin servicing an interrupt:

- Stores the CPU registers on the stack in the order shown in **Figure 6**
- Sets the I bit in the condition code register to prevent further interrupts
- Loads the program counter with the contents of the appropriate interrupt vector locations:
 - \$07FC and \$07FD (software interrupt vector)
 - \$07FA and \$07FB (external interrupt vector)
 - \$07F8 and \$07F9 (timer interrupt vector)

The return from interrupt (RTI) instruction causes the CPU to recover the CPU registers from the stack as shown in **Figure 6**.

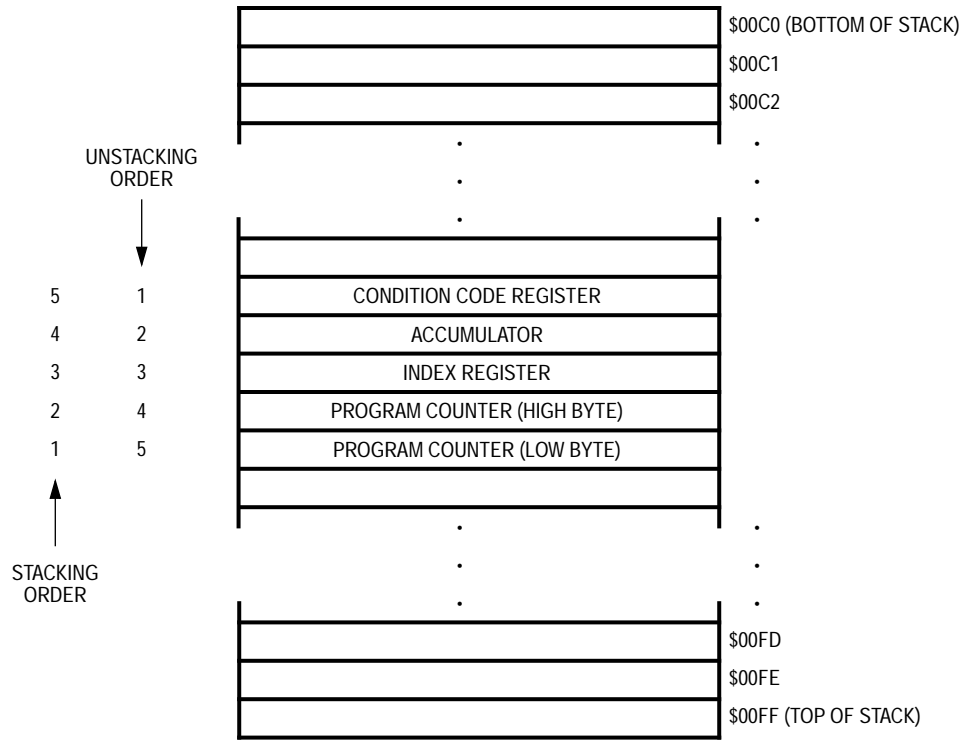


Figure 6. Interrupt Stacking Order

Table 4. Reset/Interrupt Vector Addresses

Function	Source	Local Mask	Global Mask	Priority (1 = Highest)	Vector Address
Reset	Power-On $\overline{\text{RESET}}$ Pin COP Watchdog ⁽¹⁾ Illegal Address	None	None	1	\$07FE–\$07FF
Software Interrupt (SWI)	User Code	None	None	Same Priority as Instruction	\$07FC–\$07FD
External Interrupt	$\overline{\text{IRQ}}/\text{V}_{\text{PP}}$ Pin PA3 Pin PA2 Pin PA1 Pin PA0 Pin	IRQE	I Bit	2	\$07FA–\$07FB
Timer Interrupts	RTIF Bit TOF Bit	RTIE Bit TOIE Bit	I Bit	3	\$07F8–\$07F9

1. The COP watchdog is programmable in the mask option register.

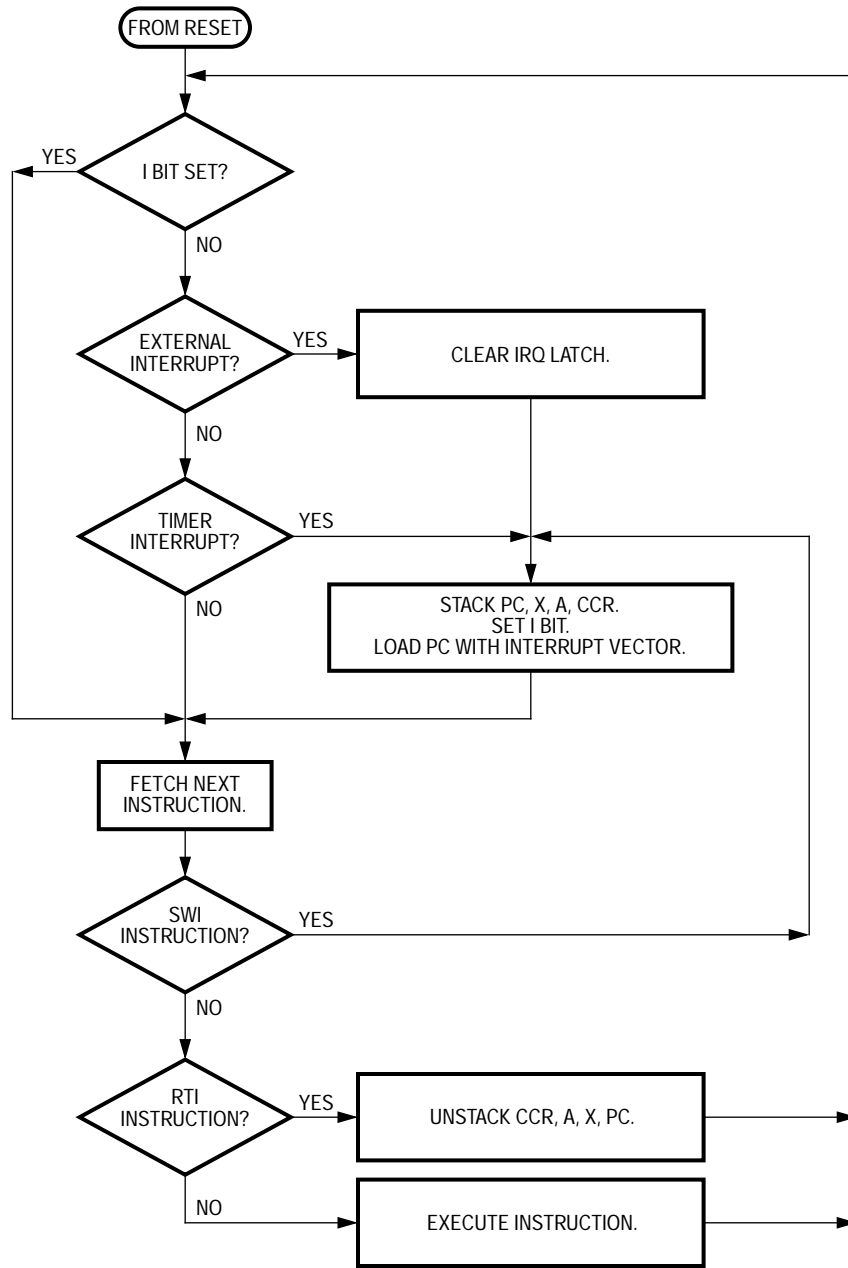


Figure 7. Interrupt Flowchart

Contents

Introduction	64
Exiting Stop and Wait Modes	64
Exiting Stop Mode	64
Exiting Wait Mode	65
Effects of Stop and Wait Modes	65
Clock Generation	65
STOP	65
WAIT	66
CPU	66
STOP	66
WAIT	66
COP Watchdog	66
STOP	66
WAIT	67
Timer	67
STOP	67
WAIT	68
EPROM/OTEPROM	68
STOP	68
WAIT	68
Data-Retention Mode	68
Timing	69

Introduction

The MCU can enter the following low-power standby modes:

- Stop mode — The STOP instruction puts the MCU in its lowest power-consumption mode.
- Wait mode — The WAIT instruction puts the MCU in an intermediate power-consumption mode.
- Halt mode — Halt mode is identical to wait mode, except that an oscillator stabilization delay of 1 to 4064 internal clock cycles occurs when the MCU exits halt mode. The stop-to-wait conversion bit, SWAIT, in the mask option register, enables halt mode.

Enabling halt mode prevents the computer operating properly (COP) watchdog from being inadvertently turned off by a STOP instruction.

- Data-retention mode — In data-retention mode, the MCU retains RAM contents and CPU register contents at V_{DD} voltages as low as 2.0 Vdc. The data-retention feature allows the MCU to remain in a low power-consumption state during which it retains data, but the CPU cannot execute instructions.

Exiting Stop and Wait Modes

The following events bring the MCU out of stop mode and load the program counter with the reset vector or with an interrupt vector:

Exiting Stop Mode

- External reset — A logic 0 on the $\overline{\text{RESET}}$ pin resets the MCU, starts the CPU clock, and loads the program counter with the contents of locations \$07FE and \$07FF.
- External interrupt — A high-to-low transition on the $\overline{\text{IRQ}}/V_{PP}$ pin or a low-to-high transition on an enabled port A external interrupt pin starts the CPU clock and loads the program counter with the contents of locations \$07FA and \$07FB.

Exiting Wait Mode

- External reset — A logic 0 on the $\overline{\text{RESET}}$ pin resets the MCU, starts the CPU clock, and loads the program counter with the contents of locations \$07FE and \$07FF.
- External interrupt — A high-to-low transition on the $\overline{\text{IRQ}}/V_{\text{PP}}$ pin or a low-to-high transition on an enabled port A external interrupt pin starts the CPU clock and loads the program counter with the contents of locations \$07FA and \$07FB.
- COP watchdog reset — A timeout of the COP watchdog resets the MCU, starts the CPU clock, and loads the program counter with the contents of locations \$07FE and \$07FF. Software can enable timer interrupts so that the MCU periodically can exit wait mode to reset the COP watchdog.
- Timer interrupt — Real-time interrupt requests and timer overflow interrupt requests start the MCU clock and load the program counter with the contents of locations \$07F8 and \$07F9.

Effects of Stop and Wait Modes

The STOP and WAIT instructions have the following effects on MCU modules.

Clock Generation Effects of STOP and WAIT:

STOP

The STOP instruction disables the internal oscillator, stopping the CPU clock and all peripheral clocks.

After exiting stop mode, the CPU clock and all enabled peripheral clocks begin running after the oscillator stabilization delay.

NOTE: *The oscillator stabilization delay holds the MCU in reset for the first 4064 internal clock cycles.*

<i>WAIT</i>	<p>The WAIT instruction disables the CPU clock.</p> <p>After exiting wait mode, the CPU clock and all enabled peripheral clocks immediately begin running.</p>
CPU	Effects of STOP and WAIT:
<i>STOP</i>	<p>The STOP instruction:</p> <ul style="list-style-type: none">• Clears the interrupt mask (I bit) in the condition code register, enabling external interrupts• Disables the CPU clock <p>After exiting stop mode, the CPU clock begins running after the oscillator stabilization delay.</p> <p>After exit from stop mode by external interrupt, the I bit remains clear.</p> <p>After exit from stop mode by reset, the I bit is set.</p>
<i>WAIT</i>	<p>The WAIT instruction:</p> <ul style="list-style-type: none">• Clears the interrupt mask (I bit) in the condition code register, enabling interrupts• Disables the CPU clock <p>After exit from wait mode by interrupt, the I bit remains clear.</p> <p>After exit from wait mode by reset, the I bit is set.</p>
COP Watchdog	Effects of STOP and WAIT:
<i>STOP</i>	<p>The STOP instruction:</p> <ul style="list-style-type: none">• Clears the COP watchdog counter• Disables the COP watchdog clock

NOTE: *To prevent the STOP instruction from disabling the COP watchdog, program the stop-to-wait conversion bit (SWAIT) in the mask option register to logic 1.*

After exit from stop mode by external interrupt, the COP watchdog counter immediately begins counting from \$0000 and continues counting throughout the oscillator stabilization delay.

NOTE: *Immediately after exiting stop mode by external interrupt, service the COP to ensure a full COP timeout period.*

After exit from stop mode by reset:

- The COP watchdog counter immediately begins counting from \$0000.
- The COP watchdog counter is cleared at the end of the oscillator stabilization delay and begins counting from \$0000 again.

WAIT

The WAIT instruction has no effect on the COP watchdog.

NOTE: *To prevent a COP timeout during wait mode, exit wait mode periodically to service the COP.*

Timer

Effects of STOP and WAIT:

STOP

- Clears the RTIE, TOFE, RTIF, and TOF bits in the timer status and control register, disabling timer interrupt requests and removing any pending timer interrupt requests
- Disables the clock to the timer

After exiting stop mode by external interrupt, the timer immediately resumes counting from the last value before the STOP instruction and continues counting throughout the oscillator stabilization delay.

After exiting stop mode by reset and after the oscillator stabilization delay, the timer resumes operation from its reset state.

<i>WAIT</i>	The WAIT instruction has no effect on the timer.
EPROM/OTPROM	Effects of STOP and WAIT:
<i>STOP</i>	The STOP instruction during EPROM programming clears the EPGM bit in the EPROM programming register, removing the programming voltage from the EPROM.
<i>WAIT</i>	The WAIT instruction has no effect on EPROM/OTPROM operation.

Data-Retention Mode

In data-retention mode, the MCU retains RAM contents and CPU register contents at V_{DD} voltages as low as 2.0 Vdc. The data-retention feature allows the MCU to remain in a low power-consumption state during which it retains data, but the CPU cannot execute instructions.

To put the MCU in data-retention mode:

1. Drive the $\overline{\text{RESET}}$ pin to logic 0.
2. Lower the V_{DD} voltage. The $\overline{\text{RESET}}$ pin must remain low continuously during data-retention mode.

To take the MCU out of data-retention mode:

1. Return V_{DD} to normal operating voltage.
2. Return the $\overline{\text{RESET}}$ pin to logic 1.

Timing

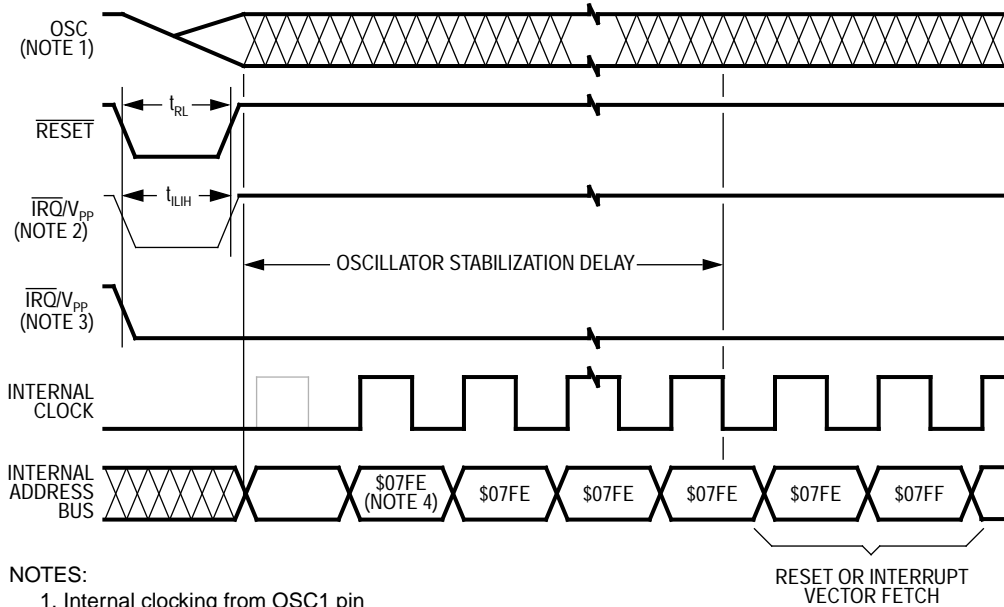


Figure 1. Stop Mode Recovery Timing

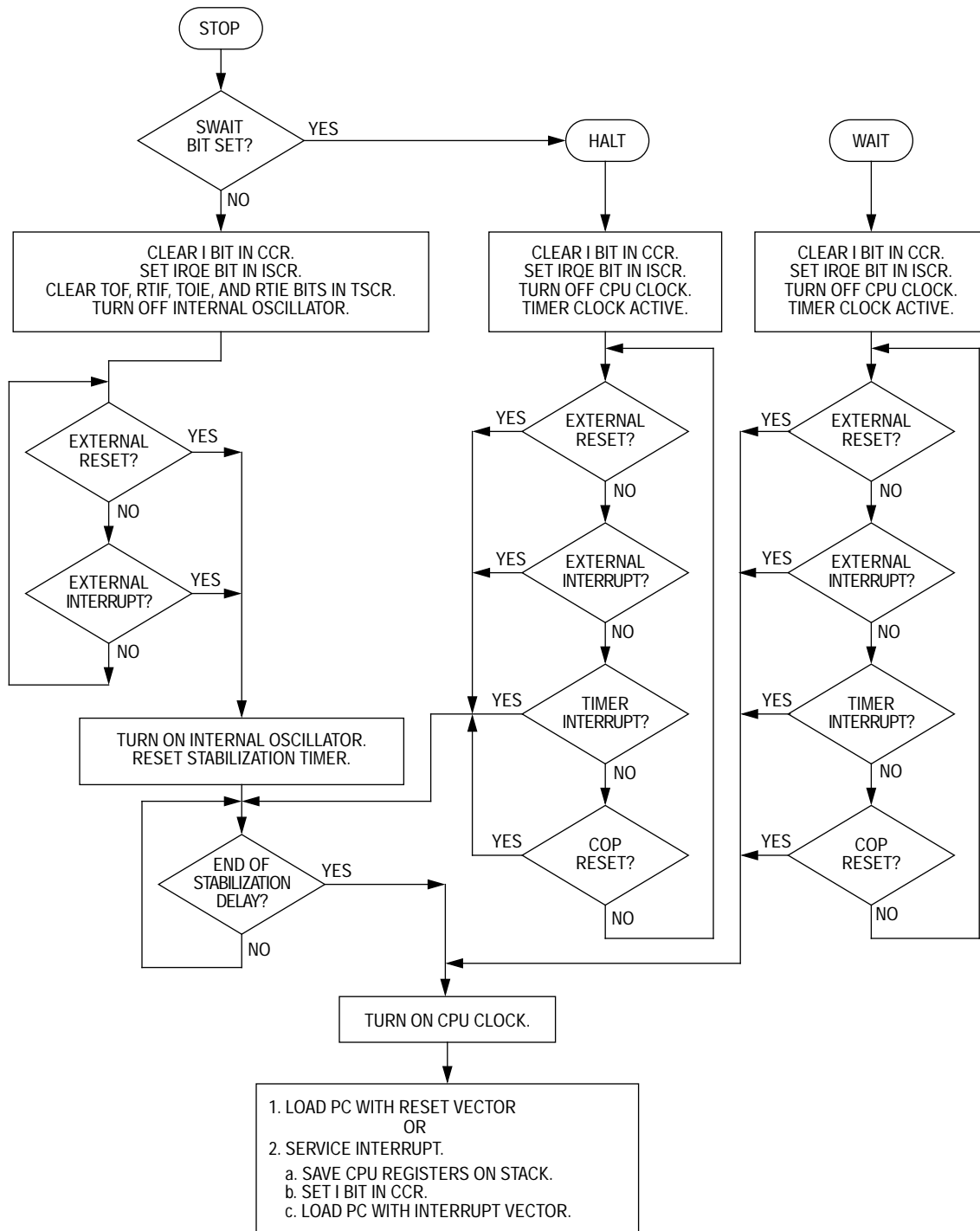


Figure 2. STOP/HALT/WAIT Flowchart

Contents

Introduction	71
Port A	73
Port A Data Register	73
Data Direction Register A	73
Pulldown Register A	75
Port A LED Drive Capability	76
Port A I/O Pin Interrupts	76
Port B	76
Port B Data Register	76
Data Direction Register B	77
Pulldown Register B	78
I/O Port Electrical Characteristics	79

Introduction

Fourteen bidirectional pins form one 8-bit input/output (I/O) port and one 6-bit I/O port. All the bidirectional port pins are programmable as inputs or outputs.

NOTE: *Connect any unused I/O pins to an appropriate logic level, either V_{DD} or V_{SS} . Although the I/O ports do not require termination for proper operation, termination reduces excess current consumption and the possibility of electrostatic damage.*

Parallel I/O Ports

Addr.	Register Name:	Bit 7	6	5	4	3	2	1	Bit 0
\$0000	Port A Data Register (PORTA)	Read: PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
		Write:							
		Reset:	Unaffected by Reset						
\$0001	Port B Data Register (PORTB)	Read: 0	0	PB5	PB4	PB3	PB2	PB1	PB0
		Write:							
		Reset:	Unaffected by Reset						
\$0004	Data Direction Register A (DDRA)	Read: DDRA7	DDRA6	DDRA5	DDRA4	DDRA3	DDRA2	DDRA1	DDRA0
		Write:							
		Reset:	0	0	0	0	0	0	0
\$0005	Data Direction Register B (DDRB)	Read: 0	0	DDRB5	DDRB4	DDRB3	DDRB2	DDRB1	DDRB0
		Write:							
		Reset:	0	0	0	0	0	0	0
\$0010	Port A Pulldown Register (PDRA)	Read:							
		Write:	PDIA7	PDIA6	PDIA5	PDIA4	PDIA3	PDIA2	PDIA1
		Reset:	0	0	0	0	0	0	0
\$0011	Port B Pulldown Register (PDRB)	Read:							
		Write:			PDIB5	PDIB4	PDIB3	PDIB2	PDIB1
		Reset:			0	0	0	0	0

= Unimplemented

Figure 1. Parallel I/O Port Register Summary

Port A

Port A is an 8-bit bidirectional port.

Port A Data Register

The port A data register contains a latch for each port A pin.

Address: \$0000

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
Write:	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
Reset:	Unaffected by Reset							

Figure 2. Port A Data Register (PORTA)

PA[7:0] — Port A Data Bits

These read/write bits are software programmable. Data direction of each port A pin is under the control of the corresponding bit in data direction register A. Reset has no effect on port A data.

Data Direction Register A

Data direction register A determines whether each port A pin is an input or an output.

Address: \$0004

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	DDRA7	DDRA6	DDRA5	DDRA4	DDRA3	DDRA2	DDRA1	DDRA0
Write:	DDRA7	DDRA6	DDRA5	DDRA4	DDRA3	DDRA2	DDRA1	DDRA0
Reset:	0	0	0	0	0	0	0	0

Figure 3. Data Direction Register A (DDRA)

DDRA[7:0] — Data Direction Register A Bits

These read/write bits control port A data direction. Reset clears DDRA[7:0], configuring all port A pins as inputs.

1 = Corresponding port A pin configured as output

0 = Corresponding port A pin configured as input

NOTE: *Avoid glitches on port A pins by writing to the port A data register before changing data direction register A bits from 0 to 1.*

Figure 4 shows the I/O logic of port A.

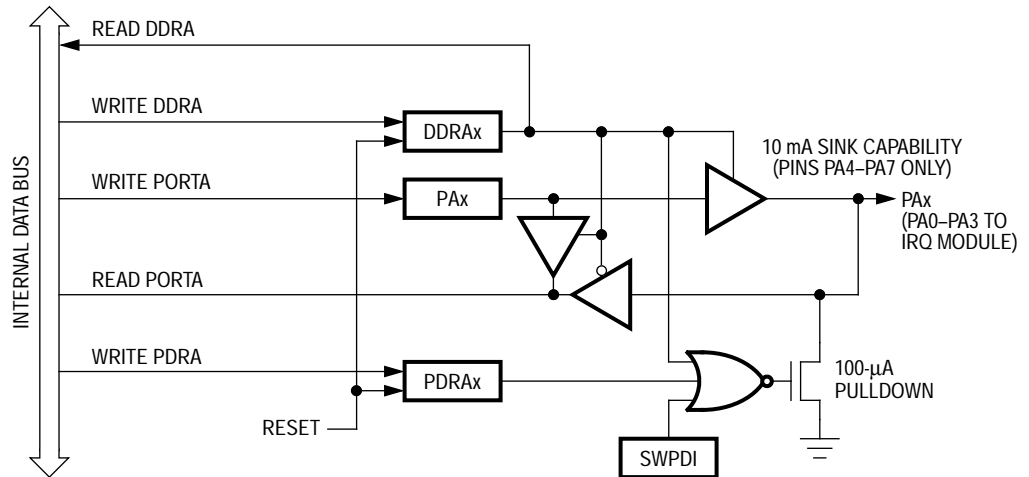


Figure 4. Port A I/O Circuitry

Writing a logic 1 to a DDRA bit enables the output buffer for the corresponding port A pin; a logic 0 disables the output buffer.

When bit DDRAx is a logic 1, reading address \$0000 reads the PAX data latch. When bit DDRAx is a logic 0, reading address \$0000 reads the voltage level on the pin. The data latch can always be written, regardless of the state of its data direction bit. **Table 1** summarizes the operation of the port A pins.

Table 1. Port A Pin Operation

Data Direction Bit	I/O Pin Mode	Accesses to Data Bit	
		Read	Write
0	Input, high-impedance	Pin	Latch ⁽¹⁾
1	Output	Latch	Latch

1. Writing affects data register but does not affect input.

Pulldown Register A

Pulldown register A inhibits the pulldown devices on port A pins programmed as inputs.

NOTE: *If the SWPDI bit in the mask option register is programmed to logic 1, reset initializes all port A pins as inputs with enabled pulldown devices.*

Address: \$0010

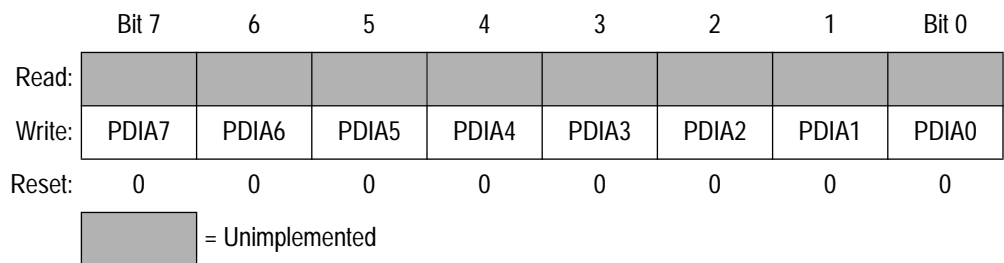


Figure 5. Pulldown Register A (PDRA)

PDIA[7:0] — Pulldown Inhibit A Bits

PDIA[7:0] disable the port A pulldown devices. Reset clears PDIA[7:0].

- 1 = Corresponding port A pulldown device disabled
- 0 = Corresponding port A pulldown device not disabled

Port A LED Drive Capability The outputs for the upper four bits of port A (PA4–PA7) can drive light-emitting diodes (LEDs). PA4–PA7 can sink approximately 10 mA of current to V_{SS} .

Port A I/O Pin Interrupts If the PIRQ bit in the mask option register is programmed to logic 1, PA0–PA3 pins function as external interrupt pins. (See [External Interrupt Module](#).)

Port B

Port B is a 6-bit bidirectional port.

Port B Data Register The port B data register contains a latch for each port B pin.

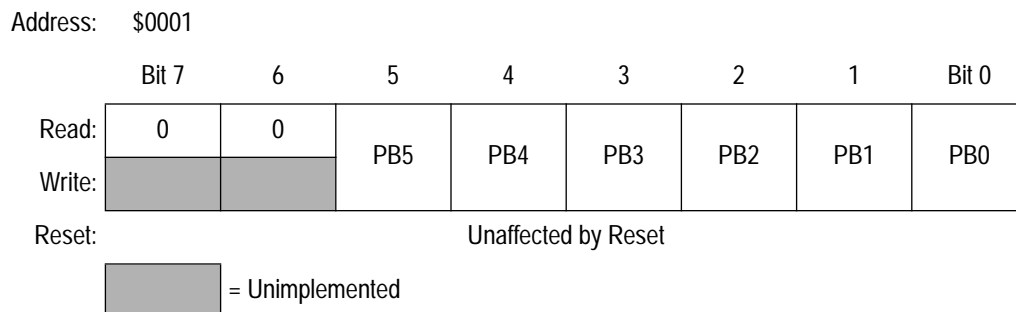


Figure 6. Port B Data Register (PORTB)

PB[5:0] — Port B Data Bits

These read/write bits are software programmable. Data direction of each port B pin is under the control of the corresponding bit in data direction register B. Reset has no effect on port B data.

Data Direction Register B

Data direction register B determines whether each port B pin is an input or an output.

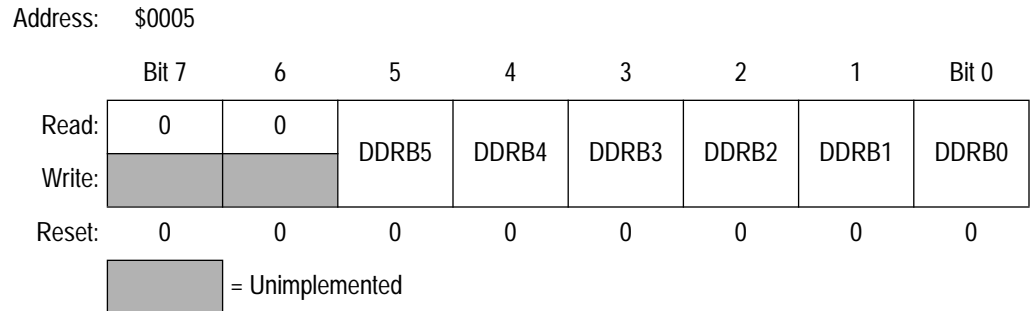


Figure 7. Data Direction Register B (DDRB)

DDRB[5:0] — Data Direction Register B Bits

These read/write bits control port B data direction. Reset clears DDRB[5:0], configuring all port B pins as inputs.

- 1 = Corresponding port B pin configured as output
- 0 = Corresponding port B pin configured as input

NOTE: Avoid glitches on port B pins by writing to the port B data register before changing data direction register B bits from 0 to 1.

Figure 8 shows the I/O logic of port B.

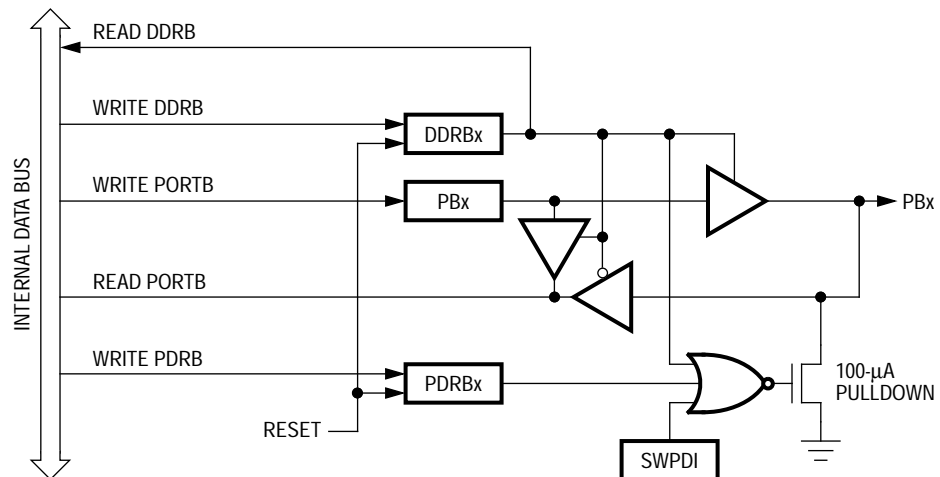


Figure 8. Port B I/O Circuitry

Writing a logic 1 to a DDRB bit enables the output buffer for the corresponding port B pin; a logic 0 disables the output buffer.

When bit DDRBx is a logic 1, reading address \$0001 reads the PBx data latch. When bit DDRBx is a logic 0, reading address \$0001 reads the voltage level on the pin. The data latch can always be written, regardless of the state of its data direction bit. **Table 2** summarizes the operation of the port B pins.

Table 2. Port B Pin Operation

Data Direction Bit	I/O Pin Mode	Accesses to Data Bit	
		Read	Write
0	Input, high-impedance	Pin	Latch ⁽¹⁾
1	Output	Latch	Latch

1. Writing affects data register, but does not affect input.

Pulldown Register B

Pulldown register B inhibits the pulldown devices on port B pins programmed as inputs.

NOTE: *If the SWPDI bit in the mask option register is programmed to logic 1, reset initializes all port B pins as inputs with enabled pulldown devices.*

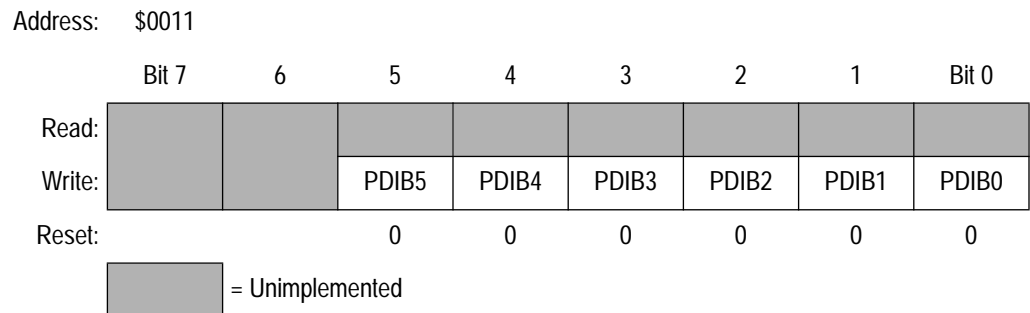


Figure 9. Pulldown Register B (PDRB)

PDIB[7:0] — Pulldown Inhibit B Bits

PDIB[7:0] disable the port B pulldown devices. Reset clears PDIB[7:0].

1 = Corresponding port B pulldown device disabled

0 = Corresponding port B pulldown device not disabled

I/O Port Electrical Characteristics

Table 3. I/O Port DC Electrical Characteristics ($V_{DD} = 5\text{ V}$)⁽¹⁾

Characteristic	Symbol	Min	Typ ⁽²⁾	Max	Unit
Current Drain Per Pin Excluding PA4–PA7	I	—	25	—	mA
Output High Voltage ($I_{Load} = -0.8\text{ mA}$) PA0–PA7, PB0–PB5	V_{OH}	V_{DD} -0.8	—	—	V
Output Low Voltage ($I_{Load} = 1.6\text{ mA}$) PA0–PA3, PB0–PB5 ($I_{Load} = 10.0\text{ mA}$) PA4–PA7	V_{OL} V_{OL}	— —	— —	0.4 0.4	V
Input High Voltage PA0–PA7, PB0–PB5	V_{IH}	0.7 x V_{DD}	—	V_{DD}	V
Input Low Voltage PA0–PA7, PB0–PB5	V_{IL}	V_{SS}	—	0.2 x V_{DD}	V
I/O Ports Hi-Z Leakage Current PA0–PA7, PB0–PB5 (Without Individual Pulldown Activated)	I_{IL}	—	0.2	±1	μA
Input Pulldown Current PA0–PA7, PB0–PB5 (With Individual Pulldown Activated)	I_{IL}	35	80	200	μA

- $V_{DD} = 5.0\text{ Vdc} \pm 10\%$, $V_{SS} = 0\text{ Vdc}$, $T_A = -40\text{ °C}$ to $+105\text{ °C}$ unless otherwise noted.
- Typical values reflect average measurements at midpoint of voltage range, 25 °C .

Table 4. I/O Port DC Electrical Characteristics ($V_{DD} = 3.3 \text{ V}$)⁽¹⁾

Characteristic	Symbol	Min	Typ ⁽²⁾	Max	Unit
Current Drain Per Pin Excluding PA4–PA7	I	—	25	—	mA
Output High Voltage ($I_{Load} = -0.2 \text{ mA}$) PA0–PA7, PB0–PB5	V_{OH}	V_{DD} -0.3	—	—	V
Output Low Voltage ($I_{Load} = 0.4 \text{ mA}$) PA0–PA3, PB0–PB5 ($I_{Load} = 5.0 \text{ mA}$) PA4–PA7	V_{OL}	— —	— —	0.3 0.3	V
Input High Voltage PA0–PA7, PB0–PB5	V_{IH}	$0.7 \times$ V_{DD}	—	V_{DD}	V
Input Low Voltage PA0–PA7, PB0–PB5	V_{IL}	V_{SS}	—	$0.2 \times$ V_{DD}	V
I/O Ports Hi-Z Leakage Current PA0–PA7, PB0–PB5 (Without Individual Pulldown Activated)	I_{IL}	—	0.1	± 1	μA
Input Pulldown Current PA0–PA7, PB0–PB5 (With Individual Pulldown Activated)	I_{IL}	12	30	100	μA

- $V_{DD} = 3.3 \text{ Vdc} \pm 10\%$, $V_{SS} = 0 \text{ Vdc}$, $T_A = -40 \text{ }^\circ\text{C}$ to $+105 \text{ }^\circ\text{C}$ unless otherwise noted.
- Typical values reflect average measurements at midpoint of voltage range, $25 \text{ }^\circ\text{C}$.

Computer Operating Properly Module

Contents

Features	81
Introduction	81
Operation	82
COP Watchdog Timeout	82
COP Watchdog Timeout Period	82
Clearing the COP Watchdog	82
Interrupts	83
COP Register	83
Low-Power Modes	84
Stop Mode	84
Wait Mode	84

Features

- Protection from Runaway Software
- Wait and Halt Mode Operation

Introduction

The computer operating properly (COP) watchdog resets the MCU in case of software failure. Software that is operating properly periodically services the COP watchdog and prevents COP reset. The COP watchdog function is programmable by the COPEN bit in the mask option register.

Operation

COP Watchdog Timeout Four counter stages at the end of the timer make up the COP watchdog. The COP resets the MCU if the timeout period occurs before the COP watchdog timer is cleared by application software and the $\overline{\text{IRQ}}/V_{\text{PP}}$ pin voltage is between V_{SS} and V_{DD} . Periodically clearing the counter starts a new timeout period and prevents COP reset. A COP watchdog timeout indicates that the software is not executing instructions in the correct sequence.

NOTE: *The internal clock drives the COP watchdog. Therefore, the COP watchdog cannot generate a reset for errors that cause the internal clock to stop.*

NOTE: *The COP watchdog depends on a power supply voltage at or above a minimum specification and is not guaranteed to protect against brownout.*

COP Watchdog Timeout Period The COP watchdog timer function is implemented by dividing the output of the real-time interrupt circuit (RTI) by eight. The RTI select bits in the timer status and control register control RTI output, and the selected output drives the COP watchdog. (See timer status and control register in [Multifunction Timer Module](#).)

Note that the minimum COP timeout period is seven times the RTI period. The COP is cleared asynchronously with the value in the RTI divider; hence, the COP timeout period will vary between 7x and 8x the RTI period.

Clearing the COP Watchdog To clear the COP watchdog and prevent a COP reset, write a logic 0 to bit 0 (COPC) of the COP register at location \$07F0 (see [Figure 1](#)). Clearing the COP bit disables the COP watchdog timer regardless of the $\overline{\text{IRQ}}/V_{\text{PP}}$ pin voltage.

If the main program executes within the COP timeout period, the clearing routine should be executed only once. If the main program takes longer than the COP timeout period, the clearing routine must be executed more than once.

NOTE: *Place the clearing routine in the main program and not in an interrupt routine. Clearing the COP watchdog in an interrupt routine might prevent COP watchdog timeouts even though the main program is not operating properly.*

Interrupts

The COP watchdog does not generate interrupts.

COP Register

The COP register (COPR) is a write-only register that returns the contents of EPROM location \$07F0 when read.

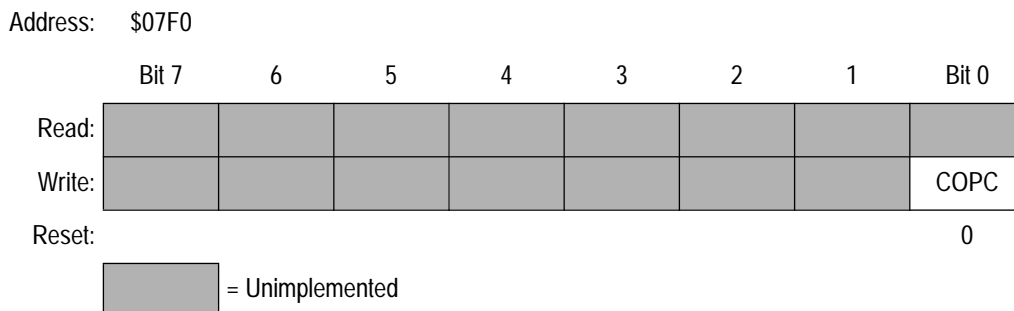


Figure 1. COP Register

COPC — COP Clear

This write-only bit resets the COP watchdog. Reading address \$07F0 returns undefined results.

Low-Power Modes

The STOP and WAIT instructions have the following effects on the COP watchdog.

Stop Mode The STOP instruction clears the COP watchdog counter and disables the clock to the COP watchdog.

NOTE: *To prevent the STOP instruction from disabling the COP watchdog, program the stop-to-wait conversion bit (SWAIT) in the mask option register to logic 1.*

Upon exit from stop mode by external reset:

- The counter begins counting from \$0000.
- The counter is cleared again after the oscillator stabilization delay and begins counting from \$0000 again.

Upon exit from stop mode by external interrupt:

- The counter begins counting from \$0000.
- The counter is not cleared again after the oscillator stabilization delay and continues counting throughout the oscillator stabilization delay.

NOTE: *Immediately after exiting stop mode by external interrupt, service the COP to ensure a full COP timeout period.*

Wait Mode The WAIT instruction has no effect on the COP watchdog.

NOTE: *To prevent a COP timeout during wait mode, exit wait mode periodically to service the COP.*

External Interrupt Module

IRQ

Contents

Features	85
Introduction	86
Operation	86
$\overline{\text{IRQ}}/V_{\text{PP}}$ Pin	89
Optional External Interrupts	90
IRQ Status and Control Register	91
Timing	92

Features

- Dedicated External Interrupt Pin ($\overline{\text{IRQ}}/V_{\text{PP}}$)
- Selectable Interrupt on Four Input/Output (I/O) Pins (PA0–PA3)
- Programmable Edge-Only or Edge- and Level-Interrupt Sensitivity

Introduction

The external interrupt (IRQ) module provides asynchronous external interrupts to the CPU. The following sources can generate external interrupts:

- $\overline{\text{IRQ}}/V_{\text{PP}}$ pin
- PA0–PA3 pins

Operation

The interrupt request/programming voltage pin ($\overline{\text{IRQ}}/V_{\text{PP}}$) and port A pins 0–3 (PA0–PA3) provide external interrupts. The PIRQ bit in the mask option register (MOR) enables PA0–PA3 as IRQ interrupt sources, which are combined into a single OR'ing function to be latched by the IRQ latch. [Figure 1](#) shows the structure of the IRQ module.

After completing its current instruction, the CPU tests the IRQ latch. If the IRQ latch is set, the CPU then tests the I bit in the condition code register and the IRQE bit in the IRQ status and control register. If the I bit is clear and the IRQE bit is set, the CPU then begins the interrupt sequence. This interrupt is serviced by the interrupt service routine located at \$07FA and \$07FB.

The CPU clears the IRQ latch while it fetches the interrupt vector, so that another external interrupt request can be latched during the interrupt service routine. As soon as the I bit is cleared during the return from interrupt, the CPU can recognize the new interrupt request. [Figure 3](#) shows the sequence of events caused by an interrupt.

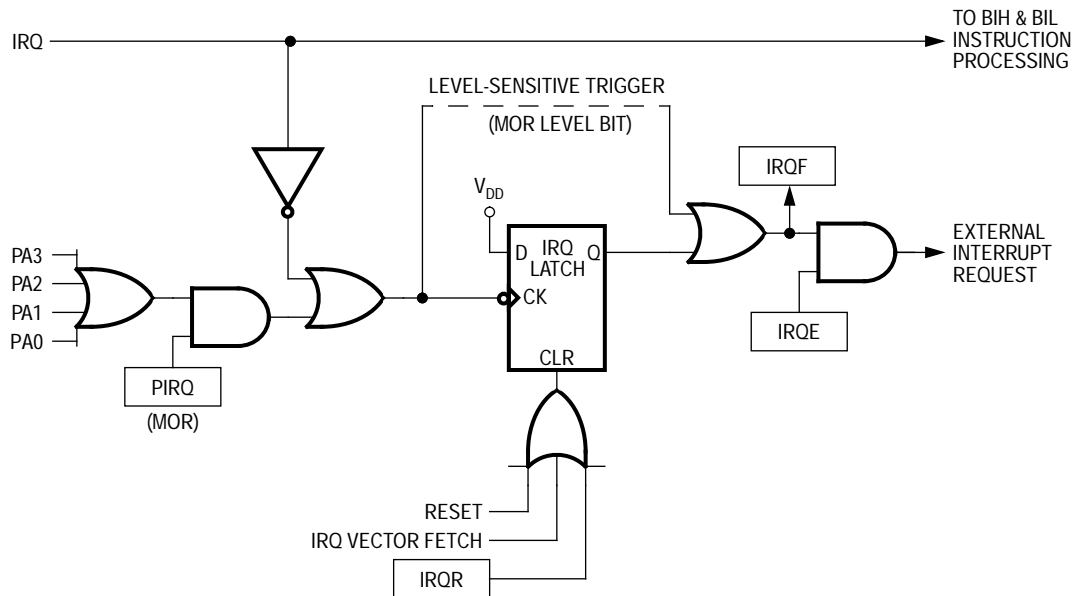


Figure 1. IRQ Module Block Diagram

Register Name	Bit 7	6	5	4	3	2	1	Bit 0
Read:	IRQE	0	0	0	IRQF	0	0	0
Write:	IRQE			R			IRQR	
Reset:	1	0	0	0	0	0	0	0

= Unimplemented R = Reserved

Figure 2. IRQ Module I/O Register Summary

Table 1. I/O Register Address Summary

Register:	ISCR
Address:	\$000A

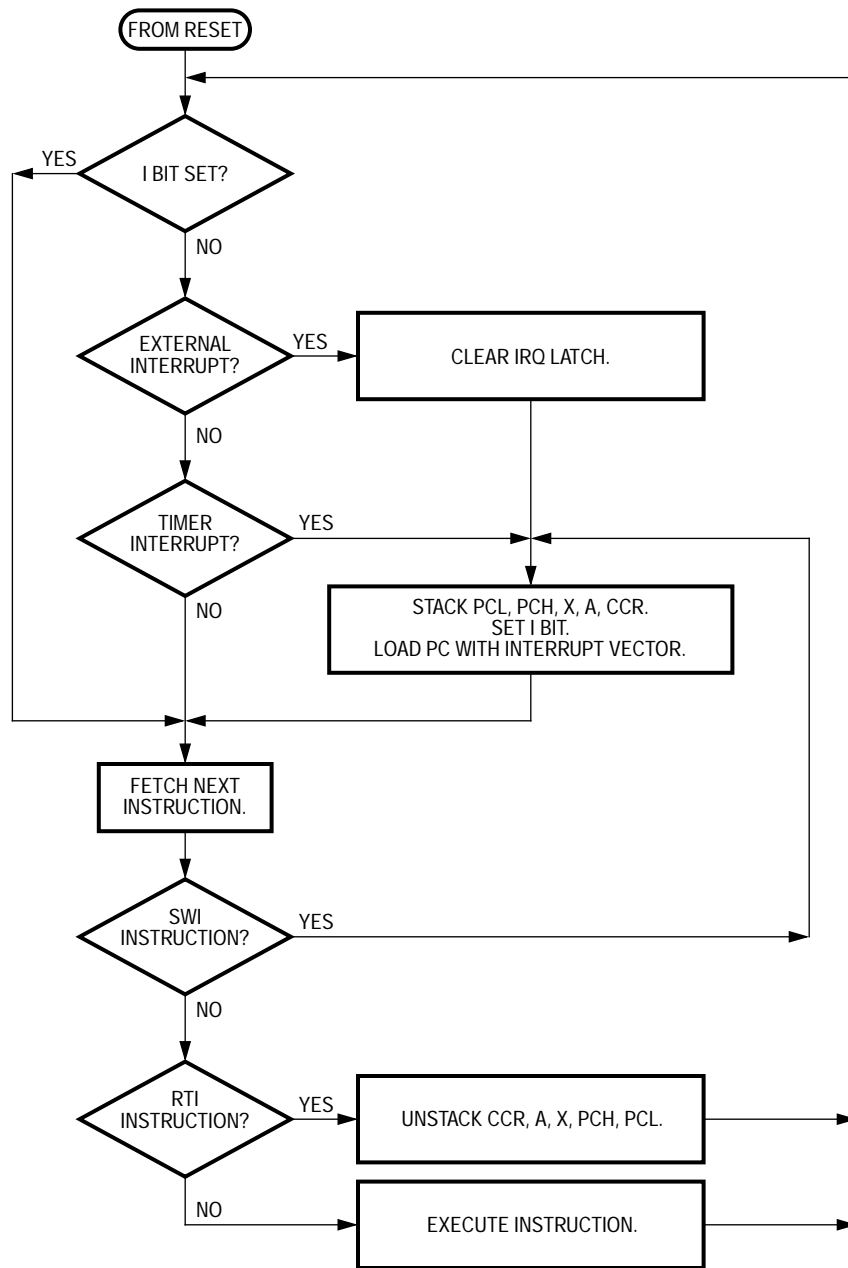


Figure 3. Interrupt Flowchart

$\overline{\text{IRQ}}/V_{\text{PP}}$ Pin

An interrupt signal on the $\overline{\text{IRQ}}/V_{\text{PP}}$ pin latches an external interrupt request. The LEVEL bit in the mask option register provides negative edge-sensitive triggering or both negative edge-sensitive and low level-sensitive triggering for the interrupt function.

If edge- and level-sensitive triggering is selected, a falling edge or a low level on the $\overline{\text{IRQ}}/V_{\text{PP}}$ pin latches an external interrupt request. Edge- and level-sensitive triggering allows the use of multiple wired-OR external interrupt sources. An external interrupt request is latched as long as any source is holding the $\overline{\text{IRQ}}/V_{\text{PP}}$ pin low.

If level-sensitive triggering is selected, the $\overline{\text{IRQ}}/V_{\text{PP}}$ input requires an external resistor to V_{DD} for wired-OR operation. If the $\overline{\text{IRQ}}/V_{\text{PP}}$ pin is not used, it must be tied to the V_{DD} supply.

If edge-sensitive-only triggering is selected, a falling edge on the $\overline{\text{IRQ}}/V_{\text{PP}}$ pin latches an external interrupt request. A subsequent external interrupt request can be latched only after the voltage level on the $\overline{\text{IRQ}}/V_{\text{PP}}$ pin returns to logic 1 and then falls again to logic 0.

The $\overline{\text{IRQ}}/V_{\text{PP}}$ pin contains an internal Schmitt trigger as part of its input to improve noise immunity. The voltage on this pin can affect the mode of operation and should not exceed V_{DD} .

Optional External Interrupts

The inputs for the lower four bits of port A (PA0–PA3) can be connected to the IRQ pin input of the CPU if enabled by the PIRQ bit in the mask option register. This capability allows keyboard scan applications where the transitions or levels on the I/O pins will behave the same as the $\overline{\text{IRQ}}/V_{PP}$ pin except for the inverted phase (logic 1, rising edge). The active state of the $\overline{\text{IRQ}}/V_{PP}$ pin is a logic 0 (falling edge).

The PA0–PA3 pins are selected as a group to function as IRQ interrupts and are enabled by the IRQE bit in the IRQ status and control register. The PA0–PA3 pins can be positive-edge triggered only or positive-edge and high-level triggered.

If edge- and level-sensitive triggering is selected, a rising edge or a high level on a PA0–PA3 pin latches an external interrupt request. Edge- and level-sensitive triggering allows the use of multiple wired-OR external interrupt sources. As long as any source is holding a PA0–PA3 pin high, an external interrupt request is latched, and the CPU continues to execute the interrupt service routine.

If edge-sensitive only triggering is selected, a rising edge on a PA0–PA3 pin latches an external interrupt request. A subsequent external interrupt request can be latched only after the voltage level of the previous interrupt signal returns to logic 0 and then rises again to logic 1.

NOTE: *The BIH and BIL instructions apply only to the level on the $\overline{\text{IRQ}}/V_{PP}$ pin itself and not to the output of the logic OR function with the PA0–PA3 pins. The state of the individual port A pins can be checked by reading the appropriate port A pins as inputs.*

NOTE: *Enabled PA0–PA3 pins cause an IRQ interrupt regardless of whether these pins are configured as inputs or outputs.*

NOTE: *The IRQ pin has an internal Schmitt trigger. The optional external interrupts (PA0–PA3) do not have internal Schmitt triggers.*

NOTE: *The interrupt mask bit (I) in the condition code register (CCR) disables all maskable interrupt requests, including external interrupt requests.*

IRQ Status and Control Register

The IRQ status and control register (ISCR) controls and monitors operation of the IRQ module. All unused bits in the ISCR read as logic 0s. The IRQF bit is cleared and the IRQE bit is set by reset.

Address: \$000A

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	IRQE	0	0	0	IRQF	0	0	0
Write:	IRQE			R			IRQR	
Reset:	1	0	0	0	0	0	0	0

= Unimplemented
 R = Reserved

Figure 4. IRQ Status and Control Register (ISCR)

IRQR — Interrupt Request Reset

This write-only bit clears the external interrupt request flag.

- 1 = Clears external interrupt and IRQF bit
- 0 = No effect on external interrupt and IRQF bit

IRQF — External Interrupt Request Flag

The external interrupt request flag is a clearable, read-only bit that is set when an external interrupt request is pending. Reset clears the IRQF bit.

- 1 = External interrupt request pending
- 0 = No external interrupt request pending

IRQE — External Interrupt Request Enable

This read/write bit enables external interrupts. Reset sets the IRQE bit.

- 1 = External interrupt requests enabled
- 0 = External interrupt requests disabled

The STOP and WAIT instructions set the IRQE bit so that an external interrupt can bring the MCU out of these low-power modes. In addition, reset sets the I bit which masks all interrupt sources.

Timing

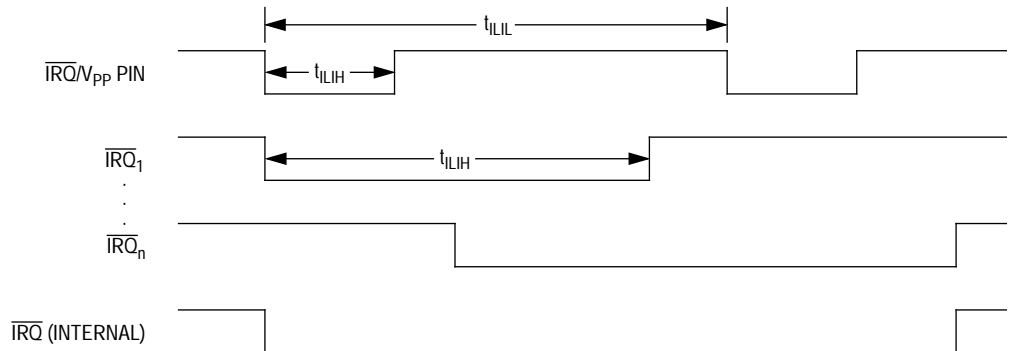


Figure 5. External Interrupt Timing

Table 2. External Interrupt Timing ($V_{DD} = 5.0 \text{ Vdc}$)⁽¹⁾

Characteristic	Symbol	Min	Max	Unit
IRQ Interrupt Pulse Width Low (Edge-Triggered)	t_{ILIH}	1.5	—	$t_{cyc}^{(2)}$
IRQ Interrupt Pulse Width (Edge- and Level-Triggered)	t_{ILIH}	1.5	Note (3)	t_{cyc}
PA0–PA3 Interrupt Pulse Width High (Edge-Triggered)	t_{ILIL}	1.5	—	t_{cyc}
PA0–PA3 Interrupt Pulse Width High (Edge- and Level-Triggered)	t_{ILIH}	1.5	Note (3)	t_{cyc}

- $V_{DD} = 5.0 \text{ Vdc} \pm 10\%$, $V_{SS} = 0 \text{ Vdc}$, $T_A = -40 \text{ }^\circ\text{C}$ to $+105 \text{ }^\circ\text{C}$ unless otherwise noted.
- $t_{cyc} = 1/f_{op}$; $f_{op} = f_{osc}/2$.
- The minimum t_{ILIL} should not be less than the number of interrupt service routine cycles plus $19 t_{cyc}$.

Table 3. External Interrupt Timing ($V_{DD} = 3.3 \text{ Vdc}$)⁽¹⁾

Characteristic	Symbol	Min	Max	Unit
IRQ Interrupt Pulse Width Low (Edge-Triggered)	t_{ILIH}	1.5	—	$t_{cyc}^{(2)}$
IRQ Interrupt Pulse Width (Edge- and Level-Triggered)	t_{ILIH}	1.5	Note (3)	t_{cyc}
PA0–PA3 Interrupt Pulse Width High (Edge-Triggered)	t_{ILIL}	1.5	—	t_{cyc}
PA0–PA3 Interrupt Pulse Width High (Edge- and Level-Triggered)	t_{ILIH}	1.5	Note (3)	t_{cyc}

- $V_{DD} = 3.3 \text{ Vdc} \pm 10\%$, $V_{SS} = 0 \text{ Vdc}$, $T_A = -40 \text{ }^\circ\text{C}$ to $+105 \text{ }^\circ\text{C}$ unless otherwise noted.
- $t_{cyc} = 1/f_{op}$; $f_{op} = f_{osc}/2$.
- The minimum t_{ILIL} should not be less than the number of interrupt service routine cycles plus $19 t_{cyc}$.

Multifunction Timer Module

Contents

Features	93
Introduction	93
Operation	95
Interrupts	96
I/O Registers	96
Timer Status and Control Register	96
Timer Counter Register	99
Low-Power Modes	100
Stop Mode	100
Wait Mode	100

Features

- Timer Overflow
- Four Selectable Interrupt Rates
- Computer Operating Properly (COP) Watchdog Timer

Introduction

The multifunction timer provides a timing reference with programmable real-time interrupt capability. [Figure 1](#) shows the timer organization.

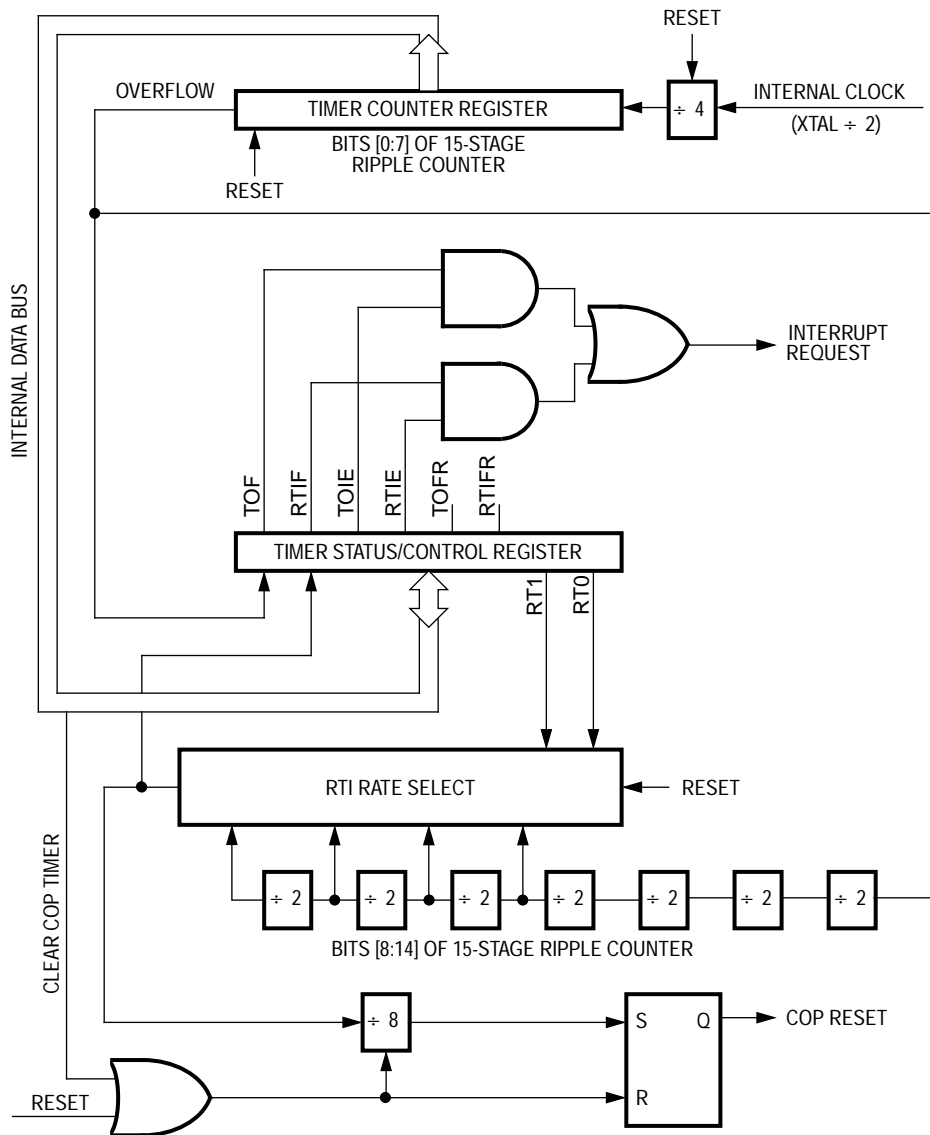


Figure 1. Multifunction Timer Block Diagram

Register Name	Bit 7	6	5	4	3	2	1	Bit 0	
Timer Status and Control Register (TSCR)	Read:	TOF	RTIF	TOIE	RTIE	0	0	RT1	RT0
	Write:					TOFR	RTIFR		
	Reset:	0	0	0	0	0	0	1	1
Timer Counter Register (TCR)	Read:	TMR7	TMR6	TMR5	TMR4	TMR3	TMR2	TMR1	TMR0
	Write:								
	Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 2. I/O Register Summary

Table 1. I/O Register Address Summary

Register:	TSCR	TCR
Address:	\$0008	\$0009

Operation

A 15-stage ripple counter, preceded by a prescaler that divides the internal clock signal by four, provides the timing reference for the timer functions. The value of the first eight timer stages can be read at any time by accessing the timer counter register at address \$0009. A timer overflow function at the eighth stage allows a timer interrupt every 1024 internal clock cycles.

The next four stages lead to the real-time interrupt (RTI) circuit. The RT1 and RT0 bits in the timer status and control register at address \$0008 allow a timer interrupt every 16,384, 32,768, 65,536, or 131,072 clock cycles. The last four stages drive the selectable COP system. (For information on the COP, refer to the [Computer Operating Properly Module](#).)

Interrupts

The following timer sources can generate interrupts:

- Timer overflow flag (TOF) — The TOF bit is set when the first eight stages of the counter roll over from \$FF to \$00. The timer overflow interrupt enable bit, TOIE, enables TOF interrupt requests.
- Real-time interrupt flag (RTIF) — The RTIF bit is set when the selected RTI output becomes active. The real-time interrupt enable bit, RTIE, enables RTIF interrupt requests.

I/O Registers

The following registers control and monitor the timer operation:

- Timer status and control register (TSCR)
- Timer counter register (TCR)

Timer Status and Control Register

The read/write timer status and control register performs the following functions:

- Flags timer interrupts
- Enables timer interrupts
- Resets timer interrupt flags
- Selects real-time interrupt rates

Address: \$0008

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	TOF	RTIF	TOIE	RTIE	0	0	RT1	RT0
Write:					TOFR	RTIFR		
Reset:	0	0	0	0	0	0	1	1


 = Unimplemented

Figure 3. Timer Status and Control Register (TSCR)

TOF — Timer Overflow Flag

This read-only flag becomes set when the first eight stages of the counter roll over from \$FF to \$00. TOF generates a timer overflow interrupt request if TOIE is also set. Clear TOF by writing a logic 1 to the TOFR bit. Writing to TOF has no effect. Reset clears TOF.

RTIF — Real-Time Interrupt Flag

This read-only flag becomes set when the selected RTI output becomes active. RTIF generates a real-time interrupt request if RTIE is also set. Clear RTIF by writing a logic 1 to the RTIFR bit. Writing to RTIF has no effect. Reset clears RTIF.

TOIE — Timer Overflow Interrupt Enable

This read/write bit enables timer overflow interrupts. Reset clears TOIE.

- 1 = Timer overflow interrupts enabled
- 0 = Timer overflow interrupts disabled

RTIE — Real-Time Interrupt Enable

This read/write bit enables real-time interrupts. Reset clears RTIE.

- 1 = Real-time interrupts enabled
- 0 = Real-time interrupts disabled

TOFR — Timer Overflow Flag Reset

Writing a logic 1 to this write-only bit clears the TOF bit. TOFR always reads as logic 0. Reset clears TOFR.

RTIFR — Real-Time Interrupt Flag Reset

Writing a logic 1 to this write-only bit clears the RTIF bit. RTIFR always reads as logic 0. Reset clears RTIFR.

RT1 and RT0 — Real-Time Interrupt Select Bits

These read/write bits select one of four real-time interrupt rates, as shown in [Table 2](#). Because the selected RTI output drives the COP watchdog, changing the real-time interrupt rate also changes the counting rate of the COP watchdog. Reset sets RT1 and RT0.

NOTE: *Changing RT1 and RT0 when a COP timeout is imminent can cause a real-time interrupt request to be missed or an additional real-time interrupt request to be generated. To prevent this occurrence, clear the COP timer before changing RT1 and RT0.*

Table 2. Real-Time Interrupt Rate Selection

RT1:RT0	RTI Rate	RTI Period ($f_{OP} = 2$ MHz)	COP Timeout Period ($-0/+1$ RTI Period)	Minimum COP Timeout Period ($f_{OP} = 2$ MHz)
00	$f_{OP} \div 2^{14}$	8.2 ms	7 x RTI Period	57.3 ms
01	$f_{OP} \div 2^{15}$	16.4 ms	7 x RTI Period	114.6 ms
10	$f_{OP} \div 2^{16}$	32.8 ms	7 x RTI Period	229.3 ms
11	$f_{OP} \div 2^{17}$	65.5 ms	7 x RTI Period	458.7 ms

Timer Counter Register

A 15-stage ripple counter is the core of the timer. The value of the first eight stages is readable at any time from the read-only timer counter register shown in [Figure 4](#).

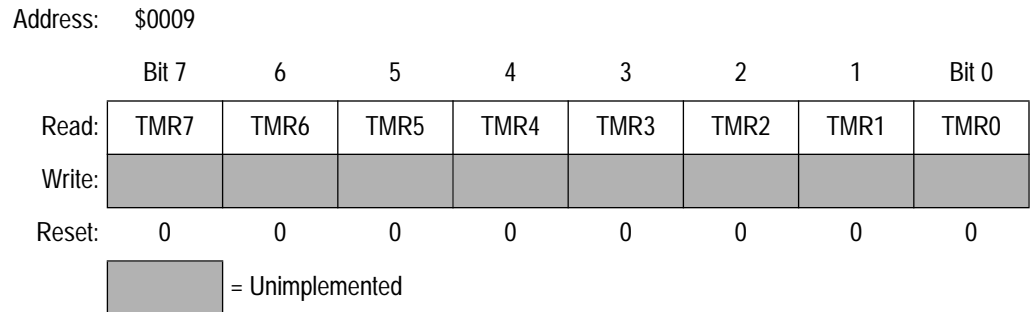


Figure 4. Timer Counter Register (TCR)

Power-on clears the entire counter chain and the internal clock begins clocking the counter. After 4064 cycles (or 16 cycles if the SOSCD bit in the mask option register is set), the power-on reset circuit is released, clearing the counter again and allowing the MCU to come out of reset.

A timer overflow function at the eighth counter stage allows a timer interrupt every 1024 internal clock cycles.

Low-Power Modes

The STOP and WAIT instructions put the MCU in low power-consumption standby states.

Stop Mode

The STOP instruction has the following effects on the timer:

- Clears the timer counter
- Clears interrupt flags (TOF and RTIF) and interrupt enable bits (TOFE and RTIE) in TSCR, removing any pending timer interrupt requests and disabling further timer interrupts.

Wait Mode

The timer remains active after a WAIT instruction. Any enabled timer interrupt request can bring the MCU out of wait mode.

Contents

Maximum Ratings	102
Operating Temperature Range	103
Thermal Characteristics	103
Power Considerations	104
5.0 V DC Electrical Characteristics	105
3.3 V DC Electrical Characteristics	106
Driver Characteristics	107
Typical Supply Currents	109
EPROM Programming Characteristics	111
Control Timing	112
Mechanical Specifications	116
20-Pin PDIP — Case #738-03	117
20-Pin SOIC — Case #751D-04	117
20-Pin Cerdip — Case #732-03	118

Maximum Ratings

Maximum ratings are the extreme limits to which the MCU can be exposed without permanently damaging it.

The MCU contains circuitry to protect the inputs against damage from high static voltages; however, do not apply voltages higher than those shown in [Table 1](#). Keep V_{in} and V_{out} within the range $V_{SS} \leq (V_{in}$ or $V_{out}) \leq V_{DD}$. Connect unused inputs to the appropriate voltage level, either V_{SS} or V_{DD} .

Table 1. Maximum Ratings⁽¹⁾

Rating	Symbol	Value	Unit
Supply Voltage	V_{DD}	-0.3 to +7.0	V
Current Drain per Pin (Excluding V_{DD} , V_{SS} , and PA4–PA7)	I	25	mA
Input Voltage	V_{in}	$V_{SS} - 0.3$ to $V_{DD} + 0.3$	V
\overline{IRQ}/V_{PP} Pin	V_{PP}	$V_{SS} - 0.3$ to $2 \times V_{DD} + 0.3$	V
Storage Temperature Range	T_{STG}	-65 to +150	°C

1. Voltages are referenced to V_{SS} .

NOTE: *This device is not guaranteed to operate properly at the maximum ratings. Refer to [5.0 V DC Electrical Characteristics](#) and [3.3 V DC Electrical Characteristics](#) for guaranteed operating conditions.*

Operating Temperature Range

Table 2. Operating Temperature Range

Package Type	Symbol	Value (T _L to T _H)	Unit
MC68HC705J1AP ⁽¹⁾ , DW ⁽²⁾ , S ⁽³⁾	T _A	0 to 70	°C
MC68HC705J1AC ⁽⁴⁾ P, CDW, CS	T _A	−40 to +85	°C
MC68HC705J1AV ⁽⁵⁾ P, VDW, VS	T _A	−40 to +105	°C

1. P = plastic dual in-line package (PDIP)
2. DW = small outline integrated circuit (SOIC)
3. S = ceramic DIP (Cerdip)
4. C = extended temperature range
5. V = automotive temperature range

Thermal Characteristics

Table 3. Thermal Characteristics

Characteristic	Symbol	Value	Unit
Thermal Resistance MC68HC705J1AP ⁽¹⁾ MC68HC705J1ADW ⁽²⁾ MC68HC705J1AS ⁽³⁾	θ _{JA}	60	°C/W

1. P = plastic dual in-line package (PDIP)
2. DW = small outline integrated circuit (SOIC)
3. S = ceramic DIP (Cerdip)

Power Considerations

The average chip junction temperature, T_J , in °C can be obtained from:

$$T_J = T_A + (P_D \times \theta_{JA}) \quad (1)$$

where:

T_A = ambient temperature in °C

θ_{JA} = package thermal resistance, junction to ambient in °C/W

$P_D = P_{INT} + P_{I/O}$

$P_{INT} = I_{CC} \times V_{CC}$ = chip internal power dissipation

$P_{I/O}$ = power dissipation on input and output pins (user-determined)

For most applications, $P_{I/O} \ll P_{INT}$ and can be neglected.

Ignoring $P_{I/O}$, the relationship between P_D and T_J is approximately:

$$P_D = \frac{K}{T_J + 273 \text{ °C}} \quad (2)$$

Solving equations (1) and (2) for K gives:

$$K = P_D \times (T_A + 273 \text{ °C}) + \theta_{JA} \times (P_D)^2 \quad (3)$$

where K is a constant pertaining to the particular part. K can be determined from equation (3) by measuring P_D (at equilibrium) for a known T_A . Using this value of K, the values of P_D and T_J can be obtained by solving equations (1) and (2) iteratively for any value of T_A .

5.0 V DC Electrical Characteristics

Table 4. DC Electrical Characteristics ($V_{DD} = 5.0 \text{ Vdc}$)⁽¹⁾

Characteristic	Symbol	Min	Typ ⁽²⁾	Max	Unit
Output Voltage $I_{Load} = 10.0 \mu\text{A}$ $I_{Load} = -10.0 \mu\text{A}$	V_{OL} V_{OH}	— $V_{DD} - 0.1$	— —	0.1 —	V
Output High Voltage ($I_{Load} = -0.8 \text{ mA}$) PA0–PA7, PB0–PB5	V_{OH}	$V_{DD} - 0.8$	—	—	V
Output Low Voltage ($I_{Load} = 1.6 \text{ mA}$) PA0–PA3, PB0–PB5 ($I_{Load} = 10.0 \text{ mA}$) PA4–PA7	V_{OL}	—	—	0.4 0.4	V
Input High Voltage PA0–PA7, PB0–PB5, \overline{IRQ}/V_{PP} , \overline{RESET} , OSC1	V_{IH}	$0.7 \times V_{DD}$	—	V_{DD}	V
Input Low Voltage PA0–PA7, PB0–PB5, \overline{IRQ}/V_{PP} , \overline{RESET} , OSC1	V_{IL}	V_{SS}	—	$0.2 \times V_{DD}$	V
Supply Current Run Mode ⁽³⁾ Wait Mode ⁽⁴⁾ Stop Mode ⁽⁵⁾ 25 °C –40 to 105 °C	I_{DD}	— —	3.5 0.45	6.0 2.75	mA mA
I/O Ports Hi-Z Leakage Current PA0–PA7, PB0–PB5 (Without Individual Pulldown Activated)	I_{IL}	—	0.2	± 1	μA
Input Pulldown Current PA0–PA7, PB0–PB5 (With Individual Pulldown Activated)	I_{IL}	35	80	200	μA
Input Pullup Current \overline{RESET}	I_{IL}	–15	–35	–85	μA
Input Current ⁽⁶⁾ \overline{RESET} , \overline{IRQ}/V_{PP} , OSC1	I_{in}	—	0.2	± 1	μA
Capacitance Ports (As Inputs or Outputs) \overline{RESET} , \overline{IRQ} , OSC1, OSC2	C_{out} C_{in}	— —	— —	12 8	pF pF
Crystal/Ceramic Resonator Oscillator Mode Internal Resistor OSC1 to OSC2 ⁽⁷⁾	R_{osc}	1.0	2.0	3.0	M Ω

- $V_{DD} = 5.0 \text{ Vdc} \pm 10\%$, $V_{SS} = 0 \text{ Vdc}$, $T_A = -40 \text{ }^\circ\text{C}$ to $+105 \text{ }^\circ\text{C}$ unless otherwise noted.
- Typical values at midpoint of voltage range, 25 °C only.
- Run mode I_{DD} is measured using external square wave clock source ($f_{osc} = 4.2 \text{ MHz}$); all inputs 0.2 V from rail; no dc loads; less than 50 pF on all outputs; $C_L = 20 \text{ pF}$ on OSC2.
- Wait mode I_{DD} : only timer system active. Wait mode is affected linearly by OSC2 capacitance. Wait mode is measured with all ports configured as inputs; $V_{IL} = 0.2 \text{ V}$; $V_{IH} = V_{DD} - 0.2 \text{ V}$. Wait mode I_{DD} is measured using external square wave clock source ($f_{osc} = 4.2 \text{ MHz}$); all inputs 0.2 V from rail; no dc loads; less than 50 pF on all outputs; $C_L = 20 \text{ pF}$ on OSC2.
- Stop mode I_{DD} is measured with OSC1 = V_{SS} . Stop mode I_{DD} is measured with all ports configured as inputs; $V_{IL} = 0.2 \text{ V}$; $V_{IH} = V_{DD} - 0.2 \text{ V}$.
- Only input high current rated to $+1 \mu\text{A}$ on \overline{RESET} .
- The R_{osc} value selected for RC oscillator versions of this device is unspecified. See [Appendix C](#) for additional information.

3.3 V DC Electrical Characteristics

Table 5. DC Electrical Characteristics ($V_{DD} = 3.3 \text{ Vdc}$)⁽¹⁾

Characteristic	Symbol	Min	Typ ⁽²⁾	Max	Unit
Output Voltage $I_{Load} = 10.0 \mu\text{A}$ $I_{Load} = -10.0 \mu\text{A}$	V_{OL} V_{OH}	— $V_{DD} - 0.1$	— —	0.1 —	V
Output High Voltage ($I_{Load} = -0.2 \text{ mA}$) PA0–PA7, PB0–PB5	V_{OH}	$V_{DD} - 0.3$	—	—	V
Output Low Voltage ($I_{Load} = 0.4 \text{ mA}$) PA0–PA3, PB0–PB5 ($I_{Load} = 5.0 \text{ mA}$) PA4–PA7	V_{OL}	—	—	0.3 0.3	V
Input High Voltage PA0–PA7, PB0–PB5, \overline{IRQ}/V_{PP} , \overline{RESET} , OSC1	V_{IH}	$0.7 \times V_{DD}$	—	V_{DD}	V
Input Low Voltage PA0–PA7, PB0–PB5, \overline{IRQ}/V_{PP} , \overline{RESET} , OSC1	V_{IL}	V_{SS}	—	$0.2 \times V_{DD}$	V
Supply Current Run Mode ⁽³⁾ Wait Mode ⁽⁴⁾ Stop Mode ⁽⁵⁾ 25 °C –40 to 105 °C	I_{DD}	— —	1.2 0.25	4.0 1.5	mA mA
I/O Ports Hi-Z Leakage Current PA0–PA7, PB0–PB5 (Without Individual Pulldown Activated)	I_{IL}	—	0.1	± 1	μA
Input Pulldown Current PA0–PA7, PB0–PB5 (With Individual Pulldown Activated)	I_{IL}	12	30	100	μA
Input Pullup Current \overline{RESET}	I_{IL}	–10	–25	–45	μA
Input Current ⁽⁶⁾ \overline{RESET} , \overline{IRQ}/V_{PP} , OSC1	I_{in}	—	0.1	± 1	μA
Capacitance Ports (As Inputs or Outputs) \overline{RESET} , \overline{IRQ}/V_{PP} , OSC1, OSC2	C_{out} C_{in}	— —	— —	12 8	pF pF
Crystal/Ceramic Resonator Oscillator Mode Internal Resistor OSC1 to OSC2 ⁽⁷⁾	R_{osc}	1.0	2.0	3.0	M Ω

1. $V_{DD} = 3.3 \text{ Vdc} \pm 10\%$, $V_{SS} = 0 \text{ Vdc}$, $T_A = -40 \text{ }^\circ\text{C}$ to $+105 \text{ }^\circ\text{C}$ unless otherwise noted.

2. Typical values at midpoint of voltage range, 25 °C only.

3. Run mode I_{DD} is measured using external square wave clock source ($f_{osc} = 2.0 \text{ MHz}$); all inputs 0.2 V from rail; no dc loads; less than 50 pF on all outputs; $C_L = 20 \text{ pF}$ on OSC2.

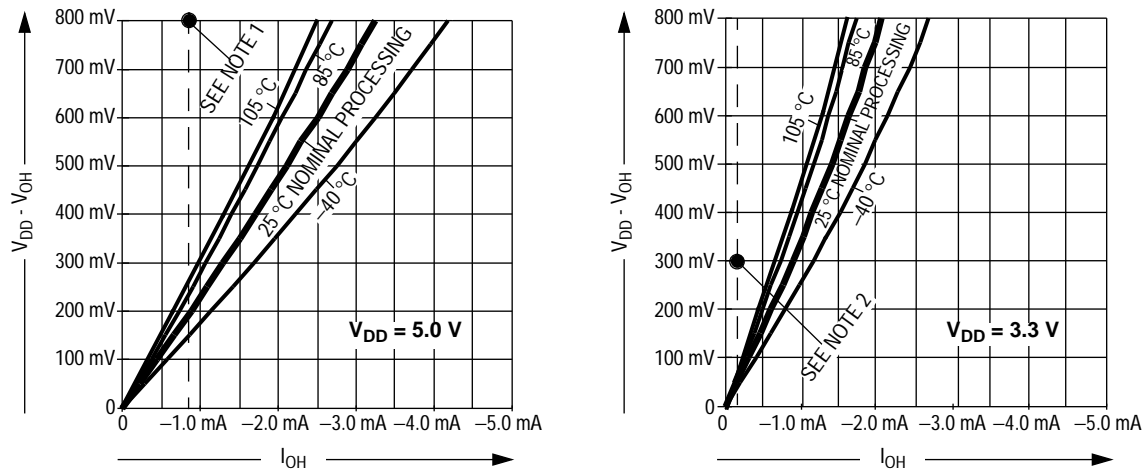
4. Wait mode I_{DD} : only timer system active. Wait mode is affected linearly by OSC2 capacitance. Wait mode is measured with all ports configured as inputs; $V_{IL} = 0.2 \text{ V}$; $V_{IH} = V_{DD} - 0.2 \text{ V}$. Wait mode I_{DD} is measured using external square wave clock source ($f_{osc} = 2.0 \text{ MHz}$); all inputs 0.2 V from rail; no dc loads; less than 50 pF on all outputs; $C_L = 20 \text{ pF}$ on OSC2.

5. Stop mode I_{DD} is measured with OSC1 = V_{SS} . Stop mode I_{DD} is measured with all ports configured as inputs; $V_{IL} = 0.2 \text{ V}$; $V_{IH} = V_{DD} - 0.2 \text{ V}$.

6. Only input high current rated to +1 μA on \overline{RESET} .

7. The R_{osc} value selected for RC oscillator versions of this device is unspecified. See [Appendix C](#) for additional information.

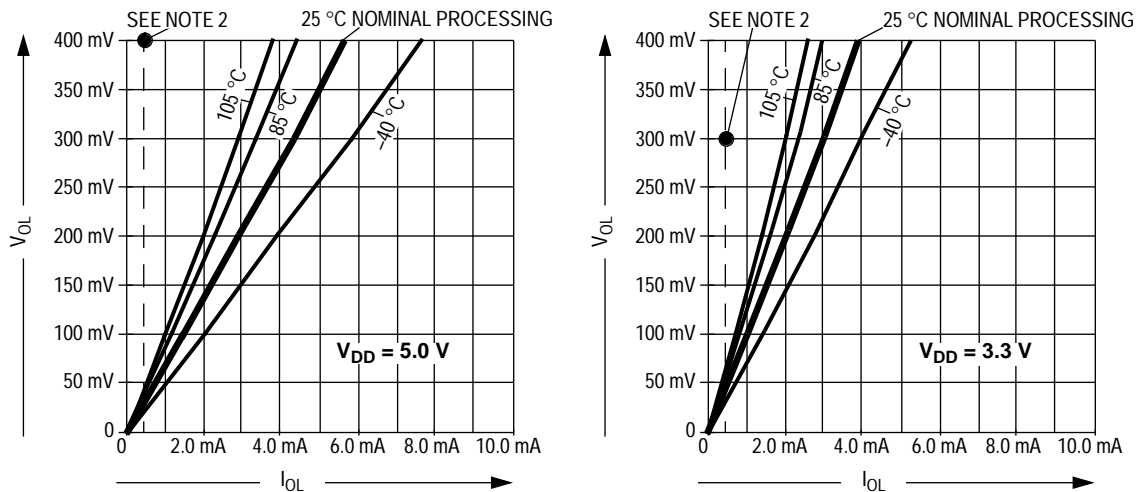
Driver Characteristics



NOTES:

1. At $V_{DD} = 5.0\text{ V}$, devices are specified and tested for $(V_{DD} - V_{OH}) \leq 800\text{ mV}$ @ $I_{OH} = -0.8\text{ mA}$.
2. At $V_{DD} = 3.3\text{ V}$, devices are specified and tested for $(V_{DD} - V_{OH}) \leq 300\text{ mV}$ @ $I_{OH} = -0.2\text{ mA}$.

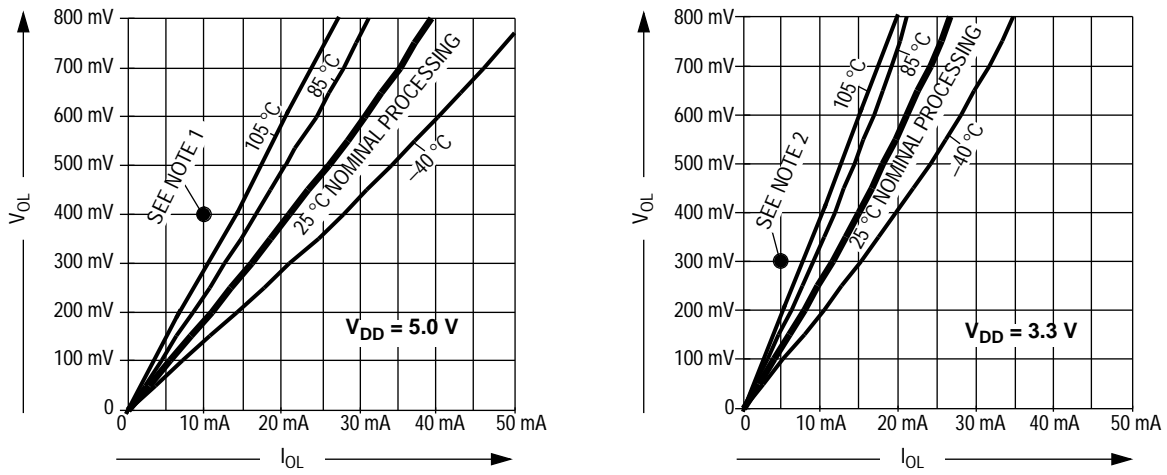
Figure 1. PA0–PA7, PB0–PB5 Typical High-Side Driver Characteristics



NOTES:

1. At $V_{DD} = 5.0\text{ V}$, devices are specified and tested for $V_{OL} \leq 400\text{ mV}$ @ $I_{OL} = 1.6\text{ mA}$.
2. At $V_{DD} = 3.3\text{ V}$, devices are specified and tested for $V_{OL} \leq 300\text{ mV}$ @ $I_{OL} = 0.4\text{ mA}$.

Figure 2. PA0–PA3, PB0–PB5 Typical Low-Side Driver Characteristics

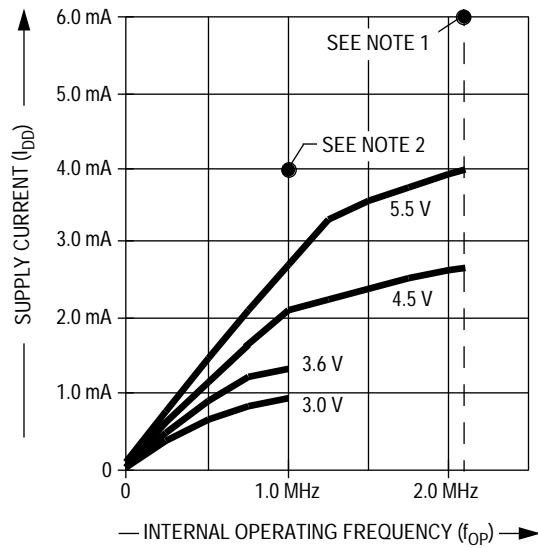


NOTES:

1. At $V_{DD} = 5.0\text{ V}$, devices are specified and tested for $V_{OL} \leq 400\text{ mV}$ @ $I_{OL} = 10.0\text{ mA}$.
2. At $V_{DD} = 3.3\text{ V}$, devices are specified and tested for $V_{OL} \leq 300\text{ mV}$ @ $I_{OL} = 5.0\text{ mA}$.

Figure 3. PA4–PA7 Typical Low-Side Driver Characteristics

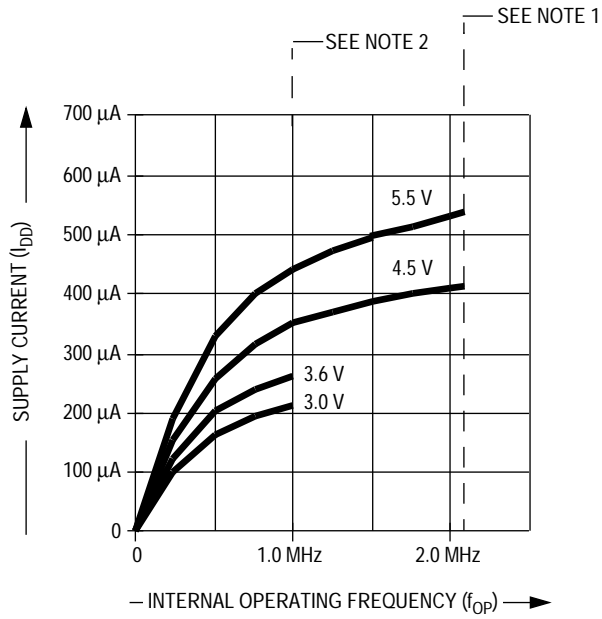
Typical Supply Currents



NOTES:

1. At $V_{DD} = 5.0$ V, devices are specified and tested for $I_{DD} \leq 6.0$ mA @ $f_{OP} = 2.1$ MHz.
2. At $V_{DD} = 3.3$ V, devices are specified and tested for $I_{DD} \leq 4.0$ mA @ $f_{OP} = 1.0$ MHz.

Figure 4. Typical Operating I_{DD} (25 °C)



NOTES:

1. At $V_{DD} = 5.0 V$, devices are specified and tested for $I_{DD} \leq 2.75 mA$ @ $f_{OP} = 2.1 MHz$.
2. At $V_{DD} = 3.3 V$, devices are specified and tested for $I_{DD} \leq 1.5 mA$ @ $f_{OP} = 1.0 MHz$.

Figure 5. Typical Wait Mode I_{DD} (25 °C)

EPROM Programming Characteristics

Table 6. EPROM Programming Characteristics⁽¹⁾

Characteristic	Symbol	Min	Typ	Max	Unit
Programming Voltage $\overline{\text{IRQ}}/V_{\text{PP}}$	V_{PP}	16.0	16.5	17.0	V
Programming Current $\overline{\text{IRQ}}/V_{\text{PP}}$	I_{PP}	—	3.0	10.0	mA
Programming Time Per Array Byte MOR	t_{EPGM}	4	—	—	ms
	t_{MPGM}	4	—	—	ms

1. $V_{\text{DD}} = 5.0 \text{ Vdc} \pm 10\%$, $V_{\text{SS}} = 0 \text{ Vdc}$, $T_{\text{A}} = -40 \text{ }^\circ\text{C}$ to $+105 \text{ }^\circ\text{C}$ unless otherwise noted.

Control Timing

Table 7. Control Timing ($V_{DD} = 5.0 \text{ Vdc}$)⁽¹⁾

Characteristic	Symbol	Min	Max	Unit
Oscillator Frequency Crystal Oscillator Option External Clock Source	f_{osc}	— dc	4.2 4.2	MHz
Internal Operating Frequency ($f_{osc} \div 2$) Crystal Oscillator External Clock	f_{OP}	— dc	2.1 2.1	MHz
Cycle Time ($1 \div f_{OP}$)	t_{cyc}	476	—	ns
$\overline{\text{RESET}}$ Pulse Width Low	t_{RL}	1.5	—	t_{cyc}
$\overline{\text{IRQ}}$ Interrupt Pulse Width Low (Edge-Triggered)	t_{LIH}	1.5	—	t_{cyc}
$\overline{\text{IRQ}}$ Interrupt Pulse Width Low (Edge- and Level- Triggered)	t_{LIL}	1.5	Note ⁽²⁾	t_{cyc}
PA0–PA3 Interrupt Pulse Width High (Edge-Triggered)	t_{HIL}	1.5	—	t_{cyc}
PA0–PA3 Interrupt Pulse Width (Edge- and Level-Triggered)	t_{HIH}	1.5	Note ⁽²⁾	t_{cyc}
OSC1 Pulse Width	t_{OH}, t_{OL}	200	—	ns

1. $V_{DD} = 5.0 \text{ Vdc} \pm 10\%$, $V_{SS} = 0 \text{ Vdc}$, $T_A = -40 \text{ }^\circ\text{C}$ to $+105 \text{ }^\circ\text{C}$ unless otherwise noted.

2. The maximum width t_{LIL} or t_{LIH} should not be more than the number of cycles it takes to execute the interrupt service routine plus $19 t_{cyc}$ or the interrupt service routine will be re-entered.

Table 8. Control Timing ($V_{DD} = 3.3 \text{ Vdc}$)⁽¹⁾

Characteristic	Symbol	Min	Max	Unit
Oscillator Frequency Crystal Oscillator Option External Clock Source	f_{osc}	— dc	2.0 2.0	MHz
Internal Operating Frequency ($f_{osc} \div 2$) Crystal Oscillator External Clock	f_{OP}	— dc	1.0 1.0	MHz
Cycle Time ($1 \div f_{OP}$)	t_{cyc}	1000	—	ns
$\overline{\text{RESET}}$ Pulse Width Low	t_{RL}	1.5	—	t_{cyc}
$\overline{\text{IRQ}}$ Interrupt Pulse Width Low (Edge-Triggered)	t_{LIH}	1.5	—	t_{cyc}
$\overline{\text{IRQ}}$ Interrupt Pulse Width Low (Edge- and Level- Triggered)	t_{LIL}	1.5	Note ⁽²⁾	t_{cyc}
PA0–PA3 Interrupt Pulse Width High (Edge-Triggered)	t_{IHIL}	1.5	—	t_{cyc}
PA0–PA3 Interrupt Pulse Width (Edge- and Level-Triggered)	t_{IHIH}	1.5	Note ⁽²⁾	t_{cyc}
OSC1 Pulse Width	t_{OH}, t_{OL}	400	—	ns

1. $V_{DD} = 3.3 \text{ Vdc} \pm 10\%$, $V_{SS} = 0 \text{ Vdc}$, $T_A = -40 \text{ }^\circ\text{C}$ to $+105 \text{ }^\circ\text{C}$ unless otherwise noted.

2. The maximum width t_{LIL} or t_{LIH} should not be more than the number of cycles it takes to execute the interrupt service routine plus $19 t_{cyc}$ or the interrupt service routine will be re-entered.

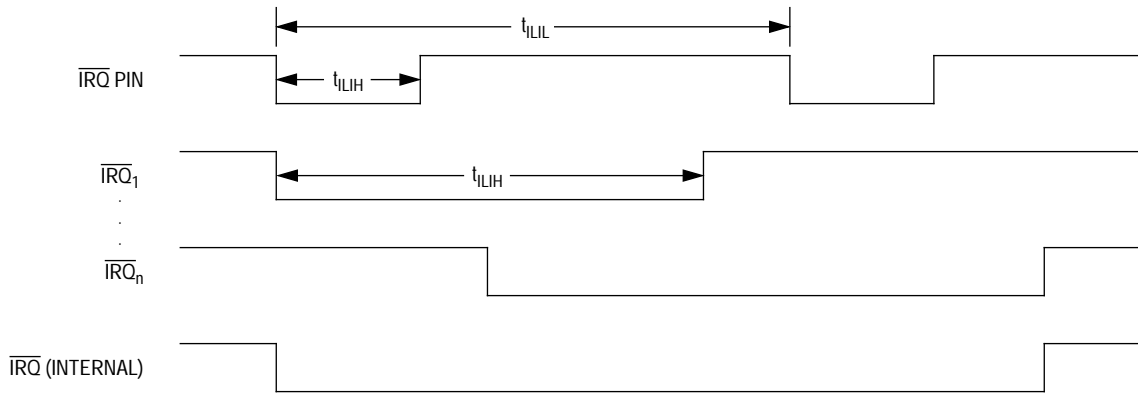
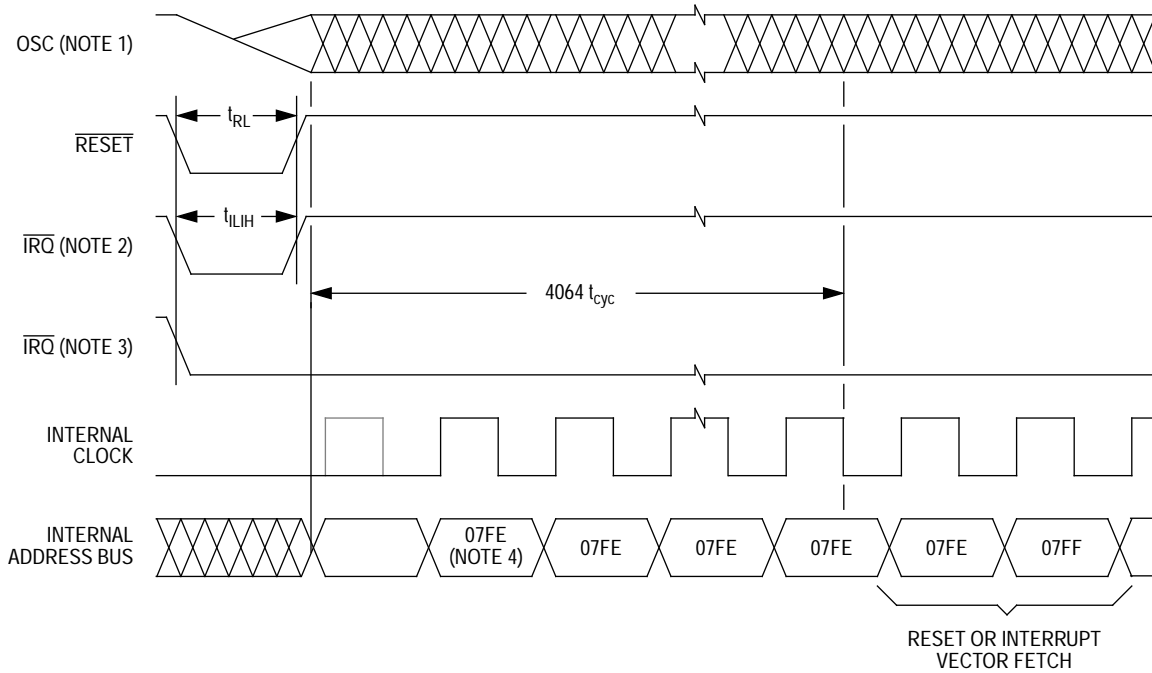


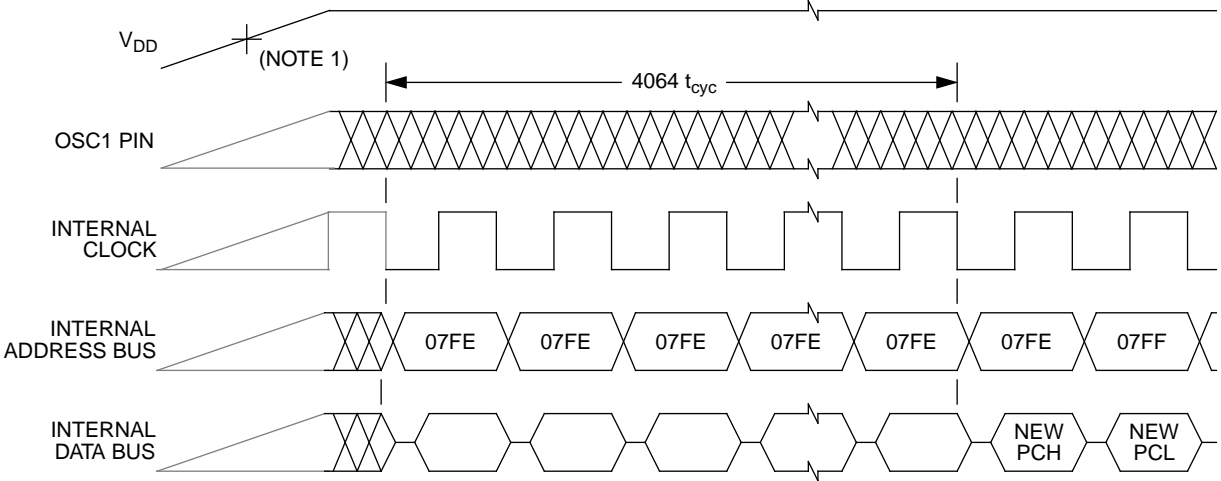
Figure 6. External Interrupt Timing



NOTES:

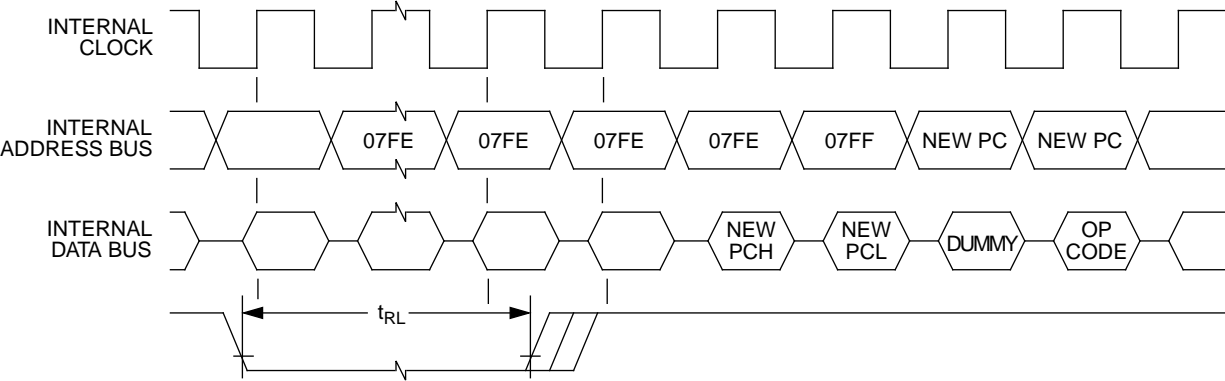
1. Internal clocking from OSC1 pin
2. Edge-triggered external interrupt mask option
3. Edge- and level-triggered external interrupt mask option
4. Reset vector shown as example

Figure 7. Stop Mode Recovery Timing



- NOTES:
1. Power-on reset threshold is typically between 1 V and 2 V.
 2. Internal clock, internal address bus, and internal data bus are not available externally.

Figure 8. Power-On Reset Timing



- NOTES:
1. Internal clock, internal address bus, and internal data bus are not available externally.
 2. The next rising edge of the internal clock after the rising edge of RESET initiates the reset sequence.

Figure 9. External Reset Timing

Mechanical Specifications

The MC68HC705J1A and the RC oscillator and high-speed option devices described in [Appendix A](#), [Appendix B](#), and [Appendix C](#) are available in the following packages:

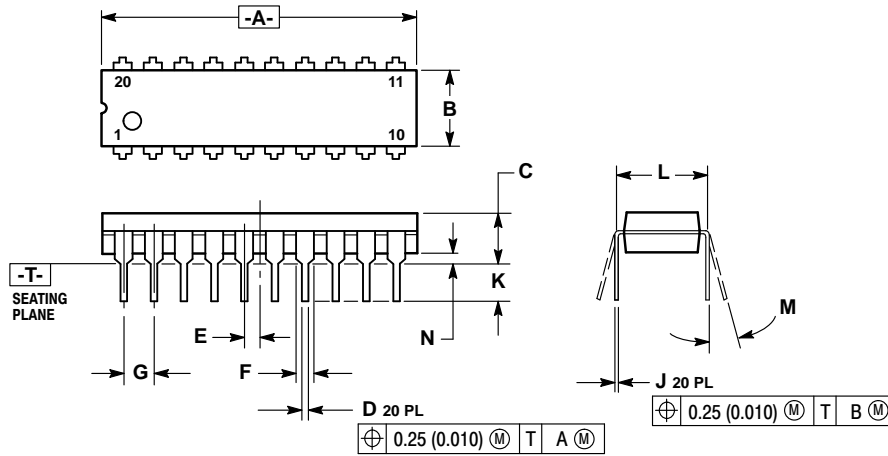
- 738-03 — plastic dual in-line package (PDIP)
- 751D-04 — small outline integrated circuit (SOIC)
- 732-03 — ceramic DIP (Cerdip) (windowed)

The following figures show the latest packages at the time of this publication. To make sure that you have the latest package specifications, contact one of the following:

- Local Motorola Sales Office
- Motorola Mfax
 - Phone 602-244-6609
 - EMAIL rmfax0@email.sps.mot.com
- Worldwide Web (wwwweb) at <http://design-net.com>

Follow Mfax or Worldwide Web on-line instructions to retrieve the current mechanical specifications.

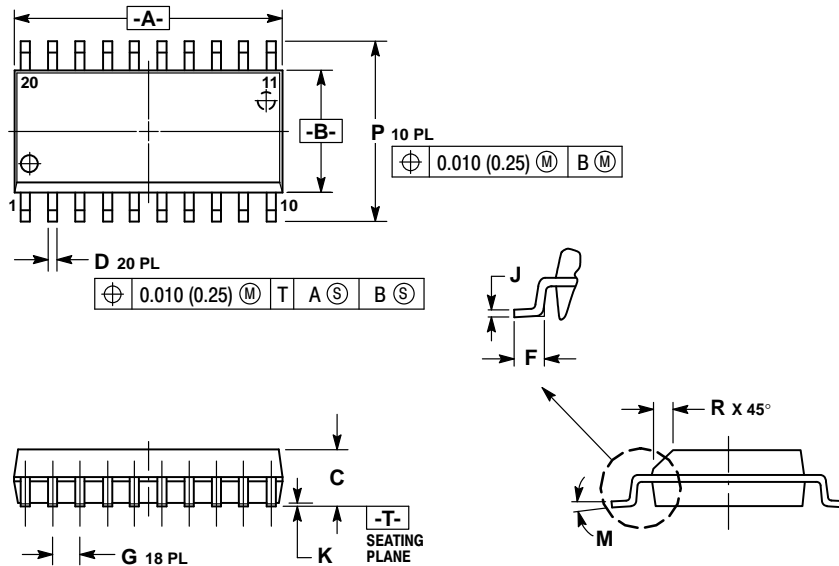
20-Pin PDIP —
Case #738-03



- NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
 2. CONTROLLING DIMENSION: INCH.
 3. DIMENSION L TO CENTER OF LEAD WHEN FORMED PARALLEL.
 4. DIMENSION B DOES NOT INCLUDE MOLD FLASH.

DIM	INCHES		MILLIMETERS	
	MIN	MAX	MIN	MAX
A	1.010	1.070	25.66	27.17
B	0.240	0.260	6.10	6.60
C	0.150	0.180	3.81	4.57
D	0.015	0.022	0.39	0.55
E	0.050 BSC		1.27 BSC	
F	0.050	0.070	1.27	1.77
G	0.100 BSC		2.54 BSC	
J	0.008	0.015	0.21	0.38
K	0.110	0.140	2.80	3.55
L	0.300 BSC		7.62 BSC	
M	0°	15°	0°	15°
N	0.020	0.040	0.51	1.01

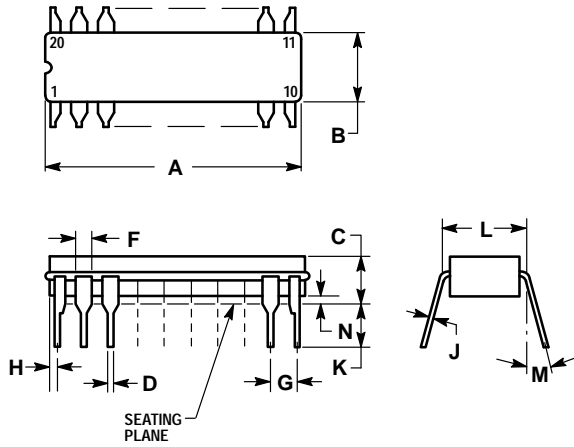
20-Pin SOIC —
Case #751D-04



- NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
 2. CONTROLLING DIMENSION: MILLIMETER.
 3. DIMENSIONS A AND B DO NOT INCLUDE MOLD PROTRUSION.
 4. MAXIMUM MOLD PROTRUSION 0.150 (0.006) PER SIDE.
 5. DIMENSION D DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL BE 0.13 (0.005) TOTAL IN EXCESS OF D DIMENSION AT MAXIMUM MATERIAL CONDITION.

DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	12.65	12.95	0.499	0.510
B	7.40	7.60	0.292	0.299
C	2.35	2.65	0.093	0.104
D	0.35	0.49	0.014	0.019
F	0.50	0.90	0.020	0.035
G	1.27 BSC		0.050 BSC	
J	0.25	0.32	0.010	0.012
K	0.10	0.25	0.004	0.009
M	0°	7°	0°	7°
P	10.05	10.55	0.395	0.415
R	0.25	0.75	0.010	0.029

20-Pin Cerdip —
Case #732-03



NOTES:

1. LEADS WITHIN 0.010 DIAMETER, TRUE POSITION AT SEATING PLANE, AT MAXIMUM MATERIAL CONDITION.
2. DIMENSION L TO CENTER OF LEADS WHEN FORMED PARALLEL.
3. DIMENSIONS A AND B INCLUDE MENISCUS.

DIM	INCHES	
	MIN	MAX
A	0.940	0.990
B	0.260	0.295
C	0.150	0.200
D	0.015	0.022
F	0.055	0.065
G	0.100 BSC	
H	0.020	0.050
J	0.008	0.012
K	0.125	0.160
L	0.300 BSC	
M	0° 15°	
N	0.010	0.040

Appendix A

MC68HRC705J1A

Contents

Introduction	119
RC Oscillator Connections	119
Typical Internal Operating Frequency for RC Oscillator Option	120
Package Types and Order Numbers	122

Introduction

Appendix A introduces the MC68HRC705J1A, a resistor-capacitor (RC) oscillator mask option version of the MC68HC705J1A. All of the information in *MC68HC705J1A Technical Data Rev. 2.0* applies to the MC68HRC705J1A with the exceptions given in this appendix.

RC Oscillator Connections

For greater cost reduction, the RC oscillator mask option allows the configuration shown in **Figure 1** to drive the on-chip oscillator. Mount the RC components as close as possible to the pins for startup stabilization and to minimize output distortion.

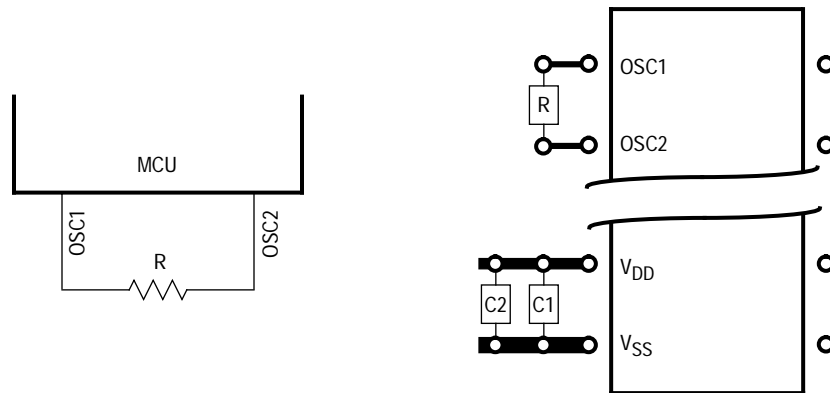


Figure 1. RC Oscillator Connections

NOTE: The optional internal resistor is **not** recommended for configurations that use the RC oscillator connections as shown in **Figure 1**. For such configurations, the oscillator internal resistor (*OSCRES*) bit of the mask option register should be programmed to a logic 0.

Typical Internal Operating Frequency for RC Oscillator Option

Figure 2 shows typical internal operating frequencies at 25 °C for the RC oscillator option.

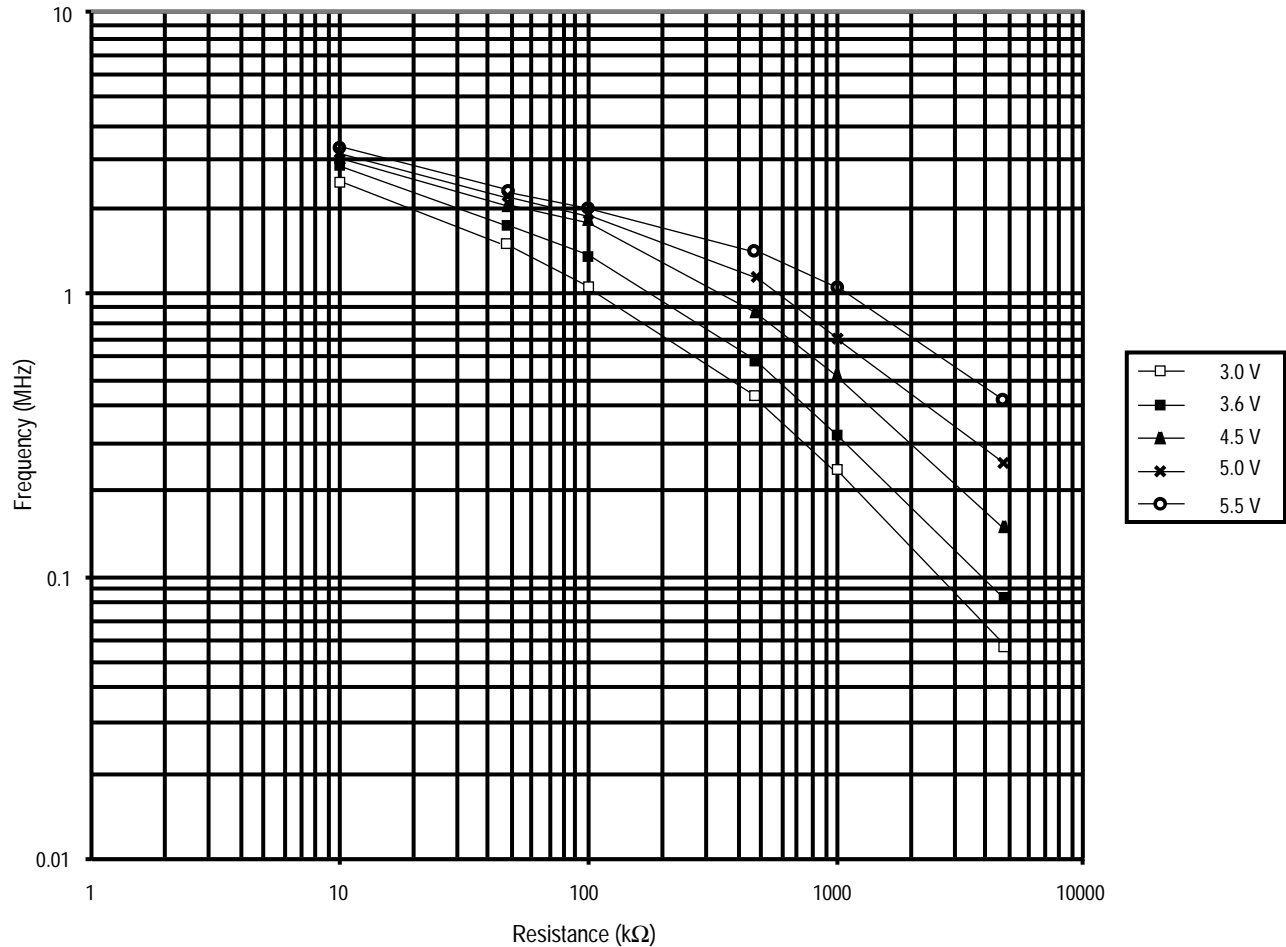


Figure 2. Typical Internal Operating Frequency for Various V_{DD} at 25 °C — RC Oscillator Option Only

NOTE: Tolerance for resistance is $\pm 50\%$. When selecting resistor size, consider the tolerance to ensure that the resulting oscillator frequency does not exceed the maximum operating frequency.

Package Types and Order Numbers

Table 1. MC68HRC705J1A (RC Oscillator Option) Order Numbers⁽¹⁾

Package Type	Case Outline	Pin Count	Operating Temperature	Order Number
PDIP	738-03	20	0 to 70 °C –40 to +85 °C –40 to + 105 °C	MC68HRC705J1AP ⁽²⁾ MC68HRC705J1AC ⁽³⁾ P MC68HRC705J1AV ⁽⁴⁾ P
SOIC	751D-04	20	0 to 70 °C –40 to +85 °C –40 to + 105 °C	MC68HRC705J1ADW ⁽⁵⁾ MC68HRC705J1ACDW MC68HRC705J1AVDW
Cerdip	732-03	20	0 to 70 °C –40 to +85 °C –40 to + 105 °C	MC68HRC705J1AS ⁽⁶⁾ MC68HRC705J1ACS MC68HRC705J1AVS

1. Refer to [Introduction](#) (opening section of manual) for standard part ordering information.
2. P = plastic dual in-line package (PDIP)
3. C = extended temperature range
4. V = automotive temperature range
5. DW = small outline integrated circuit (SOIC)
6. S = ceramic dual in-line package (Cerdip)

Appendix B

MC68HSC705J1A

Contents

Introduction	124
DC Electrical Characteristics	125
Typical Supply Currents	126
Package Types and Order Numbers	128

Introduction

Appendix B introduces the MC68HSC705J1A, a high-speed version of the MC68HC705J1A. All of the information in *MC68HC705J1A Technical Data Rev. 2.0* applies to the MC68HSC705J1A with the exceptions given in this appendix.

DC Electrical Characteristics

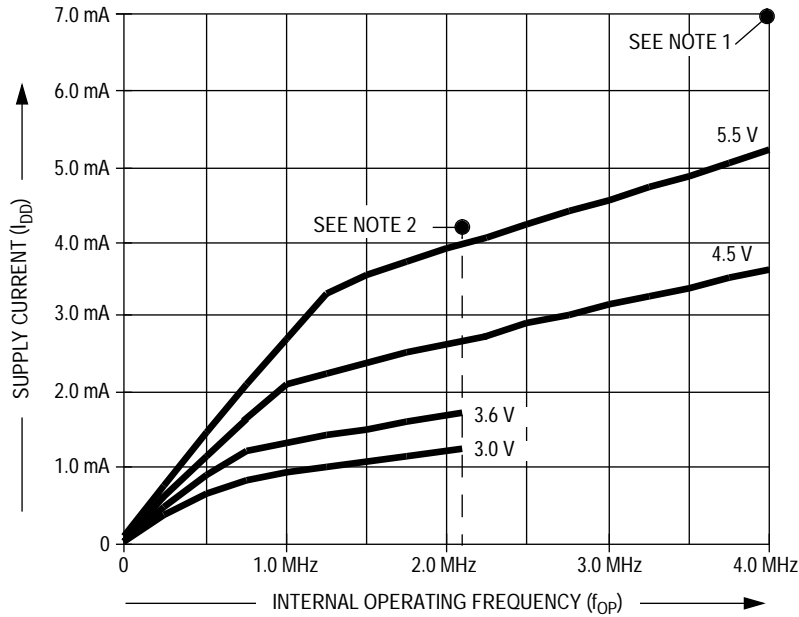
Table 1. DC Electrical Characteristics ($V_{DD} = 5\text{ V}$)

Characteristic	Symbol	Min	Typ	Max	Unit
Supply Current ($f_{OP} = 4.0\text{ MHz}$)	I_{DD}	—	4.25	7.0	mA
Run					
Wait					

Table 2. DC Electrical Characteristics ($V_{DD} = 3.3\text{ V}$)

Characteristic	Symbol	Min	Typ	Max	Unit
Supply Current ($f_{OP} = 2.1\text{ MHz}$)	I_{DD}	—	1.4	4.25	mA
Run					
Wait					

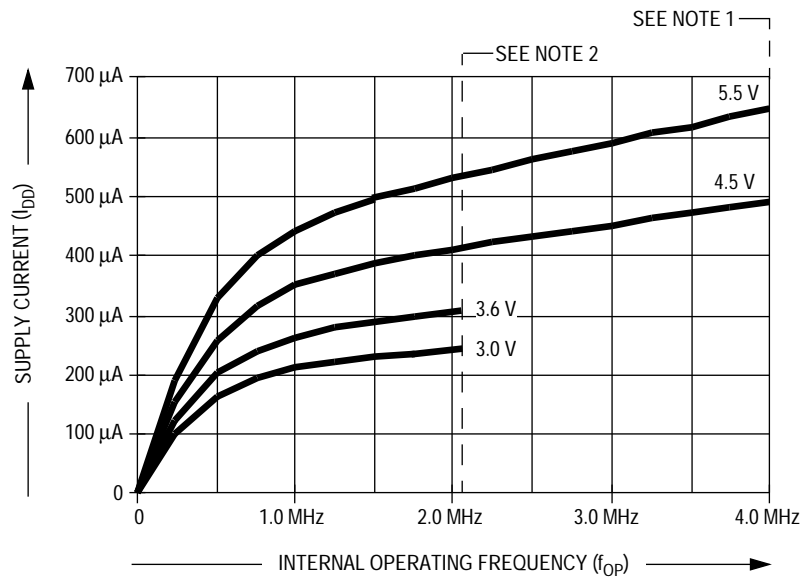
Typical Supply Currents



NOTES:

1. At $V_{DD} = 5.0$ V, high-speed devices are specified and tested for $I_{DD} \leq 7.0$ mA @ $f_{OP} = 4.0$ MHz.
2. At $V_{DD} = 3.3$ V, high-speed devices are specified and tested for $I_{DD} \leq 4.25$ mA @ $f_{OP} = 2.1$ MHz.

Figure 1. Typical High-Speed Operating I_{DD} (25 °C)



NOTES:

1. At $V_{DD} = 5.0\text{ V}$, high-speed devices are specified and tested for $I_{DD} \leq 3.25\text{ mA}$ @ $f_{OP} = 4.0\text{ MHz}$.
2. At $V_{DD} = 3.3\text{ V}$, high-speed devices are specified and tested for $I_{DD} \leq 1.75\text{ mA}$ @ $f_{OP} = 2.1\text{ MHz}$.

Figure 2. Typical High-Speed Wait Mode I_{DD} (25 °C)

Package Types and Order Numbers

Table 3. MC68HSC705J1A (High Speed) Order Numbers⁽¹⁾

Package Type	Case Outline	Pin Count	Operating Temperature	Order Number
PDIP	738-03	20	0 to 70 °C –40 to +85 °C	MC68HSC705J1AP ⁽²⁾ MC68HSC705J1AC ⁽³⁾ P
SOIC	751D-04	20	0 to 70 °C –40 to +85 °C	MC68HSC705J1ADW ⁽⁴⁾ MC68HSC705J1ACDW
Cerdip	732-03	20	0 to 70 °C –40 to +85 °C	MC68HSC705J1AS ⁽⁵⁾ MC68HSC705J1ACS

1. Refer to [Introduction](#) (opening section of manual) for standard part ordering information.
2. P = plastic dual in-line package (PDIP)
3. C = extended temperature range
4. DW = small outline integrated circuit (SOIC)
5. S = ceramic dual in-line package (Cerdip)

Appendix C

MC68HSR705J1A

Contents

Introduction	130
RC Oscillator Connections (External Resistor)	130
Typical Internal Operating Frequency at 25 °C for High-Speed RC Oscillator Option	131
RC Oscillator Connections (No External Resistor)	132
Typical Operating Frequency Versus Temperature (No External Resistor)	133
Package Types and Order Numbers	134

Introduction

Appendix C introduces the MC68HSR705J1A, a high-speed version of the MC68HRC705J1A. All of the information in *MC68HC705J1A Technical Data Rev. 2.0*, **Appendix A MC68HRC705J1A** and **Appendix B MC68HSC705J1A** applies to the MC68HSR705J1A with the exceptions given in this appendix.

RC Oscillator Connections (External Resistor)

Refer to **Appendix A MC68HRC705J1A** for a description of the RC oscillator connections with external resistor.

Typical Internal Operating Frequency at 25 °C for High-Speed RC Oscillator Option

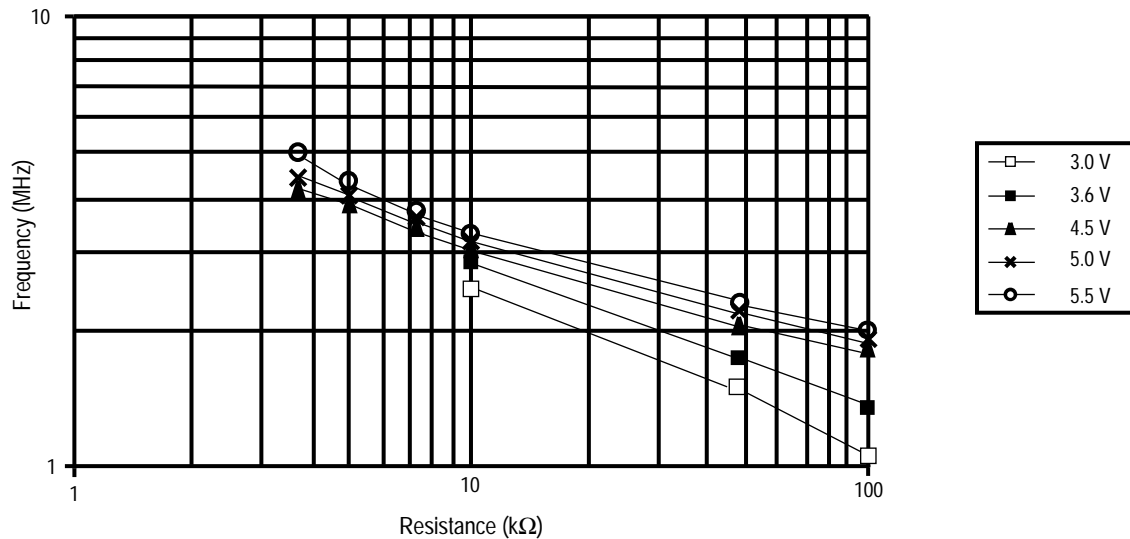


Figure 1. Typical Internal Operating Frequency at 25 °C for High-Speed RC Oscillator Option

For lower frequency operation characteristics refer to [Appendix A MC68HRC705J1A](#).

NOTE: *Tolerance for resistance is $\pm 50\%$. When selecting resistor size, consider the tolerance to ensure that resulting oscillator frequency does not exceed the maximum operating frequency.*

RC Oscillator Connections (No External Resistor)

For maximum cost reduction, the RC oscillator mask connections shown in **Figure 2** allow the on-chip oscillator to be driven with **no** external components. This can be accomplished by programming the oscillator internal resistor (OSCRES) bit in the mask option register to a logic 1. When programming the OSCRES bit for the MC68HSR705J1A, an internal resistor is selected which yields typical internal oscillator frequencies as shown in **Figure 3**. The internal resistance for this device is different than the resistance of the selectable internal resistor on the MC68HC705J1A and the MC68HSC705J1A devices.

NOTE: *This option is not available on the ROM version of this device (MC68HC05J1A).*

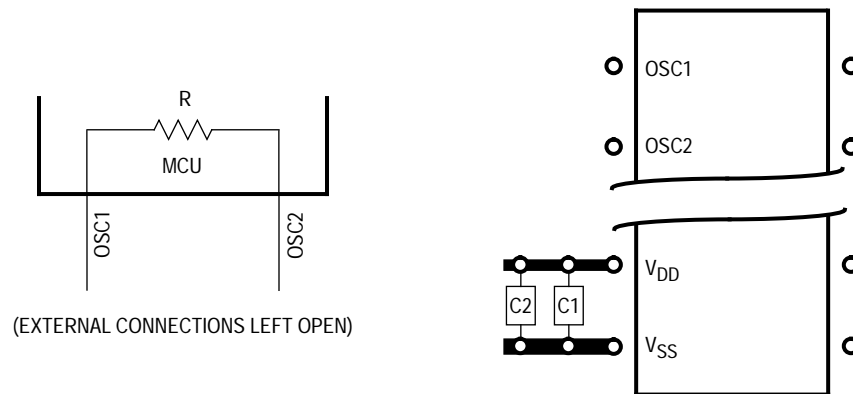


Figure 2. RC Oscillator Connections (No External Resistor)

Typical Internal Operating Frequency Versus Temperature (No External Resistor)

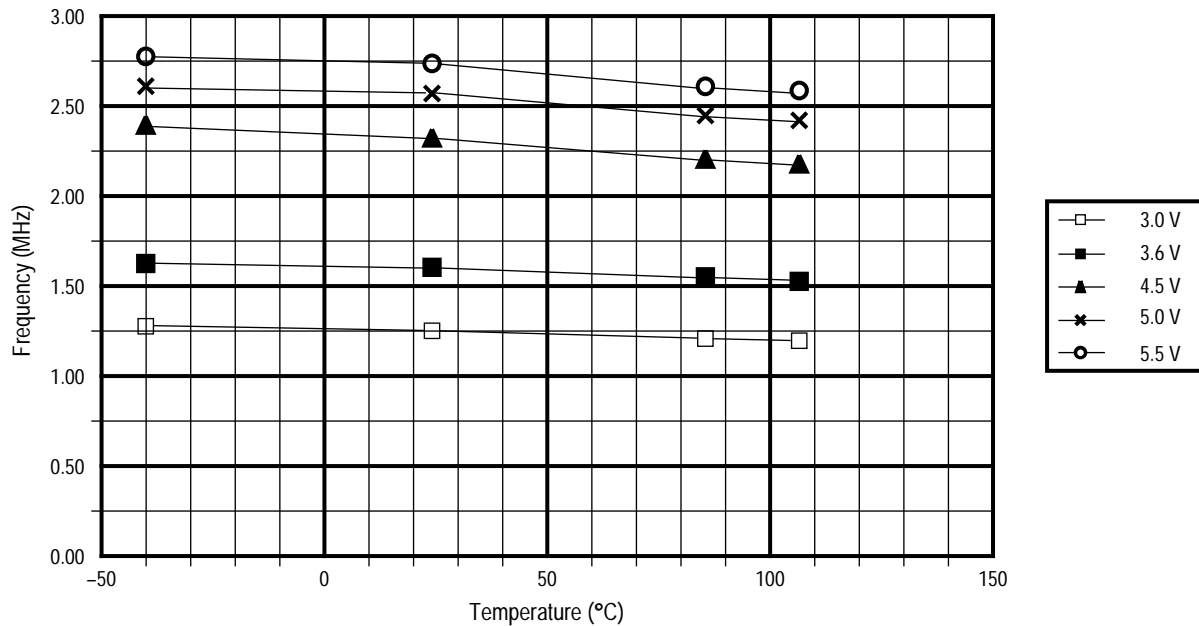


Figure 3. Typical Internal Operating Frequency Versus Temperature (OSCRE Bit = 1)

NOTE: Due to process variations, operating voltages, and temperature requirements, the internal resistance and tolerance are unspecified. Typically for a given voltage and temperature, the frequency should not vary more than ± 500 kHz. However, this data is not guaranteed. It is the user's responsibility to ensure that the resulting internal operating frequency meets user's requirements.

Package Types and Order Numbers

Table 1. MC68HSR705J1A (High-Speed RC Oscillator Option) Order Numbers⁽¹⁾

Package Type	Case Outline	Pin Count	Operating Temperature	Order Number
PDIP	738-03	20	0 to 70 °C –40 to +85 °C	MC68HSR705J1AP ⁽²⁾ MC68HSR705J1AC ⁽³⁾ P
SOIC	751D-04	20	0 to 70 °C –40 to +85 °C	MC68HSR705J1ADW ⁽⁴⁾ MC68HSR705J1ACDW
Cerdip	732-03	20	0 to 70 °C –40 to +85 °C	MC68HSR705J1AS ⁽⁵⁾ MC68HSR705J1ACS

1. Refer to [Introduction](#) (opening section of manual) for standard part ordering information.
2. P = plastic dual in-line package (PDIP)
3. C = extended temperature range
4. DW = small outline integrated circuit (SOIC)
5. S = ceramic dual in-line package (Cerdip)

A	
accumulator register (A)	35
addressing modes	38
B	
block diagram	11
brownout	82
C	
C bit	37
case outlines	117–118
central processor unit	9, 31
condition code register (CCR)	36
COP watchdog	66
COP in stop mode	84
COP in wait mode	84
COP register (COPR)	83
COP reset	56
features	81
interrupts	83
low-power modes	84
operation	82
programmable option	12
real-time interrupt rate selection table ..	98
COPEN bit	29
CPU	32
features	32
instruction set	38
instruction set summary	46
instruction types	41
opcode map	52
programming model	33
CPU registers	
accumulator register (A)	35
index register (X)	35
program counter register (PC)	36
stack pointer register (SP)	35
D	
data direction registers	
data direction register A (DDRA)	73
data direction register B (DDRB)	77
data-retention mode	68
E	
ELAT bit	26
electrical specifications	101
control timing	112–113
DC electrical characteristics	105–106
driver characteristics	107
maximum ratings	102
MC68HSC705J1A (high-speed option)	125–126
MC68HSR705J1A (high-speed RC oscillator option)	130
operating temperature range	103
port A	79–80
port B	79–80
power considerations	104
thermal characteristics	103
thermal resistance	103
typical supply currents	109

electrostatic damage	71	operation	86
EPGM bit	26	optional external interrupts	90
EPMSEC bit	28	pin sensitivity selection	89
EPROM		pin triggering option	12
EPROM security programmable option	12	port A external interrupts	
EPROM/OTPROM	25	programmable option	12
erasing	27	real-time interrupt rate selection table	98
programming	25, 27	real-time interrupts	59
programming characteristics	30	reset/interrupt vector addresses	61
programming register (EPROG)	26	software interrupt	56
external interrupt pins	20	software interrupt vector	59
external reset	55	sources	56
		timer interrupt vector	59
		timer interrupts	59, 96
		timer overflow	59
	H		
H bit	37	IRQ latch	56
		$\overline{\text{IRQ}}/\text{V}_{\text{PP}}$ pin	20, 82, 89
	I	IRQE bit	91
I bit	37	IRQF bit	91
index register (X)	35	IRQR bit	91
instruction set	38		
addressing modes	38		L
instruction set summary	46	LEVEL bit	29
instruction types	41	literature distribution centers	141
opcode map	52	low-power modes	63–64
instruction types	41	COP timeout period	67
interrupts	53	data-retention mode	64, 68
external interrupt	56–57	effects on clock generation	65
external interrupt logic	57	effects on COP	66
external interrupt module	86	effects on CPU	66
external interrupt timing	58, 92	effects on EPROM/OTPROM	68
external interrupt vector	59	effects on timer	67
features	85	exiting stop mode	64
interrupt flowchart	62, 88	exiting wait mode	65
interrupt processing	59	flowchart (STOP/HALT/WAIT)	70
interrupt sources	56	halt mode	64
interrupt stacking order	60	STOP instruction flowchart	70
IRQ module block diagram	87	stop mode	64, 66
IRQ status and control register (ISCR)	91	stop recovery timing	69
$\overline{\text{IRQ}}/\text{V}_{\text{PP}}$ pin	86, 89		

timer	100	memory map	22
timing of stop mode recovery	69	RAM	24
wait mode	64	MPGM bit	26
M		N	
mask option register (MOR)	27	N bit	37
programming	28	O	
MC68HC705J1A		opcode map	52
features	10	options (mask)	27
MC68HRC705J1A (RC oscillator option)	119	options (programmable)	12
operating frequency	121	ordering information	
order numbers	122	literature distribution centers	141
package types	122	MC68HRC705J1A (RC oscillator option)	122
RC oscillator connections	119–120	MC68HSC705J1A (high-speed option)	128
MC68HSC705J1A (high-speed option)	123–124	MC68HSR705J1A (high-speed RC oscillator option)	134
DC electrical characteristics	125	Mfax	142
order numbers	128	order numbers	12, 122, 128, 134
package types	128	Web server	142
typical operating current	126	Web site	142
typical wait mode current	127	OSC1 pin	15
MC68HSR705J1A (high-speed RC oscillator option)	130	OSC2 pin	15
operating frequencies (with OSCRES bit set)	133	oscillator	
operating frequency	131	crystal oscillator internal resistor option	12
order numbers	134	delay counter programmable option	12
package types	134	on-chip oscillator stabilization delay	54
RC oscillator	132	pins	15
RC oscillator connections	132	OSCRES bit	29
mechanical specifications		P	
Cerdip	118	PA0–PA3 pins	86
package types	116	package dimensions	
PDIP	117	Cerdip	118
SOIC	117	PDIP	117
memory	21	SOIC	117
EPROM/OTPROM	25	package types	12
EPROM/OTPROM programming	25		
features	21		
I/O register summary	23		
mask option register	27		

- pin assignments14
 - pin functions15
 - PIRQ bit29
 - port A
 - data direction register (DDRA)73
 - data register (PORTA)73
 - electrical characteristics79–80
 - I/O circuitry74
 - I/O pin interrupts (PA0–PA3)76
 - LED drive capability76
 - pin operation75
 - pins20
 - pulldown register (PDRA)75
 - port B
 - data direction register (DDRB)77
 - electrical characteristics79–80
 - I/O circuitry77
 - pin operation78
 - pins20
 - port B data register (PORTB)76
 - pulldown register (PDRB)78
 - power-on reset54
 - program counter (PC)36
 - programmable options12
 - programming model (CPU)33
 - pulldown register A (PDRA)75
 - pulldown register B (PDRB)78
 - pulldown resistors
 - programmable option12
- R**
- RAM24
 - stack RAM59
 - registers
 - CPU34
 - I/O register summary23
 - parallel I/O port register summary72
 - timer I/O register summary87, 95
 - $\overline{\text{RESET}}$ pin19, 55
 - resets53–54
 - COP register (COPR)56
 - COP watchdog reset56
 - external reset55
 - external reset timing55
 - illegal address56
 - power-on reset (POR)54
 - power-on reset timing55
 - reset sources54
 - reset/interrupt vector addresses61
 - resets and interrupts53
 - resistors (pulldown) programmable
 - option12
 - RT1, RT0 bits98
 - RTIE bit97
 - RTIF bit97
 - RTIFR bit98
- S**
- Schmitt trigger20, 89–90
 - SOSCD bit28
 - stack pointer register (SP)35
 - STOP instruction65, 84, 91
 - stop mode66, 84
 - effect on COP watchdog84
 - effects on timer100
 - STOP instruction flowchart70
 - stop recovery timing69
 - stop/halt mode programmable option12
 - SWAIT bit29
 - SWPDI bit29

T

timer	
block diagram	.94
features	.93
I/O register summary	.87, 95
I/O registers	.96
interrupts	.59, 96
low-power modes	.100
operation	.95
real-time interrupt rate selection	.98
timer counter register	.99
timer interrupt vector	.59
timer status and control register (TSCR)	.97
TOF bit	.97
TOFR bit	.98
TOIE bit	.97

V

V _{DD} pin	.15
V _{SS} pin	.15

W

WAIT instruction	.65, 84, 91
wait mode	.84
effects on timer	.100
Web server	.142
Web site	.142

Z

Z bit	.37
-------	-----

Literature Updates

This document contains the latest data available at publication time. For updates, contact one of the centers listed below:

Literature Distribution Centers

Order literature by mail or phone.

USA/Europe

Motorola Literature Distribution
P.O. Box 20912
Phoenix, Arizona 85036
Phone 1 800 441-2447 or 602 303-5454

Japan

Nippon Motorola Ltd.
Tatsumi-SPD-JLDC
Toshikatsu Otsuki
6F Seibu-Butsuryu Center
3-14-2 Tatsumi Koto-Ku
Tokyo 135, Japan
Phone 03-3521-8315

Hong Kong

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

Mfax

To access this worldwide faxing service call or contact by electronic mail:

RMFAX0@email.sps.mot.com

TOUCH-TONE 602-244-6609

Or, on the <http://Design-NET.com> home page, select the Mfax icon.

Obtain a fax of complete, easy-to-use Mfax instructions by entering your FAX number and then pressing the 1 key.

Motorola SPS World Marketing World Wide Web Server

Use the Internet to access Motorola's World Wide Web server. Use the following URL:

<http://Design-net.com>

CSIC Microcontroller Division's Web Site

Directly access the CSIC Microcontroller Division's web site with the following URL:

http://Design-net.com/csic/CSIC_home.html



Please help us to continue improving the quality and usefulness of our data books by completing this form and sending your comments to us. You can return the form by mail or FAX it to 512-891-3236.

Thank you for your help and continued support!

1. How do you rate the quality of this data book?

	High			Low	
Organization	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Readability	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Accuracy	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Figures	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

	High			Low	
Tables	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Table of contents	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Page size/binding	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Overall impression	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

2. Are you able to find the information you need easily?

Yes	No			
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

3. What are your recommendations for making this data book more useful?

4. What additional information would you like to see included in future data books?

PLEASE
PASTE
POSTAGE
HERE

Motorola
6501 William Cannon Drive West
Mail Stop OE17
Austin, Texas 78735-8598
USA

Attention: CSIC Publications Department



Second: fold back along this line

First: cut along this line to remove

Please supply the following information (optional).

Name: _____

Company Name: _____

Title: _____

Address: _____

City: _____ State: _____ Zip: _____

Phone Number: _____