68HC05SB7
68HC705SB7

SPECIFICATION
(General Release)

August 27, 1998

Consumer Systems Group
Semiconductor Products Sector

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

TABLE OF CONTENTS

Section **Page**

SECTION 1
GENERAL DESCRIPTION

SECTION 2
MEMORY

SECTION 3
CENTRAL PROCESSING UNIT

TABLE OF CONTENTS

Section **Page**

SECTION 4
INTERRUPTS

SECTION 5
RESETS

SECTION 6
LOW POWER MODES

TABLE OF CONTENTS

Section **Page**

TABLE OF CONTENTS

Section **Page**

SECTION 12
SM-BUS

SECTION 13
CURRENT SENSE AMPLIFIER

TABLE OF CONTENTS

Section **Page**

SECTION 14
TEMPERATURE SENSOR

SECTION 15
ANALOG SUBSYSTEM

SECTION 16
PERSONALITY EPROM

SECTION 17
INSTRUCTION SET

TABLE OF CONTENTS

Section **Page**

LIST OF FIGURES

LIST OF FIGURES

LIST OF TABLES

LIST OF TABLES

Table **Title** **Page**

SECTION 1
GENERAL DESCRIPTION

The MC68HC05SB7 HCMOS microcontroller is a member of the M68HC05 family of low-cost single-chip microcontrollers. This 8-bit microcontroller unit (MCU), which contains an internal oscillator, CPU, RAM, ROM, personality EPROM, I/O, 16-bit timer, core timer, watchdog system, LVR, SM-Bus, PWM, current sense amplifier, internal temperature sensor and A/D, is designed specifically for smart battery applications.

1.1 FEATURES

- Industry standard 8-bit M68HC05 CPU core
- Power saving STOP, WAIT, DATA-RETENTION and SLOW modes
- 2.1MHz maximum bus frequency from internal VCO or external pin oscillator
- 6144 bytes of user ROM with the security feature
- 224 bytes of user RAM (64 bytes for stack)
- System calibration characteristics by 64-bit Personality EPROM
- 19 bidirectional I/O lines
  - 4 shared with SM-Bus
  - 4 shared with PWM
  - 4 shared with A/D analog channels input
  - 2 shared with current detect output
  - 1 shared with Timer Input Capture (TCAP)
- 16-bit Programmable Timer with Input Capture/Output Compare (driven by interrupt)
- 15-stage multi-function Core Timer including 8-bit free-running counter and 4-stage selectable real-time interrupt generator
- Built-in current sensing amplifiers with selectable gain of 10 and 30
- Two voltage comparators which can be combined with the 16-bit Timer to create an 8-channel, single-slope Analog to Digital Converter
- Built-in internal temperature sensor from 0°C to 70°C

- 4 channels 10-bit PWM running at a fixed clock rate
- SM-Bus† serial interface compatible with I^2C†† Bus
- Slow ramp up power supply reset capability via LVR
- Selectable sensitivity on IRQ interrupt (Edge- and Level-Sensitive or Edge-Only)
- SM-Bus, current detect, 16-bit timer, analog subsystem and core timer interrupts
- Internal 100kΩ pull-up resistor on $\overline{\text{RESET}}$
- Low Voltage Reset (LVR)
- Illegal Address Reset
- Computer Operating Properly (COP) Watchdog system
- Available in 28-pin SSOP

NOTE

A bar over a signal name indicates an active low signal. For example, RESET is active high and $\overline{\text{RESET}}$ is active low. Any reference to voltage, current, or frequency specified in the following sections will refer to the nominal values. The exact values and their tolerance or limits are specified in the Electrical Specifications section.

1.2 MASK OPTION

A single mask option is available on the MC68HC05SB7.

- External oscillator on pins OSC1 and OSC2 (EPO): [enabled or disabled]

1.3 PEPROM FACTORY PREPROGRAMMED OPTIONS

The MC68HC05SB7 is available with a factory preprogrammed PEPROM containing any of the following measured parameters:

- The internal VCO minimum frequency: programmed or left blank
- The internal VCO maximum frequency: programmed or left blank
- The internal bandgap reference voltage: programmed or left blank
- The internal temperature sensor voltage at 80°C: programmed or left blank

1.4 MCU STRUCTURE

The block diagram of the MC68HC05SB7 is shown in **Figure 1-1**.

†SM-Bus is an Intel bus standard.
††I^2C Bus is a Philips bus standard.

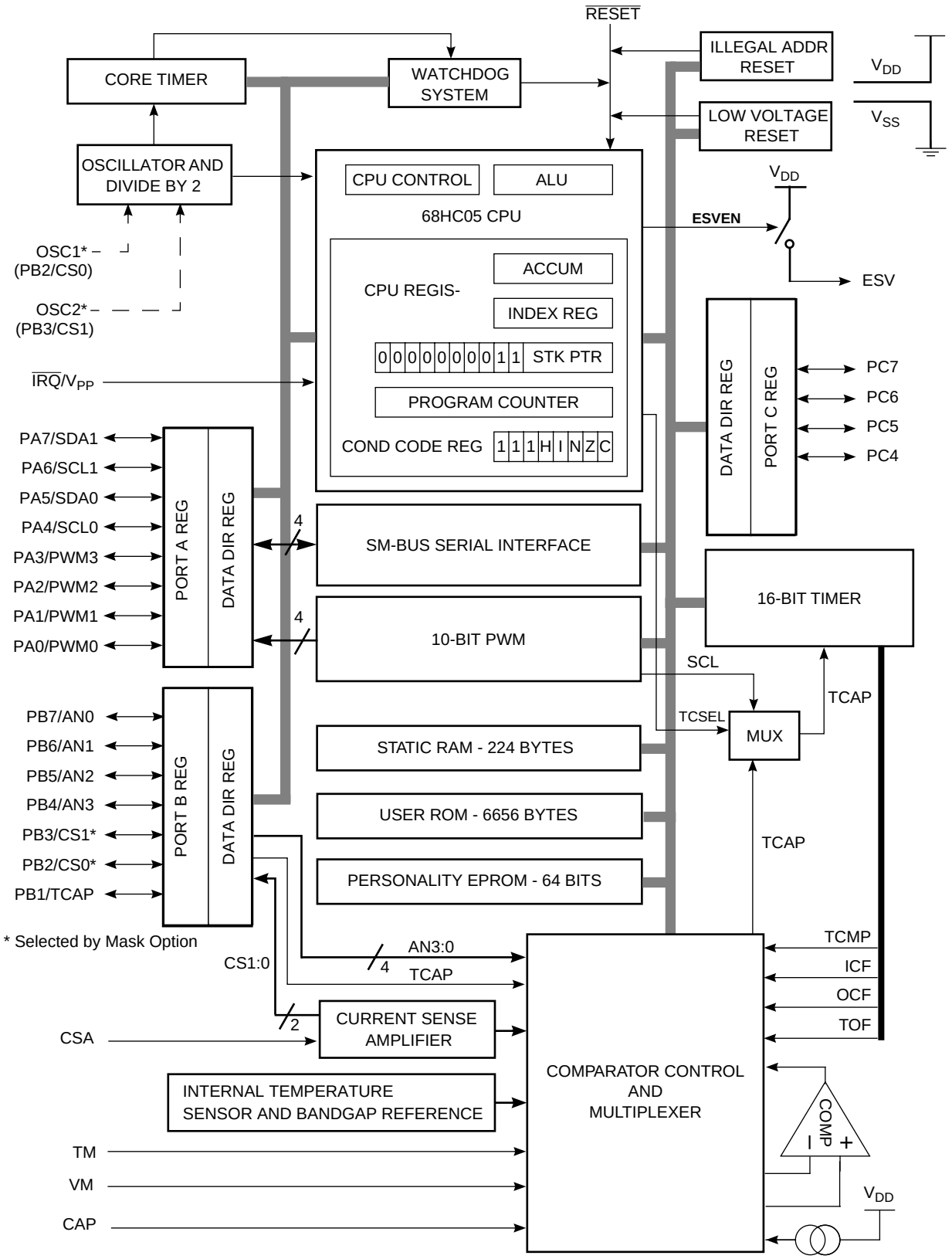


Figure 1-1. MC68HC05SB7 Block Diagram

1.5 PIN ASSIGNMENTS

The MC68HC05SB7 is available in 28-pin SSOP package. The pin assignments are shown in **Figure 1-2**.

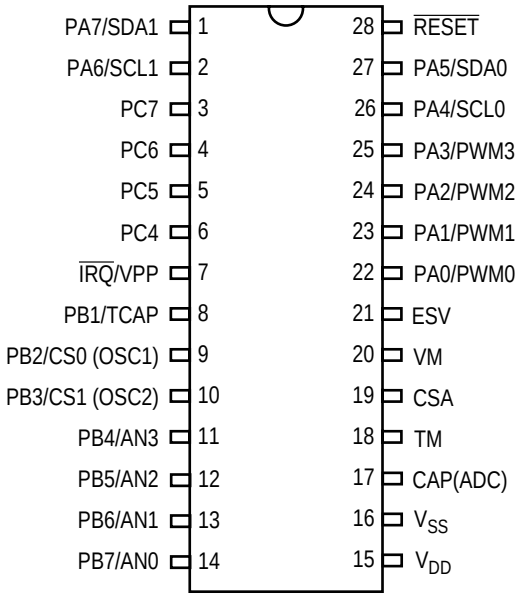


Figure 1-2. MC68HC05SB7 Pin Assignments

1.6 FUNCTIONAL PIN DESCRIPTION

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

1.6.1 V_{DD}, V_{SS}

Power is supplied to the MCU through V_{DD} and V_{SS}. V_{DD} is the positive supply, and V_{SS} is ground. The MCU operates from a single power supply.

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

1.6.2 OSC1, OSC2

When selected by a mask option, the OSC1 and OSC2 pins are the connections for the external pin oscillator (EPO). The OSC1 and OSC2 pins can accept the following sets of components:

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

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

Crystal Oscillator

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

(a) Crystal or Ceramic Resonator Connections

(b) External Clock Source Connection

Figure 1-3. EPO Oscillator Connections

Ceramic Resonator Oscillator

In cost-sensitive applications, a ceramic resonator can be used in place of the crystal. The circuit in **Figure 1-3**(a) can be used for a ceramic resonator. The resonator manufacturer's recommendations should be followed, 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. The ceramic resonator and components should be mounted as close as possible to the pins for start-up stabilization and to minimize output distortion.

External Clock

An external clock from another CMOS-compatible device can be connected to the OSC1 input, with the OSC2 input not connected, as shown in **Figure 1-3**(b). This configuration is possible regardless of whether the crystal/ceramic resonator or internal VCO is enabled.

1.6.3 $\overline{IRQ}/V_{PP}$

The $\overline{IRQ}/V_{PP}$ input pin drives the asynchronous IRQ interrupt function of the CPU. The IRQ interrupt function has a bit to provide either only negative edge-sensitive triggering or both negative edge-sensitive and low level-sensitive triggering. If the

option is selected to include level-sensitive triggering, the $\overline{IRQ}/V_{PP}$ input requires an external resistor to V_{DD} for "wired-OR" operation, if desired. If the $\overline{IRQ}/V_{PP}$ pin is not used, it must be tied to the V_{DD} supply. The $\overline{IRQ}/V_{PP}$ pin contains an internal Schmitt trigger as part of its input to improve noise immunity. The voltage on this pin may affect the mode of operation and should not exceed V_{DD}.

The $\overline{IRQ}/V_{PP}$ pin is also used for programming voltage when programming the Personality EPROM.

See section on Interrupts for more details.

1.6.4 $\overline{RESET}$

The $\overline{RESET}$ pin can be used as an input to reset the MCU to a known start-up state by pulling it to the low state. It also functions as an output to indicate that an internal COP watchdog, illegal address, or low voltage reset has occurred. The $\overline{RESET}$ pin contains a pullup device to allow the pin to be left disconnected without an external pullup resistor. The $\overline{RESET}$ pin also contains a steering diode that, when the power is removed, will discharge to V_{DD} any charge left on an external capacitor connected between the $\overline{RESET}$ pin and V_{SS}. The $\overline{RESET}$ pin also contains an internal Schmitt trigger to improve its noise immunity as an input.

See section on Resets for more details.

1.6.5 CSA

This pin is the input to the current sense amplifier. Usually one terminal of the current path shunt sensing resistor of 0.01Ω is connected to this input pin. The other terminal is connected to V_{SS}.

See section on Current Sense Amplifier for more details.

1.6.6 TM

This pin is fed from the output of an external temperature sensor. Usually a thermistor with a resistor forms a voltage divider with the voltage value applied to this input.

See section on Temperature Sensor for more details.

1.6.7 VM

This pin is the battery voltage input of the voltage measurement circuit.

See section on Temperature Sensor for more details.

1.6.8 CAP (ADC)

This pin is connected to an external ramp capacitor to form the slope voltage converter.

See section on Analog Subsystem for more details.

1.6.9 ESV

This pin provides a switchable 5mA at V_{OH} (at worst case) to an external EEPROM. The ESVEN bit in the Miscellaneous Control Register enables/disables the ESV pin.

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
MCR	R	TSEN	LVRON	0	SCLK	CSSEL	TCSEL	ESVEN	SMINLEV
$000B	W			COPON					
reset:		0	1	0	0	0	0	0	0

U = UNAFFECTED BY RESET

ESVEN — ESV Enable

This read/write bit selects whether ESV is enable or not. Reset clears the ESVEN bit.

1 = ESV enabled.
0 = ESV disabled.

1.6.10 PA0-PA7 / PWM0-PWM3, SCL0-SCL1, SDA0-SDA1

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

PA0-PA3 are multiplexed with PWM outputs PWM0-PWM3. PA4-PA7 are multiplexed with the two SM-Bus channels - SCL0, SDA0 and SCL1, SDA1.

1.6.11 PB1-PB7 / TCAP, CS0-CS1, AN0-AN3

Pins PB2/CS0 and PB3/CS1 are only available when selected by a mask option.

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

PB1 is configured as the TCAP input pin for the 16-bit timer after a reset, and is disabled by setting the ICEN bit in the Analog Control Register ($1D).

PB2 and PB3 (when selected) are multiplexed with CS0 and CS1 respectively, from the current sense interrupt circuit. See section on Current Sense Amplifier for more details.

PB4-PB7 are multiplexed with the analog input pins of the A/D converter. See section on Analog Subsystem for more details.

1.6.12 PC4-PC7

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

SECTION 2
MEMORY

This section describes the organization of the MC68HC05SB7 on-chip memory.

2.1 MEMORY MAP

In Normal operating mode, the 48 bytes of I/O, 224 bytes of user RAM and 6144 bytes of user ROM are all active as shown in **Figure 2-1**. The ROM portion of memory holds the program instructions, fixed data, user defined vectors, and interrupt service routines. The RAM portion of memory holds variable data. I/O registers are memory mapped so that the CPU can access their locations in the same way that it accesses all other memory locations.

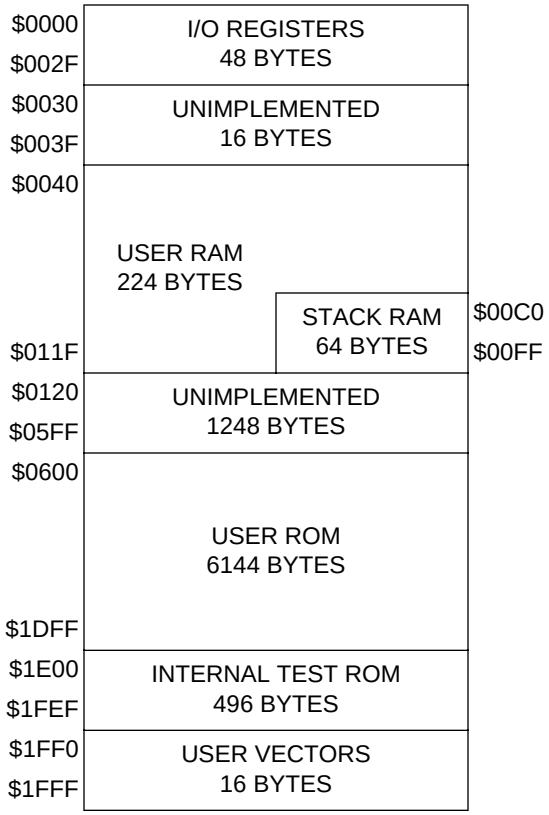


Figure 2-1. MC68HC05SB7 Memory Map

2.2 INPUT/OUTPUT SECTION

The first 48 addresses of the memory space, $0000 – $002F, are the I/O section as summarized in **Figure 2-2**. These are the addresses of the I/O control registers, status registers, and data registers. Reading from unimplemented locations will return unknown states, and writing to unimplemented locations will be ignored.

One I/O register is located outside the 48-byte I/O section which is the computer operating properly (COP) register, mapped at $1FF0.

The assignment of each control, status, and data bit in the I/O register space from $0000 through $002F are given in **Figure 2-3**, **Figure 2-4**, and **Figure 2-5**.

Addr.	Register Name
$0000	Port A Data Register
$0001	Port B Data Register
$0002	Port C Data Register
$0003	Analog MUX Register 1
$0004	Port A Data Direction Register
$0005	Port B Data Direction Register
$0006	Port C Data Direction Register
$0007	Analog MUX Register 2
$0008	Core Timer Status & Control Register
$0009	Core Timer Counter
$000A	CSA Status/Control Register
$000B	Miscellaneous Control Register
$000C	VCO Adjust Register
$000D	IRQ Status & Control Register
$000E	Personality EPROM Bit Select Register
$000F	Personality EPROM Status & Control Reg.
$0010	Reserved
$0011	Reserved
$0012	Timer Control Register
$0013	Timer Status Register
$0014	Input Capture Register MSB
$0015	Input Capture Register LSB
$0016	Output Compare Register MSB
$0017	Output Compare Register LSB

Addr.	Register Name
$0018	Timer Counter Register MSB
$0019	Timer Counter Register LSB
$001A	Alternate Counter Register MSB
$001B	Alternate Counter Register LSB
$001C	Reserved
$001D	Analog Control Register
$001E	Analog Status Register
$001F	Reserved
$0020	SM-Bus Address Register
$0021	SM-Bus Frequency Select Register
$0022	SM-Bus Control Register
$0023	SM-Bus Status Register
$0024	SM-Bus Data Register
$0025	D/A Register 0 H
$0026	D/A Register 0 L
$0027	D/A Register 1 H
$0028	D/A Register 1 L
$0029	D/A Register 2 H
$002A	D/A Register 2 L
$002B	D/A Register 3 H
$002C	D/A Register 3 L
$002D	MUX Channel Enable Register
$002E	Reserved
$002F	Reserved

Figure 2-2. MC68HC05SB7 I/O Registers

ADDR	REGISTER	R/W	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
$0000	Port A Data PORTA	R W	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
$0001	Port B Data PORTB	R W	PB7	PB6	PB5	PB4	PB3	PB2	PB1	
$0002	Port C Data PORTC	R W	PC7	PC6	PC5	PC4				
$0003	Analog MUX 1 AMUX1	R W	HOLD	DHOLD	INV	VREF	MUX3	MUX2	MUX1	MUX0
$0004	Port A Data Direction DDRA	R W	DDRA7	DDRA6	DDRA5	DDRA4	DDRA3	DDRA2	DDRA1	DDRA0
$0005	Port B Data Direction DDRB	R W	DDRB7	DDRB6	DDRB5	DDRB4	DDRB3	DDRB2	DDRB1	0
$0006	Port C Data Direction DDRC	R W	DDRC7	DDRC6	DDRC5	DDRC4	0	0	0	0
$0007	Analog MUX 2 AMUX2	R W	0		0	IREF	MUX7	MUX6	MUX5	MUX4
$0008	CTimer Status/Ctrl CTSCR	R W	CTOF	RTIF	CTOFE	RTIE	0 CTOFR	0 RTIFR	RT1	RT0
$0009	CTimer Counter CTCR	R W	TMR7	TMR6	TMR5	TMR4	TMR3	TMR2	TMR1	TMR0
$000A	CSA Status/Control CSASCR	R W	CSEN	X30	X10	CSCAL	CDEN	CDIE	0 CSIFR	CSIF
$000B	Misc Control MCR	R W	TSEN	LVRON	0 COPON	SCLK	CSSEL	TCSEL	ESVEN	SMINLEV
$000C	VCO Adjust VAR	R W				VA4	VA3	VA2	VA1	VA0
$000D	IRQ Status/Ctrl ISCR	R W	IRQE	VCOEN	LEVEL	0	IRQF	0	0 IRQR	0
$000E	PEPROM Bit Select PEBSR	R W	PEB7	PEB7	PEB7	PEB4	PEB3	PEB2	PEB1	PEB0
$000F	PEPROM Status/Ctrl PESCR	R W	PEDATA	0	PEPGM	0	0	0	0	PEPZRF

unimplemented bits reserved bits

Figure 2-3. MC68HC05SB7 I/O Registers $0000-$000F

ADDR	REGISTER	R/W	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
$0010	Reserved	R W								
$0011	Reserved	R W								
$0012	Timer Control TCR	R W	ICIE	OCIE	TOIE	0	0	0	IEDG	OLVL
$0013	Timer Status TSR	R W	ICF	OCF	TOF	0	0	0	0	0
$0014	Input Capture MSB ICRH	R W	ICRH7	ICRH6	ICRH5	ICRH4	ICRH3	ICRH2	ICRH1	ICRH0
$0015	Input Capture LSB ICRL	R W	ICRL7	ICRL6	ICRL5	ICRL4	ICRL3	ICRL2	ICRL1	ICRL0
$0016	Output Compare MSB OCRH	R W	OCRH7	OCRH6	OCRH5	OCRH4	OCRH3	OCRH2	OCRH1	OCRH0
$0017	Output Compare LSB OCRL	R W	OCRL7	OCRL6	OCRL5	OCRL4	OCRL3	OCRL2	OCRL1	OCRL0
$0018	Timer Counter MSB TMRH	R W	TMRH7	TMRH6	TMRH5	TMRH4	TMRH3	TMRH2	TMRH1	TMRH0
$0019	Timer Counter LSB TMRL	R W	TMRL7	TMRL6	TMRL5	TMRL4	TMRL3	TMRL2	TMRL1	TMRL0
$001A	Alter. Counter MSB ACRH	R W	ACRH7	ACRH6	ACRH5	ACRH4	ACRH3	ACRH2	ACRH1	ACRH0
$001B	Alter. Counter LSB ACRL	R W	ACRL7	ACRL6	ACRL5	ACRL4	ACRL3	ACRL2	ACRL1	ACRL0
$001C	Reserved	R W								
$001D	Analog Control ACR	R W	CHG	ATD2	ATD1	ICEN	CPIE	CPEN		ISEN
$001E	Analog Status ASR	R W	CPF		0 CPFR	0	0	0	0	0
$001F	Reserved	R W								

unimplemented bits reserved bits

Figure 2-4. MC68HC05SB7 I/O Registers $0010-$001F

ADDR	REGISTER	R/W	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
$0020	SM-Bus Address SMADR	R W	SMAD7	SMAD6	SMAD5	SMAD4	SMAD3	SMAD2	SMAD1	
$0021	SM-Bus Freq. Sel. SMFDR	R W				FD4	FD3	FD2	FD1	FD0
$0022	SM-Bus Control SMCR	R W	SMEN	SMIEN	SMSTA	SMTX	TXAK	SMUX		
$0023	SM-Bus Status	R	SMCF	SMAAS	SMBB	SMAL		SRW	SMIF	RXAK
	SMSR	W				SMAL clr			SMIF clr	
$0024	SM-Bus Data SMDR	R W	SMD7	SMD6	SMD5	SMD4	SMD3	SMD2	SMD1	SMD0
$0025	D/A Register 0	R								
	DAC0	W	D9	D8	D7	D6	D5	D4	D3	D2
$0026	D/A Register 0	R								
	DAC0	W							D1	D0
$0027	D/A Register 1	R								
	DAC1	W	D9	D8	D7	D6	D5	D4	D3	D2
$0028	D/A Register 1	R								
	DAC1	W							D1	D0
$0029	D/A Register 2	R								
	DAC2	W	D9	D8	D7	D6	D5	D4	D3	D2
$002A	D/A Register 2	R								
	DAC2	W							D1	D0
$002B	D/A Register 3	R								
	DAC3	W	D9	D8	D7	D6	D5	D4	D3	D2
$002C	D/A Register 3	R								
	DAC3	W							D1	D0
$002D	MUX Channel Enable MCER	R W	PWM_I				DA3-E	DA2-E	DA1-E	DA0-E
$002E	Reserved	R W								
$002F	Reserved	R W								

unimplemented bits reserved bits

Figure 2-5. MC68HC05SB7 I/O Registers $0020-$002F

ADDR	REGISTER	R/W	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
$1FF0	COP Register	R	0	0	0	0	0	0	0	0
	COPR	W								COPC

Figure 2-6. COP Register (COPR)

2.3 INTERRUPT VECTOR MAPPING

The interrupt vectors are contained in the upper memory addresses above $1FF0 as shown in **Figure 2-2**.

Addr.	Register Name
$1FF0	CDET INTERRUPT VECTOR (MSB)
$1FF1	CDET INTERRUPT VECTOR (LSB)
$1FF2	ANALOG INTERRUPT VECTOR (MSB)
$1FF3	ANALOG INTERRUPT VECTOR (LSB)
$1FF4	SM-BUS INTERRUPT VECTOR (MSB)
$1FF5	SM-BUS INTERRUPT VECTOR (LSB)
$1FF6	TIMER INTERRUPT VECTOR (MSB)
$1FF7	TIMER INTERRUPT VECTOR (LSB)
$1FF8	CTIMER INTERRUPT VECTOR (MSB)
$1FF9	CTIMER INTERRUPT VECTOR (LSB)
$1FFA	EXTERNAL IRQ VECTOR (MSB)
$1FFB	EXTERNAL IRQ VECTOR (LSB)
$1FFC	SWI VECTOR (MSB)
$1FFD	SWI VECTOR (LSB)
$1FFE	RESET VECTOR (MSB)
$1FFF	RESET VECTOR(LSB)

Figure 2-7. MC68HC05SB7 Interrupt Vector Mapping

2.4 ROM

There are a total of 6160 bytes of ROM on chip. This includes 6144 bytes of user ROM with locations $0600 through $1DFF for the user program storage and another 16 bytes for user vectors at locations $1FF0 through $1FFF.

2.5 RAM

The 224 addresses from $0040 to $011F serve as both the user RAM and the stack RAM. The stack begins at address $00C0 and proceeds down to $00FF. The stack pointer can access 64 locations from $00C0 to $00FF. Using the stack area for data storage or temporary work locations requires care to prevent it from being over written due to stacking from an interrupt or subroutine call. The CPU uses five RAM bytes to save all CPU register contents before processing an interrupt. During a subroutine call, the CPU uses two bytes to store the return address. The stack pointer decrements during pushes and increments during pulls.

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.

SECTION 3
CENTRAL PROCESSING UNIT

The MC68HC05SB7 has an 8k-bytes memory map. The stack has only 64 bytes. Therefore, the stack pointer has been reduced to only 6 bits and will only decrement down to $00C0 and then wrap-around to $00FF. All other instructions and registers behave as described in this chapter.

3.1 REGISTERS

The MCU contains five registers which are hard-wired within the CPU and are not part of the memory map. These five registers are shown in **Figure 3-1** and are described in the following paragraphs.

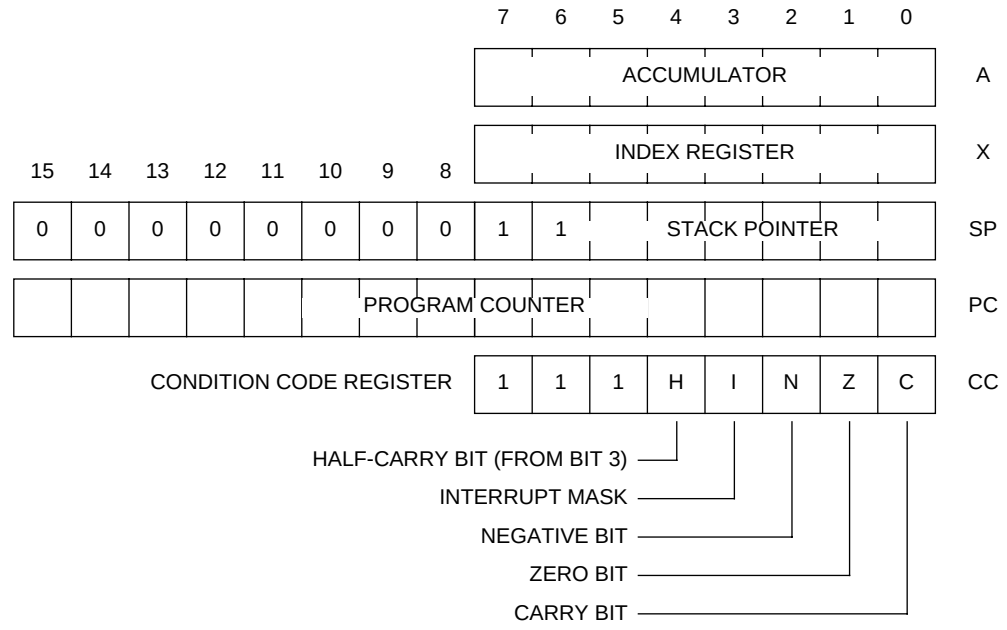


Figure 3-1. MC68HC05 Programming Model

3.2 ACCUMULATOR (A)

The accumulator is a general purpose 8-bit register as shown in **Figure 3-1**. The CPU uses the accumulator to hold operands and results of arithmetic calculations or non-arithmetic operations. The accumulator is not affected by a reset of the device.

3.3 INDEX REGISTER (X)

The index register shown in **Figure 3-1** is an 8-bit register that can perform two functions:

- Indexed addressing
- Temporary storage

In indexed addressing with no offset, the index register contains the low byte of the operand address, and the high byte is assumed to be $00. In indexed addressing with an 8-bit offset, the CPU finds the operand address by adding the index register content to an 8-bit immediate value. In indexed addressing with a 16-bit offset, the CPU finds the operand address by adding the index register content to a 16-bit immediate value.

The index register can also serve as an auxiliary accumulator for temporary storage. The index register is not affected by a reset of the device.

3.4 STACK POINTER (SP)

The stack pointer shown in **Figure 3-1** is a 16-bit register. In MCU devices with memory space less than 64k-bytes the unimplemented upper address lines are ignored. The stack pointer contains the address of the next free location on the stack. During a reset or the reset stack pointer (RSP) instruction, the stack pointer is set to $00FF. The stack pointer is then decremented as data is pushed onto the stack and incremented as data is pulled off the stack.

When accessing memory, the ten most significant bits are permanently set to 0000000011. The six least significant register bits are appended to these ten fixed bits to produce an address within the range of $00FF to $00C0. Subroutines and interrupts may use up to 64($C0) locations. If 64 locations are exceeded, the stack pointer wraps around and overwrites the previously stored information. A subroutine call occupies two locations on the stack and an interrupt uses five locations.

3.5 PROGRAM COUNTER (PC)

The program counter shown in **Figure 3-1** is a 16-bit register. In MCU devices with memory space less than 64k-bytes the unimplemented upper address lines are ignored. The program counter contains the address of the next instruction or operand to be fetched.

Normally, the address in the program counter 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.

3.6 CONDITION CODE REGISTER (CCR)

The CCR shown in **Figure 3-1** is a 5-bit register in which four bits are used to indicate the results of the instruction just executed. The fifth bit is the interrupt mask. These bits can be individually tested by a program, and specific actions can be taken as a result of their states. The condition code register should be thought of as having three additional upper bits that are always ones. Only the interrupt mask is affected by a reset of the device. The following paragraphs explain the functions of the lower five bits of the condition code register.

3.6.1 Half Carry Bit (H-Bit)

When the half-carry bit is set, it means that a carry occurred between bits 3 and 4 of the accumulator during the last ADD or ADC (add with carry) operation. The half-carry bit is required for binary-coded decimal (BCD) arithmetic operations.

3.6.2 Interrupt Mask (I-Bit)

When the interrupt mask is set, the internal and external interrupts are disabled. Interrupts are enabled when the interrupt mask is cleared. When an interrupt occurs, the interrupt mask is automatically set after the CPU registers are saved on the stack, but before the interrupt vector is fetched. If an interrupt request occurs while the interrupt mask is set, the interrupt request is latched. Normally, the interrupt is processed as soon as the interrupt mask is cleared.

A return from interrupt (RTI) instruction pulls the CPU registers from the stack, restoring the interrupt mask to its state before the interrupt was encountered. After any reset, the interrupt mask is set and can only be cleared by the Clear I-Bit (CLI), or WAIT instructions.

3.6.3 Negative Bit (N-Bit)

The negative bit is set when the result of the last arithmetic operation, logical operation, or data manipulation was negative. (Bit 7 of the result was a logical one.)

The negative bit can also be used to check an often tested flag by assigning the flag to bit 7 of a register or memory location. Loading the accumulator with the contents of that register or location then sets or clears the negative bit according to the state of the flag.

3.6.4 Zero Bit (Z-Bit)

The zero bit is set when the result of the last arithmetic operation, logical operation, data manipulation, or data load operation was zero.

3.6.5 Carry/Borrow Bit (C-Bit)

The carry/borrow bit is set when a carry out of bit 7 of the accumulator occurred during the last arithmetic operation, logical operation, or data manipulation. The carry/borrow bit is also set or cleared during bit test and branch instructions and during shifts and rotates. This bit is neither set by an INC nor by a DEC instruction.

SECTION 4
INTERRUPTS

An interrupt temporarily stops normal program execution to process a particular event. An interrupt does not stop the execution of the instruction in progress, 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 vector address.

4.1 INTERRUPT VECTORS

Table 4-1. Reset/Interrupt Vector Addresses

Function	Source	Control Bit	Global Hardware Mask	Local Software Mask	Priority (1 = Highest)	Vector Address
Reset	Power-On Logic $\overline{\text{RESET}}$ Pin Low Voltage Reset Illegal Address Reset	—	—	—	1	$1FFE–$1FFF
	COP Watchdog	COPON[1]				
Software Interrupt (SWI)	User Code	—	—	—	Same Priority As Instruction	$1FFC–$1FFD
External Interrupt (IRQ)	$\overline{\text{IRQ}}/V_{PP}$ Pin	—	I Bit	IRQE Bit	2	$1FFA–$1FFB
Core Timer Interrupts	TOF Bit RTIF Bit	—	I Bit	TOFE Bit RTIE Bit	3	$1FF8–$1FF9
Programmable Timer Interrupts	ICF Bit OCF Bit TOF Bit	—	I Bit	ICIE Bit OCIE Bit TOIE Bit	4	$1FF6–$1FF7
SM-Bus Interrupt	SMIF Bit	—	I Bit	SMIE Bit	5	$1FF4–$1FF5
Analog Interrupt	CPF1 Bit CPF2 Bit	—	I Bit	CPIE Bit	6	$1FF2–$1FF3
Current Detect Interrupt	CIF Bit	—	I Bit	CIE Bit	7	$1FF0–$1FF1

1. COPON enables the COP watchdog timer

Table 4-1 summarizes the reset and interrupt sources and vector assignments.

NOTE

If more than one interrupt request is pending, the CPU fetches the vector of the higher priority interrupt first. A higher priority interrupt does not actually interrupt a lower priority interrupt service routine unless the lower priority interrupt service routine clears the I bit.

4.2 INTERRUPT PROCESSING

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

- Stores the CPU registers on the stack in the order shown in **Figure 4-1**.
- 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 as shown in **Table 4-1**.

The return from interrupt (RTI) instruction causes the CPU to recover its register contents from the stack as shown in **Figure 4-1**. The sequence of events caused by an interrupt are shown in the flow chart in **Figure 4-2**.

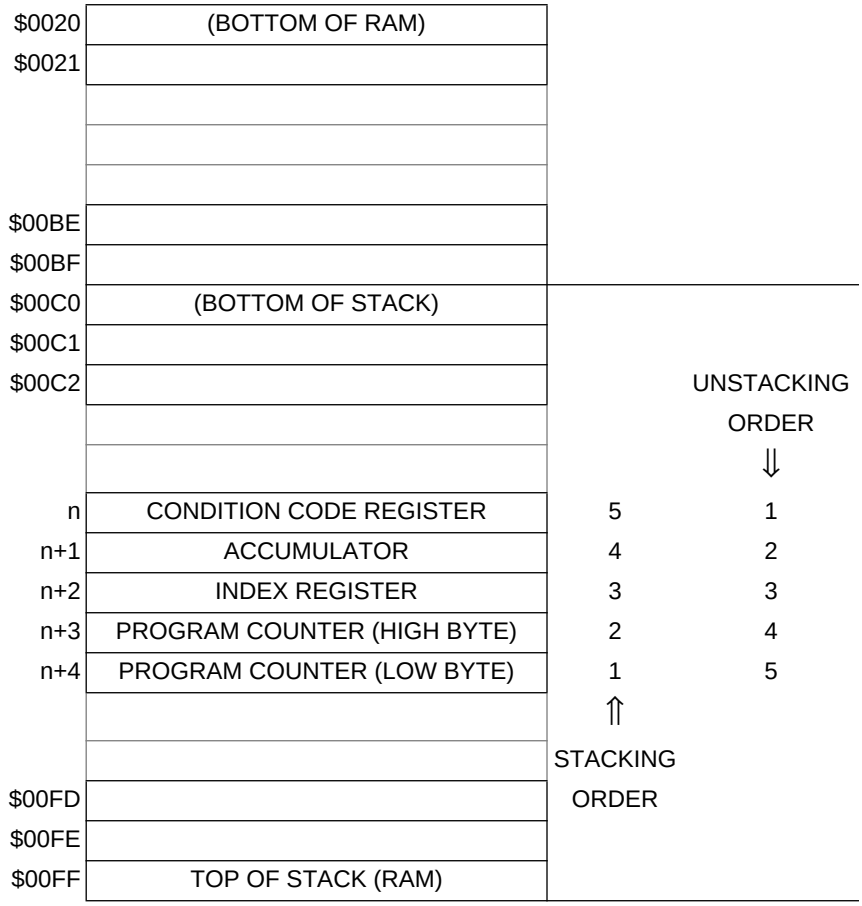


Figure 4-1. Interrupt Stacking Order

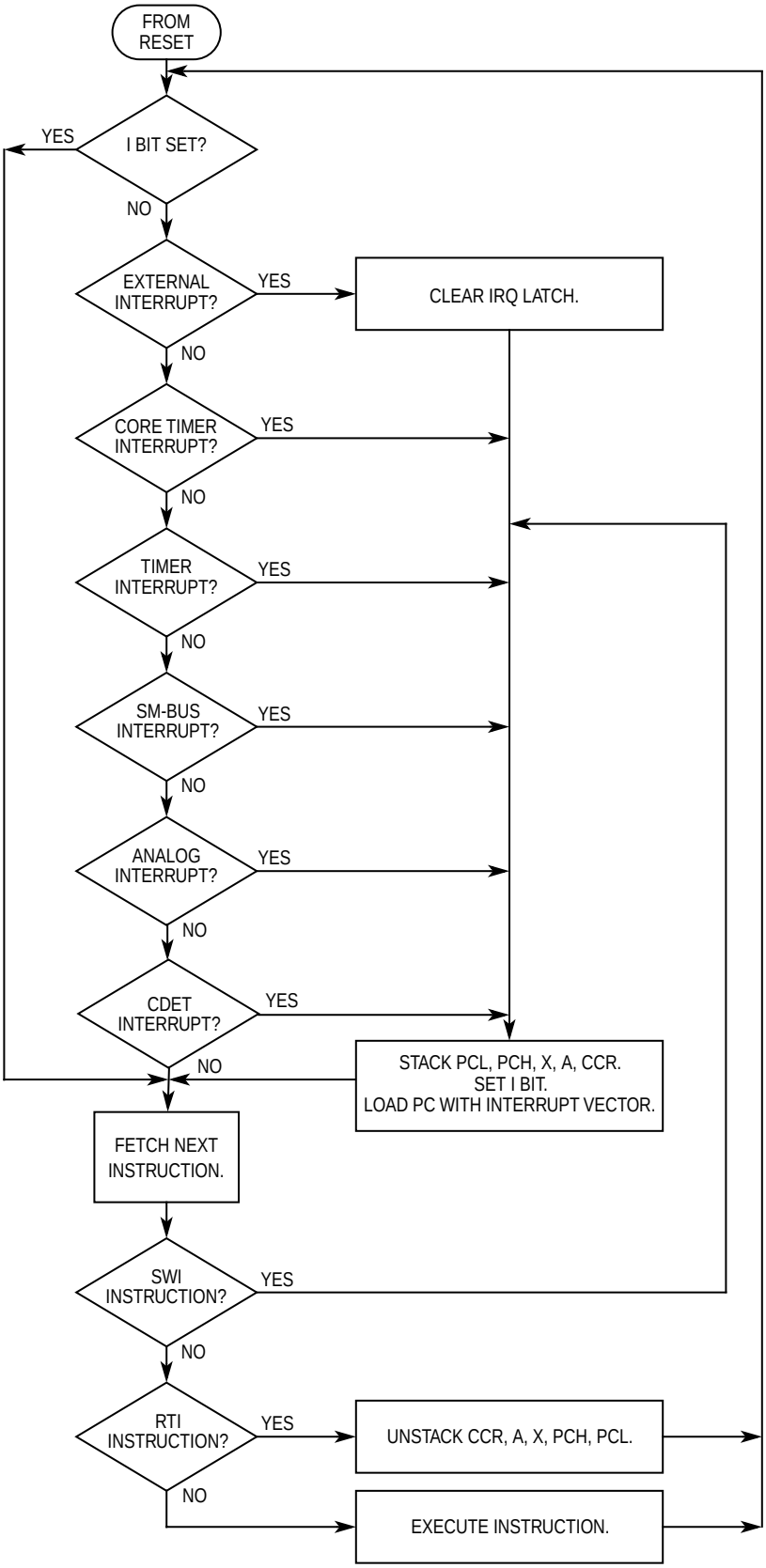


Figure 4-2. Interrupt Flow Chart

4.3 SOFTWARE INTERRUPT

The software interrupt (SWI) instruction causes a nonmaskable interrupt.

4.4 EXTERNAL INTERRUPT

The $\overline{IRQ}/V_{PP}$ pin is the source that generates external interrupt. Setting the I bit in the condition code register or clearing the IRQE bit in the interrupt status and control register disables this external interrupt.

4.4.1 $\overline{IRQ}/V_{PP}$ Pin

An interrupt signal on the $\overline{IRQ}/V_{PP}$ pin latches an external interrupt request. To help clean up slow edges, the input from the $\overline{IRQ}/V_{PP}$ pin is processed by a Schmitt trigger gate. 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 and the IRQE bit in the IRQ status and control register (ISCR). If the I bit is clear and the IRQE bit is set, then the CPU begins the interrupt sequence. 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 4-3** shows the logic for external interrupts.

NOTE

If the $\overline{IRQ}/V_{PP}$ pin is not in use, it should be connected to the V_{DD} pin.

The $\overline{IRQ}/V_{PP}$ pin can be negative edge-triggered only or negative edge- and low-level-triggered. External interrupt sensitivity is programmed with the LEVEL bit.

With the edge- and level-sensitive trigger option, a falling edge or a low level on the $\overline{IRQ}/V_{PP}$ pin latches an external interrupt request. The edge- and level-sensitive trigger option allows connection to the $\overline{IRQ}/V_{PP}$ pin of multiple wired-OR interrupt sources. As long as any source is holding the $\overline{IRQ}/V_{PP}$ low, an external interrupt request is present, and the CPU continues to execute the interrupt service routine.

With the edge-sensitive-only trigger option, a falling edge on the $\overline{IRQ}/V_{PP}$ pin latches an external interrupt request. A subsequent interrupt request can be latched only after the voltage level on the $\overline{IRQ}/V_{PP}$ pin returns to a logic one and then falls again to logic zero.

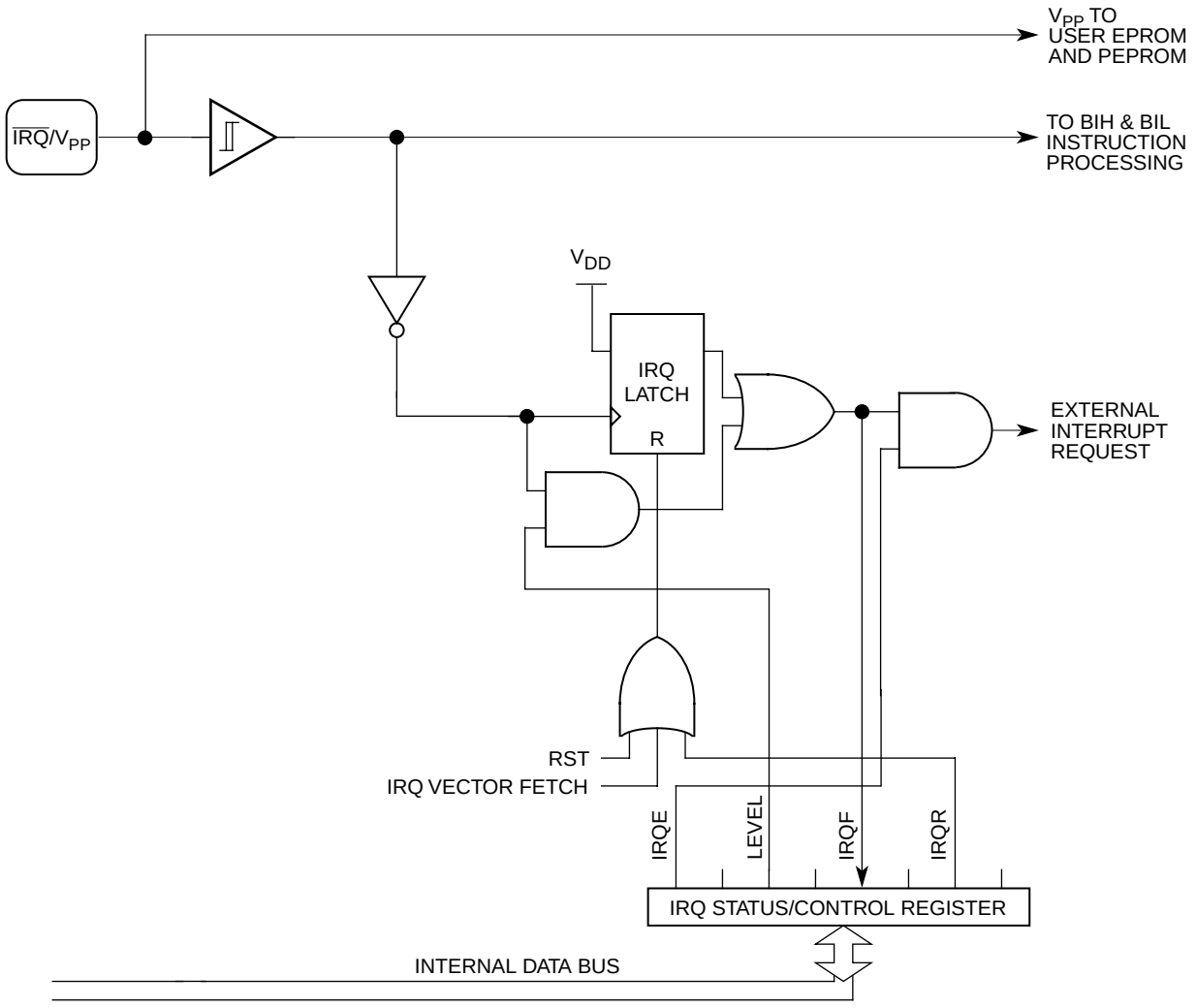


Figure 4-3. External Interrupt Logic

4.4.2 IRQ Status and Control Register (ISCR)

The IRQ status and control register (ISCR), shown in **Figure 4-4**, contains an external interrupt mask (IRQE), an external interrupt flag (IRQF), and a flag reset bit (IRQR). Unused bits will read as logic zeros. Reset sets the IRQE bit and clears all the other bits.

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
ISCR	R	IRQE	VCOEN	LEVEL	0	IRQF	0	0	0
$000D	W							IRQR	
reset:		1	1	0	0	0	0	0	0

Figure 4-4. IRQ Status and Control Register (ISCR)

IRQE — External Interrupt Request Enable

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

1 = External interrupt processing enabled.
0 = External interrupt processing disabled.

VCOEN — VCO Enable

Please refer to section on System Clock.

LEVEL — External Interrupt Sensitivity

This bit makes the external interrupt inputs level-triggered as well as edge-triggered.

1 = $\overline{IRQ}/V_{PP}$ pin negative edge-triggered and low level-triggered.
0 = $\overline{IRQ}/V_{PP}$ pin negative edge-triggered only.

IRQF — External Interrupt Request Flag

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

1 = Interrupt request pending.
0 = No interrupt request pending.

The following condition set the IRQ flag:

- An external interrupt signal on the $\overline{IRQ}/V_{PP}$ pin.

The following conditions clear the IRQ flag:

- When the CPU fetches the interrupt vector.
- When a logic “1” is written to the IRQR bit.

IRQR — Interrupt Request Reset

This write-only bit clears the IRQF flag bit and prevents redundant execution of interrupt routines. Writing a logic one to IRQR clears the IRQF. Writing a logic zero to IRQR has no effect. IRQR always reads as a logic zero. Reset has no affect on IRQR.

1 = Clear IRQF flag bit.
0 = No effect.

4.5 CORE TIMER INTERRUPTS

The Core Timer can generate the following interrupts:

- Timer overflow interrupt.
- Real-time interrupt.

Setting the I bit in the condition code register disables Core Timer interrupts. The controls and flags for these interrupts are in the Core Timer status and control register (CTSCR) located at $0008.

4.5.1 Core Timer Overflow Interrupt

An overflow interrupt request occurs if the Core Timer overflow flag (TOF) becomes set while the Core Timer overflow interrupt enable bit (TOFE) is also set. The TOF flag bit can be reset by writing a logical one to the CTOFR bit in the CTSCR or by a reset of the device.

4.5.2 Real-Time Interrupt

A real-time interrupt request occurs if the real-time interrupt flag (RTIF) becomes set while the real-time interrupt enable bit (RTIE) is also set. The RTIF flag bit can be reset by writing a logical one to the RTIFR bit in the CTSCR or by a reset of the device.

4.6 PROGRAMMABLE TIMER INTERRUPTS

The 16-bit programmable Timer can generate an interrupt whenever the following events occur:

- Input capture.
- Output compare.
- Timer counter overflow.

Setting the I bit in the condition code register disables Timer interrupts. The controls for these interrupts are in the Timer control register (TCR) located at $0012 and in the status bits are in the Timer status register (TSR) located at $0013.

4.6.1 Input Capture Interrupt

An input capture interrupt occurs if the input capture flag (ICF) becomes set while the input capture interrupt enable bit (ICIE) is also set. The ICF flag bit is in the TSR; and the ICIE enable bit is located in the TCR. The ICF flag bit is cleared by a read of the TSR with the ICF flag bit is set; and then followed by a read of the LSB of the input capture register (ICRL) or by reset. The ICIE enable bit is unaffected by reset.

4.6.2 Output Compare Interrupt

An output compare interrupt occurs if the output compare flag (OCF) becomes set while the output compare interrupt enable bit (OCIE) is also set. The OCF flag bit is in the TSR and the OCIE enable bit is in the TCR. The OCF flag bit is cleared by a read of the TSR with the OCF flag bit set; and then followed by an access to the LSB of the output compare register (OCRL) or by reset. The OCIE enable bit is unaffected by reset.

4.6.3 Timer Overflow Interrupt

A Timer overflow interrupt occurs if the Timer overflow flag (TOF) becomes set while the Timer overflow interrupt enable bit (TOIE) is also set. The TOF flag bit is in the TSR and the TOIE enable bit is in the TCR. The TOF flag bit is cleared by a

read of the TSR with the TOF flag bit set; and then followed by an access to the LSB of the timer registers (TMRL) or by reset. The TOIE enable bit is unaffected by reset.

4.7 SM-BUS INTERRUPT

There is one SM-Bus interrupt flag that causes SM-Bus interrupt whenever it is set and enabled. The interrupt flags is in the SM-Bus Status Register (SMSR) and the enable bit is in SM-Bus Control Register (SMCR). SM-Bus interrupt can wake up MCU from WAIT mode.

4.8 ANALOG INTERRUPTS

The analog subsystem can generate the following interrupts:

- Voltage on positive input of comparator is greater than the voltage on the negative input of comparator.
- Trigger of the input capture interrupt from the programmable Timer as described in **Section 4.6** above.

Setting the I bit in the condition code register disables analog subsystem interrupts. The controls for these interrupts are in the analog subsystem control register (ACR) located at $001D and the status bits are in the analog subsystem status register (ASR) located at $001E.

4.8.1 Comparator Input Match Interrupt

A comparator input match interrupt occurs if the compare flag bit (CPF) in the ASR becomes set while the comparator interrupt enable bit (CPIE) in the ACR is also set. Reset clears these bits.

4.8.2 Input Capture Interrupt

The analog subsystem can also generate an input capture interrupt through the programmable Timer. The input capture can be triggered when there is a match in the input conditions for the voltage comparator. If comparator sets the CPF flag bit in the ASR and the input capture enable (ICEN) in the ACR is set then an input capture will be performed by the programmable Timer. If the ICIE enable bit in the TCR is also set then an input compare interrupt will occur. Reset clears these bits.

NOTE

In order for the analog subsystem to generate an interrupt using the input capture function of the programmable Timer the ICEN enable bit in the ACR and the ICIE enable bit in the TCR must both be set.

4.9 CURRENT DETECT INTERRUPT

The Current Sense Amplifier circuit can be configured to generate an interrupt once it detects a current passing through the current sensing resistor.

SECTION 5
RESETS

This section describes the five reset sources and how they initialize the MCU. A reset immediately stops the operation of the instruction being executed, initializes certain control bits, and loads the program counter with a user defined reset vector address. The following conditions produce a reset:

- Initial power up of device (power on reset).
- A logic zero applied to the $\overline{\text{RESET}}$ pin (external reset).
- Timeout of the COP watchdog (COP reset).
- Low voltage applied to the device (LVR reset).
- Fetch of an opcode from an address not in the memory map (illegal address reset).

Figure 5-1 shows a block diagram of the reset sources and their interaction.

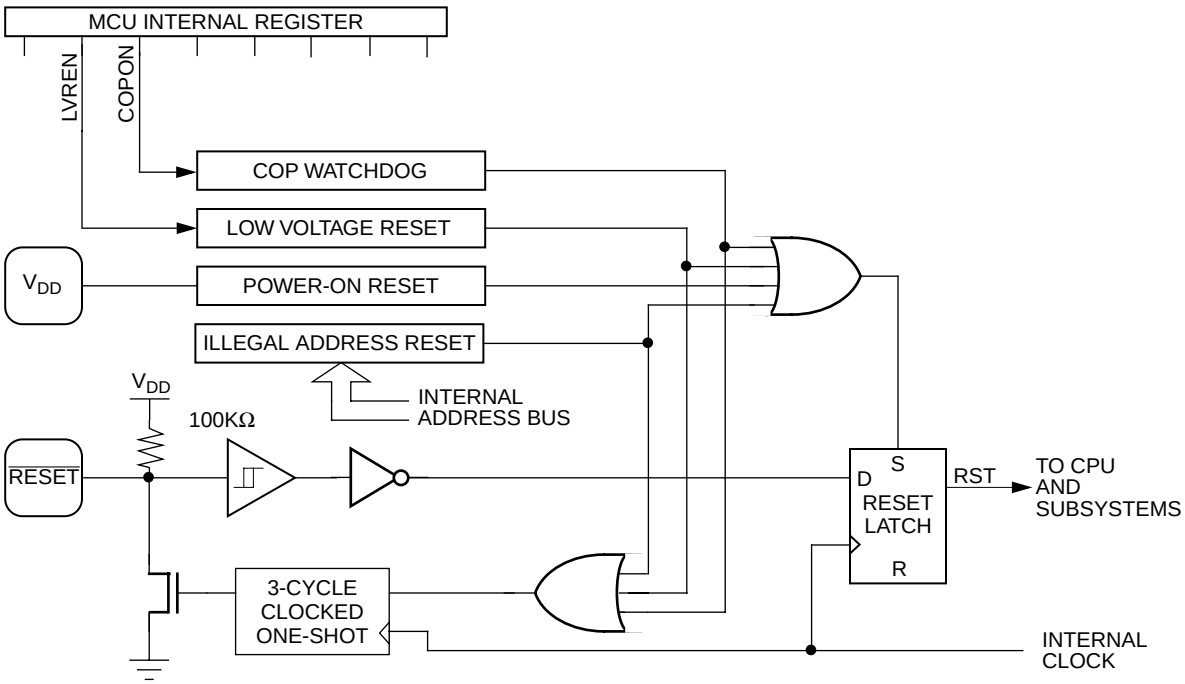


Figure 5-1. Reset Sources

5.1 POWER-ON RESET

A positive transition on the V_{DD} pin generates a power-on reset. The power-on reset is strictly for conditions during powering up 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 the $\overline{\text{RESET}}$ pin is at logic zero at the end of the multiple t_{CYC} time, the MCU remains in the reset condition until the signal on the $\overline{\text{RESET}}$ pin goes to a logic one.

5.2 EXTERNAL RESET

A logic zero applied to the $\overline{\text{RESET}}$ pin for $1.5t_{CYC}$ generates an external reset. This pin is connected to a Schmitt trigger input gate to provide and upper and lower threshold voltage separated by a minimum amount of hysteresis. The external reset occurs whenever the $\overline{\text{RESET}}$ pin is pulled below the lower threshold and remains in reset until the $\overline{\text{RESET}}$ pin rises above the upper threshold. This active low input will generate the internal RST signal that resets the CPU and peripherals.

The $\overline{\text{RESET}}$ pin can also act as an open drain output. It will be pulled to a low state by an internal pulldown device that is activated by three internal reset sources. This RESET pulldown device will only be asserted for 3 - 4 cycles of the internal clock, f_{OP}, or as long as the internal reset source is asserted. When the external $\overline{\text{RESET}}$ pin is asserted, the pulldown device will not be turned on.

NOTE

Do not connect the $\overline{\text{RESET}}$ pin directly to V_{DD}, as this may overload some power supply designs when the internal pulldown on the $\overline{\text{RESET}}$ pin activates.

5.3 INTERNAL RESETS

The four internally generated resets are the initial power-on reset function, the COP Watchdog timer reset, the low voltage reset, and the illegal address detector. Only the COP Watchdog timer reset, low voltage reset and illegal address detector will also assert the pulldown device on the $\overline{\text{RESET}}$ pin for the duration of the reset function or 3 - 4 internal clock cycles, whichever is longer.

5.3.1 Power-On Reset (POR)

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

The POR will generate the RST signal which will reset the CPU. If any other reset function is active at the end of the 4064 cycle delay, the RST signal will remain in the reset condition until the other reset condition(s) end.

POR will not activate the pulldown device on the $\overline{\text{RESET}}$ pin. V_{DD} must drop below V_{POR} in order for the internal POR circuit to detect the next rise of V_{DD}.

5.3.2 Computer Operating Properly (COP) 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 zero to the COPC bit of the COP register at location $1FF0.

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
COPR	R	0	0	0	0	0	0	0	0
$1FF0	W								COPC
reset:		U	U	U	U	U	U	U	0

U = UNAFFECTED BY RESET

Figure 5-2. COP Watchdog Register (COPR)

COPC — COP Clear

COPC is a write-only bit. Periodically writing a logic zero to COPC prevents the COP watchdog from resetting the MCU. Reset clears the COPC bit.

1 = No effect on system.
0 = Reset COP watchdog timer.

The COP Watchdog reset will assert the pulldown device to pull the $\overline{\text{RESET}}$ pin low for three to four clock cycles of the internal bus clock.

After a POR or reset, the COP watchdog is disabled. It is enabled b writing a logic “1” to the COPON bit in the Miscellaneous Control Register (see **Figure 5-2**). Once enabled, the COP watchdog can only be disabled by a POR or reset.

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
MCR	R	TSEN	LVRON	0	SCLK	CSSEL	TCSEL	ESVEN	SMINLEV
$000B	W			COPON					
reset:		0	1	0	0	0	0	0	0

U = UNAFFECTED BY RESET

Figure 5-3. Miscellaneous Control Register (MCR)

COPON — COP ON

COPON is a write-once bit.

1 = Enables COP watchdog system.
0 = No effect on system.

See section on Core Timer for detail on COP watchdog timeout periods.

5.3.3 Low Voltage Reset (LVR)

The LVR activates the RST reset signal to reset the device when the voltage on the V_{DD} pin falls below the LVR trip voltage. The LVR will assert the pulldown device to pull the $\overline{\text{RESET}}$ pin low for three to four clock cycles of the internal bus clock. The Low Voltage Reset circuit is enabled/disabled by the LVRON bit in the Miscellaneous Control Register (see **Figure 5-2**).

LVRON — LVR ON

This is a read/write bit to disable/enable the LVR circuit.

0 = Low Voltage Reset circuit disabled.

1 = Low Voltage Reset circuit enabled. This is the default setting at POR or reset.

5.3.4 Illegal Address Reset

An opcode fetch from an address that is not in the EPROM (locations $0600 – $1DFF and $1FF0 - $1FFF) or the RAM (locations $0030 – $010F) generates an illegal address reset. The illegal address reset will assert the pulldown device to pull the $\overline{\text{RESET}}$ pin low for 3 - 4 cycles of the internal bus clock.

5.4 RESET STATES

The following paragraphs describe how the various resets initialize the MCU.

5.4.1 CPU

A reset has the following effects on the CPU:

- Loads the stack pointer with $FF.
- Sets the I bit in the condition code register, inhibiting interrupts.
- Loads the program counter with the user defined reset vector from locations $1FFE and $1FFF.
- Clears the stop latch, enabling the CPU clock.
- Clears the wait latch, bringing the CPU out of the wait mode.

5.4.2 I/O Registers

A reset has the following effects on I/O registers:

- Clears bits in data direction registers configuring pins as inputs:
  - DDRA7 – DDRA0 in DDRA for port A.
  - DDRB7 – DDRB1 in DDRA for port B.
  - DDRC3–DDRC0 in DDRA for port C.
- Has no effect on port A, B or C data registers.
- Sets the IRQE bit in the interrupt status and control register.

5.4.3 Core Timer

A reset has the following effects on the Core Timer:

- Clears the Core Timer counter register (CTCR).
- Clears the Core Timer interrupt flag and enable bits in the Core Timer status and control register (CTSCR).
- Sets the real-time interrupt rate selection bits (RT0, RT1) such that the device will start with the longest real-time interrupt and COP timeout delays.

5.4.4 COP Watchdog

A reset clears the COP watchdog timeout counter.

5.4.5 16-Bit Programmable Timer

A reset has the following effects on the 16-bit programmable Timer:

- Initializes the timer counter registers (TMRH, TMRL) to a value of $FFFC.
- Initializes the alternate timer counter registers (ACRH, ACRL) to a value of $FFFC.
- Clears all the interrupt enables and the output level bit (OLVL) in the timer control register (TCR).
- Does not affect the input capture edge bit (IEDG) in the TCR.
- Does not affect the interrupt flags in the timer status register (TSR).
- Does not affect the input capture registers (ICRH, ICRL).
- Does not affect the output compare registers (OCRH, OCRL).

5.4.6 SM-Bus Serial Interface

A reset has the following effects on the SM-Bus serial interface:

- Clears all bits in the address register (SMADR) and those unimplemented bit locations are not affected.
- Clears all bits in the frequency divider register (SMFDR) and those unimplemented bit locations are not affected.
- Clears all bits in control register (SMCR) and those unimplemented bit locations are not affected.
- Sets SMCF & RXAK bits and clears other bits and those unimplemented bit locations are not affected.
- Does not affect the contents of the data I/O register (SMDR).

A reset therefore disables the SM-Bus and leaves the shared port A pins as general I/O. Any pending interrupt flag is cleared and the SM-Bus interrupt is disabled. Also the clock rate defaults to the fastest rate.

5.4.7 Analog Subsystem

A reset has the following effects on the analog subsystem:

- Clears all the bits in the multiplex registers (AMUX1, AMUX2) bits except the hold switch bit (HOLD) which is set.
- Clears all the bits in the analog control register (ACR).
- Clears all the bits in the analog status register (ASR).

A reset therefore connects the negative input of comparator to the channel selection bus, which is switched to V_{SS}. The comparator is set up as non-inverting (a higher positive voltage on the positive input results in a positive output) and both are powered down. The current source and discharge device on the CAP pin is also disabled and powered down. Any analog subsystem interrupt flags are cleared and the interrupts are disabled.

SECTION 6
LOW POWER MODES

There are four modes of operation that reduce power consumption:

- Stop mode
- Wait mode
- Data retention mode
- Slow mode

Figure 6-1 shows the sequence of events in stop and wait modes.

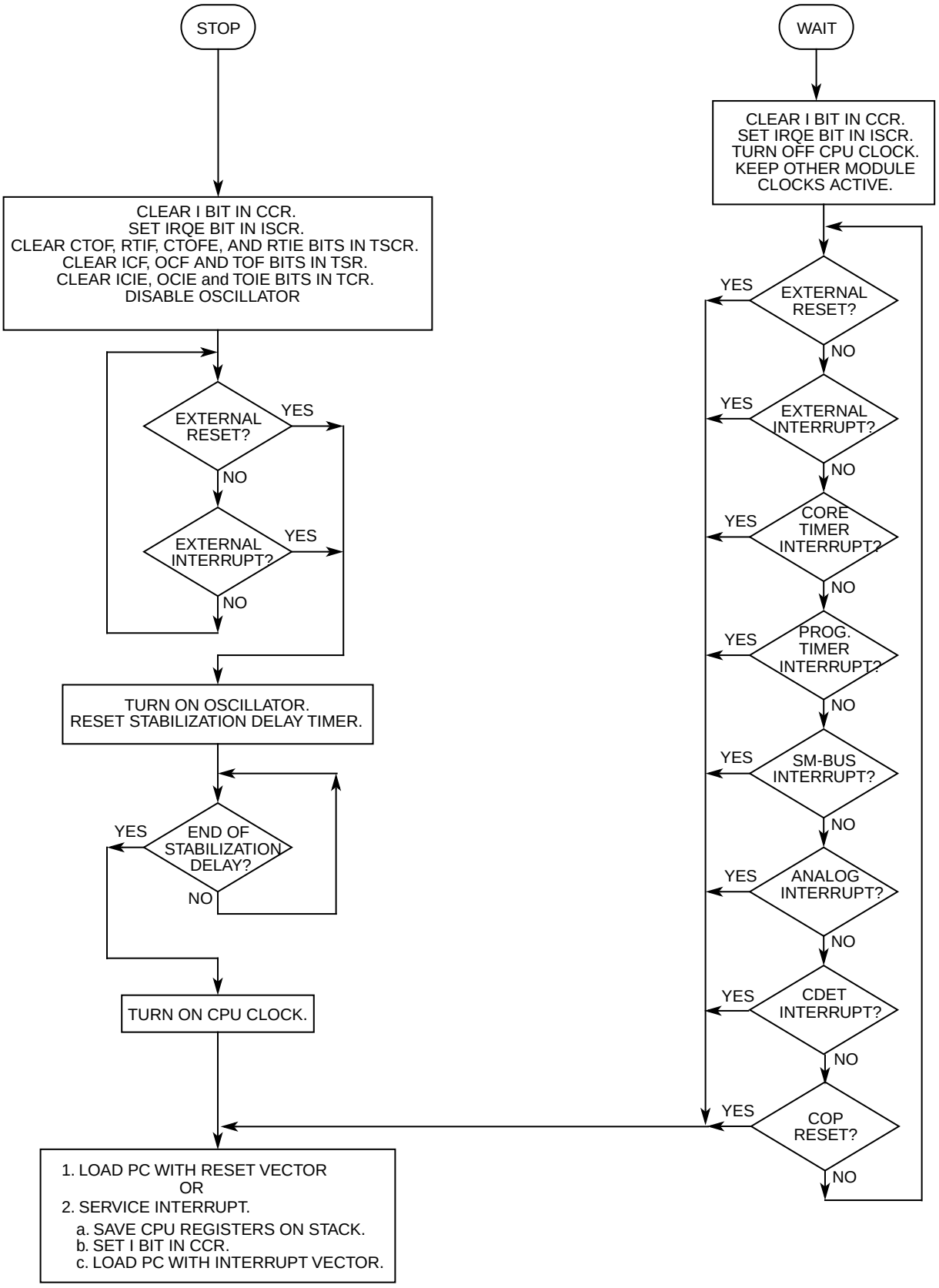


Figure 6-1. STOP and WAIT Flowchart

6.1 STOP MODE

The STOP instruction puts the MCU in a mode with the lowest power consumption and has the following affect on the MCU:

- Turns off the CPU clock and all internal clocks by stopping the internal oscillator. The stopped clocks turn off the COP watchdog, the Core Timer, the programmable Timer, the analog subsystem and the SM-Bus Interface.
- Removes any pending Core Timer interrupts by clearing the Core Timer interrupt flags (CTOF, RTIF) in the Core Timer status and control register (CTSCR).
- Disables any further Core Timer interrupts by clearing the Core Timer interrupt enable bits (CTOFE, RTIE) in the CTSCR.
- Removes any pending programmable Timer interrupts by clearing the timer interrupt flags (ICF, OCF and TOF) in the timer status register (TSR).
- Disables any further programmable Timer interrupts by clearing the timer interrupt enable bits (ICIE, OCIE and TOIE) in the timer control register (TCR).
- Enables external interrupts via the $\overline{IRQ}/V_{PP}$ pin by setting the IRQE bit in the IRQ status and control register (ISCR). Enables interrupts in general by clearing the I bit in the condition code register.

The STOP instruction does not affect any other bits, registers or I/O lines.

The following conditions bring the MCU out of stop mode:

- An external interrupt signal on the $\overline{IRQ}/V_{PP}$ pin — A high to low transition on the $\overline{IRQ}/V_{PP}$ pin loads the program counter with the contents of locations $1FFA and $1FFB.
- External reset — A logic zero on the $\overline{RESET}$ pin resets the MCU and loads the program counter with the contents of locations $1FFE and $1FFF.

When the MCU exits stop mode, processing resumes after a stabilization delay of 4064 oscillator cycles.

If an external interrupt brings the MCU out of stop mode after an active edge occurred on the PC3/TCAP during the stop mode, the ICF flag becomes set. An external interrupt also latches the value of the timer registers into the input capture registers.

If an external reset brings the MCU out of the stop mode after an active edge occurred on the PC3/TCAP pin during the stop mode, the ICF flag does not become set. An external reset has no effect on the input capture registers.

6.2 WAIT MODE

The WAIT instruction puts the MCU in a low power wait mode which consumes more power than the stop mode. The wait mode and has the following affects on the MCU:

- Enables interrupts by clearing the I bit in the condition code register.
- Enables external interrupts by setting the IRQE bit in the IRQ status and control register.
- Stops the CPU clock which drives the address and data buses, but allows the internal oscillator and its clock to continue to run and drive the Core Timer, programmable Timer, analog subsystem and SM-Bus.

The WAIT instruction does not affect any other bits, registers or I/O lines.

The following conditions restart the CPU clock and bring the MCU out of the wait mode:

- An external interrupt signal on the $\overline{IRQ}/V_{PP}$ pin — A high to low transition on the $\overline{IRQ}/V_{PP}$ pin loads the program counter with the contents of locations $1FFA and $1FFB.
- A programmable Timer interrupt — A programmable Timer interrupt driven by an input capture, output compare or timer overflow loads the program counter with the contents of locations $1FF6 and $1FF7.
- An SM-Bus interrupt — An SM-Bus interrupt driven by the completion of transmitted or received 8-bit data loads the program counter with the contents of locations $1FF4 and $1FF5.
- An analog subsystem interrupt — An analog subsystem interrupt driven by a voltage comparison loads the program counter with the contents of locations $1FF2 and $1FF3.
- A Core Timer interrupt — A Core Timer overflow or a real time interrupt loads the program counter with the contents of locations $1FF0 and $1FF1.
- A COP watchdog reset — A timeout of the COP watchdog resets the MCU and loads the program counter with the contents of locations $1FFE and $1FFF. Software can enable real time interrupts so that the MCU can periodically exit the wait mode to reset the COP watchdog.
- External reset — A logic zero on the $\overline{RESET}$ pin resets the MCU and loads the program counter with the contents of locations $1FFE and $1FFF.

6.3 DATA-RETENTION MODE

In the 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 the data retention mode:

1. Drive the $\overline{\text{RESET}}$ pin to a logic zero.
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 the data retention mode:

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

6.4 SLOW MODE

The Slow Mode feature permits a slow down of all the internal operations and thus reduces power consumption. It is particularly useful while going to the WAIT mode. Slow mode is enabled by setting the SCLK bit in the Miscellaneous Control Register ($0B).

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
MCR	R	TSEN	LVRON	0	SCLK	CSSEL	TCSEL	ESVEN	SMINLEV
$000B	W			COPON					
reset:		0	1	0	0	0	0	0	0

U = UNAFFECTED BY RESET

Figure 6-2. Miscellaneous Control Register (MCR)

SCLK — Slow Clock

Setting this bit to one will slow down the internal oscillator. Setting this bit to zero the system will run at the nominal bus speed ($f_{osc}/2$). This bit is cleared during power-on or external reset.

1 = Slow clock selected:
Internal operating frequency, $f_{OP}=f_{BUS}=f_{OSC}/1600$.

0 = Normal clock selected:
Internal operating frequency, $f_{OP}=f_{BUS}=f_{OSC}/2$.

SECTION 7
INPUT/OUTPUT PORTS

In normal operating mode there are 19 bidirectional I/O lines arranged as three I/O ports (Port A, B and C). The individual bits in these ports are programmable as either inputs or outputs under software control by the data direction registers (DDRs). All port I/O pins can sink a current of 5mA when programmed as outputs.

7.1 PARALLEL PORTS

Port A, B and C are bidirectional ports. Each port pin is controlled by the corresponding bits in a data direction register and a data register as shown in **Figure 7-1**.

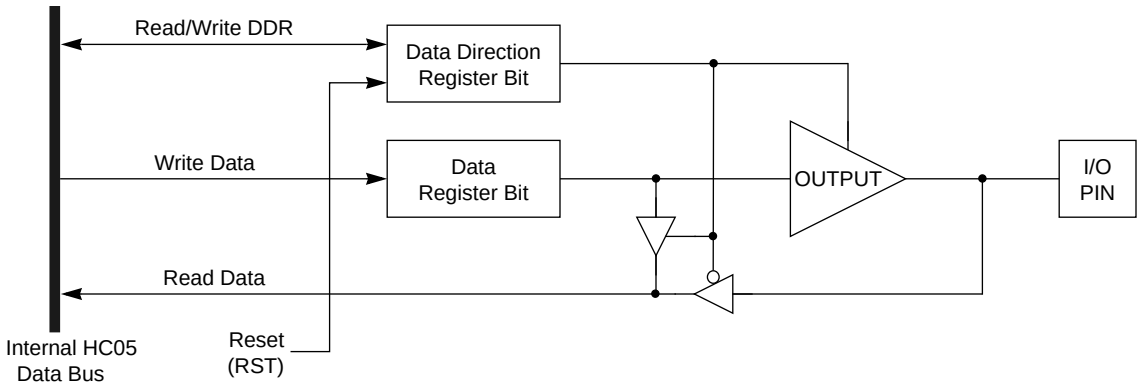


Figure 7-1. Port I/O Circuitry

Table 7-1. I/O Pin Functions

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

7.1.1 Port Data Registers

Each port I/O pin has a corresponding bit in the Port Data Register. When a port I/O pin is programmed as an output the state of the corresponding data register bit determines the state of the output pin. All port I/O pins can sink a current of 5mA when programmed as outputs. When a port pin is programmed as an input, any read of the Port Data Register will return the logic state of the corresponding I/O pin.

7.1.2 Port Data Direction Registers

Each port I/O pin may be programmed as an input by clearing the corresponding bit in the DDR, or programmed as an output by setting the corresponding bit in the DDR.

NOTE

A “glitch” can be generated on an I/O pin when changing it from an input to an output unless the data register is first preconditioned to the desired state before changing the corresponding DDR bit from a “0” to a “1”. Therefore, write data to the I/O Port Data Register before writing a “1” to the corresponding Data Direction Register.

7.2 PORT A

Port A is an 8-bit bidirectional port with pins shared with the PWM outputs and SM-Bus serial I/Os. The Port A Data Register is at address $0000 and the Data Direction Register is at address $0004.

7.3 PORT B

Port B is a 7-bit birdirectional port with pins shared with A/D converter inputs, current detect outputs, and the 16-bit timer TCAP input. The Port B Data Register is at address $0001 and the Data Direction Register is at address $0005.

When selected by mask option, port pins PB2 and PB3 becomes OSC1 and OSC2 respectively.

7.4 PORT C

Port C is a 4-bit bidirectional port. The Port C Data Register is at address $0002 and the Data Direction Register is at address $0006.

SECTION 8
SYSTEM CLOCK

This section describes the system clock options for the MC68HC05SB7.

8.1 CLOCK SOURCES

The internal operating clock of the MC68HC05SB7 is derived from two possible clock sources:

- External oscillator input via the OSC1 and OSC2 pins - this is enabled by a mask option on the MC68HC05SB7.
  (On the MC68HC705SB7, the OSCS bit in the Mask Option Register enables/disables external osc input option.)
- Internal VCO generated.

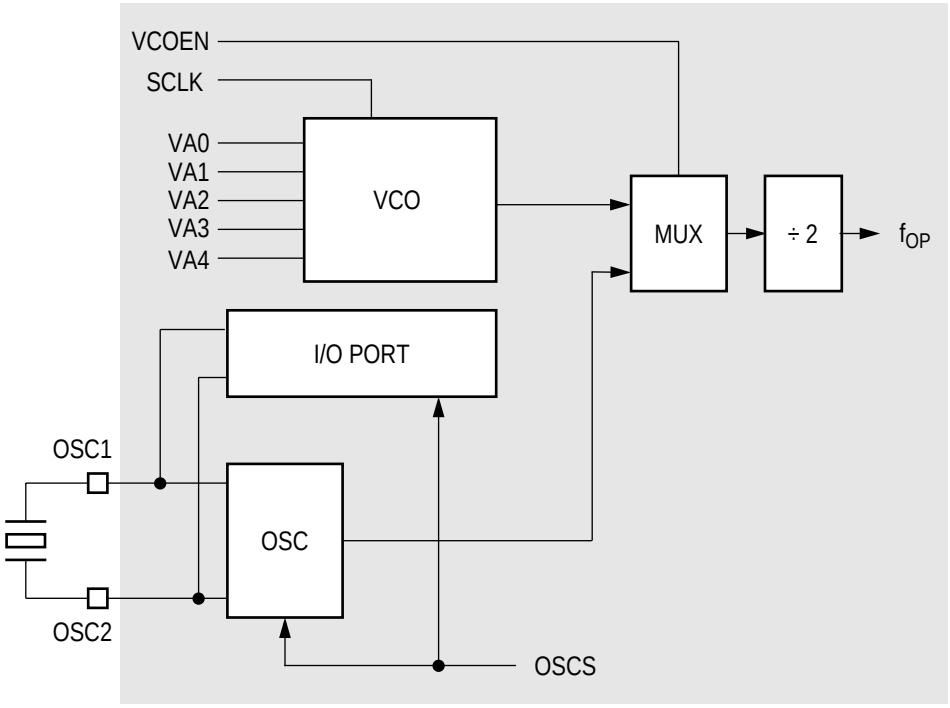


Figure 8-1. MC68HC05SB7 Input Clock Source

The clock source is selected by the VCOEN bit in the IRQ Status and Control Register at $0D. **Table 8-1** shows a summary of the clock source selection.

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
ISCR	R	IRQE	VCOEN	LEVEL	0	IRQF	0	0	0
$000D	W							IRQR	
reset:		1	1	0	0	0	0	0	0

Figure 8-2. IRQ Status and Control Register (ISCR)

VCOEN — VCO ENable

1 = Internal VCO is used as clock source for the MCU.
This is the default setting after a reset.

0 = External OSC is used as clock source for the MCU. If external OSC is disabled (mask option or MOR in MC68HC705SB7), the internal VCO is used as clock source.

After a POR or reset, the internal VCO is selected as the default clock source.

Table 8-1. Clock Source Selection

External OSC Enabled (Mask Option)	Internal VCO Enabled	Clock Source Selected
Disabled (OSCS=0 in MC68HC705SB7)	Don't care (VCOEN=X)	Internal
Enabled (OSCS=1 in MC68HC705SB7)	Disabled (VCOEN=0)	External
Enabled (OSCS=1 in MC68HC705SB7)	Enabled (VCOEN=1)	Internal

NOTE

The user must ensure that the oscillators are stable (4096 clock cycles minimum) if switching between internal and external oscillators.

8.2 VCO CLOCK SPEED

8.2.1 VCO Slow Mode

The internal VCO has two operating modes: Normal mode and Slow mode.

In Normal mode, the VCO frequency ranges from 1.5MHz to 5.8MHz.

In Slow mode, the VCO frequency ranges from 500Hz to 4kHz.

This clock speed option is selected by setting the SCLK bit in the Miscellaneous Register at $0B. The default setting at reset is Normal mode.

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
MCR	R	TSEN	LVRON	0	SCLK	CSSEL	TCSEL	ESVEN	SMINLEV
$000B	W			COPON					
reset:		0	1	0	0	0	0	0	0

U = UNAFFECTED BY RESET

Figure 8-3. Miscellaneous Control Register (MCR)

SCLK — Slow CLocK

1 = Slow clock selected – VCO frequency: 500Hz to 4kHz.
0 = Normal clock selected – VCO frequency: 1.5MHz to 5.8MHz.

NOTE

Due to process variations, operating voltages, and temperature requirements, the quoted VCO frequencies are typical limits, and should be treated as references only. It is the user's responsibility to ensure that the resulting internal operating frequency meets user's requirement by setting the appropriate value in the VCO Adjust Register. See below.

8.2.2 Setting the VCO Speed

The speed of the internal VCO can be adjusted by configuring five bits in the VCO Adjust Register (VAR) as shown in **Figure 8-4**. Setting VAR=11111 will select the VCO minimum frequency, and VAR=00000 will select the maximum frequency. On reset, VAR=10000, which selects the mid-frequency.

For Normal mode, when VAR=10000, VCO frequency is typically 2kHz.

For Slow mode, when VAR=10000, VCO frequency is typically 3.4MHz.

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
VAR	R	0	0	0	VA4	VA3	VA2	VA1	VA0
$000C	W								
reset:		0	0	0	1	0	0	0	0

U = UNAFFECTED BY RESET

Figure 8-4. VCO Adjust Register (VAR)

The VCO minimum and maximum frequencies are available preprogrammed as two 16-bit values in the Personality EPROM (PEPROM). Bit locations $00 to $0F holds the minimum value, and $10 to $1F holds the maximum value.

See section on Personality EPROM for further details.

SECTION 9
CORE TIMER

This section describes the operation of the Core Timer and the Computer Operating Properly (COP) watchdog timer. **Figure 9-1** shows a block diagram of the Core Timer.

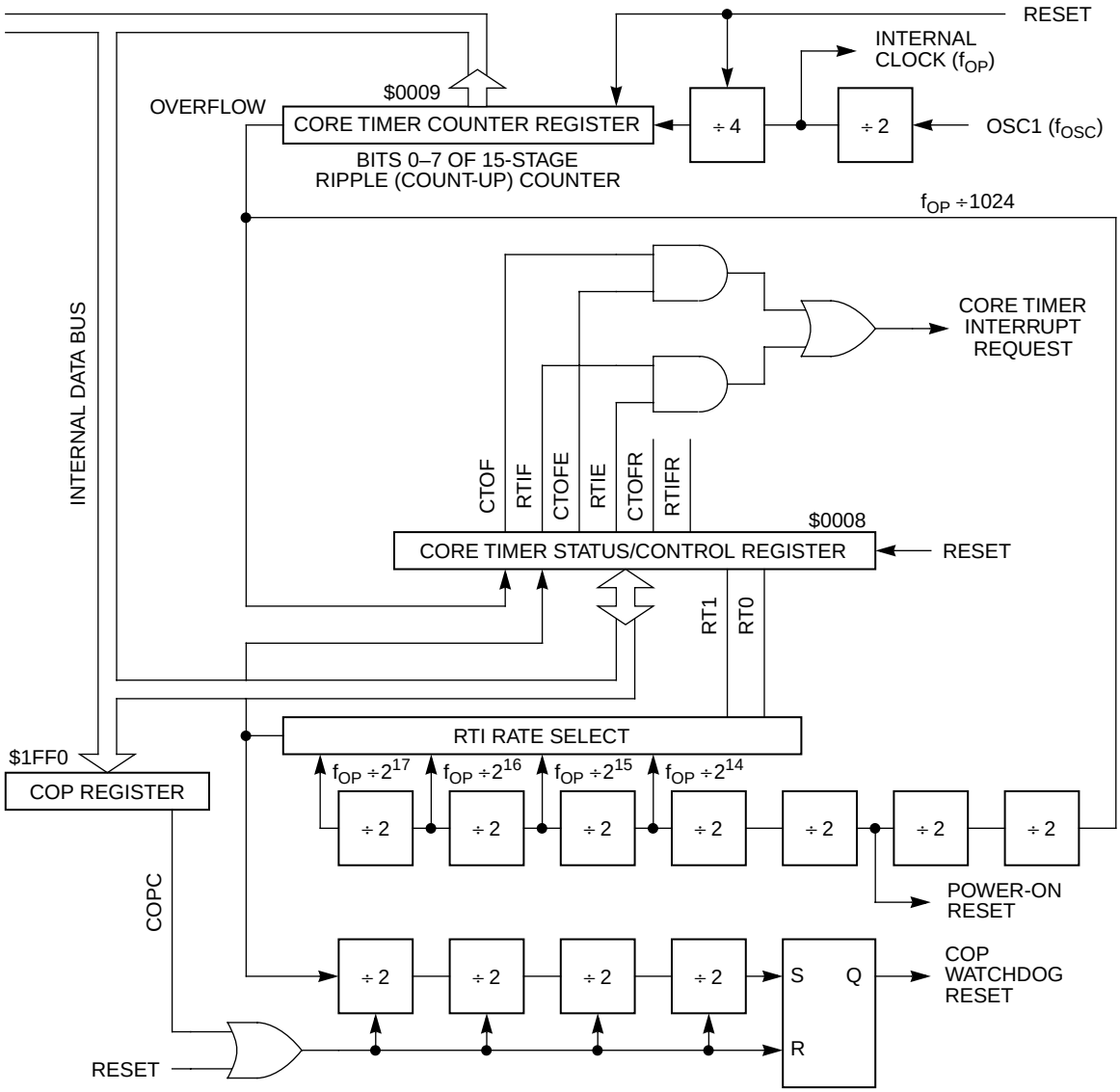


Figure 9-1. Core Timer Block Diagram

9.1 CORE TIMER STATUS AND CONTROL REGISTER

The read/write Core Timer status and control register contains the interrupt flag bits, interrupt enable bits, interrupt flag bit resets, and the rate selects for the real time interrupt as shown in **Figure 9-2**.

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
CTSCR	R	CTOF	RTIF	CTOFE	RTIE	0	0	RT1	RT0
$0008	W					CTOFR	RTIFR		
reset:		0	0	0	0	0	0	1	1

Figure 9-2. Core Timer Status and Control Register (CTSCR)

CTOF — Core Timer Overflow Flag

This read only flag becomes set when the first eight stages of the Core Timer counter roll over from $FF to $00. The CTOF flag bit generates a timer overflow interrupt request if CTOFE is also set. The CTOF flag bit is cleared by writing a logic one to the CTOFR bit. Writing to CTOF has no effect. Reset clears CTOF.

1 = Overflow in Core Timer has occurred.
0 = No overflow of Core Timer since CTOF last cleared.

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. The RTIF enable bit is cleared by writing a logic one to the RTIFR bit. Writing to RTIF has no effect. Reset clears RTIF.

1 = Overflow in real time counter has occurred.
0 = No overflow of real time counter since RTIF last cleared.

CTOFE — Core Timer Overflow Interrupt Enable

This read/write bit enables Core Timer overflow interrupts. Reset clears CTOFE.

1 = Core Timer overflow interrupts enabled.
0 = Core 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.

CTOFR — Core Timer Overflow Flag Reset

Writing a logic one to this write only bit clears the CTOF bit. CTOFR always reads as a logic zero. Reset does not affect CTOFR.

1 = Clear CTOF flag bit.
0 = No effect on CTOF flag bit.

RTIFR — Real-Time Interrupt Flag Reset

Writing a logic one to this write only bit clears the RTIF bit. RTIFR always reads as a logic zero. Reset does not affect RTIFR.

1 = Clear RTIF flag bit.
0 = No effect on RTIF flag bit.

RT1, RT0 — Real-Time Interrupt Select Bits 1 and 0

These read/write bits select one of four real time interrupt rates, as shown in **Table 9-1**. 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, selecting the longest COP timeout period and real-time interrupt period.

NOTE

Changing RT1 and RT0 when a COP timeout is imminent or uncertain may cause a real time interrupt request to be missed or an additional real time interrupt request to be generated. Therefore, the COP timer should be cleared (by writing a just before changing RT1 and RT0.

Table 9-1. Core Timer Interrupt Rates and COP Timeout Selection

Timer Overflow Interrupt (TOF) Period ($f_{OP} \div 2^{10}$)		RT1	RT0	RTI Rate	Real-Time Interrupt (RTI) Period		Minimum COP Timeout Period (7 or 8 RTI Periods)	
f_{OP} = 2.1 MHz	f_{OP} = 1.0 MHz				f_{OP} = 2.1 MHz	f_{OP} = 1.0 MHz	f_{OP} = 2.1 MHz	f_{OP} = 1.0 MHz
488 µs	1024 µs	0	0	$f_{OP} \div 2^{14}$	7.81 ms	16.4 ms	54.7 ms	114 ms
		0	1	$f_{OP} \div 2^{15}$	15.6 ms	32.8 ms	109 ms	229 ms
		1	0	$f_{OP} \div 2^{16}$	31.3 ms	65.5 ms	219 ms	458 ms
		1	1	$f_{OP} \div 2^{17}$	62.5 ms	131 ms	438 ms	916 ms

9.2 CORE TIMER COUNTER REGISTER (CTCR)

A 15-stage ripple counter is the basis of the Core Timer. The value of the first eight stages is readable at any time from the read only timer counter register as shown in **Figure 9-2**.

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
CTCR	R	TMR7	TMR6	TMR5	TMR4	TMR3	TMR2	TMR1	TMR0
$0009	W								
reset:		0	0	0	0	0	0	0	0

Figure 9-3. Core Timer Counter Register (CTCR)

Power on clears the entire counter chain and begins clocking the counter. After the startup delay (16 or 4064 internal clock cycles) the power on reset circuit is released, clearing the counter again and allowing the MCU to come out of reset.

Each count of the timer counter register takes eight oscillator cycles or four cycles of the internal clock. A timer overflow function at the eighth counter stage allows a timer interrupt every 1024 internal clock cycles.

9.3 COP WATCHDOG

Four counter stages at the end of the Core Timer make up the computer operating properly (COP) watchdog. The COP watchdog timeout period is shown in **Table 9-1**.

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 the COPC bit of the COP register at location $1FF0.

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
COPR	R	0	0	0	0	0	0	0	0
$1FF0	W								COPC
reset:		U	U	U	U	U	U	U	0

U = UNAFFECTED BY RESET

Figure 9-4. COP Watchdog Register (COPR)

COPC — COP Clear

COPC is a write-only bit. Periodically writing a logic zero to COPC prevents the COP watchdog from resetting the MCU. Reset clears the COPC bit.

1 = No effect on system.
0 = Reset COP watchdog timer.

The COP Watchdog reset will assert the pulldown device to pull the $\overline{\text{RESET}}$ pin low for three to four clock cycles of the internal bus clock.

After a POR or reset, the COP watchdog is disabled. It is enabled b writing a logic “1” to the COPON bit in the Miscellaneous Control Register (see **Figure 9-5**). Once enabled, the COP watchdog can only be disabled by a POR or reset.

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
MCR	R	TSEN	LVRON	0	SCLK	CSSEL	TCSEL	ESVEN	SMINLEV
$000B	W			COPON					
reset:		0	1	0	0	0	0	0	0

U = UNAFFECTED BY RESET

Figure 9-5. Miscellaneous Control Register (MCR)

COPON — COP ON

COPON is a write-once bit.

1 = Enables COP watchdog system.
0 = No effect on system.

NOTE

If the voltage on the $\overline{IRQ}/V_{PP}$ pin exceeds $2 \times V_{DD}$, the COP watchdog is disabled, and remains disabled until the $\overline{IRQ}/V_{PP}$ voltage falls below $2 \times V_{DD}$.

9.4 CORE TIMER DURING WAIT MODE

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

9.5 CORE TIMER DURING STOP MODE

The Core Timer is cleared when going into STOP mode. When STOP is exited by an external interrupt or an external RESET, the internal oscillator will resume, followed by 4064 cycles internal processor stabilization delay. The timer is then cleared and operation resumes.

SECTION 10
16-BIT TIMER

The MC68HC05SB7 MCU contains a 16-bit programmable Timer with an Input Capture function and an Output Compare function. **Figure 10-1** shows a block diagram of the 16-bit programmable timer.

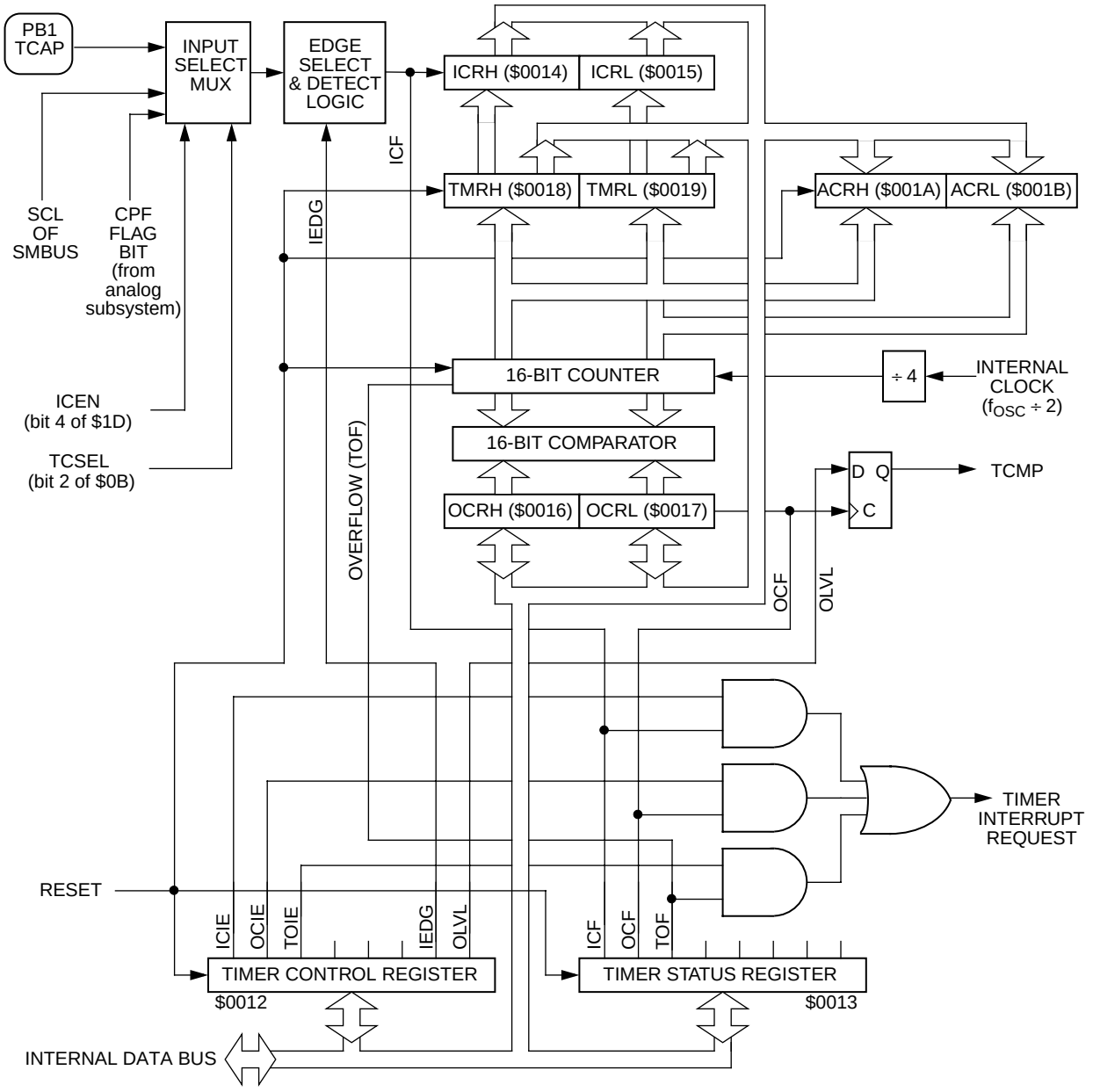


Figure 10-1. Programmable Timer Block Diagram

The basis of the capture/compare Timer is a 16-bit free-running counter which increases in count with each internal bus clock cycle. The counter is the timing reference for the input capture and output compare functions. The input capture and output compare functions provide a means to latch the times at which external events occur, to measure input waveforms, and to generate output waveforms and timing delays. Software can read the value in the 16-bit free-running counter at any time without affect the counter sequence.

Because of the 16-bit timer architecture, the I/O registers for the input capture and output compare functions are pairs of 8-bit registers. Each register pair contains the high and low byte of that function. Generally, accessing the low byte of a specific timer function allows full control of that function; however, an access of the high byte inhibits that specific timer function until the low byte is also accessed.

Because the counter is 16 bits long and preceded by a fixed divide-by-four prescaler, the counter rolls over every 262,144 internal clock cycles. Timer resolution with a 4 MHz crystal oscillator is 2 microsecond/count.

The interrupt capability, the input capture edge, and the output compare state are controlled by the timer control register (TCR) located at $0012 and the status of the interrupt flags can be read from the timer status register (TSR) located at $0013.

10.1 TIMER REGISTERS (TMRH, TMRL)

The functional block diagram of the 16-bit free-running timer counter and timer registers is shown in **Figure 10-2**. The timer registers include a transparent buffer latch on the LSB of the 16-bit timer counter.

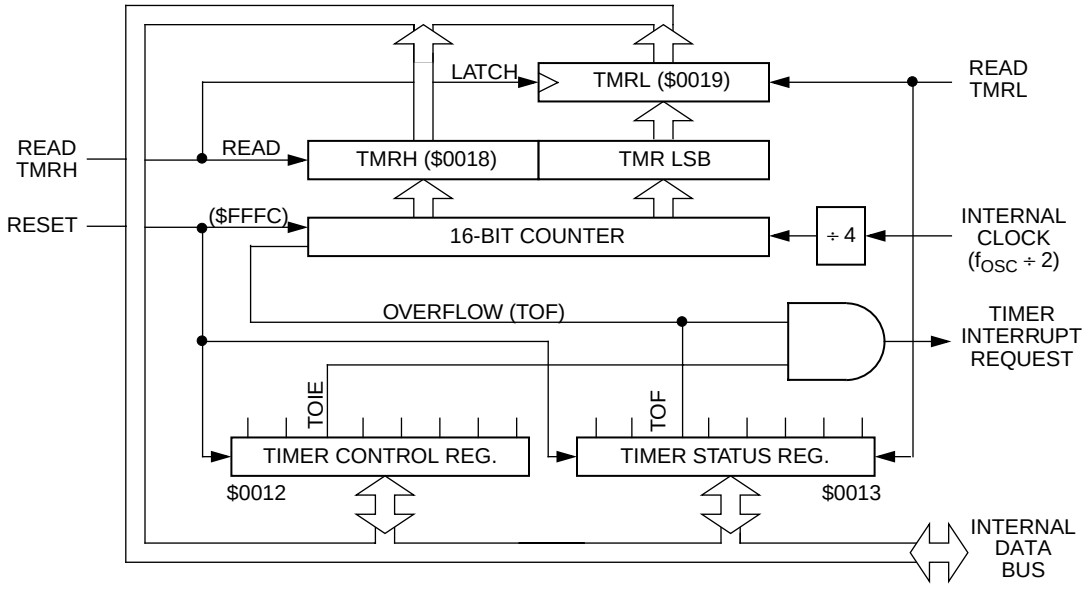


Figure 10-2. Programmable Timer Block Diagram

The timer registers (TMRH, TMRL) shown in **Figure 10-3** are read-only locations which contain the current high and low bytes of the 16-bit free-running counter. Writing to the timer registers has no effect. Reset of the device presets the timer counter to $FFFC.

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
TMRH	R	TMRH7	TMRH6	TMRH5	TMRH4	TMRH3	TMRH2	TMRH1	TMRH0
$0018	W								
reset:		1	1	1	1	1	1	1	1
TMRL	R	TMRL7	TMRL6	TMRL5	TMRL4	TMRL3	TMRL2	TMRL1	TMRL0
$0019	W								
reset:		1	1	1	1	1	1	0	0

Figure 10-3. Programmable Timer Registers (TMRH, TMRL)

The TMRL latch is a transparent read of the LSB until the a read of the TMRH takes place. A read of the TMRH latches the LSB into the TMRL location until the TMRL is again read. The latched value remains fixed even if multiple reads of the TMRH take place before the next read of the TMRL. Therefore, when reading the MSB of the timer at TMRH the LSB of the timer at TMRL must also be read to complete the read sequence.

During power-on-reset (POR), the counter is initialized to $FFFC and begins counting after the oscillator start-up delay. Because the counter is sixteen bits and preceded by a fixed divide-by-four prescaler, the value in the counter repeats every 262, 144 internal bus clock cycles (524, 288 oscillator cycles).

When the free-running counter rolls over from $FFFF to $0000, the timer overflow flag bit (TOF) is set in the TSR. When the TOF is set, it can generate an interrupt if the timer overflow interrupt enable bit (TOIE) is also set in the TCR. The TOF flag bit can only be reset by reading the TMRL after reading the TSR.

Other than clearing any possible TOF flags, reading the TMRH and TMRL in any order or any number of times does not have any effect on the 16-bit free-running counter.

NOTE

To prevent interrupts from occurring between readings of the TMRH and TMRL, set the I bit in the condition code register (CCR) before reading TMRH and clear the I bit after reading TMRL.

10.2 ALTERNATE COUNTER REGISTERS (ACRH, ACRL)

The functional block diagram of the 16-bit free-running timer counter and alternate counter registers is shown in **Figure 10-4**. The alternate counter registers behave the same as the timer registers, except that any reads of the alternate counter will not have any effect on the TOF flag bit and Timer interrupts. The alternate counter registers include a transparent buffer latch on the LSB of the 16-bit timer counter.

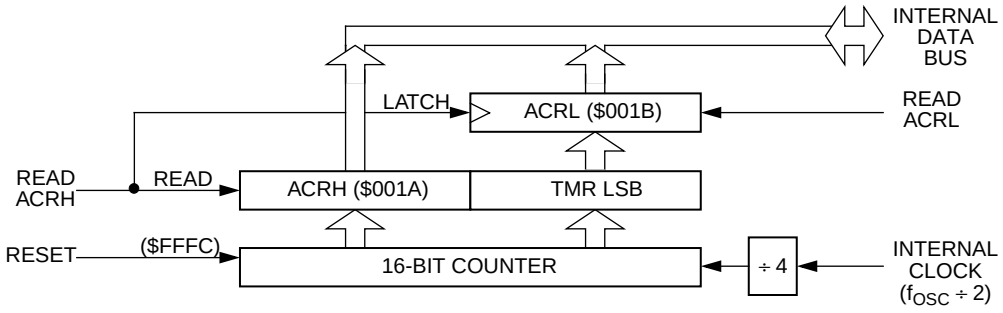


Figure 10-4. Alternate Counter Block Diagram

The alternate counter registers (ACRH, ACRL) shown in **Figure 10-5** are read-only locations which contain the current high and low bytes of the 16-bit free-running counter. Writing to the alternate counter registers has no effect. Reset of the device presets the timer counter to $FFFC.

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

ACRL	R	ACRL7	ACRL6	ACRL5	ACRL4	ACRL3	ACRL2	ACRL1	ACRL0
$001B	W								
reset:		1	1	1	1	1	1	0	0

Figure 10-5. Alternate Counter Registers (ACRH, ACRL)

The ACRL latch is a transparent read of the LSB until the a read of the ACRH takes place. A read of the ACRH latches the LSB into the ACRL location until the ACRL is again read. The latched value remains fixed even if multiple reads of the ACRH take place before the next read of the ACRL. Therefore, when reading the MSB of the timer at ACRH the LSB of the timer at ACRL must also be read to complete the read sequence.

During power-on-reset (POR), the counter is initialized to $FFFC and begins counting after the oscillator start-up delay. Because the counter is sixteen bits and preceded by a fixed divide-by-four prescaler, the value in the counter repeats every 262,144 internal bus clock cycles (524,288 oscillator cycles).

Reading the ACRH and ACRL in any order or any number of times does not have any effect on the 16-bit free-running counter or the TOF flag bit.

NOTE

To prevent interrupts from occurring between readings of the ACRH and ACRL, set the I bit in the condition code register (CCR) before reading ACRH and clear the I bit after reading ACRL.

10.3 INPUT CAPTURE REGISTERS

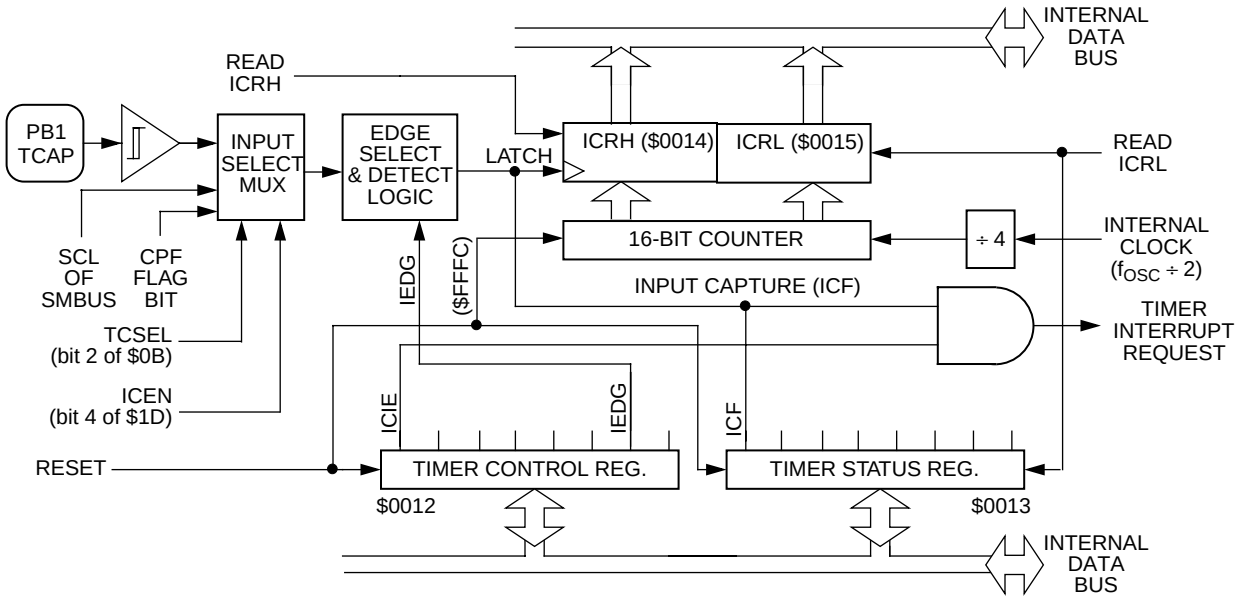


Figure 10-6. Timer Input Capture Block Diagram

The input capture function is a means to record the time at which an event occurs. The source of the event can be selected from the following:

- External input via the PB1 pin
- CPF flag from the voltage comparator in the analog subsystem
- SCL signal from the SMBUS

The input capture source is selected by the TCSEL and ICEN bits.

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
MCR	R	TSEN	LVRON	0	SCLK	CSSEL	TCSEL	ESVEN	SMINLEV
$000B	W			COPON					
reset:		0	1	0	0	0	0	0	0

Figure 10-7. Miscellaneous Control Register (MCR)

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
ACR	R	CHG	ATD2	ATD1	ICEN	CPIE	CPEN		ISEN
$001D	W								
reset:		0	0	0	0	0	0	0	0

Figure 10-8. Analog Control Register (ACR)

Table 10-1. 16-bit Timer Input Capture Source

TCSEL	ICEN	Selected TCAP Source
0	0	External TCAP via PB1
0	1	CPF from Analog Subsystem
1	0	SCL from SMBus
1	1	SCL from SMBus

When the input capture circuitry detects an active edge on the selected source, it latches the contents of the free-running timer counter registers into the input capture registers as shown in **Figure 10-6**.

Latching values into the input capture registers at successive edges of the same polarity measures the period of the selected input signal. Latching the counter values at successive edges of opposite polarity measures the pulse width of the signal.

The input capture registers are made up of two 8-bit read-only registers (ICRH, ICRL) as shown in **Figure 10-9**. The input capture edge detector contains a Schmitt trigger to improve noise immunity. The edge that triggers the counter transfer is defined by the input edge bit (IEDG) in the TCR. Reset does not affect the contents of the input capture registers.

The result obtained by an input capture will be one count higher than the value of the free-running timer counter preceding the external transition. This delay is required for internal synchronization. Resolution is affected by the prescaler, allowing the free-running timer counter to increment once every four internal clock cycles (eight oscillator clock cycles).

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
ICRH	R	ICRH7	ICRH6	ICRH5	ICRH4	ICRH3	ICRH2	ICRH1	ICRH0
$0014	W								
reset:		U	U	U	U	U	U	U	U

ICRL	R	ICRL7	ICRL6	ICRL5	ICRL4	ICRL3	ICRL2	ICRL1	ICRL0
$0015	W								
reset:		U	U	U	U	U	U	U	U

U = UNAFFECTED BY RESET

Figure 10-9. Input Capture Registers (ICRH, ICRL)

Reading the ICRH inhibits further captures until the ICRL is also read. Reading the ICRL after reading the timer status register (TSR) clears the ICF flag bit. does not inhibit transfer of the free-running counter. There is no conflict between reading the ICRL and transfers from the free-running timer counters. The input capture registers always contain the free-running timer counter value which corresponds to the most recent input capture.

NOTE

To prevent interrupts from occurring between readings of the ICRH and ICRL, set the I bit in the condition code register (CCR) before reading ICRH and clear the I bit after reading ICRL.

10.4 OUTPUT COMPARE REGISTERS

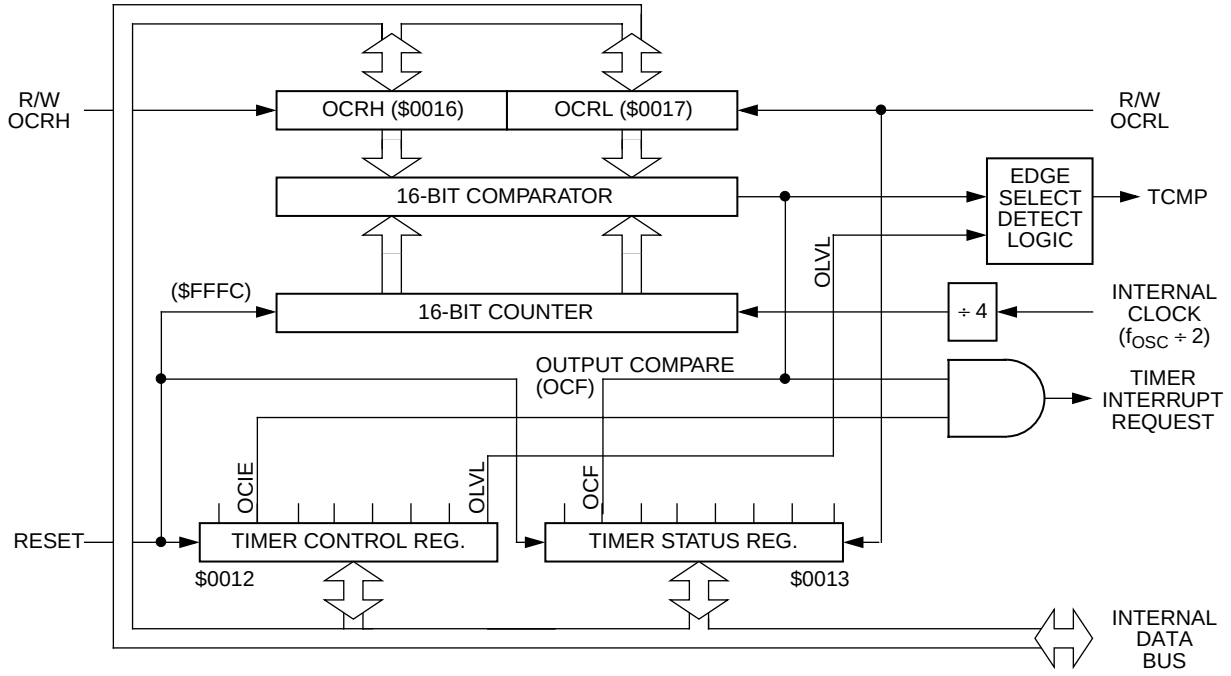


Figure 10-10. Timer Output Compare Block Diagram

The Output Compare function is a means of generating an output signal when the 16-bit timer counter reaches a selected value as shown in **Figure 10-10**. Software writes the selected value into the output compare registers. On every fourth internal clock cycle (every eight oscillator clock cycle) the output compare circuitry compares the value of the free-running timer counter to the value written in the output compare registers. When a match occurs, the timer transfers the output level (OLVL) from the timer control register (TCR) to the TCMP.

Software can use the output compare register to measure time periods, to generate timing delays, or to generate a pulse of specific duration or a pulse train of specific frequency and duty cycle on the TCMP.

The planned action on the TCMP depends on the value stored in the OLVL bit in the TCR, and it occurs when the value of the 16-bit free-running timer counter matches the value in the output compare registers shown in **Figure 10-3**. These registers are read/write bits and are unaffected by reset.

Writing to the OCRH before writing to the OCRL inhibits timer compares until the OCRL is written. Reading or writing to the OCRL after reading the TSR will clear the output compare flag bit (OCF). The output compare OLVL state will be clocked to its output latch regardless of the state of the OCF.

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

U = UNAFFECTED BY RESET

Figure 10-11. Output Compare Registers (OCRH, OCRL)

To prevent OCF from being set between the time it is read and the time the output compare registers are updated, use the following procedure:

1. Disable interrupts by setting the I bit in the condition code register.
2. Write to the OCRH. Compares are now inhibited until OCRL is written.
3. Read the TSR to arm the OCF for clearing.
4. Enable the output compare registers by writing to the OCRL. This also clears the OCF flag bit in the TSR.
5. Enable interrupts by clearing the I bit in the condition code register.

A software example of this procedure is shown below.

```
9B              SEI             DISABLE INTERRUPTS
...             ...             .....
...             ...             .....
B7      16      STA     OCRH    INHIBIT OUTPUT COMPARE
B6      13      LDA     TSR     ARM OCF FLAG FOR CLEARING
BF      17      STX     OCRL    READY FOR NEXT COMPARE, OCF CLEARED
...             ...             .....
...             ...             .....
9A              CLI             ENABLE INTERRUPTS
```

10.5 TIMER CONTROL REGISTER (TCR)

The timer control register is shown in **Figure 10-12** performs the following functions:

- Enables input capture interrupts.
- Enables output compare interrupts.
- Enables timer overflow interrupts.
- Control the active edge polarity of the TCAP signal.
- Controls the active level of the TCMP output.

Reset clears all the bits in the TCR with the exception of the IEDG bit which is unaffected.

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
TCR	R	ICIE	OCIE	TOIE	0	0	0	IEDG	OLVL
$0012	W								
reset:		0	0	0	0	0	0	Unaffected	0

Figure 10-12. Timer Control Register (TCR)

ICIE - INPUT CAPTURE INTERRUPT ENABLE

This read/write bit enables interrupts caused by an active signal on the PB1/TCAP pin or from CPF flag bit of the analog subsystem voltage comparator. Reset clears the ICIE bit.

1 = Input capture interrupts enabled.
0 = Input capture interrupts disabled.

OCIE - OUTPUT COMPARE INTERRUPT ENABLE

This read/write bit enables interrupts caused by an active signal on the TCMP pin. Reset clears the OCIE bit.

1 = Output compare interrupts enabled.
0 = Output compare interrupts disabled.

TOIE - TIMER OVERFLOW INTERRUPT ENABLE

This read/write bit enables interrupts caused by a timer overflow. Reset clears the TOIE bit.

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

IEDG - INPUT CAPTURE EDGE SELECT

The state of this read/write bit determines whether a positive or negative transition on the TCAP pin or the CPF flag bit of voltage comparator in the analog subsystem triggers a transfer of the contents of the timer register to the input capture register. Reset has no effect on the IEDG bit.

1 = Positive edge (low to high transition) triggers input capture.
0 = Negative edge (high to low transition) triggers input capture.

OLVL - OUTPUT COMPARE OUTPUT LEVEL SELECT

The state of this read/write bit determines whether a logic one or a logic zero appears on the TCMP when a successful output compare occurs. Reset clears the OLVL bit.

1 = TCMP goes high on output compare.
0 = TCMP goes low on output compare.

10.6 TIMER STATUS REGISTER (TSR)

The timer status register (TSR) shown in **Figure 10-13** contains flags for the following events:

- An active signal on the PB1/TCAP pin or the CPF flag bit of voltage comparator in the analog subsystem, transferring the contents of the timer registers to the input capture registers.
- A match between the 16-bit counter and the output compare registers, transferring the OLVL bit to the TCMP.
- An overflow of the timer registers from $FFFF to $0000.

Writing to any of the bits in the TSR has no effect. Reset does not change the state of any of the flag bits in the TSR.

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
TSR	R	ICF	OCF	TOF	0	0	0	0	0
$0013	W								
reset:		U	U	U	0	0	0	0	0

U = UNAFFECTED BY RESET

Figure 10-13. Timer Status Registers (TSR)

ICF - INPUT CAPTURE FLAG

The ICF bit is automatically set when an edge of the selected polarity occurs on the PB1/TCAP pin. Clear the ICF bit by reading the timer status register with the ICF set, and then reading the low byte (ICRL, $0015) of the input capture registers. Reset has no effect on ICF.

OCF - OUTPUT COMPARE FLAG

The OCF bit is automatically set when the value of the timer registers matches the contents of the output compare registers. Clear the OCF bit by reading the timer status register with the OCF set, and then accessing the low byte (OCRL, $0017) of the output compare registers. Reset has no effect on OCF.

TOF - TIMER OVERFLOW FLAG

The TOF bit is automatically set when the 16-bit timer counter rolls over from $FFFF to $0000. Clear the TOF bit by reading the timer status register with the TOF set, and then accessing the low byte (TMRL, $0019) of the timer registers. Reset has no effect on TOF.

10.7 TIMER OPERATION DURING WAIT MODE

During WAIT mode the 16-bit timer continues to operate normally and may generate an interrupt to trigger the MCU out of the WAIT mode.

10.8 TIMER OPERATION DURING STOP MODE

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

If a valid input capture edge occurs at the PB1/TCAP pin during the STOP mode the input capture detect circuitry will be armed. This action does not set any flags or “wake up” the MCU, but when the MCU does “wake up” there will be an active input capture flag (and data) from the first valid edge. If the STOP mode is exited by an external reset, no input capture flag or data will be present even if a valid input capture edge was detected during the STOP mode.

SECTION 11
PULSE WIDTH MODULATOR

The PWM subsystem contains four 10-bit PWM channels, which can be used as independent D/A converters. **Figure 11-1** shows the block diagram for the Pulse Width Modulator, with channel 0 in details.

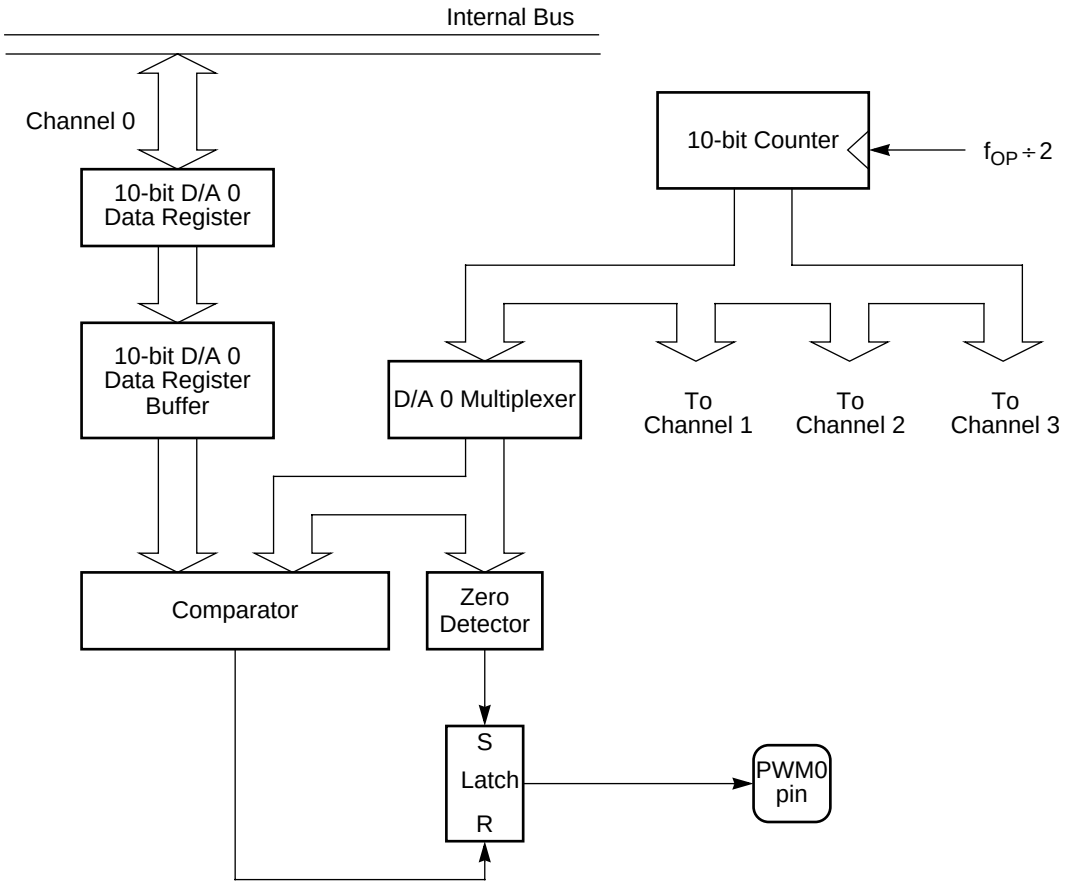


Figure 11-1. PWM Block Diagram

The PWM cycle time is 2048 times the MCU internal processor clock (f_{OP} or f_{BUS}). Duty cycle of the PWM outputs can be programmed by the corresponding D/A Data Registers (DAC0-DAC3).

11.1 D/A DATA REGISTERS (DAC0-DAC3)

Each PWM channel is programmed with a 10-bit data, in two 8-bit registers.

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
DAC0	R								
$0025	W	D9	D8	D7	D6	D5	D4	D3	D2
reset:		0	0	0	0	0	0	0	0

Figure 11-2. D/A Data Register 0 (DAC0) (MSB)

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
DAC0	R								
$0026	W							D1	D0
reset:		0	0	0	0	0	0	0	0

Figure 11-3. D/A Data Register 0 (DAC0) (LSB)

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
DAC1	R								
$0027	W	D9	D8	D7	D6	D5	D4	D3	D2
reset:		0	0	0	0	0	0	0	0

Figure 11-4. D/A Data Register 1 (DAC1) (MSB)

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
DAC1	R								
$0028	W							D1	D0
reset:		0	0	0	0	0	0	0	0

Figure 11-5. D/A Data Register 1 (DAC1) (LSB)

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
DAC2	R								
$0029	W	D9	D8	D7	D6	D5	D4	D3	D2
reset:		0	0	0	0	0	0	0	0

Figure 11-6. D/A Data Register 2 (DAC2) (MSB)

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
DAC2	R								
$002A	W							D1	D0
reset:		0	0	0	0	0	0	0	0

Figure 11-7. D/A Data Register 2 (DAC2) (LSB)

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
DAC3	R								
$002B	W	D9	D8	D7	D6	D5	D4	D3	D2
reset:		0	0	0	0	0	0	0	0

Figure 11-8. D/A Data Register 3 (DAC3) (MSB)

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
DAC3	R								
$002C	W							D1	D0
reset:		0	0	0	0	0	0	0	0

Figure 11-9. D/A Data Register 3 (DAC3) (LSB)

A value of $0000 loaded into these registers results in a continuously low output on the corresponding PWM output pin. A value of $0200 results in a 50% duty cycle output, and so on. The maximum value, $03FF corresponds to an output which is at “1” for 1023/1024 of the cycle.

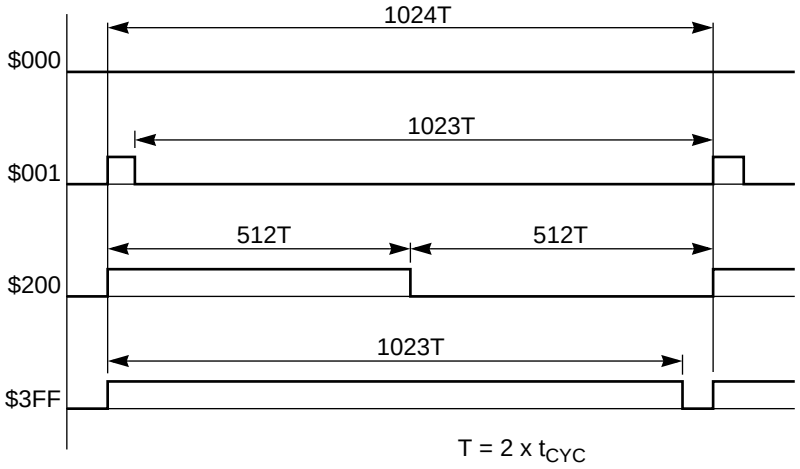


Figure 11-10. PWM Output Waveform Examples

A new value written to the a D/A register pair will not be effective until the end of the current PWM period. This provides a monotonic change of the DC component of the output without overshoots or vicious starts (a vicious start is an output which gives totally erroneous PWM during the initial period following an update of the PWM registers). This feature is achieved by double buffering of the PWM D/A registers.

11.2 MUX CHANNEL ENABLE REGISTER (MCER)

Since the PWM output pins PWM0-PWM3 are multiplexed with the standard I/O port pins PA0-PA3 respectively, the MCER is provided to switch between the PWM and standard I/O function for each pin.

Each PWM channel is enabled by setting the corresponding DAn-E bit in the MCER, shown in **Figure 11-11**. With a PWM output enabled, the corresponding port I/O is tri-stated automatically. Reset clears the four DAn-E bits.

The outputs from four channels PWM system can be inhibited by setting the PWM_I bit in MCER. This bit can be used as a global pull logic “0” for all the enabled DA’s line before enter STOP mode. The PWM_I bit is also used to disable the counter while the PWM is not in use for power saving. Reset clears this bit.

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
MCER	R	PWM_I	0	0	0	DA3-E	DA2-E	DA1-E	DA0-E
$002D	W								
reset:		0	0	0	0	0	0	0	0

Figure 11-11. MUX Channel Enable Register (MCER)

DAn-E — D/A Channel n Enable

1 = PWM output selected for PWMn/PAn pin.
0 = Standard I/O selected for PWMn/PAn pin.

PWM_I — PWM Inhibit

1 = Inhibit all four PWM channels; PWM 10-bit counter also stopped.
0 = PWM channels not inhibited.

11.3 PWM DURING WAIT MODE

In WAIT mode, the oscillator is running even though the MCU clock is not present, the PWM outputs are not affected. To reduce power consumption in WAIT mode, it is recommended to disable the PWM.

11.4 PWM DURING STOP MODE

In STOP mode, the oscillator is stopped asynchronously with PWM operation. As a consequence, the PWM output will remain at the state at the moment when the oscillator is stopped. The PWM pin’s output depended on the state of PWM_I bit. If this bit is clear, it might be at its high or low state at that moment, and it remains at that state until STOP mode is exited. If the PWM_I bit is set, it will be inhibited the state of PWM output in the process and pin output will be in logic low state. After STOP mode is exited, the PWM output resumes its unfinished portion of the stopped cycle if PWM_I bit is clear by software.

SECTION 12
SM-BUS

12.1 SM-BUS INTRODUCTION

The System Management Bus (SM-Bus) is a two wire, bidirectional serial bus which provides a simple, efficient way for data exchange between devices.

This bus is suitable for applications which need frequent communications over a short distance between a number of devices. It also provides a flexibility that allows additional devices to be connected to the bus. The maximum data rate is 100 kbit/s, and the maximum communication distance and number of devices that can be connected is limited by a maximum bus capacitance of 400 pF.

The SM-Bus is a true multi-master bus, including collision detection and arbitration to prevent data corruption if two or more masters intend to control the bus simultaneously. This feature provides the capability for complex applications with multi-processor control. It may also be used for rapid testing and alignment of end products via external connections to an assembly-line computer.

Figure 12-1 shows a block diagram of the SM-Bus interface.

12.2 SM-BUS INTERFACE FEATURES

- Fully compatible to SM-Bus standard
- Multi-master operation
- Software programmable for one of 32 different serial clock frequencies
- Software selectable acknowledge bit
- Interrupt driven byte by byte data transfer
- Arbitration lost driven interrupt with automatic mode switching from master to slave
- Calling address identification interrupt
- Generate/detect the START or STOP signal
- Repeated START signal generation
- Generate/recognize the acknowledge bit
- Bus busy detection

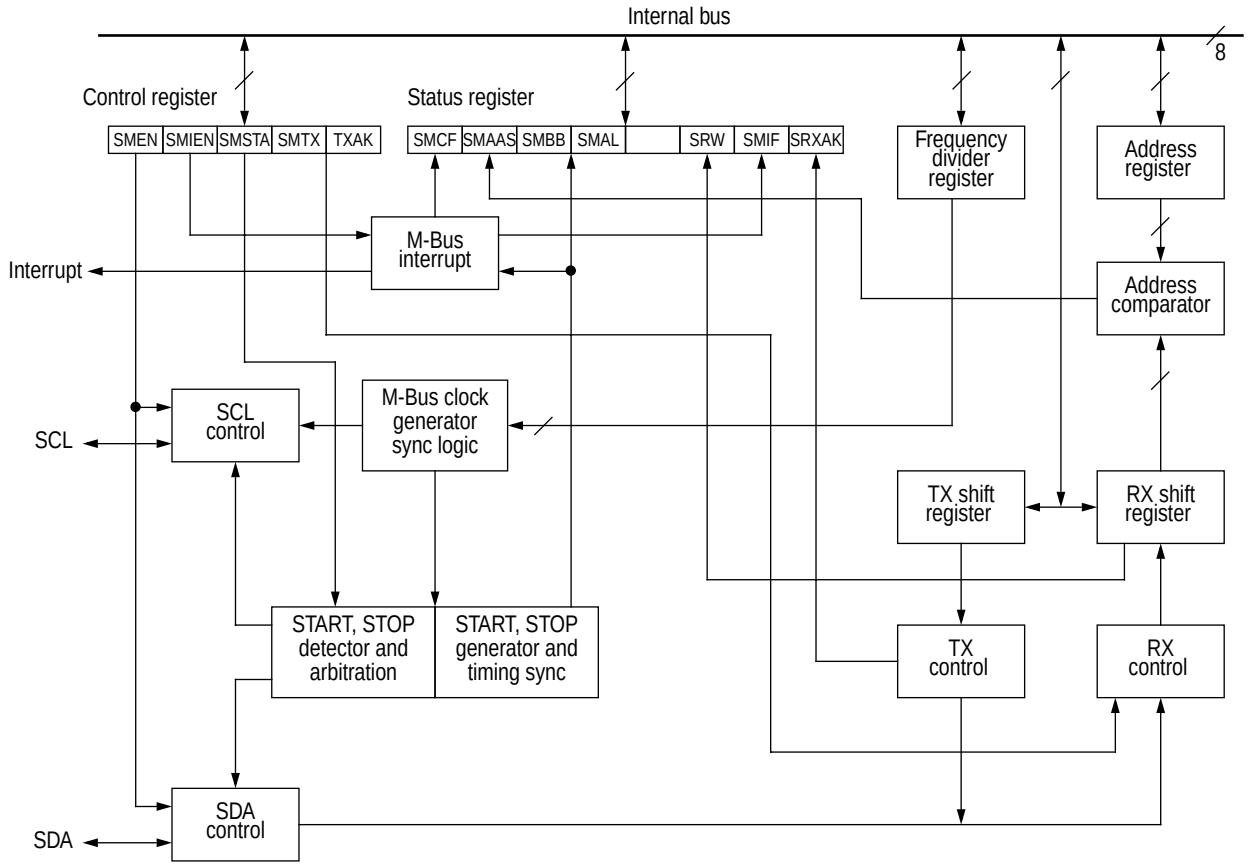


Figure 12-1. SM-Bus Interface Block Diagram

12.3 SM-BUS SYSTEM CONFIGURATION

The SM-Bus system uses a serial data line (SDA) and a serial clock line (SCL) for data transfer. All devices connected to it must have open drain or open collector outputs and the logical “AND” function is performed on both lines by two pull up resistors.

12.4 SM-BUS PROTOCOL

Normally a standard communication is composed of four parts, START signal, Slave Address transmission, Data transfer, and STOP signal. These are described briefly in the following sections and illustrated in **Figure 12-2**.

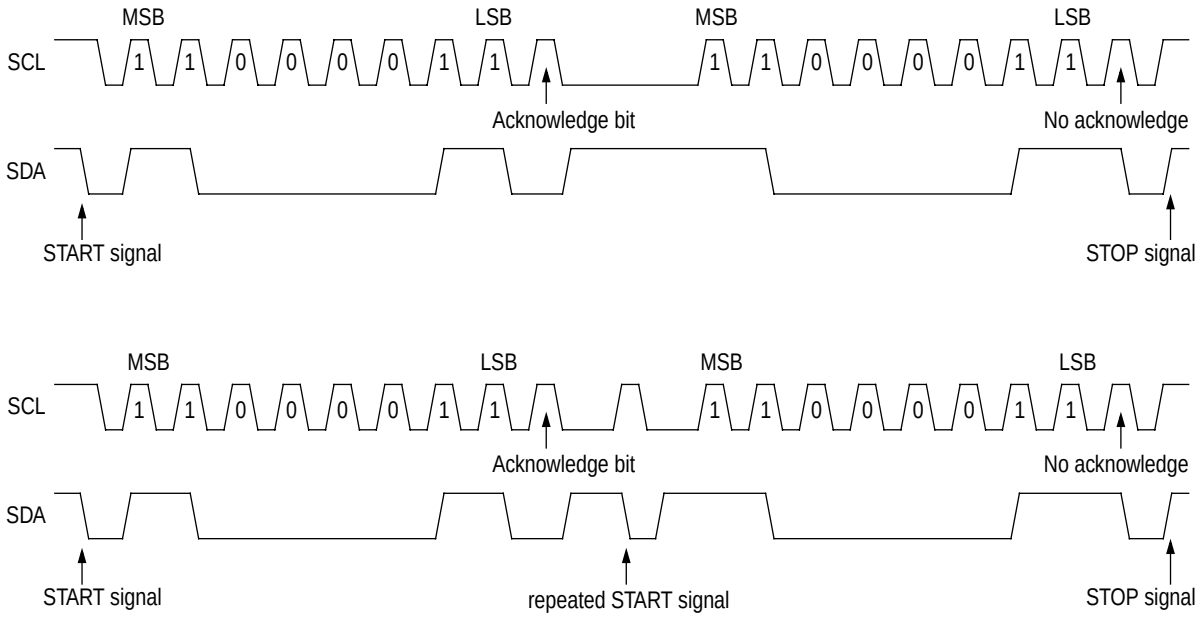


Figure 12-2. SM-Bus Transmission Signal Diagram

12.4.1 START Signal

When the bus is free, (i.e. no master device is engaging the bus and both SCL and SDA lines are at logical high) a master may initiate communication by sending a START signal. As shown in **Figure 12-2**, a START signal is defined as a high to low transition of SDA while SCL is high. This signal denotes the beginning of new data transfer (each data transfer may contain several bytes of data) and wakes up all slaves.

12.4.2 Slave Address Transmission

The first byte of data transfer immediately after the START signal is the slave address transmitted by the Master. This is a seven bit long calling address followed by a R/W-bit. The R/W-bit tells the slave the desired direction of data transfer.

Only the slave with a matched address will respond by sending back an acknowledge bit by pulling SDA low on the 9th clock cycle. (See **Figure 12-2**)

12.4.3 Data Transfer

Once a successful slave addressing is achieved, the data transfer can proceed byte by byte in the direction specified by the R/W- bit sent by the calling master.

Each data byte is 8 bits long. Data can be changed only when SCL is low and must be held stable when SCL high as shown in **Figure 12-2**. The MSB is transmitted first and each byte has to be followed by an acknowledge bit. This is signalled by the receiving device by pulling the SDA low on the 9th clock cycle. Therefore one complete data byte transfer needs 9 clock cycles.

If the slave receiver does not acknowledge the master, the SDA line should be left high by the slave. The master can then generate a STOP signal to abort the data transfer or a START signal (repeated start) to commence a new transfer.

If the master receiver does not acknowledge the slave transmitter after a byte has been transmitted, it means an “end of data” to the slave. The slave should now release the SDA line for the master to generate a “STOP” or “START” signal.

12.4.4 Repeated START Signal

As shown in **Figure 12-2**, a repeated START signal is used to generate a START signal without first generating a STOP signal to terminate the communication. This is used by the master to communicate with another slave or with the same slave in a different mode (transmit/receive mode) without releasing the bus.

12.4.5 STOP Signal

With reference to **Figure 12-2**, the master can terminate the communication by generating a STOP signal to free the bus. However, the master may generate a START signal followed by a calling command without first generating a STOP signal. This is called repeat start. A STOP signal is defined as a low to high transition of SDA while SCL is at logical high.

12.4.6 Arbitration Procedure

This interface circuit is a true multi-master system which allows more than one master to be connected to it. If two or more masters try to control the bus at the same time, a clock synchronization procedure determines the bus clock, for which the low period is equal to the longest clock low period and the high is equal to the shortest one among the masters. A data arbitration procedure determines the priority. The masters will lose arbitration if they transmit logic “1” while another transmits logic “0”. The losing masters will immediately switch over to slave receive mode and stop its data and clock outputs. The transition from master to slave mode will not generate a STOP condition in this case. Meanwhile a software bit will be set by hardware to indicate loss of arbitration.

12.4.7 Clock Synchronization

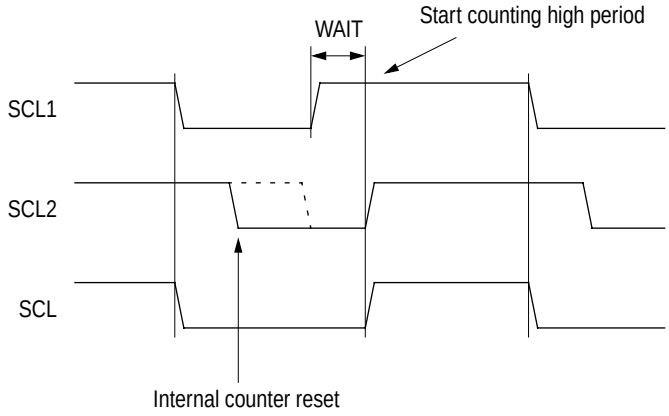


Figure 12-3. Clock Synchronization

Since wired-AND logic is performed on SCL line, a high to low transition on the SCL line will affect the devices connected to the bus. The devices start counting their low period and once a device's clock has gone low, it will hold the SCL line low until the clock high state is reached. However, the change of low to high in this device clock may not change the state of the SCL line if another device clock is still within its low period. Therefore the synchronized clock SCL will be held low by the device with the longest low period. Devices with shorter low periods enter a high wait state during this time (See **Figure 12-2**). When all devices concerned have counted off their low period, the synchronized SCL line will be released and go high. There will then be no difference between the device clocks and the state of the SCL line and all devices will start counting their high periods. The first device to complete its high period will again pull the SCL line low.

12.4.8 Handshaking

The clock synchronization mechanism can be used as a handshake in data transfer. Slave devices may hold the SCL low after completion of one byte. In such cases the device will halt the bus clock and force the master clock into a wait state until the slave releases the SCL line.

12.5 SM-BUS REGISTERS

There are five registers used in the SM-Bus interface. They are described in the following paragraphs.

12.5.1 SM-Bus Address Register (SMADR)

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
SMADR	R	SMAD7	SMAD6	SMAD5	SMAD4	SMAD3	SMAD2	SMAD1	
$0020	W								
reset:		0	0	0	0	0	0	0	U

SMAD1-SMAD7 are the slave address bits of the SM-Bus module.

12.5.2 SM-Bus Frequency Divider Register (SMFDR)

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
SMFDR	R				FD4	FD3	FD2	FD1	FD0
$0021	W								
reset:		U	U	U	0	0	0	0	0

FD0-FD4 are used for clock rate selection. The serial bit clock frequency is equal to the CPU clock divided by the divider shown in **Table 12-1**.

For a 4MHz external crystal operation (2MHz internal operating frequency), the serial bit clock frequency of the SM-Bus ranges from 460Hz to 90909Hz. After POR the clock rate is set to 90909Hz.

Table 12-1. SM-Bus Clock Prescaler

FD4, FD3, FD2, FD1, FD0	DIVIDER	FD4,FD3, FD2, FD1, FD0	DIVIDER
00000	22	10000	352
00001	24	10001	384
00010	28	10010	448
00011	34	10011	544
00100	44	10100	704
00101	48	10101	768
00110	56	10110	896
00111	68	10111	1088
01000	88	11000	1408
01001	96	11001	1536
01010	112	11010	1792
01011	136	11011	2176
01100	176	11100	2816
01101	192	11101	3072
01110	224	11110	3584
01111	272	11111	4352

12.5.3 SM-Bus Control Register (SMCR)

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
SMCR	R	SMEN	SMIEN	SMSTA	SMTX	TXAK	SMUX		
$0022	W								
reset:		0	0	0	0	0	0	U	U

SMEN — SM-Bus Enable

If the SM-Bus enable bit (SMEN) is set, the SM-Bus interface system is enabled. If SMEN is cleared, the interface is reset and disabled.

The SMEN bit must be set first before any bits of SMCR are set.

1 = SM-Bus enabled.
0 = SM-Bus disabled.

SMIEN — SM-Bus Interrupt Enable

If the SM-Bus interrupt enable bit (SMIEN) is set, the interrupt occurs provided the SMIF flag in the status register is set and the I-bit in the Condition Code Register is cleared. If SMIEN is cleared, the SM-Bus interrupt is disabled.

1 = SM-Bus interrupt enabled.
0 = SM-Bus interrupt disabled.

SMSTA — Master/Slave Select

Upon reset, this bit is cleared. When this bit is changed from 0 to 1, a START signal is generated on the bus, and master mode is selected. When this bit is changed from 1 to 0, a STOP signal is generated and the operating mode changes from master to slave.

In master mode, a bit clear immediately followed by a bit set of this bit generates a repeated START signal (see **Figure 12-2**) without generating a STOP signal.

1 = SM-Bus is set for master mode operation.
0 = SM-Bus is set for slave mode operation.

SMTX — Transmit/Receive Mode Select

This bit selects the SM-Bus to transmit or receive.

1 = SM-Bus is set for transmit mode.
0 = SM-Bus is set for receive mode.

TXAK — Acknowledge Enable

If the transmit acknowledge enable bit (TXAK) is cleared, an acknowledge signal will be sent out to the bus at the 9th clock bit after receiving one byte data. When TXAK is set, no acknowledge signal response (i.e., acknowledge bit = 1).

1 = Do not send acknowledge signal.
0 = Send acknowledge signal at 9th clock bit.

SMUX — SM-Bus Channel Select

The SMUX bit selects the channel for SM-Bus communications.

1 = Channel 1 (SDA1 and SCL1 pins) selected for SM-Bus.
0 = Channel 0 (SDA0 and SCL0 pins) selected for SM-Bus.

12.5.4 SM-Bus Status Register (SMSR)

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
SMSR	R	SMCF	SMAAS	SMBB	SMAL		SRW	SMIF	RXAK
$0023	W				SMAL clr			SMIF clr	
reset:		1	0	0	0	0	0	0	1

SMCF — SM-Bus Data Transfer Complete

This bit indicates when a byte of data is being transmitted. When this bit is set, the SMIF is also set. An interrupt request to the CPU is generated if the SMIEN bit is also set.

1 = A byte transfer has been completed.
0 = A byte is being transferred.

SMAAS — SM-Bus Addressed as Slave

This bit is set when its own specific address (SMADR) matches the calling address. When this bit is set, the SMIF is also set. An interrupt request to the CPU is generated if the SMIEN bit is also set. Then CPU needs to check the SRW bit and set its SMTX bit accordingly. Writing to the SM-Bus Control Register clears this bit.

1 = Currently addressed as a slave.
0 = Not addressed.

SMBB — SM-Bus Busy

This bit indicates the status of the bus. When a START signal is detected, the SMBB is set. If a STOP signal is detected, it is cleared.

1 = SM-Bus busy.
0 = SM-Bus idle.

SMAL — SM-Bus Arbitration Lost

This bit is set by hardware when the arbitration procedure is lost during a master transmission. When this bit is set, the SMIF is also set. An interrupt request to the CPU is generated if the SMIEN bit is also set. This bit must be cleared by software.

1 = Lost arbitration in master mode.
0 = No arbitration lost.

SRW — Slave Read/Write Select

When SMAAS is set, the R/W command bit of the calling address sent from master is latched into the R/W command bit (SRW). By checking this bit, the CPU can select slave transmit/receive mode according to the command of master.

1 = Read from slave, from calling master.
0 = Write to slave from calling master.

SMIF — SM-Bus Interrupt Flag

1 = An SM-Bus interrupt has occurred.
0 = An SM-Bus interrupt has not occurred.

This bit is set when one of the following events occur:

– Transmission (either transmit or receive mode) of one byte completed. The bit is set at the falling edge of the 9th clock.

– Receive a calling address which matches its own specific address in slave receive mode.

– Arbitration lost.

RXAK — Receive Acknowledge

When this bit is “0”, it indicates an acknowledge signal has been received after the completion of 8 bits data transmission on the bus. If RXAK is “1”, it means no acknowledge signal is detected at the 9th clock. This bit is set upon reset.

1 = No acknowledgment signal detected.
0 = Acknowledgment signal detected after 8 bits data transmitted.

12.5.5 SM-Bus Data I/O Register (SMDR)

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
SMDR	R	SMD7	SMD6	SMD5	SMD4	SMD3	SMD2	SMD1	SMD0
$0024	W								
reset:		0	0	0	0	0	0	0	0

In master transmit mode, data written to this register is sent (MSB first) to the bus automatically. In master receive mode, reading from this register initiates receiving of the next byte of data. In slave mode, the same function is available after it is addressed.

12.5.6 SM-Bus logic Level

Two choices of logic level is available for the SM-Bus: TTL or CMOS.

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
MCR	R	TSEN	LVRON	0	SCLK	CSSEL	TCSEL	ESVEN	SMINLEV
$000B	W			COPON					
reset:		0	1	0	0	0	0	0	0

Figure 12-4. Miscellaneous Control Register (MCR)

SMINLEV — SM-Bus Input Level Select

This read/write bit selects whether SM-Bus input level is TTL or CMOS. Reset clears the SMINLEV bit.

1 = TTL input level is selected.
0 = CMOS input level is selected.

12.5.7 SCL as16-bit Timer Input Capture

The SCL signal can be routed to the 16-bit Timer Input Capture by setting the TCSEL bit in the Miscellaneous Control Register.

TCSEL — 16-bit Timer Input Capture Source Select

This read/write bit selects the input capture source to the 16-bit Timer. Reset clears TCSEL.

1 = SM-Bus SCL is routed to input capture of 16-bit Timer.
0 = CPF or external TCAP pin (depends on the state of ICEN bit in ACR, $1D) is routed to 16-bit Timer.

See section Input Capture of 16-bit Timer for more details.

12.6 PROGRAMMING CONSIDERATIONS

12.6.1 Initialization

1. Update Frequency Divider Register (FDR) to select a SCL frequency.
2. Update SM-Bus Address Register (SMADR) to define its own slave address.
3. Set SMEN bit of SM-Bus Control Register (SMCR) to enable the SM-Bus interface system.
4. Modify the bits of SM-Bus Control Register (SMCR) to select Master/ Slave mode, Transmit/Receive mode, interrupt enable or not.

12.6.2 Generation of a START Signal and the First Byte of Data Transfer

After completion of the initialization procedure, serial data can be transmitted by selecting the “master transmitter” mode. If the device is connected to a multi-master bus system, the state of the SM-Bus busy bit (SMBB) must be tested to check whether the serial bus is free. If the bus is free (SMBB = 0), the start condition and the first byte (the slave address) can be sent. An example of a program which generates the START signal and transmits the first byte of data (slave address) is shown below:

```
            SEI                         ; DISABLE INTERRUPT
CHFALG      BRSET       5,SMSR,CHFLAG   ; CHECK THE SMBB BIT OF THE
                                        ; STATUS REGISTER. IF IT IS
                                        ; SET, WAIT UNTIL IT IS CLEAR
TXSTART     BSET        4,SMCR          ; SET TRANSMIT MODE
            BSET        5,SMCR          ; SET MASTER MODE
                                        ; i.e. GENERATE START CONDITION
            LDA         #CALLING        ; GET THE CALLING ADDRESS
            STA         SMDR            ; TRANSMIT THE CALLING
                                        ; ADDRESS
            CLI                         ; ENABLE INTERRUPT
```

12.6.3 Software Responses after Transmission or Reception of a Byte

Transmission or reception of a byte will set the data transferring bit (SMCF) to 1, which indicates one byte communication is finished. Also, the SM-Bus interrupt bit (SMIF) is set to generate an SM-Bus interrupt if the interrupt function is enable during initialization. Software must clear the SMIF bit in the interrupt routine first. The SMCF bit will be cleared by reading from the SM-Bus DATA I/O Register (SMDR) in receive mode or writing to SMDR in transmit mode. Software may serve the SM-Bus I/O in the main program by monitoring the SMIF bit if the interrupt function is disabled. The following is an example of a software response by a “master transmitter” in the interrupt routine (see **Figure 12-5**).

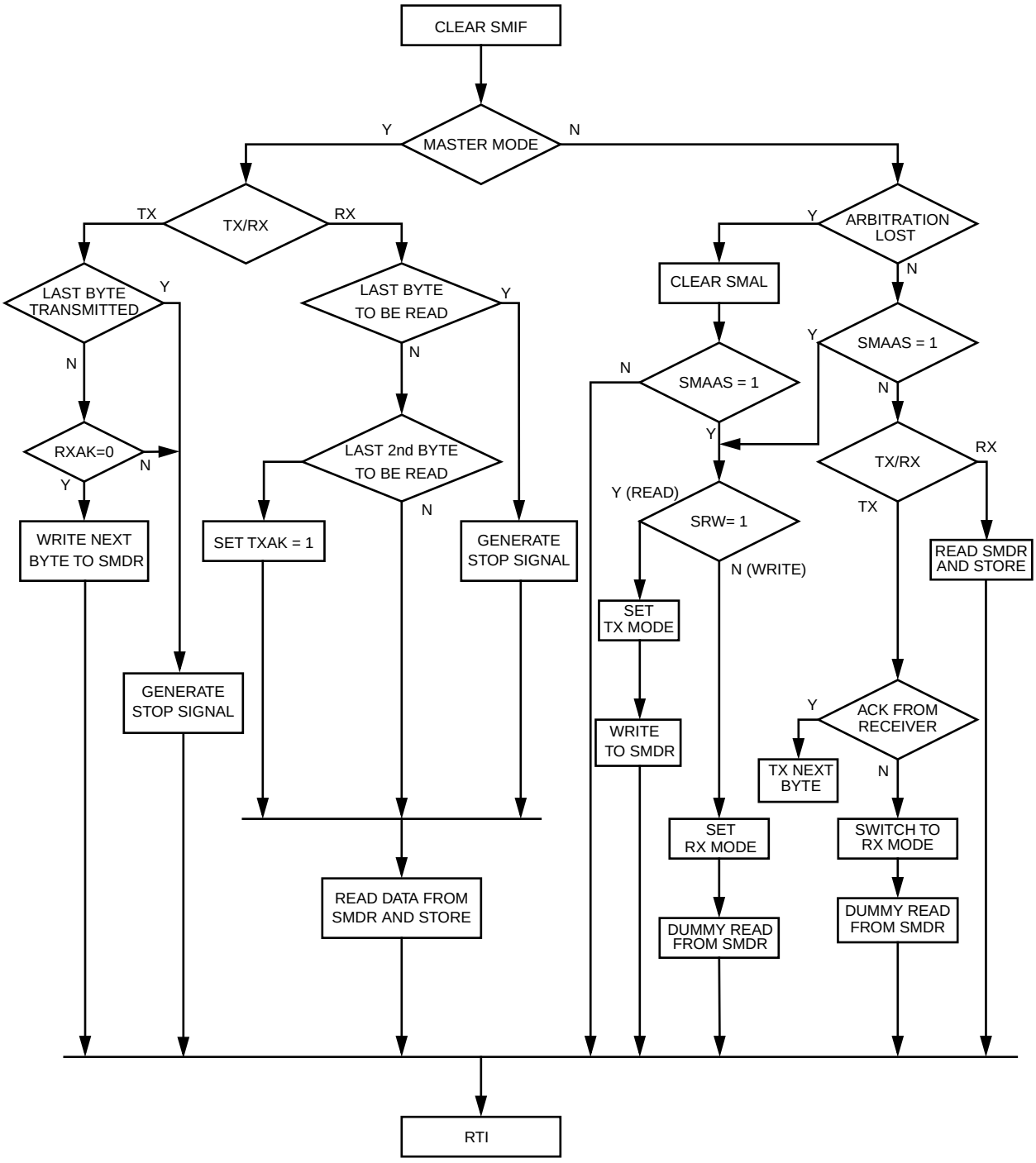


Figure 12-5. Flow-chart of SM-Bus Interrupt Routine

```
ISR         BCLR        1,SMSR                  ; CLEAR THE SMIF FLAG
            BRCLR       5,SMCR,SLAVE            ; CHECK THE SMSTA FLAG,
                                                ; BRANCH IF SLAVE MODE
            BRCLR       4,SMCR,RECEIVE          ; CHECK THE MODE FLAG,
                                                ; BRANCH IF IN RECEIVE MODE
            BRSET       0,SMSR,END              ; CHECK ACK FROM RECEIVER
                                                ; IF NO ACK, END OF
                                                ; TRANSMISSION
TRANSMIT    LDA         DATABUF                 ; GET THE NEXT BYTE OF DATA
            STA         SMDR                    ; TRANSMIT THE DATA
```

12.6.4 Generation of the STOP Signal

A data transfer ends with a STOP signal generated by the “master” device. A master transmitter can simply generate a STOP signal after all the data has been transmitted. The following is an example showing how a stop condition is generated by a master transmitter:

```
MASTX       BRSET       0,SMSR,END              ; IF NO ACK, BRANCH TO END
            LDA         TXCNT                   ; GET VALUE FROM THE
                                                ; TRANSMITTING COUNTER
            BEQ         END                     ; IF NO MORE DATA, BRANCH TO
                                                ; END
            LDA         DATABUF                 ; GET NEXT BYTE OF DATA
            STA         SMDR                    ; TRANSMIT THE DATA
            DEC         TXCNT                   ; DECREASE THE TXCNT
            BRA         EMASTX                  ; EXIT
END         BCLR        5,SMCR                  ; GENERATE A STOP CONDITION
EMASTX      RTI                                 ; RETURN FROM INTERRUPT
```

If a master receiver wants to terminate a data transfer, it must inform the slave transmitter by not acknowledging the last byte of data. This can be done by setting the transmit acknowledge bit (TXAK) before reading the 2nd last byte of data. Before reading the last byte of data, a STOP signal must be generated first. The following is an example showing how a STOP signal is generated by a master receiver.

```
MASR        DEC         RXCNT
            BEQ         ENMASR                  ; LAST BYTE TO BE READ
            LDA         RXCNT
            DECA                                ; CHECK LAST 2ND BYTE TO
                                                ; BE READ
            BNE         NXMAR                   ; NOT LAST ONE OR LAST SECOND
LAMAR       BSET        3,SMCR                  ; LAST SECOND, DISABLE ACK
                                                ; TRANSMITTING
            BRA         NXMAR
ENMASR      BCLR        5,SMCR                  ; LAST ONE, GENERATE “STOP”
                                                ; SIGNAL
NXMAR       LDA         SMDR                    ; READ DATA AND STORE
            STA         RXBUF
            RTI
```

12.6.5 Generation of a Repeated START Signal

If at the end of data transfer the master still wants to communicate on the bus, it can generate another START signal followed by another slave address without first generating a STOP signal. A program example is shown below.

```
RESTART     BCLR        5,SMCR                  ; ANOTHER START (RESTART) IS
            BSET        5,SMCR                  ; GENERATED BY THESE TWO
                                                ; CONSEQUENCE INSTRUCTION
            LDA         #CALLING                ; GET THE CALLING ADDRESS
            STA         SMDR                    ; TRANSMIT THE CALLING ADDRESS
```

12.6.6 Slave Mode

In the slave service routine, the master addressed as slave bit (SMAAS) should be tested to see if a calling of its own address has just been received. If SMAAS is set, software should set the transmit/receive mode select bit (SMTX bit of SMCR) according to the R/W-command bit (SRW). Writing to the SMCR clears the SMAAS automatically. A data transfer may then be initiated by writing information to SMDR or dummy reading from SMDR.

In the slave transmitter routine, the received acknowledge bit (RXAK) must be tested before transmitting the next byte of data. If RXAK is set, indicating an “end of data” signal from the master receiver, then it must switch from transmitter mode to receiver mode by software and a dummy read must follow to release the SCL line so that the master can generate a stop signal.

12.6.7 Arbitration Lost

If more than one master want to acquire the bus simultaneously, only one master wins and the others lost arbitration. The arbitration lost devices immediately switch to slave receive mode by hardware. Their data output to the SDA line is stopped, but internal transmitting clock still run until the end of the byte transmitting. An interrupt occurs when this dummy “byte” transmitting is accomplished with SMAL=1 and SMSTA = 0. If one master attempt to start transmission while the bus is being engaged by another master, the hardware will inhibit the transmission; switch the SMSTA bit from 1 to 0 without generating STOP condition; generate an interrupt to CPU and set the SMAL to indicate that the attempt to engage the bus is failed. Consideration of these cases, the slave service routine should test the SMAL first, software should clear the SMAL bit if it is set.

12.7 OPERATION DURING WAIT MODE

During WAIT mode the SM-Bus block is idle. If in slave mode it will wake up on receiving a valid start condition. If the interrupt is enabled the CPU will come out of WAIT mode after the end of a byte transmission.

12.8 OPERATION DURING STOP MODE

In Stop Mode the SM-Bus is disabled.

SECTION 13
CURRENT SENSE AMPLIFIER

The Current Sense Amplifier module, used in conjunction with the Analog Subsystem, is designed to monitor charge and discharge currents in smart battery applications.

13.1 CURRENT SENSE AMPLIFIER APPLICATION

A typical connection for the Current Sense Amplifier (CSA) block is illustrated in **Figure 13-1**. With a sense resistor, R_{SENSE} of 0.01Ω, the voltage setup across the node CSA and V_{SS} (ground) will vary to the current (in either charging or discharging mode) as shown in **Table 13-1**.

Table 13-1. Voltage Across the Sense Resistor against Current

Current flowing	Voltage across the Sense Resistor, R_{SENSE}
10mA	0.1mV
1A	10mV
5A	50mV
R_{SENSE}=0.01Ω	

In this case, the CSA is required to measure a current from 10mA to 5A over the operating temperature from 0°C to 70°C.

With the A/D in the Analog Subsystem set up for 12-bit resolution, the step size is approximately 1.22mV (V_{DD}=5V). To measure the 0.1mV for the 10mA current flow, a gain of greater than 10 is required.

The CSA module is designed with two gain settings, 10 and 30. With a 10-bit A/D, and a gain of 30, the CSA can measure current with a typical resolution of 17mA steps.

After amplification, the resultant signal is fed to channel 6 (MUX6) of the analog subsystem for A/D conversion.

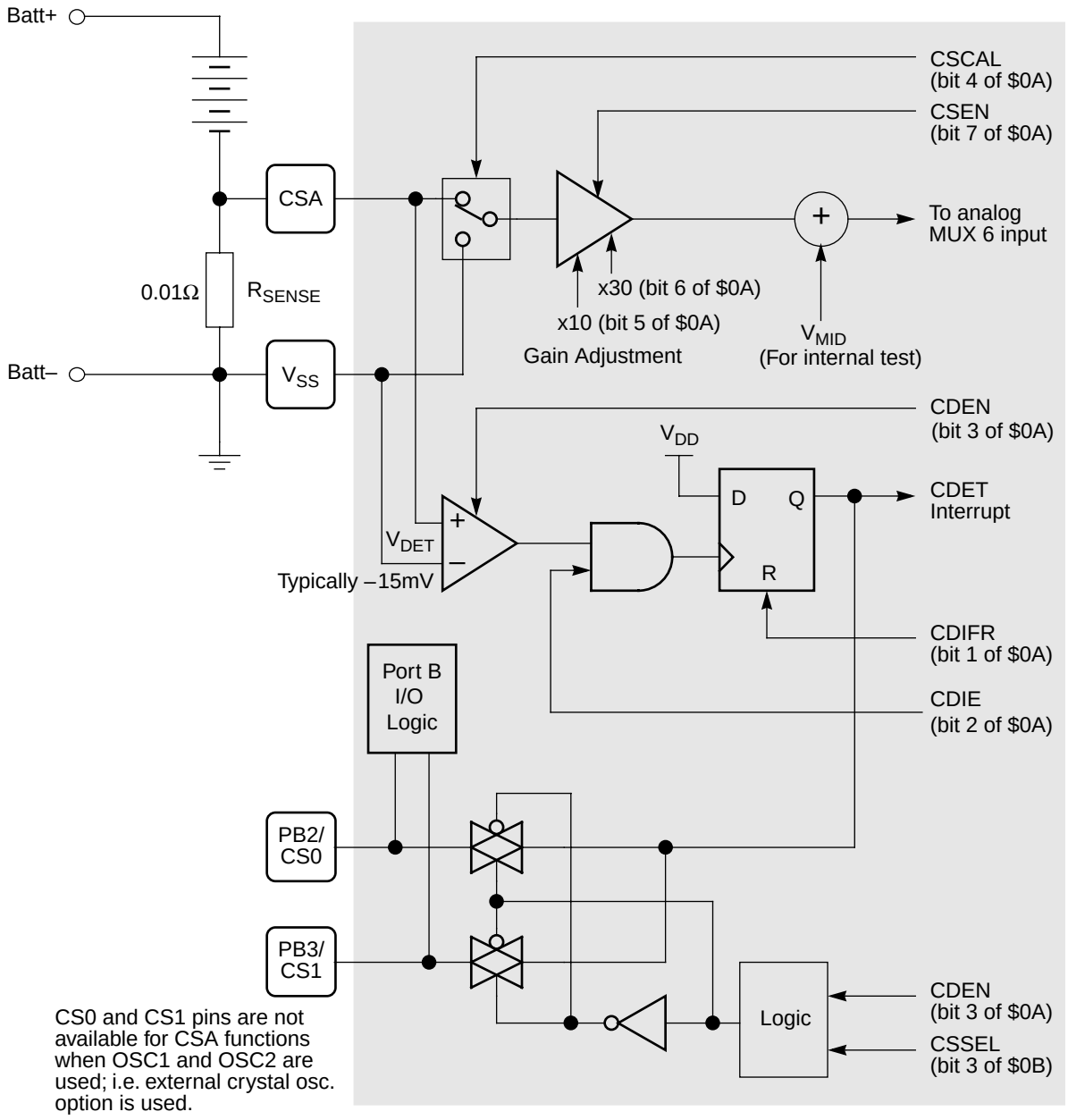


Figure 13-1. Current Sense Amplifier Block

13.2 CURRENT SENSE INTERRUPT

The CSA can generate an interrupt once it detects a (discharge) current passes through the current sensing resistor, R_{SENSE}. The trip current depends on the value of the sense resistor; it is voltage developed across R_{SENSE}, V_{DET} that trips the interrupt. V_{DET} is set typically at –15mV, with –10mV being the minimum.

13.3 CSA STATUS AND CONTROL REGISTER (CSSCR)

The CSA status and control register is shown in **Figure 13-2**.

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
CSSCR	R	CSEN	X30	X10	CSCAL	CDEN	CDIE	0	CDIF
$000A	W							CDIFR	
reset:		0	0	0	0	0	0	0	0

Figure 13-2. CSA Status and Control Register (CSSCR)

CSEN — Current Sense Amplifier Enable

This read/write bit enables the CSA module. Reset clears the CSEN bit.

1 = CSA block enabled.
0 = CSA block disabled.

X30, X10 — Current Sense Amplifier Gain Select

These read/write bits enable the respective gain to be selected. See **Table 13-2**. Reset clears the X30 and X10 bits.

Table 13-2. Current Sense Amplifier Gain Select

X30	X10	GAIN SELECTED
0	Don't care	X10
1	0	X30
1	1	Undetermined

CSCAL — Current Sense Amplifier Calibration Enable

This read/write bit enables the CSA calibration. Reset clears the CSCAL bit.

1 = CSA calibration enabled; current amplifier input connected to ground (V_{SS}).
0 = CSA calibration disabled; current amplifier input from CSA pin.

CDEN — Current Detect Enable

This read/write bit enables the current detect comparator and current detect output pin (CS0 or CS1) logic. Reset clears the CDEN bit.

1 = Current detect comparator enabled.
0 = Current detect comparator disabled.

CDIE — Current Detect Interrupt Enable

This read/write bit enables interrupts caused by detecting the current passing through the sensing resistor, R_{SENSE}. Reset clears the CDIE bit.

1 = Current detect interrupt enabled.
0 = Current detect interrupt disabled.

CDIFR — Current Detect Interrupt Flag Reset

Writing a logic "1" to this write-only bit clears the CDIF bit. CDIFR always reads as a logic zero. Reset does not affect CDIFR.

1 = Clear CDIF bit.
0 = No affect on CDIF bit.

CDIF — Current Detect Interrupt Flag

This read-only bit is set when the voltage developed across the sense resistor, R_{SENSE} is equal to or greater than V_{DET} (the CSA comparator trip voltage, typically –15mV) CDIF generates an interrupt request to the CPU if CDIE is also set. The CDIF bit is cleared by writing a logic “1” to the CDIFR bit. Writing to CDIF has no effect. Reset clears CDIF.

1 = Current detect interrupt has occurred.
0 = No current detect interrupt since CDIF last cleared.

If the OSC1 and OSC2 pins are not enabled (by mask option). The current detect interrupt from CDIF bit can be reflected to one of two output port pins, PB2/CS0 and PB3/CS1.

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
MCR	R	TSEN	LVRON	0	SCLK	CSSEL	TCSEL	ESVEN	SMINLEV
$000B	W			COPON					
reset:		0	1	0	0	0	0	0	0

U = UNAFFECTED BY RESET

Figure 13-3. Miscellaneous Control Register (MCR)

CSSEL — Current Sense detect output Select

This read/write bit selects either CS0 pin or CS1 pin is used to reflect the current detect interrupt. Reset clears the CSSEL bit.

1 = CS1 enabled, CS0 disabled.
0 = CS0 enabled, CS1 disabled.

Table 13-3. Current Detect Output Select

CDEN	CSSEL	PB2/CS0	PB3/CS1
0	0	PB2	PB3
0	1	PB2	PB3
1	0	CS0	PB3
1	1	PB2	CS1

CS0 and CS1 are not available when OSC1 and OSC2 are used for external oscillator option.

13.4 CSA OPERATION DURING WAIT MODE

In WAIT mode the CSA module continues to operate and may generate an interrupt to trigger the MCU out of WAIT mode.

13.5 CSA OPERATION DURING STOP MODE

In STOP mode the CSA module is disabled; but a CSA interrupt (by CDIF) can wake-up the MCU from the STOP mode.

SECTION 14
TEMPERATURE SENSOR

The MC68HC05SB7 MCU can measure temperature in two ways: by using the internal temperature sensor, or by using an external thermistor.

14.1 INTERNAL TEMPERATURE SENSOR

The internal temperature sensor is designed to measure temperature over the 0°C to 70°C range; with its voltage output connected to channel 5 of the Analog Subsystem (AN5, see Analog Subsystem section).

The temperature sensor is disabled/enabled by the TSEN bit in the Miscellaneous Control Register at $0B. The TSEN bit also disables/enables the BandGap reference voltage.

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
MCR	R	TSEN	LVRON	0	SCLK	CSSEL	TCSEL	ESVEN	SMINLEV
$000B	W			COPON					
reset:		0	1	0	0	0	0	0	0

Figure 14-1. Miscellaneous Control Register (MCR)

TSEN — Internal Temperature Sensor and BandGap Reference Enable

This read/write bit enables the internal temperature sensor and BandGap reference. Reset clears TSEN.

1 = Temperature sensor and bandgap reference enabled.
0 = Temperature sensor and bandgap reference disabled.

NOTE

The temperature gradient is typically 2.2mV/°C ±10%.

The internal temperature sensor is a semiconductor type sensor. Due to process variations, the absolute output voltage at a given temperature will vary from one device to another. It is the user's responsibility to measure and calibrate the temperature sensor output voltage when the MCU is in the target system.
As an option, the temperature sensor voltage at 80°C is available preprogrammed into the PEPROM. See PEPROM section.

14.2 EXTERNAL TEMPERATURE SENSOR

In fast charge control applications, where close monitoring of the charging process is required (especially temperature), an external temperature sensor (thermistor) is recommended. This external thermistor connects to the TM pin (see **Figure 14-2**), and its voltage measured via channel 4 (AN4, see Analog Subsystem section) of the Analog Subsystem. For faster temperature response time and more accurate measurement (required for fast charge control), the thermistor should be mounted directly to the battery pack.

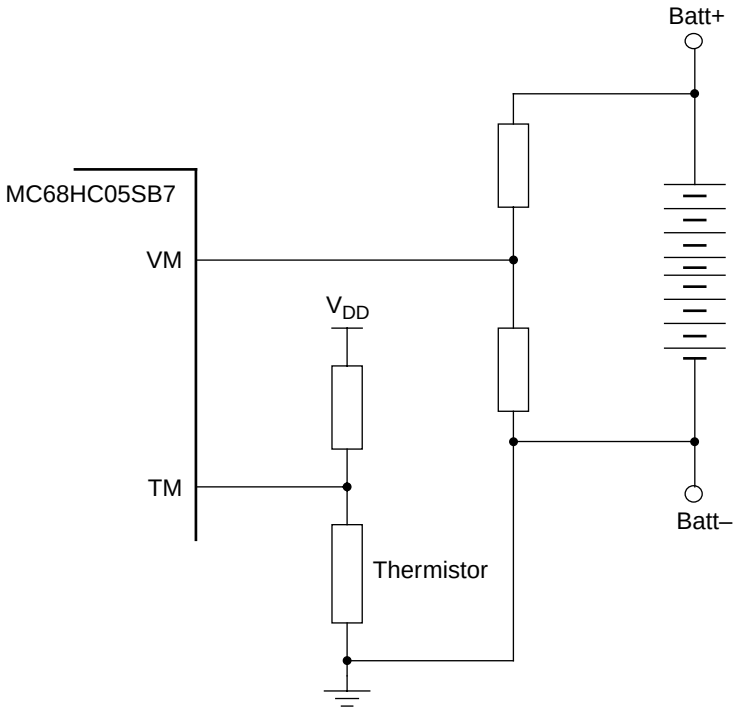


Figure 14-2. External Temperature Sensor Connection

14.3 TEMPERATURE SENSOR OPERATION DURING WAIT MODE

During WAIT mode the temperature sensor continues to operate normally.

14.4 TEMPERATURE SENSOR OPERATION DURING STOP MODE

In STOP mode the temperature sensor is disabled.

SECTION 15
ANALOG SUBSYSTEM

The analog subsystem of the MC68HC05SB7 is based on an on-chip voltage comparator as shown in **Figure 15-1**.

This configuration provides following features:

- The voltage comparator with external access to both inverting and non-inverting inputs
- The voltage comparator can be connected as a single-slope A/D. The possible single-slope A/D connection provides the following features:
  - A/D conversions can use V_{DD} or an external voltage as a reference with software used to calculate ratiometric or absolute results
  - Channel access to up to eight inputs via multiplexer control with independent multiplexer control allowing multiple input connections
  - Access to V_{DD} and V_{SS} for calibration
  - Divide by 2 to extend input voltage range
  - The comparator can be inverted to calculate input offsets
  - Internal sample and hold capacitor

Voltages are resolved by measuring the time it takes an external capacitor to charge up to the level of the unknown input voltage that is being measured. The beginning of the A/D conversion time can be started by several means:

- Output compare from the 16-bit programmable Timer
- Timer overflow from the 16-bit programmable Timer
- Direct software control via a register bit

The end of the A/D conversion time can be captured by several means:

- Input capture in the 16-bit programmable Timer
- Interrupt generated by the comparator output
- Software polling of the comparator output using software loop time

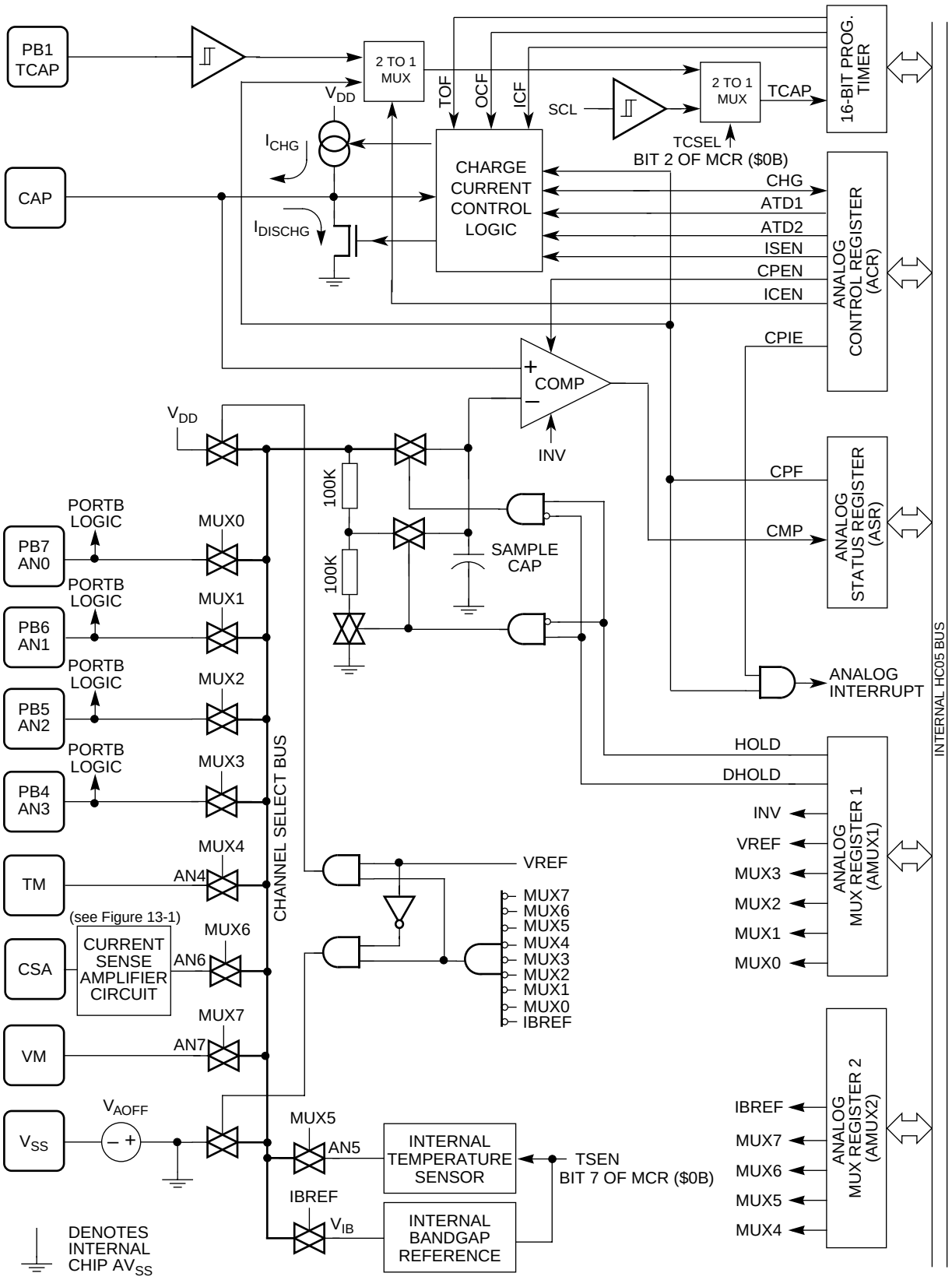


Figure 15-1. Analog Subsystem Block Diagram

15.1 ANALOG MULTIPLEX REGISTERS

The Analog Multiplex Registers (AMUX1 and AMUX2) control the general interconnection and operation. The control bits in AMUX1 and AMUX2 are shown in **Figure 15-2** and **Figure 15-2** respectively.

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
AMUX1	R	HOLD	DHOLD	INV	VREF	MUX3	MUX2	MUX1	MUX0
$0003	W								
reset:		1	0	0	0	0	0	0	0

Figure 15-2. Analog Multiplex Register 1 (AMUX1)

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
AMUX2	R	0		0	IBREF	MUX7	MUX6	MUX5	MUX4
$0007	W								
reset:		0	0	0	0	0	0	0	0

Figure 15-3. Analog Multiplex Register 2 (AMUX2)

HOLD, DHOLD

These read/write bits control the source connection to the input to the negative input of voltage comparator shown in **Figure 15-1**. This allows the channel selection bus or the 1:2 divided channel selection bus to charge the internal sample capacitor and to also be presented to comparator. The decoding of these sources is given in **Table 15-1**. During a reset the HOLD bit is set and the DHOLD bit is cleared, which connects the internal sample capacitor to the channel selection bus. And since a reset also clears the MUX0:7 bits then the channel selection bus will be connected to V_{SS} and the internal sample capacitor will be discharged to V_{SS} following the reset.

Table 15-1. Comparator Input Sources

HOLD	DHOLD	Case	Source To Negative Input of Comparator
0	0	Sample Hold	Internal sample capacitor connected to only the negative input of comparator; and subjected to a very low leakage current.
0	1	Divided Input	Signal to channel selection bus is divided by 2 and connected to both the internal sample capacitor and negative input of comparator.
1	0	Direct Input	Signal to channel selection bus is connected directly to both the internal sample capacitor and negative input of comparator.
1	1	Not allowed	—

NOTE

When sampling a voltage for later conversion the HOLD and DHOLD bit should be cleared before making any changes in the MUX channel selection. If the MUX channel and the HOLD/DHOLD are changed on the same write cycle to the AMUX1 register, the sampled voltage may be altered during the channel switching.

INV

This is a read/write bit that controls the phase of the voltage comparator. This bit allows voltage comparisons with either input node of the voltage comparator to be presented to the rest of the circuit as the “positive” or “negative” input.

The voltage comparator is defined as non-inverted when the internal positive node is connected to the external positive input and the output is not inverted. In this case the output will go to a logical one when the voltage on the positive input is higher than the voltage on the negative input. Any input offset voltage in the voltage comparator will be with respect to the negative input.

The voltage comparator is defined as inverted when the internal negative node is connected to the external positive input and the output is inverted. In this case the output will still go to a logical one when the voltage on the positive input is higher than the voltage on the negative input. In the inverted case any input offset voltage in the voltage comparator will be with respect to the positive input. This bit is cleared by a reset of the device.

1 = The voltage comparator is internally inverted.
0 = The voltage comparator is not internally inverted.

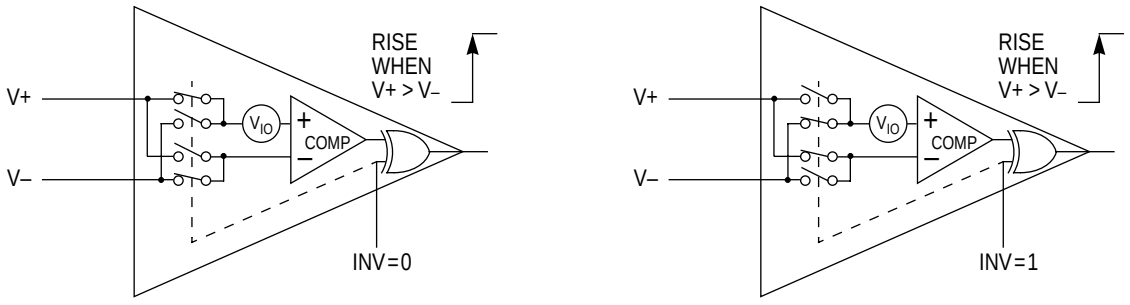


Figure 15-4. INV Bit Action

NOTE

The effect of changing the state of the INV bit is to only change the polarity of the input offset voltage. It does not change the output phase of the CPF flag with respect to the external port pins.

The comparator may generate an output flag when the inputs are exchanged due to a change in the state of the INV bit. It is therefore recommended that the INV bit

not be changed while waiting for a comparator flag. Further, any changes to the state of the INV bit should be followed by writing a logical one to CPFR bit to clear an extraneous CPF flag that may have occurred.

VREF

This is a read/write bit that connects the channel select bus to V_{DD} for purposes of making a reference voltage measurement. It cannot be selected if any of the other input sources to the channel select bus are selected as shown in **Table 15-2**. This bit is cleared by a reset of the device.

1 = Channel select bus connected to V_{DD} if MUX7:0 and IBREF are cleared.

0 = Channel select bus cannot be connected to V_{DD}.

IBREF

This is a read/write bit that connects the channel select bus to V_{IB} for purposes of making a reference voltage measurement. It cannot be selected if any of the other input sources to the channel select bus are selected as shown in **Table 15-2**. This bit is cleared by a reset of the device.

1 = Channel select bus connected to V_{IB} if MUX7:0 and VREF are cleared.

0 = Channel select bus cannot be connected to V_{IB}.

MUX7:0

These are read/write bits that connect the analog subsystem pins to the channel select bus and voltage comparator for purposes of making a voltage measurement. They can be selected individually or combined with any of the other input sources to the channel select bus as shown in **Table 15-2**.

NOTE

The V_{AOFF} voltage source shown in **Figure 15-1** depicts a small offset voltage generated by the total chip current passing through the package bond wires and lead frame that are attached to the single V_{SS} pin. The offset raises the internal V_{SS} reference (AV_{SS}) in the analog subsystem with respect to the external V_{SS} pin. Turning on the V_{SS} MUX to the channel select bus connects it to this internal AVSS reference line.

When making A/D conversions this AV_{SS} offset gets placed on the external ramping capacitor since the discharge device on the CAP pin discharges the external capacitor to the internal AV_{SS} line. Under these circumstances the positive input (+) to the comparator will always be higher than the negative input (–) until the negative input reaches the AV_{SS} offset voltage plus any offset in the comparator.

Therefore, input voltages cannot be resolved if they are less than the sum of the AV_{SS} offset and the comparator offset, because they will always yield a low output from the comparator

---

Table 15-2. Channel Select Bus Combinations

Analog Multiplex Registers (AMUX1 and AMUX2)										Channel Select Bus Connected to:										
VREF	**IBREF**	**MUX7**	**MUX6**	**MUX5**	**MUX4**	**MUX3**	**MUX2**	**MUX1**	**MUX0**	**V_{DD}**	**V_{IB}**	**AN7**	**AN6**	**AN5**	**AN4**	**AN3**	**AN2**	**AN1**	**AN0**	**V_{SS}**
0	0	0	0	0	0	0	0	0	0	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	ON
X	X	0	0	0	0	0	0	0	1	Z	Z	Z	Z	Z	Z	Z	Z	Z	ON	Z
X	X	0	0	0	0	0	0	1	0	Z	Z	Z	Z	Z	Z	Z	Z	ON	Z	Z
X	X	0	0	0	0	0	0	1	1	Z	Z	Z	Z	Z	Z	Z	Z	ON	ON	Z
X	X	0	0	0	0	0	1	0	0	Z	Z	Z	Z	Z	Z	Z	ON	Z	Z	Z
X	X	0	0	0	0	0	1	0	1	Z	Z	Z	Z	Z	Z	Z	ON	Z	ON	Z
X	X	0	0	0	0	0	1	1	0	Z	Z	Z	Z	Z	Z	Z	ON	ON	Z	Z
X	X	0	0	0	0	0	1	1	1	Z	Z	Z	Z	Z	Z	Z	ON	ON	ON	Z
X	X	0	0	0	0	1	0	0	0	Z	Z	Z	Z	Z	Z	ON	Z	Z	Z	Z
X	X	0	0	0	0	1	0	0	1	Z	Z	Z	Z	Z	Z	ON	Z	Z	ON	Z
X	X	0	0	0	0	1	0	1	0	Z	Z	Z	Z	Z	Z	ON	Z	ON	Z	Z
X	X	0	0	0	0	1	0	1	1	Z	Z	Z	Z	Z	Z	ON	Z	ON	ON	Z
X	X	0	0	0	0	1	1	0	0	Z	Z	Z	Z	Z	Z	ON	ON	Z	Z	Z
X	X	0	0	0	0	1	1	0	1	Z	Z	Z	Z	Z	Z	ON	ON	Z	ON	Z
X	X	0	0	0	0	1	1	1	0	Z	Z	Z	Z	Z	Z	ON	ON	ON	Z	Z
X	X	0	0	0	0	1	1	1	1	Z	Z	Z	Z	Z	Z	ON	ON	ON	ON	Z
X	X	0	0	0	1	0	0	0	0	Z	Z	Z	Z	Z	ON	Z	Z	Z	Z	Z
X	X	0	0	0	1	0	0	0	1	Z	Z	Z	Z	Z	ON	Z	Z	Z	ON	Z
X	X	0	0	0	1	0	0	1	0	Z	Z	Z	Z	Z	ON	Z	Z	ON	Z	Z
X	X	0	0	0	1	0	0	1	1	Z	Z	Z	Z	Z	ON	Z	Z	ON	ON	Z
X	X	0	0	0	1	0	1	0	0	Z	Z	Z	Z	Z	ON	Z	ON	Z	Z	Z
X	X	0	0	0	1	0	1	0	1	Z	Z	Z	Z	Z	ON	Z	ON	Z	ON	Z
X	X	0	0	0	1	0	1	1	0	Z	Z	Z	Z	Z	ON	Z	ON	ON	Z	Z
X	X	0	0	0	1	0	1	1	1	Z	Z	Z	Z	Z	ON	Z	ON	ON	ON	Z
X	X	0	0	0	1	1	0	0	0	Z	Z	Z	Z	Z	ON	ON	Z	Z	Z	Z
X	X	0	0	0	1	1	0	0	1	Z	Z	Z	Z	Z	ON	ON	Z	Z	ON	Z
X	X	0	0	0	1	1	0	1	0	Z	Z	Z	Z	Z	ON	ON	Z	ON	Z	Z
X	X	0	0	0	1	1	0	1	1	Z	Z	Z	Z	Z	ON	ON	Z	ON	ON	Z
X	X	0	0	0	1	1	1	0	0	Z	Z	Z	Z	Z	ON	ON	ON	Z	Z	Z
X	X	0	0	0	1	1	1	0	1	Z	Z	Z	Z	Z	ON	ON	ON	Z	ON	Z

Table 15-2. Channel Select Bus Combinations

Analog Multiplex Registers (AMUX1 and AMUX2)										Channel Select Bus Connected to:										
VREF	IBREF	MUX7	MUX6	MUX5	MUX4	MUX3	MUX2	MUX1	MUX0	V_{DD}	V_{IB}	AN7	AN6	AN5	AN4	AN3	AN2	AN1	AN0	V_{SS}
X	X	0	0	0	1	1	1	1	0	Z	Z	Z	Z	Z	ON	ON	ON	ON	Z	Z
X	X	0	0	0	1	1	1	1	1	Z	Z	Z	Z	Z	ON	ON	ON	ON	ON	Z
X	X	0	0	1	0	0	0	0	0	Z	Z	Z	Z	ON	Z	Z	Z	Z	Z	Z
X	X	0	0	1	0	0	0	0	1	Z	Z	Z	Z	ON	Z	Z	Z	Z	ON	Z
X	X	0	0	1	0	0	0	1	0	Z	Z	Z	Z	ON	Z	Z	Z	ON	Z	Z
X	X	0	0	1	0	0	0	1	1	Z	Z	Z	Z	ON	Z	Z	Z	ON	ON	Z
X	X	0	0	1	0	0	1	0	0	Z	Z	Z	Z	ON	Z	Z	ON	Z	Z	Z
X	X	0	0	1	0	0	1	0	1	Z	Z	Z	Z	ON	Z	Z	ON	Z	ON	Z
X	X	0	0	1	0	0	1	1	0	Z	Z	Z	Z	ON	Z	Z	ON	ON	Z	Z
X	X	0	0	1	0	0	1	1	1	Z	Z	Z	Z	ON	Z	Z	ON	ON	ON	Z
X	X	0	0	1	0	1	0	0	0	Z	Z	Z	Z	ON	Z	ON	Z	Z	Z	Z
X	X	0	0	1	0	1	0	0	1	Z	Z	Z	Z	ON	Z	ON	Z	Z	ON	Z
X	X	0	0	1	0	1	0	1	0	Z	Z	Z	Z	ON	Z	ON	Z	ON	Z	Z
X	X	0	0	1	0	1	0	1	1	Z	Z	Z	Z	ON	Z	ON	Z	ON	ON	Z
X	X	0	0	1	0	1	1	0	0	Z	Z	Z	Z	ON	Z	ON	ON	Z	Z	Z
X	X	0	0	1	0	1	1	0	1	Z	Z	Z	Z	ON	Z	ON	ON	Z	ON	Z
X	X	0	0	1	0	1	1	1	0	Z	Z	Z	Z	ON	Z	ON	ON	ON	Z	Z
X	X	0	0	1	0	1	1	1	1	Z	Z	Z	Z	ON	Z	ON	ON	ON	ON	Z
X	X	0	0	1	1	0	0	0	0	Z	Z	Z	Z	ON	ON	Z	Z	Z	Z	Z
X	X	0	0	1	1	0	0	0	1	Z	Z	Z	Z	ON	ON	Z	Z	Z	ON	Z
X	X	0	0	1	1	0	0	1	0	Z	Z	Z	Z	ON	ON	Z	Z	ON	Z	Z
X	X	0	0	1	1	0	0	1	1	Z	Z	Z	Z	ON	ON	Z	Z	ON	ON	Z
X	X	0	0	1	1	0	1	0	0	Z	Z	Z	Z	ON	ON	Z	ON	Z	Z	Z
X	X	0	0	1	1	0	1	0	1	Z	Z	Z	Z	ON	ON	Z	ON	Z	ON	Z
X	X	0	0	1	1	0	1	1	0	Z	Z	Z	Z	ON	ON	Z	ON	ON	Z	Z
X	X	0	0	1	1	0	1	1	1	Z	Z	Z	Z	ON	ON	Z	ON	ON	ON	Z
X	X	0	0	1	1	1	0	0	0	Z	Z	Z	Z	ON	ON	ON	Z	Z	Z	Z
X	X	0	0	1	1	1	0	0	1	Z	Z	Z	Z	ON	ON	ON	Z	Z	ON	Z
X	X	0	0	1	1	1	0	1	0	Z	Z	Z	Z	ON	ON	ON	Z	ON	Z	Z
X	X	0	0	1	1	1	0	1	1	Z	Z	Z	Z	ON	ON	ON	Z	ON	ON	Z
X	X	0	0	1	1	1	1	0	0	Z	Z	Z	Z	ON	ON	ON	ON	Z	Z	Z

Table 15-2. Channel Select Bus Combinations

Analog Multiplex Registers (AMUX1 and AMUX2)										Channel Select Bus Connected to:										
VREF	IBREF	MUX7	MUX6	MUX5	MUX4	MUX3	MUX2	MUX1	MUX0	V_{DD}	V_{IB}	AN7	AN6	AN5	AN4	AN3	AN2	AN1	AN0	V_{SS}
X	X	0	0	1	1	1	1	0	1	Z	Z	Z	Z	ON	ON	ON	ON	Z	ON	Z
X	X	0	0	1	1	1	1	1	0	Z	Z	Z	Z	ON	ON	ON	ON	ON	Z	Z
X	X	0	0	1	1	1	1	1	1	Z	Z	Z	Z	ON	ON	ON	ON	ON	ON	Z
X	X	0	!	0	0	0	0	0	0	Z	Z	Z	ON	Z	Z	Z	Z	Z	Z	Z
X	X	0	1	0	0	0	0	0	1	Z	Z	Z	ON	Z	Z	Z	Z	Z	ON	Z
X	X	0	1	0	0	0	0	1	0	Z	Z	Z	ON	Z	Z	Z	Z	ON	Z	Z
X	X	0	1	0	0	0	0	1	1	Z	Z	Z	ON	Z	Z	Z	Z	ON	ON	Z
X	X	0	1	0	0	0	1	0	0	Z	Z	Z	ON	Z	Z	Z	ON	Z	Z	Z
X	X	0	1	0	0	0	1	0	1	Z	Z	Z	ON	Z	Z	Z	ON	Z	ON	Z
X	X	0	1	0	0	0	1	1	0	Z	Z	Z	ON	Z	Z	Z	ON	ON	Z	Z
X	X	0	1	0	0	0	1	1	1	Z	Z	Z	ON	Z	Z	Z	ON	ON	ON	Z
X	X	0	1	0	0	1	0	0	0	Z	Z	Z	ON	Z	Z	ON	Z	Z	Z	Z
X	X	0	1	0	0	1	0	0	1	Z	Z	Z	ON	Z	Z	ON	Z	Z	ON	Z
X	X	0	1	0	0	1	0	1	0	Z	Z	Z	ON	Z	Z	ON	Z	ON	Z	Z
X	X	0	1	0	0	1	0	1	1	Z	Z	Z	ON	Z	Z	ON	Z	ON	ON	Z
X	X	0	1	0	0	1	1	0	0	Z	Z	Z	ON	Z	Z	ON	ON	Z	Z	Z
X	X	0	1	0	0	1	1	0	1	Z	Z	Z	ON	Z	Z	ON	ON	Z	ON	Z
X	X	0	1	0	0	1	1	1	0	Z	Z	Z	ON	Z	Z	ON	ON	ON	Z	Z
X	X	0	1	0	0	1	1	1	1	Z	Z	Z	ON	Z	Z	ON	ON	ON	ON	Z
X	X	0	1	0	1	0	0	0	0	Z	Z	Z	ON	Z	ON	Z	Z	Z	Z	Z
X	X	0	1	0	1	0	0	0	1	Z	Z	Z	ON	Z	ON	Z	Z	Z	ON	Z
X	X	0	1	0	1	0	0	1	0	Z	Z	Z	ON	Z	ON	Z	Z	ON	Z	Z
X	X	0	1	0	1	0	0	1	1	Z	Z	Z	ON	Z	ON	Z	Z	ON	ON	Z
X	X	0	1	0	1	0	1	0	0	Z	Z	Z	ON	Z	ON	Z	ON	Z	Z	Z
X	X	0	1	0	1	0	1	0	1	Z	Z	Z	ON	Z	ON	Z	ON	Z	ON	Z
X	X	0	1	0	1	0	1	1	0	Z	Z	Z	ON	Z	ON	Z	ON	ON	Z	Z
X	X	0	1	0	1	0	1	1	1	Z	Z	Z	ON	Z	ON	Z	ON	ON	ON	Z
X	X	0	1	0	1	1	0	0	0	Z	Z	Z	ON	Z	ON	ON	Z	Z	Z	Z
X	X	0	1	0	1	1	0	0	1	Z	Z	Z	ON	Z	ON	ON	Z	Z	ON	Z
X	X	0	1	0	1	1	0	1	0	Z	Z	Z	ON	Z	ON	ON	Z	ON	Z	Z
X	X	0	1	0	1	1	0	1	1	Z	Z	Z	ON	Z	ON	ON	Z	ON	ON	Z

Table 15-2. Channel Select Bus Combinations

Analog Multiplex Registers (AMUX1 and AMUX2)										Channel Select Bus Connected to:										
VREF	IBREF	MUX7	MUX6	MUX5	MUX4	MUX3	MUX2	MUX1	MUX0	V_{DD}	V_{IB}	AN7	AN6	AN5	AN4	AN3	AN2	AN1	AN0	V_{SS}
X	X	0	1	0	1	1	1	0	0	Z	Z	Z	ON	Z	ON	ON	ON	Z	Z	Z
X	X	0	1	0	1	1	1	0	1	Z	Z	Z	ON	Z	ON	ON	ON	Z	ON	Z
X	X	0	1	0	1	1	1	1	0	Z	Z	Z	ON	Z	ON	ON	ON	ON	Z	Z
X	X	0	1	0	1	1	1	1	1	Z	Z	Z	ON	Z	ON	ON	ON	ON	ON	Z
X	X	0	1	1	0	0	0	0	0	Z	Z	Z	ON	ON	Z	Z	Z	Z	Z	Z
X	X	0	1	1	0	0	0	0	1	Z	Z	Z	ON	ON	Z	Z	Z	Z	ON	Z
X	X	0	1	1	0	0	0	1	0	Z	Z	Z	ON	ON	Z	Z	Z	ON	Z	Z
X	X	0	1	1	0	0	0	1	1	Z	Z	Z	ON	ON	Z	Z	Z	ON	ON	Z
X	X	0	1	1	0	0	1	0	0	Z	Z	Z	ON	ON	Z	Z	ON	Z	Z	Z
X	X	0	1	1	0	0	1	0	1	Z	Z	Z	ON	ON	Z	Z	ON	Z	ON	Z
X	X	0	1	1	0	0	1	1	0	Z	Z	Z	ON	ON	Z	Z	ON	ON	Z	Z
X	X	0	1	1	0	0	1	1	1	Z	Z	Z	ON	ON	Z	Z	ON	ON	ON	Z
X	X	0	1	1	0	1	0	0	0	Z	Z	Z	ON	ON	Z	ON	Z	Z	Z	Z
X	X	0	1	1	0	1	0	0	1	Z	Z	Z	ON	ON	Z	ON	Z	Z	ON	Z
X	X	0	1	1	0	1	0	1	0	Z	Z	Z	ON	ON	Z	ON	Z	ON	Z	Z
X	X	0	1	1	0	1	0	1	1	Z	Z	Z	ON	ON	Z	ON	Z	ON	ON	Z
X	X	0	1	1	0	1	1	0	0	Z	Z	Z	ON	ON	Z	ON	ON	Z	Z	Z
X	X	0	1	1	0	1	1	0	1	Z	Z	Z	ON	ON	Z	ON	ON	Z	ON	Z
X	X	0	1	1	0	1	1	1	0	Z	Z	Z	ON	ON	Z	ON	ON	ON	Z	Z
X	X	0	1	1	0	1	1	1	1	Z	Z	Z	ON	ON	Z	ON	ON	ON	ON	Z
X	X	0	1	1	1	0	0	0	0	Z	Z	Z	ON	ON	ON	Z	Z	Z	Z	Z
X	X	0	1	1	1	0	0	0	1	Z	Z	Z	ON	ON	ON	Z	Z	Z	ON	Z
X	X	0	1	1	1	0	0	1	0	Z	Z	Z	ON	ON	ON	Z	Z	ON	Z	Z
X	X	0	1	1	1	0	0	1	1	Z	Z	Z	ON	ON	ON	Z	Z	ON	ON	Z
X	X	0	1	1	1	0	1	0	0	Z	Z	Z	ON	ON	ON	Z	ON	Z	Z	Z
X	X	0	1	1	1	0	1	0	1	Z	Z	Z	ON	ON	ON	Z	ON	Z	ON	Z
X	X	0	1	1	1	0	1	1	0	Z	Z	Z	ON	ON	ON	Z	ON	ON	Z	Z
X	X	0	1	1	1	0	1	1	1	Z	Z	Z	ON	ON	ON	Z	ON	ON	ON	Z
X	X	0	1	1	1	1	0	0	0	Z	Z	Z	ON	ON	ON	ON	Z	Z	Z	Z
X	X	0	1	1	1	1	0	0	1	Z	Z	Z	ON	ON	ON	ON	Z	Z	ON	Z
X	X	0	1	1	1	1	0	1	0	Z	Z	Z	ON	ON	ON	ON	Z	ON	Z	Z

Table 15-2. Channel Select Bus Combinations

Analog Multiplex Registers (AMUX1 and AMUX2)										Channel Select Bus Connected to:										
VREF	IBREF	MUX7	MUX6	MUX5	MUX4	MUX3	MUX2	MUX1	MUX0	V_{DD}	V_{IB}	AN7	AN6	AN5	AN4	AN3	AN2	AN1	AN0	V_{SS}
X	X	0	1	1	1	1	0	1	1	Z	Z	Z	ON	ON	ON	ON	Z	ON	ON	Z
X	X	0	1	1	1	1	1	0	0	Z	Z	Z	ON	ON	ON	ON	ON	Z	Z	Z
X	X	0	1	1	1	1	1	0	1	Z	Z	Z	ON	ON	ON	ON	ON	Z	ON	Z
X	X	0	1	1	1	1	1	1	0	Z	Z	Z	ON	ON	ON	ON	ON	ON	Z	Z
X	X	0	1	1	1	1	1	1	1	Z	Z	Z	ON	ON	ON	ON	ON	ON	ON	Z
X	X	1	0	0	0	0	0	0	0	Z	Z	ON	Z	Z	Z	Z	Z	Z	Z	Z
X	X	1	0	0	0	0	0	0	1	Z	Z	ON	Z	Z	Z	Z	Z	Z	ON	Z
X	X	1	0	0	0	0	0	1	0	Z	Z	ON	Z	Z	Z	Z	Z	ON	Z	Z
X	X	1	0	0	0	0	0	1	1	Z	Z	ON	Z	Z	Z	Z	Z	ON	ON	Z
X	X	1	0	0	0	0	1	0	0	Z	Z	ON	Z	Z	Z	Z	ON	Z	Z	Z
X	X	1	0	0	0	0	1	0	1	Z	Z	ON	Z	Z	Z	Z	ON	Z	ON	Z
X	X	1	0	0	0	0	1	1	0	Z	Z	ON	Z	Z	Z	Z	ON	ON	Z	Z
X	X	1	0	0	0	0	1	1	1	Z	Z	ON	Z	Z	Z	Z	ON	ON	ON	Z
X	X	1	0	0	0	1	0	0	0	Z	Z	ON	Z	Z	Z	ON	Z	Z	Z	Z
X	X	1	0	0	0	1	0	0	1	Z	Z	ON	Z	Z	Z	ON	Z	Z	ON	Z
X	X	1	0	0	0	1	0	1	0	Z	Z	ON	Z	Z	Z	ON	Z	ON	Z	Z
X	X	1	0	0	0	1	0	1	1	Z	Z	ON	Z	Z	Z	ON	Z	ON	ON	Z
X	X	1	0	0	0	1	1	0	0	Z	Z	ON	Z	Z	Z	ON	ON	Z	Z	Z
X	X	1	0	0	0	1	1	0	1	Z	Z	ON	Z	Z	Z	ON	ON	Z	ON	Z
X	X	1	0	0	0	1	1	1	0	Z	Z	ON	Z	Z	Z	ON	ON	ON	Z	Z
X	X	1	0	0	0	1	1	1	1	Z	Z	ON	Z	Z	Z	ON	ON	ON	ON	Z
X	X	1	0	0	1	0	0	0	0	Z	Z	ON	Z	Z	ON	Z	Z	Z	Z	Z
X	X	1	0	0	1	0	0	0	1	Z	Z	ON	Z	Z	ON	Z	Z	Z	ON	Z
X	X	1	0	0	1	0	0	1	0	Z	Z	ON	Z	Z	ON	Z	Z	ON	Z	Z
X	X	1	0	0	1	0	0	1	1	Z	Z	ON	Z	Z	ON	Z	Z	ON	ON	Z
X	X	1	0	0	1	0	1	0	0	Z	Z	ON	Z	Z	ON	Z	ON	Z	Z	Z
X	X	1	0	0	1	0	1	0	1	Z	Z	ON	Z	Z	ON	Z	ON	Z	ON	Z
X	X	1	0	0	1	0	1	1	0	Z	Z	ON	Z	Z	ON	Z	ON	ON	Z	Z
X	X	1	0	0	1	0	1	1	1	Z	Z	ON	Z	Z	ON	Z	ON	ON	ON	Z
X	X	1	0	0	1	1	0	0	0	Z	Z	ON	Z	Z	ON	ON	Z	Z	Z	Z
X	X	1	0	0	1	1	0	0	1	Z	Z	ON	Z	Z	ON	ON	Z	Z	ON	Z

Table 15-2. Channel Select Bus Combinations

Analog Multiplex Registers (AMUX1 and AMUX2)										Channel Select Bus Connected to:										
VREF	IBREF	MUX7	MUX6	MUX5	MUX4	MUX3	MUX2	MUX1	MUX0	V_{DD}	V_{IB}	AN7	AN6	AN5	AN4	AN3	AN2	AN1	AN0	V_{SS}
X	X	1	0	0	1	1	0	1	0	Z	Z	ON	Z	Z	ON	ON	Z	ON	Z	Z
X	X	1	0	0	1	1	0	1	1	Z	Z	ON	Z	Z	ON	ON	Z	ON	ON	Z
X	X	1	0	0	1	1	1	0	0	Z	Z	ON	Z	Z	ON	ON	ON	Z	Z	Z
X	X	1	0	0	1	1	1	0	1	Z	Z	ON	Z	Z	ON	ON	ON	Z	ON	Z
X	X	1	0	0	1	1	1	1	0	Z	Z	ON	Z	Z	ON	ON	ON	ON	Z	Z
X	X	1	0	0	1	1	1	1	1	Z	Z	ON	Z	Z	ON	ON	ON	ON	ON	Z
X	X	1	0	1	0	0	0	0	0	Z	Z	ON	Z	ON	Z	Z	Z	Z	Z	Z
X	X	1	0	1	0	0	0	0	1	Z	Z	ON	Z	ON	Z	Z	Z	Z	ON	Z
X	X	1	0	1	0	0	0	1	0	Z	Z	ON	Z	ON	Z	Z	Z	ON	Z	Z
X	X	1	0	1	0	0	0	1	1	Z	Z	ON	Z	ON	Z	Z	Z	ON	ON	Z
X	X	1	0	1	0	0	1	0	0	Z	Z	ON	Z	ON	Z	Z	ON	Z	Z	Z
X	X	1	0	1	0	0	1	0	1	Z	Z	ON	Z	ON	Z	Z	ON	Z	ON	Z
X	X	1	0	1	0	0	1	1	0	Z	Z	ON	Z	ON	Z	Z	ON	ON	Z	Z
X	X	1	0	1	0	0	1	1	1	Z	Z	ON	Z	ON	Z	Z	ON	ON	ON	Z
X	X	1	0	1	0	1	0	0	0	Z	Z	ON	Z	ON	Z	ON	Z	Z	Z	Z
X	X	1	0	1	0	1	0	0	1	Z	Z	ON	Z	ON	Z	ON	Z	Z	ON	Z
X	X	1	0	1	0	1	0	1	0	Z	Z	ON	Z	ON	Z	ON	Z	ON	Z	Z
X	X	1	0	1	0	1	0	1	1	Z	Z	ON	Z	ON	Z	ON	Z	ON	ON	Z
X	X	1	0	1	0	1	1	0	0	Z	Z	ON	Z	ON	Z	ON	ON	Z	Z	Z
X	X	1	0	1	0	1	1	0	1	Z	Z	ON	Z	ON	Z	ON	ON	Z	ON	Z
X	X	1	0	1	0	1	1	1	0	Z	Z	ON	Z	ON	Z	ON	ON	ON	Z	Z
X	X	1	0	1	0	1	1	1	1	Z	Z	ON	Z	ON	Z	ON	ON	ON	ON	Z
X	X	1	0	1	1	0	0	0	0	Z	Z	ON	Z	ON	ON	Z	Z	Z	Z	Z
X	X	1	0	1	1	0	0	0	1	Z	Z	ON	Z	ON	ON	Z	Z	Z	ON	Z
X	X	1	0	1	1	0	0	1	0	Z	Z	ON	Z	ON	ON	Z	Z	ON	Z	Z
X	X	1	0	1	1	0	0	1	1	Z	Z	ON	Z	ON	ON	Z	Z	ON	ON	Z
X	X	1	0	1	1	0	1	0	0	Z	Z	ON	Z	ON	ON	Z	ON	Z	Z	Z
X	X	1	0	1	1	0	1	0	1	Z	Z	ON	Z	ON	ON	Z	ON	Z	ON	Z
X	X	1	0	1	1	0	1	1	0	Z	Z	ON	Z	ON	ON	Z	ON	ON	Z	Z
X	X	1	0	1	1	0	1	1	1	Z	Z	ON	Z	ON	ON	Z	ON	ON	ON	Z
X	X	1	0	1	1	1	0	0	0	Z	Z	ON	Z	ON	ON	ON	Z	Z	Z	Z

Table 15-2. Channel Select Bus Combinations

Analog Multiplex Registers (AMUX1 and AMUX2)										Channel Select Bus Connected to:										
VREF	IBREF	MUX7	MUX6	MUX5	MUX4	MUX3	MUX2	MUX1	MUX0	V_{DD}	V_{IB}	AN7	AN6	AN5	AN4	AN3	AN2	AN1	AN0	V_{SS}
X	X	1	0	1	1	1	0	0	1	Z	Z	ON	Z	ON	ON	ON	Z	Z	ON	Z
X	X	1	0	1	1	1	0	1	0	Z	Z	ON	Z	ON	ON	ON	Z	ON	Z	Z
X	X	1	0	1	1	1	0	1	1	Z	Z	ON	Z	ON	ON	ON	Z	ON	ON	Z
X	X	1	0	1	1	1	1	0	0	Z	Z	ON	Z	ON	ON	ON	ON	Z	Z	Z
X	X	1	0	1	1	1	1	0	1	Z	Z	ON	Z	ON	ON	ON	ON	Z	ON	Z
X	X	1	0	1	1	1	1	1	0	Z	Z	ON	Z	ON	ON	ON	ON	ON	Z	Z
X	X	1	0	1	1	1	1	1	1	Z	Z	ON	Z	ON	ON	ON	ON	ON	ON	Z
X	X	1	!	0	0	0	0	0	0	Z	Z	ON	ON	Z	Z	Z	Z	Z	Z	Z
X	X	1	1	0	0	0	0	0	1	Z	Z	ON	ON	Z	Z	Z	Z	Z	ON	Z
X	X	1	1	0	0	0	0	1	0	Z	Z	ON	ON	Z	Z	Z	Z	ON	Z	Z
X	X	1	1	0	0	0	0	1	1	Z	Z	ON	ON	Z	Z	Z	Z	ON	ON	Z
X	X	1	1	0	0	0	1	0	0	Z	Z	ON	ON	Z	Z	Z	ON	Z	Z	Z
X	X	1	1	0	0	0	1	0	1	Z	Z	ON	ON	Z	Z	Z	ON	Z	ON	Z
X	X	1	1	0	0	0	1	1	0	Z	Z	ON	ON	Z	Z	Z	ON	ON	Z	Z
X	X	1	1	0	0	0	1	1	1	Z	Z	ON	ON	Z	Z	Z	ON	ON	ON	Z
X	X	1	1	0	0	1	0	0	0	Z	Z	ON	ON	Z	Z	ON	Z	Z	Z	Z
X	X	1	1	0	0	1	0	0	1	Z	Z	ON	ON	Z	Z	ON	Z	Z	ON	Z
X	X	1	1	0	0	1	0	1	0	Z	Z	ON	ON	Z	Z	ON	Z	ON	Z	Z
X	X	1	1	0	0	1	0	1	1	Z	Z	ON	ON	Z	Z	ON	Z	ON	ON	Z
X	X	1	1	0	0	1	1	0	0	Z	Z	ON	ON	Z	Z	ON	ON	Z	Z	Z
X	X	1	1	0	0	1	1	0	1	Z	Z	ON	ON	Z	Z	ON	ON	Z	ON	Z
X	X	1	1	0	0	1	1	1	0	Z	Z	ON	ON	Z	Z	ON	ON	ON	Z	Z
X	X	1	1	0	0	1	1	1	1	Z	Z	ON	ON	Z	Z	ON	ON	ON	ON	Z
X	X	1	1	0	1	0	0	0	0	Z	Z	ON	ON	Z	ON	Z	Z	Z	Z	Z
X	X	1	1	0	1	0	0	0	1	Z	Z	ON	ON	Z	ON	Z	Z	Z	ON	Z
X	X	1	1	0	1	0	0	1	0	Z	Z	ON	ON	Z	ON	Z	Z	ON	Z	Z
X	X	1	1	0	1	0	0	1	1	Z	Z	ON	ON	Z	ON	Z	Z	ON	ON	Z
X	X	1	1	0	1	0	1	0	0	Z	Z	ON	ON	Z	ON	Z	ON	Z	Z	Z
X	X	1	1	0	1	0	1	0	1	Z	Z	ON	ON	Z	ON	Z	ON	Z	ON	Z
X	X	1	1	0	1	0	1	1	0	Z	Z	ON	ON	Z	ON	Z	ON	ON	Z	Z
X	X	1	1	0	1	0	1	1	1	Z	Z	ON	ON	Z	ON	Z	ON	ON	ON	Z

Table 15-2. Channel Select Bus Combinations

Analog Multiplex Registers (AMUX1 and AMUX2)										Channel Select Bus Connected to:										
VREF	IBREF	MUX7	MUX6	MUX5	MUX4	MUX3	MUX2	MUX1	MUX0	V_{DD}	V_{IB}	AN7	AN6	AN5	AN4	AN3	AN2	AN1	AN0	V_{SS}
X	X	1	1	0	1	1	0	0	0	Z	Z	ON	ON	Z	ON	ON	Z	Z	Z	Z
X	X	1	1	0	1	1	0	0	1	Z	Z	ON	ON	Z	ON	ON	Z	Z	ON	Z
X	X	1	1	0	1	1	0	1	0	Z	Z	ON	ON	Z	ON	ON	Z	ON	Z	Z
X	X	1	1	0	1	1	0	1	1	Z	Z	ON	ON	Z	ON	ON	Z	ON	ON	Z
X	X	1	1	0	1	1	1	0	0	Z	Z	ON	ON	Z	ON	ON	ON	Z	Z	Z
X	X	1	1	0	1	1	1	0	1	Z	Z	ON	ON	Z	ON	ON	ON	Z	ON	Z
X	X	1	1	0	1	1	1	1	0	Z	Z	ON	ON	Z	ON	ON	ON	ON	Z	Z
X	X	1	1	0	1	1	1	1	1	Z	Z	ON	ON	Z	ON	ON	ON	ON	ON	Z
X	X	1	1	1	0	0	0	0	0	Z	Z	ON	ON	ON	Z	Z	Z	Z	Z	Z
X	X	1	1	1	0	0	0	0	1	Z	Z	ON	ON	ON	Z	Z	Z	Z	ON	Z
X	X	1	1	1	0	0	0	1	0	Z	Z	ON	ON	ON	Z	Z	Z	ON	Z	Z
X	X	1	1	1	0	0	0	1	1	Z	Z	ON	ON	ON	Z	Z	Z	ON	ON	Z
X	X	1	1	1	0	0	1	0	0	Z	Z	ON	ON	ON	Z	Z	ON	Z	Z	Z
X	X	1	1	1	0	0	1	0	1	Z	Z	ON	ON	ON	Z	Z	ON	Z	ON	Z
X	X	1	1	1	0	0	1	1	0	Z	Z	ON	ON	ON	Z	Z	ON	ON	Z	Z
X	X	1	1	1	0	0	1	1	1	Z	Z	ON	ON	ON	Z	Z	ON	ON	ON	Z
X	X	1	1	1	0	1	0	0	0	Z	Z	ON	ON	ON	Z	ON	Z	Z	Z	Z
X	X	1	1	1	0	1	0	0	1	Z	Z	ON	ON	ON	Z	ON	Z	Z	ON	Z
X	X	1	1	1	0	1	0	1	0	Z	Z	ON	ON	ON	Z	ON	Z	ON	Z	Z
X	X	1	1	1	0	1	0	1	1	Z	Z	ON	ON	ON	Z	ON	Z	ON	ON	Z
X	X	1	1	1	0	1	1	0	0	Z	Z	ON	ON	ON	Z	ON	ON	Z	Z	Z
X	X	1	1	1	0	1	1	0	1	Z	Z	ON	ON	ON	Z	ON	ON	Z	ON	Z
X	X	1	1	1	0	1	1	1	0	Z	Z	ON	ON	ON	Z	ON	ON	ON	Z	Z
X	X	1	1	1	0	1	1	1	1	Z	Z	ON	ON	ON	Z	ON	ON	ON	ON	Z
X	X	1	1	1	1	0	0	0	0	Z	Z	ON	ON	ON	ON	Z	Z	Z	Z	Z
X	X	1	1	1	1	0	0	0	1	Z	Z	ON	ON	ON	ON	Z	Z	Z	ON	Z
X	X	1	1	1	1	0	0	1	0	Z	Z	ON	ON	ON	ON	Z	Z	ON	Z	Z
X	X	1	1	1	1	0	0	1	1	Z	Z	ON	ON	ON	ON	Z	Z	ON	ON	Z
X	X	1	1	1	1	0	1	0	0	Z	Z	ON	ON	ON	ON	Z	ON	Z	Z	Z
X	X	1	1	1	1	0	1	0	1	Z	Z	ON	ON	ON	ON	Z	ON	Z	ON	Z
X	X	1	1	1	1	0	1	1	0	Z	Z	ON	ON	ON	ON	Z	ON	ON	Z	Z

Table 15-2. Channel Select Bus Combinations

Analog Multiplex Registers (AMUX1 and AMUX2)										Channel Select Bus Connected to:										
VREF	IBREF	MUX7	MUX6	MUX5	MUX4	MUX3	MUX2	MUX1	MUX0	V_{DD}	V_{IB}	AN7	AN6	AN5	AN4	AN3	AN2	AN1	AN0	V_{SS}
X	X	1	1	1	1	0	1	1	1	Z	Z	ON	ON	ON	ON	Z	ON	ON	ON	Z
X	X	1	1	1	1	1	0	0	0	Z	Z	ON	ON	ON	ON	ON	Z	Z	Z	Z
X	X	1	1	1	1	1	0	0	1	Z	Z	ON	ON	ON	ON	ON	Z	Z	ON	Z
X	X	1	1	1	1	1	0	1	0	Z	Z	ON	ON	ON	ON	ON	Z	ON	Z	Z
X	X	1	1	1	1	1	0	1	1	Z	Z	ON	ON	ON	ON	ON	Z	ON	ON	Z
X	X	1	1	1	1	1	1	0	0	Z	Z	ON	ON	ON	ON	ON	ON	Z	Z	Z
X	X	1	1	1	1	1	1	0	1	Z	Z	ON	ON	ON	ON	ON	ON	Z	ON	Z
X	X	1	1	1	1	1	1	1	0	Z	Z	ON	ON	ON	ON	ON	ON	ON	Z	Z
X	X	1	1	1	1	1	1	1	1	Z	Z	ON	ON	ON	ON	ON	ON	ON	ON	Z
0	1	0	0	0	0	0	0	0	0	Z	ON	Z	Z	Z	Z	Z	Z	Z	Z	Z
1	0	0	0	0	0	0	0	0	0	ON	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z

X = Don't Care

Z = High Impedance

15.2 ANALOG CONTROL REGISTER

The Analog Control Register (ACR) controls the power up, interrupt and flag operation. The analog subsystem draws about 470 μA of current while it is operating. The resulting power consumption can be reduced by powering down the analog subsystem when not in use. This can be done by clearing two enable bits (ISEN and CPEN) in the ACR at $001D. Since these bits are cleared following a reset, the voltage comparator and the charge current source will be powered down following a reset of the device.

The control bits in the ACR are shown in **Figure 15-2**. All the bits in this register are cleared by a reset of the device.

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
ACR	R	CHG	ATD2	ATD1	ICEN	CPIE	CPEN		ISEN
$001D	W								
reset:		0	0	0	0	0	0	0	0

Figure 15-5. Analog Control Register (ACR)

CHG

The CHG enable bit allows direct control of the charge current source and the discharge device; and also reflects the state of the discharge device. This bit is cleared if the ISEN enable bit is also cleared. This bit is cleared by a reset of the device.

1 = The charge current source is sourcing current out of the CAP pin. Writing a logical one enables the charging current out of the CAP pin, if the ISEN bit is also set.

0 = The discharge device is sinking current into the CAP pin. Writing a logical zero disables the charging current and enables the discharging current into the CAP pin.

ATD1:2

The ATD1:2 enable bits select one of the four operating modes used for making A/D conversions via the single-slope method.These four modes are given in **Table 15-3**. These bits have no effect if the ISEN enable bit is cleared. These bits are cleared by a reset of the device; and thereby returning the analog subsystem to the manual A/D conversion method.

Table 15-3. A/D Conversion Options

A/D Option Mode	Charge Control	A/D Options				Current Flow To/From CAP
		ISEN	ATD2	ATD1	CHG	
Disabled	Current Source and Discharge Disabled	0	X	X	X	Current control disabled, no source or sink current.
0	Manual Charge and Discharge	1	0	0	0	Begin sinking current when the CHG bit is cleared; and continue to sink current until the CHG bit is set.
		1	0	0	1	Begin sourcing current when the CHG bit is set; and continue to source current until the CHG bit is cleared.
1	Manual Charge and Automatic Discharge	1	0	1	0	Begin sinking current when the CHG bit is cleared; and continue to sink current until the CHG bit is set. (The CHG bit is cleared by writing a logical zero to it; or when the CPF flag bit is set.)
		1	0	1	1	Begin sourcing current when the CHG bit is set; and continue to source current until the CHG bit is cleared. (The CHG bit is cleared by writing a logical zero to it; or when the CPF flag bit is set.)
2	Automatic Charge and Discharge (TOF-ICF) Synchronized to Timer	1	1	0	0	The CHG bit remains cleared as long as current is being sunk. Begin sourcing current when the next Timer TOF occurs.
		1	1	0	1	The CHG bit remains set as long as current is being sourced. Begin sinking current when the next Timer ICF occurs.

Table 15-3. A/D Conversion Options

A/D Option Mode	Charge Control	A/D Options				Current Flow To/From CAP
		ISEN	ATD2	ATD1	CHG	
3	Automatic Charge and Discharge (OCF-ICF) Synchronized to Timer	1	1	1	0	The CHG bit remains cleared as long as current is being sunk. Begin sourcing current when the next Timer OCF occurs.
		1	1	1	1	The CHG bit remains set as long as current is being sourced. Begin sinking current when the next Timer ICF occurs.

ICEN

This is a read/write bit that enables a voltage comparison to trigger the input capture register of the programmable Timer when the CPF flag bit is set. Therefore an A/D conversion could be started by receiving an OCF or TOF from the programmable Timer; and then terminated when the voltage on the external ramping capacitor reaches the level of the unknown voltage. The time of termination will be stored in the 16-bit buffer located at $0014 and $0015. This bit is automatically set whenever Mode 2 or 3 is selected by setting the ATD2 control bit. This bit is cleared by a reset of the device.

1 = Connects the CPF flag bit to the Timer input capture register.
0 = Connects the PB1/TCAP pin to the Timer input capture register.

NOTE

When the ICEN bit is set the input capture function of the programmable Timer is not connected to the PB1/TCAP pin but is driven by the CPF output flag from the comparator. To return to capturing times from external events, the ICEN bit must first be cleared before the timed event occurs.

NOTE

The TCSEL bit in the Miscellaneous Control Register (bit 2 in $0B) must be cleared for ICEN control. TCSEL=1 will select the SCL signal from the SMBus as 16-bit Timer Input Capture source, irrespective of ICEN setting.

CPIE

This is a read/write bit that enables an analog interrupt when the CPF flag bits is set to a logical one. This bit is cleared by a reset of the device.

1 = Enables analog interrupt when comparator flag bit is set.
0 = Disables analog interrupt when comparator flag bit is set.

CPEN

The CPEN enable bit will power down voltage comparator in the analog subsystem. Powering down a comparator will drop the supply current by about 100μA. This bit is cleared by a reset of the device.

1 = Writing a logical one powers up voltage comparator.
0 = Writing a logical zero powers down voltage comparator

NOTE

The voltage comparator powers up slower than digital logic; and its output may go through indeterminate states which might set the CPF flag. It is therefore recommended to power up the charge current source first (ISEN); then to power up the comparator, and finally clear the bit by writing a logic one to the CPFR bit in the ACR.

ISEN

The ISEN enable bit will power down the charge current source and disable the discharge device in the analog subsystem. Powering down the current source will drop the supply current by about 200 μA. This bit is cleared by a reset of the device.

1 = Writing a logical one powers up the ramping current source and enables the discharge device on the CAP pin.
0 = Writing a logical zero powers down the ramping current source and disables the discharge device on the CAP pin.

NOTE

The analog subsystem has support circuitry which draws about 70μA of current. This current will be powered down if the comparator and the charge current source are powered down (ISEN and CPEN all cleared). Powering up the comparator or the charge current source will activate the support circuitry.

15.3 ANALOG STATUS REGISTER

The Analog Status Register (ASR) controls the interrupt and flag operation. The control bits in the ASR are shown in **Figure 15-2**. All the bits in this register are cleared by a reset of the device.

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
ASR	R	CPF		0	0	0	0	0	0
$001E	W			CPFR					
reset:		0	0	0	0	0	0	0	0

Figure 15-6. Analog Status Register

CPF

This read-only flag bit is set when the voltage on the positive input of comparator rises above the voltage on its negative input. This bit is reset by writing a logical one to the CPFR reset bit in the ASR. This bit is cleared by a reset of the device.

1 = The voltage on positive input of comparator was above the voltage on its negative input since CPF had been cleared.

0 = The voltage on positive input of comparator has not been above the voltage on its negative input since CPF had been cleared.

CPFR

Writing a logical one to this write-only flag clears the CPF flag in the ASR. Writing a logical zero to this bit has no effect. Reading the CPFR bit will return a logical zero. By default this bit looks cleared following a reset of the device.

1 = Clears the CPF flag bit.

0 = No effect.

NOTE

The CPFR bit should be written with a logical one following a power up of the comparator. This will clear out any latched CPF flag bit which might have been set during the slower power up sequence of the analog circuitry.

If both inputs to the comparator are above the maximum common-mode input voltage (V_{DD}−1.5V) the output of the comparator is indeterminate and may set the comparator flag. Applying a reset to the device may only temporarily clear this flag as long as both inputs of a comparator remain above the maximum common-mode input voltages.

15.4 A/D CONVERSION METHODS

The control bits in the ACR provide various options to charge or discharge current through the CAP pin in order to perform single-slope A/D conversions using an external capacitor from the CAP pin to V_{SS} as shown in **Figure 15-7**. The various A/D conversion triggering options are given in **Table 15-3**.

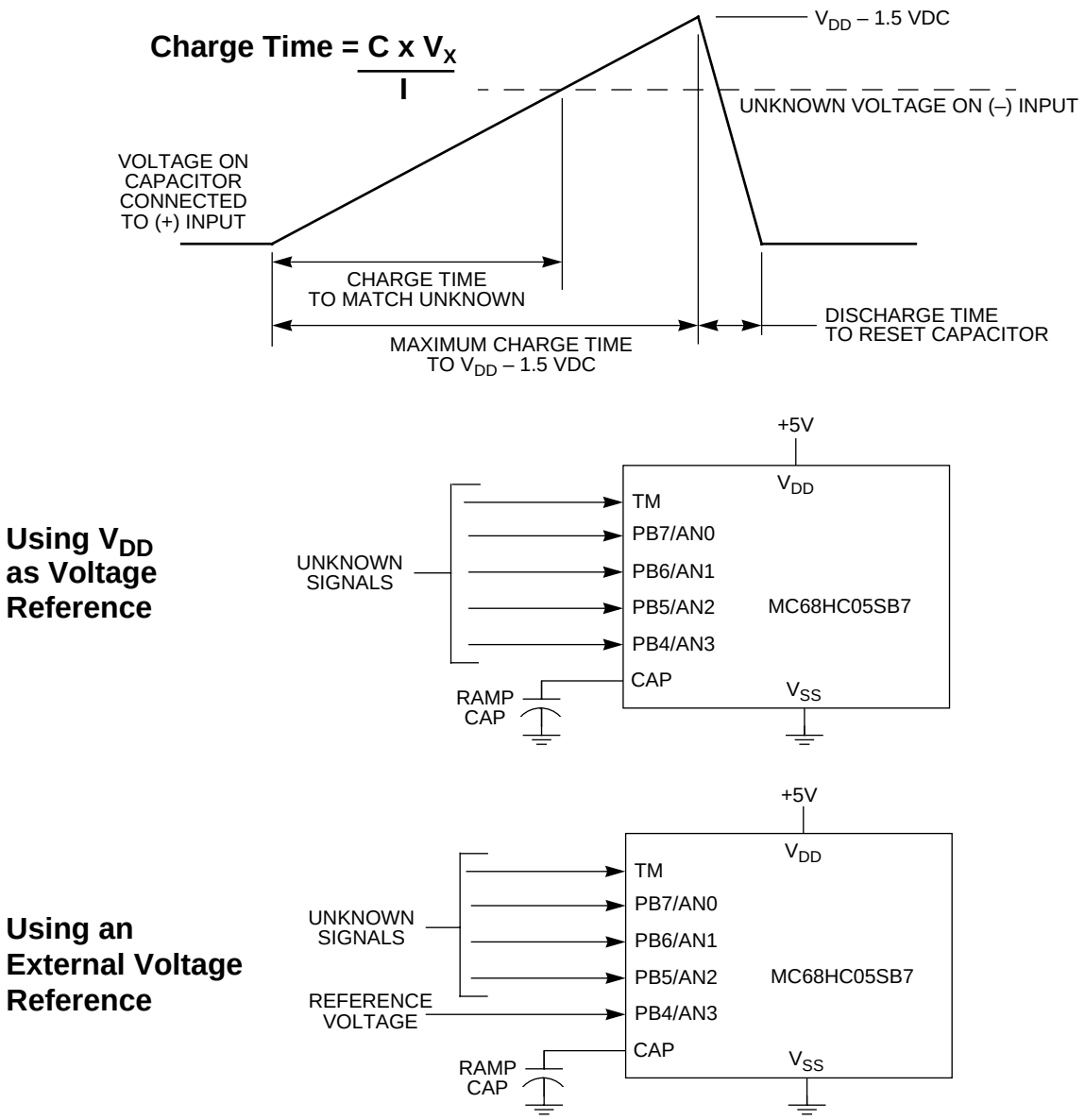


Figure 15-7. Single-Slope A/D Conversion Method

The top three bits of the ACR control the charging and discharging current into or out of the CAP pin. These three bits will have no affect on the CAP pin if the ISEN enable bit is cleared. Any clearing of the ISEN bit will immediately disable both the

charge current source and the discharge device. Since all these bits and the ISEN bit are cleared when the device is reset, the MC68HC05SB7 starts with the charge and discharge function disabled.

The length of time required to reach the maximum voltage to be measured will determine the resolution of the reading. The time to ramp the external capacitor voltage to match the maximum voltage is dependent on:

- Desired resolution.
- Clock rate for timing function.
- Any prescaling of the clock to the timing function.
- Charging current to external capacitor.
- Value of the external capacitor.

The values of each parameter are related by the general equation:

$$t_{CHG} = \frac{C \times V_{MAX}}{I_{CHG}}$$

Each parameter can also be expressed by the following equations:

$$t_{CHG} = \frac{P \times N}{f_{OSC}}$$

$$V_{MAX} = \frac{I_{CHG} \times P \times N}{C \times f_{OSC}}$$

$$N = \frac{V_{MAX} \times C \times f_{OSC}}{I_{CHG} \times P}$$

$$C = \frac{I_{CHG} \times N \times P}{V_{MAX} \times f_{OSC}}$$

where the signal names and parameters used are given in **Table 15-4**.

NOTE

Noise on the system ground or the external ramping capacitor can cause the comparator to trip prematurely. Therefore in any given application it is best to use the fastest possible ramp rate (shortest possible time).

NOTE

The value of any capacitor connected directly to the CAP pin should be limited to less than 2µF. Larger capacitances will create signal noise.

NOTE

Sufficient time should be allowed to discharge the external capacitor or subsequent charge times will be shortened with resultant errors in timing conversion.

NOTE

If the unknown voltage applied to the comparator is greater than its common-mode range (V_{DD}−1.5 volts) the external capacitor will try to charge to the same level. This will cause both comparator inputs to be above the common-mode range and the output will be indeterminate. All A/D conversion software methods should have a maximum time check to determine if this case is occurring.

Table 15-4. A/D Conversion Signals and Definitions

Name	Function	Conditions
I_{CHG}	Charging current on external ramping capacitor	I_{CHG} = 80 - 120 µA
I_{DIS}	Discharge current on external ramping capacitor	I_{DIS} > 1 mA
V_{CAP}	Voltage on external ramping capacitor	$V_{SS} < V_{CAP} < (V_{DD} – 1.5)$
V_X	Voltage of unknown on (–) input of voltage comparator	$V_{SS} < V_X < (V_{DD} –1.5)$
V_{MAX}	Maximum voltage on external ramping capacitor	$V_{MAX} = V_{DD} – 1.0$
t_{CHG}	TIme to charge external capacitor	Δt from V_{SS} to V_X
t_{DIS}	Time to discharge external capacitor	Δt from V_{MAX} to V_{SS}
C	Capacitance of external ramping capacitor	0.001 to 1.000 µF
N	Number of counts for I_{CHG} to charge C to V_X	0 to 65536
P	Prescaler into timing function	f_{OSC} ÷ loop time for software timing f_{OSC} ÷ 8 for Core Timer f_{OSC} ÷ 8 for Programmable TImer
f_{OSC}	Clock source frequency (excluding any prescaling)	0 to 4.2 MHz

Table 15-5 gives examples of voltage ranges, resolution, ramp times and capacitor sizes for various conversion methods.

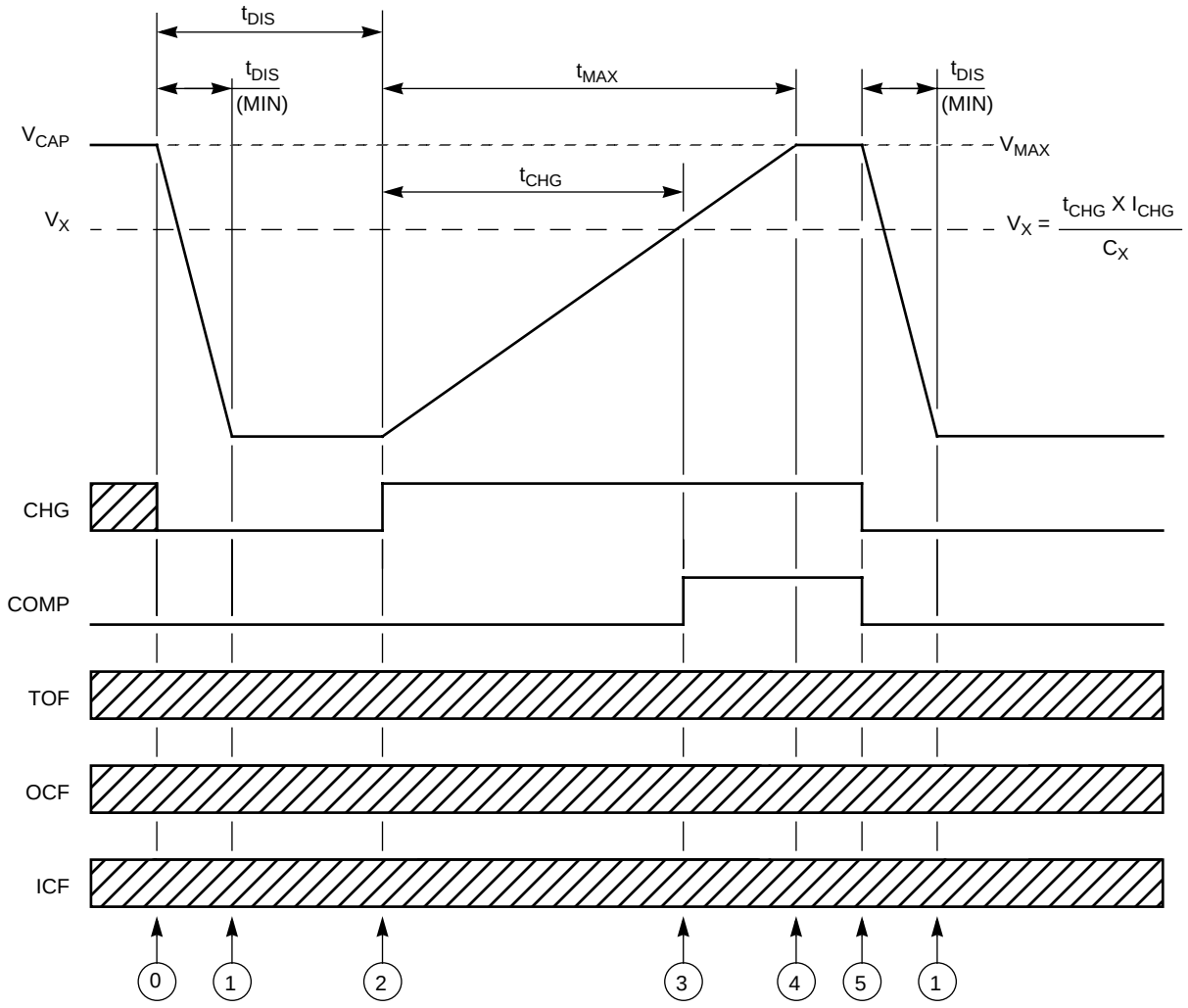
Table 15-5. Sample Conversion Timing

Bits	Counts	V_{MAX} (VDC)	A/D Method	Clock Source	f_{OSC} (MHz)	t_{CHG} (μs)	C (μF)
8	256	3.5	Software Loop (10 cycles) Mode 0 (manual)	Ext Pin Oscillator	2.0	2560	0.073
8	256	3.5	Programmable Timer, Mode 1 (TOF to ICF)	VCO	0.5	4096	0.117
				Ext Pin Oscillator	1.0	2048	0.059
					2.0	1024	0.029
					4.0	512	0.015
10	1024	3.5	Programmable Timer, Mode 1 (TOF to ICF)	VCO	0.5	16384	0.468
				Ext Pin Oscillator	1.0	8192	0.234
					2.0	4096	0.117
					4.0	2048	0.059
12	4096	3.5	Programmable Timer, Mode 1 (TOF to ICF)	VCO	0.5	65536	1.872
				Ext Pin Oscillator	1.0	32768	0.936
					2.0	16384	0.468
					4.0	8192	0.234

The general architecture of the MC68HC05SB7 and mode selection bits in the ACR allow four methods based on simple single-slope A/D conversion. Each of these methods is shown in the following figures:

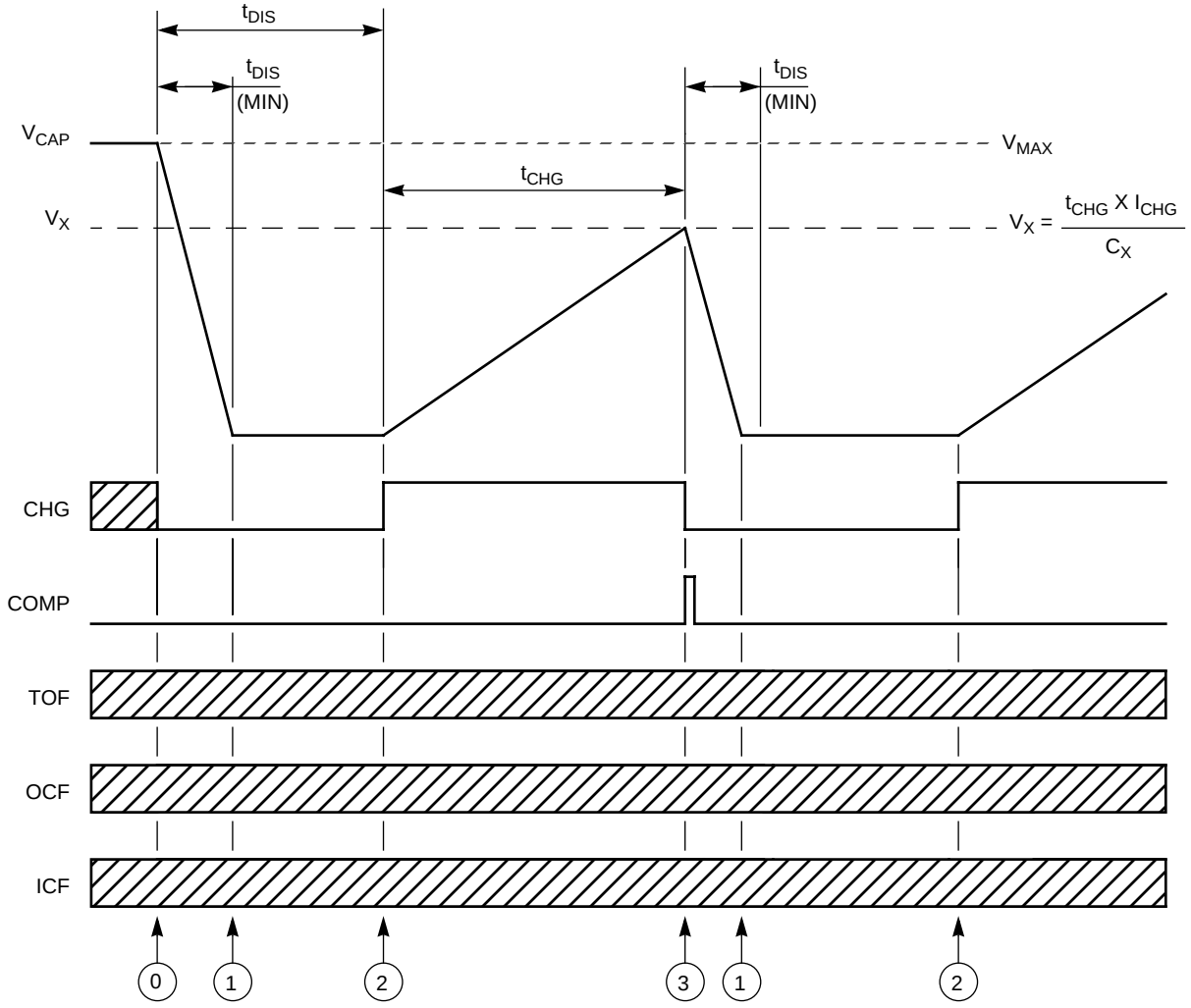
- Manual start and stop (Mode 0) **Figure 15-8**.
- Manual start and automatic discharge (Mode 1) **Figure 15-9**.
- Automatic start and stop from TOF to ICF (Mode 2) **Figure 15-10**.
- Automatic start and stop from OCF to ICF (Mode 3) **Figure 15-11**.

The description of the signals and parameters used in these figures are given in **Table 15-4**.



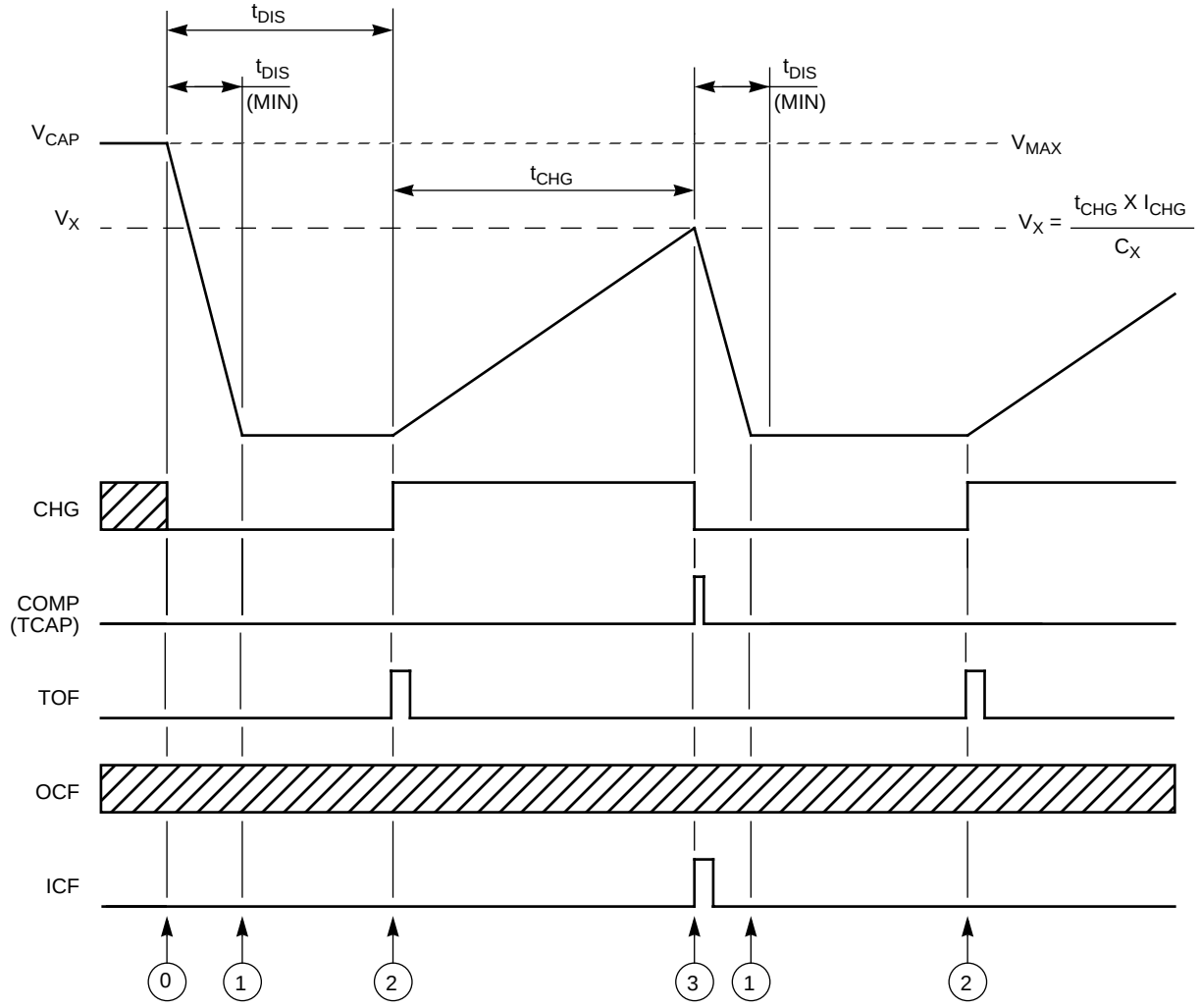
Point	Action	Software/Hardware Action	Dependent Variable(s)
0	Begin initial Discharge and select Mode 0 by clearing the CHG, ATD2 and ATD1 control bits in the ACR.	Software write.	Software.
1	V_{CAP} falls to V_{SS}.	Wait out minimum t_{DIS} time.	V_{MAX}, I_{DIS}, $C_{X.}$
2	Stop Discharge and begin Charge by setting CHG control bit in ACR.	Software write.	Software.
3	V_{CAP} rises to V_X and comparator output trips, setting CPF.	Wait out t_{CHG} time.	V_X, I_{CHG}, $C_{X.}$
4	V_{CAP} Reaches $V_{MAX.}$	Wait out t_{CHG} time.	V_{MAX}, I_{CHG}, $C_{X.}$
5	Begin next Discharge by clearing the CHG control bit in the ACR.	Software write.	Software.

Figure 15-8. A/D Conversion - Full Manual Control (Mode 0)



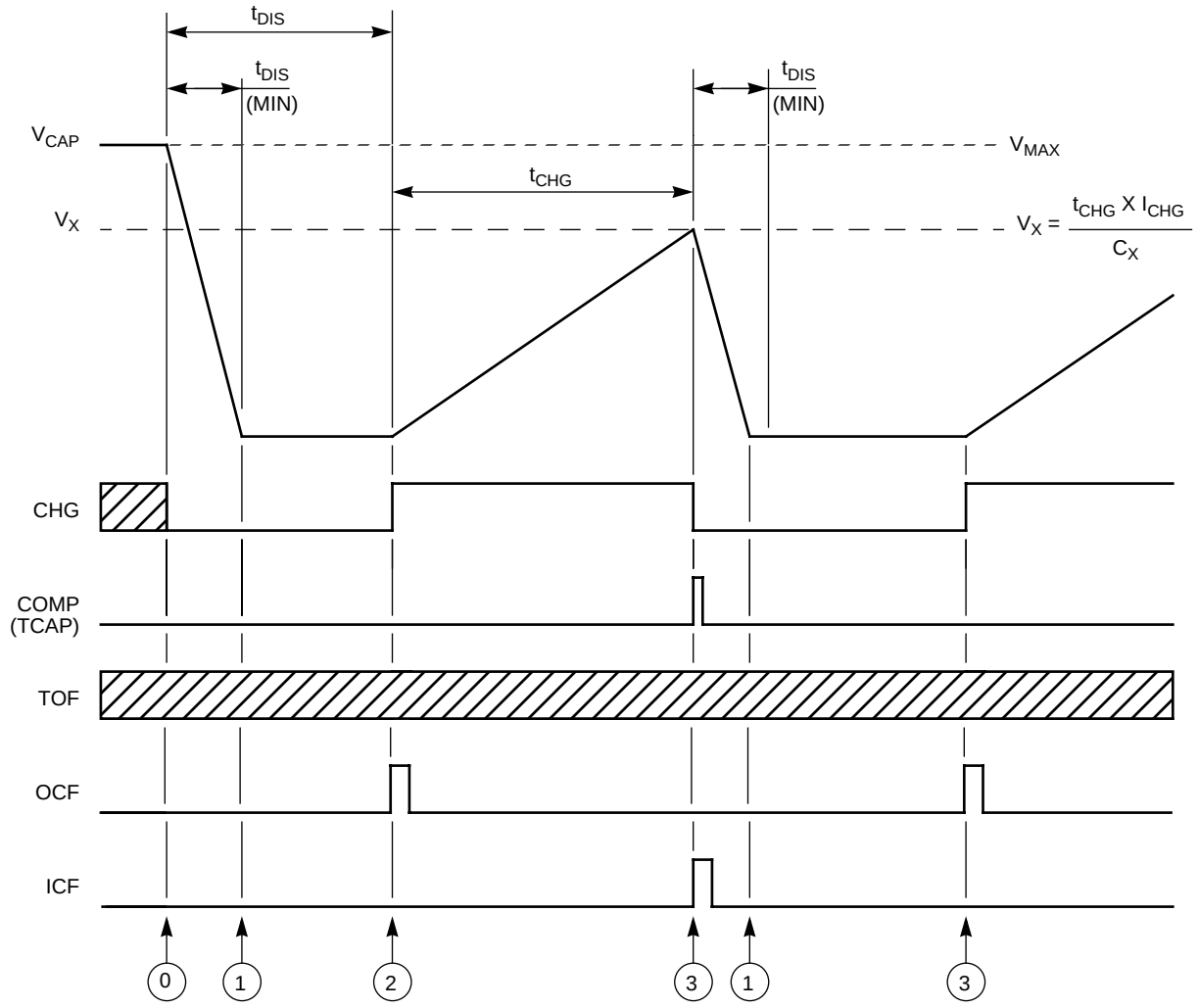
Point	Action	Software/Hardware Action	Dependent Variable(s)
0	Begin initial Discharge and select Mode 1 by clearing CHG and ATD2; and setting ATD1 in the ACR.	Software write.	Software.
1	V_{CAP} falls to V_{SS}.	Wait out minimum t_{DIS} time.	V_{MAX}, I_{DIS}, $C_{X.}$
2	Stop Discharge and begin Charge by setting CHG control bit in ACR.	Software write.	Software.
3	V_{CAP} rises to V_X and comparator output trips, setting CPF which clears CHG control bit in the ACR.	Wait out t_{CHG} time. CPF clears CHG control bit.	V_X, I_{CHG}, $C_{X.}$

Figure 15-9. A/D Conversion - Manual/Auto Discharge Control (Mode 1)



Point	Action	Software/Hardware Action	Dependent Variable(s)
0	Begin initial Discharge and select Mode 2 by clearing CHG and ATD1 and setting ATD2 in the ACR.	Software write. (ICEN bit also set)	Software.
1	V_{CAP} falls to V_{SS}.	Wait out minimum t_{DIS} time.	V_{MAX}, I_{DIS}, $C_{X.}$
2	Stop Discharge and begin Charge when the next TOF sets the CHG control bit in ACR.	Timer TOF sets the CHG control bit in the ACR.	Free-running Timer counter overflow, f_{OSC}, P.
3	V_{CAP} rises to V_X and comparator output trips, setting CPF which causes an ICF from the Timer and clears the CHG control bit in ACR.	Wait out t_{CHG} time. Timer ICF clears the CHG control bit in the ACR.	V_X, I_{CHG}, $C_{X.}$

Figure 15-10. A/D Conversion - TOF/ICF Control (Mode 2)



Point	Action	Software/Hardware Action	Dependent Variable(s)
0	Begin initial Discharge and select Mode 3 by clearing CHG and setting ATD2 and ATD1 in the ACR.	Software write. (ICEN bit also set)	Software.
1	V_{CAP} falls to V_{SS}. Set Timer output compare registers (OCRU, OCRL) to desired charge start time.	Wait out minimum t_{DIS} time. Software write to OCRH, OCRL.	V_{MAX}, I_{DIS}, C_X, Software.
2	Stop Discharge and begin Charge when the next OCF sets the CHG control bit in ACR.	Timer OCF sets the CHG control bit in the ACR.	Free-running Timer counter overflow, f_{OSC}, P.
3	V_{CAP} rises to V_X and comparator output trips, setting CPF which causes an ICF from the Timer and clears the CHG control bit in ACR.	Wait out t_{CHG} time. Timer ICF clears the CHG control bit in the ACR.	V_X, I_{CHG}, C_X.

Figure 15-11. A/D Conversion - OCF/ICF Control (Mode 3)

15.5 VOLTAGE MEASUREMENT METHODS

The various methods for obtaining a voltage measurement can use software techniques to express these voltages as absolute or ratiometric readings.

NOTE

All A/D conversion methods should include a test for a maximum elapsed time in order to detect error cases where the inputs may be outside of the design specification.

15.5.1 Absolute Voltage Readings

The absolute value of a voltage measurement can be calculated in software by first taking a reference reading from a fixed source and then comparing subsequent unknown voltages to that reading as a percentage of the reference voltage multiplied times the known reference value.

The accuracy of absolute readings will depend on the error sources taken into account using the features of the analog subsystem and appropriate software as described in **Table 15-6**. As can be seen from this table, most of the errors can be reduced by frequent comparisons to a known voltage, use of the inverted comparator inputs, and averaging of multiple samples.

Table 15-6. Absolute Voltage Reading Errors

Error Source	Accuracy Improvements Possible	
	In Hardware	In Software
Change in Reference Voltage	Provide closer tolerance reference	Calibration and storage of reference source over temperature and supply voltage
Change in Magnitude of Ramp Current Source	Not adjustable	Compare unknown with recent measurement from reference
Non-Linearity of Ramp Current Source vs. Voltage	Not adjustable	Calibration and storage of voltages at 1/4, 1/2, 3/4 and FS
Change in Magnitude of Ramp Capacitor	Provide closer tolerance ramp capacitor	Compare unknown with recent measurement from reference
Frequency Shift in Internal Low-Power Oscillator	Use external oscillator with crystal	Compare unknown with recent measurement from reference
Frequency Shift in External Oscillator	Provide closer tolerance crystal	Compare unknown with recent measurement from reference
Sampling Capacitor Leakage	Use faster conversion times	Compare unknown with recent measurement from reference
Internal Voltage Divider Ratio	Not adjustable	Compare unknown with recent measurement from reference OR avoid use of divided input
Noise internal to MCU	Close decoupling at V_{DD} and V_{SS} pins and reduce supply source impedance	Average multiple readings on both the reference and the unknown voltage

Table 15-6. Absolute Voltage Reading Errors

Error Source	Accuracy Improvements Possible	
	In Hardware	In Software
Noise external to MCU	Close decoupling of power supply, low source impedances, good board layout, use of multi-layer board	Average multiple readings on both the reference and the unknown voltage

Internal Absolute Reference

If a stable source of V_{DD} is provided, the reference measurement point can be internally selected. In this case the reference reading can be taken by setting the VREF bit and clearing the MUX7:0 bits in the AMUX register. This connects the channel selection bus to the V_{DD} pin.

Alternatively, the internal bandgap voltage can be used as the reference measurement point, by setting the IBREF bit in AMUX2 Register and TSEN bit in the Miscellaneous Control Register.

External Absolute Reference

If a stable external source is provided, the reference measurement point can be any one of the channel selected pins from PB4 through PB7. In this case the reference reading can be taken by setting the MUX bit in the AMUX which connects channel selection bus to the pin connected to the external reference source.

15.5.2 Ratiometric Voltage Readings

The ratiometric value of a voltage measurement can be calculated in software by first taking a reference reading from a reference source and then comparing subsequent unknown voltages to that reading as a percentage of the reference value. The accuracy of ratiometric readings will depend on variety of sources, but will generally be better than for absolute readings. Many of these error sources can be taken into account using the features of the analog subsystem and appropriate software as described in **Table 15-7**. As with absolute measurements most of the errors can be reduced by frequent comparisons to the reference voltage, use of the inverted comparator inputs, and averaging of multiple samples.

Internal Ratiometric Reference

If readings are to be ratiometric to V_{DD}, the reference measurement point can be internally selected. In this case the reference reading can be taken by setting the VREF bit and clearing the MUX7:0 bits in the AMUX register which connects the channel selection bus to the V_{DD} pin.

Alternatively, the internal bandgap voltage can be used as the reference measurement point, by setting the IBREF bit in AMUX2 Register and TSEN bit in the Miscellaneous Control Register.

External Ratiometric Reference

If readings are to be ratiometric to some external source, the reference measurement point can be connected to any one of the channel selected pins from PB4 through PB7. In this case the reference reading can be taken by setting the MUX bit in the AMUX which connects channel selection bus to the pin connected to the external reference source.

Table 15-7. Ratiometric Voltage Reading Errors

Error Source	Accuracy Improvements Possible	
	In Hardware	In Software
Change in Reference Voltage	Not required for ratiometric	Compare unknown with recent measurement from reference
Change in Magnitude of Ramp Current Source	Not required for ratiometric	Compare unknown with recent measurement from reference
Non-Linearity of Ramp Current Source vs. Voltage	Not adjustable	Calibration and storage of voltages at 1/4, 1/2, 3/4 and FS
Change in Magnitude of Ramp Capacitor	Not required for ratiometric	Compare unknown with recent measurement from reference
Frequency Shift in Internal Low-Power Oscillator	Not required for ratiometric	Compare unknown with recent measurement from reference
Frequency Shift in External Oscillator	Not required for ratiometric	Compare unknown with recent measurement from reference
Sampling Capacitor Leakage	Use faster conversion times	Compare unknown with recent measurement from reference
Internal Voltage Divider Ratio	Not required for ratiometric	Compare unknown with recent measurement from reference
Noise internal to MCU	Close decoupling at V_{DD} and V_{SS} pins and reduce supply source impedance	Average multiple readings on both the reference and the unknown voltage
Noise external to MCU	Close decoupling of power supply, low source impedances, good board layout, use of multi-layer board	Average multiple readings on both the reference and the unknown voltage

15.6 VOLTAGE COMPARATOR FEATURES

The internal comparator can also be used as simple voltage comparator.

Voltage Comparator

Voltage comparator can be used as a simple comparator if its charge current source and discharge device are disabled by clearing the ISEN bit in the ACR. If ISEN bit is set the internal ramp discharge device connected to CAP may become active and try to pulldown any voltage source that may be connected to that pin. Also, since voltage comparator is always connected to two of the Port B I/O pins, these pins should be configured as inputs and have their software programmable pulldowns disabled. The required setup to use voltage comparator as a simple comparator are shown in **Table 15-8**.

Table 15-8. Voltage Comparator Setup Conditions

Current Source Enable	Discharge Device Disable	Port B Pin as Inputs	Prog. Timer Input Capture Source
ISEN = 0	ISEN = 0	DDRB4 = 0 DDRB5 = 0	ICEN = 0

15.7 CURRENT SOURCE FEATURES

The internal current source connected to the CAP pin supplies about 100 μA of current when the ramp discharge device is disabled and the current source is active. Therefore this current source can be used in an application if the ISEN enable bit is set to power up the current source is enabled by setting the A/D conversion method to manual Mode 0 (ATD1 and ATD2 cleared) and the charge current enabled (CHG set).

15.8 SAMPLE AND HOLD

When using the internal sample capacitor to capture a voltage for later conversion, the HOLD and DHOLD bit must be cleared first before changing any channel selection. If both the HOLD (or DHOLD) bit and the channel selection are changed on the same write cycle, the sample may be corrupted during the switching transitions.

NOTE

The sample capacitor can be affected by excessive noise created with respect to the device's V_{SS} pin such that it may appear to leak down or charge up depending on the voltage level stored on the sample capacitor. It is recommended to avoid switching large currents through the port pins while a voltage is to remain stored on the same capacitor.

The additional option of adding an offset voltage to the bottom of the sample capacitor allows unknown voltages near VSS to be sampled and then shifted up past the comparator offset and the device offset caused by a single V_{SS} return pin. The offset also provides a means to measure the internal V_{SS} level regardless of the comparator offset in order to determine N_{OFF} as described in **Section 15.5**.

15.9 PORT B INTERACTION WITH ANALOG INPUTS

The analog subsystem is connected directly to the Port B I/O pins without any intervening gates. It is therefore possible to measure the voltages on Port B pins set as inputs; or to have the analog voltage measurements corrupted by Port B pins set as outputs.

15.9.1 Port B Pins As Inputs

All the Port B pins will power up as inputs or return to inputs following a reset of the device since the bits in the Port B Data Direction register will be reset.

If any Port B pins are to be used for analog voltage measurements they should be left as inputs. In this case, not only can the voltage on the pin be measured, but the “logic” state of the Port B pins to be read from location $0002.

15.10 NOISE SENSITIVITY

In addition to the normal effects of electrical noise on the analog input signal there can also be other noise related effects caused by the digital-to-analog interface. Since there is only one V_{SS} return for both the digital and the analog subsystems on the device, currents in the digital section may affect the analog ground reference within the device. This can add voltage offsets to measured inputs or cause channel-to-channel crosstalk.

In order to reduce the impact of these effects, there should be no switching of heavy I/O currents to or from the device while there is a critical analog conversion or voltage comparison in process. Limiting switched I/O currents to 2 to 4 mA during these times is recommended.

A noise reduction benefit can be gained with 0.1µF bypass capacitors from each analog input (PB7:4) to the V_{SS} pin. Also, try to keep all the digital power supply or load currents from passing through any conductors which are the return paths for an analog signal.

SECTION 16
PERSONALITY EPROM

This section describes how to program the 64-bit personality EPROM (PEPROM). **Figure 16-1** shows the structure of the bit programmable PEPROM subsystem.

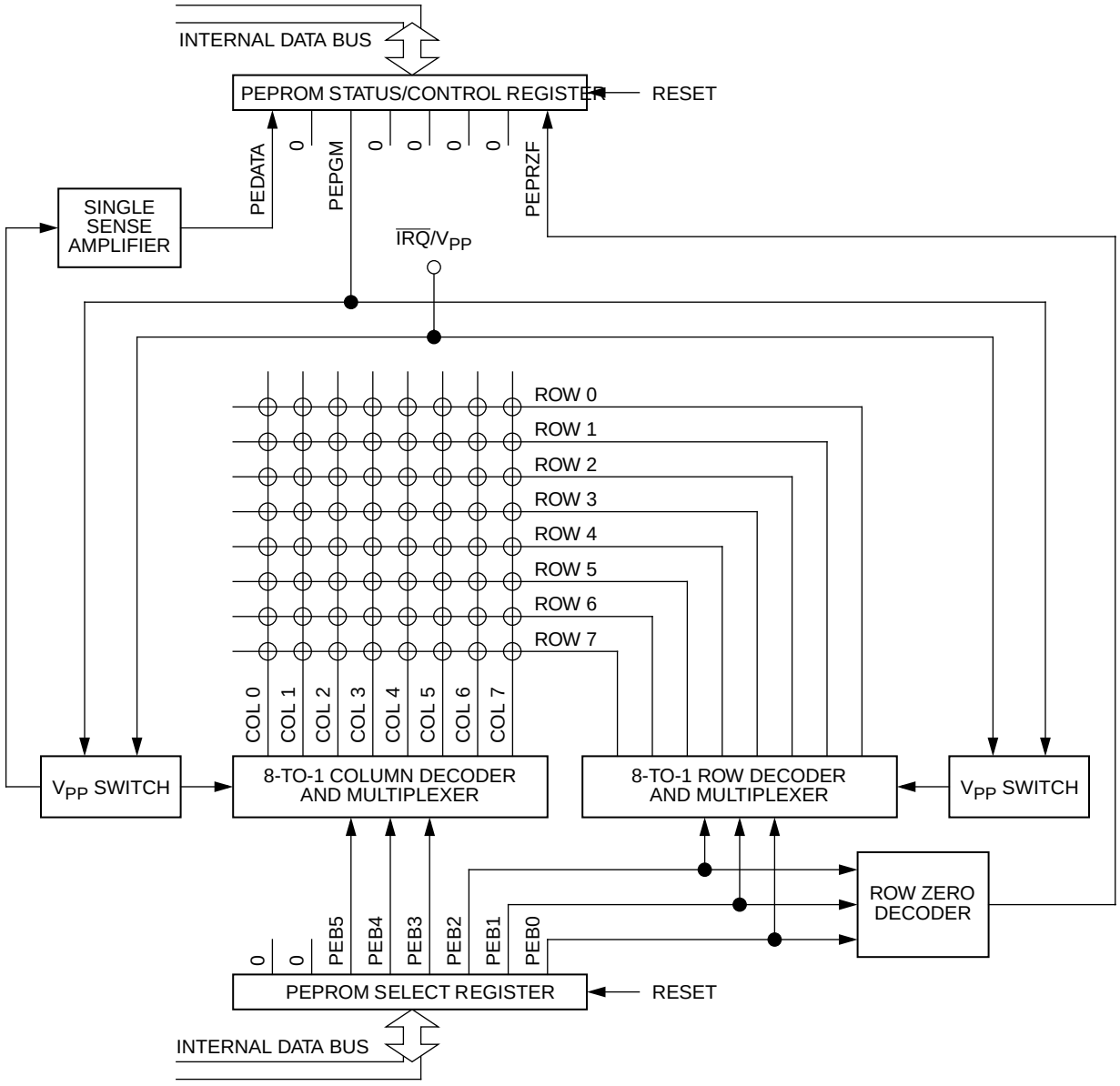


Figure 16-1. Personality EPROM

16.1 PEPROM REGISTERS

Two I/O registers control programming and reading of the PEPROM:

- The PEPROM bit select register (PEBSR).
- The PEPROM status and control register (PESCR).

16.1.1 PEPROM Bit Select Register (PEBSR)

The PEPROM bit select register (PEBSR) selects one of 64 bits in the PEPROM array. Reset clears all the bits in the PEPROM bit select register.

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
PEBSR	R	PEB7	PEB6	PEB5	PEB4	PEB3	PEB2	PEB1	PEB0
$000E	W								
reset:		0	0	0	0	0	0	0	0

Figure 16-2. PEPROM Bit Select Register (PEBSR)

PEB7 and PEB6 — Not connected to the PEPROM array

These read/write bits are available as storage locations. Reset clears PEB7 and PEB6.

PEB5–PEB0 — PEPROM Bit Select Bits

These read/write bits select one of 64 bits in the PEPROM as shown in **Table 16-1**. Bits PEB2–0 select the PEPROM row, and bits PEB5–3 select the PEPROM column. Reset clears PEB5–PEB0, selecting the PEPROM bit in row zero, column zero.

16.1.2 PEPROM Status and Control Register (PESCR)

The PEPROM status and control register (PESCR) controls the PEPROM programming voltage. This register also transfers the PEPROM bits to the internal data bus and contains a flag bit when row zero is selected.

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
PESCR	R	PEDATA	0	PEPGM	0	0	0	0	PEPRZF
$000F	W								
reset:		U	0	0	0	0	0	0	1

U = UNAFFECTED BY RESET

Figure 16-3. PEPROM Status and Control Register (PESCR)

PEDATA — PEPROM Data

This read-only bit is the state of the PEPROM sense amplifier and shows the state of the currently selected bit. Reset does not affect the PEDATA bit.

1 = PEPROM data is a logic one.
0 = PEPROM data is a logic zero.

PEPGM — PEPROM Program Control

This read/write bit controls the switches that apply the programming voltage on the $\overline{IRQ}/V_{PP}$ pin to the selected PEPROM cell. Reset clears the PEPGM bit.

1 = Programming voltage applied to array bit.
0 = Programming voltage not applied to array bit.

PEPRZF — PEPROM Row Zero Flag

This read-only bit is set when the PEPROM bit select register selects the first row (row zero) of the PEPROM array. Selecting any other row clears PEPRZF. Monitoring PEPRZF can reduce the code needed to access one byte of eight PEPROM locations. Reset clears the PEPROM bit select register thereby setting the PEPRZF bit by default.

1 = Row zero selected.
0 = Row zero not selected.

Table 16-1. PEPROM Bit Selection

PEBSR	PEPROM Bit Selected	
$00 - $07	Row 0 - Row 7	Column 0
$08 - $0F	Row 0 - Row 7	Column 1
$10 - $17	Row 0 - Row 7	Column 2
$18 - $1F	Row 0 - Row 7	Column 3
$20 - $27	Row 0 - Row 7	Column 4
$28 - $2F	Row 0 - Row 7	Column 5
$30 - $37	Row 0 - Row 7	Column 6
$38 - $3F	Row 0 - Row 7	Column 7

16.2 PEPROM PROGRAMMING

The PEPROM can be programmed by user software with the V_{PP} voltage level applied to the $\overline{IRQ}/V_{PP}$ pin. The following sequence shows how to program each PEPROM bit:

1. Select a PEPROM bit by writing to the PEBSR.
2. Set the PEPGM bit in the PESCR.
3. Wait 3 milliseconds.
4. Clear the PEPGM bit.
5. Move to next PEPROM bit to be programmed in step 1.

NOTE

While the PEPGM bit is set and the V_{PP} voltage level is applied to the $\overline{IRQ}/V_{PP}$ pin, do not access bits that are to be left unprogrammed (erased).

To program the PEPROM, V_{DD} must be greater than 4.5 Vdc.

16.3 PEPROM READING

The following sequence shows how to read the PEPROM:

1. Select a bit by writing to the PEBSR.
2. Read the PEDATA bit in the PESCR.
3. Store the PEDATA bit in RAM or in a register.
4. Select another bit by changing the PEBSR.
5. Continue reading and storing the PEDATA bits until the required personality EPROM data is retrieved and stored.

Reading the PEPROM is easiest when each PEPROM column contains one byte. Selecting a row 0 bit selects the first bit, and incrementing the PEPROM bit select register (PEBSR) selects the next bit in row 1 from the same column. Incrementing PEBSR seven more times selects the remaining bits of the column and ends up selecting the bit in row 0 of the next column, thereby setting the row 0 flag, PEPRZF.

NOTE

A PEPROM byte that has been read can be transferred to the personality EPROM bit select register (PEBSR) so that subsequent reads of the PEBSR quickly yield that PEPROM byte.

16.4 PEPROM ERASING

MCUs with windowed packages permit PEPROM erasure with ultraviolet light. Erase the PEPROM by exposing it to 15 Ws/cm^2 of ultraviolet light with a wavelength of 2537 angstroms. Position the ultraviolet light source 1 inch from the window. Do not use a shortwave filter. The erased state of a PEPROM bit is a logic zero.

16.5 PEPROM PREPROGRAMMED OPTIONS

The MC68HC05SB7 is available with a factory preprogrammed PEPROM. The following measured parameters are available:

- The internal VCO minimum frequency: [programmed or left blank]
- The internal VCO maximum frequency: [programmed or left blank]
- The internal bandgap reference voltage: [programmed or left blank]
- The internal temperature sensor voltage: [programmed or left blank]

Each parameter is stored as a 16-bit value in the PEPROM, as shown in the **Table 16-1**. Unprogrammed bits are blank, and are available for user programming.

Table 16-2. PEPROM Preprogrammed Option

PEBSR (LSB - MSB)	DATA
$00 - $0F	VCO Minimum Frequency (f_{VCOMIN})
$10 - $1F	VCO Maximum Frequency (f_{VCOMAX})
$20 - $2F	Internal Bandgap Voltage (V_{IB})
$30 - $3F	Temperature Sensor Voltage[1]

Note: 1. measured at 80°C

16.5.1 Data Format in Preprogrammed PEPROM

The 16-bit value is a binary representation of the measured data (4 digits, with the decimal point removed). Some examples are shown below.

For a measured data for f_{VCOMIN}=1.500MHz, it is converted to 1500 or 5DC (hex) and the hex data is programmed as 0000 0101 1101 1100.

For a measured data for f_{VCOMAX}=5.800MHz, it is converted to 5800 or 16A8 (hex) and the hex data is programmed as 0001 0110 1010 1000.

For a measured data for V_{IB}=1.211V, it is converted to 1211 or 4BB (hex) and the hex data is programmed as 0000 0100 1011 1011.

For a measured data for V_{TEMP}=836.0mV, it is converted to 8360 or 20A8 (hex) and the hex data is programmed as 0010 0000 1010 1000.

SECTION 17
INSTRUCTION SET

This section describes the addressing modes and instruction types.

17.1 ADDRESSING MODES

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

- Inherent
- Immediate
- Direct
- Extended
- Indexed, No Offset
- Indexed, 8-Bit Offset
- Indexed, 16-Bit Offset
- Relative

17.1.1 Inherent

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

17.1.2 Immediate

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

17.1.3 Direct

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

17.1.4 Extended

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

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

17.1.5 Indexed, No Offset

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

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

17.1.6 Indexed, 8-Bit Offset

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

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

17.1.7 Indexed, 16-Bit Offset

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

Indexed, 16-bit offset instructions are useful for selecting the 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.

17.1.8 Relative

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

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

17.1.9 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

17.1.10 Register/Memory Instructions

Most of these instructions use two operands. One operand is in either the accumulator or the index register. The CPU finds the other operand in memory. **Table 17-1** lists the register/memory instructions.

Table 17-1. 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

17.1.11 Read-Modify-Write Instructions

These instructions read a memory location or a register, modify its contents, and write the modified value back to the memory location or to the register. The test for negative or zero instruction (TST) is an exception to the read-modify-write sequence because it does not write a replacement value. **Table 17-2** lists the read-modify-write instructions.

Table 17-2. Read-Modify-Write Instructions

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

17.1.12 Jump/Branch Instructions

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

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

third byte to the program counter if the specified bit tests true. The bit to be tested and its condition (set or clear) is part of the opcode. The span of branching is from –128 to +127 from the address of the next location after the branch instruction. The CPU also transfers the tested bit to the carry/borrow bit of the condition code register. **Table 17-3** lists the jump and branch instructions.

Table 17-3. 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

17.1.13 Bit Manipulation Instructions

The CPU can set or clear any writable bit in the first 256 bytes of memory. Port registers, port data direction registers, timer registers, and on-chip RAM locations are in the first 256 bytes of memory. The CPU can also test and branch based on the state of any bit in any of the first 256 memory locations. Bit manipulation instructions use direct addressing. **Table 17-4** lists these instructions.

Table 17-4. Bit Manipulation Instructions

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

17.1.14 Control Instructions

These register reference instructions control CPU operation during program execution. Control instructions, listed in **Table 17-5**, use inherent addressing.

Table 17-5. 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

17.1.15 Instruction Set Summary

Table 17-6 is an alphabetical list of all M68HC05 instructions and shows the effect of each instruction on the condition code register.

Table 17-6. Instruction Set Summary

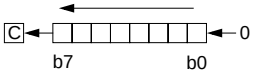
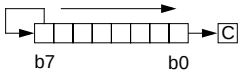
Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			H	I	N	Z	C				
ADC #*opr* ADC *opr* ADC *opr* ADC *opr*,X ADC *opr*,X ADC ,X	Add with Carry	A ← (A) + (M) + (C)	↕	—	↕	↕	↕	IMM DIR EXT IX2 IX1 IX	A9 B9 C9 D9 E9 F9	ii dd hh ll ee ff ff	2 3 4 5 4 3
ADD #*opr* ADD *opr*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

Table 17-6. Instruction Set Summary (Continued)

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

Table 17-6. Instruction Set Summary (Continued)

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

Table 17-6. Instruction Set Summary (Continued)

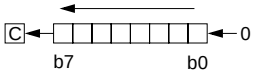
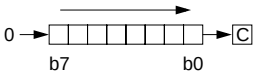
Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			H	I	N	Z	C				
INC *opr* INCA INCX INC *opr*,X INC ,X	Increment Byte	M ← (M) + 1 A ← (A) + 1 X ← (X) + 1 M ← (M) + 1 M ← (M) + 1	—	—	↕	↕	—	DIR INH INH IX1 IX	3C 4C 5C 6C 7C	dd ff 	5 3 3 6 5
JMP *opr* JMP *opr* JMP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

Table 17-6. Instruction Set Summary (Continued)

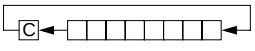
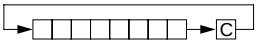
Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			H	I	N	Z	C				
ORA #*opr* ORA *opr* ORA *opr* ORA *opr*,X ORA *opr*,X ORA ,X	Logical OR Accumulator with Memory	A ← (A) ∨ (M)	—	—	↕	↕	—	IMM DIR EXT IX2 IX1 IX	AA BA CA DA EA FA	ii dd hh ll ee ff ff	2 3 4 5 4 3
ROL *opr* ROLA ROLX ROL *opr*,X ROL ,X	Rotate Byte Left through Carry Bit		—	—	↕	↕	↕	DIR INH INH IX1 IX	39 49 59 69 79	dd ff	5 3 3 6 5
ROR *opr* RORA RORX ROR *opr*,X ROR ,X	Rotate Byte Right through Carry Bit		—	—	↕	↕	↕	DIR INH INH IX1 IX	36 46 56 66 76	dd ff	5 3 3 6 5
RSP	Reset Stack Pointer	SP ← $00FF	—	—	—	—	—	INH	9C		2
RTI	Return from Interrupt	SP ← (SP) + 1; Pull (CCR) SP ← (SP) + 1; Pull (A) SP ← (SP) + 1; Pull (X) SP ← (SP) + 1; Pull (PCH) SP ← (SP) + 1; Pull (PCL)	↕	↕	↕	↕	↕	INH	80		9
RTS	Return from Subroutine	SP ← (SP) + 1; Pull (PCH) SP ← (SP) + 1; Pull (PCL)	—	—	—	—	—	INH	81		6
SBC #*opr* SBC *opr* SBC *opr* SBC *opr*,X SBC *opr*,X SBC ,X	Subtract Memory Byte and Carry Bit from Accumulator	A ← (A) – (M) – (C)	—	—	↕	↕	↕	IMM DIR EXT IX2 IX1 IX	A2 B2 C2 D2 E2 F2	ii dd hh ll ee ff ff	2 3 4 5 4 3
SEC	Set Carry Bit	C ← 1	—	—	—	—	1	INH	99		2
SEI	Set Interrupt Mask	I ← 1	—	1	—	—	—	INH	9B		2
STA *opr* STA *opr* STA *opr*,X STA *opr*,X STA ,X	Store Accumulator in Memory	M ← (A)	—	—	↕	↕	—	DIR EXT IX2 IX1 IX	B7 C7 D7 E7 F7	dd hh ll ee ff ff	4 5 6 5 4
STOP	Stop Oscillator and Enable $\overline{\text{IRQ}}$ Pin		—	0	—	—	—	INH	8E		2
STX *opr* STX *opr* STX *opr*,X STX *opr*,X STX ,X	Store Index Register In Memory	M ← (X)	—	—	↕	↕	—	DIR EXT IX2 IX1 IX	BF CF DF EF FF	dd hh ll ee ff ff	4 5 6 5 4

Table 17-6. Instruction Set Summary (Continued)

Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			H	I	N	Z	C				
SUB #*opr* SUB *opr* SUB *opr* SUB *opr*,X SUB *opr*,X SUB ,X	Subtract Memory Byte from Accumulator	A ← (A) – (M)	—	—	↕	↕	↕	IMM DIR EXT IX2 IX1 IX	A0 B0 C0 D0 E0 F0	ii dd hh ll ee ff ff	2 3 4 5 4 3
SWI	Software Interrupt	PC ← (PC) + 1; Push (PCL) SP ← (SP) – 1; Push (PCH) SP ← (SP) – 1; Push (X) SP ← (SP) – 1; Push (A) SP ← (SP) – 1; Push (CCR) SP ← (SP) – 1; I ← 1 PCH ← Interrupt Vector High Byte PCL ← Interrupt Vector Low Byte	—	1	—	—	—	INH	83		10
TAX	Transfer Accumulator to Index Register	X ← (A)	—	—	—	—	—	INH	97		2
TST *opr* TSTA TSTX TST *opr*,X TST ,X	Test Memory Byte for Negative or Zero	(M) – $00	—	—	↕	↕	—	DIR INH INH IX1 IX	3D 4D 5D 6D 7D	dd ff	4 3 3 5 4
TXA	Transfer Index Register to Accumulator	A ← (X)	—	—	—	—	—	INH	9F		2
WAIT	Stop CPU Clock and Enable Interrupts		—	0	—	—	—	INH	8F		2

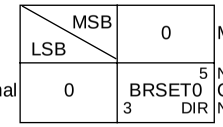
A	Accumulator	*opr*	Operand (one or two bytes)
C	Carry/borrow flag	PC	Program counter
CCR	Condition code register	PCH	Program counter high byte
dd	Direct address of operand	PCL	Program counter low byte
dd rr	Direct address of operand and relative offset of branch instruction	REL	Relative addressing mode
DIR	Direct addressing mode	*rel*	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 17-7. 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	BRSET0 5 3 DIR	BSET0 5 2 DIR	BRA 3 2 REL	NEG 5 2 DIR	NEGA 3 1 INH	NEGX 3 1 INH	NEG 6 2 IX1	NEG 5 1 IX	RTI 9 1 INH		SUB 2 2 IMM	SUB 3 2 DIR	SUB 4 3 EXT	SUB 5 3 IX2	SUB 4 2 IX1	SUB 3 1 IX	0
1	BRCLR0 5 3 DIR	BCLR0 5 2 DIR	BRN 3 2 REL						RTS 6 1 INH		CMP 2 2 IMM	CMP 3 2 DIR	CMP 4 3 EXT	CMP 5 3 IX2	CMP 4 2 IX1	CMP 3 1 IX	1
2	BRSET1 5 3 DIR	BSET1 5 2 DIR	BHI 3 2 REL		MUL 11 1 INH						SBC 2 2 IMM	SBC 3 2 DIR	SBC 4 3 EXT	SBC 5 3 IX2	SBC 4 2 IX1	SBC 3 1 IX	2
3	BRCLR1 5 3 DIR	BCLR1 5 2 DIR	BLS 3 2 REL	COM 5 2 DIR	COMA 3 1 INH	COMX 3 1 INH	COM 6 2 IX1	COM 5 1 IX	SWI 10 1 INH		CPX 2 2 IMM	CPX 3 2 DIR	CPX 4 3 EXT	CPX 5 3 IX2	CPX 4 2 IX1	CPX 3 1 IX	3
4	BRSET2 5 3 DIR	BSET2 5 2 DIR	BCC 3 2 REL	LSR 5 2 DIR	LSRA 3 1 INH	LSRX 3 1 INH	LSR 6 2 IX1	LSR 5 1 IX			AND 2 2 IMM	AND 3 2 DIR	AND 4 3 EXT	AND 5 3 IX2	AND 4 2 IX1	AND 3 1 IX	4
5	BRCLR2 5 3 DIR	BCLR2 5 2 DIR	BCS/BLO 3 2 REL								BIT 2 2 IMM	BIT 3 2 DIR	BIT 4 3 EXT	BIT 5 3 IX2	BIT 4 2 IX1	BIT 3 1 IX	5
6	BRSET3 5 3 DIR	BSET3 5 2 DIR	BNE 3 2 REL	ROR 5 2 DIR	RORA 3 1 INH	RORX 3 1 INH	ROR 6 2 IX1	ROR 5 1 IX			LDA 2 2 IMM	LDA 3 2 DIR	LDA 4 3 EXT	LDA 5 3 IX2	LDA 4 2 IX1	LDA 3 1 IX	6
7	BRCLR3 5 3 DIR	BCLR3 5 2 DIR	BEQ 3 2 REL	ASR 5 2 DIR	ASRA 3 1 INH	ASRX 3 1 INH	ASR 6 2 IX1	ASR 5 1 IX		TAX 2 1 INH		STA 4 2 DIR	STA 5 3 EXT	STA 6 3 IX2	STA 5 2 IX1	STA 4 1 IX	7
8	BRSET4 5 3 DIR	BSET4 5 2 DIR	BHCC 3 2 REL	ASL/LSL 5 2 DIR	ASLA/LSLA 3 1 INH	ASLX/LSLX 3 1 INH	ASL/LSL 6 2 IX1	ASL/LSL 5 1 IX		CLC 2 1 INH	EOR 2 2 IMM	EOR 3 2 DIR	EOR 4 3 EXT	EOR 5 3 IX2	EOR 4 2 IX1	EOR 3 1 IX	8
9	BRCLR4 5 3 DIR	BCLR4 5 2 DIR	BHCS 3 2 REL	ROL 5 2 DIR	ROLA 3 1 INH	ROLX 3 1 INH	ROL 6 2 IX1	ROL 5 1 IX		SEC 2 1 INH	ADC 2 2 IMM	ADC 3 2 DIR	ADC 4 3 EXT	ADC 5 3 IX2	ADC 4 2 IX1	ADC 3 1 IX	9
A	BRSET5 5 3 DIR	BSET5 5 2 DIR	BPL 3 2 REL	DEC 5 2 DIR	DECA 3 1 INH	DECX 3 1 INH	DEC 6 2 IX1	DEC 5 1 IX		CLI 2 1 INH	ORA 2 2 IMM	ORA 3 2 DIR	ORA 4 3 EXT	ORA 5 3 IX2	ORA 4 2 IX1	ORA 3 1 IX	A
B	BRCLR5 5 3 DIR	BCLR5 5 2 DIR	BMI 3 2 REL							SEI 2 1 INH	ADD 2 2 IMM	ADD 3 2 DIR	ADD 4 3 EXT	ADD 5 3 IX2	ADD 4 2 IX1	ADD 3 1 IX	B
C	BRSET6 5 3 DIR	BSET6 5 2 DIR	BMC 3 2 REL	INC 5 2 DIR	INCA 3 1 INH	INCX 3 1 INH	INC 6 2 IX1	INC 5 1 IX		RSP 2 1 INH		JMP 2 2 DIR	JMP 3 3 EXT	JMP 4 3 IX2	JMP 3 2 IX1	JMP 2 1 IX	C
D	BRCLR6 5 3 DIR	BCLR6 5 2 DIR	BMS 3 2 REL	TST 4 2 DIR	TSTA 3 1 INH	TSTX 3 1 INH	TST 5 2 IX1	TST 4 1 IX		NOP 2 1 INH	BSR 6 2 REL	JSR 5 2 DIR	JSR 6 3 EXT	JSR 7 3 IX2	JSR 6 2 IX1	JSR 5 1 IX	D
E	BRSET7 5 3 DIR	BSET7 5 2 DIR	BIL 3 2 REL						STOP 2 1 INH		LDX 2 2 IMM	LDX 3 2 DIR	LDX 4 3 EXT	LDX 5 3 IX2	LDX 4 2 IX1	LDX 3 1 IX	E
F	BRCLR7 5 3 DIR	BCLR7 5 2 DIR	BIH 3 2 REL	CLR 5 2 DIR	CLRA 3 1 INH	CLRX 3 1 INH	CLR 6 2 IX1	CLR 5 1 IX	WAIT 2 1 INH	TXA 2 1 INH		STX 4 2 DIR	STX 5 3 EXT	STX 6 3 IX2	STX 5 2 IX1	STX 4 1 IX	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 of Opcode in Hexadecimal
Number of Cycles
Opcode Mnemonic
Number of Bytes/Addressing Mode

SECTION 18
ELECTRICAL SPECIFICATIONS

This section describes the electrical and timing specifications of the MC68HC05SB7.

18.1 MAXIMUM RATINGS

NOTE

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

The MCU contains circuitry that protect the inputs against damage from high static voltages; however, do not apply voltages higher than those shown in the table below. Keep V_{IN} and V_{OUT} within the range from $V_{SS} \le (V_{IN}$ or $V_{OUT}) \le V_{DD}$. Connect unused inputs to the appropriate voltage level, either V_{SS} or V_{DD}.

Rating	Symbol	Value	Unit
Supply Voltage	V_{DD}	–0.3 to +7.0	V
Bootloader Mode ($\overline{IRQ}/V_{PP}$ Pin Only)	V_{IN}	V_{SS} – 0.3 to 17	V
Current Drain Per Pin Excluding V_{DD} and V_{SS}	I	25	mA
Operating Junction Temperature	T_J	+150	°C
Storage Temperature Range	T_{stg}	–65 to +150	°C

18.2 OPERATING TEMPERATURE RANGE

Characteristic	Symbol	Value	Unit
Operating Temperature Range MC68HC05SB7 (Standard)	 T_A	T_L to T_H 0 to +70	 °C

18.3 THERMAL CHARACTERISTICS

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

18.4 SUPPLY CURRENT CHARACTERISTICS

Characteristic	Symbol	Min	Typ	Max	Unit
V_{DD} = 4.5 to 5.5 Vdc					
Run, All Analog and LVR enabled					
Internal VCO at 2.5kHz	I_{DD}	—	3	5	mA
External Oscillator at 4.2MHz	I_{DD}	—	4.8	8	mA
Wait					
Internal VCO at 2.5kHz	I_{DD}	—	1	1.5	mA
External Oscillator at 4.2 MHz	I_{DD}	—	1.3	2	mA
Stop - all clocks disabled					
All Analog/LVR disabled and CSA enabled	I_{DD}	—	280	500	µA
All Analog and LVR disabled	I_{DD}	—	6	10	µA
All Analog disabled and LVR enabled	I_{DD}	—	8	20	µA
All Analog and LVR enabled	I_{DD}	—	200	350	µA

NOTES:

1. V_{DD} as indicated, V_{SS} = 0 V, $T_L \leq T_A \leq T_H$, unless otherwise noted.
2. All values shown reflect average measurements.
3. Typical values at midpoint of voltage range, 25°C only.
4. Run (Operating) I_{DD}, Wait I_{DD}: Measured using external square wave clock source to OSC1 pin or internal oscillator, all inputs 0.2 Vdc from either supply rail (V_{DD} or V_{SS}); no dc loads, less than 50pF on all outputs, C_L = 20pF on OSC2.
5. Wait, Stop I_{DD}: All ports configured as inputs, V_{IL} = 0.2 VDC, $V_{IH} = V_{DD} – 0.2$ VDC.
6. Stop I_{DD} measured with OSC1 = V_{DD}.
7. Wait I_{DD} is affected linearly by the OSC2 capacitance.

18.5 PEPROM PROGRAMMING CHARACTERISTICS

Characteristic	Symbol	Min	Typ	Max	Unit
PEPROM Programming Voltage	V_{PP}	—	13.7	—	V
PEPROM Programming Current	I_{PP}	—	3	10	mA
PEPROM Programming Time per Byte	t_{EPGM}	2	—	—	ms

NOTES:

1. V_{DD}=5V ± 10%, V_{SS} = 0 V, $T_L \leq T_A \leq T_H$, unless otherwise noted.

18.6 DC ELECTRICAL CHARACTERISTICS

Characteristic	Symbol	Min	Typ	Max	Unit
Output Voltage I_{load} = 10.0 µA I_{load} = –10.0 µA	 V_{OL} V_{OH}	 — V_{DD}– 0.1	 — —	 0.1 —	 V V
Output High Voltage (I_{load} = –0.8 mA) PA0:7, PB1:7, PC4:7, $\overline{RESET}$	V_{OH}	V_{DD}– 0.8	—	—	V
Output Low Voltage (I_{load} = 1.6 mA) PA0:7, PB1:7, PC4:7, $\overline{RESET}$	V_{OL}	—	—	0.4	V
High Source Current (V_{OH} = V_{DD} – 0.5 to 1.0 Vdc) Source current per pin, PA0:7, PB1:7, PC4:7 Source current total for all pins	 I_{OH} I_{OH}	 — —	 — —	 4 —	 mA mA
High Sink Current (V_{OL} = V_{SS} + 1.5 Vdc) Sink current per pin, PA0:7, PB1:7, PC4:7 Sink current total for all pins	 I_{OL} I_{OL}	 — —	 — —	 12 —	 mA mA
High Source Current (V_{OH} = V_{DD} – 0.2Vdc) Source current for pin, ESV	I_{OH}	—	—	3	mA
Input High Voltage PA0:7, PB1:7, PC4:7, $\overline{RESET}$, OSC1, $\overline{IRQ}$/V_{PP}	V_{IH}	0.7 X V_{DD}	—	V_{DD}	V
Input Low Voltage PA0:7, PB1:7, PC4:7, $\overline{RESET}$, OSC1, $\overline{IRQ}$/V_{PP}	V_{IL}	V_{SS}	—	0.3 x V_{DD}	V
Input Current PA0:7, PB1:7, PC4:7, $\overline{RESET}$, OSC1, $\overline{IRQ}$/V_{PP}	I_{IN}	—	—	±1	µA
I/O Ports High-Z Leakage Current PA0:7, PB1:7, PC4:7	I_{OZ}	—	—	±10	µA
Internal Bandgap Voltage	V_{BG}	1.1	1.2	1.3	V
Internal Temperature Sensor Temperature Gradient		2.0	2.2	2.4	mV/°C

NOTES:

1. V_{DD}=5V ± 10%, V_{SS} = 0 V, $T_L \le T_A \le T_H$, unless otherwise noted.
2. All values shown reflect average measurements.
3. Typical values at midpoint of voltage range, 25°C only.

18.7 ANALOG SUBSYSTEM CHARACTERISTICS

Characteristic	Symbol	Min	Max	Unit
Voltage Comparator Input Offset Voltage	V_{IO}	5	10	mV
Voltage Comparator Input Common-Mode Range	V_{CMR}	—	V_{DD} – 1.5	V
Voltage Comparator Supply Current	I_{CMP}	—	150	µA
Voltage Comparator Input Divider Ratio	R_{DIV}	0.49	0.51	
External Capacitor Current Source	I_{SOURCE}	80	120	µA
Current Source Supply Current	I_{RAMP}	—	220	µA
Source Current Linearity	I_{RAMP}	—	1.0	%FS
Discharge Sink Current (V_{OUT} = 0.4 V)	I_{DIS}	0.8	—	mA
External Capacitor Voltage Range	V_{IN}	V_{SS}	V_{DD} – 1.5	V
Comparator Input Impedance Comparator Used as comparator only (DHOLD =0) Used as A/D function (DHOLD =1)	 Z_{IN} Z_{IN}	 0.8 80	 — —	 MΩ kΩ
Multiplexer Switch Resistance	R_{MUX}	4.5	7	kΩ
Internal Sample & Hold Capacitor Capacitance Charge/Discharge Time (0 to 3.5 VDC, DHOLD =0) Charge/Discharge Time (0 to 3.5 VDC, DHOLD =1)	 C_{SH} t_{SHCHG} t_{SHDCHG}	 4 20 60	 6 — —	 pF µs µs

NOTES:

1. V_{DD}=5V ± 10%, V_{SS} = 0 V, $T_L \leq T_A \leq T_H$, unless otherwise noted.

18.8 CONTROL TIMING

Characteristic	Symbol	Min	Max	Unit
Frequency of Oscillation (OSC) Crystal Oscillator Option External Clock Source Internal VCO (SCLK = 0)[2] Internal VCO (SCLK = 1)[2]	 f_{OSC} f_{OSC} f_{OSC} f_{OSC}	 0.1 DC 1.5 0.5	 4.2 4.2 5.8 4	 MHz MHz MHz kHz
Internal Operating Frequency, Crystal or External Clock (f_{OSC}/2) Crystal Oscillator Option External Clock Source	 f_{OP} f_{OP}	 0.05 DC	 2.1 2.1	 MHz MHz
Cycle Time Crystal Oscillator or External Clock source	 t_{CYC}	 476	 —	 ns
Timer Resolution Input Capture (TCAP) pulse width	 t_{RESL} t_{TH}, t_{TL}	 4.0 284	 — —	 t_{CYC} ns
Interrupt Pulse Width Low (Edge-Triggered)	t_{ILIH}	284	—	ns
Interrupt Pulse Period	t_{ILIL}	see note 3	—	t_{CYC}
OSC1 Pulse Width (external clock input)	t_{OH},t_{OL}	110	—	ns
Voltage Comparator Switching Time (10 mV overdrive, either input)	 t_{PROP}	 —	 10	 µs
Voltage Comparator Power Up Delay (Bias Circuit already powered up)	 t_{DELAY}	 —	 100	 µs
External Capacitor Switching Time (I_{DIS} to I_{RAMP})	t_{PROP}	—	10	µs
External Capacitor Current Source Power Up Delay (Bias Circuit already powered up)	 t_{DELAY}	 —	 100	 µs
Bias Circuit Power Up Delay	t_{DELAY}	—	100	µs

NOTES:

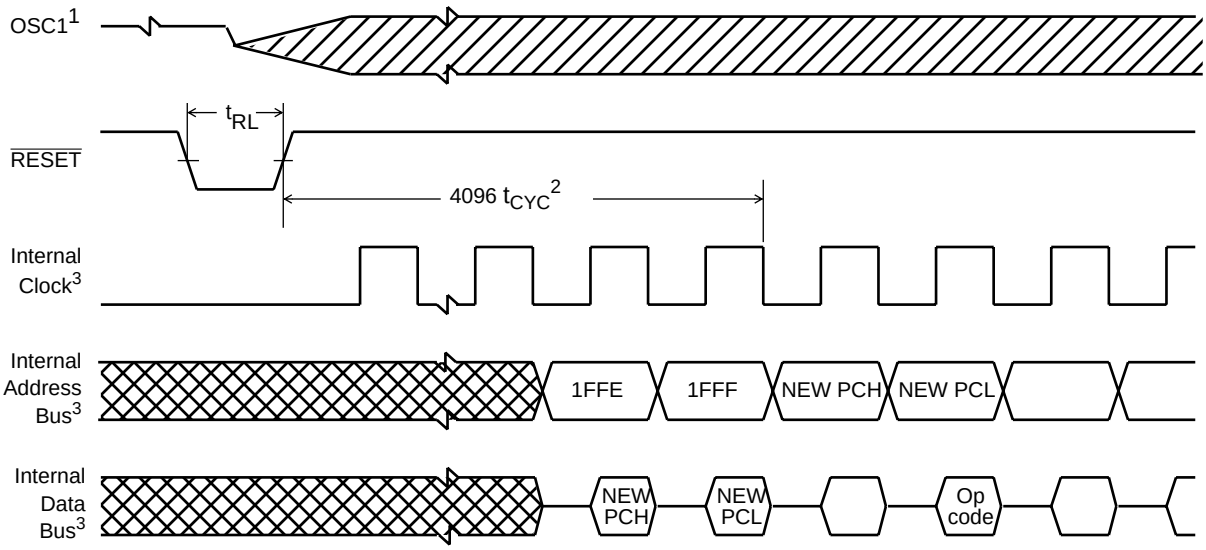
1. V_{DD}=5V ± 10%, V_{SS} = 0 V, $T_L \leq T_A \leq T_H$, unless otherwise noted.
2. Due to process variations, operating voltages, and temperature requirements, the quoted VCO frequencies are typical limits, and should be treated as reference only. It is the user's responsibility to ensure that the resulting internal operating frequency meets user's requirement by setting the appropriate value in the VCO Adjust Register.
3. The minimum period t_{ILIL} should not be less than the number of cycle times it takes to execute the interrupt service routine plus 21 t_{CYC}.

18.9 RESET CHARACTERISTICS

Characteristic	Symbol	Min	Typ	Max	Unit
Low Voltage Reset					
Rising Recovery Voltage	V_{LVRR}	1.3	2.3	3.1	V
Falling Reset Voltage	V_{LVRF}	1.2	2.2	3.0	V
LVR Hysteresis	V_{LVRH}	100	—	—	mV
POR Recovery Voltage[2]	V_{POR}	0	—	100	mV
POR V_{DD} Slew Rate [2]					
Rising	S_{VDDR}	0.1	—	—	V/µs
Falling	S_{VDDF}	0.05	—	—	V/µs
$\overline{\text{RESET}}$ Pulse Width (when bus clock active)	t_{RL}	1.5	—	—	t_{CYC}
$\overline{\text{RESET}}$ Pulldown Pulse Width (from internal reset)	t_{RPD}	3	—	4	t_{CYC}

NOTES:

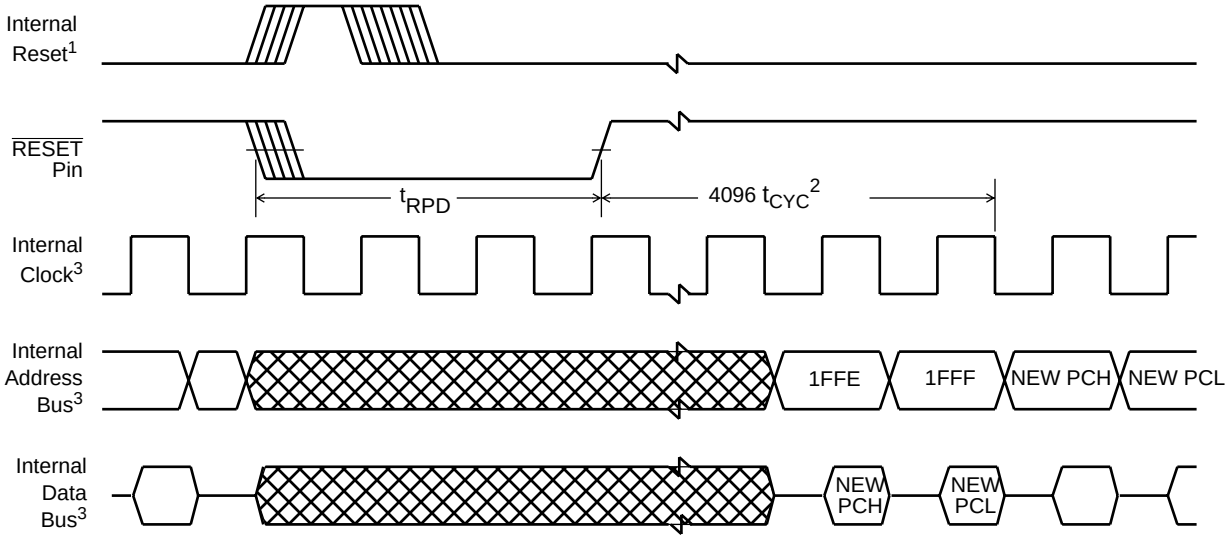
1. V_{DD}=5V ± 10%, V_{SS} = 0 V, $T_L \le T_A \le T_H$, unless otherwise noted.
2. By design, not tested.



NOTES:

1. Represents the internal gating of the OSC1 pin.
2. Normal delay of 4064 $t_{CYC.}$
3. Internal timing signal and data information not available externally.

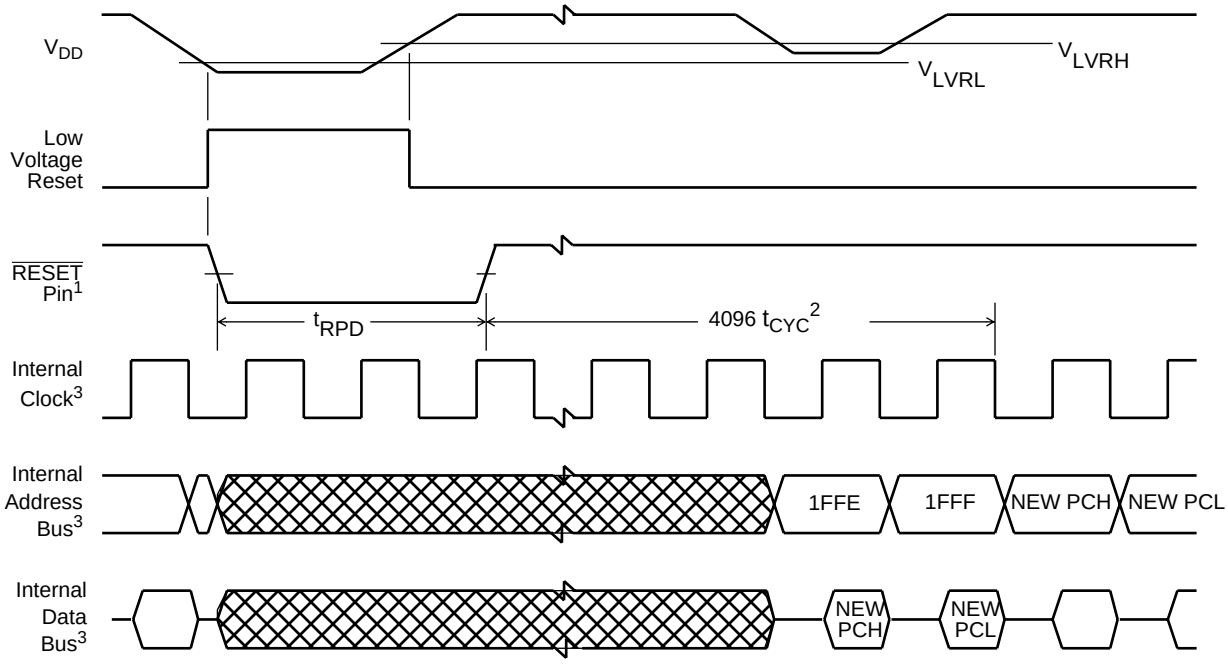
Figure 18-1. Stop Recovery Timing Diagram



NOTES:

1. Represents the internal reset from low voltage reset, illegal opcode fetch or COP Watchdog timeout.
2. Normal delay of 4064 $t_{CYC.}$
3. Internal timing signal and data information not available externally.

Figure 18-2. Internal Reset Timing Diagram



NOTES:

1. $\overline{\text{RESET}}$ pin pulled down be internal device.
2. Normal delay of 4064 $t_{CYC.}$
3. Internal timing signal and data information not available externally.

Figure 18-3. Low Voltage Reset Timing Diagram

18.10 SM-BUS DC ELECTRICAL CHARACTERISTICS

Characteristic	Symbol	Min	Max	Unit
LOW level input voltage	V_{IL}	V_{SS}	0.3 x V_{DD}	V
HIGH level input voltage	V_{IH}	0.7 x V_{DD}	5.5	V
LOW level output voltage (open drain); at 2.86mA sink current (R_{PULLUP}=1.7kΩ and C_{LOAD}=400pF)	V_{OL}	V_{SS}	0.135	V

NOTES:

1. V_{DD}=5V ± 10%, V_{SS} = 0 V, $T_L \le T_A \le T_H$, unless otherwise noted.
2. For SM-Bus specification: logic "LOW"=0.6V or less; logic "HIGH"=1.4V or above.

18.11 SM-BUS CONTROL TIMING

18.11.1SM-Bus Interface Input Signal Timing

Parameter	Symbol	Min	Max	Unit
Start condition hold time	$t_{HD.STA}$	2	—	t_{CYC}
Clock low period	t_{LOW}	4.7	—	t_{CYC}
SDA/SCL rise time	t_R	—	1	µs
Data hold time	$t_{HD.DAT}$	300	—	ns
SDA/SCL fall time	t_F	—	300	ns
Clock high period	t_{HIGH}	4	—	t_{CYC}
Data set up time	$t_{SU.DAT}$	250	—	ns
Start condition set up time (for repeated start condition only)	$t_{SU.STA}$	2	—	t_{CYC}
Stop condition set up time	$t_{SU.STO}$	2	—	t_{CYC}

NOTES:

1. V_{DD}=5V ± 10%, V_{SS} = 0 V, $T_L \le T_A \le T_H$, unless otherwise noted.

18.11.2SM-Bus Interface Output Signal Timing

Parameter	Symbol	Min	Max	Unit
Start condition hold time	$t_{HD.STA}$	8	—	t_{CYC}
Clock low period	t_{LOW}	11	—	t_{CYC}
SDA/SCL rise time	t_R	—	1	µs
Data hold time	$t_{HD.DAT}$	300	—	ns
SDA/SCL fall time	t_F	—	300	ns
Clock high period	t_{HIGH}	11	—	t_{CYC}
Data set up time	$t_{SU.DAT}$	t_{LOW}–t_{CYC}	—	ns
Start condition set up time (for repeated start condition only)	$t_{SU.STA}$	10	—	t_{CYC}
Stop condition set up time	$t_{SU.STO}$	10	—	t_{CYC}

NOTES:

1. V_{DD}=5V ± 10%, V_{SS} = 0 V, $T_L \le T_A \le T_H$, unless otherwise noted.

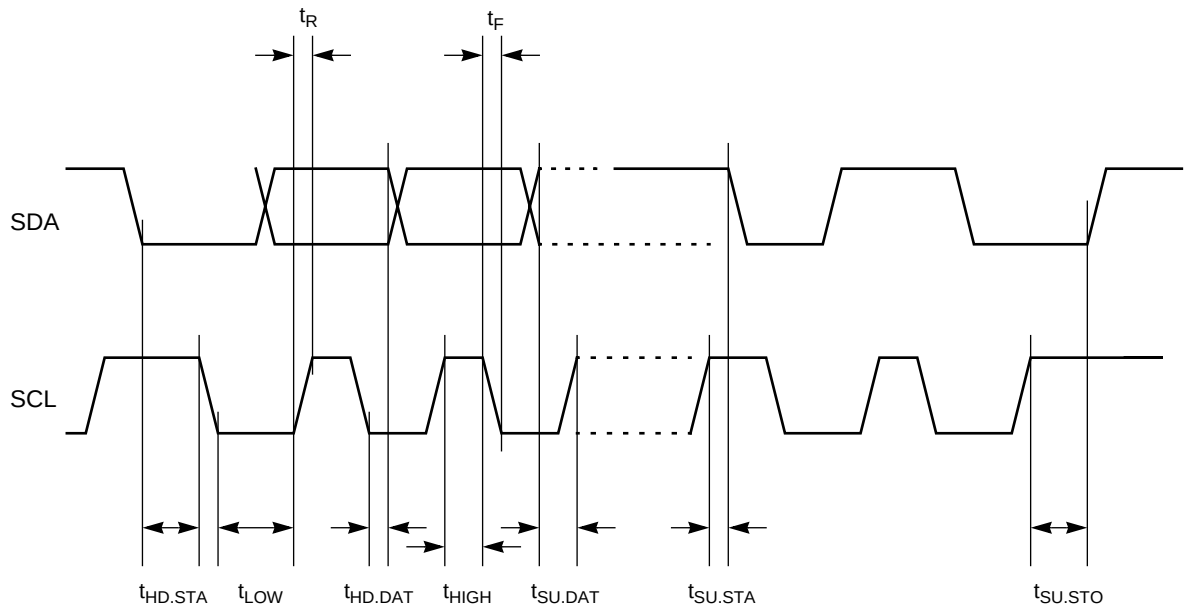
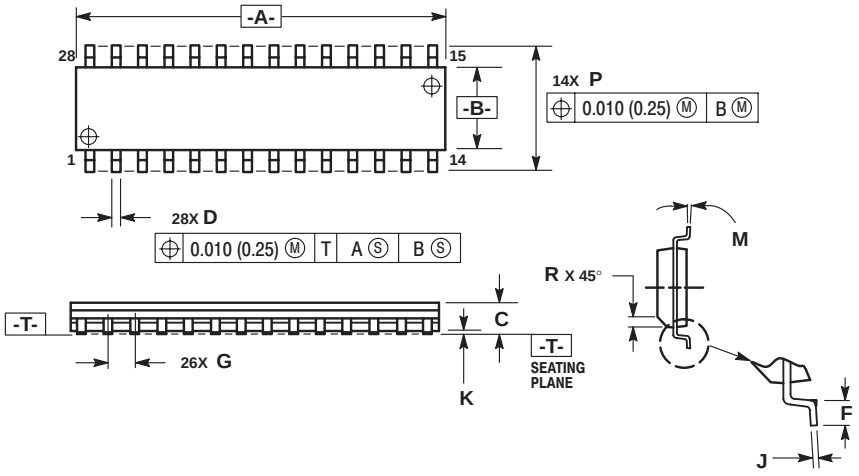


Figure 18-4. SM-Bus Timing Diagram

SECTION 19
MECHANICAL SPECIFICATIONS

This section provides the mechanical dimensions for the 28-pin SOIC and 28-pin SSOP packages.

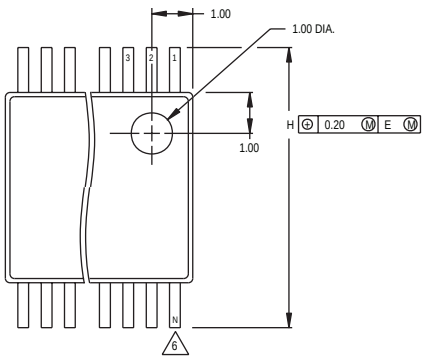
19.1 28-PIN SOIC (CASE 751F)



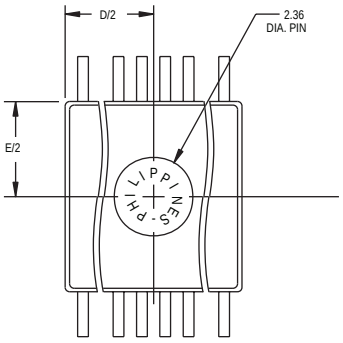
NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: MILLIMETER.
3. DIMENSION A AND B DO NOT INCLUDE MOLD PROTRUSION.
4. MAXIMUM MOLD PROTRUSION 0.15 (0.006) PER SIDE.
5. DIMENSION D DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL BE 0.13 (0.005)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

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

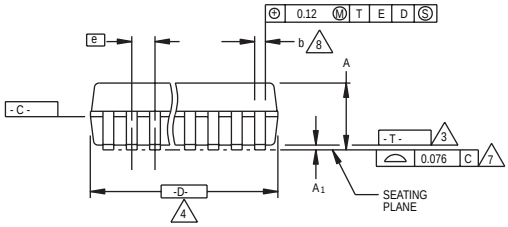
19.2 28-PIN SSOP



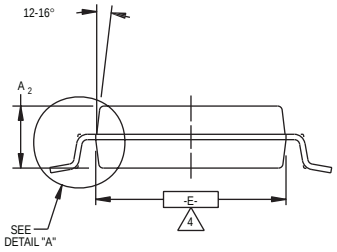
TOP VIEW



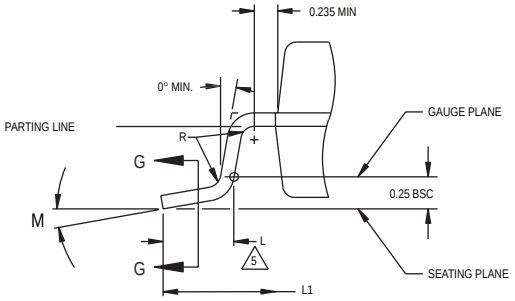
BOTTOM VIEW



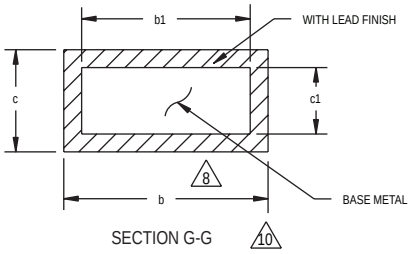
SIDE VIEW



END VIEW



DETAIL 'A'



SECTION G-G 10

NOTES:

1. MAXIMUM DIE THICKNESS ALLOWABLE IS 0.43mm (.017 INCHES).
2. DIMENSIONING & TOLERANCES PER ANSI.Y14.5M-1982.
3. "T" IS A REFERENCE DATUM.
4. "D" & "E" ARE REFERENCE DATUMS AND DO NOT INCLUDE MOLD FLASH OR PROTRUSIONS, BUT DO INCLUDE MOLD MISMATCH AND ARE MEASURED AT THE PARTING LINE, MOLD FLASH OR PROTRUSIONS SHALL NOT EXCEED 0.15mm PER SIDE.
5. DIMENSION IS THE LENGTH OF TERMINAL FOR SOLDERING TO A SUBSTRATE.
6. TERMINAL POSITIONS ARE SHOWN FOR REFERENCE ONLY.
7. FORMED LEADS SHALL BE PLANAR WITH RESPECT TO ONE ANOTHER WITHIN 0.08mm AT SEATING PLANE.
8. DIMENSION b DOES NOT INCLUDE DAMBAR PROTRUSION/INTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL BE 0.13mm TOTAL IN EXCESS OF b DIMENSION AT MAXIMUM MATERIAL CONDITION. DAMBAR INTRUSION SHALL NOT REDUCE DIMENSION b BY MORE THAN 0.07mm AT LEAST MATERIAL CONDITION.
9. CONTROLLING DIMENSION: MILLIMETERS.
10. THESE DIMENSIONS APPLY TO THE FLAT SECTION OF THE LEAD BETWEEN 0.10 AND 0.25mm FROM LEAD TIPS.
11. THIS PACKAGE OUTLINE DRAWING COMPLIES WITH JEDEC SPECIFICATION NO. MO-150 FOR THE LEAD COUNTS SHOWN

SYMBOL	DIMENSIONS IN MM			DIMENSIONS IN INCH			NOTE
	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.	
A	1.73	1.86	1.99	.068	.073	.078	
A1	0.05	0.13	0.21	.002	.005	.008	
A2	1.68	1.73	1.78	.066	.068	.070	
b	0.25	—	0.38	.010	—	.015	8,10
b1	0.25	0.30	0.33	.010	.012	.013	10
c	0.09	—	0.20	.004	—	.008	10
c1	0.09	0.15	0.16	.004	.006	.006	10
D	10.07	10.20	10.33	.397	.402	.407	4
E	5.20	5.30	5.38	.205	.209	.212	4
e	0.65 BSC			.0256 BSC			
H	7.65	7.80	7.90	.301	.307	.311	
L	0.63	0.75	0.95	.025	.030	.037	5
L1	1.25 REF.			.049 REF.			
N	28			28			6
M	0	4	8	0	4	8	
R	0.09	0.15		.004	.006		

APPENDIX A
MC68HC705SB7

This appendix describes the MC68HC705SB7, the emulation part for MC68HC05SB7. The entire MC68HC05SB7 data sheet applies to the MC68HC705SB7, with exceptions outlined in this appendix.

A.1 INTRODUCTION

The MC68HC705SB7 is an EPROM version of the MC68HC05SB7, and is available for user system evaluation and debugging. The MC68HC705SB7 is functionally identical to the MC68HC05SB7 with the exception of the 6106 bytes user ROM is replaced by 6106 bytes user EPROM. The mask option for the external pin oscillator on the MC68HC05SB7 is controlled by the Mask Option Register at $002F on the MC68HC705SB7. This device is available in 28-pin SOIC package.

A.2 MEMORY

The MC68HC705SB7 memory map is shown on **Figure A-1**.

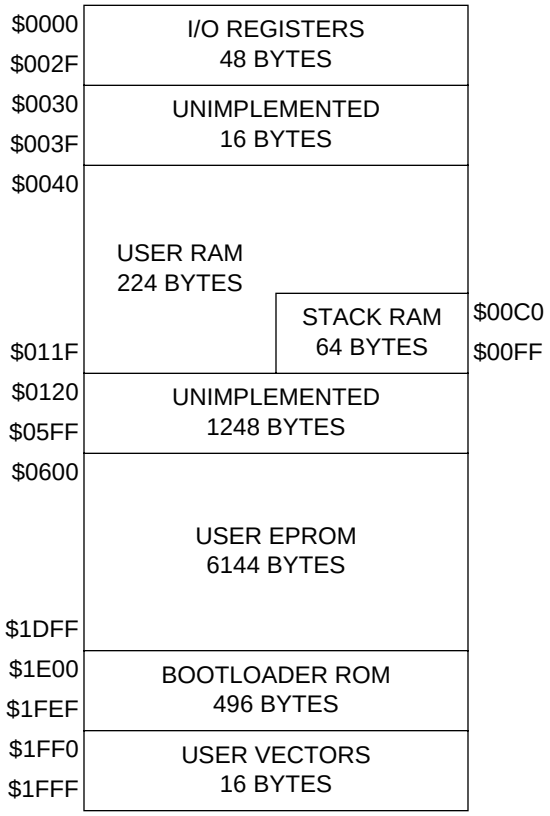


Figure A-1. MC68HC705SB7 Memory Map

A.3 PERSONALITY EPROM (PEPROM)

The 64-bit PEPROM is left blank for user programming.

A.4 MASK OPTION REGISTER

The EPROM programmable Mask Option Register is used for setting EPROM security and enabling the external pin oscillator.

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
MOR	R	EPMSEC	OSCS	-	-	-	-	-	-
$002F	W								
reset:		U	U	U	U	U	U	U	U
erased:		0	0	-	-	-	-	-	-

U = UNAFFECTED BY RESET

Figure A-2. MC68HC705SB7 Mask Option Register (MOR)

EPMSEC — EPROM Security Bit

1 = Access to the EPROM array in non-user mode is denied.
0 = Access to the EPROM array in non-user mode is enabled.

This write-only bit controls the non-user mode access to the EPROM array on the MCU. When programmed to “1”, any accesses of the EPROM locations will return undefined results.

EPMSEC Programming

The state of the EPMSEC security bit should be programmed using a programmer board (available from Motorola). In order to program the EPMSEC bit the desired state must be written to the MOR address and then the MPGM bit in the EPROG register must be used. The following sequence will program the EPMSEC bit:

1. Write the desired data to the EPMSEC bit in MOR.
2. Apply the programming voltage to the $\overline{IRQ}/V_{PP}$ pin.
3. Set the MPGM bit in the EPROG.
4. Wait for the programming time (t_{MPGM}).
5. Clear the MPGM bit in the EPROG.
6. Remove the programming voltage from the $\overline{IRQ}/V_{PP}$ pin.

Once the EPMSEC bit has been programmed to a “1”, access to the contents of the EPROM in the non-user mode will be denied. It is therefore recommended that the User EPROM in the part first be programmed and fully verified before setting the EPMSEC bit.

OSCS — Oscillator Select Bit

1 = External pin oscillator (EPO) enabled.
0 = External pin oscillator (EPO) disabled.

The OSCS bit enables the OSC1 and OSC2 pins for external oscillator connection. OSC1 replaces PB2/CS0 and OSC2 replaces PB3/CS1. This is selected by a mask option on the MC68HC05SB7 device.

A.5 BOOTLOADER MODE

Bootloader mode is entered upon the rising edge of $\overline{\text{RESET}}$ if $\overline{\text{IRQ}}/V_{PP}$ pin is at V_{TST} and PB1/TCAP at V_{DD}. The Bootloader program is masked in the ROM area from $1E00 to $1FEF. This program handles copying of user code from an external EPROM into the on-chip EPROM. The bootload function has to be done from an external EPROM. The bootloader performs one programming pass at 1ms per byte then does a verify pass.

A.6 EPROM PROGRAMMING

This section describes how to program the 6160-byte EPROM and the EPROM security bit.

In packages with no quartz window, the EPROM functions as one-time programmable ROM (OTPROM)

Programming the on-chip EPROM is achieved by using the Program Control Register located at address $001E.

The programming software copies to the 6144-byte space located at EPROM addresses $0600 – $1DFF and to the 16-byte space at addresses $1FF0 – $1FFF which includes the mask option register (MOR) at address $002F.

Please contact Motorola for programming board availability.

A.6.1 EPROM Programming Register (EPROG)

The EPROM programming register shown in **Figure A-3** contains the control bits for programming the EPROM and MOR. In normal operation, the EPROM programming register is a read-only register that contains all logic zeros.

		BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
EPROG	R	0	0	0	0	0	ELAT	MPGM	EPGM
$001C	W								
reset:		0	0	0	0	0	0	0	0

UNIMPLEMENTED RESERVED FOR TEST

Figure A-3. EPROM Programming Register (EPROG)

EPGM — EPROM Programming

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

1 = EPROM programming power switched on.
0 = EPROM programming power switched off.

MPGM — Mask Option Register (MOR) Programming

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

1 = MOR programming power switched on.
0 = MOR programming power switched off.

ELAT — EPROM Bus Latch

This read/write bit configures address and data buses for programming the EPROM array. EPROM data cannot be read when ELAT is set. Clearing the ELAT bit also clears the EPGM bit. Reset clears ELAT.

1 = Address and data buses configured for EPROM programming of the array. The address and data bus are latched in the EPROM array when a subsequent write to the array is made. Data in the EPROM array cannot be read.
0 = Address and data buses configured for normal operation.

Whenever the ELAT bit is cleared the EPGM bit is also cleared. Both the EPGM and the ELAT bit cannot be set using the same write instruction. Any attempt to set both the ELAT and EPGM bit on the same write instruction cycle will result in the ELAT bit being set and the EPGM bit being cleared.

A.6.2 Programming Sequence

The EPROM programming sequence is:

1. Set the ELAT bit in the EPROG register.
2. Write the desired data to the desired EPROM address.
3. Set the EPGM bit in the EPROG register for the specified programming time (t_{EPGM}).
4. Clear the EPGM bit
5. Clear the ELAT bit

The last two steps must be performed with separate CPU writes.

CAUTION

It is important to remember that an external programming voltage must be applied to the V_{PP} pin while programming, but it should be equal to V_{DD} during normal operations.

Figure A-4 shows the flow required to successfully program the EPROM.

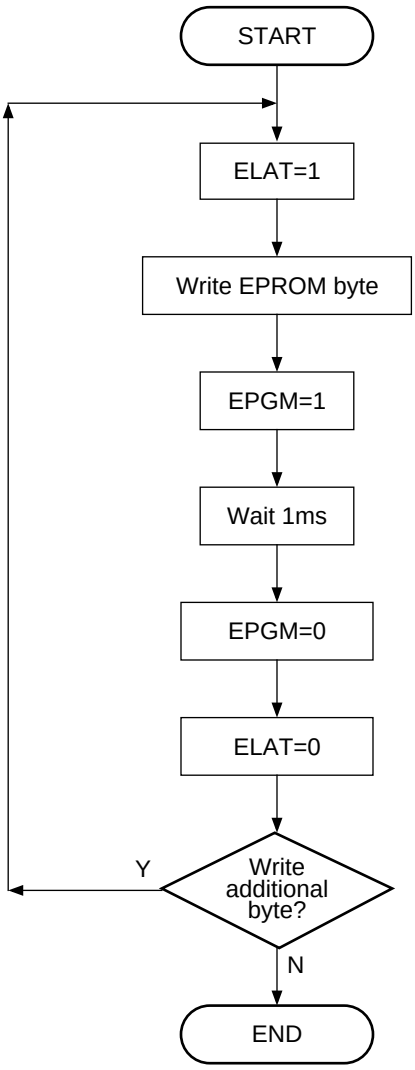


Figure A-4. EPROM Programming Sequence

A.7 EPROM ERASING

MCUs with windowed packages permit EPROM erasure with ultraviolet light. 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 1 inch from the window. Do not use a shortwave filter. The erased state of an EPROM bit is a logic one.

A.8 EPROM PROGRAMMING SPECIFICATIONS

Characteristic	Symbol	Min	Typ	Max	Unit
MOR Programming Time	t_{MPGM}	4	—	—	µs
EPROM Programming Voltage	V_{PP}		13.7		V
EPROM Programming Current	I_{PP}	—	3	5	mA
EPROM Programming Time per Byte	t_{EPGM}	4	—	—	µs

NOTES:

1. V_{DD}=5V ± 10%, V_{SS} = 0 V, $T_L \leq T_A \leq T_H$, unless otherwise noted.

How to reach us:

USA/EUROPE/Locations Not Listed: Motorola Literature Distribution; P.O. Box 5405, Denver, Colorado 80217. 1-800-441-2447 or 1-303-675-2140
JAPAN: Nippon Motorola Ltd. SPD, Strategic Planning Office 4-32-1, Nishi-Gotanda, Shinagawa-ku, Tokyo 141, Japan. 03-5487-8488
ASIA/PACIFIC: Motorola Semiconductors H.K. Ltd.; 8B Tai Ping Industrial Park, 51 Ting Kok Road, Tai Po, N.T., Hong Kong. 852-26629298
Mfax™, Motorola Fax Back System: RMFAX0@email.sps.mot.com; http://sps.motorola.com/mfax/; TOUCHTONE 1-602-244-6609;
US and Canada ONLY 1-800-774-1848
HOME PAGE: http://motorola.com/sps/