68HC05PD6
68HC705PD6

SPECIFICATION
(General Release)

Technical Operations Taiwan
Asia Pacific Semiconductor Products Group

TABLE OF CONTENTS

Section	Title	Page

SECTION 1
GENERAL DESCRIPTION

1.1 FEATURES 1-1
1.2 MASK OPTIONS 1-2
1.3 SIGNAL DESCRIPTION 1-4
1.3.1 VDD and VSS 1-4
1.3.2 OSC2, OSC1 1-5
1.3.3 XOSC2, XOSC1 1-5
1.3.4 $\overline{\text{RESET}}$ 1-6
1.3.5 VPP 1-6
1.3.6 PA0-PA7 1-6
1.3.7 PB0/$\overline{\text{KWI0}}$-PB7/$\overline{\text{KWI7}}$ 1-6
1.3.8 PC0/RDI, PC1/TDO 1-6
1.3.9 PC2/TCMP, PC3/TCAP 1-7
1.3.10 PC4/EVI, PC5/EVO 1-7
1.3.11 PC6/$\overline{\text{IRQ2}}$, PC7/$\overline{\text{IRQ1}}$ 1-7
1.3.12 BP0/PD0-BP3/PD3 1-7
1.3.13 FP32/PD4-FP35/PD7 1-7
1.3.14 FP24/PE0-FP31/PE7 1-7
1.3.15 FP16/PF0-FP23/PF7 1-8
1.3.16 FP8/PG0-FP15/PG7 1-8
1.3.17 FP0/PH0-FP7/PH7 1-8
1.3.18 VLCD3 1-8
1.3.19 BS1-BS3 1-8
1.3.20 DIN 1-8

SECTION 2
MEMORY

2.1 MEMORY MAP 2-1
2.2 ROM 2-2
2.3 RAM 2-2
2.4 I/O AND CONTROL REGISTERS 2-2

SECTION 3
CENTRAL PROCESSING UNIT

3.1 REGISTERS 3-1
3.2 ACCUMULATOR (A) 3-2
3.3 INDEX REGISTER (X) 3-2
3.4 STACK POINTER (SP) 3-2
3.5 PROGRAM COUNTER (PC) 3-3
3.6 CONDITION CODE REGISTER (CCR) 3-3
3.6.1 Half Carry Bit (H-Bit) 3-3
3.6.2 Interrupt Mask (I-Bit) 3-3

TABLE OF CONTENTS

Section	Title	Page
3.6.3	Negative Bit (N-Bit)	3-3
3.6.4	Zero Bit (Z-Bit)	3-4
3.6.5	Carry/Borrow Bit (C-Bit)	3-4

SECTION 4
INTERRUPTS

4.1	CPU INTERRUPT PROCESSING	4-1
4.2	RESET INTERRUPT SEQUENCE	4-3
4.3	SOFTWARE INTERRUPT (SWI)	4-3
4.4	HARDWARE INTERRUPTS	4-4
4.4.1	P-Decoder Interrupt (PDI)	4-4
4.4.2	$\overline{\text{IRQ1}}$ and $\overline{\text{IRQ2}}$	4-4
4.4.3	Key Wake-up Interrupt ($\overline{\text{KWI}}$)	4-6
4.4.4	Timer Interrupt	4-7
4.4.5	Serial Communication Interface (SCI)	4-7
4.4.6	Real Time Clock Interrupt (RTC)	4-7
4.4.7	Interrupt Control Register (INTCR)	4-7
4.4.8	Interrupt Status Register (INTSR)	4-8
4.4.9	Key Wake-Up Input Enable Register (KWIEN)	4-9

SECTION 5
RESETS

5.1	EXTERNAL RESET (RESET)	5-1
5.2	INTERNAL RESETS	5-2
5.2.1	Power-On Reset (POR)	5-2
5.2.2	Computer Operating Properly Reset (COPR)	5-2
5.2.3	Illegal Address Reset (ILADR)	5-2

SECTION 6
LOW POWER MODES

6.1	SINGLE-CHIP (NORMAL) MODE	6-1
6.2	SELF-CHECK MODE	6-1
6.3	LOW-POWER MODES	6-1
6.3.1	STOP Instruction	6-2
6.3.2	WAIT Instruction	6-2

SECTION 7
INPUT/OUTPUT PORTS

7.1	PORT A	7-1
7.2	PORT B	7-1
7.3	PORT C	7-2
7.4	PORT D	7-3
7.4.1	Port D MUX Register (PDMUX)	7-4

TABLE OF CONTENTS

Section	Title	Page
7.5	PORT E	7-4
7.5.1	Port E MUX Register (PEMUX)	7-5
7.6	PORT F, PORT G AND PORT H	7-5
7.6.1	Port F MUX Register (PFMUX)	7-5
7.6.2	Port G MUX Register (PGMUX)	7-6
7.6.3	Port H MUX Register (PHMUX)	7-6
7.7	INPUT/OUPUT PROGRAMMING	7-6
7.8	PORT OPTION CONTROL REGISTERS	7-7
7.8.1	Resistor Control Register 1 (RCR1)	7-7
7.8.2	Resistor Control Register 2 (RCR2)	7-8
7.8.3	Open Drain Output Control Register 1 (WOM1)	7-8
7.8.4	Open Drain Output Control Register 2 (WOM2)	7-9

SECTION 8
CLOCK DISTRIBUTION

8.1	OSC CLOCK DIVIDER AND POR COUNTER	8-1
8.2	SYSTEM CLOCK CONTROL	8-2
8.3	OSC AND XOSC	8-2
8.3.1	OSC On Line	8-2
8.3.2	XOSC On Line	8-3
8.4	TIME BASE	8-4
8.4.1	LCDCLK	8-5
8.4.2	STUP	8-5
8.4.3	Watchdog Timer (COP)	8-5
8.4.4	Time Base Control Register 1 (TBCR1)	8-6
8.4.5	Time Base Control Register 2 (TBCR2)	8-7
8.5	REAL TIME CLOCK (RTC)	8-7
8.5.1	RTC Time Registers	8-7
8.5.2	RTC Alarm Registers	8-9
8.5.3	RTC Status Register	8-9
8.5.4	RTC Control Register	8-10
8.6	MISCELLANEOUS REGISTER (MISC)	8-11

SECTION 9
TIMER SYSTEM

9.1	TIMER1	9-1
9.1.1	Counter	9-2
9.1.2	Output Compare Register	9-6
9.1.3	Input Capture Register	9-8
9.1.4	Timer Control Register (TCR)	9-10
9.1.5	Timer Status Register (TSR)	9-11
9.1.6	Operation During Low Power Mode	9-12
9.2	TIMER 2	9-12

TABLE OF CONTENTS

Section	Title	Page
9.2.1	Timer Control Register 2 (TCR2)	9-16
9.2.2	Timer Status Register 2 (TSR2)	9-17
9.2.3	Output Compare Register 2 (OC2)	9-18
9.2.4	Timer Counter 2 (CNT2)	9-18
9.2.5	Time Base Control Register 1 (TBCR1)	9-18
9.2.6	Timer Input 2 (EVI)	9-19
9.2.7	Event Output (EVO)	9-21
9.3	PRESCALER	9-23
	SECTION 10 LCD DRIVER	
10.1	LCD WAVEFORM EXAMPLES	10-1
10.2	VLCD3 BIAS INPUT & BIAS RESISTORS	10-4
10.3	BACKPLANE DRIVER AND PORT SELECTION	10-5
10.4	LCD CONTROL REGISTER (LCDCR)	10-5
10.5	LCD DATA REGISTER (LCDRx)	10-6
	SECTION 11 SERIAL COMMUNICATIONS INTERFACE	
11.1	SCI TWO-WIRE SYSTEM FEATURES	11-1
11.2	SCI RECEIVER FEATURES	11-1
11.3	SCI TRANSMITTER FEATURES	11-1
11.4	DATA FORMAT	11-3
11.5	WAKE-UP FEATURE	11-3
11.6	RECEIVE DATA IN (RDI)	11-4
11.7	START BIT DETECTION FOLLOWING A FRAMING ERROR	11-4
11.8	TRANSMIT DATA OUT (TDO)	11-6
11.9	SCI REGISTERS	11-6
11.9.1	Serial Communications Data Register (SCDAT)	11-6
11.9.2	Serial Communications Control Register 1 (SCCR1)	11-7
11.9.3	Serial Communications Control Register 2 (SCCR2)	11-8
11.9.4	Serial Communications Status Register (SCSR)	11-10
11.9.5	Baud Rate Register	11-12
	SECTION 12 P-DECODER	
12.1	P-DECODER FEATURES	12-1
12.2	CCIR Radiopaging Code No.1	12-1
12.2.1	Preamble	12-1
12.2.2	Batch Structure	12-2
12.2.3	Codeword	12-3
12.3	FUNCTIONAL BLOCKS	12-3
12.3.1	Clock Generator	12-5

TABLE OF CONTENTS

Section	Title	Page
12.3.2	BCH Corrector	12-5
12.3.3	Address Comparator	12-6
12.3.4	Battery Saving Generator	12-6
12.3.5	Mode Generator	12-7
12.3.6	Programming Mode	12-7
12.3.7	Waiting Mode	12-7
12.3.8	Frame State Generator	12-9
12.3.9	Preamble and Synchronization Codeword Detector	12-9
12.4	REGISTERS	12-10
12.4.1	P-Decoder Control Register 1 (PDCR1)	12-10
12.4.2	P-Decoder Control Register 2 (PDCR2)	12-11
12.4.3	P-decoder Control Register 3 (PDCR3)	12-11
12.4.4	P-Decoder Status Register (PDSR)	12-12
12.4.5	User Address Registers	12-13
12.4.6	Received Address Information Register (RAIR)	12-15
12.4.7	Received Message Information Registers (RMIRx)	12-15

SECTION 13
INSTRUCTION SET

13.1	ADDRESSING MODES	13-1
13.1.1	Inherent	13-1
13.1.2	Immediate	13-1
13.1.3	Direct	13-2
13.1.4	Extended	13-2
13.1.5	Indexed, No Offset	13-2
13.1.6	Indexed, 8-Bit Offset	13-2
13.1.7	Indexed, 16-Bit Offset	13-3
13.1.8	Relative	13-3
13.1.9	Instruction Types	13-3
13.1.10	Register/Memory Instructions	13-4
13.1.11	Read-Modify-Write Instructions	13-5
13.1.12	Jump/Branch Instructions	13-5
13.1.13	Bit Manipulation Instructions	13-7
13.1.14	Control Instructions	13-7
13.1.15	Instruction Set Summary	13-8

SECTION 14
ELECTRICAL SPECIFICATIONS

14.1	MAXIMUM RATINGS	14-1
14.2	THERMAL CHARACTERISTICS	14-1
14.3	DC ELECTRICAL CHARACTERISTICS	14-2
14.4	CONTROL TIMING	14-4

SECTION 15

TABLE OF CONTENTS

Section **Title** **Page**

MECHANICAL SPECIFICATIONS

15.1 80-PIN THIN-QUAD-FLAT-PACKAGE (Case 917-01) 15-2
15.2 80-PIN QUAD-FLAT-PACKAGE (Case 841B-01)..... 15-3

APPENDIX A
MC68HC705PD6

A.1 INTRODUCTION.....A-1
A.2 MEMORY.....A-1
A.3 MASK OPTION REGISTER (MOSR), $000F A-1
A.4 BOOTLOADER MODE A-2
A.5 EPROM PROGRAMMING.....A-3
A.5.1 EPROM Program Control Register (PCR).....A-3
A.5.2 Programming Sequence.....A-4
A.6 EPROM PROGRAMMING SPECIFICATIONS.....A-4

LIST OF FIGURES

Figure	Title	Page
1-1	MC68HC05PD6 Block Diagram	1-3
1-2	80-Pin TQFP Pin Assignment	1-4
1-3	OSC Connections	1-5
1-4	XOSC Connections	1-6
2-1	MC68HC05PD6 Memory Map	2-1
2-2	MC68HC05PD6 Main I/O Register $00-$0F (OPTM = 0)	2-3
2-3	MC68HC05PD6 Main I/O Register $10-$1F (OPTM =0)	2-4
2-4	MC68HC05PD6 Main I/O Register $20-$2F (OPTM = 0)	2-5
2-5	MC68HC05PD6 Main I/O Register $30-$3F (OPTM = 0)	2-6
2-6	MC68HC05PD6 Option I/O Register $00-$0F (OPTM = 1)	2-7
2-7	MC68HC05PD6 Option I/O Register $10-$1F (OPTM = 1)	2-8
2-8	MC68HC05PD6 Option I/O Register $20-$2F (OPTM = 1)	2-9
2-9	MC68HC05PD6 Option I/O Register $30-$3F (OPTM = 1)	2-10
3-1	MC68HC05 Programming Model	3-1
4-1	Interrupt Processing Flowchart	4-3
4-2	External Interrupt	4-5
4-3	Key Wake-up Interrupt (KWI)	4-6
5-1	Reset Block Diagram	5-1
6-1	Clock State and STOP Recovery/POR Delay Diagrams	6-3
6-2	STOP/WAIT Flowcharts	6-4
7-1	Port I/O Circuitry	7-7
8-1	Clock Signal Distribution	8-1
8-2	Time Base Clock Divider	8-5
8-3	RTC Holding Register	8-8
9-1	Timer System Block Diagram	9-1
9-2	Timer 1 Block Diagram	9-3
9-3	Timer State Timing Diagram for Reset	9-4
9-4	Timer State Timing Diagram for Timer Overflow	9-4
9-5	Timer State Timing Diagram For Output Compare	9-7
9-6	Timer State Timing Diagram For Input Capture	9-9
9-7	Timer 2 Block Diagram	9-13
9-8	Timer 2 Timing Diagram for f(PH2) > f(TIMCLK)	9-14
9-9	Timer 2 Timing Diagram for f(PH2) = f(TIMCLK)	9-15
9-10	EVI Block Diagram	9-19
9-11	EVI Timing Diagram	9-21
9-12	EVO Block Diagram	9-22
9-13	EVO Timing Diagram	9-23
9-14	Prescaler Block Diagram	9-24
10-1	LCD 1/3 Duty and 1/3 Bias Timing Diagram	10-2
10-2	LCD 1/4 Duty and 1/4 Bias Timing Diagram	10-3
10-3	Simplified LCD Voltage Divider Schematic	10-4
11-1	Serial Communications Interface Block Diagram	11-2
11-2	Data Format	11-3

LIST OF FIGURES

Figure	Title	Page
11-3	Sampling Technique used on All bits	11-4
11-4	Example of Start-Bit Sampling Technique	11-5
11-5	SCI Artificial Start following a Framing Error	11-5
11-6	SCI Start following a Break	11-5
11-7	Rate Generator Division	11-13
12-1	CCIR Radiopaging Code No.1 Format	12-2
12-2	P-Decoder Block Diagram	12-4
12-3	BCH Decoder Flow Chart	12-5
12-4	Mode Transition Diagram	12-7
12-5	Battery Saving Signals In Waiting Mode	12-8
12-6	P-Decoder Flags Timing	12-17
12-7	Receiving Mode Timing	12-18
15-1	80-Pin TQFP Mechanical Dimensions	15-2
15-2	80-Pin QFP Mechanical Dimensions	15-3
A-1	MC68HC705PD6 Memory Map	A-2
A-2	EPROM Programming Sequence	A-5

LIST OF TABLES

Table	Title	Page
4-1	Vector Address for Interrupts and Reset	4-2
6-1	Operating Mode Initialization	6-1
7-1	I/O Pin Functions	7-7
8-1	System Clock Frequencies	8-2
8-2	Recovery Time Requirements	8-4
8-3	COP Time Out Period	8-6
9-1	EVI Mode Select	9-20
9-2	CLK2 Divide Ratio	9-24
10-1	Backplanes and Port Selection	10-5
10-2	LCD Bias Resistors	10-6
11-1	Prescaler Highest Baud Rate Frequency Output	11-13
11-2	Transmit Baud Rate Output For a Given Prescaler Output	11-14
12-1	Timing Of Battery Saving Signals	12-8
12-2	Frame Number Definition	12-11
13-1	Register/Memory Instructions	13-4
13-2	Read-Modify-Write Instructions	13-5
13-3	Jump and Branch Instructions	13-6
13-4	Bit Manipulation Instructions	13-7
13-5	Control Instructions	13-7
13-6	Instruction Set Summary	13-8
13-7	Opcode Map	13-14
14-1	Maximum Ratings	14-1
14-2	Thermal Characteristics	14-1
14-3	DC Electrical Characteristics (5V)	14-2
14-4	DC Electrical Characteristics (3.6V)	14-3
14-5	Control Timing (5V)	14-4
14-6	Control Timing (3.6V)	14-5
A-1	Operating Mode Initialization	A-3
A-2	EPROM Programming Electrical Characteristics	A-4

LIST OF TABLES

Table **Title** **Page**

SECTION 1
GENERAL DESCRIPTION

The MC68HC05PD6 is a member of the MC68HC05 family of HCMOS Microcontroller Units (MCUs). This sophisticated 80-pin MCU has 16k-bytes of user ROM, 512 bytes of RAM, and eight parallel ports. Ports A, B, C, F, G, and H have eight I/O pins, port D and E have 8 output-only pins. The MC68HC05PD6 includes a Time Base circuit, 8 and 16-bit timers, a COP Watchdog timer, LCD drivers, serial communication interface and P-Decoder. It is targeted for communication applications such as pagers.

1.1 FEATURES

- Industry standard M68HC05 8-bit CPU core
- 16400 bytes of user ROM
- 512 bytes of user RAM (64 bytes for stack)
- 48 bidirectional I/O lines
- 16 output-only lines
- 16-bit timer with Input Capture and Output Compare functions
- COP Watchdog timer
- Serial Communication Interface (SCI)
- LCD drivers (3 or 4 backplane drivers) x (1 to 36 frontplane drivers)
- On-chip Time Base circuits
- Dual oscillators (76.8kHz and 4MHz) and selectable system clock frequency
- 24-hour Real Time Clock
- 8-bit Event counter / Modulus clock divider
- Key Wake-up Interrupt with 8-bit input
- P-Decoder
- Two IRQ inputs
- Available in 80-pin TQFP package

NOTE

A line over a signal name indicates an active low signal. 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 **Section 14**.

1.2 MASK OPTIONS

The following mask options are available:

- $\overline{\text{RESET}}$ pin pull-up resistor:
  [connected or disconnected
- OSC feedback resistor (between OSC1 and OSC2):
  [connected or disconnected]
- XOSC feedback resistor (between XOSC1 and XOSC2):
  [connected or disconnected]

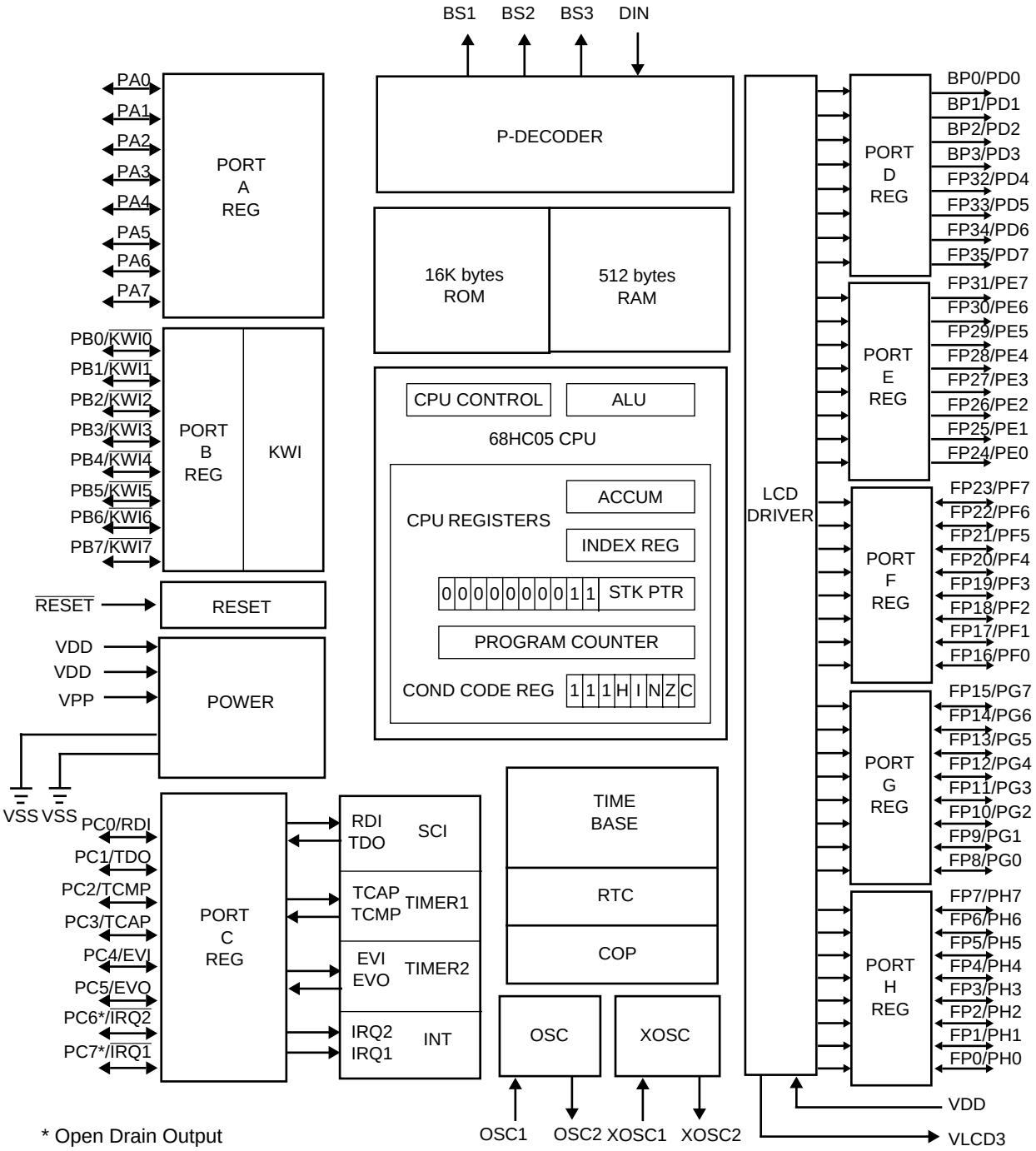


Figure 1-1. MC68HC05PD6 Block Diagram

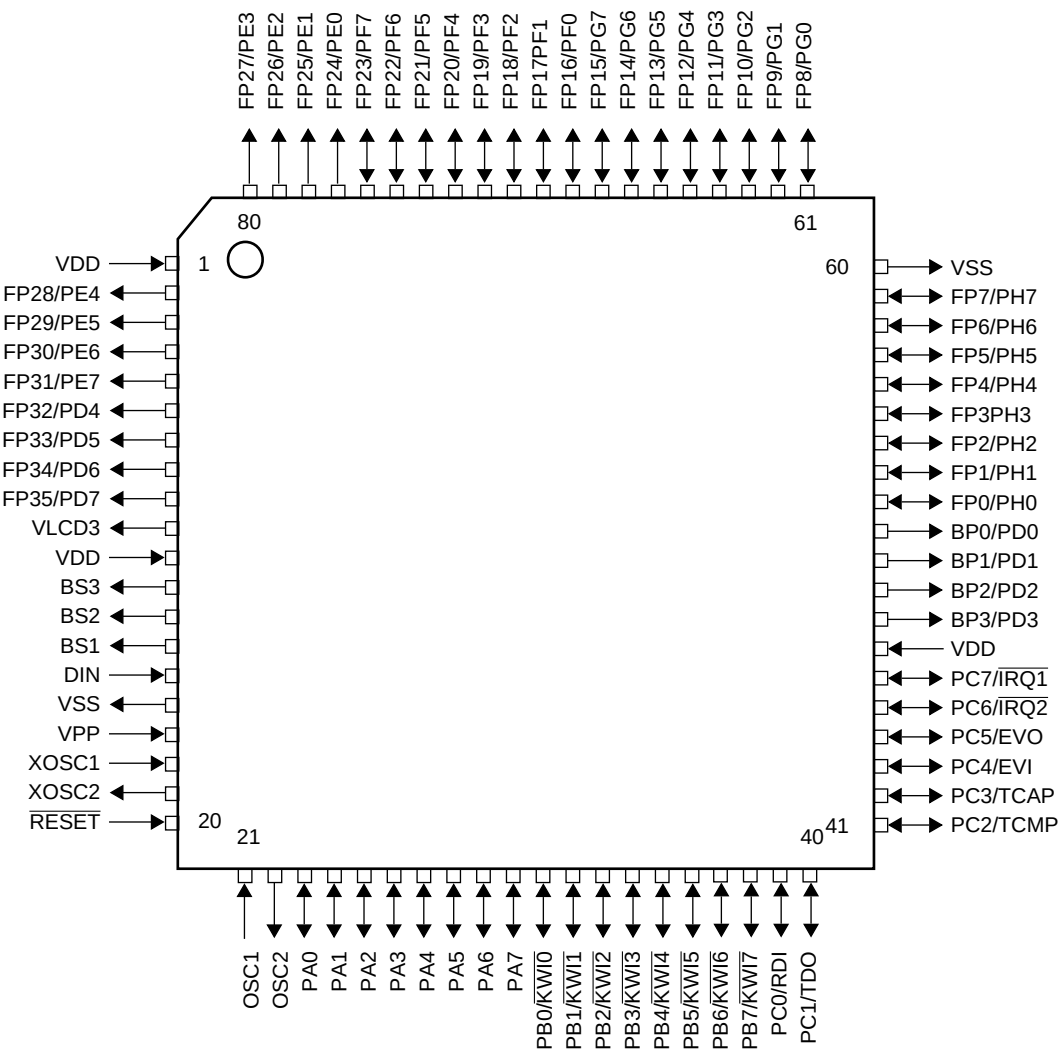


Figure 1-2. 80-Pin TQFP Pin Assignment

1.3 SIGNAL DESCRIPTION

1.3.1 VDD and VSS

Power is supplied to the microcontroller using these pins. VDD is the positive supply and VSS is ground pin. There are two pairs of power pins to improve noise immunity inside the chip.

1.3.2 OSC2, OSC1

These pins provide an on-chip clock oscillator circuit. A crystal, a ceramic resonator, or an external signal connects to these pins providing a system clock. The oscillator frequency divided by 2, 4, and 64 are available for the system clock. An external clock signal source should be connected to OSC1 while leaving OSC2 pin unconnected. 4 MHz is recommended at VDD=+5V.

1.3.3 XOSC2, XOSC1

XOSC has the same structure as OSC. It is the secondary oscillator run after Power-Up and can be selected as system clock instead of OSC. It also provides the clock source for P-Decoder. A 76.8KHz crystal is recommended.

The OSC1/XOSC1 and OSC2/XOSC2 pins are the connections for the on-chip oscillator. The pins can accept the following sets of components:

1. A crystal as shown in **Figure 1-3(a)** and **Figure 1-3(a)**

2. An external clock signal as shown in **Figure 1-4(b)** and **Figure 1-4(b)**

The circuits show in **Figure 1-3(a)** and **Figure 1-4(a)** are 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. OSC and XOSC pins have mask options for feedback and damping resistor implementations. See **Section 1.2** for these mask options and **Section 14** for the resistor values.

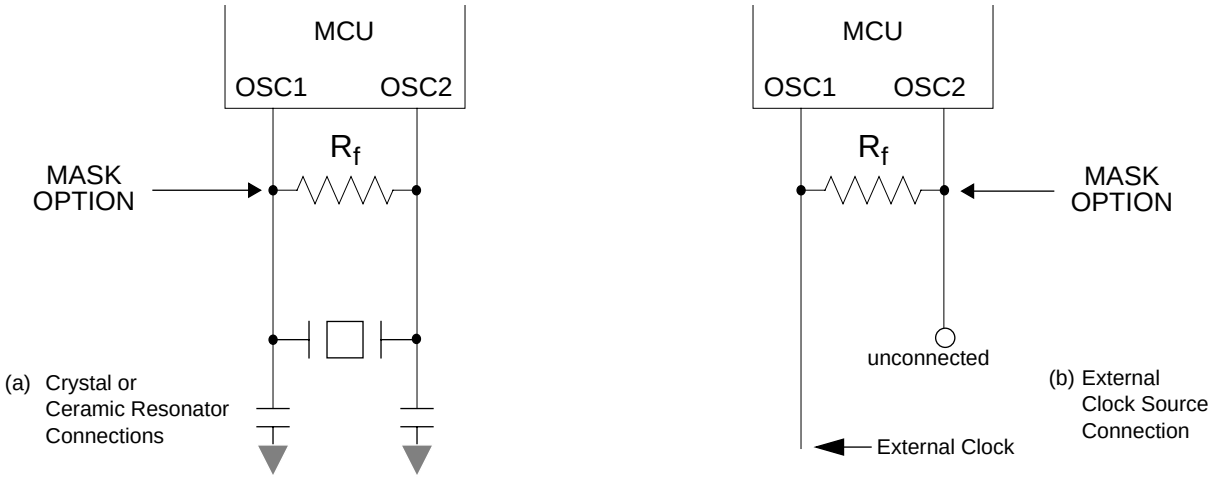


Figure 1-3. OSC Connections

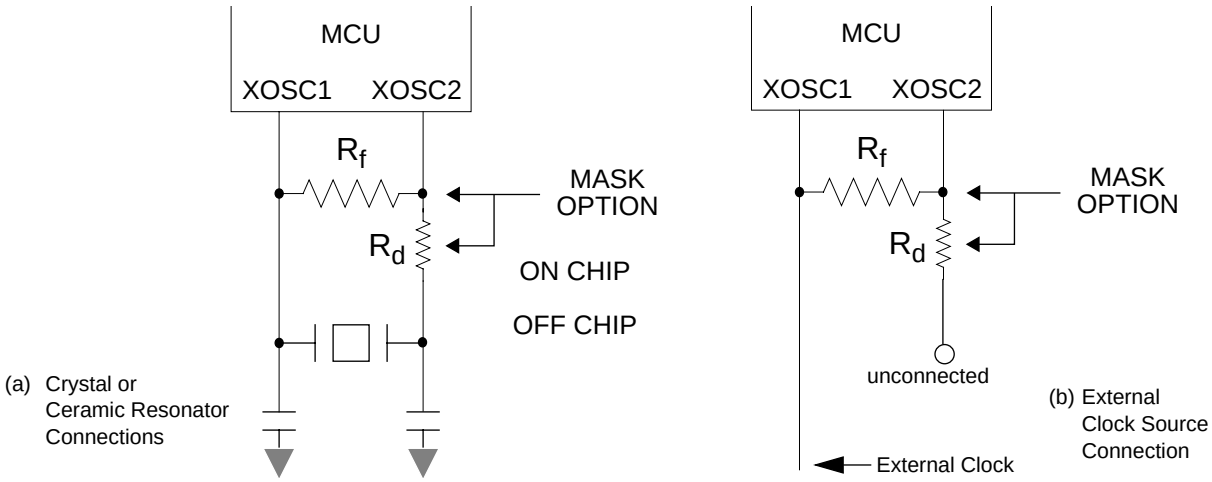


Figure 1-4. XOSC Connections

1.3.4 $\overline{\text{RESET}}$

This active low input-only pin is used to reset the MCU to a known start-up state. The $\overline{\text{RESET}}$ pin contains an internal Schmitt trigger as part of its input to improve noise immunity. See **Section 5** for more details.

1.3.5 VPP

This pin is used to supply high voltage needed for programming the user EPROM on the MC68HC705PD6 device. In normal operation, this pin should be connected to VDD.

1.3.6 PA0-PA7

These eight I/O lines comprise Port A. The state of any pin is software programmable and all Port A lines are configured as inputs during Reset. See **Section 7** for a detailed description of I/O programming.

1.3.7 PB0/$\overline{\text{KWI0}}$-PB7/$\overline{\text{KWI7}}$

These eight pins are either general purpose I/O pins port B or Key Wake-up interrupt input. See **Section 7** for a detailed description of I/O programming and **Section 4** for a detailed description of Interrupts.

1.3.8 PC0/RDI, PC1/TDO

These two pins are either general purpose I/O pins port C or the receive data

input, RDI, and transmit data output, TDO, of serial communication interface. See **Section 7** for a detailed description of I/O programming and **Section 11** for a detailed description of SCI.

1.3.9 PC2/TCMP, PC3/TCAP

These two pins are either the input capture, TCAP and output compare pins, TCMP, of Timer1 or general purpose I/O port C. See **Section 7** for a detailed description of I/O programming and **Section 9** for a detailed description of the Timer System.

1.3.10 PC4/EVI, PC5/EVO

These two pins are either the event counter input, EVI and output, EVO of Timer2 or general purpose I/O port C. See **Section 7** for a detailed description of I/O programming and **Section 9** for a detailed description of the Timer System.

1.3.11 PC6/$\overline{\text{IRQ2}}$, PC7/$\overline{\text{IRQ1}}$

These two pins are either the external interrupts input or general purpose I/O port C. The external interrupts input are software programmable for two different choices of interrupt triggering sensitivity. These options are: 1) negative edge-sensitive triggering only, or 2) both negative edge-sensitive and level-sensitive triggering. In the latter case, either type of input to the $\overline{\text{IRQ1}}$ or $\overline{\text{IRQ2}}$ pin will produce the interrupt. See **Section 7** for a detailed description of I/O programming and **Section 4** for a detailed description of Interrupts.

1.3.12 BP0/PD0-BP3/PD3

These four pins are either general purpose output pins port D or Backplane outputs (BP0-BP3) of LCD driver. See **Section 7** for a detailed description of I/O programming and **Section 10** for a detail description of LCD Drivers.

1.3.13 FP32/PD4-FP35/PD7

These four pins are either general purpose output pins port D or Frontplane outputs (FP32-FP35) of LCD driver.

1.3.14 FP24/PE0-FP31/PE7

These eight pins are either general purpose output pins port E or Frontplane outputs (FP24-FP31) of LCD driver.

1.3.15 FP16/PF0-FP23/PF7

These eight pins are either general purpose I/O pins port F or Frontplane outputs (FP23-FP16) of LCD driver.

1.3.16 FP8/PG0-FP15/PG7

These eight pins are either general purpose I/O pins port G or Frontplane outputs (FP8-FP15) of LCD driver.

1.3.17 FP0/PH0-FP7/PH7

These eight pins are either general purpose I/O pins port H or Frontplane outputs (FP0-FP7) of LCD driver.

1.3.18 VLCD3

VLCD3 provides the voltage reference for the LCD driver circuitry. See **Section 10** for a detail description of LCD driver.

1.3.19 BS1-BS3

These are the power saving pins output from P-Decoder.

1.3.20 DIN

This is the CCIR Radiopaging Code No.1 data input pin.

SECTION 2
MEMORY

The MC68HC05PD6 has 64k-bytes of addressable memory, consisting of 64 bytes of I/O, 512 bytes of user RAM, and 16384 bytes of user ROM, as shown in **Figure 2-1**.

2.1 MEMORY MAP

The MC68HC05PD6 Memory Map is shown in **Figure 2-1**.

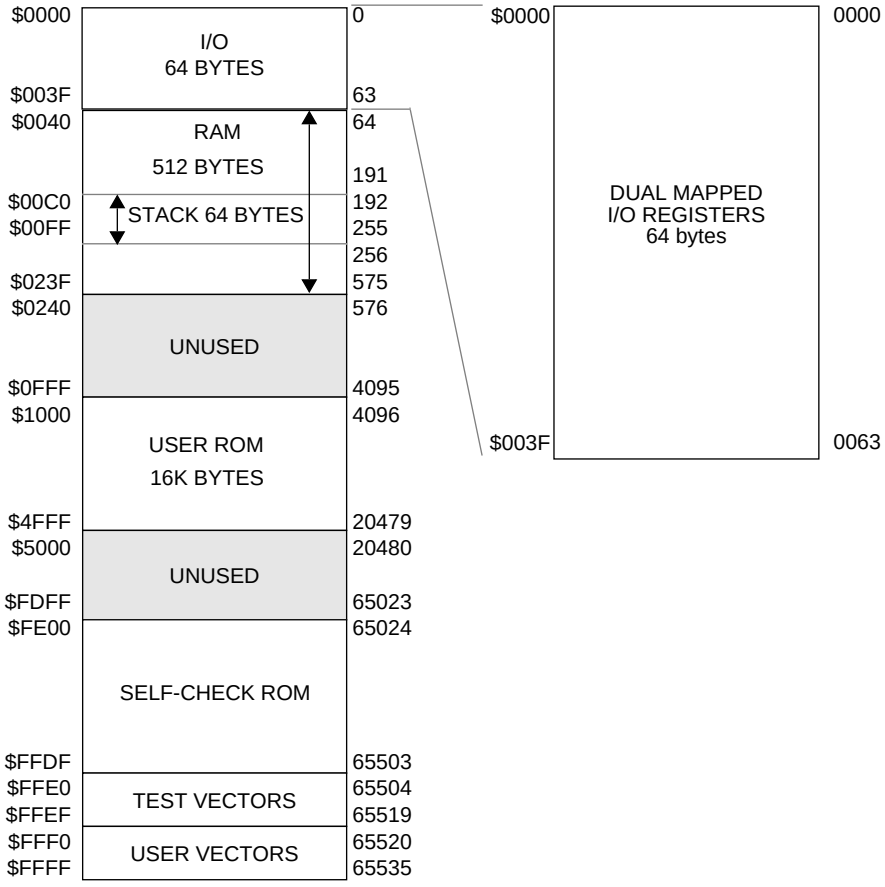


Figure 2-1. MC68HC05PD6 Memory Map

2.2 ROM

The user ROM consists of 16K bytes of ROM from $1000 through $4FFF and 16 bytes of user vectors from $FFF0 through $FFFF. The Self-Check ROM is located from $FE00 through $FFDF and Self-Check vectors are located from $FFE0 through $FFEF.

2.3 RAM

The user RAM consists of 512 bytes from $0040 to $023F. The stack pointer can access 64 bytes of RAM from $00FF to $00C0. Using the stack area for data storage or temporary work locations requires care to prevent it from being overwritten due to stacking from an interrupt or subroutine call.

2.4 I/O AND CONTROL REGISTERS

There are two I/O memory map – Main I/O map and Option I/O map. The Main I/O map is located at $0000-$003F and is accessible when OPTM bit in the MISC register ($003E) is clear. The Option Map is located at $0000-$003F and is accessible when OPTM bit in the MISC register ($003E) is set. **Figure 2-2** to **Figure 2-9** show the two I/O mappings.

ADDR	REGISTER	READ / WRITE	7	6	5	4	3	2	1	0
$0000	PORT A DATA PORTA	R	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
		W								
$0001	PORT B DATA PORTB	R	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
		W								
$0002	PORT C DATA PORTC	R	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
		W								
$0003	PORT D DATA PORTD	R	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
		W								
$0004	PORT E DATA PORTE	R	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0
		W								
$0005	PORT F DATA PORTF	R	PF7	PF6	PF5	PF4	PF3	PF2	PF1	PF0
		W								
$0006	PORT G DATA PORTG	R	PG7	PG6	PG5	PG4	PG3	PG2	PG1	PG0
		W								
$0007	PORT H DATA PORTH	R	PH7	PH6	PH5	PH4	PH3	PH2	PH1	PH0
		W								
$0008	INTERRUPT CONTROL REG INTCR	R	IRQ1E	IRQ2E	0	KWIE	IRQ1S	IRQ2S	0	0
		W								
$0009	INTERRUPT STATUS REG INTSR	R	IRQ1F	IRQ2F	0	KWIF	0	0	0	0
		W					RIRQ1	RIRQ2		RKWIF
$000A	BAUD RATE REG BRR	R	0	0	SCP1	SCP0	0	SCR2	SCR1	SCR0
		W								
$000B	SERIAL COM CONTROL REG 1 SCCR1	R	R8	T8		M	WAKE			
		W								
$000C	SERIAL COM CONTROL REG 2 SCCR2	R	TIE	TCIE	RIE	ILIE	TE	RE	RWU	WBK
		W								
$000D	SERIAL COM STATUS REG SCSR	R	TDRE	TC	RDRF	IDLE	OR	NF	FE	
		W								
$000E	SERIAL COM DATA REG SCDAT	R	SCD7	SCD6	SCD5	SCD4	SCD3	SCD2	SCD1	SCD0
		W								
$000F	KEY INT ENABLE REG KWIEN	R	KWIE7	KWIE6	KWIE5	KWIE4	KWIE3	KWIE2	KWIE1	KWIE0
		W								

UNIMPLEMENTED RESERVED

Figure 2-2. MC68HC05PD6 Main I/O Register $00-$0F (OPTM = 0)

ADDR	REGISTER	READ / WRITE	7	6	5	4	3	2	1	0
$0010	TIME BASE CONTROL REG 1 TBCR1	R	TBCLK	0	LCLK	0	0	0	T2R1	T2R0
		W								
$0011	TIME BASE CONTROL REG 2 TBCR2	R	0	0	RTR1	RTR0	0	0	COPE	0
		W								
$0012	TIMER CONTROL REG TCR	R	ICIE	OC1IE	TOIE	0	0	OE1	IEDG	OLVL
		W								
$0013	TIMER STATUS REG TSR	R	ICF	OC1F	TOF	0	0	0	0	0
		W								
$0014	INPUT CAPTURE REG H ICAPH	R	IC15	IC14	IC13	IC12	IC11	IC10	IC9	IC8
		W								
$0015	INPUT CAPTURE REG L ICAPL	R	IC7	IC6	IC5	IC4	IC3	IC2	IC1	IC0
		W								
$0016	OUTPUT COMPARE REG H OC1H	R	1OC15	1OC14	1OC13	1OC12	1OC11	1OC10	1OC9	1OC8
		W								
$0017	OUTPUT COMPARE REG L OC1L	R	1OC7	1OC6	1OC5	1OC4	1OC3	1OC2	1OC1	1OC0
		W								
$0018	COUNTER REG H CNTH	R	CNT15	CNT14	CNT13	CNT12	CNT11	CNT10	CNT9	CNT8
		W								
$0019	COUNTER REG L CNTL	R	CNT7	CNT6	CNT5	CNT4	CNT3	CNT2	CNT1	CNT0
		W								
$001A	COUNTER ALTERNATE REG H ACNTH	R	ACNT15	ACNT14	ACNT13	ACNT12	ACNT11	ACNT10	ACNT9	ACNT8
		W								
$001B	COUNTER ALTERNATE REG L ACNTL	R	ACNT7	ACNT6	ACNT5	ACNT4	ACNT3	ACNT2	ACNT1	ACNT0
		W								
$001C	TIMER CONTROL REG 2 TCR2	R	TI2IE	OC2IE	0	T2CLK	IM2	IL2	OE2	OL2
		W								
$001D	TIMER STATUS REG 2 TSR2	R	TI2F	OC2F	0	0	RTI2F	ROC2F	0	0
		W								
$001E	OUTPUT COMPARE REG 2 OC2	R	2OC7	2OC6	2OC5	2OC4	2OC3	2OC2	2OC1	2OC0
		W								
$001F	TIMER COUNTER REG 2 CNT2	R	2CNT7	2CNT6	2CNT5	2CNT4	2CNT3	2CNT2	2CNT1	2CNT0
		W	Timer Reset to $01							

UNIMPLEMENTED RESERVED

Figure 2-3. MC68HC05PD6 Main I/O Register $10-$1F (OPTM =0)

ADDR	REGISTER	READ WRITE	7	6	5	4	3	2	1	0
$0020	LCD CONTROL REG LCDCR	R	LCDE	0	DUTYS	DUTYOF	0	0	FC	LC
		W								
$0021	LCD REGISTER 1 LCDR1	R	F1B3	F1B2	F1B1	F1B0	F0B3	F0B2	F0B1	F0B0
		W								
$0022	LCD REGISTER 2 LCDR2	R	F3B3	F3B2	F3B1	F3B0	F2B3	F2B2	F2B1	F2B0
		W								
$0023	LCD REGISTER 3 LCDR3	R	F5B3	F5B2	F5B1	F5B0	F4B3	F4B2	F4B1	F4B0
		W								
$0024	LCD REGISTER 4 LCDR4	R	F7B3	F7B2	F7B1	F7B0	F6B3	F6B2	F6B1	F6B0
		W								
$0025	LCD REGISTER 5 LCDR5	R	F9B3	F9B2	F9B1	F9B0	F8B3	F8B2	F8B1	F8B0
		W								
$0026	LCD REGISTER 6 LCDR6	R	F11B3	F11B2	F11B1	F11B0	F10B3	F10B2	F10B1	F10B0
		W								
$0027	LCD REGISTER 7 LCDR7	R	F13B3	F13B2	F13B1	F13B0	F12B3	F12B2	F12B1	F12B0
		W								
$0028	LCD REGISTER 8 LCDR8	R	F15B3	F15B2	F15B1	F15B0	F14B3	F14B2	F14B1	F14B0
		W								
$0029	LCD REGISTER 9 LCDR9	R	F17B3	F17B2	F17B1	F17B0	F16B3	F16B2	F16B1	F16B0
		W								
$002A	LCD REGISTER 10 LCDR10	R	F19B3	F19B2	F19B1	F19B0	F18B3	F18B2	F18B1	F18B0
		W								
$002B	LCD REGISTER 11 LCDR11	R	F21B3	F21B2	F21B1	F21B0	F20B3	F20B2	F20B1	F20B0
		W								
$002C	LCD REGISTER 12 LCDR12	R	F23B3	F23B2	F23B1	F23B0	F22B3	F22B2	F22B1	F22B0
		W								
$002D	LCD REGISTER 13 LCDR13	R	F25B3	F25B2	F25B1	F25B0	F24B3	F24B2	F24B1	F24B0
		W								
$002E	LCD REGISTER 14 LCDR14	R	F27B3	F27B2	F27B1	F27B0	F26B3	F26B2	F26B1	F26B0
		W								
$002F	LCD REGISTER 15 LCDR15	R	F29B3	F29B2	F29B1	F29B0	F28B3	F28B2	F28B1	F28B0
		W								

UNIMPLEMENTED RESERVED

Figure 2-4. MC68HC05PD6 Main I/O Register $20-$2F (OPTM = 0)

ADDR	REGISTER	READ / WRITE	7	6	5	4	3	2	1	0
$0030	LCD REGISTER 16 LCDR16	R	F31B3	F31B2	F31B1	F31B0	F30B3	F30B2	F30B1	F30B0
		W								
$0031	LCD REGISTER 17 LCDR17	R	F33B3	F33B2	F33B1	F33B0	F32B3	F32B2	F32B1	F32B0
		W								
$0032	LCD REGISTER 18 LCDR18	R	F35B3	F35B2	F35B1	F35B0	F34B3	F34B2	F34B1	F34B0
		W								
$0033	RTC HOUR REG HOUR	R	0	0	0	HOUR4	HOUR3	HOUR2	HOUR1	HOUR0
		W								
$0034	RTC MINUTE REG MIN	R	0	0	MIN4	MIN4	MIN3	MIN2	MIN1	MIN0
		W								
$0035	RTC SECOND REG SEC	R	0	0	SEC5	SEC4	SEC3	SEC2	SEC1	SEC0
		W								
$0036	RTC HOUR ALARM REG HOURA	R	0	0	0	HOURA4	HOURA3	HOURA2	HOURA1	HOURA0
		W								
$0037	RTC MINUTE ALARM REG MINA	R	0	0	MINA5	MINA4	MINA3	MINA2	MINA1	MINA0
		W								
$0038	RTC SECOND ALARM REG SECA	R	0	0	SECA5	SECA4	SECA3	SECA2	SECA1	SECA0
		W								
$0039	RTC STATUS REG RTCS	R	0	0	0	0	0	SECF	ALF	RTCF
		W								
$003A	RTC CONTROL REG RTCC	R	0	0	0	0	0	SECE	ALE	RTCE
		W								
$003B	UNIMPLEMENTED	R								
		W								
$003C	UNIMPLEMENTED	R								
		W								
$003D	EPROM CONTROL REG PCR	R								
		W								
$003E	MISCELLEOUS REG MISC	R	FTUP	STUP	0	0	SYS1	SYS0	FOSCE	OPTM
		W								
$003F	TEST REG TEST	R								
		W								

UNIMPLEMENTED RESERVED

Figure 2-5. MC68HC05PD6 Main I/O Register $30-$3F (OPTM = 0)

ADDR	REGISTER	READ / WRITE	7	6	5	4	3	2	1	0
$0000	PORT A DATA DIRECTION REG DDRA	R	DDRA7	DDRA6	DDRA5	DDRA4	DDRA3	DDRA2	DDRA1	DDRA0
		W								
$0001	PORT B DATA DIRECTION REG DDRB	R	DDRB7	DDRB6	DDRB5	DDRB4	DDRB3	DDRB2	DDRB1	DDRB0
		W								
$0002	PORT C DATA DIRECTION REG DDRC	R	DDRC7	DDRC6	DDRC5	DDRC4	DDRC3	DDRC2	DDRC1	DDRC0
		W								
$0003	PORT D MUX REGISTER PDMUX	R	PDM7	PDM6	PDM5	PDM4	0	0	0	0
		W								
$0004	PORT E MUX REGISTER PEMUX	R	PEM7	PEM6	PEM5	PEM4	PEM3	PEM2	PEM1	PEM0
		W								
$0005	PORT F DATA DIRECTION REG DDRF	R	DDRF7	DDRF6	DDRF5	DDRF4	DDRF3	DDRF2	DDRF1	DDRF0
		W								
$0006	PORT G DATA DIRECTION REG DDRG	R	DDRG7	DDRG6	DDRG5	DDRG4	DDRG3	DDRG2	DDRG1	DDRG0
		W								
$0007	PORT H DATA DIRECTION REG DDRH	R	DDRH7	DDRH6	DDRH5	DDRH4	DDRH3	DDRH2	DDRH1	DDRH0
		W								
$0008	RESISTOR CONTROL REG 1 RCR1	R	0	0	0	0	RBH	RBL	RAH	RAL
		W								
$0009	RESISTOR CONTROL REG 2 RCR2	R	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0
		W								
$000A	OPEN DRAIN CONTROL REG WOM1	R	DWOMH	DWOML	EWOMH	EWOML	0	0	AWOMH	AWOML
		W								
$000B	OPEN DRAIN CONTROL REG WOM2	R	1	1	CWOM5	CWOM4	CWOM3	CWOM2	CWOM1	CWOM0
		W								
$000C	PORT F MUX REGISTER PFMUX	R	PFM7	PFM6	PFM5	PFM4	PFM3	PFM2	PFM1	PFM0
		W								
$000D	PORT G MUX REGISTER PGMUX	R	PGM7	PGM6	PGM5	PGM4	PGM3	PGM2	PGM1	PGM0
		W								
$000E	PORT H MUX REGISTER PHMUX	R	PHM7	PHM6	PHM5	PHM4	PHM3	PHM2	PHM1	PHM0
		W								
$000F	MASK OPTION STATUS REG MOSR	R	RSTR	OSCR	XOSCR	0	0	0	0	0
		W								

UNIMPLEMENTED RESERVED

Figure 2-6. MC68HC05PD6 Option I/O Register $00-$0F (OPTM = 1)

ADDR	REGISTER	READ / WRITE	7	6	5	4	3	2	1	0
$0010	PD STATUS REGISTER PDSR	R	0	0	ERRF	SCF	DCF	DBF	MSGF	ADRF
		W								
$0011	PD CONTROL REGISTER 1 PDCR1	R	PDEN	0	0	PROGC	DCIE	DBIE	MSGIE	ADRIE
		W								
$0012	PD CONTROL REGISTER 2 PDCR2	R	0	0	0	0	0	FB2	FB1	FB0
		W								
$0013	PD CONTROL REGISTER 3 PDCR3	R	PLL1	PLL0	DS1	DS0	DPOL	FA2	FA1	FA0
		W								
$0014	USER ADDREESS A REG 0 ADRA0	R	A7	A6	A5	A4	A3	A2	A1	A0
		W								
$0015	USER ADDREESS A REG 1 ADRA1	R	A15	A14	A13	A12	A11	A10	A9	A8
		W								
$0016	USER ADDREESS A REG 2 ADRA2	R	AFNS	0	0	0	0	0	A17	A16
		W								
$0017	USER ADDREESS B REG 0 ADRB0	R	B7	B6	B5	B4	B3	B2	B1	B0
		W								
$0018	USER ADDREESS B REG 1 ADRB1	R	B15	B14	B13	B12	B11	B10	B9	B8
		W								
$0019	USER ADDREESS B REG 2 ADRB2	R	BFNS	0	0	0	0	0	B17	B16
		W								
$001A	USER ADDREESS C REG 0 ADRC0	R	C7	C6	C5	C4	C3	C2	C1	C0
		W								
$001B	USER ADDREESS C REG 1 ADRC1	R	C15	C14	C13	C12	C11	C10	C9	C8
		W								
$001C	USER ADDREESS C REG 2 ADRC2	R	CFNS	0	0	0	0	0	C17	C16
		W								
$001D	USER ADDREESS D REG 0 ADRD0	R	D7	D6	D5	D4	D3	D2	D1	D0
		W								
$001E	USER ADDREESS D REG 1 ADRD1	R	D15	D14	D13	D12	D11	D10	D9	D8
		W								
$001F	USER ADDREESS D REG 2 ADRD2	R	DFNS	0	0	0	0	0	D17	D16
		W								

UNIMPLEMENTED RESERVED

Figure 2-7. MC68HC05PD6 Option I/O Register $10-$1F (OPTM = 1)

ADDR	REGISTER	READ / WRITE	7	6	5	4	3	2	1	0
$0020	USER ADDREESS E REG 0 ADRE0	R	E7	E6	E5	E4	E3	E2	E1	E0
		W								
$0021	USER ADDRESS E REG 1 ADRE1	R	E15	E14	E13	E12	E11	E10	E9	E8
		W								
$0022	USER ADDRESS E REG 2 ADRE2	R	EFNS	0	0	0	0	0	E17	E16
		W								
$0023	USER ADDRESS F REG 0 ADRF0	R	F7	F6	F5	F4	F3	F2	F1	F0
		W								
$0024	USER ADDRESS F REG 1 ADRF1	R	F15	F14	F13	F12	F11	F10	F9	F8
		W								
$0025	USER ADDREESS F REG 2 ADRF2	R	FFNS	0	0	0	0	0	F17	F16
		W								
$0026	RECEIVED ADDR INFO REG RAIR	R	0	0	F1	F0	0	RA2	RA1	RA0
		W								
$0027	RECEIVED MSG INFO REG 0 RMIR0	R	RMI7	RMI6	RMI5	RMI4	RMI3	RMI2	RMI1	RMI0
		W								
$0028	RECEIVED MSG INFO REG 1 RMIR1	R	RMI15	RMI14	RMI13	RMI12	RMI11	RMI10	RMI9	RMI8
		W								
$0029	RECEIVED MSG INFO REG 2 RMIR2	R	0	0	0	0	RMI19	RMI18	RMI17	RMI16
		W								
$002A	UNIMPLEMENTED	R								
		W								
$002B	UNIMPLEMENTED	R								
		W								
$002C	UNIMPLEMENTED	R								
		W								
$002D	UNIMPLEMENTED	R								
		W								
$002E	UNIMPLEMENTED	R								
		W								
$002F	UNIMPLEMENTED	R								
		W								

UNIMPLEMENTED RESERVED

Figure 2-8. MC68HC05PD6 Option I/O Register $20-$2F (OPTM = 1)

ADDR	REGISTER	READ WRITE	7	6	5	4	3	2	1	0
$0030	UNIMPLEMENTED	R								
		W								
$0031	UNIMPLEMENTED	R								
		W								
$0032	UNIMPLEMENTED	R								
		W								
$0033	UNIMPLEMENTED	R								
		W								
$0034	UNIMPLEMENTED	R								
		W								
$0035	UNIMPLEMENTED	R								
		W								
$0036	UNIMPLEMENTED	R								
		W								
$0037	UNIMPLEMENTED	R								
		W								
$0038	UNIMPLEMENTED	R								
		W								
$0039	UNIMPLEMENTED	R								
		W								
$003A	UNIMPLEMENTED	R								
		W								
$003B	UNIMPLEMENTED	R								
		W								
$003C	UNIMPLEMENTED	R								
		W								
$003D	UNIMPLEMENTED	R								
		W								
$003E	MISCELLEOUS REG MISC	R	FTUP	STUP	0	0	SYS1	SYS0	FOSCE	OPTM
		W								
$003F	UNIMPLEMENTED	R								
		W								

UNIMPLEMENTED RESERVED

Figure 2-9. MC68HC05PD6 Option I/O Register $30-$3F (OPTM = 1)

SECTION 3
CENTRAL PROCESSING UNIT

The MC68HC05PD6 has a 64 kbytes 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.

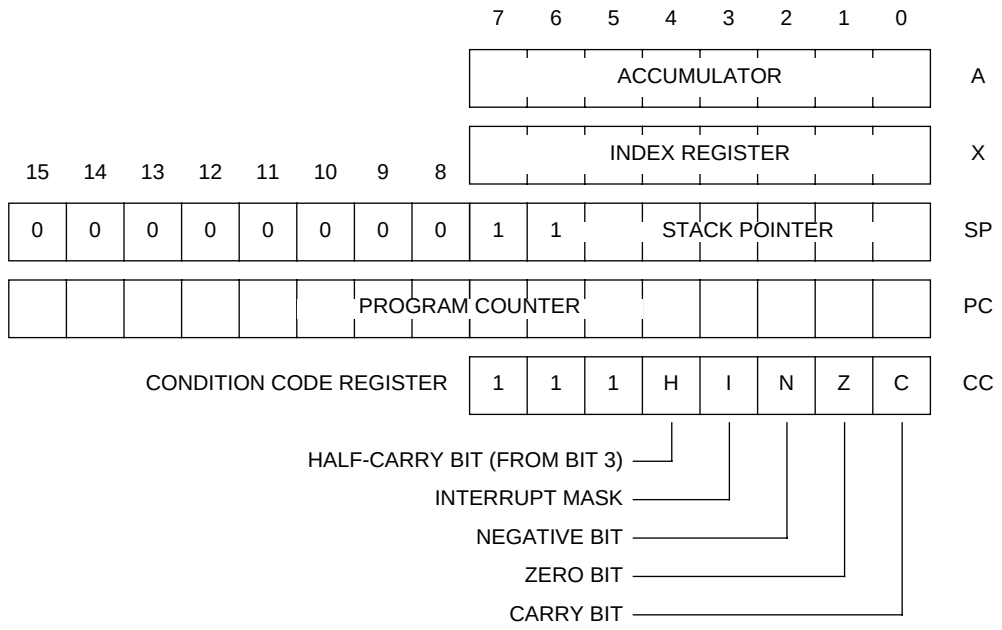


Figure 3-1. MC68HC05 Programming Model

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 64 Kbytes 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 64 Kbytes 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

The MCU can be interrupted in seven different ways:

- Non-maskable Software Interrupt Instruction (SWI)
- P-Decoder Interrupt
- External Interrupt ($\overline{IRQ}$)
- Key Wake-Up Interrupt ($\overline{KWI}$)
- Timer Interrupt
- SCI Interrupt
- RTC Interrupt

4.1 CPU INTERRUPT PROCESSING

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

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

When an interrupt is to be processed the CPU fetches the address of the appropriate interrupt software service routine from the vector table at locations $FFF0 thru $FFFF as defined in **Table 4-1**.

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

Table 4-1. Vector Address for Interrupts and Reset

Function	Source	Local Mask	Global Mask	Priority	Vector Address
Reset	Power-On Logic	None	None	1	$FFFE-$FFFF
	$\overline{\text{RESET}}$ Pin	None			
	COP Watchdog	COPE Bit			
SWI	User Code	None	None	Same Priority As Instruction	$FFFC-$FFFD
P-Decoder Interrupts	DCF Bit	DCIE	I Bit	2	$FFFA-$FFFB
	DBF Bit	DBIE			
	MSGF Bit	MSGIE			
	ADRF Bit	ADRIE			
External Interrupts	IRQ1F Bit	IRQ1E	I Bit	3	$FFF8-$FFF9
	IRQ2F Bit	IRQ2E			
Key Wake-up	KWIF Bit	KWIE	I Bit	4	$FFF6-$FFF7
Timer Interrupts	TI2F Bit	TI2IE	I Bit	5	$FFF4-$FFF5
	OC2F Bit	OC2IE			
	ICF Bit	ICIE			
	OC1F Bit	OC1IE			
	TOF Bit	TOIE			
SCI Interrupts	TDRE Bit	TIE	I Bit	6	$FFF2-$FFF3
	TC Bit	TCIE			
	RDRF Bit	RIE			
	IDLE Bit	ILIE			
RTC Interrupts	RTCF Bit	RTCE	I Bit	7	$FFF0-$FFF1
	ALF Bit	ALE			
	SECF Bit	SECE			

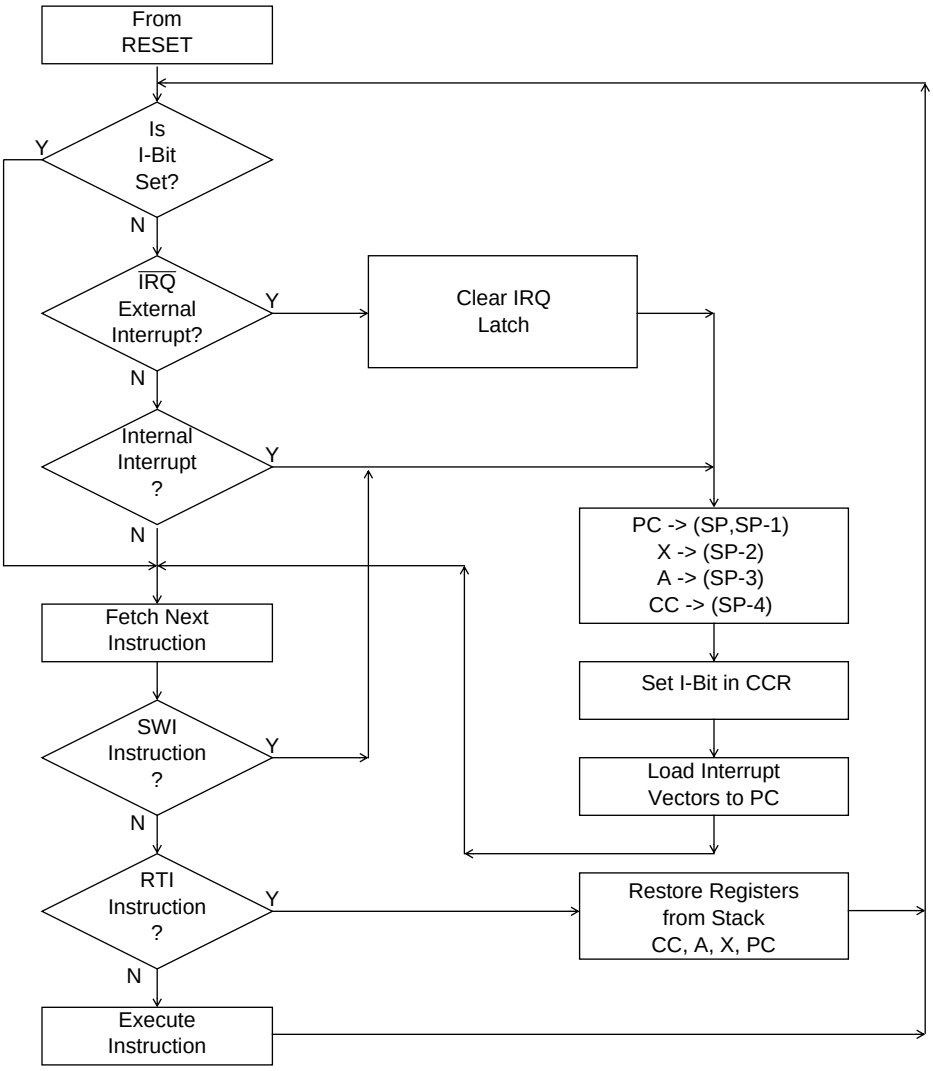


Figure 4-1. Interrupt Processing Flowchart

4.2 RESET INTERRUPT SEQUENCE

The RESET function is not in the strictest sense an interrupt; however, it is acted upon in a similar manner. A low level input on the $\overline{\text{RESET}}$ pin or an internally generated reset signal causes the program to vector to its starting address which is specified by the contents of $FFFE and $FFFF. The I-bit in the condition code register is also set. The MCU is configured to a known state during this type of reset as described in **Section 5**.

4.3 SOFTWARE INTERRUPT (SWI)

The SWI is an executable instruction and a non-maskable interrupt since it is executed regardless of the state of the I-bit in the CCR. If the I-bit is zero (interrupts enabled), the

SWI instruction executes after interrupts which were pending before the SWI was fetched, or before interrupts generated after the SWI was fetched. The interrupt service routine address is specified by the contents of $FFFC and $FFFD.

4.4 HARDWARE INTERRUPTS

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

4.4.1 P-Decoder Interrupt (PDI)

The P-Decoder interrupt vectors located at $FFFA and $FFFB. It contains four interrupt sources (DCI, DBI, MSGI, ADRI). These interrupts are generated only if the corresponding enable bit is set and the I bit of the CCR is cleared. See **Section 12** for more information on P-Decoder interrupts.

4.4.2 $\overline{\text{IRQ1}}$ and $\overline{\text{IRQ2}}$

Two external interrupt request inputs, $\overline{\text{IRQ1}}$ and $\overline{\text{IRQ2}}$ share the same vector address at $FFF8 and $FFF9.

If the IRQ option is edge and level sensitive triggering (IRQ*x*S=0), a low level at the $\overline{\text{IRQ}}$ pin and a cleared interrupt mask bit of the condition code register will cause an EXTERNAL INTERRUPT to occur. If the MCU has finished with the interrupt service routine, but the $\overline{\text{IRQ}}$ pin is still low, the EXTERNAL INTERRUPT will start again. In fact, the MCU will keep on servicing the EXTERNAL INTERRUPT as long as the $\overline{\text{IRQ}}$ pin is low. If the $\overline{\text{IRQ}}$ pin goes low for a while and resumes to high (a negative pulse) before the interrupt mask bit is cleared, the MCU will not recognize there was an interrupt request, and no interrupt will occur after the interrupt mask bit is cleared. IRQ*x*S is located in Interrupt Control Register (INTCR).

If the IRQ option is negative edge sensitive triggering (IRQ*x*S=1), a negative edge occurs at the $\overline{\text{IRQ}}$ pin and a cleared interrupt mask bit of the condition code register will cause an EXTERNAL INTERRUPT to occur. If the MCU has finished with the interrupt service routine, but the $\overline{\text{IRQ}}$ pin has not returned back to high, no further interrupt will be generated. The interrupt logic recognizes negative edge transitions and pulses (special case of negative edges) only. If the negative edge occurs while the interrupt mask bit is set, the interrupt signal will be latched, and interrupt will occur as soon as the interrupt mask bit is cleared. The latch will be cleared by RESET or cleared automatically during fetch of the EXTERNAL INTERRUPT vectors. Therefore, one (and only one) external interrupt edge could be latched while the interrupt mask bit is set.

The $\overline{\text{IRQ1}}$ and $\overline{\text{IRQ2}}$ are enabled by IRQ1E and IRQ2E bits and IRQ1F and IRQ2F bits are provided as an indicator in the Interrupt Status Register (INTSR). Since the IRQ1(2)F

can be set by either the pins or the data latches of PC7(6), be sure to clear the flags by software before setting the IRQ1(2)E bit.

The $\overline{\text{IRQ1}}$ and the $\overline{\text{IRQ2}}$ pins are shared with the Port C bit 7 and bit 6, respectively, and IRQ*x* pin states can be determined by reading Port C pins. The BIL and BIH instructions apply only to the $\overline{\text{IRQ1}}$ input.

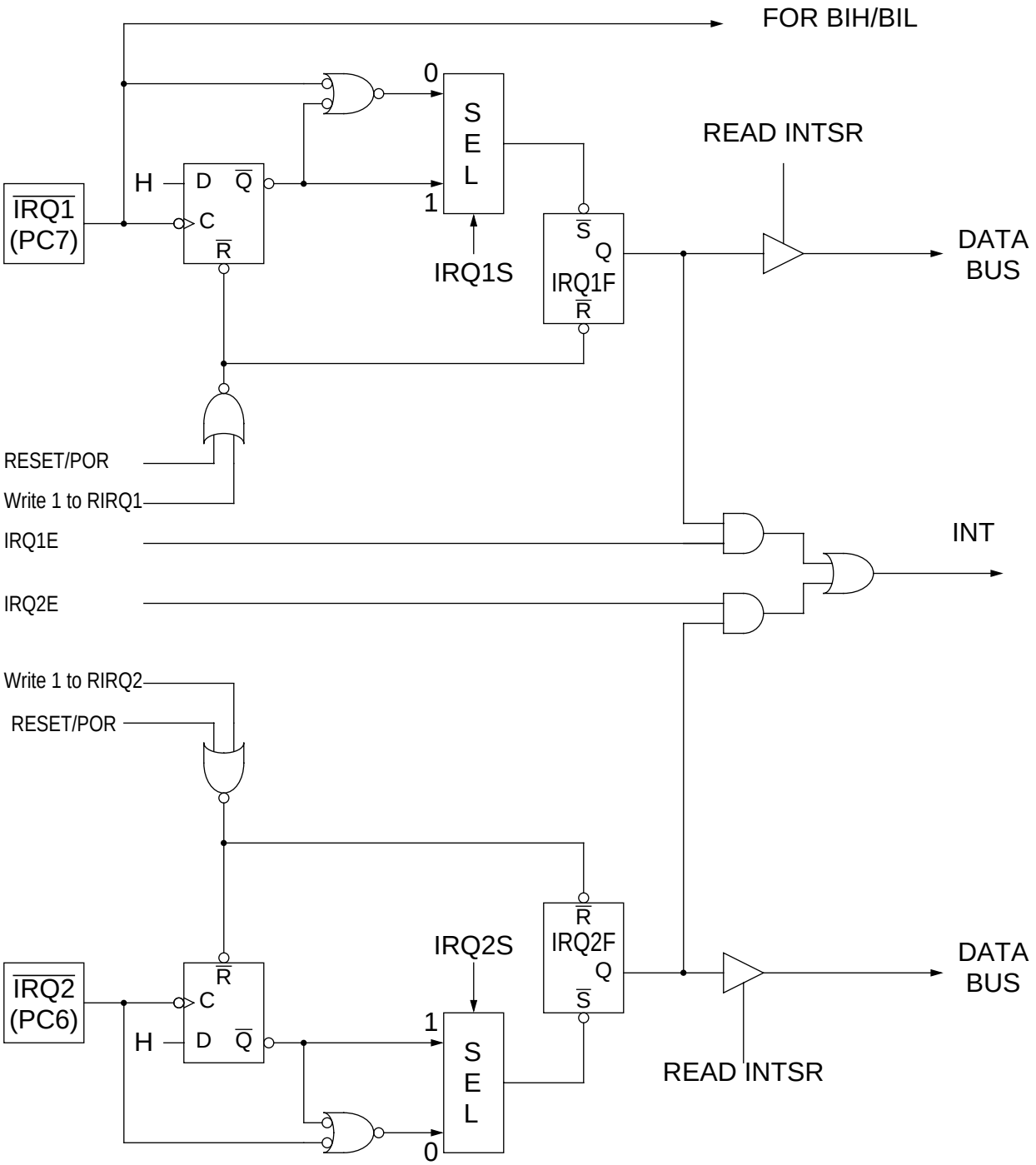


Figure 4-2. External Interrupt

4.4.3 Key Wake-up Interrupt ($\overline{\text{KWI}}$)

The eight Key Wake-up interrupt vectors located at $FFF6 and $FFF7.

These eight Key Wake-up inputs ($\overline{\text{KWI0}}$-$\overline{\text{KWI7}}$) share pins with Port B. Provided the I bit in CCR is cleared, each key wake-up input is enabled by the corresponding bit in the KWIEN register, and Key Wake-up Interrupt (KWI) is enabled by the KWIE bit in the INTCR.

When a falling edge is detected at one of the enabled key wake-up inputs, the KWIF bit in the INTSR is set and KWI is generated if KWIE = 1. Each input has a latch which responds only to the falling edge at the pin and all input latches are cleared at the same time by clearing KWIF bit (see **Figure 4-3**).

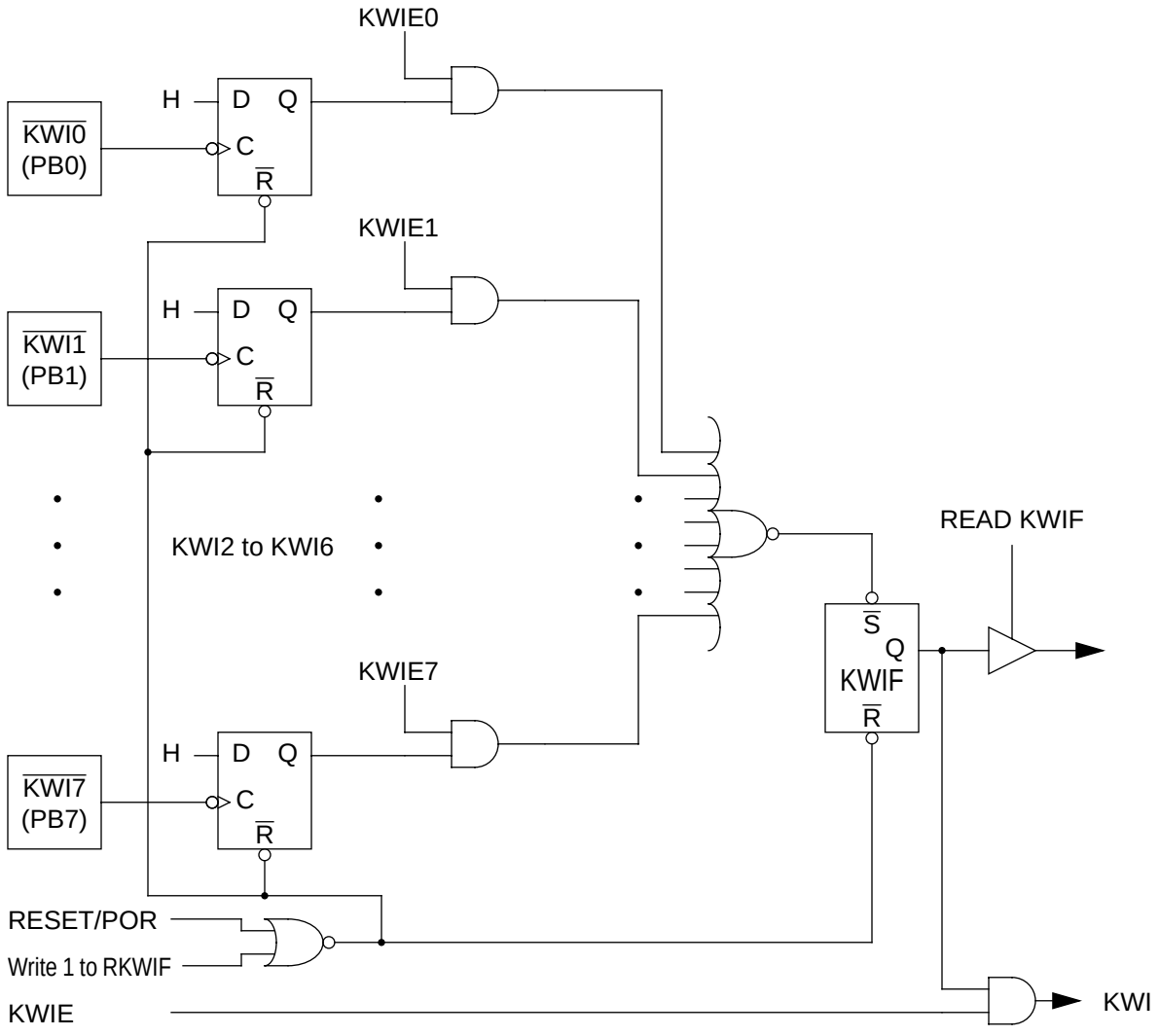


Figure 4-3. Key Wake-up Interrupt (KWI)

4.4.4 Timer Interrupt

Timer 1 and Timer 2 Interrupts share the same interrupt vector at $FFF4 and $FFF5. Timer 1 contains three interrupt sources (TOI, ICI and OC1I) and Timer 2 contains two interrupt sources (TI2I and OC2I). These interrupts are generated only if the corresponding enable bit is set and the I bit of the CCR is cleared. See **Section 9** for more information on timer interrupts.

4.4.5 Serial Communication Interface (SCI)

The SCI interrupt use the vector at $FFF2 and $FFF3. The SCI interrupt occurs when one of the interrupt flag bits in the serial communications status register is set, provided the I bit in the CCR is clear and the enable bit in the serial communications control register 2 is set. Software in the serial interrupt service routine must determine the cause and priority of the SCI interrupt by examining the interrupt flags and status bits in the SCI status register. See **Section 11** for more information on SCI interrupts.

4.4.6 Real Time Clock Interrupt (RTC)

The RTC interrupt is enabled when either RTCE, ALE or SECE bit in the control register is set, provided the interrupt mask bit of the CCR is cleared. The interrupt service routine address is specified by the contents of $FFF0 and $FFF1.

4.4.7 Interrupt Control Register (INTCR)

		7	6	5	4	3	2	1	0
INTCR $0008	R	IRQ1E	IRQ2E		KWIE	IRQ1S	IRQ2S		
	W								
reset ⇒		0	0	0	0	0	0	0	0

IRQ1E – Interrupt 1 Enable

IRQ1E bit enables IRQ1 interrupt when IRQ1F is set. This bit is cleared on reset.

0 = IRQ1 Interrupt is disabled
1 = IRQ1 Interrupt is enabled

IRQ2E – Interrupt 2 Enable

IRQ2E bit enables IRQ2 interrupt when IRQ2F is set. This bit is cleared on reset.

0 = IRQ2 Interrupt is disabled
1 = IRQ2 Interrupt is enabled

KWIE – Key Wake-up Enable

Key Wake-up Interrupt (KWI) Enable bit enables Key Wake-up Interrupt when KWIF is set. This bit is cleared on reset.

0 = KWI is disabled
1 = KWI is enabled

IRQ1S – IRQ1 Select Edge Sensitive Only

0 = IRQ1 is configured for low LEVEL and negative edge sensitive
1 = IRQ1 is configured to respond only to negative EDGEs

IRQ2S – IRQ2 Select Edge Sensitive Only

0 = IRQ2 is configured for low LEVEL and negative edge sensitive
1 = IRQ2 is configured to respond only to negative EDGEs

4.4.8 Interrupt Status Register (INTSR)

		7	6	5	4	3	2	1	0
INTSR $0009	R	IRQ1F	IRQ2F	0	KWIF	0	0	0	0
	W					RIRQ1	RIRQ2		RKWIF
reset ⇒		0	0	0	0	0	0	0	0

IRQ1F – IRQ1 Interrupt Flag

When IRQ1S = 0, the falling edge or low level at $\overline{\text{IRQ1}}$ pin sets IRQ1F. When IRQ1S = 1, only the falling edge at pin sets IRQ1F bit. If IRQ1E bit and this bit are set, an interrupt is generated. This bit is read only bit and cleared by writing a 1 to the RIRQ1 bit. Reset clears this bit.

IRQ2F – IRQ2 Interrupt Flag

When IRQ2S = 0, the falling edge or low level at $\overline{\text{IRQ2}}$ pin sets IRQ2F. When IRQ2S = 1, only the falling edge at pin sets IRQ2F bit. If IRQ2E bit and this bit are set, an interrupt is generated. This bit is read only bit and cleared by writing a 1 to the RIRQ2 bit. Reset clears this bit.

KWIF – Key Wake-up Interrupt Flag

When KWIE*x* bit in the KWIEN register is set, the falling edge at KWI*x* pin sets KWIF bit. If KWIE bit and this bit are set, an interrupt is generated. This bit is a read only bit and clearing KWIF is accomplished by writing a 1 to the RKWIF bit. Reset clears this bit.

RIRQ1 – Reset IRQ1 Flag

The RIRQ1 bit is a write only bit and always read as 0. Writing a 1 to this bit clears the IRQ1F bit and writing 0 to this bit has no effect.

RIRQ2 – Reset IRQ2 Flag

The RIRQ2 bit is a write only bit and always read as 0. Writing a 1 to this bit clears the IRQ2F bit and writing a 0 to this bit has no effect.

RKWIF – Reset KWI Flag

The RKWIF bit is a write only bit and always read as 0. Writing a 1 to this bit clears the KWIF bit and writing a 0 to this bit has no effect.

4.4.9 KEY WAKE-UP INPUT ENABLE REGISTER (KWIEN)

Option map KWIEN $000F

	7	6	5	4	3	2	1	0
R	KWIE7	KWIE6	KWIE5	KWIE4	KWIE3	KWIE2	KWIE1	KWIE0
W								
reset ⇒	0	0	0	0	0	0	0	0

KWIE*x* – Key Wake-up Input Enable (bit *x*)

When KWIE*x* bit is set, the KWI*x* (PB*x*) input is enabled for Key Wake-up Interrupt. This bit is cleared on reset.

SECTION 5
RESETS

The MCU can be reset in four ways:

- by an active low input to the $\overline{\text{RESET}}$ pin,
- by initial power-on reset,
- by COP watchdog reset, and
- by an illegal address access.

The $\overline{\text{RESET}}$ pin is an I/O pin as shown in **Figure 5-1**. All the peripheral modules which drive external pins will be reset by the synchronous reset signal (RST) coming from a latch, which is synchronized to the internal bus clock and set by any of the four reset sources.

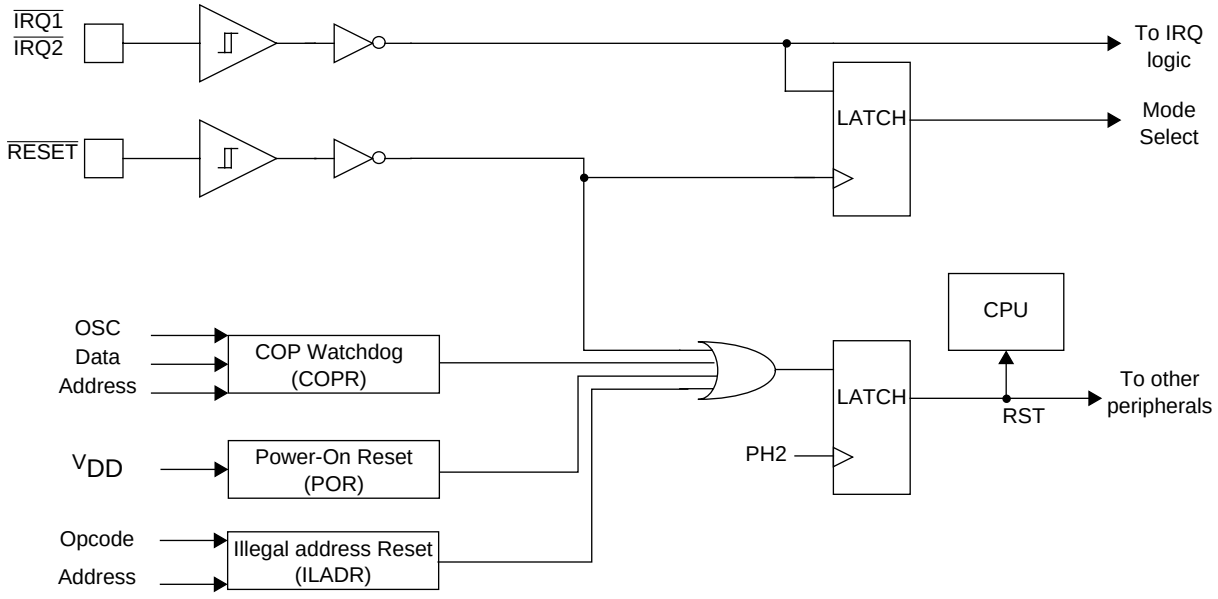


Figure 5-1. Reset Block Diagram

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

The $\overline{\text{RESET}}$ pin is the only external reset source. This pin is connected to a Schmitt trigger input gate to provide an upper and lower threshold voltage separated by a minimum amount of hysteresis. This external reset occurs whenever the $\overline{\text{RESET}}$ pin is pulled below

the lower threshold and remains in reset until the $\overline{\text{RESET}}$ pin rises above the upper threshold. This active low input will generate the RST signal and reset the CPU and peripherals. Termination of the external $\overline{\text{RESET}}$ input can alter the operating mode of the MCU.

NOTE

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

5.2 INTERNAL RESETS

The three internally generated resets are the initial power-on reset, the COP Watchdog Timer reset, and the illegal address reset

5.2.1 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-on condition and is not able to detect a drop in the power supply voltage (brown-out). There is an oscillator stabilization delay of 8072 OSC 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 this 8192 cycles delay, the RST signal will remain in the reset condition until the other reset conditions end.

5.2.2 Computer Operating Properly Reset (COPR)

The internal COPR reset is generated automatically (if enabled) by a time-out of the COP Watchdog Timer. This time-out occurs if the counter in the COP Watchdog Timer is not reset (cleared) within a specific time by a program reset sequence. See **Section 8** for more information on this time-out feature.

5.2.3 Illegal Address Reset (ILADR)

The MCU monitors all opcode fetches. If an illegal address is accessed during an opcode fetch, an internal reset is generated. Illegal address space consists of all unused locations within the memory space and the I/O registers. (See **Figure 2-1** for MC68HC05PD6 Memory Map.) Because the internal reset signal is used, the MCU comes out of an ILADR Reset in the same operating mode it was in when the opcode was fetched.

SECTION 6
LOW POWER MODES

The MC68HC05PD6 has two operating modes: Single-Chip (Normal) Mode, and Self-check Mode. The Single-Chip Mode is the normal operating mode for the MCU. The mode of operation is determined by the logic state on PC6 and PC7 pins, and the voltage on V_{PP} on the rising edge of the external $\overline{\text{RESET}}$ input.

Table 6-1. Operating Mode Initialization

MODE	$\overline{\text{RESET}}$	PC7/$\overline{\text{IRQ1}}$	PC6/$\overline{\text{IRQ2}}$	VPP
Single-Chip (Normal)	↑	V_{SS} to V_{DD}	V_{SS} to V_{DD}	V_{SS} to V_{DD}
Self-Check		V_{SS}	V_{DD}	V_{TST}

$V_{TST}=2 \times V_{DD}$

6.1 SINGLE-CHIP (NORMAL) MODE

The Single-Chip Mode allows the MCU to function as a self-contained microcontroller, with maximum use of the pins for on-chip peripheral functions.

In the Single-Chip Mode all address and data activity occurs within the MCU and is not available externally. Single-Chip Mode is entered if the VPP pin is within the normal operating voltage range when the rising edge of a $\overline{\text{RESET}}$. In Single-Chip Mode, all I/O port pins are available.

6.2 SELF-CHECK MODE

The self-check program is mask at location $FE00 to $FFDF, and is used for checking device functionality under minimum hardware support.

6.3 LOW-POWER MODES

In each of its configuration modes the MCU is capable of running in one of several low-power operating modes. The WAIT and STOP instructions provide two modes that reduce the power required for the MCU by stopping various internal clocks and/or the oscillator. The flow of the STOP, and WAIT modes are shown in **Figure 6-1**.

6.3.1 STOP Instruction

The STOP instruction places the MCU in its lowest power consumption mode. In STOP mode, the internal main oscillator OSC is turned off, halting all internal processing, including timer operations (Timer1, Timer2, and COP Watchdog timer). Sub oscillator XOSC does not stop oscillating. Therefore if XOSC is used as the clock source for COP, COP is still functional in STOP mode. See **Section 8** on Clock Distribution.

During the STOP mode, the TCR bits are altered to remove any pending timer interrupt request and to disable any further timer interrupts. The timer prescaler is cleared. The I bit in the CCR is cleared to enable external interrupts. All other registers and memory remain unaltered. All input/output lines remain unchanged. The processor can be brought out of the STOP mode only by $\overline{\text{RESET}}$ or interrupt from $\overline{\text{IRQ1}}$, $\overline{\text{IRQ2}}$, $\overline{\text{KWI}}$, PDI, or RTC.

6.3.2 WAIT Instruction

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

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

The reduction of power in the WAIT mode depends on how many of the on-chip peripheral's clock can be shut down. Therefore the amount of power that will be consumed is very dependent on the application and that it would be prohibitive to test all parts for all variations. For these reasons the data sheet will include values for a limited number of variations. These variations and the corresponding MAX power consumptions will be decided upon after initial characterization of silicon.

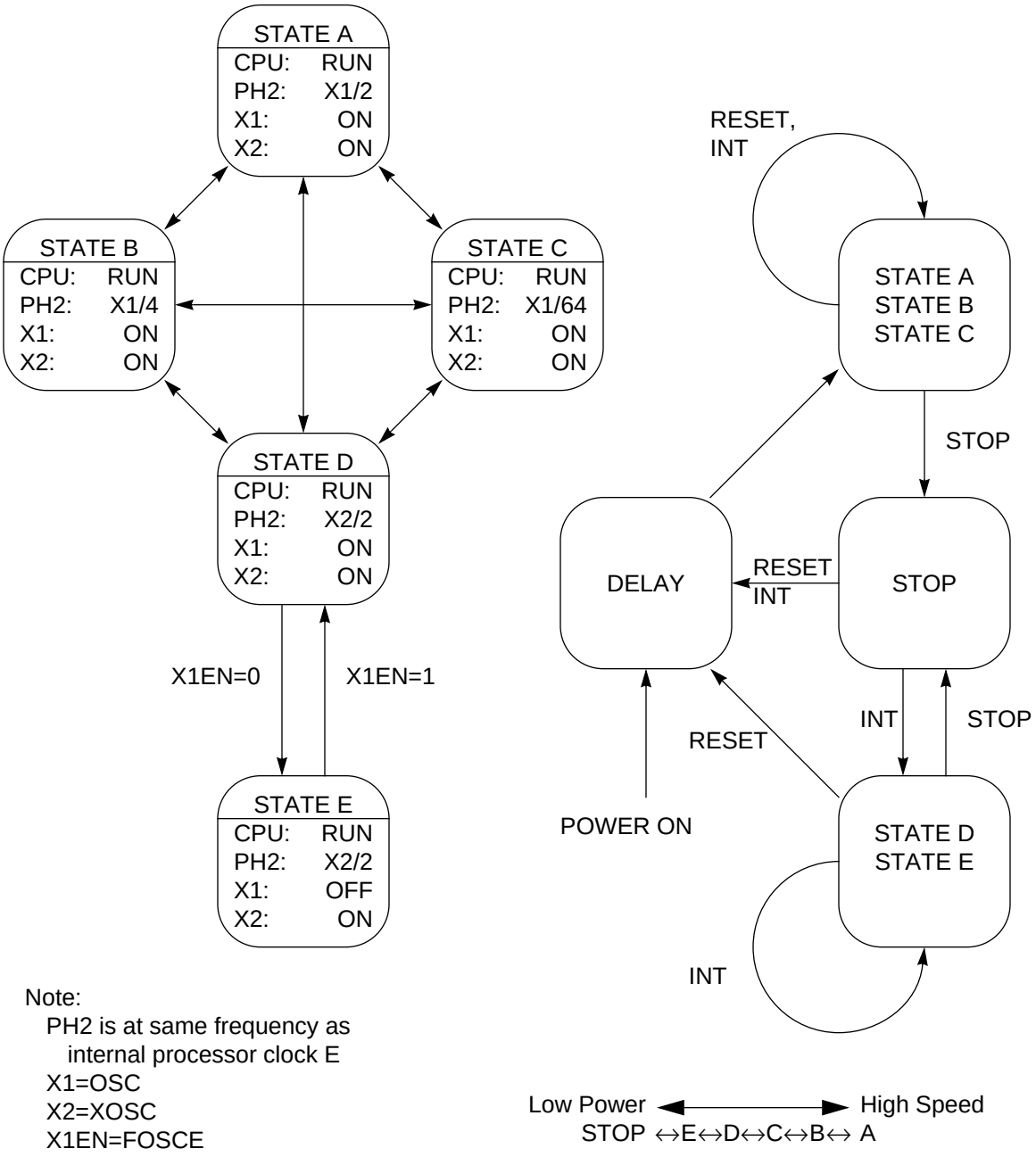


Figure 6-1. Clock State and STOP Recovery/POR Delay Diagrams

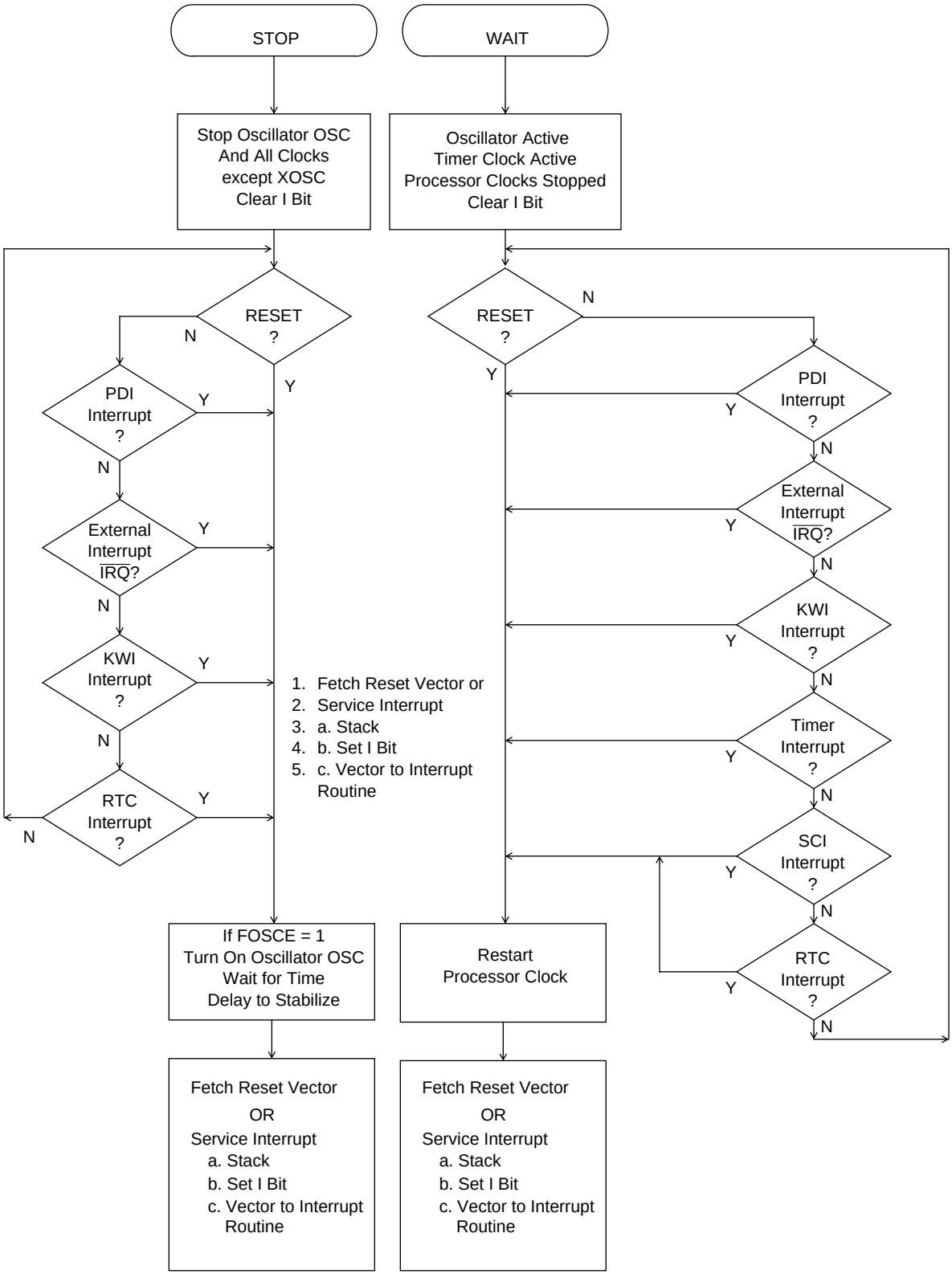


Figure 6-2. STOP/WAIT Flowcharts

SECTION 7
INPUT/OUTPUT PORTS

The MC68HC05PD6 a has 8 parallel ports A, B, C, D, E, F, G and H. Port A, B, C, F, G and H have 8 I/O pins, Port D and E have 8 output-only pins. The individual bits in these ports are programmable as either inputs or outputs under software control by the data direction registers (DDRs). All of these pins except Port A serve multiple purposes depending on the configuration of these pins.

7.1 PORT A

Port A is an 8-bit bidirectional general purpose port. The data direction of a Port A pin is determined by its corresponding DDRA bit.

When a Port A pin is programmed as an output by the corresponding DDRA bit, data in the PORTA data register becomes output data to the pin and this data is returned when PORTA register is read.

Open drain or CMOS outputs are selected by AWOMH and AWOML bits in the WOM1 register. If the AWOMH bit is set, the P-channel drivers of output buffers of bit 7 thru bit 4 are disabled (open drain). If the AWOML bit is set, the P-channel drivers of output buffers of bit 3 thru bit 0 are disabled (open drain).

When a pin is programmed as input by the corresponding DDRA bit, the pin level is read by the CPU.

Port A has an optional pull-up resistors. When the RAH or RAL bit in the RCR1 is set, pull-up resistors are attached to the upper 4 bits or lower 4 bits of Port A pins, respectively. (The typical resistor values are 200KΩ at V_{DD}=3V.) When a pin outputs a low level, the pull-up resistor is disconnected regardless of RAH or RAL bit state.

7.2 PORT B

Port B pins serve two basic functions: Key Wake-up Interrupt (KWI) input pins and general purpose I/O pins.

Each KWI input is enabled or disabled by the corresponding KWIE*x* bit in the KWIEN register, and the usage of the KWI input does not affect the general purpose input function. When any of these bits is used as Port B I/O, the

corresponding KWIE*x* bit in the KWIEN should be disabled, otherwise KWI interrupt will occur.

When a Port B pin is programmed as an output by the corresponding DDRB bit, data in the PORTB data register becomes output data to the pin and this data is returned when PORTB register is read.

Pull-up resistors are provided for both upper and lower 4 bits of Port B pins which are controlled by the RBH and RBL bits, respectively, in the RCR1 register. (The typical resistor values are 200KΩ at V_{DD}=3V.)

7.3 PORT C

Port C pins share functions with several on chip peripherals. A pin function is controlled by the enable bit of each associated peripheral.

Bit 7 and bit 6 of Port C are general purpose I/O pins and IRQ input pins. The DDRC7/6 bits determine whether the pin states or the data latch states should be read by the CPU. Since IRQ1(2)F can be set by either the pins or the data latches, when using IRQs, be sure to clear the flags by software before enabling the IRQ1(2)E bits.

The PC5 pin is a general purpose I/O pin and the direction of the pin is determined by the DDRC5 bit in the Data Direction Register C (DDRC). When the event output (EVO) is enabled, the PC5 is configured as an event output pin and the DDRC5 bit has meaning only for the read of PC5 bit in the PORTC register; if the DDRC5 is set the PC5 data latch is read by the CPU, otherwise PC5 pin level (EVO state) is read. When EVO is disabled, it becomes general purpose I/O pin, PC5. This PC5/EVO output has the capability to drive 10 mA source current when ($V_{OH} \geq$ Vdd–0.8V).

The PC4 and PC3 pins share functions with the Timer input pins (EVI and TCAP). These bits are not affected by the usage of timer input functions and the directions of pins are always controlled by the DDRC4 and DDRC3 bits. Also the DDRC4 and DDRC3 bits determine whether the pin states or data latch states should be read by the CPU.

NOTE

Since the TCAP pin is shared with the PC3 I/O pin, changing the state of the PC3 DDR or Data Register can cause an unwanted TCAP interrupt. This can be handled by clearing the ICIE bit before changing the configuration of PC3, and clearing any pending interrupts before enabling ICIE.

NOTE

Since the EVI pin is shared with the PC4 I/O pin, DDRC4 should always be cleared whenever EVI is used. EVI should not be used when DDRC4 is high.

The PC2 pin is a general purpose I/O pin and the direction of the pin is determined by the DDRC2 bit in the Data Direction Register C (DDRC). When the Output Compare (TCMP) is enabled, the PC2 is configured as TCMP output pin and the DDRC2 bit has meaning only for the read of PC2 bit in the PORTC register; if the DDRC2 is set the PC2 data latch is read by the CPU, otherwise PC2 pin level is read. When TCMP is disabled, it becomes general purpose I/O pin, PC2. This PC2/TCMP output has the capability to drive 10 mA source current when ($V_{OH} \geq$ Vdd–0.8V).

The PC1 thru PC0 pins are shared with the Serial Communication Interface (SCI). When the SCI is not used (TE, RE = 0), DDRC1 and DDRC0 bits control the directions of the pins, and when the SCI is enabled, the pins are configured as transmit data out (TDO) and receive data in (RDI). When the PORTC is read, the value read will be determined by the Data Direction Register. When the port is configured for input (DDRC1 or DDRC0 equal to 0) the pin state is read. When the port is configured for output (DDRC1 or DDRC0 equal to 0) output data latch is read.

Each Port C pin has pull-up resistor option which is controlled by the corresponding RCR2 register bit. (The typical resistor values are 200KΩ at V_{DD}=3V.) When a pin outputs low, the resistor is disconnected regardless of a RCR2 register bit being set.

Bit 5 thru bit 0 have open drain or CMOS output options, which are controlled by the corresponding WOM2 register bits. These open drain or CMOS output options may be selected for either the general purpose output ports or the peripheral outputs (EVO, TCMP, and TDO).

7.4 PORT D

Port D pins serve one of two basic functions depending on the MCU mode selected; LCD Frontplanes and Backplanes driver outputs, or general purpose output pins. Since Port D is an output only port there is no DDRD register. In place of DDRD is Port D MUX Control Register (PDMUX). Bits 7-4 of this register control the Port/LCD muxing of Port D bits 7-4 respectively on a bit-wise basis. These bits are cleared on reset, and writing a 1 to any bit will turn that pin into a Port output. These outputs have the capability to drive 10 mA sink current when (VOL ≤ Vss + 0.8V).

On reset, all Port D outputs are disconnected from the pins and the Port D data latches are set to 1.

The pin connections of the lower 4-bits of Port D depend on the LCD duty selection by the DUTYS bit in the LCDCR. When the LCD duty is not “1/4”, the unused Backplanes driver(s) is (are) replaced by the Port D output pin(s) automatically.

If DWOMH or DWOML bits in the WOM1 register is set, the P-channel drivers of output buffers at the upper 4 bits or lower 3 bits, respectively are disabled (open drain mode). This open drain controls do not apply to the pins which are configured as Frontplanes or Backplanes driver outputs.

7.4.1 Port D MUX Register (PDMUX)

option map

PDMUX $0003

	7	6	5	4	3	2	1	0
R	PDM7	PDM6	PDM5	PDM4	0	0	0	0
W								
reset ⇒	0	0	0	0	0	0	0	0

When PDM*x* is set, the corresponding pin, PD*x*, is configure to port D output.

When PDM*x* is clear, the corresponding pin, PD*x*, is configure to LCD output.

7.5 PORT E

Port E pins serve one of two basic functions depending on the MCU mode selected: LCD Frontplanes driver outputs, or general purpose output pins. Since Port E is an output only port there is no DDRE register.In place of DDRE is Port E MUX Control Register (PEMUX). Bits 7-0 of this register control the Port/LCD muxing of Port E bits 7-0 respectively on a bit-wide basis. These bits are cleared on reset, and writing a 1 to any bit will turn that pin into a Port output. These outputs have the capability to drive 10 mA sink current when (VOL ≤ Vss + 0.8V).

On reset, all Port E outputs are disconnected from the pins and the Port E data latches are set to 1.

If EWOMH or EWOML bits in the WOM1 register is set, the P-channel driver of output buffers at the upper or lower 4 bits, respectively, are disabled (open drain mode). This open drain controls do not apply to the pins which are configured as Frontplanes driver outputs.

7.5.1 Port E MUX Register (PEMUX)

option map

PEMUX $0004	7	6	5	4	3	2	1	0
R	PEM7	PEM6	PEM5	PEM4	PEM3	PEM2	PEM1	PEM0
W								
reset ⇒	0	0	0	0	0	0	0	0

When PEM*x* is set, the corresponding pin, PE*x*, is configure to port E output.

When PEM*x* is clear, the corresponding pin, PE*x*, is configure to LCD output.

7.6 PORT F, PORT G AND PORT H

Port F, Port G and Port H serve one of two basic functions depending on the MCU mode selected; LCD Frontplanes driver outputs, or general purpose I/O pins. Their MUX Control Register (PFMUX, PGMUX, PHMUX) control the Port/LCD muxing of corresponding Port on a bit-wide basis. These bits are cleared on reset, and writing a 1 to any bit will turn that pin into a Port output. These outputs have the capability to drive 10 mA sink current when (VOL ≤ Vss + 0.8V).

The data direction of a Port pin is determined by its corresponding DDR bit. When a Port pin is programmed as an output by the corresponding DDR bit, data in the PORT data register becomes output data to the pin and this data is returned when PORT register is read.

7.6.1 Port F MUX Register (PFMUX)

option map

PFMUX $000C	7	6	5	4	3	2	1	0
R	PFM7	PFM6	PFM5	PFM4	PFM3	PFM2	PFM1	PFM0
W								
reset ⇒	0	0	0	0	0	0	0	0

When PFM*x* is set, the corresponding pin, PF*x*, is configure to port F I/O.

When PFM*x* is clear, the corresponding pin, PF*x*, is configure to LCD output.

7.6.2 Port G MUX Register (PGMUX)

option map

PGMUX $000D		7	6	5	4	3	2	1	0
	R	PGM7	PGM6	PGM5	PGM4	PGM3	PGM2	PGM1	PGM0
	W								
reset ⇒		0	0	0	0	0	0	0	0

When PGM*x* is set, the corresponding pin, P*Gx*, is configure to port G I/O.

When PGM*x* is clear, the corresponding pin, P*Gx*, is configure to LCD output.

7.6.3 Port H MUX Register (PHMUX)

option map

PHMUX $000E		7	6	5	4	3	2	1	0
	R	PHM7	PHM6	PHM5	PHM4	PHM3	PHM2	PHM1	PHM0
	W								
reset ⇒		0	0	0	0	0	0	0	0

When PHM*x* is set, the corresponding pin, PH*x*, is configure to port H I/O.

When PHM*x* is clear, the corresponding pin, PH*x*, is configure to LCD output.

7.7 INPUT/OUPUT PROGRAMMING

Bidirectional port lines may be programmed as an input or an output under software control. The direction of the pins is determined by the state of the corresponding bit in the port data direction register (DDR). Each port has an associated DDR. Any I/O port pin is configured as an output if its corresponding DDR bit is set. A pin is configured as an input if its corresponding DDR bit is cleared.

During Reset, all DDRs are cleared, which configure all port pins as inputs. The data direction registers are capable of being written to or read by the processor. During the programmed output state, a read of the data register actually reads the value of the output data latch and not the I/O pin. See **Figure 7-1** and **Table 7-1**.

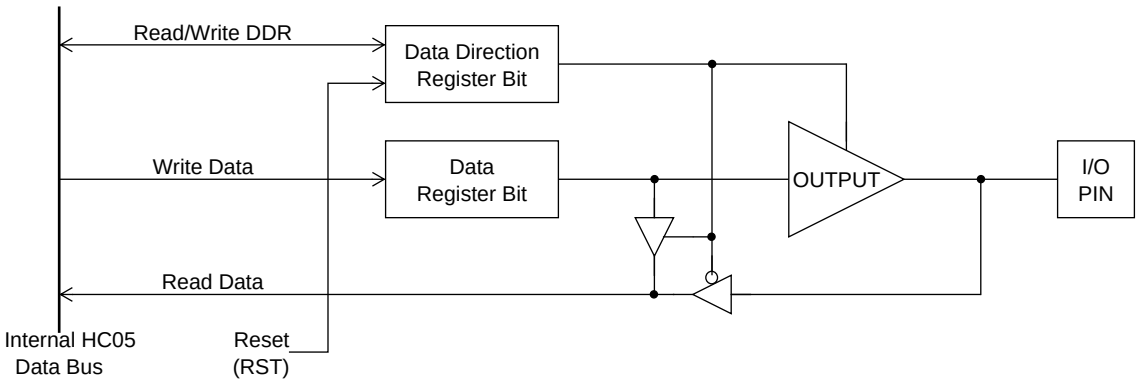


Figure 7-1. Port I/O Circuitry

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 output mode. The output data latch is read.

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 pre-conditioned to the desired state before changing the corresponding DDR bit from a zero to a one.

7.8 PORT OPTION CONTROL REGISTERS

7.8.1 Resistor Control Register 1 (RCR1)

option map RCR1 $0008

	7	6	5	4	3	2	1	0
R	0	0	0	0	RBH	RBL	RAH	RAL
W								
reset ⇒	0	0	0	0	0	0	0	0

RBH – Port B pull up Resistor (H)

When this bit is set, pull up resistors are connected to the upper 4 bits of Port B pins. This bit is cleared on reset.

RBL– Port B pull up Resistor (L)

When this bit is set, pull up resistors are connected to the lower 4 bits of Port B pins. This bit is cleared on reset.

RAH – Port A pull up Resistor (H)

When this bit is set, pull up resistors are connected to the upper 4 bits of Port A pins. This bit is cleared on reset.

RAL – Port A pull up Resistor (L)

When this bit is set, pull up resistors are connected to the lower 4 bits of Port A pins. This bit is cleared on reset.

The typical resistor values are 200KΩ at V_{DD}=3V for both Port A and Port B. When a pin outputs a low level, the pull-up resistor is disconnected regardless of pull-up enable state.

7.8.2 Resistor Control Register 2 (RCR2)

option map RCR1 $0009

	7	6	5	4	3	2	1	0
R	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0
W								
reset ⇒	0	0	0	0	0	0	0	0

RC*x* – Port C pull up Resistor (Bit *x*)

When RC*x* bit is set, pull up resistor is connected to the corresponding bit of Port C pin. This bit is cleared on reset.

The typical resistor values are 200KΩ at V_{DD}=3V.

7.8.3 Open Drain Output Control Register 1 (WOM1)

option map WOM1 $000A

	7	6	5	4	3	2	1	0
R	DWOMH	DWOML	EWOMH	EWOML	0	0	AWOMH	AWOML
W								
reset ⇒	0	0	0	0	0	0	0	0

DWOMH – Port D Open Drain Mode (H)

When this bit is set, the upper 4 bits of Port D are configured as open drain outputs if these bits are selected as Port D output by the PDM*x* bits in the PDMUX. This bit is cleared on reset.

DWOML – Port D Open Drain Mode (L)

When this bit is set, the lower 4 bits of Port D are configured as open drain outputs if the corresponding BP*x* pin is not used by the LCD driver. This bit is cleared on reset.

EWOMH – Port E Open Drain Mode (H)

When this bit is set, the upper 4 bits of Port E that are configured as I/O output by the PEM*x* bits in the PEMUX are configured as open drain outputs. This bit is cleared on reset.

EWOML – Port E Open Drain Mode (L)

When this bit is set, the lower 4 bits of Port E that are configured as I/O output by the PEM*x* bits in the PEMUX are configured as open drain outputs. This bit is cleared on reset.

AWOMH – Port A Open Drain Mode (H)

When this bit is set, upper 4 bits of Port A that are configured as output (corresponding DDRA bit set) becomes open drain outputs. This bit is cleared on reset.

AWOML – Port A Open Drain Mode (L)

When this bit is set, lower 4 bits of Port A that are configured as output (corresponding DDRA bit set) becomes open drain outputs. This bit is cleared on reset.

7.8.4 Open Drain Output Control Register 2 (WOM2)

option map WOM2 $000B

	7	6	5	4	3	2	1	0
R	1	1	CWOM5	CWOM4	CWOM3	CWOM2	CWOM1	CWOM0
W								
reset ⇒	1	1	0	0	0	0	0	0

CWOM*x* – Port C Open Drain Mode (bit *x*)

When CWOM*x* bit is set, Port C bit *x* are configured as open drain outputs if DDRC*x* is set. This bit is cleared on reset.

SECTION 8
CLOCK DISTRIBUTION

There are two oscillator blocks: OSC and XOSC. Several combinations of the clock distributions are allowed for the modules in the MC68HC05PD6. Refer to the following block diagram.

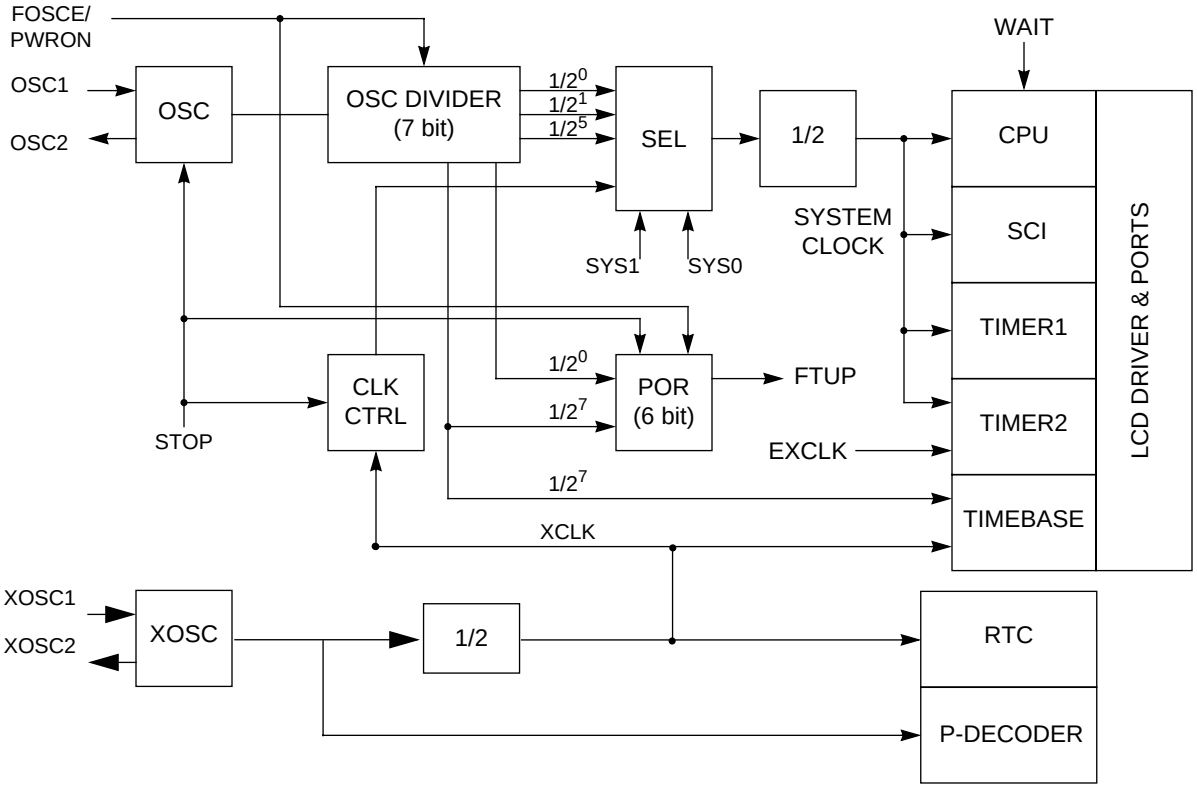


Figure 8-1. Clock Signal Distribution

8.1 OSC CLOCK DIVIDER AND POR COUNTER

The OSC clock is divided by a 7 bit counter which is used for the system clock, Time Base and POR Counter. Clocks divided by 2, 4, and 64 are available for the system clock selections and clock divided by 128 is provided for the Time Base and POR Counter.

The POR counter is a 6 bit clock counter that is driven by the OSC divided by 128. The overflow of this counter is used for setting FTUP bit, release of power on reset (POR), and resuming operation from STOP mode.

The 7 bit divider and POR counter are initialized to $0078 by the following conditions.

- Power on detection
- When FOSCE bit is cleared

8.2 SYSTEM CLOCK CONTROL

The system clock is provided for all internal modules except Time Base.

Both OSC and XOSC are available as the system clock source. The divide ratio is selected by the SYS1 and SYS0 bits in the MISC register. By default OSC divided by 64 is selected on reset.

Table 8-1. System Clock Frequencies

SYS1	SYS0	DIVIDE RATIO	OSC=4MHz	OSC=4.1943MHz	XOSC=76.8kHz
0	0	OSC ÷ 2	2MHz	2.0972MHz	—
0	1	OSC ÷ 4	1MHz	1.0486MHz	—
1	0	OSC ÷ 64	62.5kHz	65.536kHz	—
1	1	XOSC ÷ 4	—	—	19.2kHz

8.3 OSC AND XOSC

The secondary oscillator (XOSC) runs continuously after Power-Up. The main oscillator (OSC) can be stopped to conserve power via the STOP instruction or the FOSCE bit in the MISC register. The effects of restarting the OSC will vary depending on current state of the MCU, including SYS0:1 and FOSCE.

8.3.1 OSC On Line

If the system clock is OSC, FOSCE should remain 1. Executing the STOP instruction in this condition will halt OSC, put the MCU into a low power mode and clear the 6-bit POR counter. The 7-bit divider is not initialized. Exiting STOP with external IRQ or Reset re-starts the oscillator. When the POR counter overflows, internal Reset is released and execution can begin. The stabilization time will vary between 7944 and 8072 counts.

NOTE

Exiting STOP with external Reset will always return the MCU to the state as defined by the register definitions. i.e. SYS0:1=1:0, FOSCE=1.

8.3.2 XOSC On Line

If XOSC is the System Clock (SYS0:1=1:1), OSC can be stopped either by the STOP instruction or by clearing the FOSCE bit.

The sub oscillator (XOSC) never stops except during power down. This clock may also be used as the clock source of the system clock and Time Base.

8.3.2.1 XOSC with FOSCE=1

If the System Clock is XOSC and FOSCE=1, executing the STOP instruction will halt OSC, put the MCU into a low power mode and clear the 6-bit POR counter. The 7-bit divider is not initialized. Exiting STOP with external IRQ re-starts the oscillator, however execution begins immediately using XOSC. When the POR counter overflows, FTUP is set signaling that OSC is stable and OSC can be used as the System Clock. The stabilization time will vary between 7944 and 8072 counts.

8.3.2.2 XOSC with FOSCE=0

If XOSC is the System Clock, clearing FOSCE will stop OSC, and preset the 7-bit divider and 6-bit POR counter to $0078. Execution will continue with XOSC and when FOSCE is set again, OSC will re-start. When the POR counter overflows, FTUP is set signaling that OSC is stable and OSC can be used as the System Clock. The stabilization time will be 8072 counts.

8.3.2.3 XOSC with FOSCE=0 and STOP

If XOSC is the System Clock and FOSCE is cleared, further power reduction can be achieved by executing the STOP instruction. In this case OSC is stopped, the7-bit divider and 6-bit POR counter is preset to $0078 (since FOSCE=0) and execution is halted. Exiting STOP with external IRQ does not re-start the OSC, however execution begins immediately using XOSC. OSC may be re-started by setting FOSCE, and when the POR counter overflows, FTUP will be set signaling that OSC is stable and can be used as the System Clock. The stabilization time will be 8072 counts.

8.3.2.4 STOP and WAIT Modes

During STOP mode the main oscillator (OSC) is shut down and the clock path from the second oscillator (XOSC) is disconnected, such that all modules except Time Base are halted. Entering STOP mode clears FTUP flag in the MISC register, and initializes POR counter. The STOP mode is exited by RESET, PDI, IRQ1, IRQ2, KWI, or RTC Interrupt.

If OSC is selected as system clock source during STOP mode, CPU resumes after the overflow of POR counter and this overflow also sets FTUP status flag.

If XOSC is selected as system clock source during STOP mode, no stop recovery time is required for exiting STOP mode because XOSC never stops and re-start of main oscillator depends on FOSCE bit.

During WAIT mode, only the CPU clocks are halted and the peripheral modules are not affected. The WAIT mode is exited by the RESET and any interrupts.

Table 8-2. Recovery Time Requirements

BEFORE RESET OR INTERRUPT			POWER-ON RESET	EXTERNAL RESET	EXIT STOP MODE BY INTERRUPT
CPU CLK SOURCE	**STOP**	**FOSCE**			
—	—	—	WAIT	—	—
OSC (OSC ON)	OUT	1	—	NO WAIT	—
OSC (OSC OFF)	OUT	0(1)	—	WAIT	—
	IN	1	—	WAIT	WAIT
	IN(2)	0(2)	—	WAIT	WAIT
XOSC (OSC ON)	OUT	1	—	NO WAIT	—
XOSC (OSC OFF)	OUT	0	—	WAIT	—
	IN	1	—	WAIT	NO WAIT
	IN	0	—	WAIT	NO WAIT

(1) This case has no meaning for applications.
(2) This case never occurs.

8.4 TIME BASE

Time Base is a 14 bit up-counter which is clocked by XOSC input or OSC input divided by 128. TBCLK bit in the TBCR1 register selects the clock source.

This 14 bit divider is initialized to $0078 only upon power on reset (POR). After counting 8072 clocks, the STUP bit in the MISC register is set.

The divided clocks from the Time Base are used for LCDCLK, STUP, and COP.

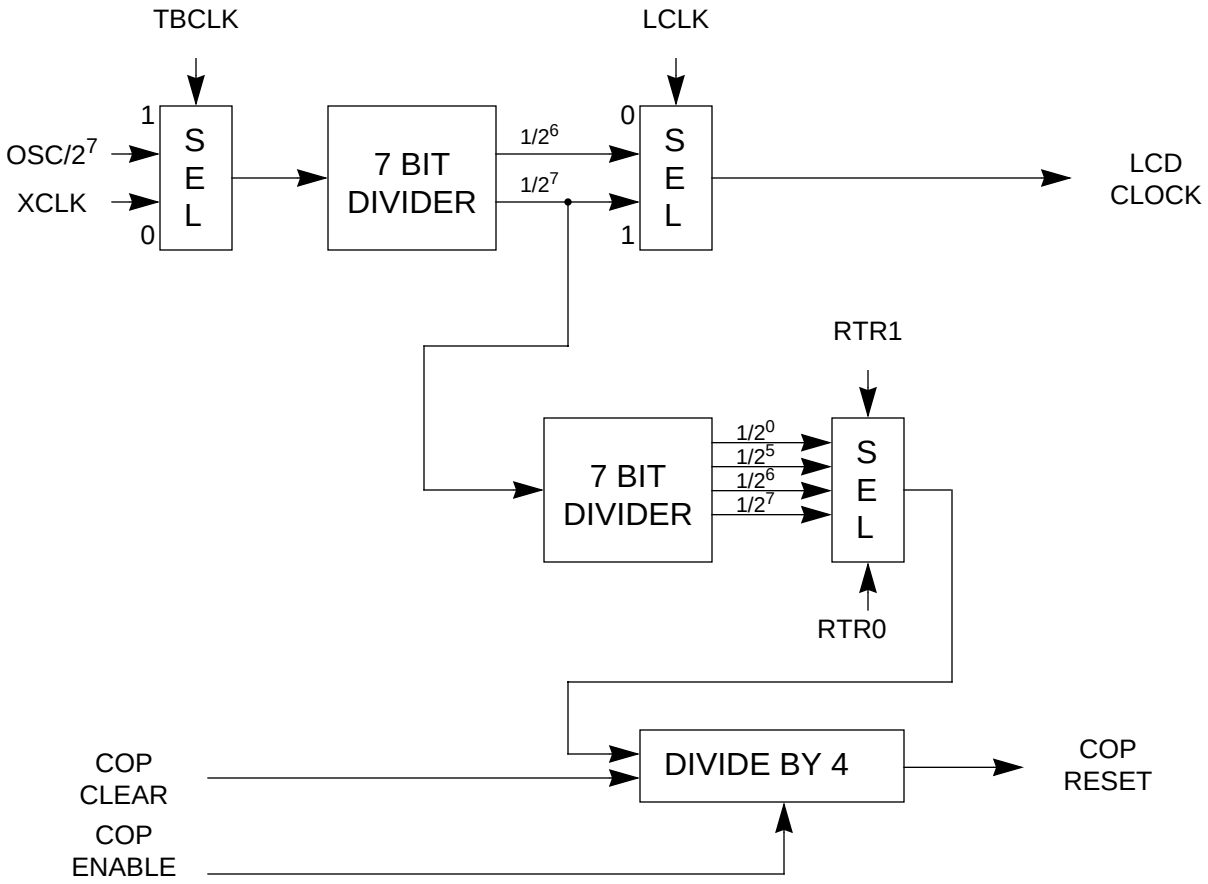


Figure 8-2. Time Base Clock Divider

8.4.1 LCDCLK

The clocks divided by 64 and 128 are used as LCD clock at the LCD driver module, and clocks are selected by the LCLK bit in the TBCR1.

8.4.2 STUP

Time Base divider is initialized to $0078 by the power on detection and when the count reaches 8072, STUP flag in the MISC register is set. Once STUP flag is set, it is never cleared until power down.

8.4.3 Watchdog Timer (COP)

The Computer Operating Properly Watchdog Timer (COP) is controlled by the COPE bit in the TBCR2 register.

The COP uses the clock that is selected by the RTR1 and RTR0 bits. COP time-out reset will be generated if the COP enable (COPE) bit is set. The COP time-out reset has the same vector address as POR and external RESET. To prevent the COP time-out the COP divider is cleared by writing a '0' to bit 0 of address $FFF0.

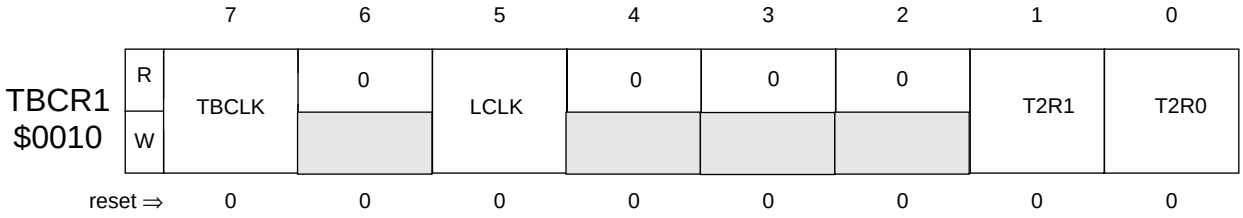
When the Time Base divider is driven by the OSC clock, clock for the divider is suspended during STOP mode or when FOSCE is 0. This may cause COP period stretching or no COP time-out reset when processing errors occur. It is recommended that XOSC clock to be used for the COP functions to avoid these problems.

When the Time Base (COP) divider is driven by the XOSC clock, the divider does not stop counting and writing a '0' to bit 0 of address $FFF0 must be triggered to prevent the COP time-out.

Table 8-3. COP Time Out Period

RTR1	RTR0	COP PERIOD (ms)					
		OSC=4MHz		OSC=4.1943MHz		XOSC=76.8kHz	
		MIN	MAX	MIN	MAX	MIN	MAX
0	1	12.3	16.4	11.7	15.6	10	13.33
0	1	393	524	375	500	320	426.67
1	0	786	1048	750	1000	640	853.33
1	1	1573	2097	1500	2000	1280	1706.66

8.4.4 Time Base Control Register 1 (TBCR1)



TBCLK – Time Base Clock

The TBCLK bit selects Time Base clock source. This bit is cleared on reset. After reset, write to this bit is allowed only once.

0 = XOSC clock is selected
1 = OSC clock divide-by 128 is selected

LCLK – LCD Clock

The LCLK bit selects clock for the LCD driver. This bit is cleared on reset.

0 = Divide by 64 is selected
1 = Divide by 128 is selected

T2R1/0 – Timer 2 Prescale Rate select bits

T2R1 and T2R0 select Timer 2 clock rate. See **Section 9** for a detailed description.

8.4.5 Time Base Control Register 2 (TBCR2)

		7	6	5	4	3	2	1	0
TBCR2 $0011	R	0	0	RTR1	RTR0	0	0	COPE	0
	W								
reset ⇒		0	X	1	1	0	0	0	0

RTI1/0 – Real Time Interrupt Rate Select

The RTR1 and RTR0 bits select one of four rates for the Real Time Interrupt period. The RTI rate is related to the COP time-out reset period. These bits are set to 1 on reset.

RT1	RT0	DIVIDE RATIO	OSC=4MHz	OSC=4.1943MHz	XOSC=76.8kHz
0	0	TBCLK ÷ 128	244Hz	256Hz	600Hz
0	1	TBCLK ÷ 4096	7.63Hz	8Hz	18.75Hz
1	0	TBCLK ÷ 8192	3.81Hz	4Hz	9.38Hz
1	1	TBCLK ÷ 16384	1.91Hz	2Hz	4.69Hz

COPE – COP Enable

When the COPE bit is 1, COP reset function is enabled. This bit is cleared on reset (including COP time-out reset) and write to this bit is allowed only once after rest.

8.5 REAL TIME CLOCK (RTC)

Real time clock is software programmable which can be either enabled or disabled. Real time clock consists of three binary counters which are divided down from the 76.8KHz oscillator. There are three bits in the RTC status register (address $39) and three bits in the RTC Control register (address $3A) where the operation of the real time clock is controlled.

8.5.1 RTC Time Registers

Three locations are reserved for real time clock operation; they are:

		7	6	5	4	3	2	1	0
HOUR $0033	R	0	0	0	HOUR4	HOUR3	HOUR2	HOUR1	HOUR0
	W								
reset ⇒		0	0	0	0	0	0	0	0

		7	6	5	4	3	2	1	0
MIN $0034	R	0	0	MIN5	MIN4	MIN3	MIN2	MIN1	MIN0
	W								
reset ⇒		0	0	0	0	0	0	0	0

		7	6	5	4	3	2	1	0
SEC $0035	R	0	0	SEC5	SEC4	SEC3	SEC2	SEC1	SEC0
	W								
reset ⇒		0	0	0	0	0	0	0	0

The hours and minutes registers are in fact holding registers (See **Figure 8-3**). When setting the RTC time, the hours and minutes are written first, then writing to the seconds register will cause the hours and minutes in their holding registers to be latched into the RTC hardware.

When reading the RTC time, the seconds register is first accessed; this will cause the hardware to latch the hours and minutes into their respective holding registers. Now, reading the hours and minutes registers will give the time for the moment when the seconds register was read.

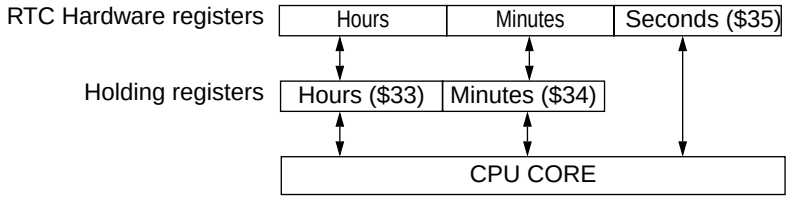


Figure 8-3. RTC Holding Register

8.5.2 RTC Alarm Registers

There are three locations associated with the alarm registers as shown.

The ALF bit in the RTC Status register is set when the contents of the RTC Alarm registers matches the contents of the RTC Time registers.

		7	6	5	4	3	2	1	0
HOURA $0036	R	0	0	0	HOURA4	HOURA3	HOURA2	HOURA1	HOURA0
	W								
reset ⇒		0	0	0	0	0	0	0	0

		7	6	5	4	3	2	1	0
MINA $0037	R	0	0	MINA5	MINA4	MINA3	MINA2	MINA1	MINA0
	W								
reset ⇒		0	0	0	0	0	0	0	0

		7	6	5	4	3	2	1	0
SECA $0038	R	0	0	SECA5	SECA4	SECA3	SECA2	SECA1	SECA0
	W								
reset ⇒		0	0	0	0	0	0	0	0

8.5.3 RTC Status Register

There are three interrupts associated with the RTC, their flags are in the RTC Status register ($39), and their respective enable bits are the RTC Control register ($3A).

		7	6	5	4	3	2	1	0
RTCS $0039	R	0	0	0	0	0	SECF	ALF	RTCF
	W								
reset ⇒		0	0	0	0	0	0	0	0

RTCF

When this bit is set, real time clock interrupts CPU once a day. After serving this interrupt, it is the user's responsibility to clear this bit, otherwise the CPU will keep on serving this once a day interrupt when a new RTC interrupt occurs (even there is no once a day interrupt occurs). This bit is cleared by writing a 1 to it.

ALF

When this bit is set, alarm interrupt has occurred. After serving this interrupt, it is the user's responsibility to clear this bit, otherwise the CPU will keep on serving this alarm interrupt when a new RTC interrupt occurs (even there is no alarm interrupt occurs). This bit is cleared by writing a 1 to it.

SECF

When this bit is set, real time clock interrupts CPU once a second. After serving this interrupt, it is the user's responsibility to clear this bit, otherwise the CPU will keep on serving this once a second interrupt when a new RTC interrupt occurs (even there is no once a second interrupt occurs). This bit is cleared by writing a 1 to it.

8.5.4 RTC Control Register

There are three interrupts associated with the RTC, their flags are in the RTC Status register ($39), and their respective enable bits are the RTC Control register ($3A).

		7	6	5	4	3	2	1	0
RTCC $003A	R	0	0	0	0	0	SECE	ALE	RTCE
	W								
reset ⇒		0	0	0	0	0	0	0	0

RTCE

1 = Real time clock once a day interrupt enabled
0 = Real time clock once a day interrupt disabled

ALE

1 = Alarm interrupt enabled
0 = Alarm interrupt disabled

SECE

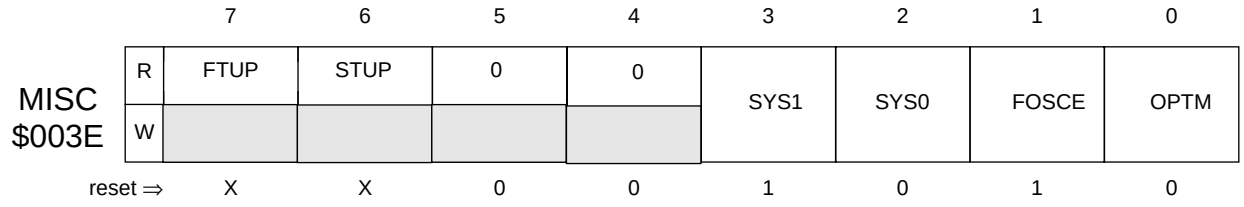
1 = Real time clock once a second interrupt enabled
0 = Real time clock once a second interrupt disabled

It should be noted that the flags in the interrupt status register will be set if the corresponding event is detected, irrespective of the setting of the interrupt enable bits.

Following is an example showing how to use the RTC interrupt:

```
*Main program
                              BSET0,$3AEnable RTC (once a day) interrupt.
       BSET    1,$3A          Enable RTC (alarm) interrupt.
       BSET    2,$3A          Enable RTC (once a second) interrupt.
       STOP                   MCU execute STOP instruction for power
                              conservation.
*Real time clock interrupt service routine
       BRSET   0,$39,ODAY     This bit is set indicating this is an once a day
                              RTC interrupt.
       BRSET   1,$39,ALINT    This bit is set indicating this is a RTC interrupt
                              caused by a match of alarm registers and real time
                              clock registers.
       BRSET   2,$39,OSEC     This bit is set indicating this is an once a second
                              RTC interrupt.
ODAY   BSET    0,$39          Clear this bit so that this once a day interrupt
                              will not be recognized as a new one on next RTC
                              interrupt.
       JSR     OADAY          Once a day interrupt service routine.
       BRSET   1,$39,ALINT    This bit is set indicating alarm interrupt also
                              occurs at the same time.
       BRSET   2,$39,OSEC     This bit is set indicating once a second RTC
                              interrupt also occurs at the same time.
RTCR   RTI
ALINT  BSET    1,$39          Clear this bit so that this alarm interrupt will
                              not be recognized as a new one on next RTC
                              interrupt.
       JSR     ALARM          Alarm service interrupt routine.
       BRSET   2,$39,OSEC     This bit is set indicating once a second interrupt
                              also occurs at the same time.
       BRA     RTCR           Return from interrupt.
OSEC   BSET    2,$39          Clear this bit so that this once a second interrupt
                              will not be recognized as a new one on next RTC
                              interrupt.
       JSR     OASEC          Once a second interrupt service routine.
       BRA     RTCR           Return from interrupt.
```

8.6 MISCELLANEOUS REGISTER (MISC)



FTUP – OSC Time Up Flag

Power on detection or clearing FOSCE bit clears this bit. This bit is set by the overflow of the POR counter. An external reset does not affect this bit.

0 = during POR or OSC shut down

1 = OSC clock is available for the system clock

STUP – XOSC Time Up Flag

The power on detection clears this bit. This bit is set after the Time Base has counted 8072 clocks. An reset does not affect this bit.

0 = XOSC is not stabilized or no signal on XOSC1-XOSC2 pins
1 = XOSC clock is available for the system clock

SYS1, SYS0 – System Clock Select

These two bits select the system clock source for all internal modules except Time Base. On reset the SYS1 and SYS0 bits are initialized to 1 and 0, respectively.

SYS1	SYS0	DIVIDE RATIO	OSC=4MHz	OSC=4.1943MHz	XOSC=76.8kHz
0	0	OSC ÷ 2	2MHz	2.0972MHz	—
0	1	OSC ÷ 4	1MHz	1.0486MHz	—
1	0	OSC ÷ 64	62.5kHz	65.536kHz	—
1	1	XOSC ÷ 4	—	—	19.2kHz

FOSCE - Fast (Main) Oscillator Enable

The FOSCE bit controls the main oscillator activity. This bit should not be cleared by the CPU when the main oscillator is selected as the system clock source.

0 = OSC is shut down; 7 bit divider at the OSC input and POR counter are initialized to $0078; FTUP flag is cleared.
1 = Main oscillator starts again; FTUP flag is set by the POR counter overflow (8072 clocks).

OPTM - Option Map Select

The OPTM bit selects one of two register maps at $0000-$003F. This bit is cleared on reset.

0 = Main register map is selected
1 = Option map is selected

SECTION 9
TIMER SYSTEM

The MC68HC05PD6 has two timer modules, Timer 1 with a 16 bit counter and Timer 2 with an 8 bit counter. Timer 1 has TCAP input pin and TCMP output pin. Timer 2 has EVI input pin and EVO output pin. The following block diagram describes the Timer System of MC68HC05PD6.

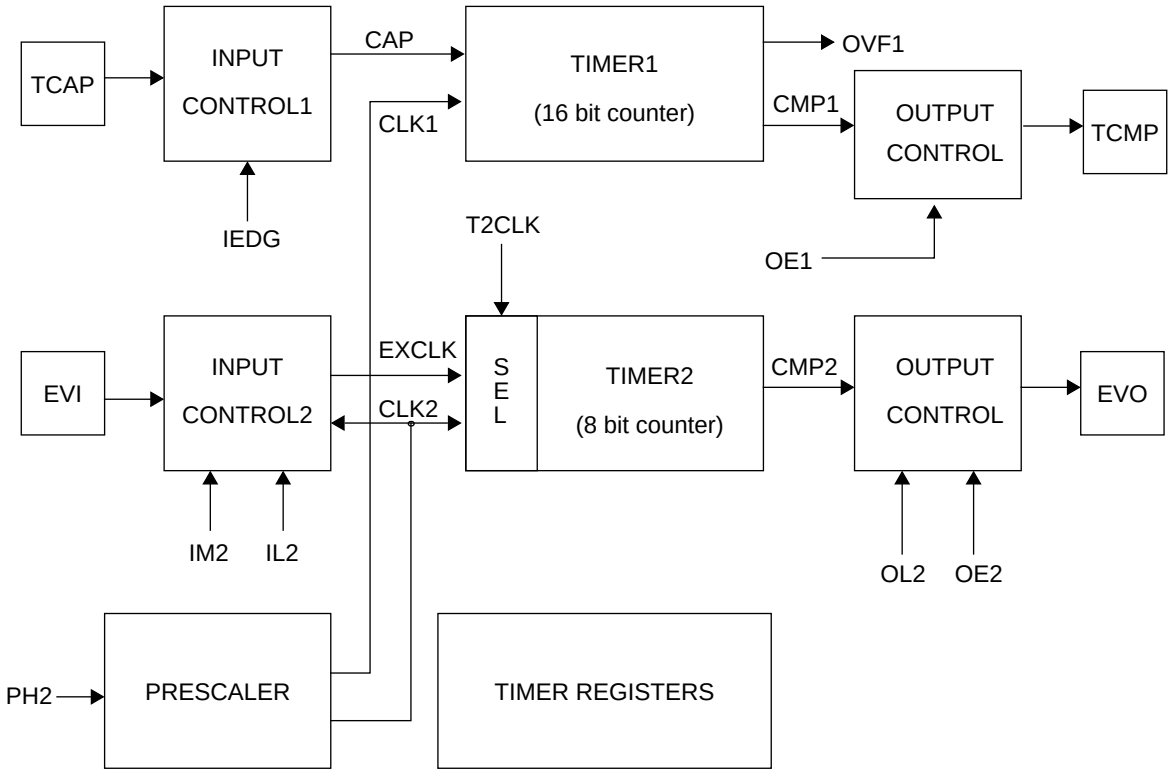


Figure 9-1. Timer System Block Diagram

9.1 TIMER1

The timer consists of a 16-bit free-running counter driven by a fixed divide-by-four prescaler. This timer can be used for many purposes, including input waveform measurements while simultaneously generating an output waveform. Pulse widths can vary from several microseconds to many seconds. Refer to **Figure 9-2** for Timer 1 block diagram.

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

NOTE

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

9.1.1 Counter

The key element in the programmable timer is a 16-bit, free-running counter or counter register, preceded by a prescaler that divides the internal processor clock by four. The prescaler gives the timer a resolution of 2 microseconds if the internal bus clock is 2.0 MHz. The counter is incremented during the low portion of the internal bus clock. Software can read the counter at any time without affecting its value.

The double-byte, free-running counter can be read from either of two locations, $18-$19 (counter register) or $1A-$1B (counter alternate register). A read from only the least significant byte (LSB) of the free-running counter ($19, $1B) receives the count value at the time of the read. If a read of the free-running counter or counter alternate register first addresses the most significant byte (MSB) ($18, $1A), the LSB ($19, $1B) is transferred to a buffer. This buffer value remains fixed after the first MSB read, even if the user reads the MSB several times. This buffer is accessed when reading the free-running counter or counter alternate register LSB ($19 or $1B) and, thus, completes a read sequence of the total counter value. In reading either the free-running counter or counter alternate register, if the MSB is read, the LSB must also be read to complete the sequence.

The counter alternate register differs from the counter register in one respect: a read of the counter register LSB can clear the timer overflow flag (TOF). Therefore, the counter alternate register can be read at any time without the possibility of missing timer overflow interrupts due to clearing of the TOF.

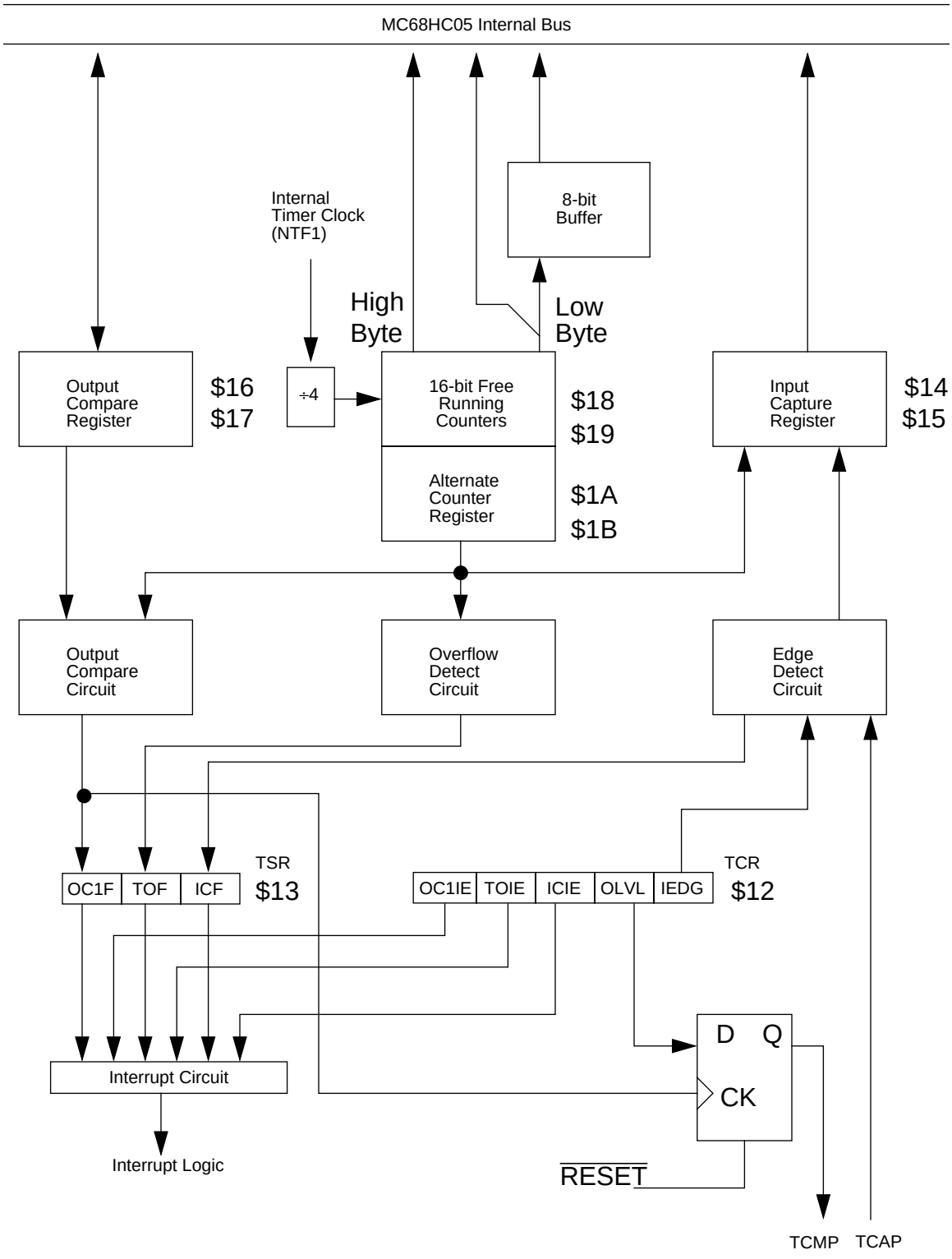
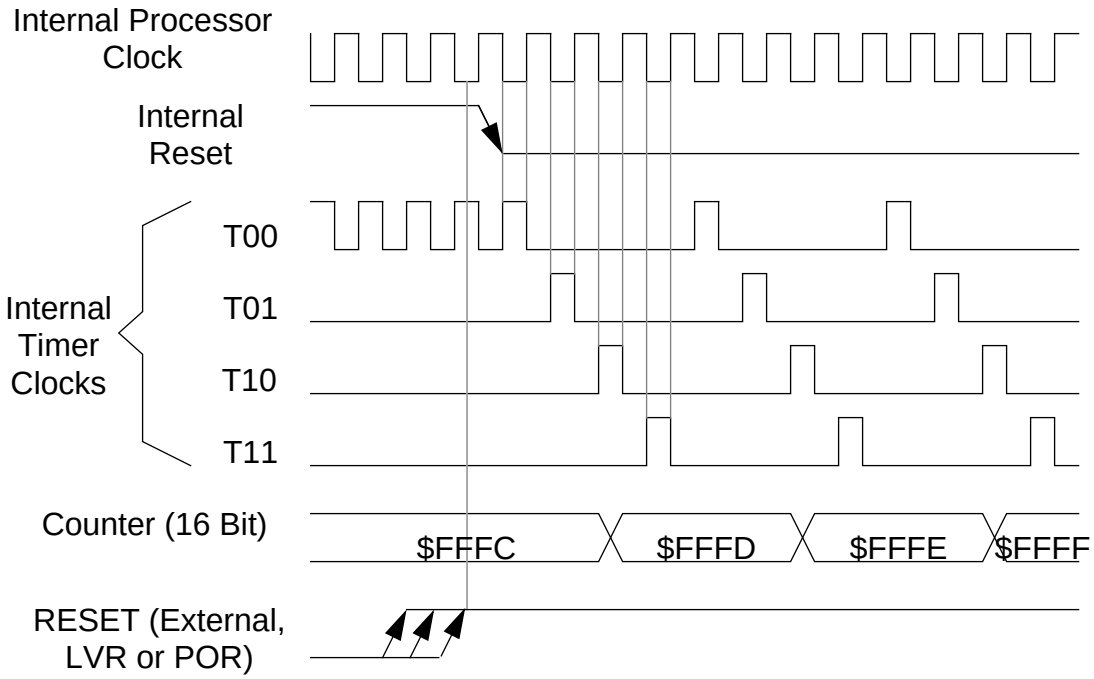
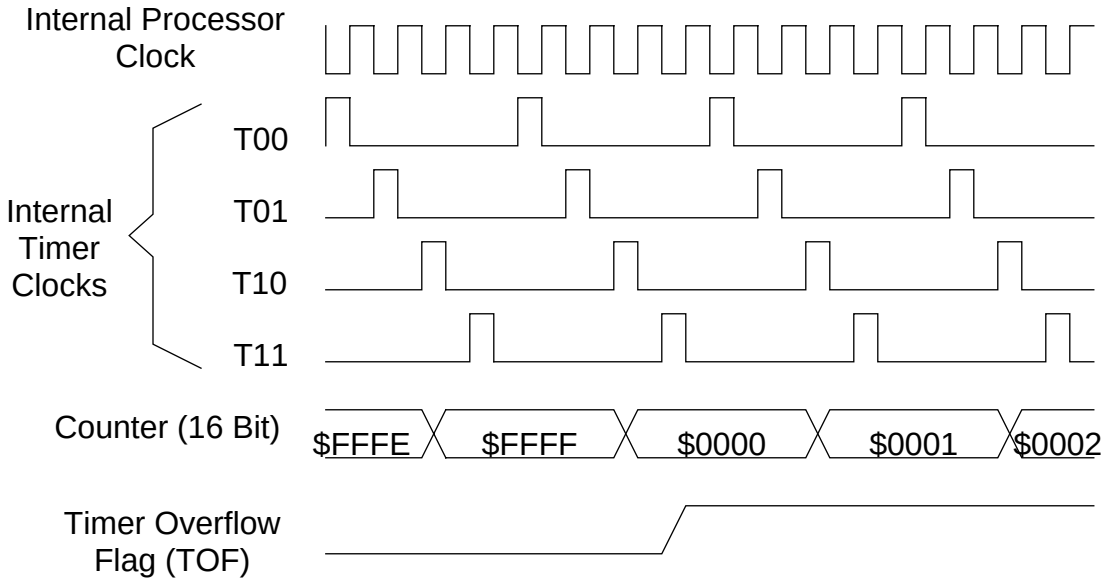


Figure 9-2. Timer 1 Block Diagram



NOTE: The Counter Register and Timer Control Register are the only ones affected by RESET

Figure 9-3. Timer State Timing Diagram for Reset



NOTE: The TOF bit is set at timer state T11 (transition of counter from $FFFF to $0000). It is cleared by read of the timer status register during the internal processor clock high time followed by a read of the counter low register.

Figure 9-4. Timer State Timing Diagram for Timer Overflow

CNTH $0018

	7	6	5	4	3	2	1	0
R	CNT15	CNT14	CNT13	CNT12	CNT11	CNT10	CNT9	CNT8
W								
reset	1	1	1	1	1	1	1	1

CNTL $0019

	7	6	5	4	3	2	1	0
R	CNT7	CNT6	CNT5	CNT4	CNT3	CNT2	CNT1	CNT0
W								
reset	1	1	1	1	1	1	0	0

ACNTH $001A

	7	6	5	4	3	2	1	0
R	ACNT15	ACNT14	ACNT13	ACNT12	ACNT11	ACNT10	ACNT9	ACNT8
W								
reset	1	1	1	1	1	1	1	1

ACNTL $001B

	7	6	5	4	3	2	1	0
R	ACNT7	ACNT6	ACNT5	ACNT4	ACNT3	ACNT2	ACNT1	ACNT0
W								
reset	1	1	1	1	1	1	0	0

The free-running counter is configured to $FFFC during reset and is always a read-only register. During a power-on reset, the counter is also preset to $FFFC and begins running after the oscillator start-up delay. Because the free-running counter is 16 bits preceded by a fixed divide-by-four prescaler, the value in the free-running counter repeats every 262,144 internal bus clock cycles. When the counter rolls over from $FFFF to $0000, the TOF bit is set. An interrupt can also be enabled when counter roll over occurs by setting its interrupt enable bit (TOIE).

9.1.2 Output Compare Register

		7	6	5	4	3	2	1	0
OCMPH $0016	R	OC15	OC14	OC13	OC12	OC11	OC10	OC9	OC8
	W								
	reset	X	X	X	X	X	X	X	X

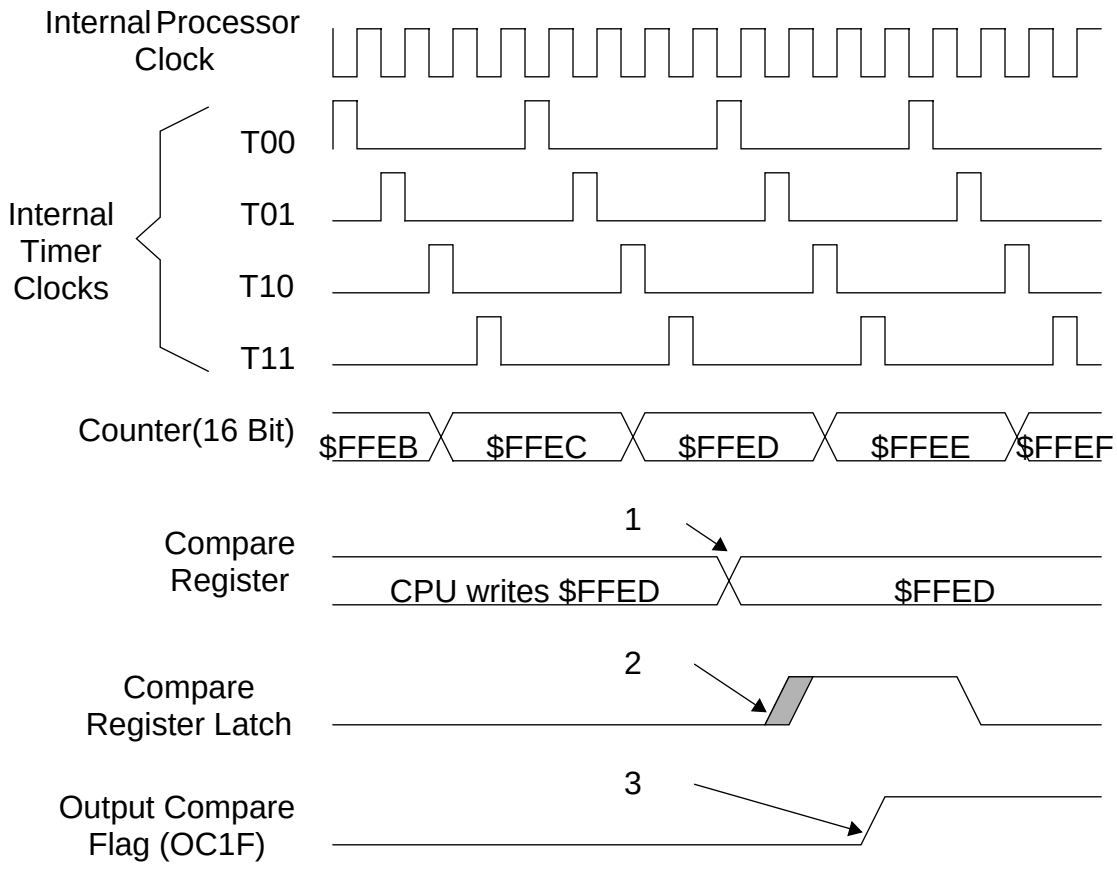
		7	6	5	4	3	2	1	0
OCMPL $0017	R	OC7	OC6	OC5	OC4	OC3	OC2	OC1	OC0
	W								
	reset	X	X	X	X	X	X	X	X

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

The output compare register contents are compared with the contents of the free-running counter continually, and if a match is found, the corresponding output compare flag (OC1F) bit is set and the corresponding output level (OLVL) bit is clocked to an output level register. The output compare register values and the output level bit should be changed after each successful comparison to establish a new elapsed time-out. An interrupt can also accompany a successful output compare provided the corresponding interrupt enable bit (OCIE) is set.

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

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



1. The CPU writes to the compare register may take place at any time, but a compare only occurs at timer state T01. Thus, a 4-cycle different may exist between the write to the compare register and the actual compare.

2. Internal compare takes place during timer state T01.

3. OC1F is set at timer state T11 which follows the comparison match ($FFED in this example).

Figure 9-5. Timer State Timing Diagram For Output Compare

9.1.3 Input Capture Register

		7	6	5	4	3	2	1	0
ICAPH $0014	R	IC15	IC14	IC13	IC12	IC11	IC10	IC9	IC8
	W								
	reset	X	X	X	X	X	X	X	X

		7	6	5	4	3	2	1	0
ICAPL $0015	R	IC7	IC6	IC5	IC4	IC3	IC2	IC1	IC0
	W								
	reset	X	X	X	X	X	X	X	X

Two 8-bit registers, which make up the 16-bit input capture register, are read-only and are used to latch the value of the free-running counter after the corresponding input capture edge detector senses a defined transition. The level transition which triggers the counter transfer is defined by the corresponding input edge bit (IEDG). Reset does not affect the contents of the input capture register.

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

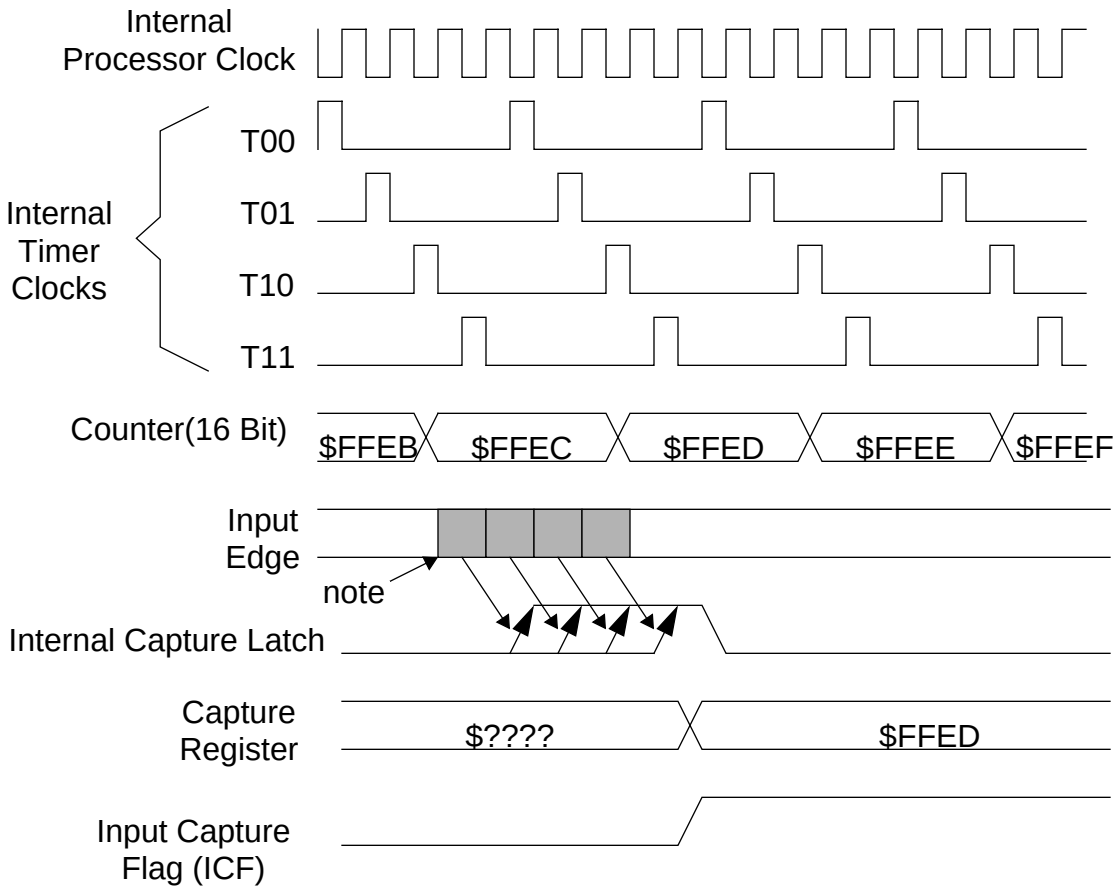
The free-running counter contents are transferred to the input capture register on each proper signal transition regardless of whether the input capture flag (ICF) is set or clear. The input capture register always contains the free-running counter value that corresponds to the most recent input capture.

After a read of the input capture register ($14) MSB, the counter transfer is inhibited until the LSB ($15) is also read. This characteristic causes the time used in the input capture software routine and its interaction with the main program to determine the minimum pulse period.

A read of the input capture register LSB ($15) does not inhibit the free-running counter transfer since they occur on opposite edges of the internal bus clock.

NOTE

Since the TCAP pin is shared with the PC3 I/O pin, changing the state of the PC3 DDR or Data Register can cause an unwanted TCAP interrupt. This can be handled by clearing the ICIE bit before changing the configuration of PC3, and clearing any pending interrupts before enabling ICIE.

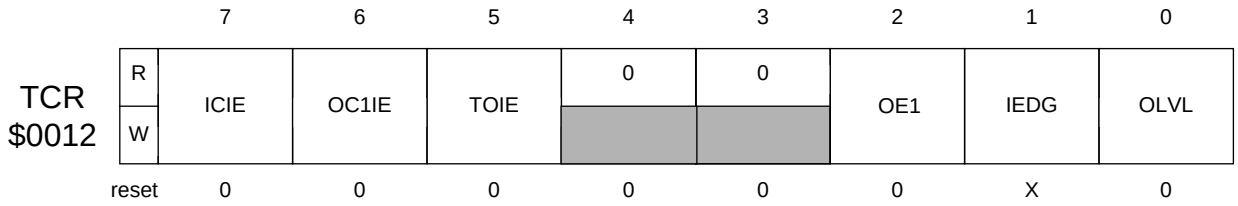


NOTE: If the input edge occurs in the shaded area from one timer state T10 to the other timer state T10 the input capture flag is set during the next state T11.

Figure 9-6. Timer State Timing Diagram For Input Capture

9.1.4 Timer Control Register (TCR)

The TCR is a read/write register containing six control bits. Three bits control interrupts associated with each of the three flag bits found in the timer status register. The other two bits control: 1) which edge is significant to the input capture edge detector (i.e., negative or positive), and 2) the next value to be clocked to the output level register in response to a successful output compare. The timer control register and the free running counter are the only sections of the timer affected by reset. The TCMP pin is forced low during external reset and stays low until a valid compare changes it to high. The timer control register is illustrated below by a definition of each bit.



		7	6	5	4	3	2	1	0
TCR $0012	R	ICIE	OC1IE	TOIE	0	0	OE1	IEDG	OLVL
	W								
reset		0	0	0	0	0	0	X	0

ICIE

If the input capture interrupt enable (ICIE) bit is set, a timer interrupt is enable when the ICF status flag is set, provided the I bit in CCR is cleared. If the ICIE bit is cleared, the interrupt is inhibited. The ICIE bit is cleared by reset.

OC1IE

If the output compare interrupt enable (OC1IE) bit is set, a timer interrupt is enabled whenever the OC1F status flag is set, provided the I bit in CCR is cleared. If the OC1IE bit is cleared, the interrupt is inhibited. The OC1IE bit is cleared by reset.

TOIE

If the timer overflow interrupt enable (TOIE) bit is set, a timer interrupt is enabled whenever the TOF status flag is set, provided the I bit in CCR is cleared. If the TOIE bit is cleared, the interrupt is inhibited. The TOIE bit is cleared by reset.

OE1

The OE1 bit configures whether Port C bit 2 as I/O pin or as output pin of TCMP.

0 = PC2 is selected
1 = TCMP is selected

IEDG

The value of the input edge (IEDG) bit determines which level transition on TCAP pin will trigger a free running counter transfer to the input capture register. Reset does not affect the IEDG bit.

0 = negative edge
1 = positive edge

OLVL

The value of the output level (OLVL) bit is clocked into the output level register by the next successful output compare and will appear at TCMP pin. This bit and the output level register are cleared by reset.

0 = low output
1 = high output

9.1.5 Timer Status Register (TSR)

The timer status register is a read-only register and is illustrated below followed by a definition of each bit. Refer to timing diagrams shown in **Figure 9-3**, **Figure 9-4** and **Figure 9-5** for timing relationship to the timer status register bits.

TSR $0013

	7	6	5	4	3	2	1	0
R	ICF	OC1F	TOF	0	0	0	0	0
W								
reset	X	X	X	0	0	0	0	0

ICF

The input capture flag (ICF) is set when a proper edge has been sensed by the input capture edge detector. It is cleared by a processor access of the timer status register (with ICF set) followed by accessing the low byte ($15) of the input capture register. Reset does not affect the input compare flag.

OC1F

The output compare flag (OC1F) is set when the output compare register contents matches the contents of the free running counter. The OC1F is cleared by accessing the timer status register (with OC1F set) and then accessing the low byte ($17) of the output compare register. Reset does not affect the output compare flag.

TOF

The timer overflow flag (TOF) bit is set by transition of the free run from $FFFF to $0000. It is cleared by accessing the timer status register (with TOF set) followed by an access of the free running counter least significant byte ($19). Reset does not affect the TOF bit.

Accessing the timer status register satisfies the first condition required to clear status bits. The remaining step is to access the register corresponding to the status bit.

A problem can occur when using the timer overflow function and reading the free-running counter at random times to measure an elapsed time. Without incorporating the proper precautions into software, the timer overflow flag could

unintentionally be cleared if: 1)The timer status register is read or written when TOF is set, and 2)The LSB of the free-running counter is read but not for the purpose of servicing the flag. The counter alternate register at address $1A and $1B contains the same value as the free-running counter (at address $18 and $19); therefore, this alternate register can be read at any time without affecting the timer overflow flag in the timer status register.

9.1.6 Operation During Low Power Mode

During STOP and WAIT instructions, the programmable Timer 1 functions as follows: during the wait mode, the Timer 1 continues to operate normally and may generate an interrupt to trigger the CPU out of the wait state; during the STOP mode, the Timer 1 holds at its current state, retaining all data, and resumes operation from this point when external interrupt (IRQ), or internal interrupt is received.

9.2 TIMER 2

Timer 2 is an 8-bit event counter which has one compare register, one event input pin (EVI), and one event output pin (EVO). The event counter is clocked by the external clock (EXCLK) or prescaled system clock (CLK2), selected by the T2CLK bit in the TCR2 register. The EXCLK may be EVI direct or EVI gated by CLK2, which is selected by the IM2 bit at the EVI block (refer to the EVI description).

Timer 2 may be used as a modulus clock divider with EVO pin, free running counter (when compare register is $00), or periodic interrupt timer.

The Timer Counter 2 (CNT2) is an 8-bit up counter with preset input. The counter is preset to $01 by a CMP2 signal from the comparator or by a CPU write to it that is done while the system clock (PH2) is low.

The CLK2 from the prescaler or the EXTCLK from the EVI block are selected as timer clock by the T2CLK bit in the TCR2 register. The CLK2 and the EXCLK are synchronized to the falling edge of system clock in the prescaler and the EVI blocks. The minimum pulse width of CLK2 is the same as the system clock, and the minimum pulse width of EXCLK (event mode) is one PH2 cycle.

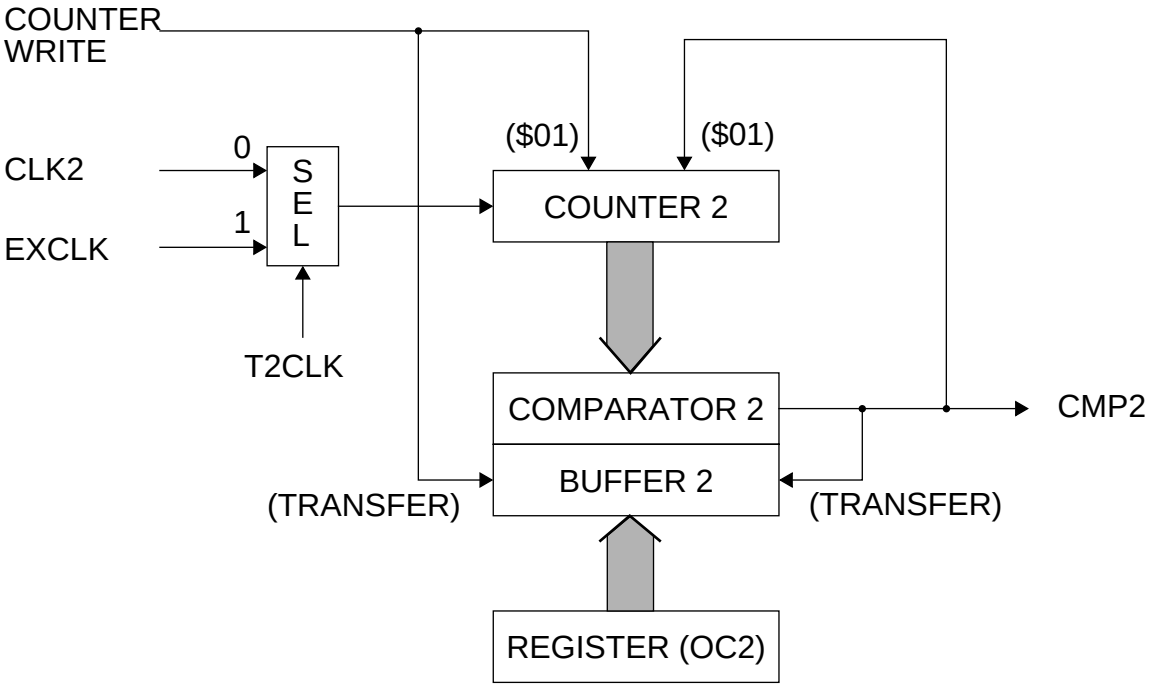


Figure 9-7. Timer 2 Block Diagram

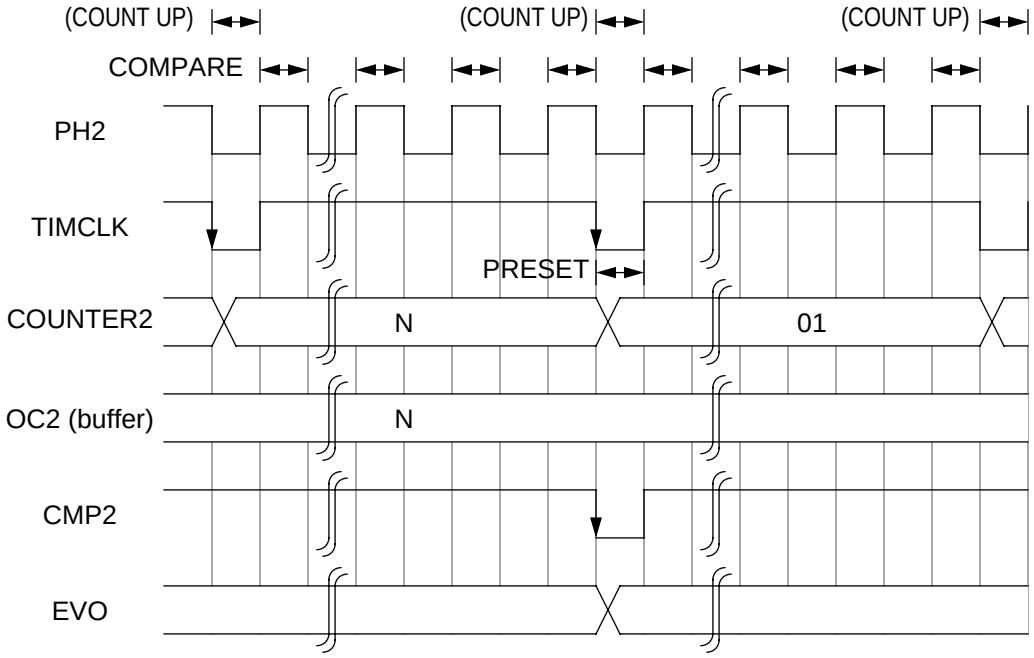
The counter is incremented by the falling edge of the timer clock and the period between two falling edges is defined as one timer cycle in the following description.

The compare register (OC2) is provided for comparison with the timer counter 2 (CNT2). The OC2 data is transferred to the buffer register when the counter is preset by a CPU write or by a compare output (CMP2). This buffer register is compared with the timer counter 2 (CNT2).

The comparison between the counter and the OC2 buffer register is done when the system clock high in each bus cycles. If counter matches with the OC2 buffer register, the comparator latches this result during the current timer cycle. When the next timer cycle begins, the comparator outputs CMP2 signal (if the compare match is detected during previous timer cycle). This CMP2 is used in the counter preset, data transfer to the buffer register, setting OC2F in the TSR2, and the EVO block. The counter preset overrides the counter increment.

The OC2F bit may generate interrupt request if the OC2IE bit in the TCR2 is set.

OC2=2,3,4,...,FF,0



OC2=1

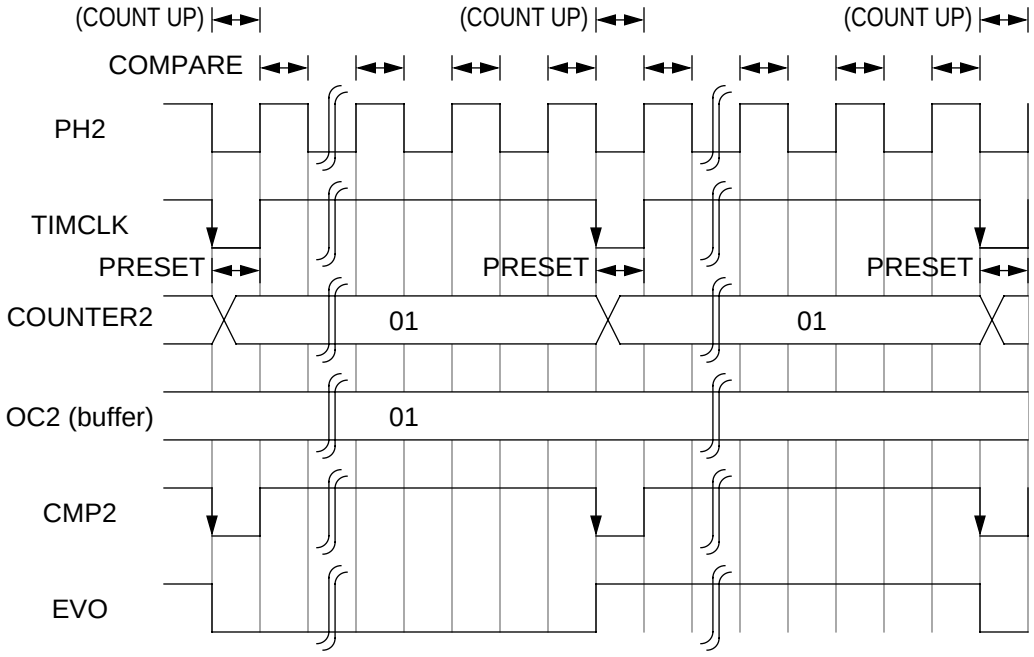
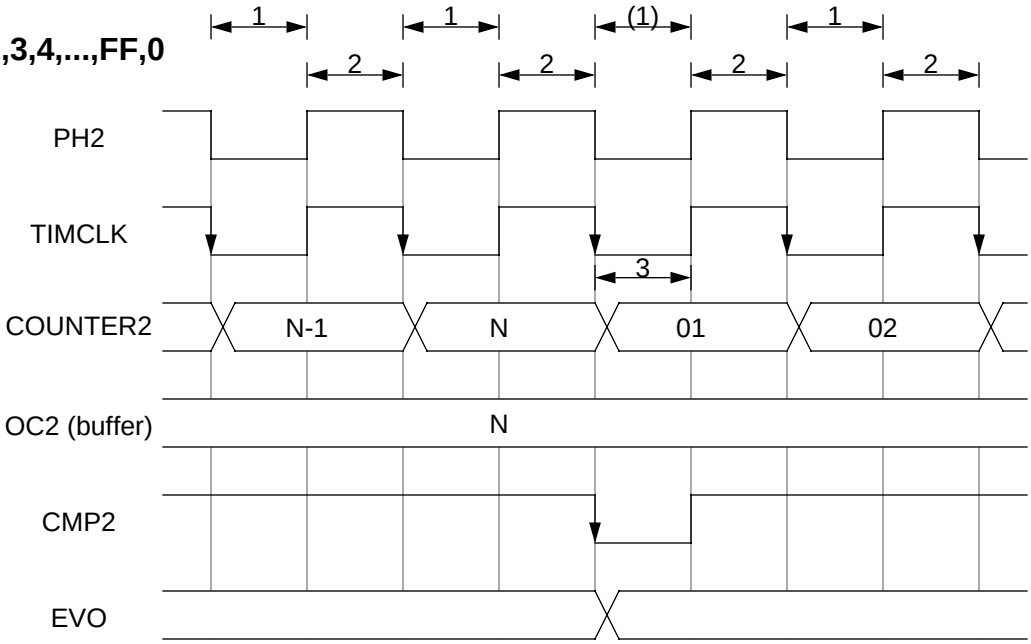


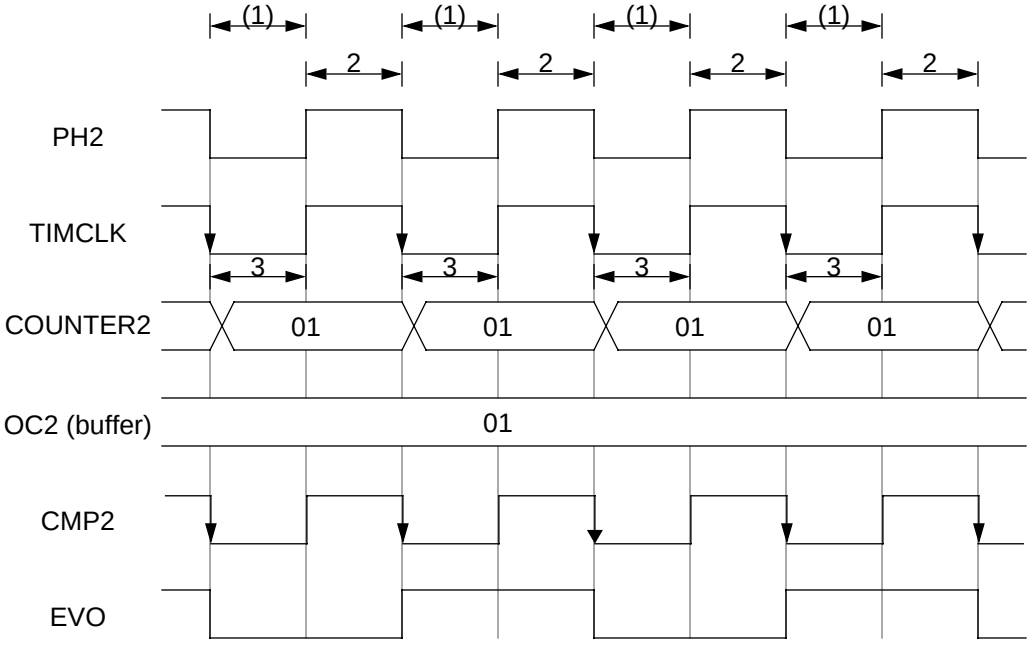
Figure 9-8. Timer 2 Timing Diagram for f(PH2) > f(TIMCLK)

OC2=2,3,4,...,FF,0



1. COUNT UP
2. COMPARE
3. PRESET (that overrides COUNT UP)

OC2=1



1. COUNT UP
2. COMPARE
3. PRESET (that overrides COUNT UP)

Figure 9-9. Timer 2 Timing Diagram for f(PH2) = f(TIMCLK)

9.2.1 Timer Control Register 2 (TCR2)

		7	6	5	4	3	2	1	0
TCR2 $0001C	R	TI2IE	OC2IE	0	T2CLK	IM2	IL2	OE2	OL2
	W								
reset ⇒		0	0	0	0	0	0	0	0

TI2IE – Timer Input 2 Interrupt Enable

TI2IE bit enables Timer Input 2 (EVI) interrupt when TI2F is set. This bit is cleared on reset.

0 = Timer Input 2 Interrupt is disabled
1 = Timer Input 2 Interrupt is enabled

OC2IE – Compare 2 Interrupt Enable

OC2IE bit enables Compare 2 (CMP2) interrupt when compare match is detected (OC2F is set). This bit is cleared on reset.

0 = Compare 2 Interrupt is disabled
1 = Compare 2 Interrupt is enabled

T2CLK – Timer 2 Clock select

The T2CLK bit selects clock source for the timer counter 2. This bit is cleared on reset.

0 = CLK2 from prescaler is selected
1 = EXCLK from EVI input block is selected

IM2 – Timer Input 2 Mode select

The IM2 bit selects whether EVI input is gated by CLK2 or not gated by CLK2. This bit is cleared on reset.

0 = EVI is not gated by CLK2 (Event Mode)
1 = EVI is gated by CLK2 (Gated Mode)

IL2 – Timer Input 2 active edge (Level) select

The IL2 bit selects the active edge of EVI to increment counter for the Event Mode (IM2 = 0), or gate enable level of EVI for the Gate Mode (IM2 = 1). This bit is cleared on reset.

0 = Falling edge is selected (Event Mode)
Low level enables counting (Gate Mode)
1 = Rising edge is selected (Event Mode)
High level enables counting (Gated Mode)

IM2	IL2	Action on Clock
0	0	EVI falling edge increments counter
0	1	EVI rising edge increments counter
1	0	Low level on EVI enables counting
1	1	High level on EVI enables counting

OE2 – Timer Output 2 (EVO) Output Enable

The OE2 bit enables EVO output on PC5 pin. When this bit is changed, control of the pin is delayed (synchronized) until the next active edge of EVO that is selected by OL2 bit occurs. This bit is cleared on reset.

0 = EVO output is disabled
1 = EVO output is enabled

OL2 – Timer Output 2 Edge select for synchronization

The OL2 bit selects which edge of EVO clock should be synchronized by the OE2 bit control. The OL2 bit also decides the initial value of the CMP2 divider, when counter 2 is written to by the CPU. This bit is cleared on reset.

0 = The falling edge of EVO switches EVO output and PC5 if OE2 bit has been changed
1 = The rising edge of EVO switches EVO output and PC5 if OE2 bit has been changed

9.2.2 Timer Status Register 2 (TSR2)

		7	6	5	4	3	2	1	0
TSR2 $001D	R	TI2F	OC2F	0	0	0	0	0	0
	W					RTI2F	ROC2F		
reset ⇒		0	0	0	0	0	0	0	0

TI2F – Timer Input 2 (EVI) Interrupt Flag

In Event mode the event edge sets TI2F and in gated time accumulation mode the trailing edge of the gate signal at the EVI input pin sets TI2F. When TI2IE bit and this bit are set, an interrupt is generated. This bit is a read only bit and writes have no effect. The TI2F is cleared by writing a 1 to the RTI2F bit and on reset.

OC2F – Compare 2 Interrupt Flag

The OC2F bit is set when the compare match is detected between counter 2 and OC2 register. When OC2IE bit and this bit are set, an interrupt is generated. This bit is a read only bit and writes have no effect. The OC2F is cleared by writing a 1 to ROC2F bit and on reset.

RTI2F – Reset Timer Input 2 Flag

The RTI2F bit is a write only bit and always read as 0. Writing 1 to this bit clears TI2F bit and writing a 0 to this bit has no effect.

ROC2F – Reset Output Compare 2 Flag

The ROC2F bit is a write only bit and always read as 0. Writing 1 to this bit clears OC2F bit and writing a 0 to this bit has no effect.

9.2.3 Output Compare Register 2 (OC2)

		7	6	5	4	3	2	1	0
OC2 $001E	R	2OC7	2OC6	2OC5	2OC4	2OC3	2OC2	2OC1	2OC0
	W								
reset ⇒		1	1	1	1	1	1	1	1

The OC2 register data is transferred to the buffer register when the CPU writes to CNT2, when the CMP2 presets the CNT2, or when system reset.

When the OC2 buffer register matches the CNT2 register, the OC2F bit in the Tsr2 Register Is Set And Cnt2 Is Preset To $01.

9.2.4 Timer Counter 2 (CNT2)

		7	6	5	4	3	2	1	0
CNT2 $001F	R	2CNT7	2CNT6	2CNT5	2CNT4	2CNT3	2CNT2	2CNT1	2CNT0
	W	Counter Preset to 1							
reset ⇒		0	0	0	0	0	0	0	1

The Timer Counter 2 (CNT2) is incremented by the falling edge of the timer clock (which is synchronized and has the same timing as falling edge of PH2).

The CNT2 register is compared with OC2 buffer register and initialized to $01 if it matches. It is also initialized to $01 on reset and any CPU write to this register.

The CPU read of this counter should be done while PH2 is high and data may be latched by the local or main data bus while PH2 is low.

9.2.5 Time Base Control Register 1 (TBCR1)

		7	6	5	4	3	2	1	0
TBCR1 $0010	R	TBCLK	0	LCLK	0	0	0	T2R1	T2R0
	W								
reset ⇒		0	0	0	0	0	0	0	0

TBCLK, LCLK

See **Section 8.4** for a detailed description.

T2R1, T2R0 – Prescale Rate select bits for Timer 2

The T2R1 and T2R0 bits select prescale rate of CLK2 for Timer 2. These bits are cleared on reset.

T2R1	T2R0	System clock divide by
0	0	1
0	1	4
1	0	32
1	1	256

9.2.6 Timer Input 2 (EVI)

The Event Input (EVI) is used as an external clock input for Timer 2.

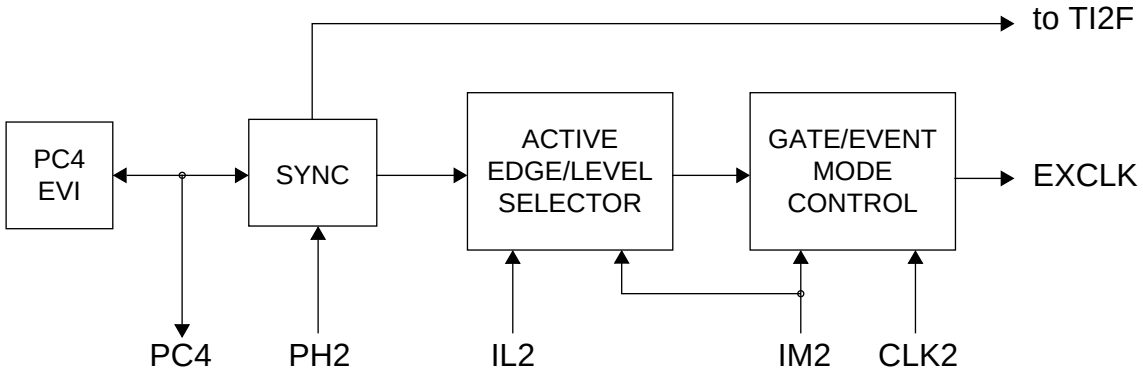


Figure 9-10. EVI Block Diagram

Since the external clock may be asynchronous to the internal clock, this input has a synchronizer which samples external clock by the internal system clock. (The input transition synchronizes to the falling edge of PH2. Therefore the minimum pulse width for EVI must be larger than one system clock to be measured.)

The IM2 and IL2 bits in the TCR2 determine how this synchronized external clock is used. IM2 bit decides between Event mode and Gated mode, and IL2 bit decides which level or edge is activated.

In the Event mode (IM2 = 0), the external clock drives the Timer 2 counter directly and the active edge at the EVI pin is selected by the IL2 bit. When active edge is detected the TI2F bit in the TCR2 is set.

In the Gated mode (IM2 = 1), the EVI input is gated by CLK2 from the prescaler and gate output drives the Timer 2 counter. IL2 bit decides active level of the external input. When the transition from active level to inactive level is detected the TI2F bit is set.

Changing the IM2 bit may cause an illegal count up of CNT2. Thus presetting CNT2 after initializing IM2 is required.

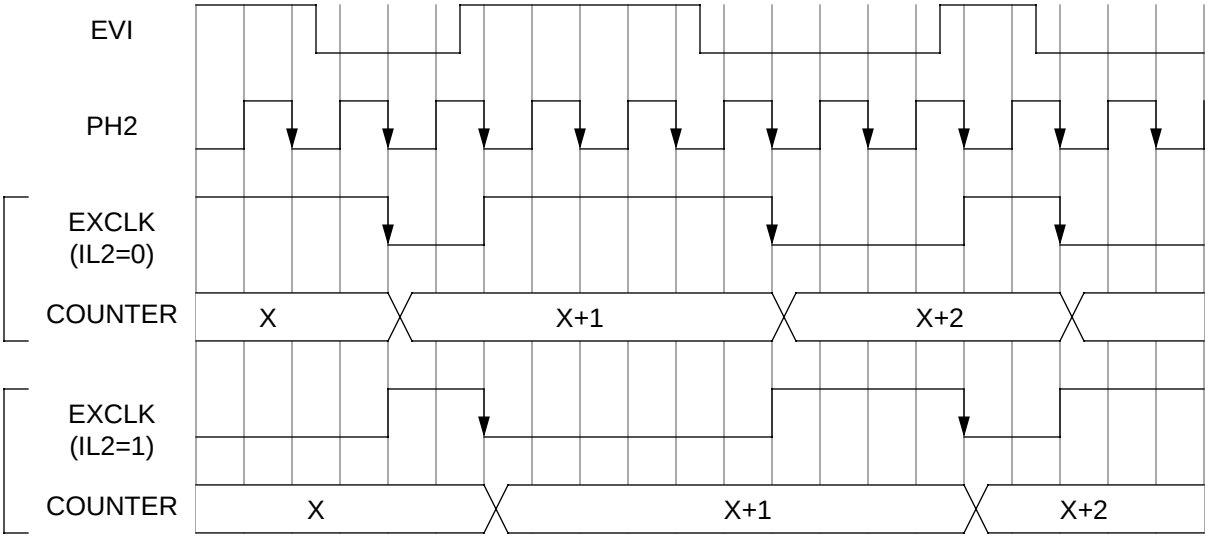
Table 9-1. EVI Mode Select

IM2	IL2	Action on Clock
0	0	EVI falling edge increments counter
0	1	EVI rising edge increments counter
1	0	Low level on EVI enables counting
1	1	High level on EVI enables counting

NOTE

Since the EVI pin is shared with the PC4 I/O pin, DDRC4 should always be cleared whenever EVI is used. EVI should not be used when DDRC4 is high.

IM2=0 (Event Mode)



IM2=1 (Gate Mode)

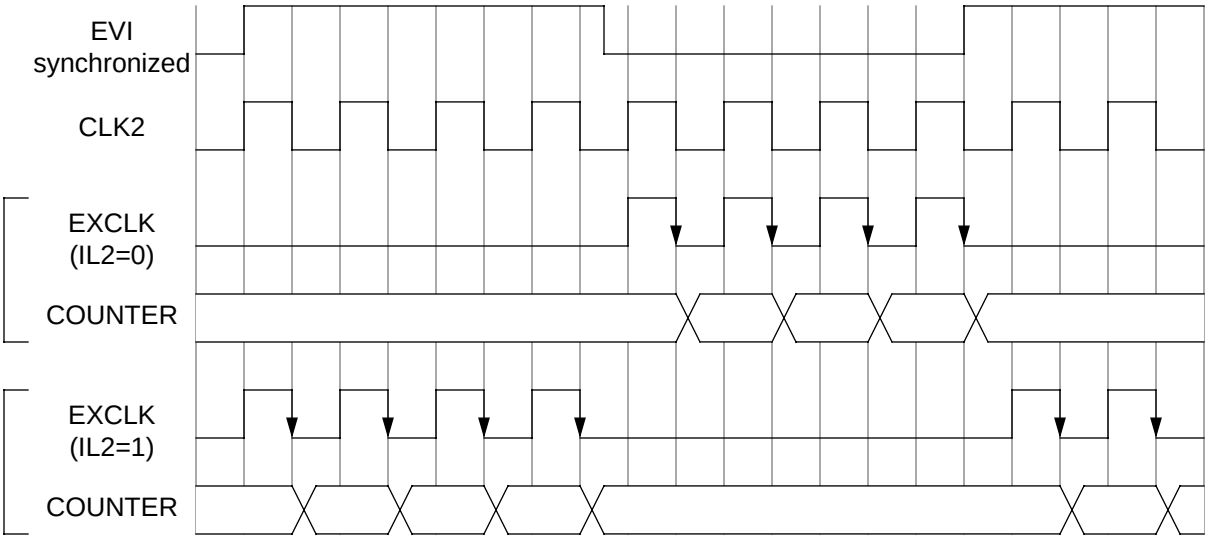


Figure 9-11. EVI Timing Diagram

9.2.7 Event Output (EVO)

The EVO pin is the clock output pin of Timer 2. The compare output from the Timer 2 (CMP2) is divided in this block for 50% duty output signal. This 1/2 divider is initialized to the level of the OL2 bit when the timer counter 2 is written to by the CPU (initialized).

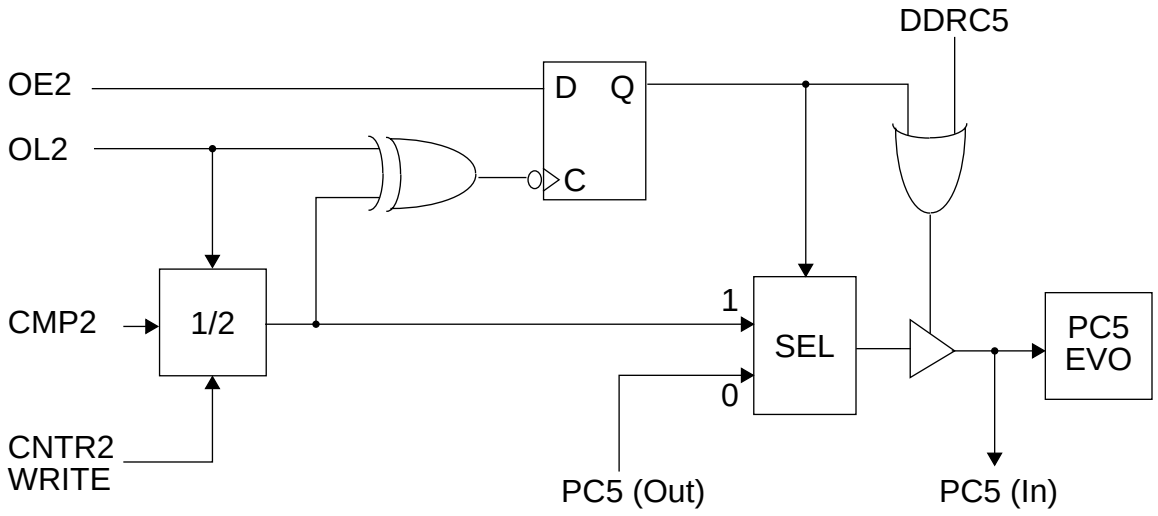


Figure 9-12. EVO Block Diagram

When the OE2 bit in the Timer Control Register 2 (TCR2) is set, the EVO output is activated and when OE2 is cleared EVO is deactivated. These controls must be done synchronously to the EVO output signal to avoid an incomplete pulse on the pin. The OL2 bit in the TCR2 decides which edge of EVO should be synchronized.

When DDRC5 bit is set or the synchronized output enable is high (clock on), the output buffer at the EVO/PC5 pin is enabled. If DDRC5 bit is set to one, the pin state during the idling condition (clock off) depends on the PC5 output data latch. If DDRC5 is cleared, the pin becomes high impedance during clock off.

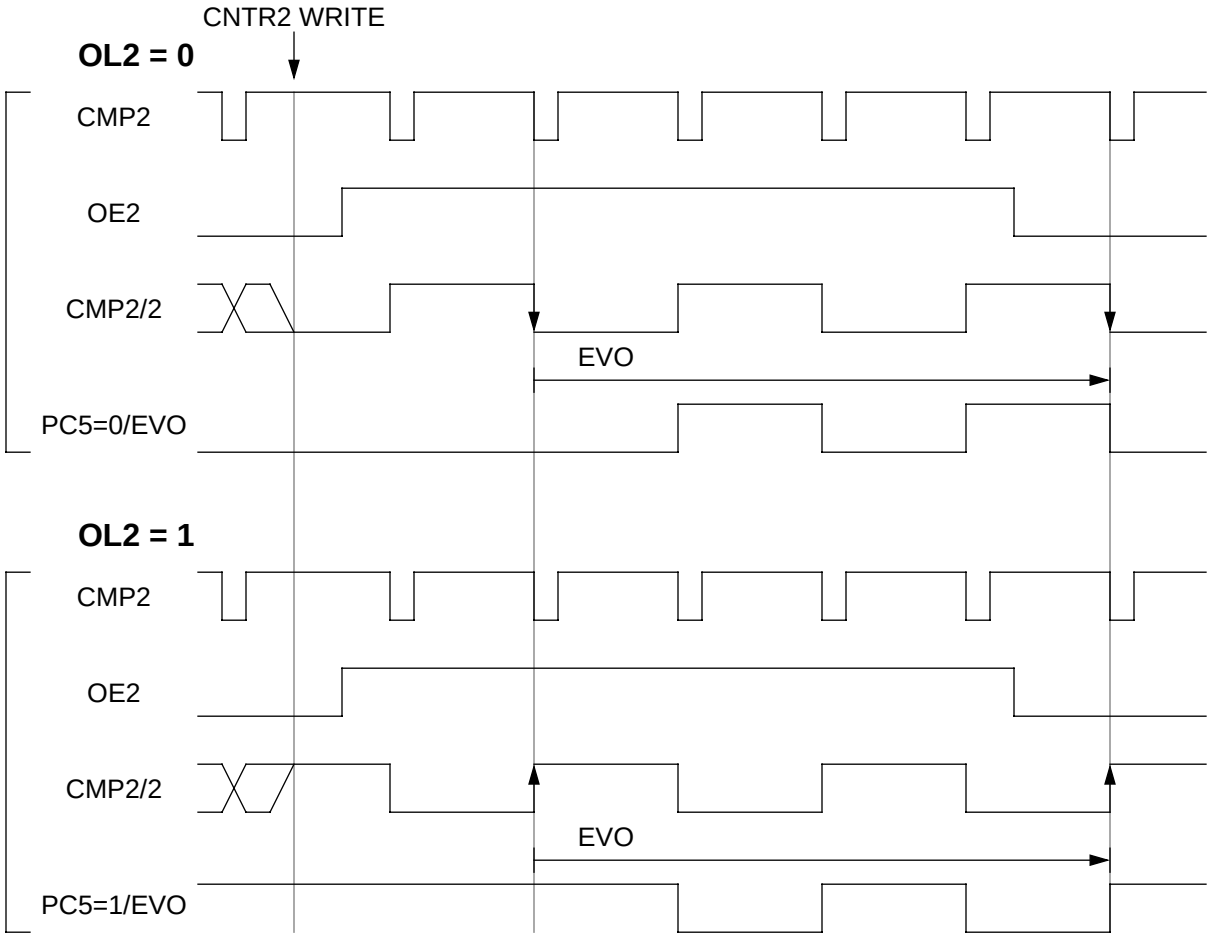


Figure 9-13. EVO Timing Diagram

9.3 PRESCALER

The 8-bit prescaler in the timer system divides system clock (PH2) and provides divided clock to each timer and Event input.

CLK1 for the Timer 1 is a fixed frequency clock (PH2/4).

CLK2 for the Timer 2 is selected by T2R1 and T2R0 bits in the TBCR1, and this clock is also used as the event input for the gate mode. The CLK2 transitions must be synchronous to the falling edge of PH2.

Table 9-2. CLK2 Divide Ratio

T2R1	T2R0	System clock divide by
0	0	1
0	1	4
1	0	32
1	1	256

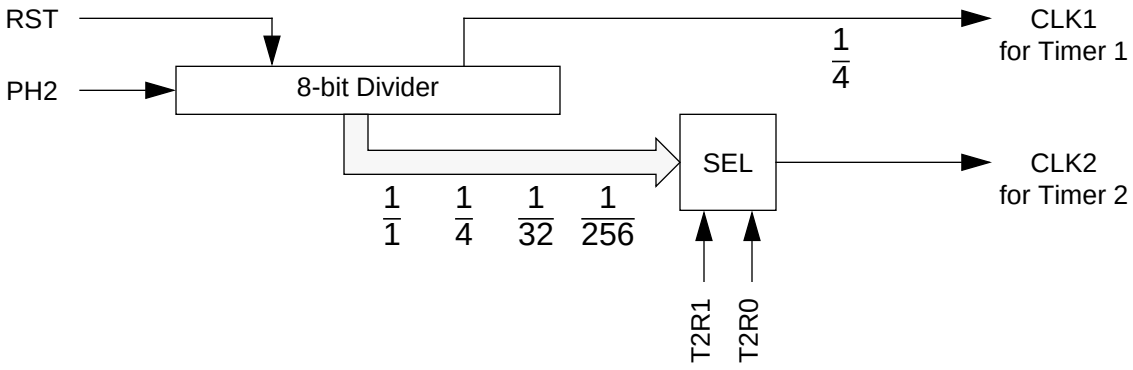


Figure 9-14. Prescaler Block Diagram

SECTION 10
LCD DRIVER

The LCD driver may be configured with 4 Backplanes (BP) and 36 Frontplanes (FP) maximum. The VDD voltage is the highest level of the output waveform and the lower three levels are VLCD1, VLCD2, and VLCD3. VLCD3 can be externally driven, while VLCD1 and VLCD2 are internally generated.

On reset, LCD enable bit (LCDE) in the LCD control register (LCDCR) is cleared (LCD drivers at a disabled state) and all BP and FP pins output Vdd level.

The LCD clock is generated by the Time Base module and LCLK bit in the TBCR1 selects clock frequency.

10.1 LCD WAVEFORM EXAMPLES

The following figures illustrate the LCD timing examples.

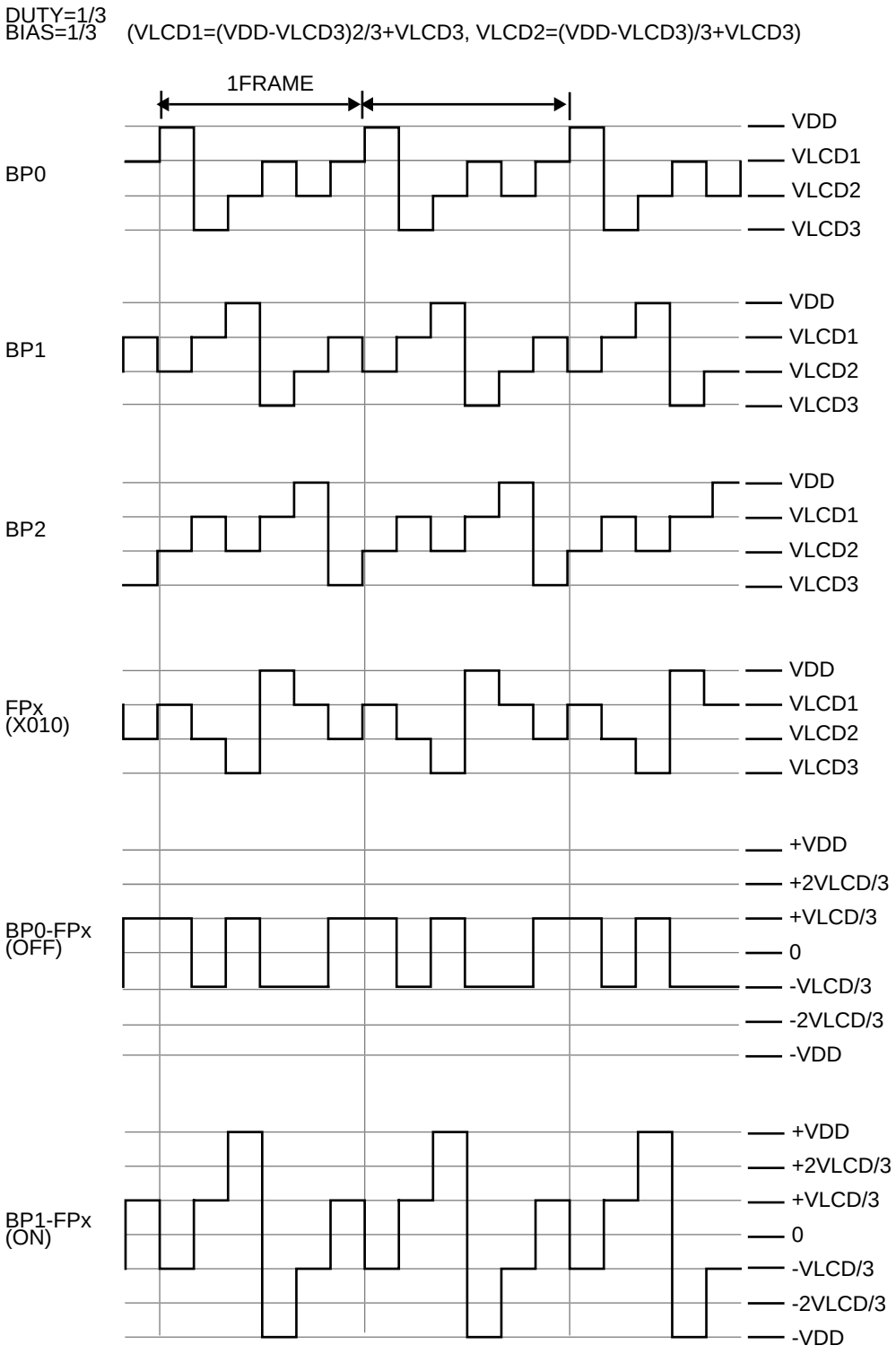


Figure 10-1. LCD 1/3 Duty and 1/3 Bias Timing Diagram

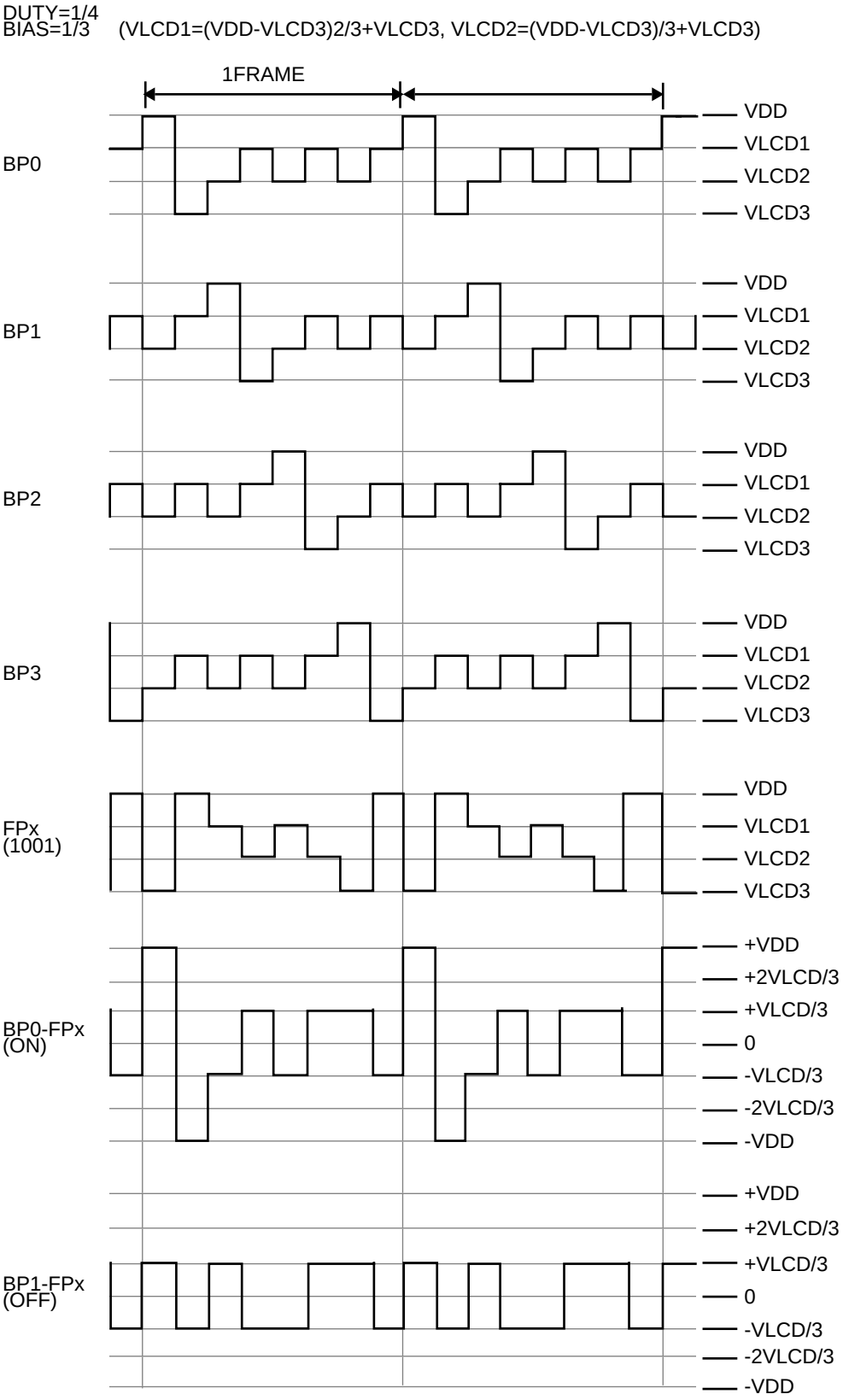


Figure 10-2. LCD 1/4 Duty and 1/4 Bias Timing Diagram

10.2 VLCD3 BIAS INPUT & BIAS RESISTORS

VLCD3 is the bias input for the MC68HC(7)05PD6 LCD module. The highest level of the output waveform is VDD voltage and the lower three levels are VLCD1, VLCD2 and VLCD3. VLCD3 is externally driven, while the other reference voltages, VLCD1 and VLCD2, are internally generated.

The MC68HC(7)05PD6 LCD module has software selectable internal bias resistors $R_{LCD1,}$ R_{LCD2} and R_{LCD3} which form a resistor ladder. This ladder is used to generate VLCD1, VLCD2 and VLCD3. The resistors are arranged such that VDD>VLCD1>VLCD2>VLCD3. Bias voltages may be adjusted using these internal resistors. Three possible values may be selected for these resistors using the FC and LC bits in the LCD Control Register at $0020. A contrast resistor, R_V, as illustrated in **Figure 10-3**, may be placed between VDD and VSS.

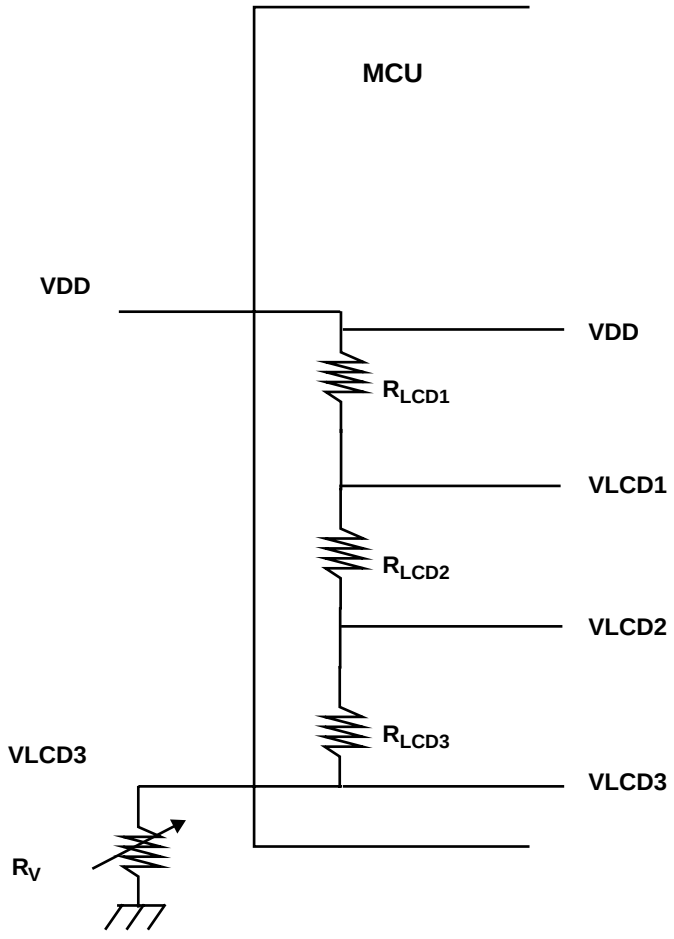


Figure 10-3. Simplified LCD Voltage Divider Schematic

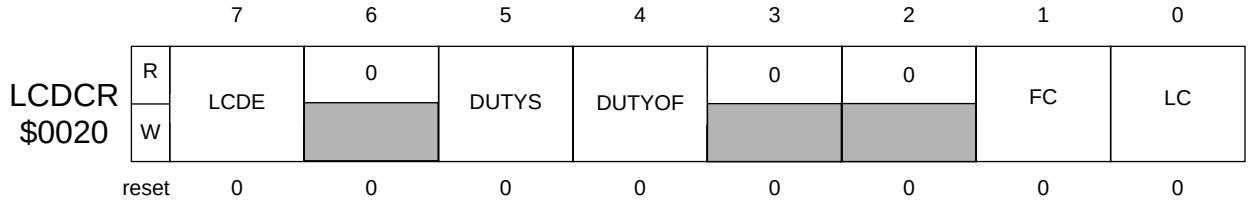
10.3 BACKPLANE DRIVER AND PORT SELECTION

The number of Backplanes (Port D) pins depends on the LCD duty. It is automatically selected by DUTY1 and DUTY0 bits in the LCD Control Register (LCDCR). On reset, these bits are cleared and 1/4 duty is selected.

Table 10-1. Backplanes and Port Selection

DUTY	LCD CONTROL	PIN SELECTION			
	DUTYS	BP3/PD3	BP2/PD2	BP1/PD1	BP0/PD0
1/3	1	PD3	BP2	BP1	BP0
1/4	0	BP3	BP2	BP1	BP0

10.4 LCD CONTROL REGISTER (LCDCR)



LCDE – LCD Output Enable

The LCDE bit enables all BP and FP outputs. This bit is cleared on reset.

0 = Ports that configure as BP and FP pins will output Vdd.
1 = all BP and FP pins output LCD waveform

DUTYS – LCD Duty Select

The DUTYS bits select the duty of LCD driver as shown in **Table 10-1**. The number of BP pins is related to this duty selection. The unused BP pin is used as Port D pin. Default duty is 1/4 duty. These bits are cleared on reset.

DUTYOF – LCD Duty Disable

0 = LCD Duty is enable. The number of BP pins is determined by DUTYS.
1 = LCD Duty is disable. BP0 - BP3 are replaced by PD0 - PD3 respectively regardless the selection of LCD duty.

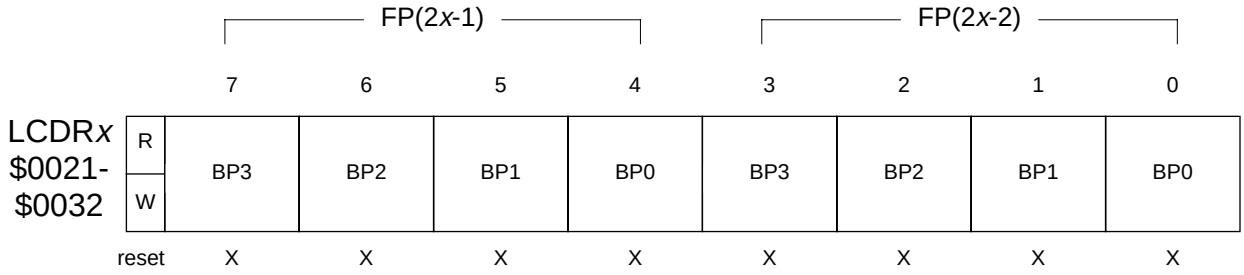
FC, LC – Fast Charge, Low Current

These bits are used to select resistor values in the voltage generator resistor chain. Reset clears these bits.

Table 10-2. LCD Bias Resistors

LC	FC	Action
0	X	Default value of typically 30kΩ per resistor
1	0	Resistor value of typically 200kΩ per resistor
1	1	Fast-charge: for period of LCD CLK/128 in each frame the resistor values are reduced to default (value for LC=0)

10.5 LCD DATA REGISTER (LCDR*x*)



Data in the LCDR*x* (LCDR1-LCDR18) controls the waveform of the two Frontplanes drivers. Bit 0 thru 3 and bit 4 thru 7 of this register decide the waveforms at the BP0 thru BP3 timings. If LCD duty is not 1/4, the register bit for the unused Backplanes have no meaning.

0 = output deselect waveform at the corresponding Backplanes timing.
1 = output select waveform at the corresponding Backplanes timing.

DUTY	FRONTPLANE DATA REGISTER BIT USAGE							
	7	6	5	4	3	2	1	0
1/3	—	BP2	BP1	BP0	—	BP2	BP1	BP0
1/4	BP3	BP2	BP1	BP0	BP3	BP2	BP1	BP0

SECTION 11
SERIAL COMMUNICATIONS INTERFACE

A full-duplex asynchronous SCI is provided with a standard NRZ format and a selection of baud rates. The SCI transmitter and receiver are functionally independent but use the same data format and baud rate. The terms baud and bit rate are used synonymously in the following description.

11.1 SCI TWO-WIRE SYSTEM FEATURES

- Standard NRZ (mark/space) format
- Advanced error detection method includes noise detection for noise duration of up to 1/16 bit time
- Full-duplex operation (simultaneous transmit and receive)
- Software programmable for different baud rates
- Software-selectable word length (eight or nine bit words)
- Separate transmitter and receiver enable bits
- SCI may be interrupt driven
- Four separate interrupt conditions

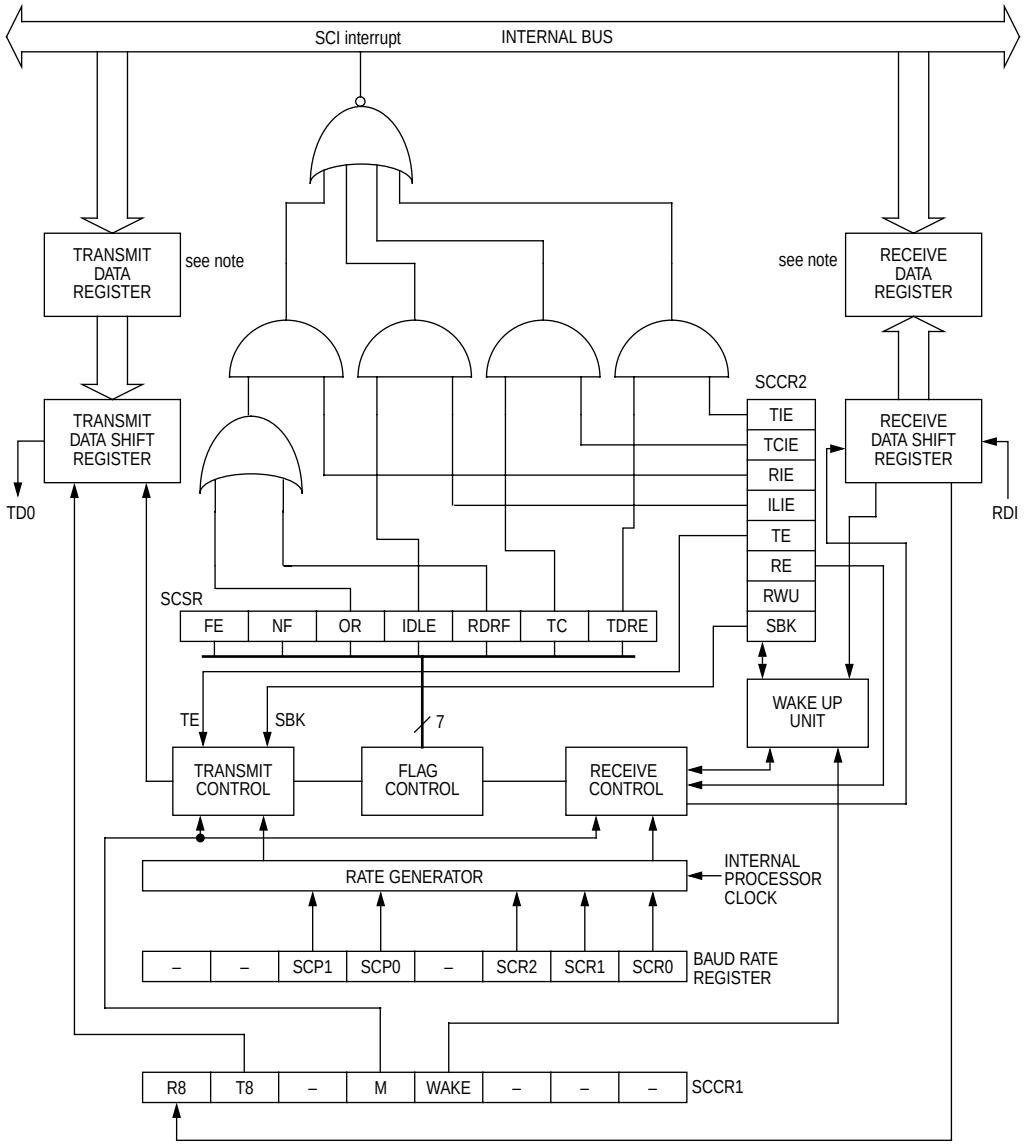
11.2 SCI RECEIVER FEATURES

- Receiver wake-up function (idle or address bit)
- Idle line detect
- Framing error detect
- Noise detect
- Overrun detect
- Receiver data register full flag

11.3 SCI TRANSMITTER FEATURES

- Transmit data register empty flag
- Transmit complete flag
- Break send

Any SCI two-wire system requires receive data in (RDI) and transmit data out (TDO).



NOTE: The serial data communications data register (SCDAT) is controlled by the R/$\overline{W}$ signal. It is the transmit data register when written and receive data register when read.

Figure 11-1. Serial Communications Interface Block Diagram

11.4 DATA FORMAT

Receive data in (RDI) or transmit data out (TDO) is the serial data presented between the internal data bus and the output pin (TDO) and between the input pin (RDI) and the internal data bus. Data format is as shown for the NRZ in **Figure 11-2** and must meet the following criteria:

1. A high level indicates a logic one and a low level indicates a logic zero.
2. The idle line is in a high (logic one) state prior to transmission/reception of a message.
3. A start bit (logic zero) is transmitted/received indicating the start of a message.
4. The data is transmitted and received least-significant-bit first.
5. A stop bit (high in the tenth or eleventh bit position) indicates the byte is complete.
6. A break is defined as the transmission or reception of a low (logic zero) for some multiple of the data format.

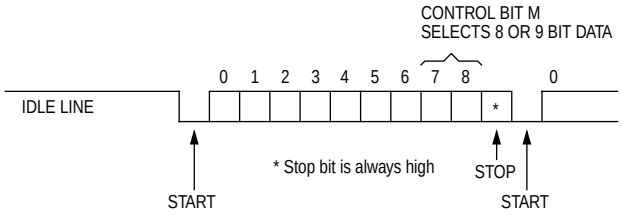


Figure 11-2. Data Format

11.5 WAKE-UP FEATURE

In a typical multiprocessor configuration, the software protocol will usually identify the addressee(s) at the beginning of the message. To permit uninterested MPUs to ignore the remainder of the message, a wake-up feature is included, whereby all further SCI receiver flag (and interrupt) processing can be inhibited until its data line returns to the idle state. An SCI receiver is re-enabled by an idle string of at least ten (or eleven) consecutive ones. Software for the transmitter must provide for the required idle string between consecutive messages and prevent it from occurring within messages.

A second wake-up method is available in which sleeping SCI receivers can be awakened by a logic one in the high-order bit of a received character.

11.6 RECEIVE DATA IN (RDI)

Receive data in (RDI) is the serial data which is presented from the input pin via the SCI to the receive data register (RDR). While waiting for a start bit, the receiver samples the input at a rate 16 times higher than the set baud rate. This increased rate is referred to as the RT rate. When the input (idle) line is detected low, it is tested for three more sample times. If at least two of these three samples detect a logic low, a valid start bit is assumed to be detected. If in two or more samples, a logic high is detected, the line is assumed to be idle. The receive clock generator is controlled by the baud rate register (see **Section 11.9.5**); however, the SCI is synchronized by the start bit independent of the transmitter.

Once a valid start bit is detected, the start bit, each data bit, and the stop bit are each sampled three times. The value of the bit is determined by voting logic, which takes the value of a majority of samples. A noise flag is set when all three samples on a valid start bit, data bit, or stop bit do not agree. A noise flag is also set when the start verification samples do not agree.

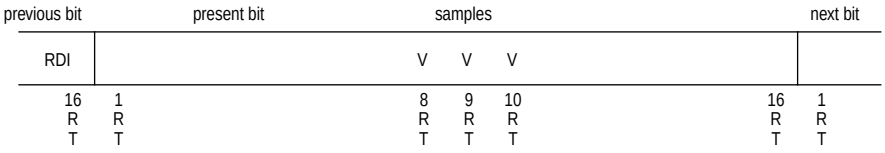


Figure 11-3. Sampling Technique used on All bits

11.7 START BIT DETECTION FOLLOWING A FRAMING ERROR

If there has been a framing error (FE) without detection of a break (10 zeros for 8-bit format or 11 zeros for a 9-bit format), the circuit continues to operate as if there actually were a stop bit, and the start edge will be placed artificially. The last bit received in the data shift register is inverted to a logic one, and the three logic-one start qualifiers (shown in **Figure 11-4**) are forced into the sample shift register during the interval when detection of a start bit is anticipated (see **Figure 11-5**); therefore, the start bit will be accepted no sooner than it is anticipated.

If the receiver detects that a break (RDRF=1, FE=1, receiver data register=$00) produced the framing error, the start bit will not be artificially induced, and the receiver must actually receive a logic one before start. See **Figure 11-6**.

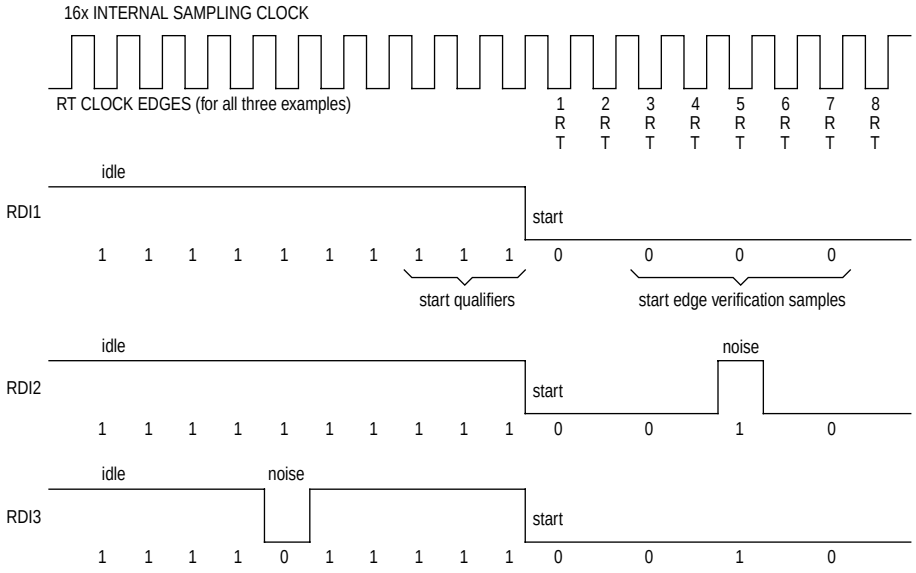


Figure 11-4. Example of Start-Bit Sampling Technique

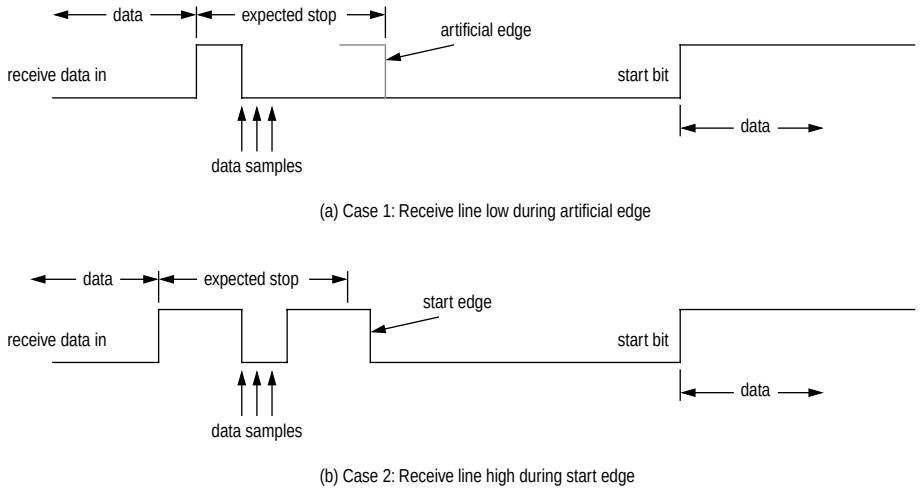


Figure 11-5. SCI Artificial Start following a Framing Error

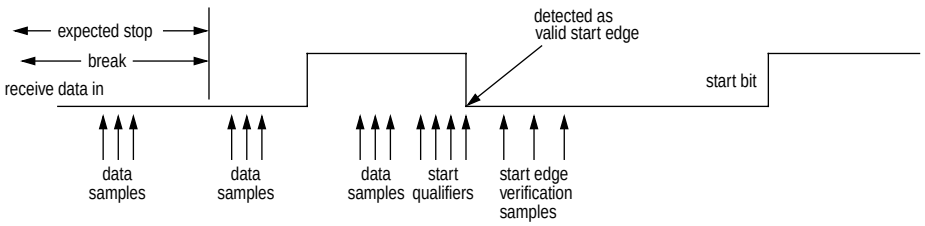


Figure 11-6. SCI Start following a Break

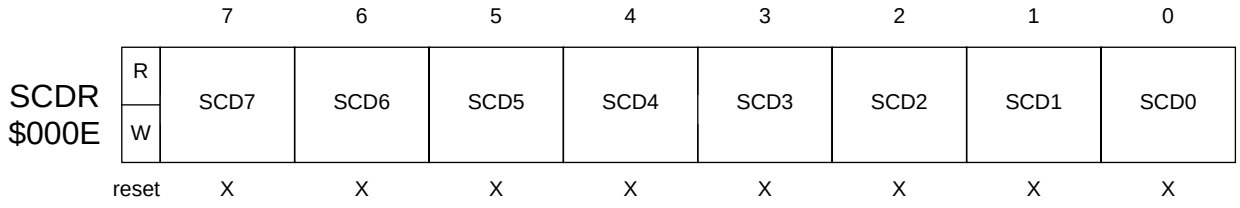
11.8 TRANSMIT DATA OUT (TDO)

Transmit data out is the serial data which is presented from the internal data bus via the SCI and then to the output pin. Data format is as discussed above and shown in **Figure 11-2**. The transmitter generates a bit time by using a derivative of the RT clock, thus producing a transmission rate equal to 1/16 that of the receiver sample clock.

11.9 SCI REGISTERS

There are five registers used in the SCI; the internal configuration of these registers is discussed in the following paragraphs. A block diagram of the SCI system is shown in **Figure 11-1**.

11.9.1 Serial Communications Data Register (SCDAT)



The serial communications data register performs two functions in the serial communications interface; i.e. it acts as the receive data register when it is read and as the transmit data register when it is written. **Figure 11-1** shows this register as two separate registers, namely: the receive data register (RDR) and the transmit data register (TDR). As shown in **Figure 11-1**, the TDR (transmit data register) provides the parallel interface from the internal data bus to the transmit shift register and the receive data register (RDR) provides the interface from the receive shift register to the internal data bus.

When SCDAT is read, it becomes the receive data register and contains the last byte of data received. The receive data register, represented above, is a read-only register containing the last byte of data received from the shift register for the internal data bus. The RDRF bit (receive data register full bit in the serial communications status register) is set to indicate that a byte has been transferred from the input serial shift register to the serial communications data register. The transfer is synchronized with the receiver bit rate clock (from the receive control) as shown in **Figure 11-1**. All data is received least-significant-bit first.

When SCDAT is written, it becomes the transmit data register and contains the next byte of data to be transmitted. The transmit data register, also represented above, is a write-only register containing the next byte of data to be applied to the

transmit shift register from the internal data bus. As long as the transmitter is enabled, data stored in the serial communications data register is transferred to the transmit shift register (after the current byte in the shift register has been transmitted). The transfer from the SCDAT to the transmit shift register is synchronized with the bit rate clock (from the transmit control) as shown in **Figure 11-1**. All data is transmitted least-significant-bit first.

11.9.2 Serial Communications Control Register 1 (SCCR1)

		7	6	5	4	3	2	1	0
SCCR1 $000B	R	R8	T8		M	WAKE			
	W								
reset		X	X	-	X	X	-	-	-

The serial communications control register 1 (SCCR1) provides the control bits which: 1) determine the word length (either 8 or 9 bits), and 2) selects the method used for the wake-up feature. Bits 6 and 7 provide a location for storing the ninth bit for longer bytes.

R8

If the M bit is a one, then this bit provides a storage location for the ninth bit in the receive data byte. Reset does not affect this bit.

T8

If the M bit is a one, then this bit provides a storage location for the ninth bit in the transmit data byte. Reset does not affect this bit.

M

The option of the word length is selected by the configuration of this bit and is shown below. Reset does not affect this bit.

0 = 1 start bit, 8 data bits, 1 stop bit
1 = 1 start bit, 9 data bits, 1 stop bit

WAKE

This bit allows the user to select the method for receive “wake up”. If the WAKE bit is a logic zero, an idle line condition will “wake up” the receiver. If the WAKE bit is set to a logic one, the system acknowledges an address bit (most significant bit). The address bit is dependent on both the WAKE bit and the M bit level (table shown below). (Additionally, the receiver does not use the wake-up feature unless the RWU control bit in serial communications control register 2 is set as discussed below.) Reset does not affect this bit.

WAKE	M	METHOD OF RECEIVER 'WAKEUP'
0	X	Detection of an idle line allows the next data byte received to cause the receive data register to fill and produce an RDRF.
1	0	Detection of a received one in the eight data bit allows an RDRF flag and associated error flags.
1	1	Detection of a received one in the ninth data bit allows an RDRF flag and associated error flag.

11.9.3 Serial Communications Control Register 2 (SCCR2)

		7	6	5	4	3	2	1	0
SCCR2 $000C	R	TIE	TCIE	RIE	ILIE	TE	RE	RWU	SBK
	W								
	reset	0	0	0	0	0	0	0	0

The serial communications control register 2 (SCCR2) provides the control bits which: individually enable/disable the transmitter or receiver, enable the system interrupts, and provide the wake-up enable bit and a "send break code" bit. Each of these bits is described below. (The individual flags are discussed in the **Section 11.9.4**.)

TIE

When the transmit interrupt enable bit is set, the SCI interrupt occurs provided TDRE is set (see **Figure 11-1**). When TIE is cleared, the TDRE interrupt is disabled. Reset clears the TIE bit.

TCIE

When the transmission complete interrupt enable bit is set, the SCI interrupt occurs provided TC is set (see **Figure 11-1**). When TCIE is clear, the TC interrupt is disabled. Reset clears the TCIE bit.

RIE

When the receive interrupt enable bit is set, the SCI interrupt occurs provided OR is set or RDRF is set (see **Figure 11-1**). When RIE is cleared, the OR and RDRF interrupts are disabled. Reset clears the RIE bit.

ILIE

When the idle line interrupt enable bit is set, the SCI interrupt occurs provided IDLE is set (see **Figure 11-1**). When ILIE is cleared, the IDLE interrupt is disabled. Reset clears the ILIE bit.

TE

When the transmit enable bit is set, the transmit shift register output is applied to the TDO line. Depending on the state of control bit M in serial communications control register 1, a preamble of 10 (M=0) or 11 (M=1) consecutive ones is transmitted when software sets the TE bit from a cleared state. If a transmission is in progress, and TE is written to a zero, then the transmitter will wait until after the present byte has been transmitted before placing the TDO pin in the idle high-impedance state. If the TE bit has been written to a zero and then set to a one before the current byte is transmitted, the transmitter will wait until that byte is transmitted and will then initiate transmission of a new preamble. After the preamble is transmitted, and provided the TDRE bit is set (no new data to transmit), the line remains idle (driven high while TE = 1); otherwise, normal transmission occurs. This function allows the user to “neatly” terminate a transmission sequence. After loading the last byte in the serial communications data register and receiving the interrupt from TDRE, indicating the data has been transferred into the shift register, the user should clear TE. The last byte will then be transmitted and the line will go idle (high impedance). Reset clears the TE bit.

RE

When the receive enable bit is set, the receiver is enabled. When RE is cleared, the receiver is disabled and all of the status bits associated with the receiver (RDRF, IDLE, OR, NF, and FE) are inhibited. Reset clears the RE bit.

RWU

When the receiver wake-up bit is set, it enables the “wake up” function. The type of “wake up” mode for the receiver is determined by the WAKE bit discussed above (in the SCCR1). When the RWU bit is set, no status flags will be set. Flags which were set previously will not be cleared with RWU is set. If the WAKE bit is cleared, RWU is cleared after receiving 10 (M=0) or 11 (M=1) consecutive ones. Under these conditions, RWU cannot be set if the line is idle. If the WAKE bit is set, RWU is cleared after receiving an address bit. The RDRF flag will then be set and the address byte will be stored in the receiver data register. Reset clears the RWU bit.

SBK

When the send break bit is set the transmitter sends zeros in some number equal to a multiple of the data format bits. If the SBK bit is toggled set and clear, the transmitter sends 10 (M=0) or 11 (M=1) zeros and then reverts to idle or sending data. The actual number of zeros sent when SBK is toggled depends on the data format set by the M bit in the serial communications control register 1; therefore, the break code will be synchronous with respect to the data stream. At the completion of the break code, the transmitter sends at least one high bit to guarantee recognition of a valid start bit. Reset clears the SBK bit.

11.9.4 Serial Communications Status Register (SCSR)

		7	6	5	4	3	2	1	0
SCSR $000D	R	TDRE	TC	RDRF	IDLE	OR	NF	FE	
	W								
reset		1	1	0	0	0	0	0	-

The serial communications status register (SCSR) provides inputs to the interrupt logic circuits for generation of the SCI system interrupt. In addition, a noise flag bit and a framing error bit are also contained in the SCSR.

TDRE

The transmit data register empty bit is set to indicate that the contents of the serial communications data register have been transferred to the transmit serial shift register. If the TDRE bit is clear, it indicates that the transfer has not yet occurred and a write to the serial communications data register will overwrite the previous value. The TDRE bit is cleared by accessing the serial communications status register (with TDRE set), followed by writing to the serial communications data register. Data can not be transmitted unless the serial communications status register is accessed before writing to the serial communications data register to clear the TDRE flag bit. Reset sets the TDRE bit.

TC

The transmit complete bit is set at the end of a data frame, preamble, or break condition if:

1. TE = 1, TDRE = 1, and no pending data, preamble, or break is to be transmitted; or

2. TE = 0, and the data, preamble, or break (in the transmit shift register) has been transmitted.

The TC bit is a status flag which indicates that one of the above conditions has occurred. The TC bit is cleared by accessing the serial communications status register (with TC set), followed by writing to the serial communications data register. It does not inhibit the transmitter function in any way. Reset sets the TC bit.

RDRF

When the receive data register full bit is set, it indicates that the receiver serial shift register is transferred to the serial communications data register. If multiple errors are detected in any one received word, the NF, FE, and RDRF bits will be affected as appropriate during the same clock cycle. The RDRF bit is cleared when the serial communications status register is accessed (with RDRF set) followed by a read of the serial communications data register. Reset clears the RDRF bit.

IDLE

When the idle line detect bit is set, it indicates that a receiver idle line is detected (receipt of a minimum number of ones to constitute the number of bits in the byte format). The minimum number of ones needed will be 10 (M=0) or 11 (M=1). This allows a receiver that is not in the wake-up mode to detect the end of a message, detect the preamble of a new message, or to resynchronize with the transmitter. The IDLE bit is cleared by accessing the serial communications status register (with IDLE set) followed by a read of the serial communications data register. The IDLE bit will not be set again until after an RDRF has been set; i.e., a new idle line occurs. The IDLE is not set by an idle line when the receiver “wakes up” from the wake-up mode. Reset clears the IDLE bit.

OR

When the overrun error bit is set, it indicates that the next byte is ready to be transferred from the receive shift register to the serial communications data register when it is already full (RDRF it is set). Data transfer is then inhibited until the RDRF bit is cleared. Data in the serial communications data register is valid in this case, but additional data received during an overrun condition (including the byte causing overrun) will be lost. The OR bit is cleared when the serial communications status register is accessed (with OR set), followed by a read of the serial communications data register. Reset clears the OR bit.

NF

The noise flag bit is set if there is noise on a “valid” start bit or if there is noise on any of the data bits or if there is noise on the stop bit. It is not set by noise on the idle line nor by invalid (false) start bits. If there is noise, the NF bit is not set until the RDRF flag is set. Each data bit is sampled three times as described above in RECEIVE DATA IN and shown in Fig. 5-3. The NF bit represents the status of the byte in the serial communications data register. For the byte being received (shifted in) there will also be a “working” noise flag the value of which will be transferred to the NF bit when the serial data is loaded into the serial communications data register. The NF bit does not generate an interrupt because the RDRF bit gets set with NF and can be used to generate the interrupt. The NF bit is cleared when the serial communications status register is accessed (with NF set), followed by a read of the serial communications data register. Reset clears the NF bit.

FE

The framing error bit is set when the byte boundaries in the bit stream are not synchronized with the receiver bit counter (generated by a “lost” stop bit). The byte is transferred to the serial communications data register and the RDRF bit is set. The FE bit does not generate an interrupt because the RDRF bit is set at the same time as FE and can be used to generate the interrupt. Note that if the byte received causes a framing error and it will also cause an overrun if transferred to the serial communications data register, then the overrun bit will

be set, but not the framing error bit, and the byte will not be transferred to the serial communications data register. The FE bit is cleared when the serial communications status register is accessed (with FE set) followed by a read of the serial communications data register. Reset clears the FE bit.

11.9.5 Baud Rate Register

		7	6	5	4	3	2	1	0
SCBRR $000A	R	0	0	SCP1	SCP0	0	SCR2	SCR1	SCR0
	W								
reset		0	0	0	0	0	U	U	U

The baud rate register provides the means for selecting different baud rates which may be used as the rate control for the transmitter and receiver. The SCP0-SCP1 bits function as a prescaler for the SCR0-SCR2 bits. Together, these five bits provide multiple, baud rate combinations for a given internal processor clock frequency.

SCP0, SCP1

These two bits in the baud rate register are used as a prescaler to increase the range of standard baud rates controlled by the SCR0-SCR2 bits. A table of the prescaler internal processor clock division versus bit levels is provided below. Reset clears SCP1-SCP0 bit (divide-by-one).

SCP1	SCP0	INTERNAL PROCESSOR CLOCK DIVIDE BY
0	0	1
0	1	3
1	0	4
1	1	13

SCR2, SCR1, SCR0

These three bits in the baud rate register are used to select the baud rates of both the transmitter and receiver. A table of baud rates versus bit levels is shown below. Reset does not affect the SCR2-SCR0 bits.

The diagram of **Figure 11-7** and **Table 11-1** and **Table 11-2** illustrate the divided chain used to obtain the baud rate clock (transmit clock). Note that there is a fixed rate divide-by-16 between the receive clock (RT) and the transmit clock (Tx). The actual divider chain is controlled by the combined SCP0-SCP1 and SCR0-SCR2 bits in the baud rate register as illustrated. All divided frequencies shown in the first table represent the final transmit clock (the actual baud rate) resulting from the internal processor clock division shown in the “divide-by” column only

SCR2	SCR1	SCR0	PRESCALER OUTPUT DIVIDE BY
0	0	0	1
0	0	1	2
0	1	0	4
0	1	1	8
1	0	0	16
1	0	1	32
1	1	0	64
1	1	1	128

(prescaler division only). The second table illustrates how the prescaler output can be further divided by action of the SCI select bits (SCR0-SCR2). For example, assume that a 9600 Hz baud rate is required with a 4 MHz oscillator frequency. In this case the prescaler bits (SCP0-SCP1) could be configured as a divide-by-thirteen. If a divide-by-thirteen prescaler is used, then the SCR0-SCR2 bits must be configured as a divide-by-one. This results in a divide-by-one of the internal processor clock to produce a 9600 Hz baud rate clock.

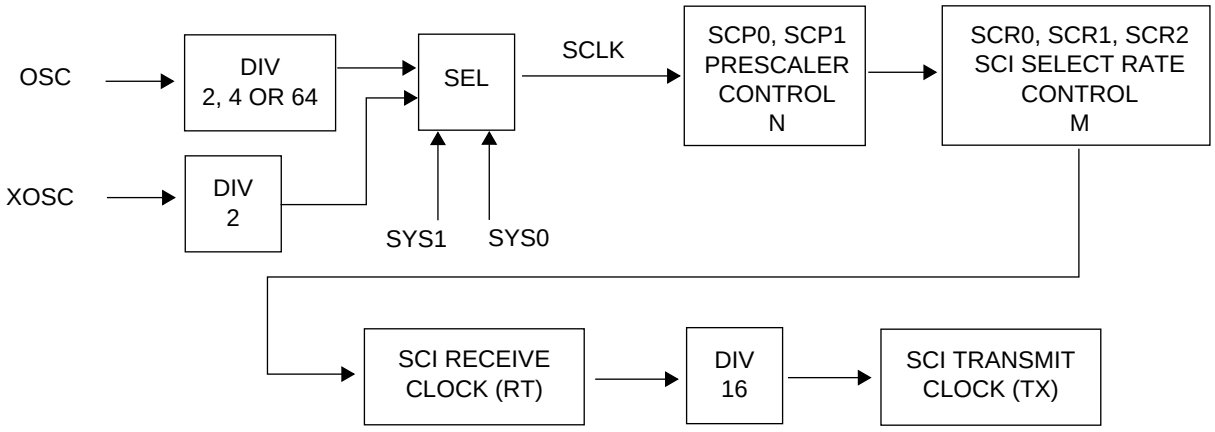


Figure 11-7. Rate Generator Division

Table 11-1. Prescaler Highest Baud Rate Frequency Output

SCP1	SCP0	CLOCK* DIVIDED BY	SYSTEM CLOCK (SCLK)			
			2M	1M	62.5K	38.4
0	0	1	125 KHz	62.5KHz	3906 Hz	2400 Hz
0	1	3	41.666 KHz	20.833 KHz	1302 Hz	800 Hz
1	0	4	31.250 KHz	15.625 KHz	977 Hz	600 Hz
1	1	13	9600 Hz	4800 Hz	300 Hz	185 Hz

* This clock is the system clock (SCLK) shown in **Figure 11-7**.

NOTE

The divided frequencies shown in **Table 11-1** represent baud rates which are the highest transmit baud rate (Tx) that can be obtained by a specific clock frequency and only using the prescaler division. Lower baud rate may be obtained by providing a further division using the SCI rate select bits as shown below for some representative prescaler outputs

Table 11-2. Transmit Baud Rate Output For a Given Prescaler Output

SCR2	SCR1	SCR0	DIVIDED BY	REPRESENTATIVE HIGHEST PRESCALER BAUD RATE OUPUT			
				125.000 KHz	39.06 KHz	15.625 KHz	9.600KHz
0	0	0	1	125.000 KHz	39.06 KHz	15.625KHz	9600Hz
0	0	1	2	62.5 KHz	19.53 KHz	7.813 KHz	4800Hz
0	1	0	4	31.250 KHz	9.765 KHz	3.906 KHz	2400Hz
0	1	1	8	15.625 KHz	4.883 KHz	1.953 KHz	1200Hz
1	0	0	16	7.813 KHz	2.441 KHz	977Hz	600Hz
1	0	1	32	3.906 KHz	1.221 KHz	488 Hz	300Hz
1	1	0	64	1.953 KHz	610 Hz	244 Hz	150Hz
1	1	1	128	977 Hz	305 Hz	122 Hz	75Hz

NOTE

Table 11-2 illustrates how the SCI select bits can be used to provide lower transmitter baud rates by further dividing the prescaler output frequency. The four examples are only representative samples. In all cases, the baud rates shown are transmit baud rates (transmit clock) and the receiver clock is 16 times higher in frequency than the actual baud rate.

SECTION 12
P-DECODER

The P-Decoder is fully compatible with CCIR Radiopaging Code Number 1 (recommendation 584). The architecture allows for flexible application in a wide variety of Radiopager designs. It can receive messages containing tone, numeric and character data at 512bps, 1200bps and 2400bps data speed.

12.1 P-DECODER FEATURES

- Fully compatible with CCIR Radiopaging Code No.1
- Programmable data rates: 512, 1200 and 2400 bps (bits per second)
- Software programmable data polarity
- Up to 6 user addresses capacity
- Two independent frame addresses
- Battery saving operation
- Programmable battery saving time
- Up to 2 bits error correction ability in both address and message codewords with 10-bit BCH (Bose-Chaudhuri-Hocquenghem) codes

12.2 CCIR Radiopaging Code No.1

The transmission of data using CCIR Radiopaging Code No.1 is shown in **Figure 12-1**. It consists of a preamble followed by batches of complete codewords, each batch commencing with a synchronization codeword (SC). Transmission may cease at the end of a batch when there are no further calls.

12.2.1 Preamble

Each transmission starts with a preamble to help the pagers attain bit synchronization and thus help in acquiring word and batch synchronization. The preamble is a pattern of bit reversals, 10101... repeated for a period of at least 576 bits, i.e. the duration of a batch plus a codeword.

12.2.2 Batch Structure

Codewords are structured in batches which comprise a synchronization codeword followed by 8 frames with each frame containing 2 codewords. The frames are numbered 0 to 7 and the pager population is divided into 8 groups. Thus each pager is allocated to one of the 8 frames according to the 3 least significant bits (LSB) of its 21 bit identity, i.e. 000 = frame 0, 111 = frame 7, and only examines address codewords in that frame. Therefore each pager's address codewords must only be transmitted in the frame that is allocated to it.

Message codewords for any pager may be transmitted in any frame but will follow, directly, the associated address codeword. A message may consist of any number of codewords transmitted consecutively and may span one or more batches.

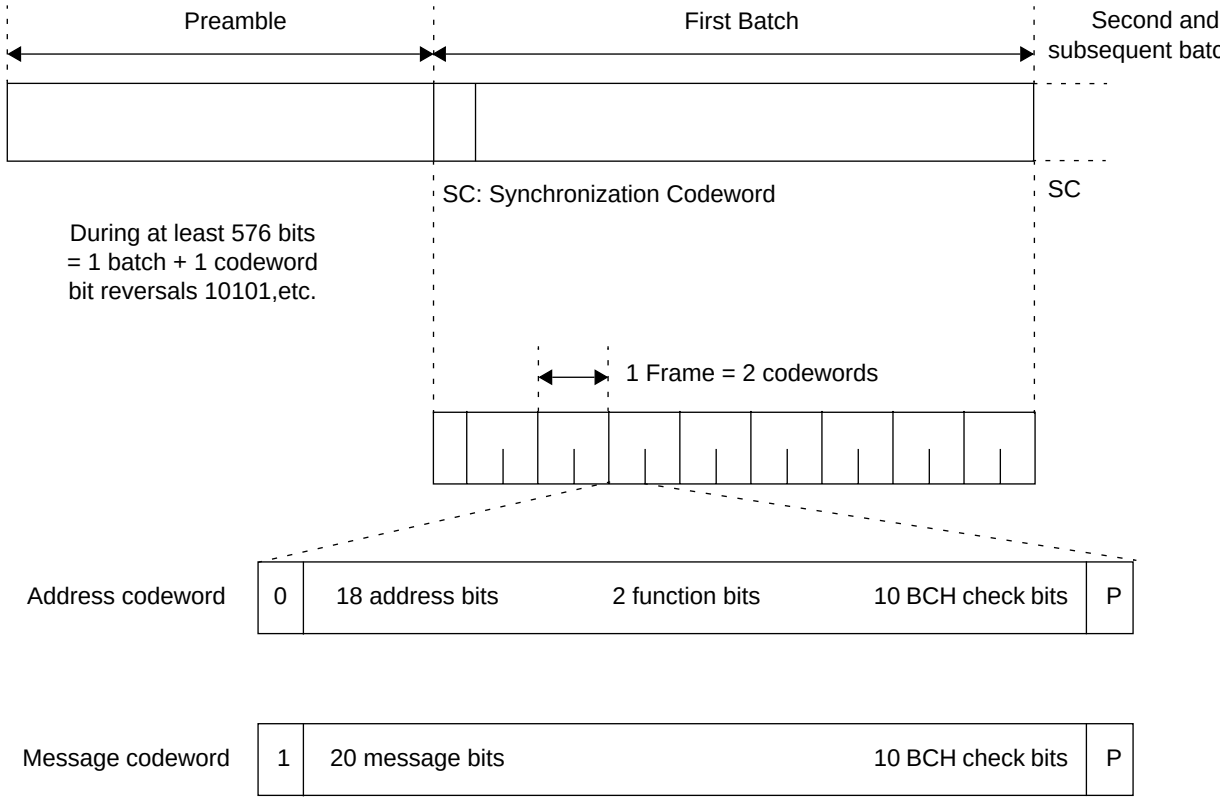


Figure 12-1. CCIR Radiopaging Code No.1 Format

12.2.3 Codeword

A frame consists of two codewords, each 32 bits long. A codeword is either an address, a message or an idle codeword. Idle codewords are transmitted to fill empty batches or to separate messages.

An address codeword always starts with a ‘0’. It has 18 bits of the 21 bit user address (bit 2 to 19) which is protected against transmission errors by 10 bits of BCH check bits (bit 22 to 31). The missing three bits of user address are coded in the frame number in the batch, in which the address codeword is transmitted. Two function bits (bit 20 and 21) allow for distinguishing one of the four different calls to the same user address. The final bit (bit 32) is chosen to give even parity.

A message codeword always starts with a ‘1’ and the whole message always directly follows the address codeword. In a message codeword, 20 bits of any display information can be put into the message bits (bit 2 to 21).

12.3 FUNCTIONAL BLOCKS

The P-Decoder consists of a clock divider, power saving generator, check bits decoder and corrector, preamble and synchronization code detector. The functional block is shown in **Figure 12-2**.

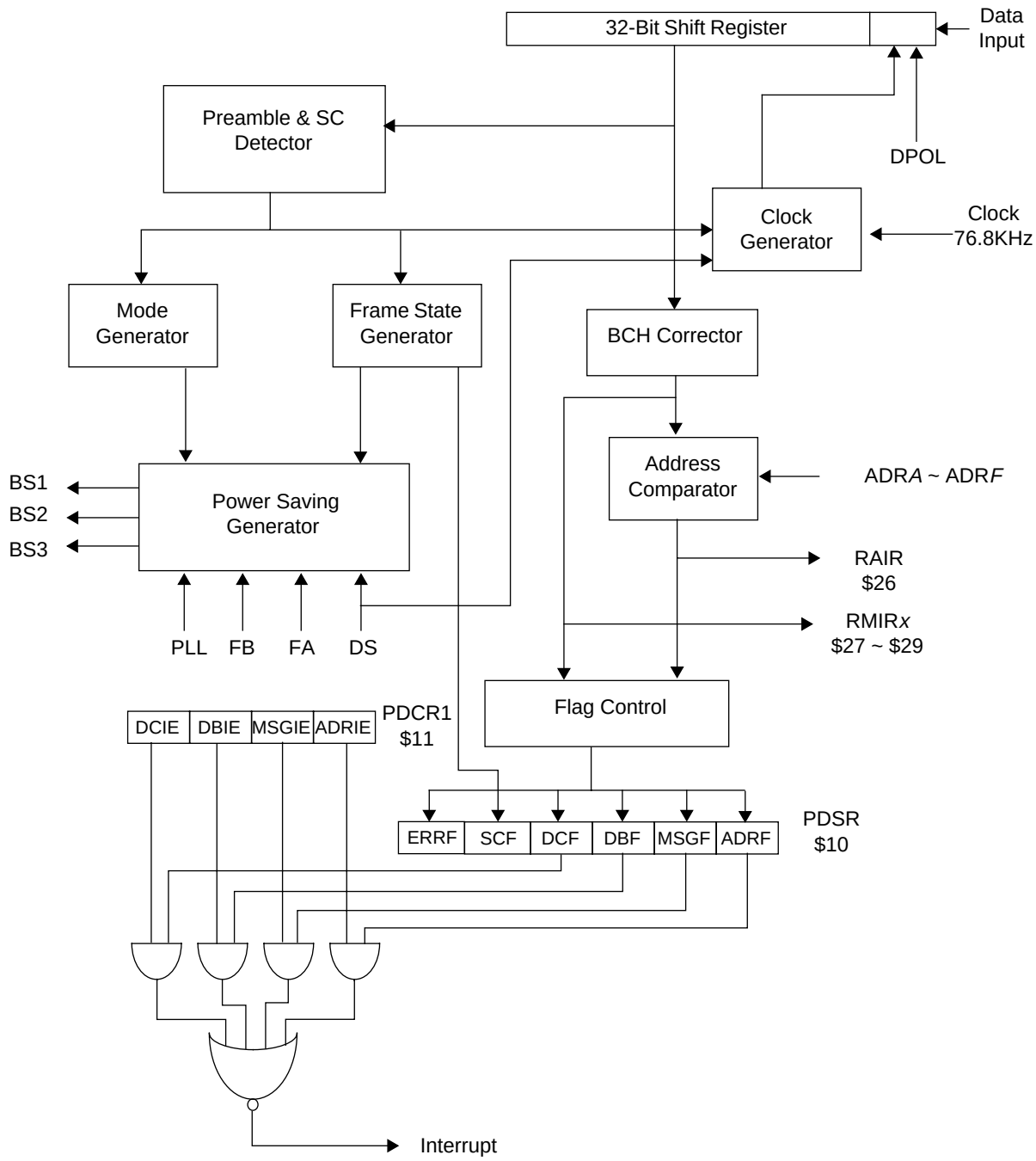


Figure 12-2. P-Decoder Block Diagram

12.3.1 Clock Generator

76.8KHz clock signal feeds this clock generator. It generates the internal system clock according to the DS1 and DS0 bits in the PDCR1 register.

12.3.2 BCH Corrector

Each codeword has 21 information bits, which correspond to the coefficients of a polynomial having the terms from x^{30} down to x^{10}. This polynomial is divided, modulo-2, by the generating polynomial $x^{10}+x^9+x^8+x^6+x^5+x^3+x^1$. The check bits correspond to the coefficients of the terms from x^9 to x^0 in the remainder polynomial found at the completion of this division. The complete block, consisting of the information bits followed by the check bits, corresponds to the coefficients of a polynomial which is integrally divisible in modulo-2 fashion by the generating polynomial. **Figure 12-3** shows the flowchart of the BCH error detection and correction. The BCH has the ability of correcting up to 2 bits error. The ERRF flag in the PDSR register shows the error status. If ERRF = '0', there are 2 or less than 2 bits error and the data in the RMIR register is valid. If ERRF = '1', there are more than 2 bits error and the data in the RMIR register is invalid.

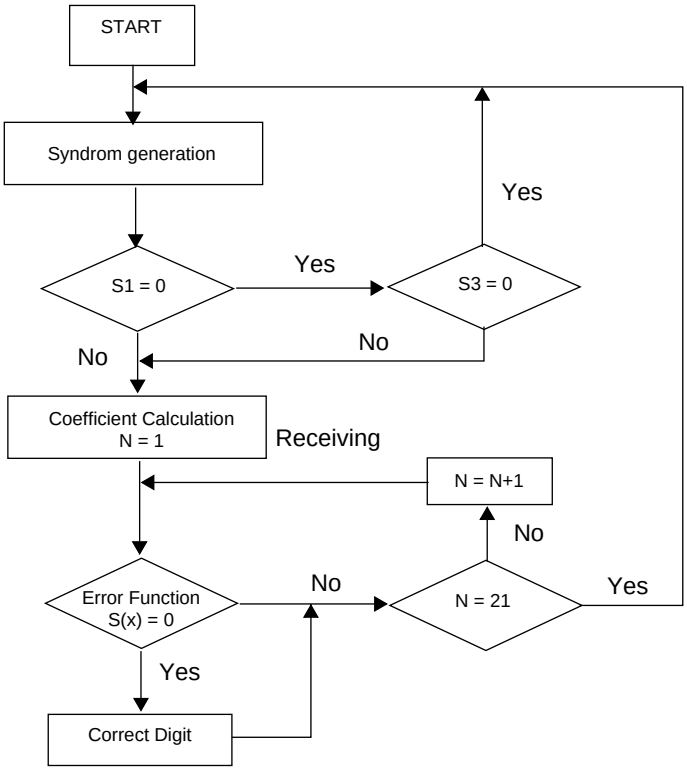


Figure 12-3. BCH Decoder Flow Chart

12.3.3 Address Comparator

The address comparator checks whether the incoming address codeword is addressed to itself according to the FA2-FA0, FB2-FB0 of the P-Decoder Control Register and the six user addresses. Both 32-bit codewords in the self-frame will be received and will be compared to the six user addresses stored in ADR*A*-ADR*F*. If the 1st 32 bits of received data is not the same as any of the six addresses, the 2nd 32 bits of received data in the same frame will also be compared. If the 1st or the 2nd codeword does match, the index of the matched user address out of the six user addresses and the 2 function bits in the received address codeword are stored in the RAIR register, subsequent multiple 20 bits of error-corrected message codewords are also stored in RMIR registers.

12.3.4 Battery Saving Generator

This block generates the battery saving signals that minimize the battery consumption. **Figure 12-5** shows the timing of the battery saving signals in waiting mode and **Figure 12-7** shows the timing in receiving mode.

12.3.4.1 BS1

It controls the main RF circuit ON/OFF. If BS1 is high, the power of RF circuit should be turned ON. If BS1 is low, the power of RF circuit should be turned OFF.

12.3.4.2 BS2

It is used to discharge the capacitor of the RF circuit. It changes to high simultaneously when BS1 changes to high and then changes to low after a certain time interval. This signal is active high.

12.3.4.3 BS3

It controls the ON/OFF of the PLL circuit. PLL circuit should be turned on before the RF circuit is turned on. The time which BS3 is activated before BS1 can be set by the PLL1 and PLL0 bit in PDCR1 register. BS3 goes low when BS1 goes low.

PLL1	PLL0	BS3 activate time before BS1
0	0	0 ms
0	1	9.76 ms
1	0	33.3 ms
1	1	62.5 ms

12.3.5 Mode Generator

There are four kinds of operating modes in the P-Decoder as shown in **Figure 12-4**: programming, waiting, preamble and receiving mode.

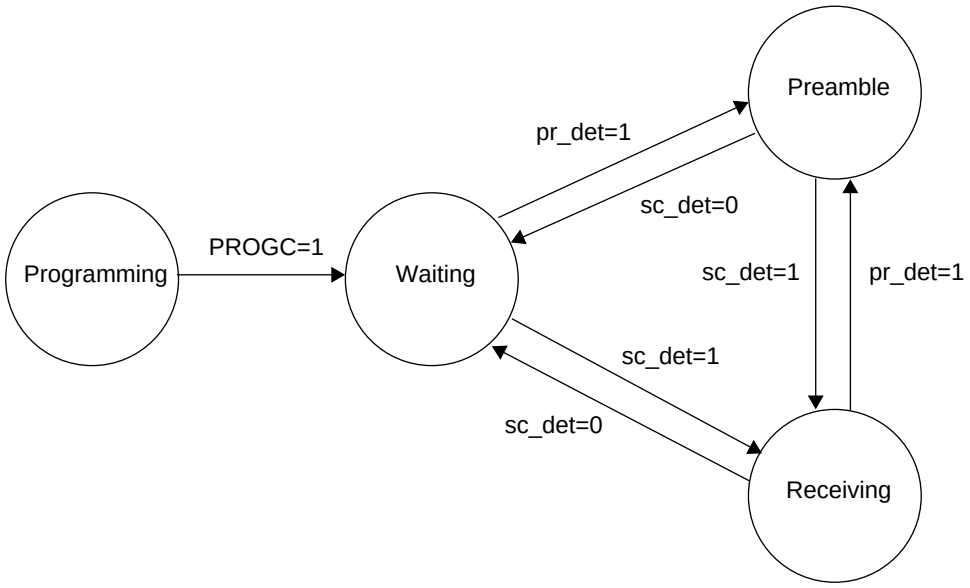


Figure 12-4. Mode Transition Diagram

12.3.6 Programming Mode

The P-Decoder is in programming mode when the PROGC bit in the PDCR1 is not set. PDCR1, PDCR2 and the six user address registers should be defined before the PROGC bit is set. After PROGC is set, the P-Decoder enter the waiting mode and the contents of PDCR1, PDCR2 and the six user address registers can no longer be modified until PROGC bit is cleared again.

12.3.7 Waiting Mode

The operating mode changes to waiting mode from preamble mode after the PROGC bit is set. In the waiting mode, the three battery saving signals toggle periodically. **Figure 12-5** shows the timing relationship. When BS1 = 1 and BS2 = 0, the P-Decoder synchronizes the incoming data from RF circuit to the rising edge of bit clock and checks 6 bits of the preamble pattern. If there is a preamble pattern, it will enter the preamble mode. Otherwise, it will stay in the waiting mode.

After the operating mode changes from the receiving mode to the waiting mode, the P-Decoder will synchronize to the position of the Synchronization Code (SC). Within t_{b1o} interval, it receives 32 bits of data to check whether it is the SC. If it is the SC, it will enter the receiving mode again. Otherwise, it will stay in the waiting mode.

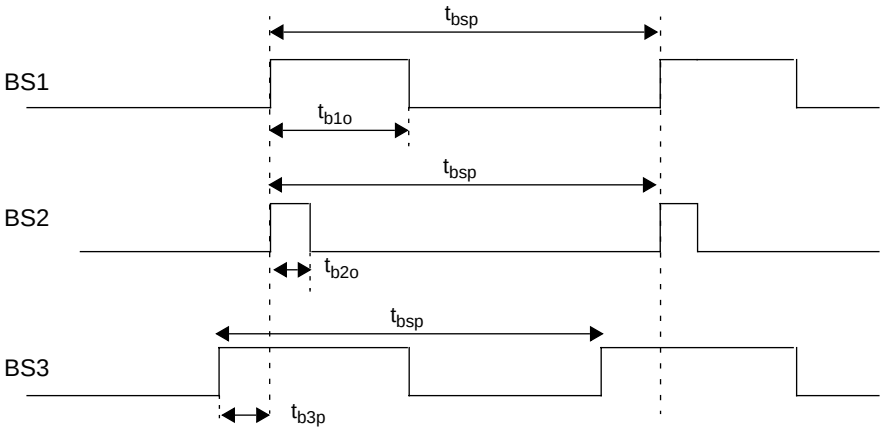


Figure 12-5. Battery Saving Signals In Waiting Mode

Table 12-1. Timing Of Battery Saving Signals

Characteristic		Typ (ms)
t_{bsp}	512 bps	1062.5
	1200 bps	453.3
	2400 bps	226.6
t_{b1o}	512 bps	80.08
	1200 bps	42.5
	2400 bps	26.6
t_{b2o}	512 bps	17.58
	1200 bps	15.83
	2400 bps	13.3
t_{b3p}	PLL=00	0
	PLL=01	9.76
	PLL=10	33.3
	PLL=11	62.5

NOTE

PLL = 11 is valid only in HC05PD6 device. Writing 11 to PLL should be avoided when using a HC705PD6 device.

12.3.7.1 Preamble Mode

After receiving the preamble pattern, it enters preamble mode and continues to check the following 544 bits of data. If there is an SC inside the 544-bit time frame, it will change to the receiving mode. Otherwise, it will change to the waiting mode. The BS1 and BS3 signals hold at '1' and the BS2 signal holds at '0' for the whole preamble mode duration.

12.3.7.2 Receiving Mode

The receiving mode starts with the frame 0 when the SC is detected in the preamble mode. The battery saving signals, BS1, BS2 and BS3, will be changed according to DS1-DS0, FA2-FA0, FB2-FB0, PLL1-PLL0 of the P-Decoder Control Registers. The 32 bits of data in the self-frame number will be received and compared to the six user addresses stored in ADR*A*-ADR*F*. If the 1st 32 bits of received data is not the same as one of the six addresses, the 2nd 32 bits of received data in the same frame will also be compared. If it does not match, the battery saving signals will hold at '0'. If the 1st or the 2nd codeword does match, the DBF flag in the PDSR register is set. After the address information and 2 function bits are stored in the RAIR register, ADRF flag in PDSR register will be set. The P-Decoder will keep BS1 = '1', BS2 = '0', BS3 = '1' to receive the message data. When 20 bits of message data are stored in the RMIR registers, the MSGF flag is set. The receiving of message data will be terminated when it meets other address codeword or the idle codeword and the DCF flag in PDSR register will be set. In the receiving mode, it continues to detect the SC. If it does not detect the SC, the SCF flag in PDSR will be cleared. If the SC is absent consecutively for 2 times, the operating mode will be changed to the waiting mode. The SCF flag in PDSR register will be set when the SC is detected and cleared when the SC is not detected. The ERRF flag in PDSR shows the validity of the 20-bit message data. In most situation, if ERRF = '0', there is 2 or less than 2 bits error and the data in the RMIR register is valid. If ERRF = '1', there is more than 2 bits error and the data in the RMIR register is invalid. Message codewords following this error codeword will be ignored. **Figure 12-6** and **Figure 12-7** show the timing of signals in the receiving mode.

12.3.8 Frame State Generator

In each batch, there are one SC and sixteen codewords. State 16 is assigned to SC, state 0 is assigned to the 1st codeword and state 15 is assigned to the 16th codeword sequentially. It provides the information to generate the battery saving signals.

12.3.9 Preamble and Synchronization Codeword Detector

It is used to detect the preamble pattern and synchronization codeword.

12.4 REGISTERS

There are several registers associated with the P-Decoder as described below.

12.4.1 P-Decoder Control Register 1 (PDCR1)

option map PDCR1 $0011		7	6	5	4	3	2	1	0
	R	PDEN	0	0	PROGC	DCIE	DBIE	MSGIE	ADRIE
	W								
reset ⇒		0	0	0	0	0	0	0	0

PDEN – P-Decoder Enable

If the PDEN bit is set, the P-Decoder module is enabled, If PDEN is cleared, the module is reset and disabled.

PROGC – Programming Completed

If the PROGC bit is set, the P-Decoder system is enabled. This bit must be CLEARed before the 6 user addresses and bits in PDCR2 and PDCR3 have been set. Once PROGC is set, user cannot change the data contents in PDCR2, PDCR3 and the 6 user address registers.

DCIE – Data Received Complete Interrupt Enable

When the DCIE is set, interrupt occurs when all the data is received completely provided the DCF in the status register is set and the I-bit in the Condition Code Register is cleared. If DCIE is cleared, this interrupt is disabled.

DBIE – Data Receiving Begin Interrupt Enable

When the DBIE is set, interrupt occurs when it begins to receive data into the address and message registers provided the DBF in the status register is set and the I-bit in the Condition Code Register is cleared. If DBIE is cleared, this interrupt is disabled.

MSGIE – Message Received Interrupt Enable

When the MSGIE is set, interrupt occurs when 20 bits of message has been stored in the Received Message Information Registers provided the MSGF in the status register is set and the I-bit in the Condition Code Register is cleared. If MSGIE is cleared, this interrupt is disabled.

ADRIE – Address Received Interrupt Enable

When the ADRIE is set, interrupt occurs when 3 bits of address and 2 function bits has been stored in the Received Address Information Register provided the ADRF in the status register is set and the I-bit in the Condition Code Register is cleared. If ADRIE is cleared, this interrupt is disabled.

12.4.2 P-Decoder Control Register 2 (PDCR2)

option map PDCR2 $0012		7	6	5	4	3	2	1	0
	R	0	0	0	0	0	FB2	FB1	FB0
	W								
reset ⇒		0	0	0	0	0	0	0	0

FB specify the frame number that the 6 user defined addresses belong to.

Table 12-2. Frame Number Definition

FA / FB	Frame number
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

12.4.3 P-decoder Control Register 3 (PDCR3)

option map PDCR3 $0013		7	6	5	4	3	2	1	0
	R	PLL1	PLL0	DS1	DS0	DPOL	FA2	FA1	FA0
	W								
reset ⇒		0	0	0	0	0	0	0	0

PLL1, PLL0

These 2 bits decide the time that BS3 goes high before BS1. When the data speed is configured to 512 bps or 1200 bps, BS3 timing looks like the table shown in **Table 12-1**. When the data speed is configured to 2400 bps, BS3 timing is shown in the following table.

PLL1	PLL0	PLL Time
0	0	0.42 ms
0	1	10.17 ms
1	0	33.81 ms
1	1	62.96 ms

NOTE

PLL1,0 = 11 is valid only in HC05PD6. Writing 11 to PLL should be avoided when using a HC705PD6 device.

DS1, DS0

These two bits set the data speed of the incoming data.

DS1	DS0	Data Speed
0	0	512 bps
0	1	1200 bps
1	0	2400 bps
1	1	512 bps

DPOL

This bit defines the polarity of the incoming data.

0 = positive polarity
1 = negative polarity

FA0, FA1, FA2

These specify the frame number that the 6 user defined addresses belong to. Refer to **Table 12-2** for detailed description.

12.4.4 P-Decoder Status Register (PDSR)

option map PDSR $0010

	7	6	5	4	3	2	1	0
R	0	0	ERRF	SCF	DCF	DBF	MSGF	ADRF
W								
reset ⇒	0	0	0	0	0	0	0	0

ERRF – Error Flag

In most situation, when ERRF= 0, it indicates that there is 2 or less than 2 bits error in the received message codeword and correction has been done. But in some cases, ERRF= 0 even there is more than 2 bits error.

When ERRF= 1, it indicates that there is more than 2 bits error in the received message codeword and the data in the Received Message Information Registers are invalid. The P-Decoder will then stop receiving the codewords following this error codeword.

SCF – Synchronization Code Detection Flag

When the P-Decoder cannot detect the synchronization code, SCF is cleared. Otherwise, SCF is set.

DCF – Data Received Complete Flag

This flag is set when all the data is completely received. After serving this interrupt, it is the user's responsibility to clear this bit, otherwise the CPU will keep on serving this interrupt.

DBF – Data Receiving Begin Flag

This flag is set when the P-Decoder begins to receive data into the address and message registers. After serving this interrupt, it is the user's responsibility to clear this bit, otherwise the CPU will keep on serving this interrupt.

MSGF – Message Received Flag

This flag is set when 20 bits of message has been stored in the Received Message Information Registers. After serving this interrupt, it is the user's responsibility to clear this bit, otherwise the CPU will keep on serving this interrupt.

ADRF

This flag is set when 3 bits of address and 2 function bits have been stored in the Received Address Information Register. After serving this interrupt, it is the user's responsibility to clear this bit, otherwise the CPU will keep on serving this interrupt.

12.4.5 User Address Registers

There are six user addresses and two self-frames for receiving public services. Each of these six addresses has 18 bits and 1 frame selection bit to select whether FA or FB in PDCR1 are used. The six user addresses are A0-A17, B0-B17, C0-C17, D0-D17, E0-E17 and F0-F17 respectively from $14-$25.

option map ADRA0 $0014

	7	6	5	4	3	2	1	0
R/W	A7	A6	A5	A4	A3	A2	A1	A0
reset ⇒	0	0	0	0	0	0	0	0

option map ADRA1 $0015

	7	6	5	4	3	2	1	0
R/W	A15	A14	A13	A12	A11	A10	A9	A8
reset ⇒	0	0	0	0	0	0	0	0

option map ADRA2 $0016

	7	6	5	4	3	2	1	0
R	AFNS	0	0	0	0	0	A17	A16
W								
reset ⇒	0	0	0	0	0	0	0	0

A17-A0 is the 18-bit user defined address.

AFNS

0 = The value in FA is assigned to address A. User address, A17-A0, is compared to the frame number specified in FA of the incoming data.

1 = The values in FB is assigned to address A. User address, A17-A0, is compared to the frame numbers specified in FB of the incoming data.

NOTE

If less than 6 user addresses are used, say only 3 user addresses are used, the 3 unused addresses must be written with the same values as any of the addresses used. For example, if address A = 000008, B = 801AA3, D = 016913, then address C, E, F should be written to 000008, 016913 and 016913 respectively or both written to 000008 or etc...

12.4.6 Received Address Information Register (RAIR)

option map RAIR $0026

	7	6	5	4	3	2	1	0
R	0	0	F1	F0	0	RA2	RA1	RA0
W								
reset ⇒	0	0	0	0	0	0	0	0

F1, F0

These two bits store the data of the function bits.

F1	F0	Function
0	0	A
0	1	B
1	0	C
1	1	D

RA2-RA0

These three bits show which one of the six user address matches the incoming address.

12.4.7 Received Message Information Registers (RMIR*x*)

option map RMIR0 $0027

	7	6	5	4	3	2	1	0
R	RMI7	RMI6	RMI5	RMI4	RMI3	RMI2	RMI1	RMI0
W								
reset ⇒	0	0	0	0	0	0	0	0

option map RMIR1 $0028

	7	6	5	4	3	2	1	0
R	RMI15	RMI14	RMI13	RMI12	RMI11	RMI10	RMI9	RMI8
W								
reset ⇒	0	0	0	0	0	0	0	0

option map RMIR2 $0029

	7	6	5	4	3	2	1	0
R	0	0	0	0	RMI19	RMI18	RMI17	RMI16
W								
reset ⇒	0	0	0	0	0	0	0	0

These three registers store the 20 bits of message received.

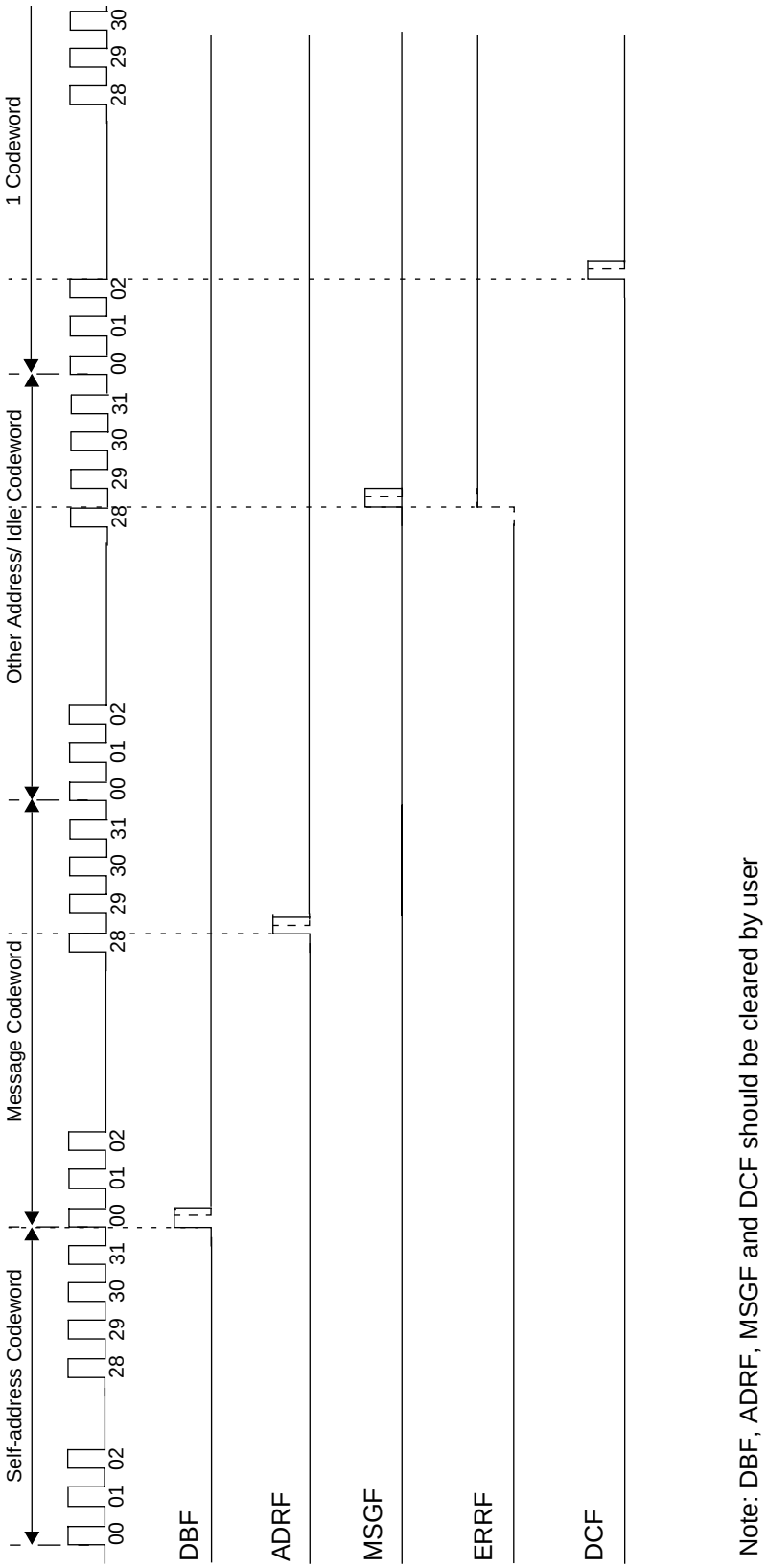


Figure 12-6. P-Decoder Flags Timing

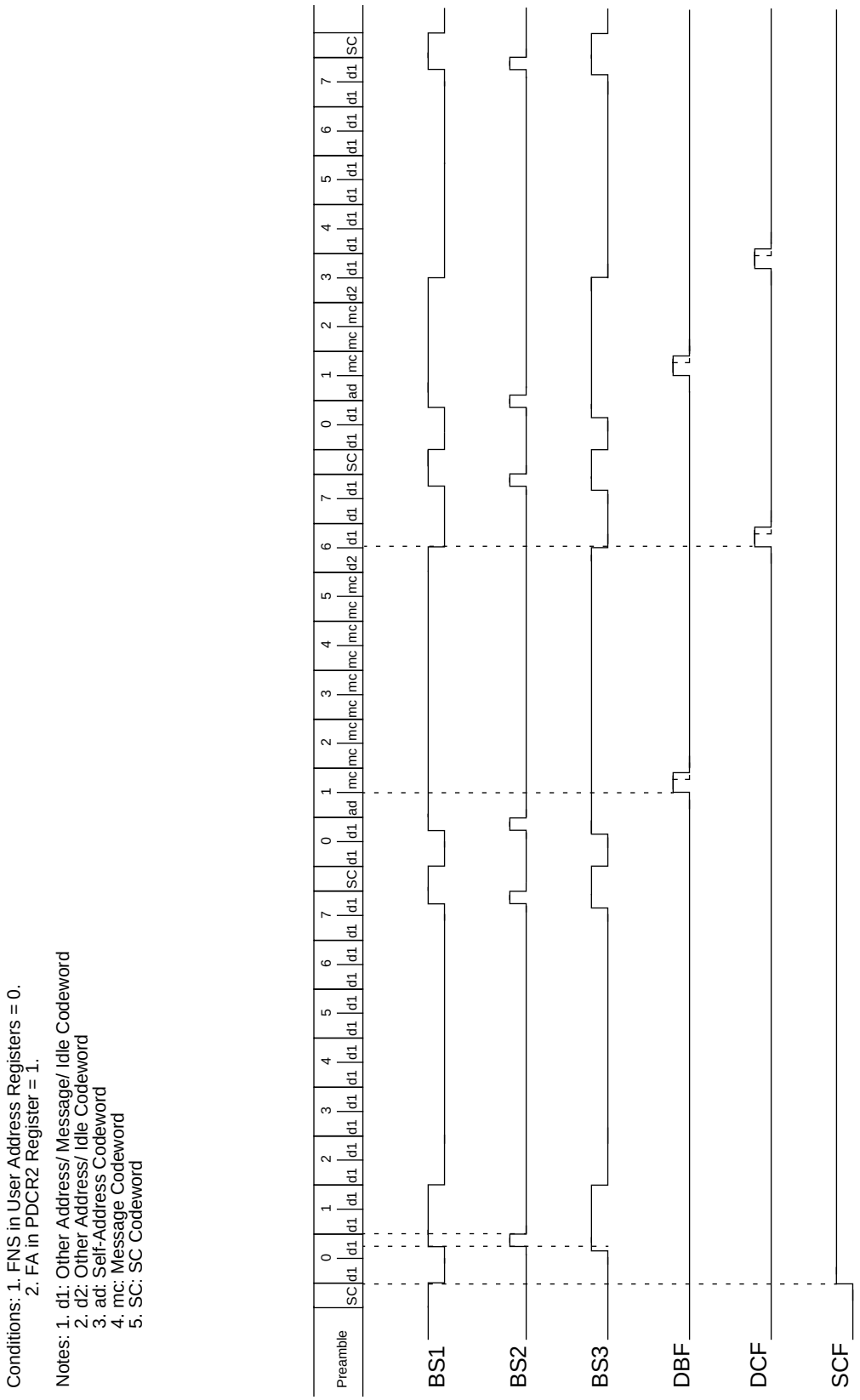


Figure 12-7. Receiving Mode Timing

SECTION 13
INSTRUCTION SET

This section describes the addressing modes and instruction types.

13.1 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

13.1.1 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.

13.1.2 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.

13.1.3 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.

13.1.4 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.

13.1.5 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.

13.1.6 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.

13.1.7 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.

13.1.8 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.

13.1.9 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

13.1.10 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 13-1** lists the register/memory instructions.

Table 13-1. 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

13.1.11 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 13-2** lists the read-modify-write instructions.

Table 13-2. 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

13.1.12 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 13-3** lists the jump and branch instructions.

Table 13-3. 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

13.1.13 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 13-4** lists these instructions.

Table 13-4. Bit Manipulation Instructions

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

13.1.14 Control Instructions

These register reference instructions control CPU operation during program execution. Control instructions, listed in **Table 13-5**, use inherent addressing.

Table 13-5. 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

13.1.15 Instruction Set Summary

Table 13-6 is an alphabetical list of all M68HC05 instructions and shows the effect of each instruction on the condition code register.

Table 13-6. Instruction Set Summary

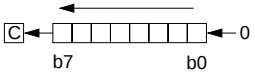
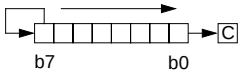
Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			H	I	N	Z	C				
ADC #*opr* ADC *opr* ADC *opr* ADC *opr*,X ADC *opr*,X ADC ,X	Add with Carry	A ← (A) + (M) + (C)	◊	—	◊	◊	◊	IMM DIR EXT IX2 IX1 IX	A9 B9 C9 D9 E9 F9	ii dd hh ll ee ff ff	2 3 4 5 4 3
ADD #*opr* ADD *opr*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

Table 13-6. 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 $\overline{IRQ}$ Pin High	PC ← (PC) + 2 + *rel* ? $\overline{IRQ}$ = 1	—	—	—	—	—	REL	2F	rr	3
BIL *rel*	Branch if $\overline{IRQ}$ Pin Low	PC ← (PC) + 2 + *rel* ? $\overline{IRQ}$ = 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 dd rr dd rr dd rr dd rr dd rr dd rr dd rr	5 5 5 5 5 5 5 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 dd rr dd rr dd rr dd rr dd rr dd rr dd rr	5 5 5 5 5 5 5 5
BRN *rel*	Branch Never	PC ← (PC) + 2 + *rel* ? 1 = 0	—	—	—	—	—	REL	21	rr	3

Table 13-6. 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)	—	—	◊	◊	1	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 13-6. Instruction Set Summary (Continued)

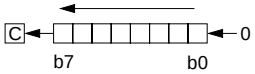
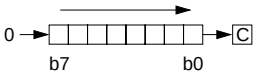
Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			H	I	N	Z	C				
INC *opr* INCA INCX INC *opr*,X INC ,X	Increment Byte	M ← (M) + 1 A ← (A) + 1 X ← (X) + 1 M ← (M) + 1 M ← (M) + 1	—	—	◊	◊	—	DIR INH INH IX1 IX	3C 4C 5C 6C 7C	dd ff 	5 3 3 6 5
JMP *opr* JMP *opr* JMP *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 13-6. Instruction Set Summary (Continued)

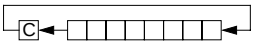
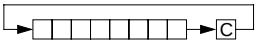
Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			H	I	N	Z	C				
ORA #*opr* ORA *opr* ORA *opr* ORA *opr*,X ORA *opr*,X ORA ,X	Logical OR Accumulator with Memory	A ← (A) ∨ (M)	—	—	◊	◊	—	IMM DIR EXT IX2 IX1 IX	AA BA CA DA EA FA	ii dd hh ll ee ff ff	2 3 4 5 4 3
ROL *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		6
RTS	Return from Subroutine	SP ← (SP) + 1; Pull (PCH) SP ← (SP) + 1; Pull (PCL)						INH			
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 13-6. 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		—	◊	—	—	—	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 13-7. 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 / LSB	0
0	BRSET0 5 3 DIR

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

SECTION 14
ELECTRICAL SPECIFICATIONS

14.1 MAXIMUM RATINGS

Table 14-1. Maximum Ratings

(Voltages referenced to V_{SS})

Rating	Symbol	Value	Unit
Supply Voltage	V_{DD}	–0.3 to +7.0	V
LCD Supply Voltage	VLCD3	–0.3 to +7.0	V
Input Voltage	V_{IN}	V_{SS}–0.3 to V_{DD} +0.3	V
$\overline{IRQ1}$ and $\overline{IRQ2}$ Pin	V_{IN}	V_{SS}–0.3 to 2V_{DD} +0.3	V
Current Drain Per Pin Excluding PB1, PB2, V_{DD} and V_{SS}	I	–25	mA
Operating Temperature Range (Standard) (Extended)	T_A	T_L to T_H 0 to +70 –40 to +85	°C
Storage Temperature Range	T_{STG}	–65 to +150	°C

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum-rated voltages to this high-impedance circuit. For proper operation, it is recommended that V_{IN} and V_{OUT} be constrained to the range $V_{SS} \le (V_{IN}$ or $V_{OUT}) \le V_{DD}$. Reliability of operation is enhanced if unused inputs are connected to an appropriate logic voltage level (e.g., either V_{SS} or V_{DD}).

14.2 THERMAL CHARACTERISTICS

Table 14-2. Thermal Characteristics

Characteristic	Symbol	Value	Unit
Thermal Resistance 80-Pin TQFP 80-Pin QFP	 θ_{JA} θ_{JA}	 100 120	 °C/W °C/W

14.3 DC ELECTRICAL CHARACTERISTICS

Table 14-3. DC Electrical Characteristics (5V)

(V_{DD} = 5V ± 10%, V_{SS} = 0 Vdc, T_A = 0°C to +70°C, unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Output Voltage I_{Load} = 10.0 µA	V_{OL} V_{OH}	— V_{DD}–0.1	— —	0.1 —	V
Output High Voltage (V_{DD}=5V) (I_{Load} =–0.4 mA) All I/O ports	V_{OH}	V_{DD}–0.8	—	—	V
Output Low Voltage (V_{DD}=5V) (I_{Load} =0.8 mA) Ports A, B, C	V_{OL}	—	—	0.4	V
Output Low Voltage (V_{DD}=5V) (I_{Load} = 5.0 mA) Ports D, E, F, G, H	V_{OL}	—	—	0.4	V
Input High Voltage All I/O ports, $\overline{\text{RESET}}$, OSC1, XOSC1	V_{IH}	0.7×V_{DD}	—	V_{DD}	V
Input Low Voltage All I/O ports, $\overline{\text{RESET}}$, OSC1, XOSC1	V_{IL}	V_{SS}	—	0.2×V_{DD}	V
Supply Current (see Notes) Run (f_{OP}=2MHz) Wait (f_{OP}=2MHz) Stop XOSC=76.8kHz, V_{DD}=5V, T_A=+25°C	I_{DD}		 5.5 1.8 250		 mA mA µA
I/O Ports Hi-Z Leakage Current All I/O ports	I_Z	—	—	±10	µA
Input Pulldown Current Ports A, B, C (with pulldown activated)	I_{IL}	50	70	90	µA
Input Current OSC1, XOSC1	I_{in}	—	—	±1	µA
Capacitance Ports (as Input or Output) $\overline{\text{RESET}}$, OSC1, OSC2, XOSC1, XOSC2	 C_{out} C_{in}	 — —	 — —	 12 8	 pF pF
Crystal Oscillator Mode Feedback Resistor OSC1 to OSC2 XOSC1 to XOSC2	 R_{OF} R_{XOF}	 1.5 5.0	 2.0 6.0	 3.0 7.0	 MΩ MΩ
Crystal Oscillator Mode Damping Resistor XOSC1 to XOSC2	R_{XOD}	1.5	3	3.5	MΩ
Reset Pull-up Resistor	R_{RST}	8	15	25	kΩ

Table 14-4. DC Electrical Characteristics (3.6V)

(V_{DD} = 3.6V ± 10%, V_{SS} = 0 Vdc, T_A = 0°C to +70°C, unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Output Voltage I_{Load} = 10.0 µA	V_{OL} V_{OH}	— V_{DD}–0.1	— —	0.1 —	V
Output High Voltage (V_{DD}=5V) (I_{Load} =–0.4 mA) All I/O ports	V_{OH}	V_{DD}–0.8	—	—	V
Output Low Voltage (V_{DD}=5V) (I_{Load} =0.8 mA) Ports A, B, C	V_{OL}	—	—	0.4	V
Output Low Voltage (V_{DD}=5V) (I_{Load} = 5.0 mA) Ports D, E, F, G, H	V_{OL}	—	—	0.4	V
Input High Voltage All I/O ports, $\overline{RESET}$, OSC1, XOSC1	V_{IH}	0.7×V_{DD}	—	V_{DD}	V
Input Low Voltage All I/O ports, $\overline{RESET}$, OSC1, XOSC1	V_{IL}	V_{SS}	—	0.2×V_{DD}	V
Supply Current (see Notes) Run (f_{OP}=1MHz) Wait (f_{OP}=1MHz) Stop XOSC=76.8kHz, V_{DD}=5V, T_A=+25°C	I_{DD}		 2.5 600 250		 mA µA µA
I/O Ports Hi-Z Leakage Current All I/O ports	I_Z	—	—	±10	µA
Input Pulldown Current Ports A, B, C (with pulldown activated)	I_{IL}	50	70	90	µA
Input Current OSC1, XOSC1	I_{in}	—	—	±1	µA
Capacitance Ports (as Input or Output) $\overline{RESET}$, OSC1, OSC2, XOSC1, XOSC2	 C_{out} C_{in}	 — —	 — —	 12 8	 pF pF
Crystal Oscillator Mode Feedback Resistor OSC1 to OSC2 XOSC1 to XOSC2	 R_{OF} R_{XOF}	 1.5 5.0	 2.0 6.0	 3.0 7.0	 MΩ MΩ
Crystal Oscillator Mode Damping Resistor XOSC1 to XOSC2	R_{XOD}	1.5	3	3.5	MΩ
Reset Pull-up Resistor	R_{RST}	8	15	25	kΩ

NOTES:
1. All values shown reflect average measurements.
2. Typical values at midpoint of voltage range, 25°C only.
3. Run (Operating) I_{DD}, Wait I_{DD}: Measured using external square wave clock source to OSC1 (f_{OSC} = 4.2 MHz/2.1MHz), all inputs 0.2 VDC from rail; no DC loads, less than 50pF on all outputs, C_L = 20 pF on OSC2.
4. Wait I_{DD}: All ports configured as inputs, V_{IL} = 0.2 VDC, V_{IH} = V_{DD}–0.2 VDC.
5. Stop I_{DD} measured with OSC1 = V_{SS}.
6. Wait I_{DD} is affected linearly by the OSC2 capacitance.

14.4 CONTROL TIMING

Table 14-5. Control Timing (5V)

(V_{DD} = 5V ±10%, V_{SS} = 0 Vdc, T_A = 0°C to +70°C, unless otherwise noted)

Characteristic	Symbol	Min	Max	Units
Frequency of Operation Crystal Oscillator Option External Clock Source	 f_{OSC} f_{OSC}	 — DC	 4.2 4.2	 MHz MHz
Internal Operating Frequency Crystal Oscillator ($f_{OSC} \div 2$) External Clock ($f_{OSC} \div 2$)	 f_{OP} f_{OP}	 — DC	 2.1 2.1	 MHz MHz
Cycle Time ($1/f_{OP}$)	t_{CYC}	480	—	ns
Crystal Oscillator Start-Up Time (Crystal Oscillator Option)	t_{OXON}	—	100	ms
OSC Stop Recovery Time (Crystal Oscillator Option)	t_{OXSR}	—	100	ms
$\overline{\text{RESET}}$ Pulse Width Low	t_{RL}	1.5	—	t_{CYC}
$\overline{\text{IRQ}}$ Interrupt Pulse Width Low (Edge-Triggered)	t_{ILIH}	125	—	ns
$\overline{\text{IRQ}}$ Interrupt Pulse Period	t_{ILIL}	note 1	—	t_{CYC}
OSC1 Pulse Width	t_{OH}, t_{OL}	100	—	ns

Table 14-6. Control Timing (3.6V)

(V_{DD} = 3.6V ±10%, V_{SS} = 0 Vdc, T_A = 0°C to +70°C, unless otherwise noted)

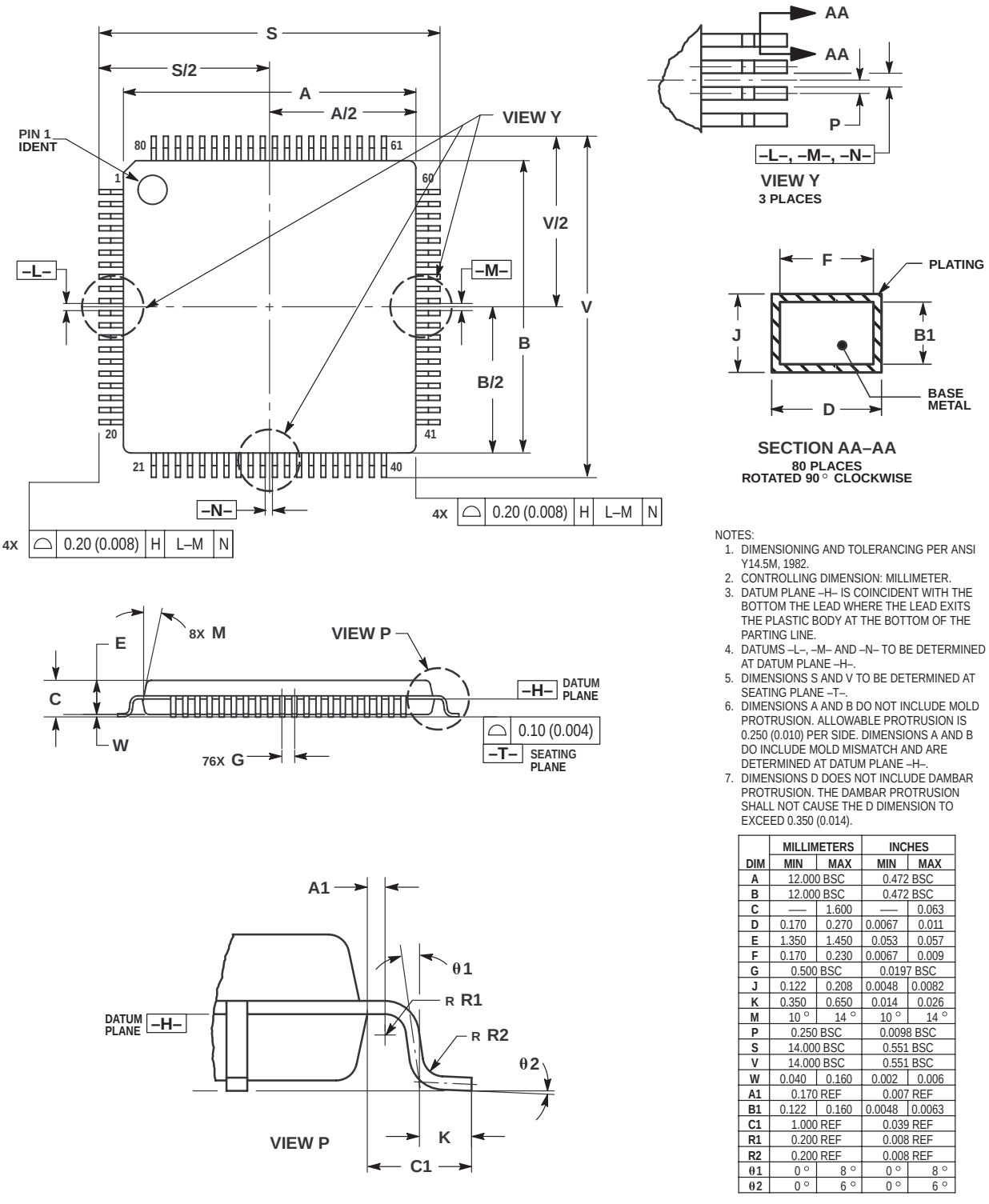
Characteristic	Symbol	Min	Max	Units
Frequency of Operation Crystal Oscillator Option External Clock Source	 f_{OSC} f_{OSC}	 — DC	 2.0 2.0	 MHz MHz
Internal Operating Frequency Crystal Oscillator (f_{OSC} ÷ 2) External Clock (f_{OSC} ÷ 2)	 f_{OP} f_{OP}	 — DC	 1.0 1.0	 MHz MHz
Cycle Time ($1/f_{OP}$)	t_{CYC}	960	—	ns
Crystal Oscillator Start-Up Time (Crystal Oscillator Option)	t_{OXON}	—	200	ms
OSC Stop Recovery Time (Crystal Oscillator Option)	t_{OXSR}	—	200	ms
$\overline{\text{RESET}}$ Pulse Width Low	t_{RL}	1.5	—	t_{CYC}
$\overline{\text{IRQ}}$ Interrupt Pulse Width Low (Edge-Triggered)	t_{ILIH}	250	—	ns
$\overline{\text{IRQ}}$ Interrupt Pulse Period	t_{ILIL}	note 1	—	t_{CYC}
OSC1 Pulse Width	t_{OH}, t_{OL}	200	—	ns

NOTES:
1. The minimum period t_{ILIL} or t_{IHIH} should not be less than the number of cycles it takes to execute the interrupt service routine plus 19 t_{CYC}.

SECTION 15
MECHANICAL SPECIFICATIONS

This section provides the mechanical dimensions for the 80-Pin TQFP and 80-Pin QFP.

15.1 80-PIN THIN-QUAD-FLAT-PACKAGE (Case 917-01)

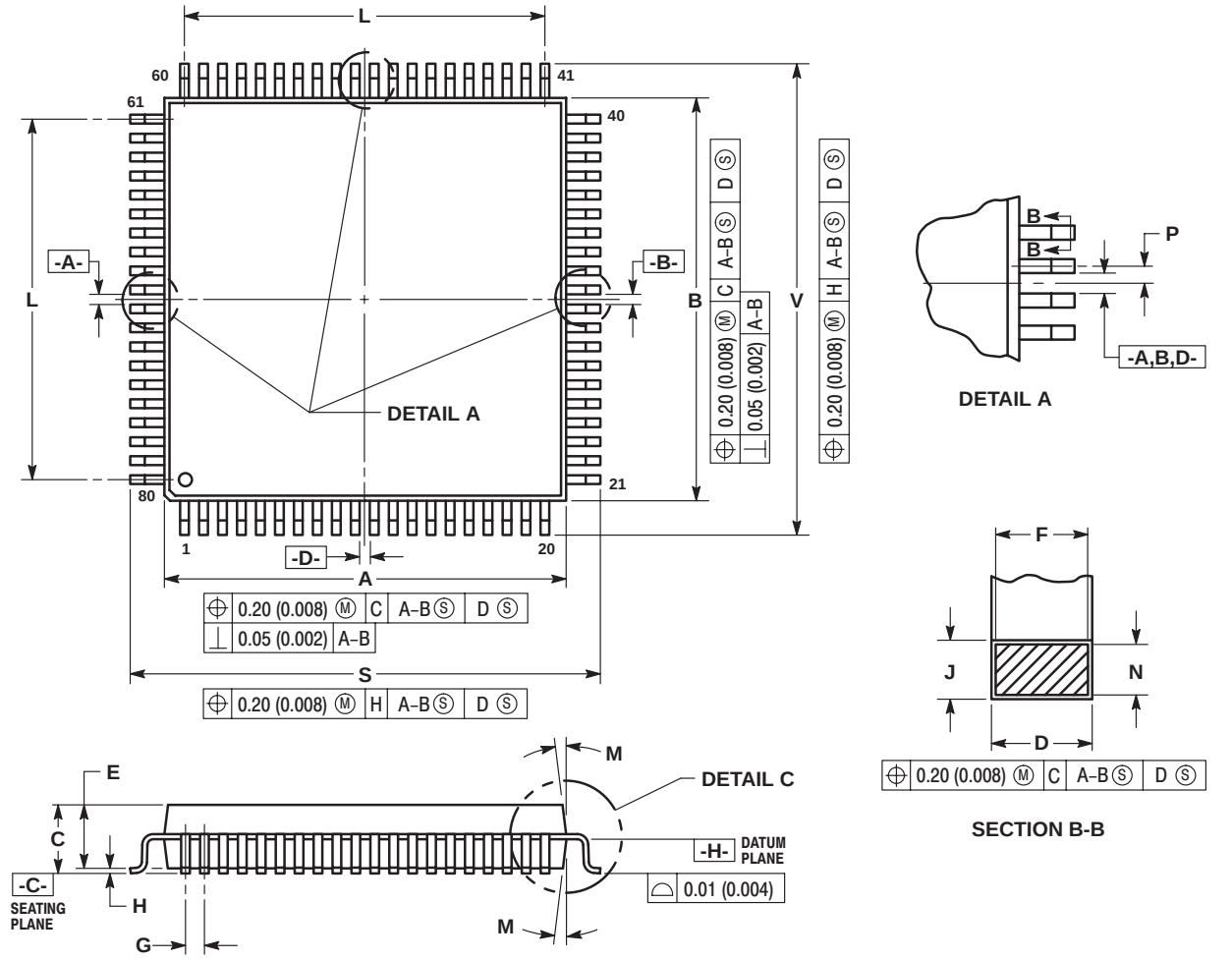


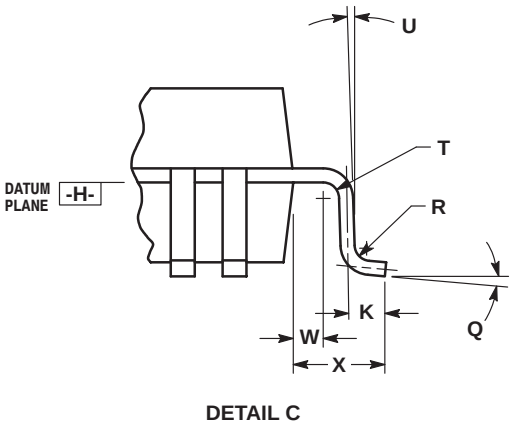
NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: MILLIMETER.
3. DATUM PLANE –H– IS COINCIDENT WITH THE BOTTOM THE LEAD WHERE THE LEAD EXITS THE PLASTIC BODY AT THE BOTTOM OF THE PARTING LINE.
4. DATUMS –L–, –M– AND –N– TO BE DETERMINED AT DATUM PLANE –H–.
5. DIMENSIONS S AND V TO BE DETERMINED AT SEATING PLANE –T–.
6. DIMENSIONS A AND B DO NOT INCLUDE MOLD PROTRUSION. ALLOWABLE PROTRUSION IS 0.250 (0.010) PER SIDE. DIMENSIONS A AND B DO INCLUDE MOLD MISMATCH AND ARE DETERMINED AT DATUM PLANE –H–.
7. DIMENSIONS D DOES NOT INCLUDE DAMBAR PROTRUSION. THE DAMBAR PROTRUSION SHALL NOT CAUSE THE D DIMENSION TO EXCEED 0.350 (0.014).

	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	12.000 BSC		0.472 BSC	
B	12.000 BSC		0.472 BSC	
C	—	1.600	—	0.063
D	0.170	0.270	0.0067	0.011
E	1.350	1.450	0.053	0.057
F	0.170	0.230	0.0067	0.009
G	0.500 BSC		0.0197 BSC	
J	0.122	0.208	0.0048	0.0082
K	0.350	0.650	0.014	0.026
M	10°	14°	10°	14°
P	0.250 BSC		0.0098 BSC	
S	14.000 BSC		0.551 BSC	
V	14.000 BSC		0.551 BSC	
W	0.040	0.160	0.002	0.006
A1	0.170 REF		0.007 REF	
B1	0.122	0.160	0.0048	0.0063
C1	1.000 REF		0.039 REF	
R1	0.200 REF		0.008 REF	
R2	0.200 REF		0.008 REF	
θ1	0°	8°	0°	8°
θ2	0°	6°	0°	6°

Figure 15-1. 80-Pin TQFP Mechanical Dimensions

15.2 80-PIN QUAD-FLAT-PACKAGE (Case 841B-01)





NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: MILLIMETER.
3. DATUM PLANE -H- IS LOCATED AT BOTTOM OF LEAD AND IS COINCIDENT WITH THE LEAD WHERE THE LEAD EXITS THE PLASTIC BODY AT THE BOTTOM OF THE PARTING LINE.
4. DATUMS -A-, -B- AND -D- TO BE DETERMINED AT DATUM PLANE -H-.
5. DIMENSIONS S AND V TO BE DETERMINED AT SEATING PLANE -C-.
6. DIMENSIONS A AND B DO NOT INCLUDE MOLD PROTRUSION. ALLOWABLE PROTRUSION IS 0.25 (0.010) PER SIDE. DIMENSIONS A AND B DO INCLUDE MOLD MISMATCH AND ARE DETERMINED AT DATUM PLANE -H-.
7. DIMENSION D DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL BE 0.08 (0.003) TOTAL IN EXCESS OF THE D DIMENSION AT MAXIMUM MATERIAL CONDITION. DAMBAR CANNOT BE LOCATED ON THE LOWER RADIUS OR THE FOOT.

	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	13.90	14.10	0.547	0.555
B	13.90	14.10	0.547	0.555
C	2.15	2.45	0.084	0.096
D	0.22	0.38	0.009	0.015
E	2.00	2.40	0.079	0.094
F	0.22	0.33	0.009	0.013
G	0.65 BSC		0.026 BSC	
H	—	0.25	—	0.010
J	0.13	0.23	0.005	0.009
K	0.65	0.95	0.026	0.037
L	12.35 BSC		0.486 BSC	
M	5°	10°	5°	10°
N	0.13	0.17	0.005	0.007
P	0.325 BSC		0.013 BSC	
Q	0°	7°	0°	7°
R	0.13	0.30	0.005	0.012
S	16.95	17.45	0.667	0.687
T	0.13	—	0.005	—
U	0°	—	0°	—
V	16.95	17.45	0.667	0.687
W	0.35	0.45	0.014	0.018
X	1.6 REF		0.06 REF	

Figure 15-2. 80-Pin QFP Mechanical Dimensions

APPENDIX A
MC68HC705PD6

This appendix describes the differences between the MC68HC705PD6 and MC68HC05PD6.

A.1 INTRODUCTION

The MC68HC705PD6 is an EPROM version of the MC68HC05PD6, and is available for user system evaluation and debugging. The MC68HC705PD6 is functionally identical to the MC68HC05PD6 with the exception of the 16400 bytes user ROM is replaced by 16400 bytes user EPROM. Also, the mask options available on the MC68HC05PD6 are implemented using the Mask Option Register (MOSR) in the MC68HC705PD6.

The MC68HC705PD6 is available in 80-pin Quad-Flat-Package (QFP).

A.2 MEMORY

The MC68HC705PD6 memory map is shown in **Figure A-1**.

A.3 MASK OPTION REGISTER (MOSR), $000F

The Mask Option Register (MOSR) is a byte of EPROM used to select the features controlled by mask options on the MC68HC05PD6.

		7	6	5	4	3	2	1	0
MOSR $000F	R	RSTR	OSCR	XOSCR	0	0	0	0	0
	W								
Erased ⇒		U	U	U	0	0	0	0	0
Reset ⇒		Unaffected By Reset							

RSTR – $\overline{\text{RESET}}$ pin pull-up resistor

1 = Internal pull-up resistor connected.
0 = Internal pull-up resistor not connected.

OSCR – OSC feedback resistor

1 = Internal feedback resistor connected between OSC1 and OSC2.
0 = Internal feedback resistor not connected.

XOSCR – XOSC feedback resistor

1 = Internal feedback resistor connected between XOSC1 and XOSC2.
0 = Internal feedback resistor not connected.

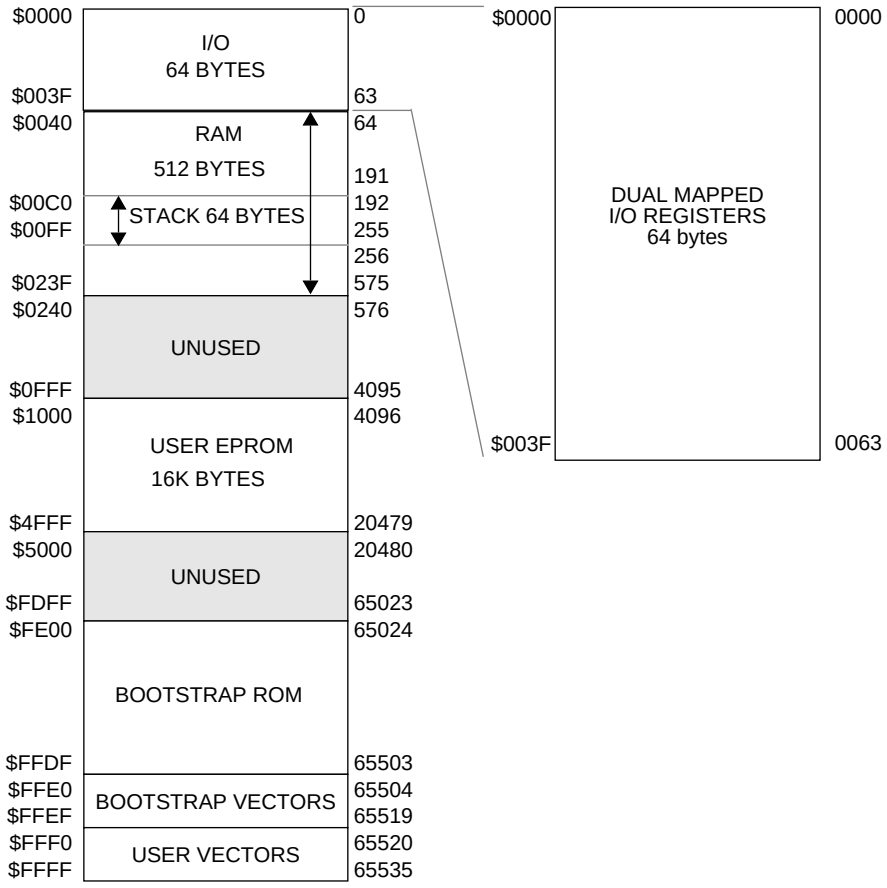


Figure A-1. MC68HC705PD6 Memory Map

A.4 BOOTLOADER MODE

Bootloader mode is entered upon the rising edge of $\overline{\text{RESET}}$ if the V_{PP} pin is at V_{TST}, PC6 at logic one, and PC7 at logic zero. The Bootloader program is masked in the ROM area from $FE00 to $FFDF. 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.

The user code must be a one-to-one correspondence with the internal EPROM addresses.

Table A-1. Operating Mode Initialization

MODE	$\overline{\text{RESET}}$	PC7/$\overline{\text{IRQ1}}$	PC6/$\overline{\text{IRQ2}}$	VPP
Single-Chip (Normal)		V_{SS} to V_{DD}	V_{SS} to V_{DD}	V_{SS} to V_{DD}
Bootstrap		V_{SS}	V_{DD}	V_{TST}

$V_{TST}=2 \times V_{DD}$

A.5 EPROM PROGRAMMING

Programming the on-chip EPROM is achieved by using the Program Control Register located at address $3D.

Please contact Motorola for programming board availability.

A.5.1 EPROM Program Control Register (PCR)

This register is provided for programming the on-chip EPROM in the MC68HC705PD6.

		bit-7	bit-6	bit-5	bit4	bit-3	bit-2	bit1	bit-0
PCR $003D	**Read**	RESERVED						ELAT	PGM
	Write								
	Reset	0	0	0	0	0	0	0	0

ELAT – EPROM LATch control

0 = EPROM address and data bus configured for normal reads

1 = EPROM address and data bus configured for programming (writes to EPROM cause address and data to be latched). EPROM is in programming mode and cannot be read if ELAT is 1. This bit should not be set when no programming voltage is applied to the V_{pp} pin.

PGM – EPROM ProGraM command

0 = Programming power is switched OFF from EPROM array.

1 = Programming power is switched ON to EPROM array. If ELAT ≠ 1, then PGM = 0.

A.5.2 Programming Sequence

The EPROM programming sequence is:

1. Set the ELAT bit
2. Write the data to the address to be programmed
3. Set the PGM bit
4. Delay for a time t_{PGMR}
5. Clear the PGM bit
6. 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-2 shows the flow required to successfully program the EPROM.

A.6 EPROM PROGRAMMING SPECIFICATIONS

Table A-2. EPROM Programming Electrical Characteristics

(V_{DD} = 5V ± 10%, V_{SS} = 0 Vdc, T_A = 0°C to +70°C, unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Programming Voltage $\overline{IRQ}/V_{PP}$	V_{PP}	12.5	—	14.5	V
Programming Current $\overline{IRQ}/V_{PP}$	I_{PP}				mA
Programming Time per byte	t_{EPGM}	—	3	—	ms

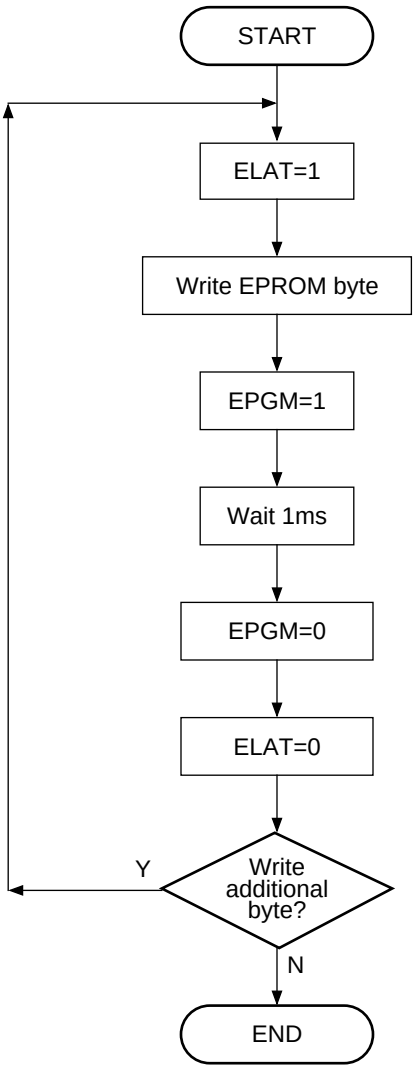


Figure A-2. EPROM Programming Sequence

How to reach us:

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

INTERNET: http://www.mot-sps.com/csic

USA/EUROPE/Locations Not Listed: Motorola Literature Distribution; P.O. Box 5405, Denver, Colorado 80217. 303-675-2140 or 1-800-441-2447

JAPAN: Nippon Motorola Ltd.; Tatsumi-SPD-JLDC, 6F Seibu-Butsuryu-Center, 3-14-2 Tatsumi Koto-Ku, Tokyo 135, Japan. 03-81-3521-8315

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