MOTOROLA
intelligence everywhere™

digital dna™

MC68HC05L25

Technical Data

M68HC05
Microcontrollers

MC68HC05L25/D
Rev. 3, 5/2002

WWW.MOTOROLA.COM/SEMICONDUCTORS

MC68HC05L25

Technical Data

To provide the most up-to-date information, the revision of our documents on the World Wide Web will be the most current. Your printed copy may be an earlier revision. To verify you have the latest information available, refer to:

http://www.motorola.com/semiconductors/

The following revision history table summarizes changes contained in this document. For your convenience, the page number designators have been linked to the appropriate location.

Revision History

Revision History

Date	Revision Level	Description	Page Number(s)
May, 2002	3.0	Reformatted to add additional page references and correct World Wide Web address	N/A

List of Sections

List of Sections

Table of Contents

Section 1. General Description

Table of Contents

Section 2. Memory Map

Section 3. Operating Modes

Section 4. Central Processor Unit (CPU) Core

Section 5. Resets

Section 6. Interrupts

Section 7. Input/Output Ports (I/O)

Section 8. Oscillators and Clock

Section 9. Time Base

Section 10. Serial Peripheral Interface

Section 11. LCD Driver

Section 12. Analog Subsystem

Section 13. Event Counter

Section 14. Instruction Set

Section 15. Electrical Specifications

Section 16. Mechanical Specifications

Section 17. Ordering Information

Table of Contents

List of Figures

List of Figures

Figure	Title	Page

List of Figures

List of Tables

List of Tables

Section 1. General Description

1.1 Contents

1.2 Introduction

The Motorola MC68HC05L25 is a member of M68HC05 Family of low-cost microcontroller units (MCUs). A functional block diagram of the MC68HC05L25 is shown in **Figure 1-1**.

1.3 Features

- Low-Cost, HC05 Core
- 48-Pin Quad Flat Pack (VQFP) and
  52-Pin Quad Flat Pack (TQFP)
- 6160 Bytes of User ROM, Including 16 Bytes of User Vectors
- 176 Bytes of User RAM
- 24 x 4 or 25 x 3 Multiplexed Liquid Crystal Display (LCD) Driver
- Serial Peripheral Interface (SPI)
- Two-Channel Analog-to-Digital (A/D) Converter
- 16-Bit Event Counter
- Time Base Timer
- Computer Operating Properly (COP) Watchdog Timer
- Infrared (IR) Remote Carrier Output (Software Selectable 33-to-67 Percent or 50-to-50 Percent Duty)
- Buzzer Output (Software-Selectable Frequencies)
- 20 Bidirectional Input/Output (I/O) Lines, Including:
  - Four Key Wakeup Input Lines
  - Software-Programmable Pullups
  - Software-Programmable Open-Drain Lines
  - High-Current (20 mA) Lines
- Software-Selectable Sensitivity on IRQ Interrupt
  (Edge- and Level-Sensitive or Edge-Sensitive Only)
- STOP Instruction Disable Option
- On-Chip Dual 4-MHz/32-kHz (Typical) Oscillator Circuits
- Single-Chip, Self-Check, and Test Modes
- Power-Saving Stop and Wait Modes

1.4 Mask Options

Table 1-1 shows the mask programmable options available on the MC68HC05L25.

Table 1-1. Mask Option Selection

Name	Selection	Description
RSTR		$\overline{\text{RESET}}$ pin pullup resistor
	RE	RESET pullup resistor enable (connected)
	RD	RESET pullup resistor disable (not connected)
OSCR		OSC feedback resistor
	OE	OSC feedback resistor enable (connected)
	OD	OSC feedback resistor disable (not connected)
XOSCR		XOSC feedback/ damping resistor
	XE	XOSC feedback and damping resistor enable (both connected)
	XD	XOSC feedback and damping resistor disable (both not connected)
STOPE		STOP instruction
	SE	STOP instruction enable (STOP instruction operates normally)
	SD	STOP instruction disable (executing STOP will not halt OSC clock)

1.5 MCU Structure

The overall block diagram of the MC68HC05L25 is shown in Figure 1-1.

General Description

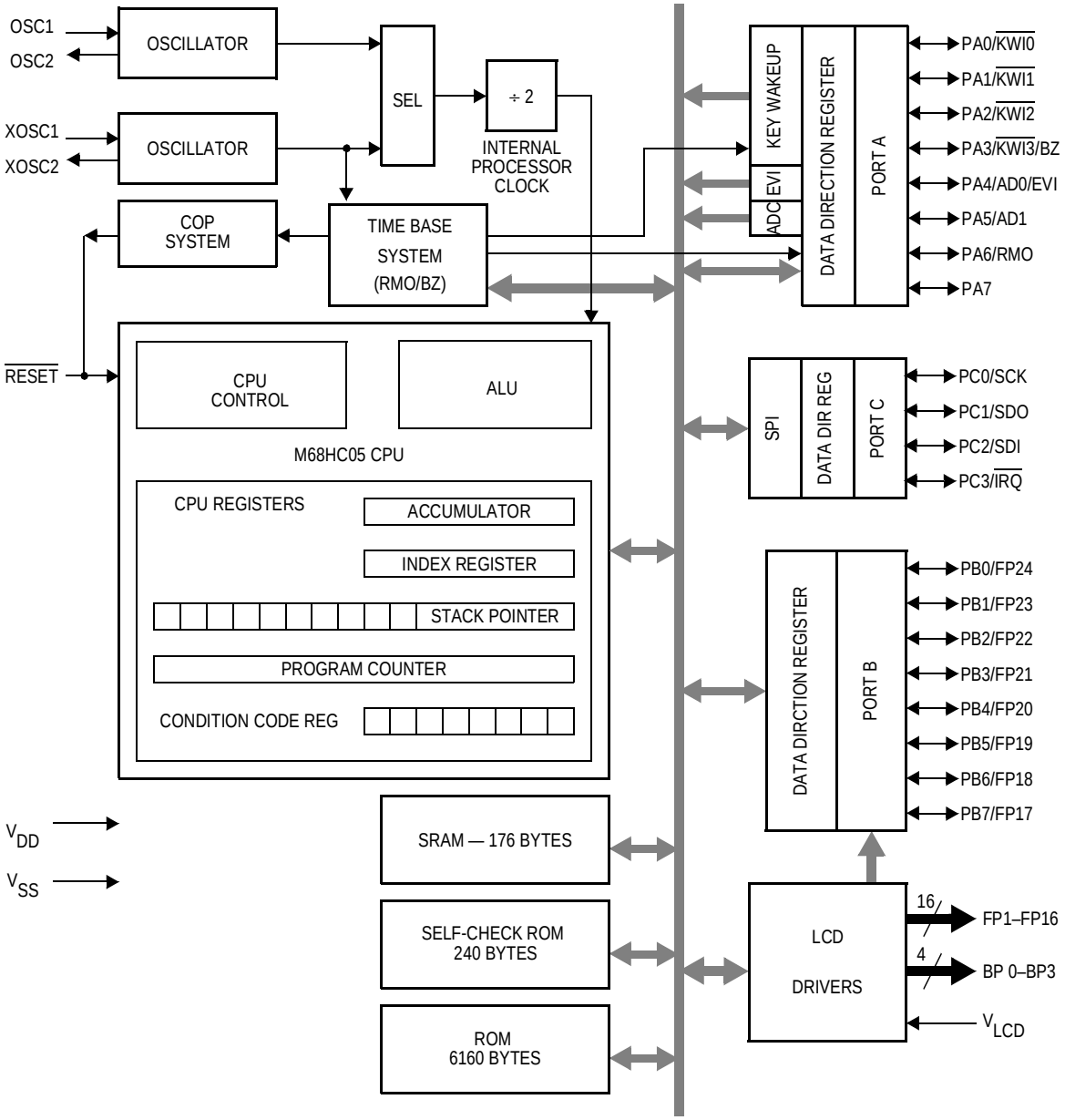


Figure 1-1. MC68HC05L25 Block Diagram

1.6 Functional Pin Description

NOTE: *A line over a signal name indicates an active low signal. For example, RESET is active high and $\overline{RESET}$ is active low. Any reference to voltage, current, resistance, capacitance, time, or frequency specified in the following paragraphs will refer to the nominal values. The exact values and their tolerance or limits are specified in* ***Section 15. Electrical Specifications****.*

The MC68HC05L25 is available in the 48-pin VQFP and 52-pin TQFP. The pin assignments for the 48-pin VQFP are shown in **Figure 1-2** and **Table 1-2**.

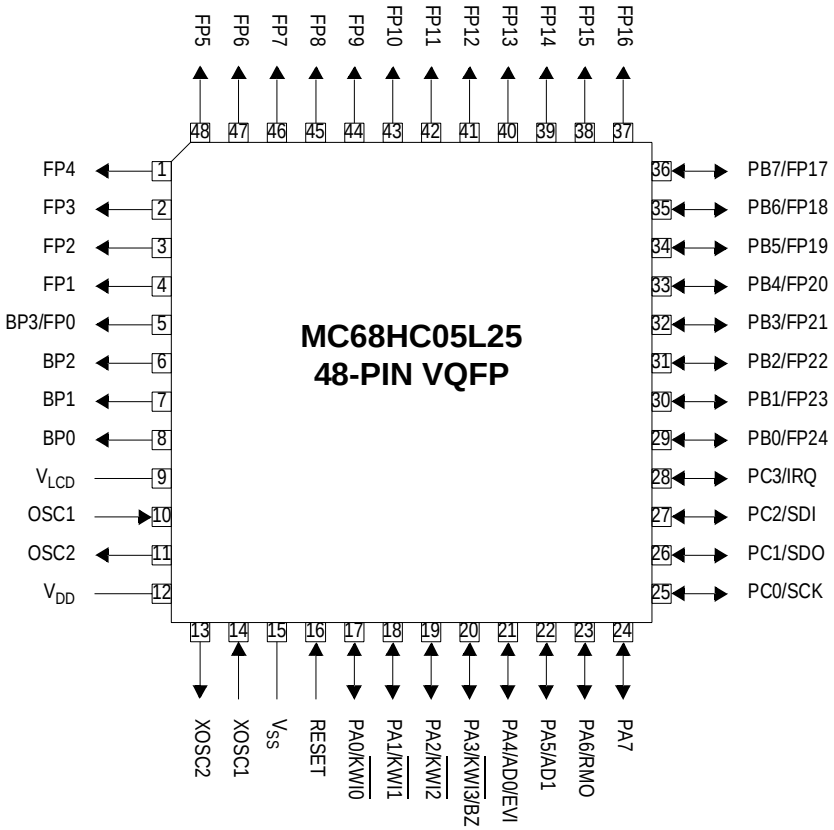


Figure 1-2. 48-Pin VQFP Single-Chip Mode Pinout

General Description

Table 1-2 summarizes the 48-pin VQFP pin configurations.

Table 1-2. 48-Pin VQFP Pin Configurations

Pin No.	Pin Name	I/O
10	OSC1	I
11	OSC2	O
12	V_{DD}	DC
13	XOSC2	O
14	XOSC1	I
15	V_{SS}	DC
16	$\overline{\text{RESET}}$	I
17	PA0/$\overline{\text{KWI0}}$	I/O
18	PA1/$\overline{\text{KWI1}}$	I/O
19	PA2/$\overline{\text{KWI2}}$	I/O
20	PA3/$\overline{\text{KWI3}}$/BZ	I/O
21	PA4/AD0/EVI	I/O
22	PA5/AD1	I/O
23	PA6/RMO	I/O
24	PA7	I/O
25	PC0/SCK	I/O
26	PC1/SDO	I/O
27	PC2/SDI	I/O
28	PC3/$\overline{\text{IRQ}}$	I/O
29	PB0/FP24	I/O
30	PB1/FP23	I/O
31	PB2/FP22	I/O
32	PB3/FP21	I/O
33	PB4/FP20	I/O
34	PB5/FP19	I/O
35	PB6/FP18	I/O
36	PB7/FP17	I/O
37	FP16	O
38	FP15	O
39	FP14	O
40	FP13	O
41	FP12	O
42	FP11	O
43	FP10	O
44	FP9	O
45	FP8	O
46	FP7	O
47	FP6	O
48	FP5	O
1	FP4	O
2	FP3	O
3	FP2	O
4	FP1	O
5	BP3/FP0	O
6	BP2	O
7	BP1	O
8	BP0	O
9	V_{LCD}	DC

The pin assignments for the 52-pin TQFP are shown in **Figure 1-3** and **Table 1-3**.

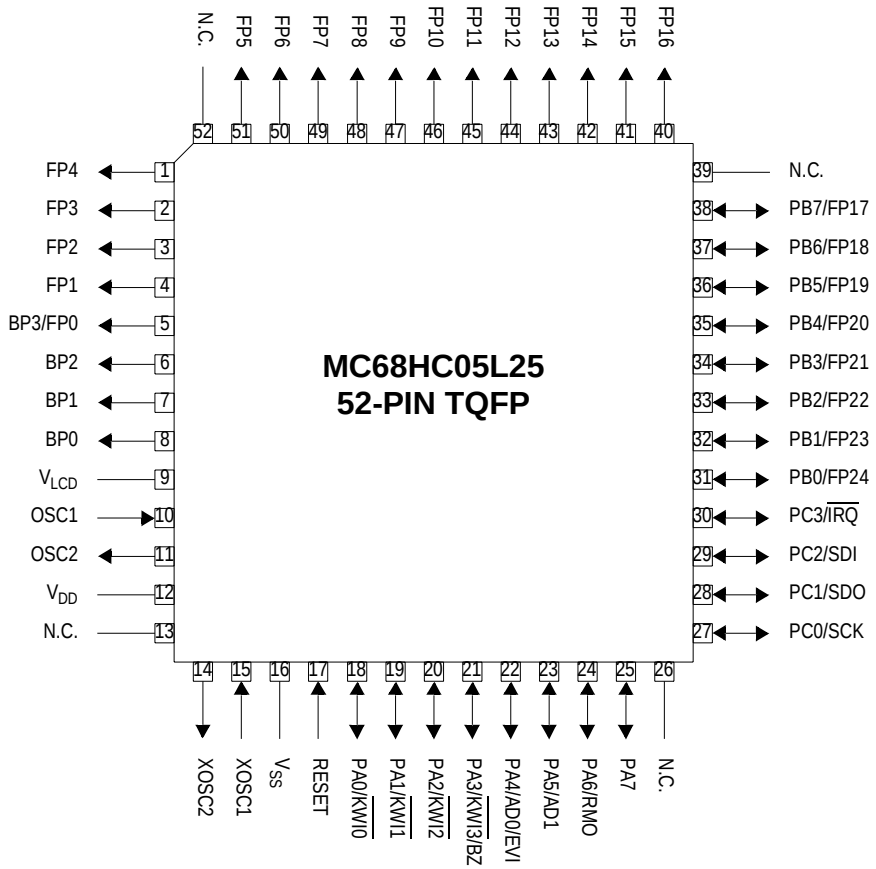


Figure 1-3. 52-Pin TQFP Single-Chip Mode Pinout

General Description

Table 1-3 summarizes the 52-pin VQFP pin configurations.

Table 1-3. 52-Pin TQFT Pin Configurations

Pin No.	Pin Name	I/O
10	OSC1	I
11	OSC2	O
12	V_{DD}	DC
14	XOSC2	O
15	XOSC1	I
16	V_{SS}	DC
17	RESET	I
18	PA0/$\overline{\text{KWI0}}$	I/O
19	PA1/$\overline{\text{KWI1}}$	I/O
20	PA2/$\overline{\text{KWI2}}$	I/O
21	PA3/$\overline{\text{KWI3}}$/BZ	I/O
22	PA4/AD0/EVI	I/O
23	PA5/AD1	I/O
24	PA6/RMO	I/O
25	PA7	I/O
27	PC0/SCK	I/O
28	PC1/SDO	I/O
29	PC2/SDI	I/O
30	PC3/$\overline{\text{IRQ}}$	I/O

13	N.C.	
26	N.C.	
39	N.C.	
52	N.C.	

Pin No.	Pin Name	I/O
31	PB0/FP24	I/O
32	PB1/FP23	I/O
33	PB2/FP22	I/O
34	PB3/FP21	I/O
35	PB4/FP20	I/O
36	PB5/FP19	I/O
37	PB6/FP18	I/O
38	PB7/FP17	I/O
40	FP16	O
41	FP15	O
42	FP14	O
43	FP13	O
44	FP12	O
45	FP11	O
46	FP10	O
47	FP9	O
48	FP8	O
49	FP7	O
50	FP6	O
51	FP5	O
1	FP4	O
2	FP3	O
3	FP2	O
4	FP1	O
5	BP3/FP0	O
6	BP2	O
7	BP1	O
8	BP0	O
9	V_{LCD}	DC

The following paragraphs describe the general function of each pin.

1.6.1 V_{DD} and V_{SS}

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

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

1.6.2 V_{LCD}

This pin provides an offset to the LCD driver bias for adjusting the contrast of LCD. See **Section 11. LCD Driver** for additional information.

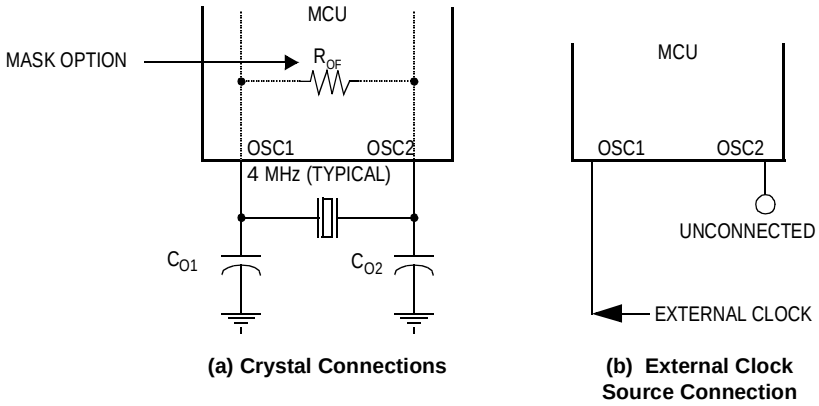
1.6.3 $\overline{\text{RESET}}$

This pin can be used as an input to reset the MCU to a known startup state by pulling it to the low state. The $\overline{\text{RESET}}$ pin contains a steering diode to discharge any voltage on the pin to V_{DD} when the power is removed. The $\overline{\text{RESET}}$ pin contains an internal Schmitt trigger to improve its noise immunity as an input. Refer to **Section 5. Resets**.

1.6.4 OSC1 and OSC2

The OSC1 and OSC2 pins are the connections for the 2-pin on-chip oscillator. The OSC1 and OSC2 pins can accept these sets of components:

1. A crystal or ceramic resonator as shown in **Figure 1-4(a)**
2. An external clock signal as shown in **Figure 1-4(b)**



(a) Crystal Connections

(b) External Clock Source Connection

Figure 1-4. Oscillator Connections

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

1.6.4.1 Crystal or Ceramic Resonator

The circuit in **Figure 1-4(a)** shows a typical 2-pin oscillator circuit for an AT-cut, parallel, resonant crystal. The crystal manufacturer's recommendations should be followed, since the crystal parameters determine the external component values required to provide maximum stability and reliable startup. 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 startup stabilization and to minimize output distortion. An internal startup feedback resistor of R_{OF} between OSC1 and OSC2 can be selected as a mask option.

1.6.4.2 External Clock

An external clock from another CMOS-compatible device can be connected to the OSC1 input, with the OSC2 output not connected, as shown in **Figure 1-4(b)**. This configuration is possible regardless of the oscillator setup.

1.6.5 XOSC1 and XOSC2

The XOSC1 and XOSC2 pins are the connections for the 2-pin on-chip oscillator. The XOSC1 and XOSC2 pins can accept these sets of components:

1. A crystal as shown in **Figure 1-5(a)**
2. An external clock signal as shown in **Figure 1-5(b)**

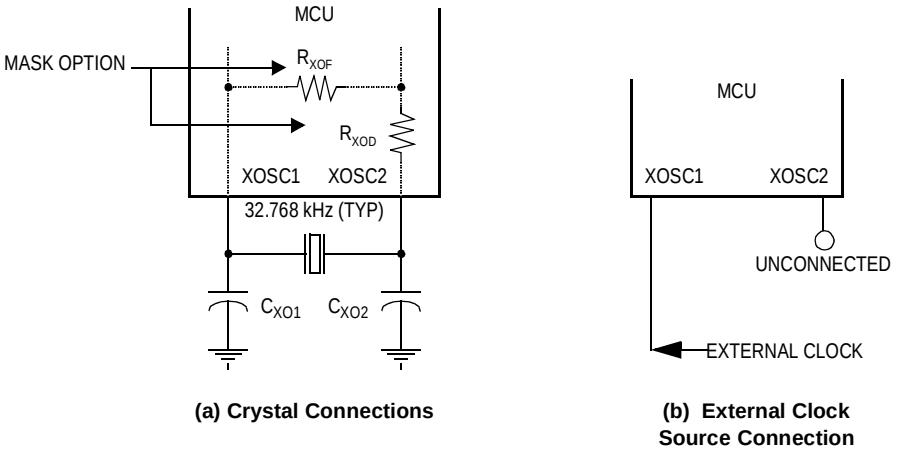


Figure 1-5. Oscillator Connections

The frequency, f_{XOSC}, of the oscillator or external clock source is divided by two to produce the internal operating frequency, f_{OP}, if selected by SYS1:SYS0 bits.

1.6.5.1 Crystal

The circuit in **Figure 1-5(a)** shows a typical 2-pin oscillator circuit for a 32.768-kHz “watch” crystal. The crystal manufacturer’s recommendations should be followed, since the crystal parameters determine the external component values required to provide maximum stability and reliable startup. 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. An internal startup feedback resistor of R_{xof} between XOSC1 and XOSC2 and a

damping resistor of R_{xod} in series to XOSC2 can be selected as a mask option.

1.6.5.2 External Clock

As shown in **Figure 1-5(b)**, an external clock from another CMOS-compatible device can be connected to the XOSC1 input (with the XOSC2 output not connected). This configuration is possible regardless of the oscillator setup.

1.6.5.3 XOSC Not Used

When XOSC is not used, the XOSC1 pin must be connected to the RESET pin to assure proper initialization of the clock circuitry. XOSC2 pin should remain unconnected.

1.6.6 PA0–PA2/KWI0–KWI2, PA3/KWI3/BZ, PA4/AD0/EVI, PA5/ADI, PA6/RMO, and PA7

Port A is an 8-bit I/O port. The state of any pin is software programmable and all port A lines are configured as inputs during power-on or reset. Bits 0 through 3 are shared with the key wakeup subsystem, and bit 3 also is shared with the buzzer subsystem. Bit 4 is shared with the A/D converter and event counter. Bit 5 is shared with the A/D converter. Bit 6 is shared with the infrared (IR) remote output. See **Section 7. Input/Output Ports (I/O)** for more details on the I/O ports.

1.6.7 PB0–PB7/FP24–FP17

These eight I/O lines comprise port B. The state of any pin is software programmable, and all bits are configured as LCD output during power-on or reset. These bits are shared with LCD frontplane drivers. See **Section 7. Input/Output Ports (I/O)** for more details on the I/O ports.

1.6.8 PC0/SCK, PC1/SDO, PC2/SDI, and PC3/$\overline{\text{IRQ}}$

These four I/O lines comprise port C. Bits 0 through 2 are shared with the SPI subsystem. Bit 3 is shared with the $\overline{\text{IRQ}}$ input. The state of any pin is software programmable, and all port C lines are configured as port inputs during power-on or reset. Each port C pin can be configured with a pullup resistor by a software option. SPI output pins SCK and SDO can be configured as open-drain outputs by a software option. See **Section 7. Input/Output Ports (I/O)** for more details on the I/O ports. The PC3/$\overline{\text{IRQ}}$ pin is used for special mode entry. Do not apply voltages above V_{DD} for normal single-chip mode operation. See **Section 15. Electrical Specifications** for more details.

1.6.9 BP3/FP0, FP1–FP18, and PB0–PB7/FP24–FP17

The LCD display has 25 frontplane drivers. Frontplanes 17 through 24 are shared with port B bits 7 through 0, respectively. Frontplane 0 is shared with backplane 3. See **Section 11. LCD Driver** for additional information.

1.6.10 BP0–BP2 and BP3/FP0

The LCD display has four backplane drivers. Backplane 3 is multiplexed with frontplane 0. See **Section 11. LCD Driver** for additional information.

General Description

Section 2. Memory Map

2.1 Contents

2.2 Introduction

When the MC68HC05L25 is in the single-chip mode, 80 bytes of input/output (I/O) registers, 176 bytes of user RAM (including a 64-byte stack), 6144 bytes of user ROM, and 16 bytes of user vectors are available in the 8-K memory map as shown in **Figure 2-1**.

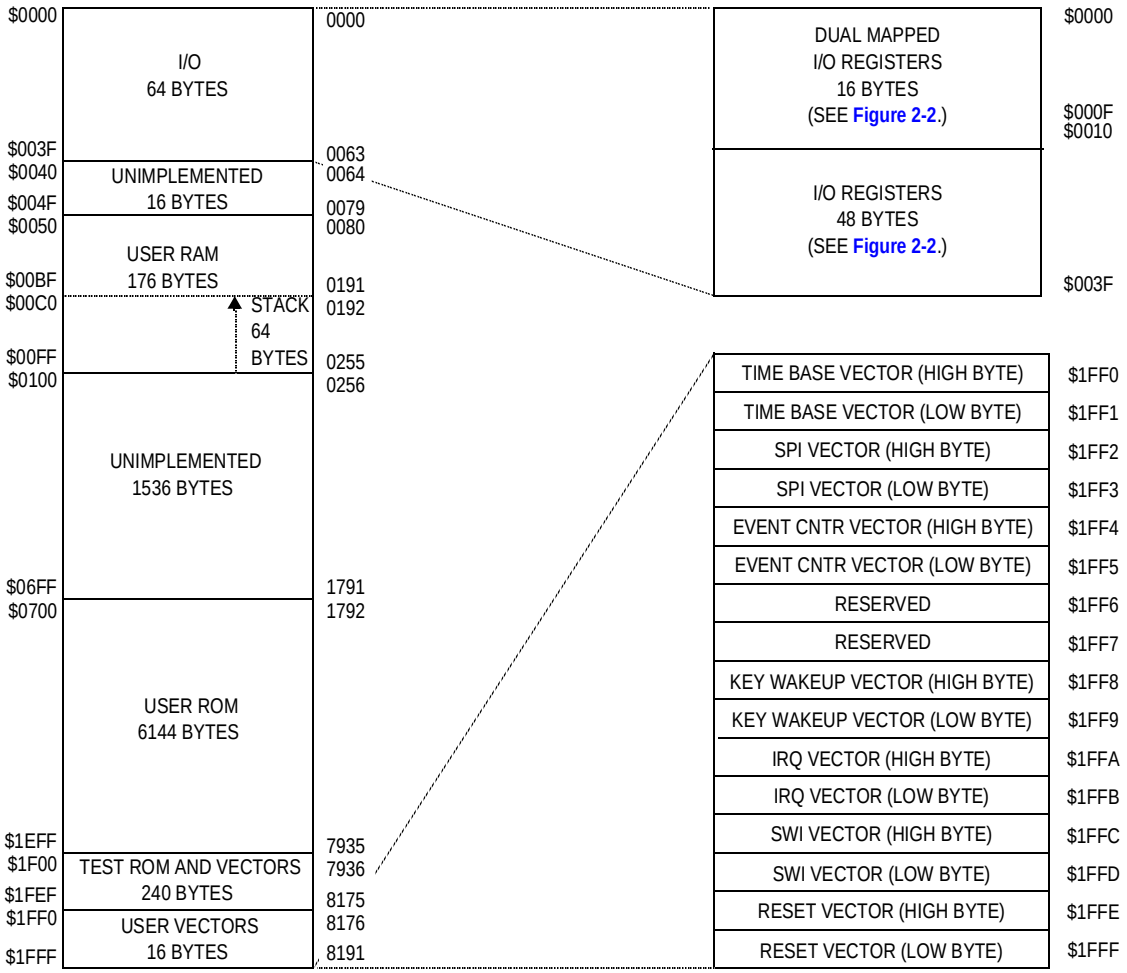


Figure 2-1. MC68HC05L25 Single-Chip Mode Memory Map

2.3 Input/Output and Control Registers

The I/O and control registers reside in locations $0000 through $003F. A summary of these registers is shown in **Figure 2-3**. The bit assignments for each register are shown in **Figure 2-4**. Reading from unimplemented bits (denoted with —) will return unknown states (unless explicitly defined to read 0), and writing to unimplemented bits will have no effect. See also **Figure 2-2**.

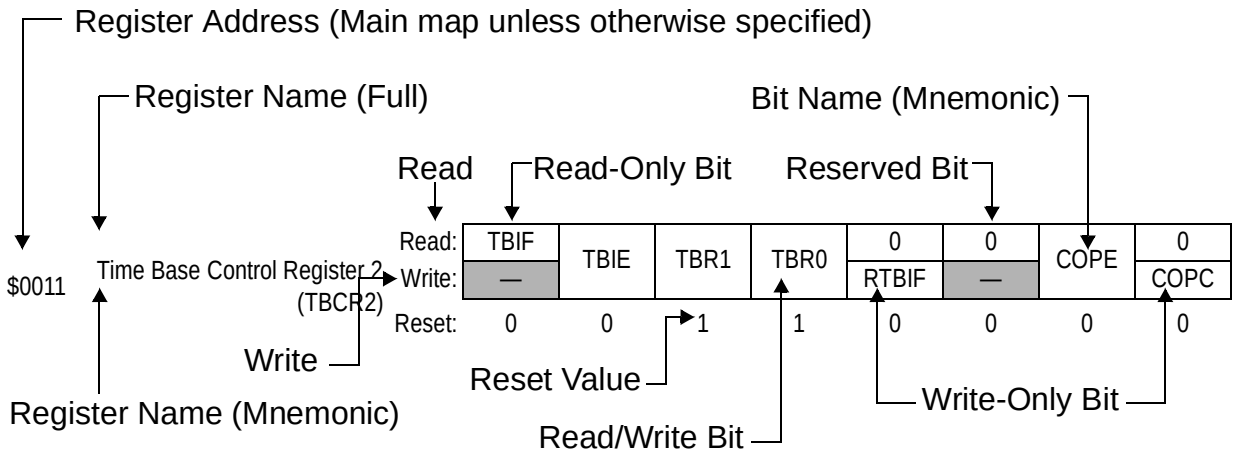


Figure 2-2. Register Description Key

2.3.1 Read/Write Bits

Read/write bits are typically control bits. They are, in general, not modified by a module. Reset: indicates the initial value of the latch.

2.3.2 Read-Only Bits

Read-only bits are status flag bits. They are indicators of module status. Reset: indicates the value that will be read immediately after system reset or before the module is enabled.

2.3.3 Write-Only Bits

Write-only bits are control bits. They typically return a state of 0 to prevent an inadvertent write to this bit by a READ-MODIFY-WRITE instruction. Reset: indicates the value that will be read immediately after system reset, which is the forced read value (typically 0).

2.3.4 Reserved Bits

Reserved bits are read-only bits that typically read 0. Writes to these bits are ignored, and the user should not write 1 for future compatibility. Reset: indicates the value that will be read immediately after system reset which is the forced read value of 0, typically.

2.3.5 Reset Value

Values specified on the row marked Reset: are initial values of register bits after system reset. Those bits unaffected by reset are marked with the letter U. Those bits that are unaffected by reset but initialized by power-on reset are marked with an asterisk (*).

2.3.6 Option Map

Address locations $0000 through $000F are dual mapped. When the OPTM bit in the MISC register is cleared, the main address map is accessed. When the OPTM bit in the MISC register is set, the option address map is accessed.

NOTE: *Although not necessary for this device, the OPTM bit should be cleared when accessing memory locations $0010 and above for future compatibility.*

2.3.7 Random-Access Memory (RAM)

The user RAM consists of 176 bytes (including the stack) at locations $0050 through $00FF. The stack can access 64 locations beginning at address $00FF and proceeding down to $00C0.

NOTE: *Using the stack area for data storage or temporary work locations requires care to prevent it from being overwritten due to stacking from an interrupt or subroutine call.*

2.4 Read-Only Memory (ROM)

This chip has a total of 6160 bytes of ROM. These are implemented as 6144 bytes of user ROM at locations $0700 through $1EFF and 16 bytes of user vectors at locations $1FF0 through $1FFF. A total of 240 bytes of test ROM and vectors is located from $1F00 through $1FEF.

Address	Name (Main Map Registers)
$0000	Port A Data Register
$0001	Port B Data Register
$0002	Port C Data Register
$0003	Unimplemented
$0004	Unimplemented
$0005	Unimplemented
$0006	Unimplemented
$0007	Unimplemented
$0008	Interrupt Control Register
$0009	Interrupt Status Register
$000A	Serial Peripheral Control Register
$000B	Serial Peripheral Status Register
$000C	Serial Peripheral Data Register
$000D	Unimplemented
$000E	Unimplemented
$000F	Unimplemented
$0010	Time Base Control Register 1
$0011	Time Base Control Register 2
$0012	Unimplemented
$0013	Unimplemented
$0014	Unimplemented
$0015	Unimplemented
$0016	Unimplemented
$0017	Unimplemented
$0018	Unimplemented
$0019	Unimplemented
$001A	Unimplemented
$001B	Unimplemented
$001C	Unimplemented
$001D	A/D Data Register
$001E	A/D Control/Status Register
$001F	Time Base Control Register 3
$0020	LCD Control Register
$0021	LCD Data Register 1
$0022	LCD Data Register 2
$0023	LCD Data Register 3
$0024	LCD Data Register 4
$0025	LCD Data Register 5
$0026	LCD Data Register 6
$0027	LCD Data Register 7

Address	Name
$0028	LCD Data Register 8
$0029	LCD Data Register 9
$002A	LCD Data Register 10
$002B	LCD Data Register 11
$002C	LCD Data Register 12
$002D	LCD Data Register 13
$002E	Event Counter Control/Status Register
$002F	Event Counter Timing Register
$0030	Event Counter Data High Register
$0031	Event Counter Data Low Register
$0032	Unimplemented
$0033	Unimplemented
$0034	Unimplemented
$0035	Unimplemented
$0036	Unimplemented
$0037	Unimplemented
$0038	Unimplemented
$0039	Unimplemented
$003A	Unimplemented
$003B	Unimplemented
$003C	Unimplemented
$003D	Unimplemented
$003E	Miscellaneous Register
$003F	Unimplemented

OPTN Address	Name (Option Map Registers)
$0000	Port A Data Direction Register
$0001	Port B Data Direction Register
$0002	Port C Data Direction Register
$0003	Unimplemented
$0004	Unimplemented
$0005	Unimplemented
$0006	Unimplemented
$0007	Unimplemented
$0008	Resistor Control Register
$0009	Resistor Control Register
$000A	Wired-OR Mode Register
$000B	Unimplemented
$000C	Unimplemented
$000D	Unimplemented
$000E	Key Wakeup Input Enable Register
$000F	Mask Option Status Register

Figure 2-3. I/O Register Memory Map Summary

Memory Map

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$0000	Port A Data Register (PORTA) See page 75.	Read:	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
		Write:								
		Reset:	Unaffected by reset							
$0001	Port B Data Register (PORTB) See page 85.	Read:	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
		Write:								
		Reset:	Unaffected by reset							
$0002	Port C Data Register (PORTC) See page 93.	Read:	0	0	0	0	PC3	PC2	PC1	PC0
		Write:								
		Reset:	0	0	0	0	U	U	U	U
$0003 ↓ $0007	Unimplemented									
$0008	Interrupt Control Register (INTCR) See page 64.	Read:	IRQE	0	0	KWIE	IRQS	0	0	0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$0009	Interrupt Status Register (INTSR) See page 65.	Read:	IRQF	0	0	KWIF	0	0	0	0
		Write:					RIRQ			RKWIF
		Reset:	0	0	0	0	0	0	0	0
$000A	Serial Peripheral Control Register (SPCR) See page 135.	Read:	SPIE	SPE	DORD	MSTR	0	0	0	SPR
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$000B	Serial Peripheral Status Register (SPSR) See page 137.	Read:	SPIF	DCOL	0	0	0	0	0	0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$000C	Serial Peripheral Data Register (SPDR) See page 138.	Read:	SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0
		Write:								
		Reset:	Unaffected by reset							
$000D ↓ $000F	Unimplemented									

= Unimplemented R = Reserved U = Unaffected

Figure 2-4. I/O Registers (Sheet 1 of 3)

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$0020	LCD Control Register (LCDCR) See page 144.	Read:	LCDE	PBEH	DUTY	PBEL	0	0	FC	LC
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$0021	LCD Data Register (LDAT1) See page 148.	Read:	F1B3	F1B2	F1B1	F1B0	F0B3	F0B2	F0B1	F0B0
		Write:								
		Reset:	Unaffected by reset							
$0022	LCD Data Register (LDAT2) See page 148.	Read:	F3B3	F3B2	F3B1	F3B0	F2B3	F2B2	F2B1	F2B0
		Write:								
		Reset:	Unaffected by reset							
$0023	LCD Data Register (LDAT3) See page 148.	Read:	F5B3	F5B2	F5B1	F5B0	F4B3	F4B2	F4B1	F4B0
		Write:								
		Reset:	Unaffected by reset							
$0024	LCD Data Register (LDAT4) See page 148.	Read:	F7B3	F7B2	F7B1	F7B0	F6B3	F6B2	F6B1	F6B0
		Write:								
		Reset:	Unaffected by reset							
$0025	LCD Data Register (LDAT5) See page 148.	Read:	F9B3	F9B2	F9B1	F9B0	F8B3	F8B2	F8B1	F8B0
		Write:								
		Reset:	Unaffected by reset							
$0026	LCD Data Register (LDAT6) See page 148.	Read:	F11B3	F11B2	F11B1	F11B0	F10B3	F10B2	F10B1	F10B0
		Write:								
		Reset:	Unaffected by reset							
$0027	LCD Data Register (LDAT7) See page 148.	Read:	F13B3	F13B2	F13B1	F13B0	F12B3	F12B2	F12B1	F12B0
		Write:								
		Reset:	Unaffected by reset							
$0028	LCD Data Register (LDAT8) See page 148.	Read:	F15B3	F15B2	F15B1	F15B0	F14B3	F14B2	F14B1	F14B0
		Write:								
		Reset:	Unaffected by reset							
$0029	LCD Data Register (LDAT9) See page 148.	Read:	F17B3	F17B2	F17B1	F17B0	F16B3	F16B2	F16B1	F16B0
		Write:								
		Reset:	Unaffected by reset							

= Unimplemented R = Reserved U = Unaffected

Figure 2-4. I/O Registers (Sheet 2 of 3)

Memory Map

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$002A	LCD Data Register (LDAT10) See page 148.	Read:	F19B3	F19B2	F19B1	F19B0	F18B3	F18B2	F18B1	F18B0
		Write:								
		Reset:	Unaffected by reset							
$002B	LCD Data Register (LDAT11) See page 148.	Read:	F21B3	F21B2	F21B1	F21B0	F20B3	F20B2	F20B1	F20B0
		Write:								
		Reset:	Unaffected by reset							
$002C	LCD Data Register (LDAT12) See page 148.	Read:	F23B3	F23B2	F23B1	F23B0	F22B3	F22B2	F22B1	F22B0
		Write:								
		Reset:	Unaffected by reset							
$002D	LCD Data Register (LDAT13) See page 148.	Read:	0	0	0	0	F24B3	F24B2	F24B1	F24B0
		Write:								
		Reset:	Unaffected by reset							
$002E	Event Control Status/Counter Register (EVSCR) See page 61.	Read:	EVCE	EVIE	EVOE	EVIF	EVOF	0	0	0
		Write:						RCCF	ROIF	
		Reset:								
$002F	Event Counter Timing Register (EVTR) See page 172.	Read:	WT3	WT2	WT1	WT0	MT3	MT2	MT1	MT0
		Write:								
		Reset:								
$0030	Event Counter Data Register High (EVDH) See page 176.	Read:	BIT15	BIT14	BIT13	BIT12	BIT11	BIT10	BIT9	BIT8
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$0031	Event Counter Data Register Low (EVDL) See page 176.	Read:	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$0032	Unimplemented									
↓										
$003D	Unimplemented									
$003E	Miscellaneous Register (MISC) See page 113.	Read:	FTUP	STUP	0	0	SYS1	SYS0	FOSCE	OPTM
		Write:								
		Reset:	U	U	0	0	0	0	1	0
$003F	Reserved		R	R	R	R	R	R	R	R

= Unimplemented R = Reserved U = Unaffected

Figure 2-4. I/O Registers (Sheet 3 of 3)

Section 3. Operating Modes

3.1 Contents

3.2 Introduction

The MC68HC05L25 has three modes of operation that affect the pinout and architecture of the MCU: single-chip mode, internal test mode, and expanded test mode. The single-chip mode normally will be used, while the test modes are required for the special needs of production test and burn-in.

3.3 Single-Chip Mode

Single-chip mode allows the MCU to function as a self-contained microcontroller with maximum use of the pins for on-chip peripheral functions. The pinout for the single-chip mode is shown in **Figure 1-2** and **Figure 1-3**.

In single-chip mode, all address and data activity occurs within the MCU and is not available externally.

3.4 Low-Power Modes

In each of its configuration modes, the MC68HC05L25 is capable of running in one of two low-power operational modes. The WAIT and STOP instructions provide two modes that reduce the power required for the MCU by stopping various internal clocks and/or the on-chip oscillator. The STOP and WAIT instructions are not normally used if the COP watchdog timer is enabled. The flow of the stop and wait modes is shown in **Figure 3-1**.

3.4.1 STOP Instruction

Execution of the STOP instruction places the MCU in its lowest power-consumption mode. In stop mode, the internal oscillator is turned off, halting *all* internal processing except the time base/COP watchdog timer, if it is enabled and clocked from XOSC.

Execution of the STOP instruction automatically clears the I bit in the condition code register. All other registers and memory remain unaltered. All input/output lines remain unchanged. Therefore, unused ports must be programmed as output or tied to the power rails to prevent excessive current consumption.

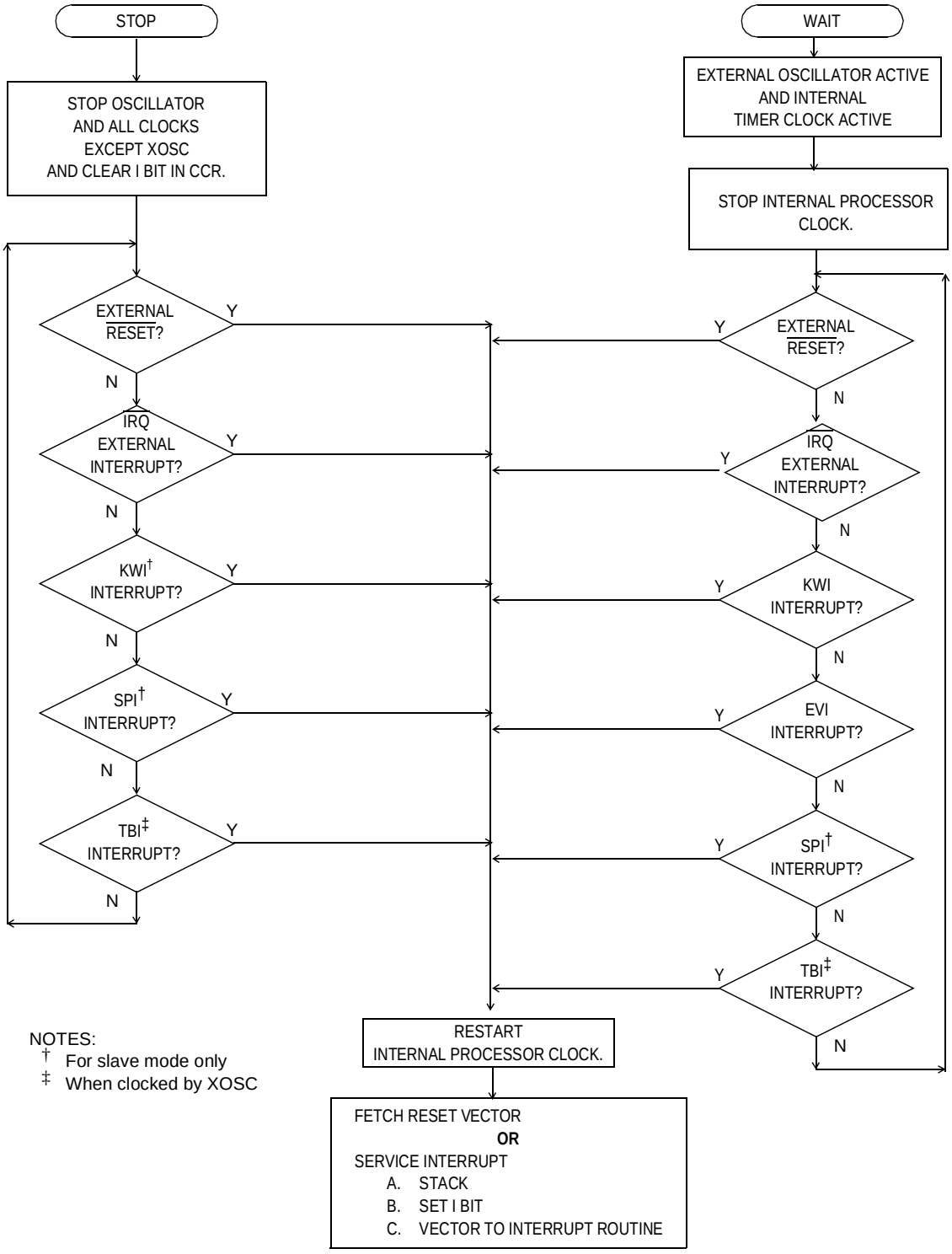


Figure 3-1. Stop/Wait Flowcharts

The MCU can be brought out of stop mode by an external IRQ interrupt, KWI interrupt, SPI (slave mode only) interrupt or TBI interrupt clocked by XOSC or a reset.

3.4.2 WAIT Instruction

The WAIT instruction places the MCU in a low-power mode, which consumes more power than stop mode. In wait mode, the internal processor clock is halted, suspending all processor and internal bus activity. Internal timer clocks remain active, permitting interrupts to be generated from the timer or a reset to be generated from the COP watchdog timer. Execution of the WAIT instruction automatically clears the I bit in the condition code register and external interrupt is allowed. All other registers, memory, and input/output lines remain in their previous states.

If time base interrupts are enabled, a time base interrupt will cause the processor to exit the wait mode and resume normal operation. The time base may be used to generate a periodic exit from the wait mode. The wait mode also may be exited when an external interrupt ($\overline{\text{IRQ}}$) or reset occurs.

3.5 COP Watchdog Timer Considerations

The COP watchdog timer is active in all modes of operation if enabled by a TBCR2 select bit. If the COP watchdog timer is selected by the TBCR2 bit, any execution of the STOP instruction (either intentional or inadvertent due to the CPU being disturbed) will cause the oscillator (OSC) to halt and thus the COP watchdog timer will not time out if driven from OSC. Thus for applications that require use of COP watchdog from OSC, STOP instruction must be disabled, or COP must be driven from XOSC.

If the COP watchdog timer is selected by the TBCR2 select bit, the COP will reset the MCU when it times out. Therefore, it is recommended that the COP watchdog should be **disabled** for a system that must have intentional uses of the wait or stop modes for periods longer than the COP timeout period.

Section 4. Central Processor Unit (CPU) Core

4.1 Contents

4.2 Introduction

The MC68HC05L25 has an 8-K memory map. Therefore, it uses 13 bits of the address bus.

4.3 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 4-1**.

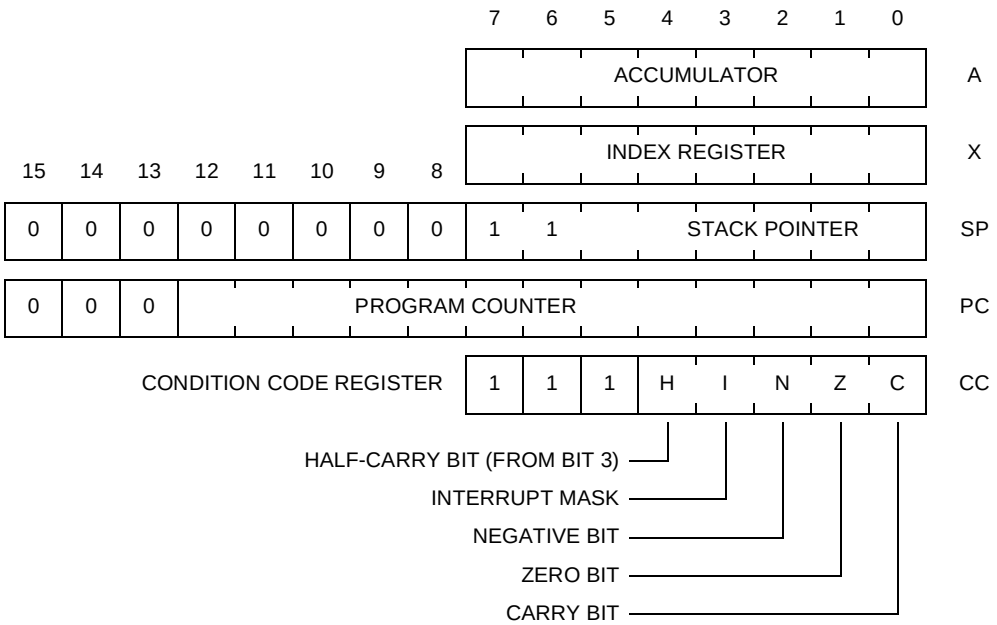


Figure 4-1. M68HC05 Programming Model

4.3.1 Accumulator

The accumulator is a general-purpose 8-bit register as shown in **Figure 4-1**. The CPU uses the accumulator to hold operands and results of arithmetic calculations or non-arithmetic operations. The accumulator is unaffected by a reset of the device.

4.3.2 Index Register

The index register shown in **Figure 4-1** is an 8-bit register that can perform two functions:

1. Indexed addressing
2. 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 contents 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 contents to a 16-bit immediate value.

The index register also can serve as an auxiliary accumulator for temporary storage. The index register is unaffected by a reset of the device.

4.3.3 Stack Pointer

The stack pointer shown in **Figure 4-1** is an internal 16-bit register. In devices with memory maps 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 from the stack.

When accessing memory, the 10 most significant bits are permanently set to 0000000011. The five least significant register bits are appended to these 11 fixed bits to produce an address within the range of $00FF to $00C0. Subroutines and interrupts **CAN** use up to 64 ($40) locations. If 64 locations are exceeded, the stack pointer wraps around and writes over the previously stored information. A subroutine call occupies two locations on the stack; and an interrupt uses five locations.

4.3.4 Program Counter

The program counter shown in **Figure 4-1** is an internal 16-bit register. In devices with memory maps less than 64 Kbytes, the unimplemented upper address lines are ignored, and memory image is mirrored. 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.

4.3.5 Condition Code Register

The condition code register shown in **Figure 4-1** 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 tested individually by a program, and specific actions can be taken as a result of their state. The condition code register should be thought of as having three additional upper bits that are always 1s. 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.

Half-Carry Bit (H Bit)

When the H bit is set, a carry has 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.

Interrupt Mask (I Bit)

When the I bit 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 set automatically 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 be cleared only by the clear I bit (CLI), STOP, or WAIT instructions.

Negative Bit (N Bit)

The N 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 1.)

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.

Zero Bit (Z Bit)

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

Carry/Borrow Bit (C Bit)

The C bit is set when a carry out of accumulator bit 7 occurs during the last arithmetic operation, logical operation, or data manipulation. The carry/borrow bit also is set or cleared during bit test and branch instructions and during shifts and rotates. This bit is not set by an INC or DEC instruction.

Section 5. Resets

5.1 Contents

5.2 Introduction

The MCU can be reset from three sources: one external input and two internal restart conditions. The RESET pin is an input with a Schmitt trigger as shown in **Figure 5-1**. All peripheral modules which drive external pins will be reset by the synchronous reset signal (RST) coming from a latch, which is synchronized to the PH2 bus clock and set by any of the three reset sources.

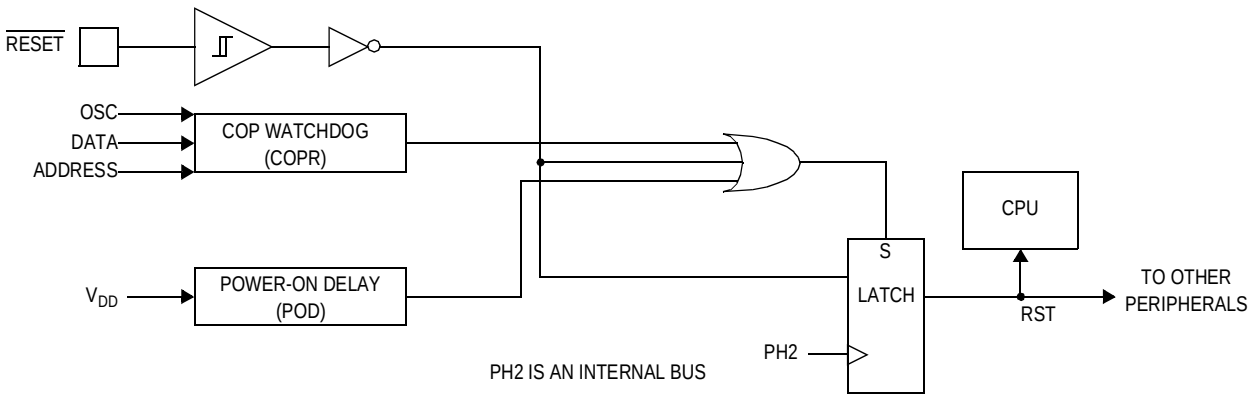


Figure 5-1. Reset Block Diagram

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

The $\overline{\text{RESET}}$ pin is the only external source of a reset. This pin is connected to a Schmitt trigger input gate to provide an upper and lower threshold voltage separated by a minimum amount of hysteresis. This external reset occurs whenever the $\overline{\text{RESET}}$ pin is pulled below the lower threshold and remains in reset until the $\overline{\text{RESET}}$ pin rises above the upper threshold. This active low input will generate the RST signal and reset the CPU and peripherals. Termination of the external $\overline{\text{RESET}}$ input or the internal COP watchdog reset are the only reset sources that can alter the operating mode of the MCU.

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

5.4 Internal Resets

The two internally generated resets are the initial power-on delay function and the COP watchdog timer reset. Termination of the external $\overline{\text{RESET}}$ input or the internal COP watchdog timer are the only reset sources that can alter the operating mode of the MCU. The other internal resets will not have any effect on the mode of operation when their reset state ends.

5.4.1 Power-On Delay (POD)

The internal POD is generated on power-up to allow the clock oscillator to stabilize. The POD is strictly for power turn-on conditions and is not able to detect a drop in the power supply voltage (brown-out). There is an oscillator stabilization delay of between 8,064 and 8,192 internal processor bus clock cycles (PH2) after the oscillator becomes active.

The power-on reset (POR) will generate the RST signal which will reset the CPU. If any other reset function is active at the end of this 8,064- to 8,192-cycle delay, the RST signal will remain in the reset condition until the other reset condition(s) end.

5.4.2 Computer Operating Properly Reset (COPR)

The internal COPR reset is generated automatically (if enabled via a TBCR2 select bit) by a timeout of the COP watchdog timer. This timeout occurs if the counter in the COP watchdog timer is not reset (cleared) within a specific time by a program reset sequence. The COP watchdog timer can be disabled by a TBCR2 select bit. Refer to **9.3.4 COP** for more information on this timeout feature.

The COPR will generate the RST signal which will reset the CPU and other peripherals. If any other reset function is active at the end of the COPR reset signal, the RST signal will remain in the reset condition until the other reset condition(s) end.

Section 6. Interrupts

6.1 Contents

6.2 Introduction

The MCU can be interrupted in these ways:

1. Nonmaskable software interrupt instruction (SWI)
2. External interrupt via $\overline{\text{IRQ}}$ (IRQ)
3. Serial peripheral interface interrupt (SPII)
4. Internal time base interrupt (TBI)
5. Key wakeup interrupt (KWI)
6. Event counter overflow interrupt (EVOF)

6.3 CPU Interrupt Processing

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

If interrupts are not masked (I bit in the CCR is clear) and the corresponding interrupt enable bit is set, 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 after the stacking operation, the interrupt with the highest vector location shown in **Table 6-1** will be serviced first. The SWI is executed in the same way 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 $1FF0 through $1FFF as defined in **Table 6-1**.

Table 6-1. Vector Address for Interrupts and Reset

Register	Flag Name	Interrupts	CPU Interrupt	Vector Address
N/A	N/A	Reset	RESET	$1FFE:$1FFF
N/A	N/A	Software	SWI	$1FFC:$1FFD
INTCR	IRQF	External Interrupt	IRQ	$1FFA:$1FFB
KWIEN	KWIF	Key Wakeup	KWI	$1FF8:$1FF9
—	—	Reserved	—	$1FF6:$1FF7
EVSCR	ECOF	Event Counter	EVI	$1FF4:$1FF5
SPSR	SPIF	Serial Peripheral	SPII	$1FF2:$1FF3
TBCR2	TBIF	Time Base Periodical	TBI	$1FF0:$1FF1

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 6-1** shows the sequence of events that occurs during interrupt processing.

6.4 Reset Interrupt Sequence

The reset function is not in the strictest sense an interrupt; however, it is acted upon in a similar manner as shown in **Figure 6-1**. A low level input on the $\overline{\text{RESET}}$ pin or internally generated RST signal causes the program to vector to its starting address, which is specified by the contents of memory locations $1FFE through $1FFF. The I bit in the condition code register also is set. The MCU is configured to a known state during this type of reset as previously described in **Section 5. Resets**.

6.5 Software Interrupt (SWI)

The SWI is an executable instruction and a nonmaskable interrupt since it is executed regardless of the state of the I bit in the CCR. If the I bit is

zero (interrupts enabled), the SWI instruction executes after interrupts which were pending before the SWI was fetched or before interrupts generated after the SWI was fetched. The interrupt service routine address is specified by the contents of memory locations $1FFC and $1FFD.

6.6 Hardware Interrupts

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

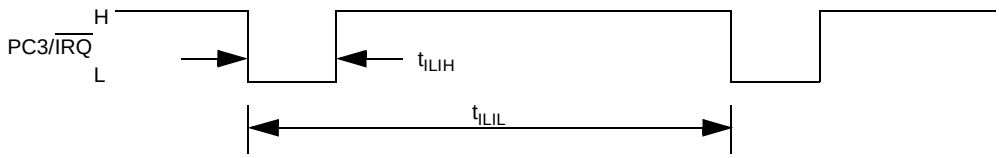
6.7 External Interrupt ($\overline{IRQ}$)

The $\overline{IRQ}$ pin provides an asynchronous interrupt to the CPU. The $\overline{IRQ}$ pin is enabled by the IRQE bit in the INTCR. Also see **7.5 Port C**. The interrupt service routine address is specified by the contents of memory locations $1FFA:$1FFB.

CPU instructions BIH and BIL test the pin state of the PC3/$\overline{IRQ}$ pin.

6.7.1 External Interrupt Trigger Condition

External interrupt (IRQ) is activated by the negative-edged signal.



The limit on the minimum pulse width (t_{ILIH}) is as specified. The pulse interval (t_{ILIL}) must be longer than the interrupt service routine's service time + 21 machine cycles.

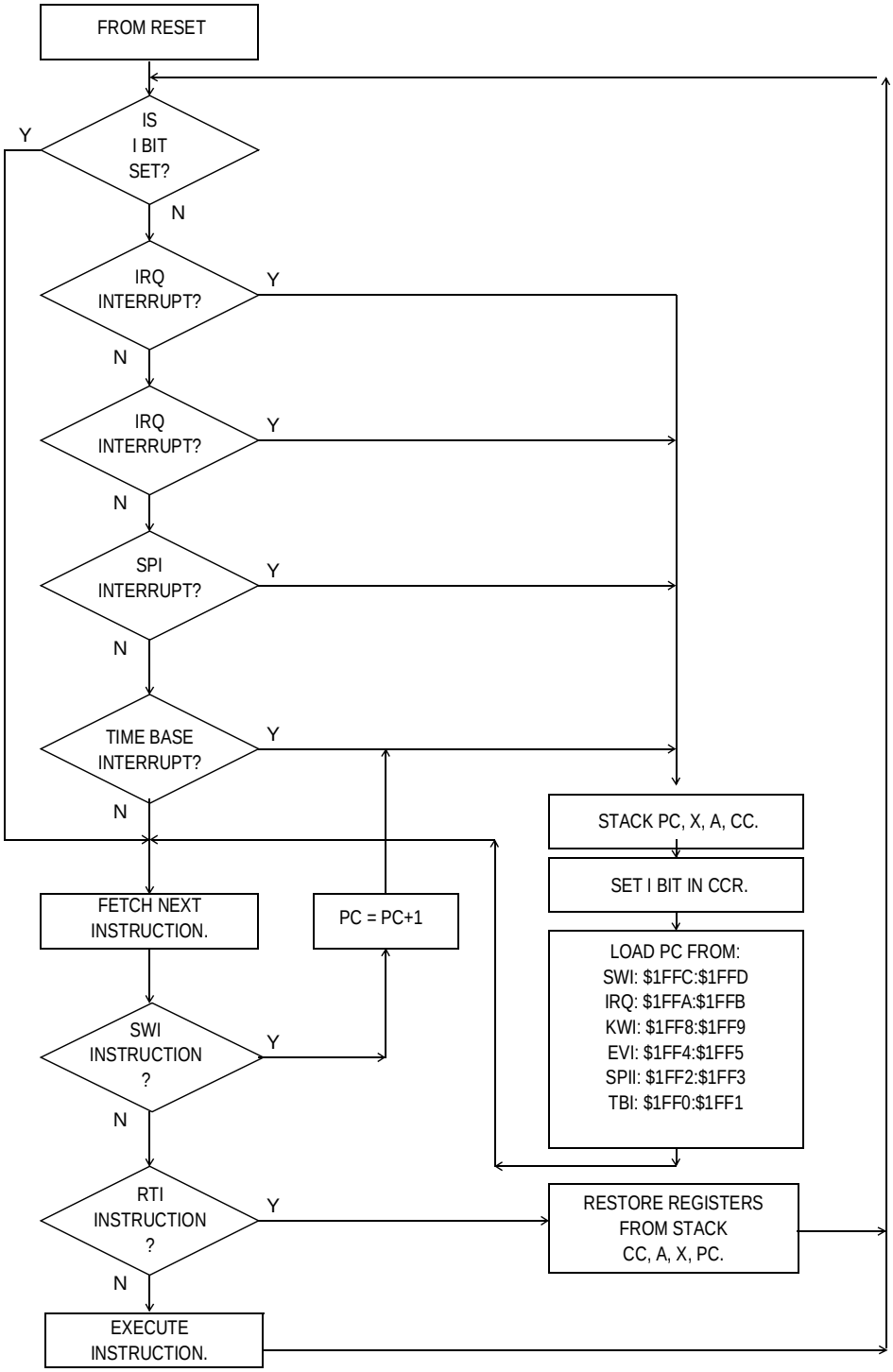


Figure 6-1. Interrupt Processing Flowchart

6.7.2 Interrupt Control Register

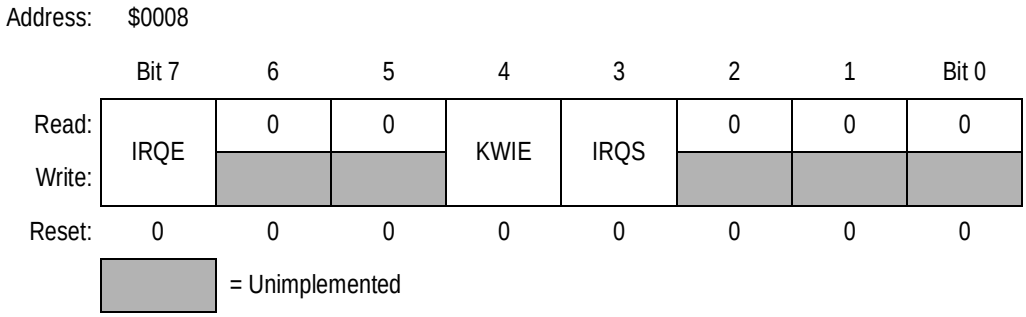


Figure 6-2. Interrupt Control Register (INTCR)

IRQE — External Interrupt (IRQ) Enable

The IRQE bit enables external interrupt when the interrupt mask is cleared and IRQF is set. This bit is cleared at reset.

1 = IRQ enabled
0 = IRQ disabled

Bits 6–5 — Reserved

These bits are not used and always return to zero.

KWIE — KWI Enable

The KWIE bit enables key wakeup interrupt when the KWIF bit is set. The KWIEx bit in the KWIEN register also must be set for enabling KWI. This bit is cleared at reset.

1 = KWI enabled
0 = KWI disabled

IRQS — External Interrupt (IRQ) Select Edge Sensitivity Only

The IRQS bit determines whether the LEVEL and EDGE or EDGE only will trigger the IRQ interrupt. This bit is cleared at reset.

1 = Trigger only on negative EDGEs
0 = Trigger on low LEVEL and negative EDGEs

Bits 2–0 — Reserved

These bits are not used and always return to zero.

6.7.3 Interrupt Status Register

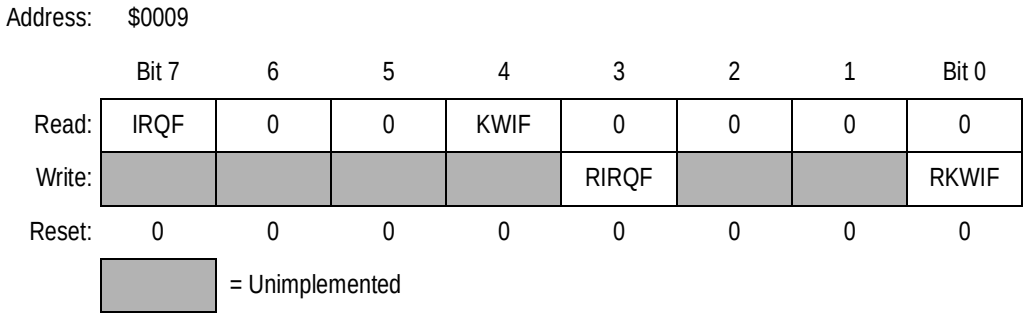


Figure 6-3. Interrupt Status Register (INSTR)

IRQF — External Interrupt (IRQ) Flag

A falling edge on the IRQ pin sets the IRQF bit. If the IRQE bit and this bit are set and the interrupt mask is cleared, an interrupt is generated. This is a read-only bit. Clearing IRQF is accomplished by writing a one to the RIRQF bit. Reset clears this bit.

Bits 6–5 — Reserved

These bits are not used and always read zero.

KWIF — Key Wakeup Interrupt Flag

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

RIRQF — Reset IRQ Flag

The RIRQF bit is a write-only bit and always read as zero. Writing a one to this bit clears the IRQF bit and writing zero to this bit has no effect.

1 = Clear IRQF
0 = No effect

Bits 2–1 — Reserved

These bits are not used and always read zero.

RKWIF — Reset KWI Flag

The RKWIF bit is a write-only bit and always read as zero. Writing a one to this bit clears the KWIF bit, and writing zero to this bit has no effect.

1 = Clear KWIF
0 = No effect

6.8 Serial Peripheral Interface Interrupt (SPII)

The SPII is generated by the serial peripheral interface system at the end of one byte of data transmission or reception. The I bit in the CCR must be clear and the SPIE bit of SPCR must be set for the SPII to be generated. This interrupt will vector to the interrupt service routine located at the address specified by the contents of memory locations $1FF2 and $1FF3. See **Section 10. Serial Peripheral Interface** for more information.

6.9 Event Counter Interrupt (EVI)

The EVI interrupt is generated by the event counter system. The I bit in the CCR must be clear for the EVI interrupt to be enabled. This interrupt will vector to the interrupt service routine located at the address specified by the contents of memory locations $1FF4 and $1FF5. See **Section 13. Event Counter** for more information.

6.10 Time Base Interrupt (TBI)

The TBI is generated periodically by the time base system. The I bit in the CCR must be clear for the TBI to be enabled. This interrupt will vector to the interrupt service routine located at the address specified by the contents of memory locations $1FF0:$1FF1. See **Section 9. Time Base** for more information.

6.11 Key Wakeup Interrupt (KWI)

The KWI interrupt is generated by the key wakeup system. The I bit in the CCR must be clear for the KWI interrupt to be enabled. This interrupt will vector to the interrupt service routine located at the address specified by the contents of memory locations $1FF8 and $1FF9. See **Section 7. Input/Output Ports (I/O)** for more information.

6.12 IRQ/KWI Software Consideration

IRQ and KWI interrupts have a timing delay in a case as shown in **Figure 6-5**. This section shows programming for proper interrupts with IRQ or KWI.

Figure 6-4 shows an example of a timer interrupt. In this case, the interrupt by TOF occurs as soon as TOIE (timer overflow interrupt enable) bit is set.

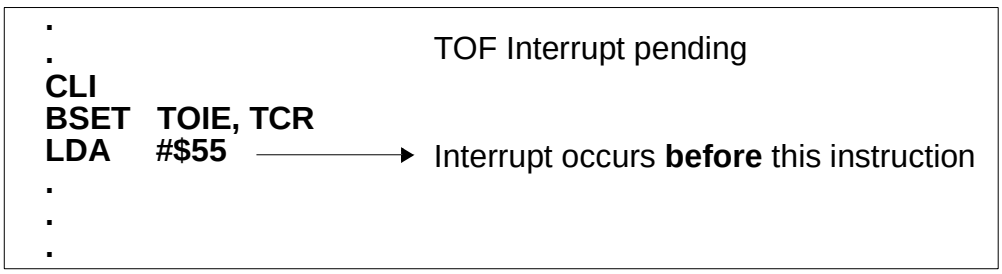


Figure 6-4. Timer Interrupt

Figure 6-5 shows an example of an IRQ interrupt. In this case, the interrupt occurs **after** execution of the instruction which sets the IRQE bit. A similar precaution is required for KWI interrupts.

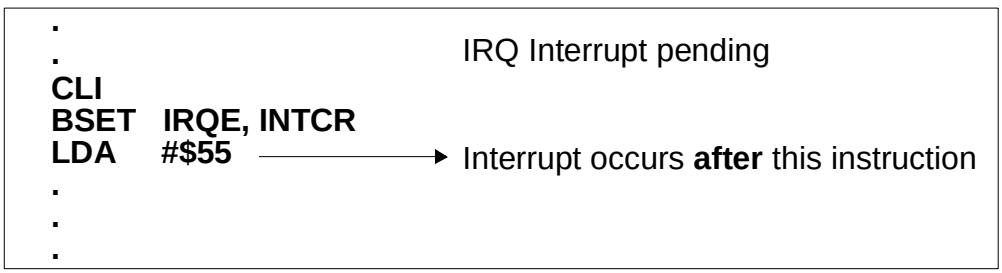


Figure 6-5. IRQ Timing Delay

This problem can be solved by using a software code as illustrated in **Figure 6-6**. A similar procedure should be used for KWI.

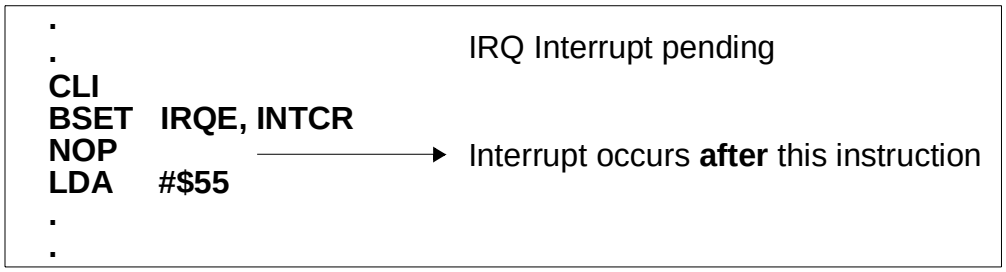


Figure 6-6. Software Patch for IRQ

Section 7. Input/Output Ports (I/O)

7.1 Contents

7.2 Introduction

In single-chip mode, 20 bidirectional input/output (I/O) lines are arranged as three ports: A, B, and C. Individual bits in these ports are programmable as either inputs or outputs under software control by the data direction registers (DDRs). If enabled by select bits in RCR or WOMR, port pins may have software programmable pullup resistors or open-drain outputs, respectively.

7.3 Port A

Port A is an 8-bit bidirectional port which shares bits 0–3 with the key wakeup subsystem, and bit 3 also is shared with the buzzer subsystem as shown in **Figure 7-1** and **Figure 7-2**. Bit 4 is shared with the analog-to-digital (A/D) converter and the event counter subsystems as shown in **Figure 7-3**. Bit 5 is shared with the A/D converter subsystem as shown in **Figure 7-4**. Bit 6 is shared with the time base subsystem as shown in **Figure 7-5**. Each port A pin is controlled by the corresponding bits in a data direction register and data register enable bits of appropriate subsystems. The port A data register is located at address $0000. The port A data direction register (DDRA) is located at address $0000 of the option map. Reset clears the DDRA. The port A data register is unaffected by reset.

Port A bits 0–3, when configured as an output port, is an open-drain output. Each pin can sink a maximum of 20 mA at V_{DD} = 5.0 V and V_{OL} (max) = 0.8 V. See **Section 15. Electrical Specifications**.

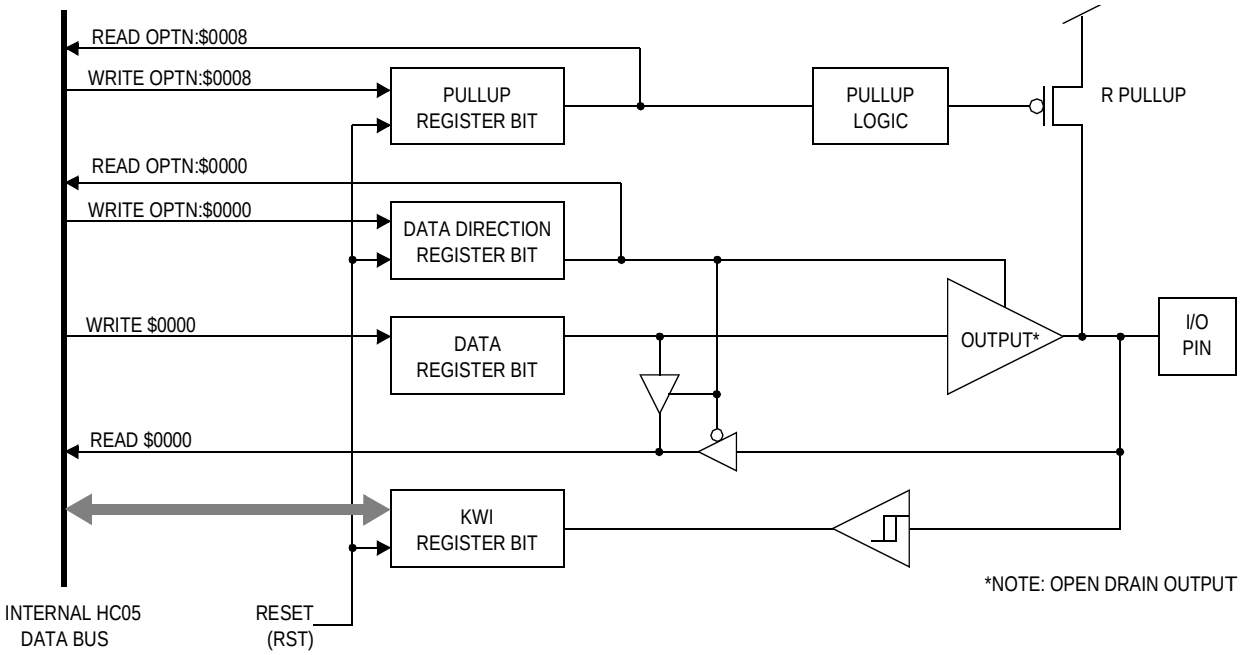


Figure 7-1. Port A0:A2/KWI0:KWI2 I/O Circuitry

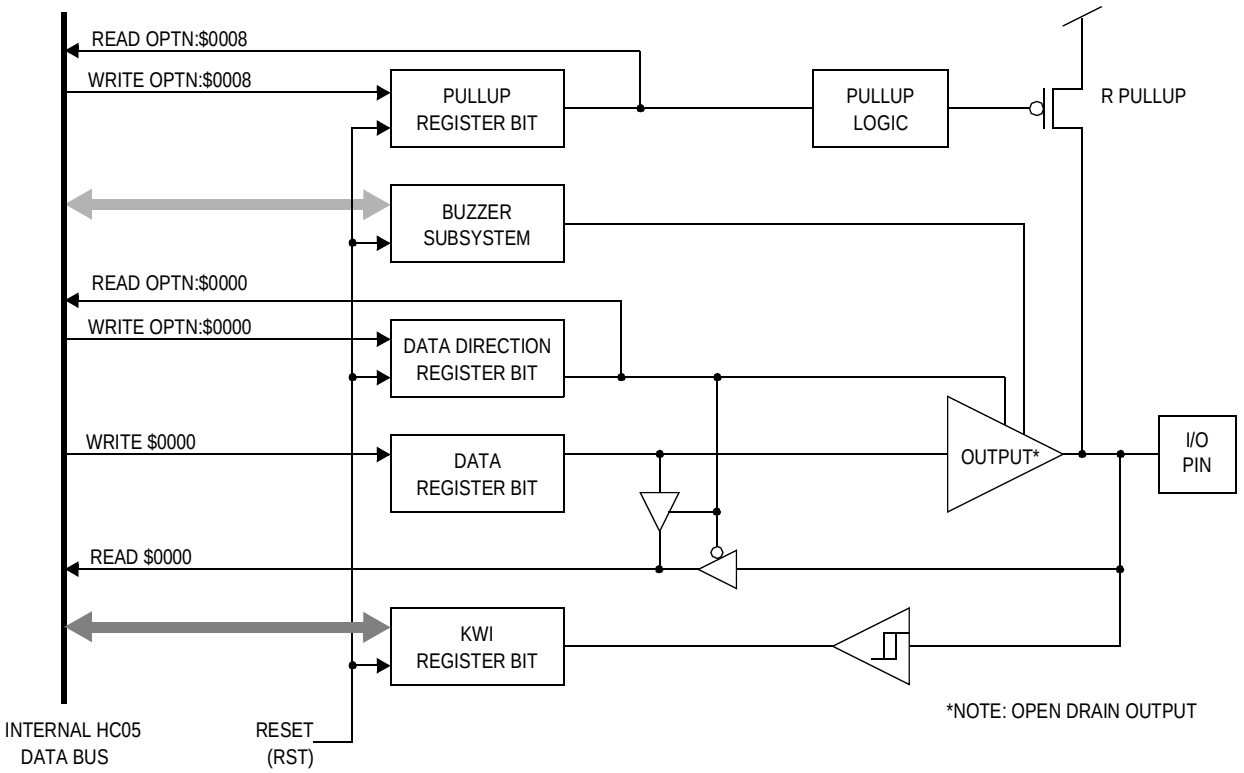


Figure 7-2. Port A3/KWI3/BZ I/O Circuitry

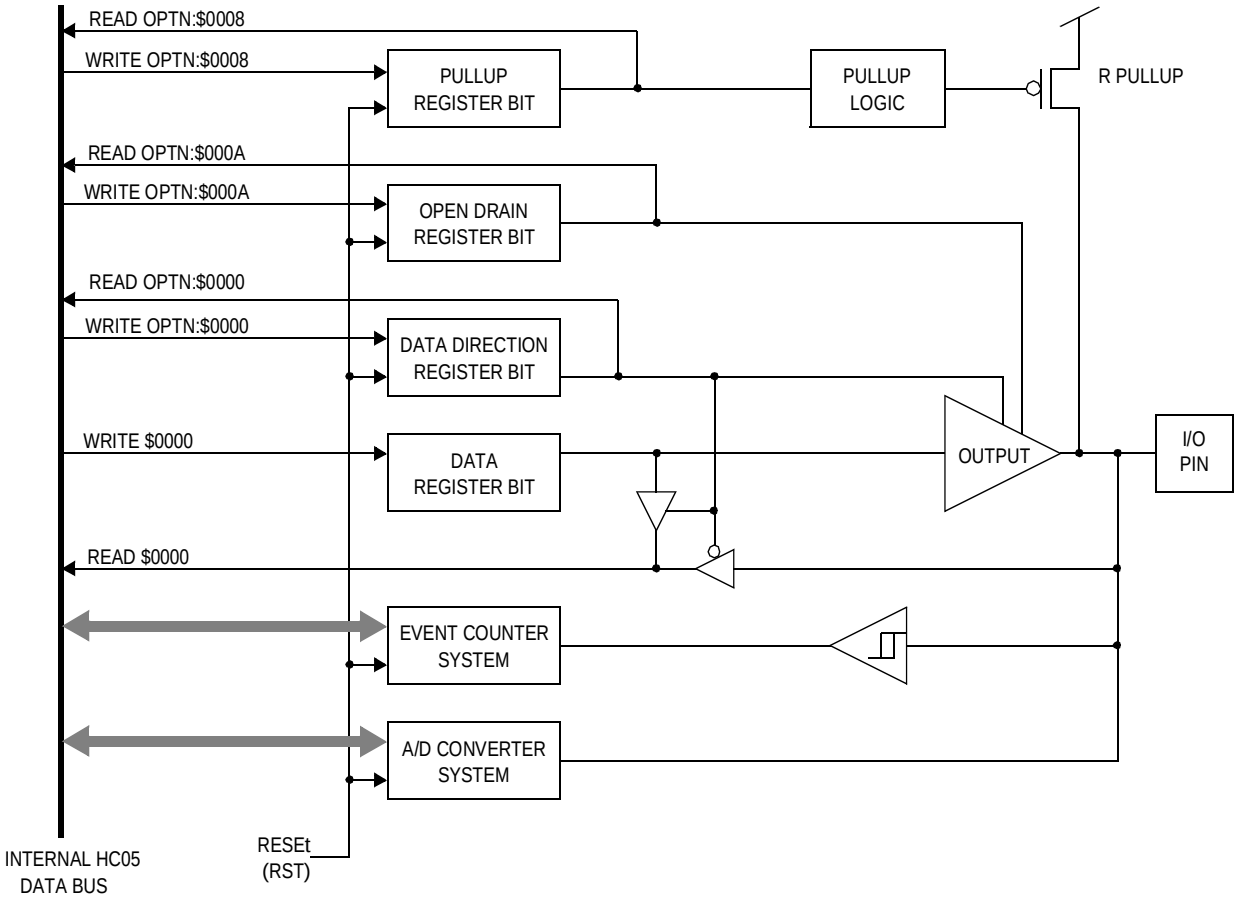


Figure 7-3. Port A4/AD0/EVI I/O Circuitry

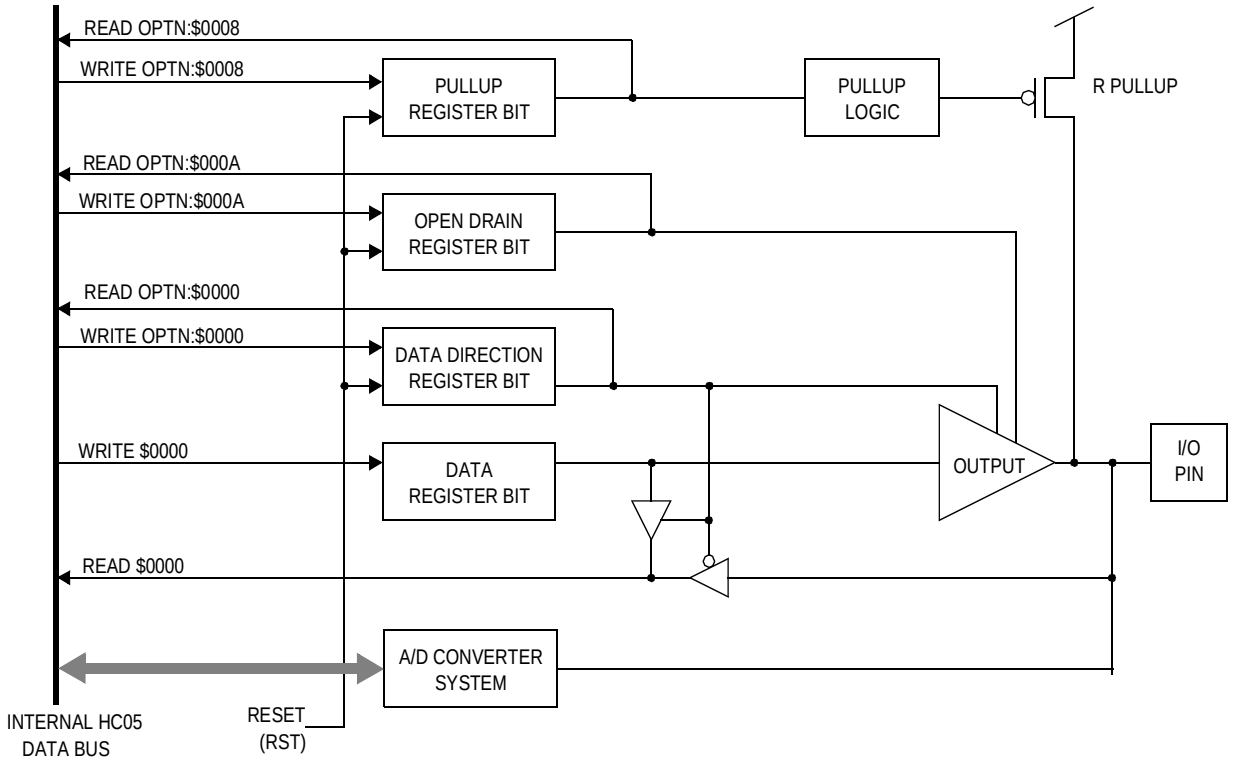


Figure 7-4. Port A5/AD1 I/O Circuitry

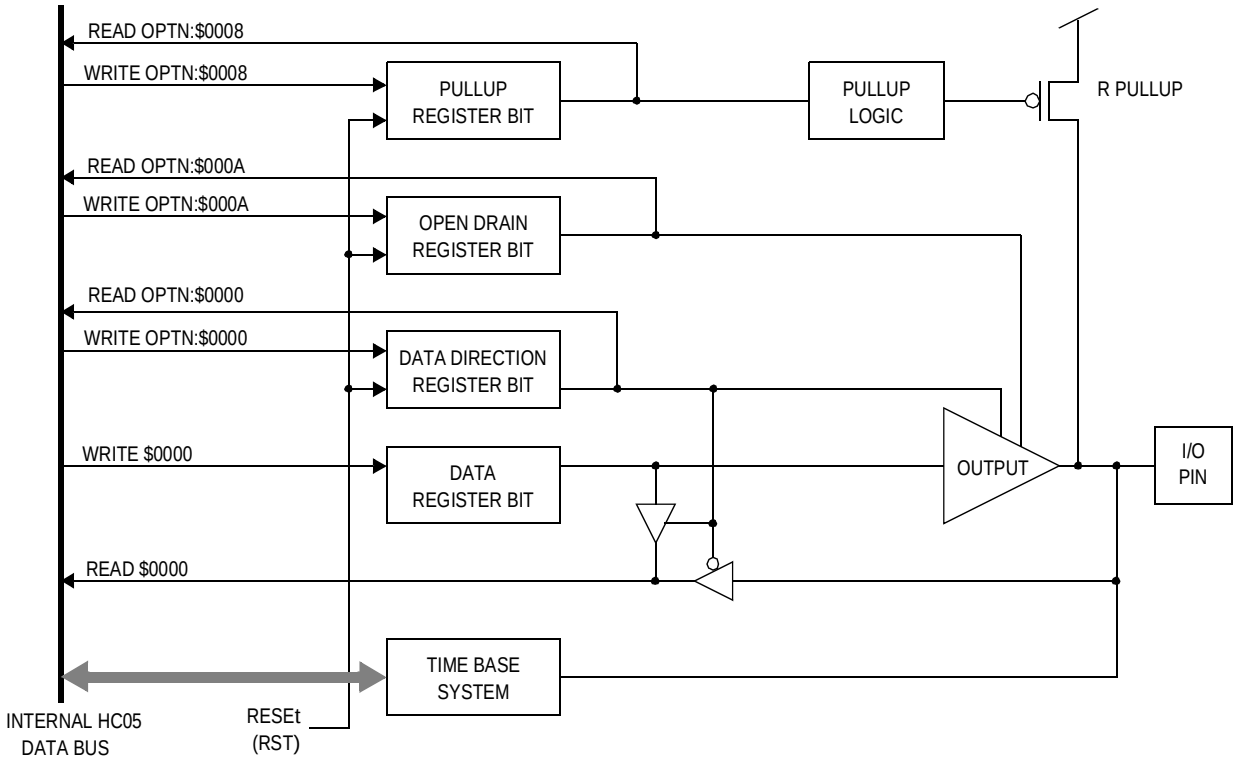


Figure 7-5. Port A6/RMO I/O Circuitry

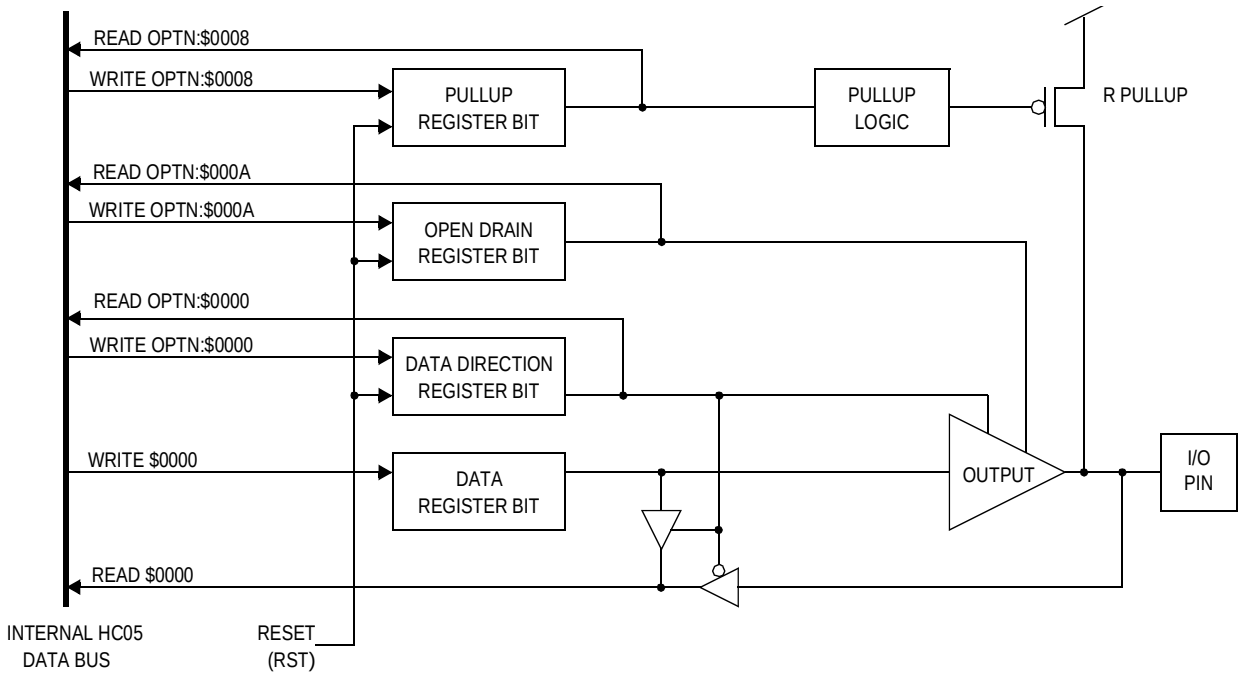


Figure 7-6. Port A7 I/O Circuitry

7.3.1 Port A Data Register

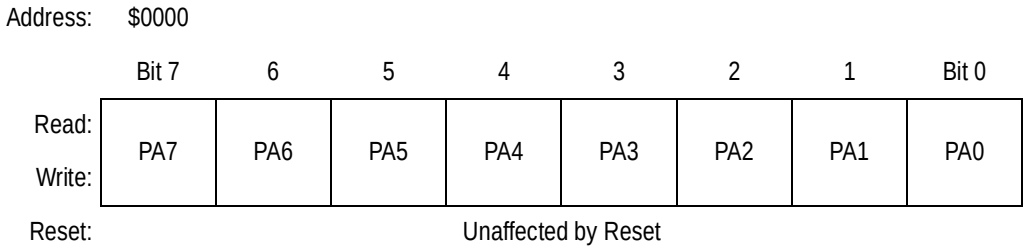


Figure 7-7. Port A Data Register

Each port A I/O pin has a corresponding bit in the port A data register. When a port A pin is programmed as an output, the state of the corresponding data register bit determines the state of the output pin. When a port A pin is programmed as an input, any read of the port A data register will return the logic state of the corresponding I/O pin. The port A data register is unaffected by reset.

7.3.2 Port A Data Direction Register

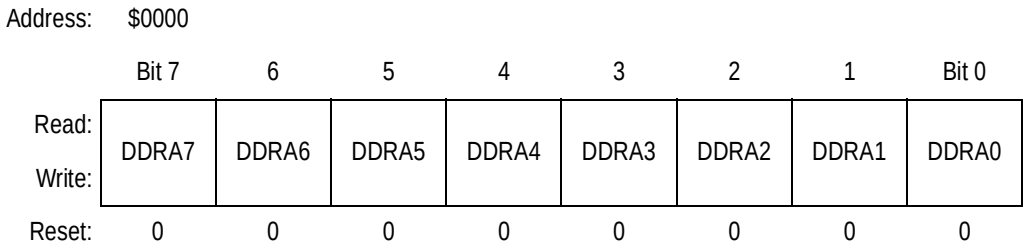


Figure 7-8. Port A Data Direction Register

Each port A I/O pin may be programmed as an input by clearing the corresponding bit in the DDRA or may be programmed as an output by setting the corresponding bit in the DDRA. The DDRA can be accessed at address $0000 of the option map. The DDRA is cleared by reset.

7.3.3 Port A Pullup Register

Each port A pin may have a software programmable pullup device enabled by the RCR select bits RAH and RAL. The pullup is activated whenever the corresponding bit in the RCR is set. Since reset clears the RCR, all pins will initialize with the pullup devices disabled. See **7.6.6 Resistor Control Register 1**.

7.3.4 Port A Wired-OR Mode Register

Port A bits 0:3 configured for output pins are wired-OR mode (open drain) only. Port A bits 4:7 configured for output pins may have software programmable wired-OR mode (open drain) output enabled by the AWOM bit in the WOMR. Since reset clears the WOMR, the wired-OR mode is disabled on reset. See **7.6.8 Open Drain Output Control Register**.

7.3.5 Key Wakeup Interrupt (KWI)

Four key wakeup inputs ($\overline{\text{KWI0}}$:$\overline{\text{KWI3}}$) share pins with port A. Each key wakeup input is enabled by the corresponding bit in the KWIEN register which resides in the option map. KWI is enabled by the KWIE bit in the INTCR.

When a falling edge is detected at one of the enabled key wakeup 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. All input latches are cleared at the same time by clearing the KWIF bit. See **Figure 7-9**.

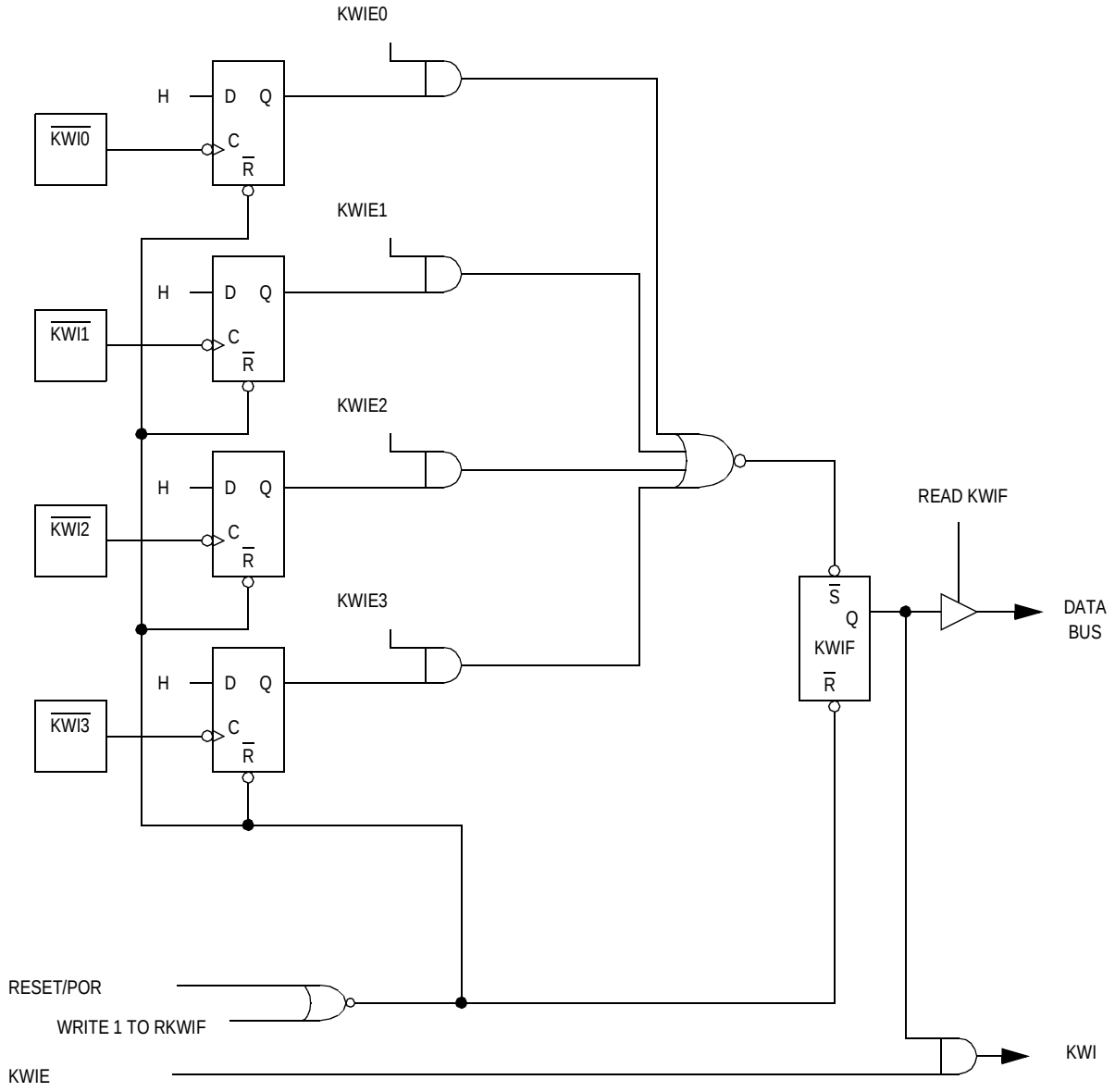


Figure 7-9. Key Wakeup Interrupt (KWI)

7.3.6 I/O Pin Truth Tables

Table 7-1 through Table 7-6 summarize the input or output mode programming for port A.

Table 7-1. PA0–PA2/$\overline{KWI0}$–$\overline{KWI2}$ I/O Pin Functions

DDR	Output Latch	KWIE0–KWIE2	RAL	I/O Pin Modes	Access to DDRA0–DDRA2	Access to Data Register Latch PA0–PA2	
					Read/Write	Read	Write
0	X	0	0	IN, Hi-Z, KWI Disable	DDRA0–DDRA2	Pin	Latch[2]
0	X	0	1	IN, KWI Disable, Pullup	DDRA0–DDRA2	Pin	Latch[2]
0	X	1	0	IN, Hi-Z, KWI Enable	DDRA0–DDRA2	Pin	Latch[2]
0	X	1	1	IN, KWI Enable, Pullup	DDRA0–DDRA2	Pin	Latch[2]
1	0	0	X	OUT, OD	DDRA0–DDRA2	Latch	Latch, Pin
1	1	0	0	OUT, OD, Hi-Z	DDRA0–DDRA2	Latch	Latch, Pin
1	1	0	1	OUT, OD, Pullup	DDRA0–DDRA2	Latch	Latch, Pin
1	0	1	X	OUT, OD, KWI Enable	DDRA0–DDRA2	Latch	Latch, Pin
1	1	1	0	OUT, OD, Hi-Z, KWI Enable	DDRA0–DDRA2	Latch	Latch, Pin
1	1	1	1	OUT, OD, Pullup, KWI Enable	DDRA0–DDRA2	Latch	Latch, Pin

NOTES:
1. X is don't care state.
2. Does not affect input, but stored to data register latch

Table 7-2. PA3/$\overline{\text{KWI3}}$/BZ I/O Pin Functions

DDR	Output Latch	KWIE3	BZPE	RAL	I/O Pin Modes	Access to DDRA3	Access to Data Register Latch PA3	
						Read/Write	Read	Write
0	X	0	0	0	IN, Hi-Z, KWI Disable	DDRA3	Pin	Latch[2]
0	X	0	0	1	IN, Hi-Z, KWI Disable, Pullup	DDRA3	Pin	Latch[2]
0	X	1	0	0	IN, Hi-Z, KWI Enable	DDRA3	Pin	Latch[2]
0	X	1	0	1	IN, Hi-Z, KWI Enable, Pullup	DDRA3	Pin	Latch[2]
1	0	0	0	X	Port OUT, OD, KWI Disable	DDRA3	Latch	Latch, Pin
1	1	0	0	0	Port OUT, OD, Hi-Z, KWI Disable	DDRA3	Latch	Latch, Pin
1	1	0	0	1	Port OUT, OD, KWI Disable, Pullup	DDRA3	Latch	Latch, Pin
1	0	1	0	X	Port OUT, OD, KWI Enable	DDRA3	Latch	Latch, Pin
1	1	1	0	0	Port OUT, OD, Hi-Z, KWI Enable	DDRA3	Latch	Latch, Pin
1	1	1	0	1	Port OUT, OD, Pullup, KWI Enable	DDRA3	Latch	Latch, Pin
0	X	0	1	0	BZ OUT, OD, KWI Disable	DDRA3	Pin	Latch[2]
1							Latch	
0	X	0	1	1	BZ OUT, OD, KWI Disable, Pullup	DDRA3	Pin	Latch[2]
1							Latch	
0	X	1	1	0	BZ OUT, OD, KWI Enable	DDRA3	Pin	Latch[2]
1							Latch	
0	X	1	1	1	BZ OUT, OD, KWI Enable, Pullup	DDRA3	Pin	Latch[2]
1							Latch	

NOTES:

1. X is don't care state.
2. Does not affect input, but stored to data register latch

Table 7-3. PA4/AD0/EVI I/O Pin Functions

DDR	Output Latch	EVCE	ADON	A/D CH	AWOM	RAH	I/O Pin Modes	Access to DDRA4	Access to Data Register Latch PA4	
								Read/Write	Read	Write
0	X	0	0	X	X	0	Port IN, Hi-Z	DDRA4	Pin	Latch[2]
0	X	0	0	X	X	1	Port IN, Pullup	DDRA4	Pin	Latch[2]
0	X	X	1	0	X	X	A/D IN	DDRA4	0	Latch[2]
0	X	X	1	1-7	X	0	Port IN, Hi-Z	DDRA4	Pin	Latch[2]
0	X	X	1	1-7	X	1	Port IN, Pullup	DDRA4	Pin	Latch[2]
0	X	1	X	X	X	0	Port IN, Hi-Z, Event In	DDRA4	Pin	Latch[2]
0	X	1	X	X	X	1	Port IN, Pullup, Event In	DDRA4	Pin	Latch[2]
1	X	0	0	X	0	X	OUT, CMOS	DDRA4	Latch	Latch, Pin
1	0	0	0	X	1	X	OUT, OD	DDRA4	Latch	Latch, Pin
1	1	0	0	X	1	0	OUT, OD, Hi-Z	DDRA4	Latch	Latch, Pin
1	1	0	0	X	1	1	OUT, OD, Pullup	DDRA4	Latch	Latch, Pin
1	X	0	1	X	0	X	OUT, CMOS	DDRA4	Latch	Latch, Pin
1	0	0	1	X	1	X	OUT, OD	DDRA4	Latch	Latch, Pin
1	1	X	1	0	1	0	OUT, OD	DDRA4	Latch	Latch, Pin
1	1	0	1	1-7	1	0	OUT, OD, Hi-Z	DDRA4	Latch	Latch, Pin
1	1	0	1	X	1	1	OUT, OD, Pullup	DDRA4	Latch	Latch, Pin
1	X	1	0	X	0	X	OUT, CMOS, Event In	DDRA4	Latch	Latch, Pin
1	0	1	0	X	1	X	OUT, OD, Event In	DDRA4	Latch	Latch, Pin
1	1	1	0	X	1	0	OUT, OD, Hi-Z, Event In	DDRA4	Latch	Latch, Pin
1	1	1	0	X	1	1	OUT, OD, Event In, Pullup	DDRA4	Latch	Latch, Pin
1	X	1	1	1-7	0	X	OUT, CMOS, Event In	DDRA4	Latch	Latch, Pin
1	0	1	1	1-7	1	X	OUT, OD, Event In	DDRA4	Latch	Latch, Pin
1	1	1	1	1-7	1	0	OUT, OD, Hi-Z, Event In	DDRA4	Latch	Latch, Pin
1	1	1	1	1-7	1	1	OUT, OD, Event In, Pullup	DDRA4	Latch	Latch, Pin

NOTES:

1. X is don't care state.
2. Does not affect input, but stored to data register latch
3. Do not apply analog voltage to this pin unless the I/O pin mode is set to A/D IN. Excessive current may be drawn if this pin is read as a digital input port while analog voltage is applied.

Table 7-4. PA5/AD1 I/O Pin Functions

DDR	Output Latch	ADON	A/D CH	AWOM	RAH	I/O Pin Modes	Access to DDRA5	Access to Data Register Latch PA5	
							Read/Write	Read	Write
0	X	0	X	X	0	Port IN, Hi-Z	DDRA5	Pin	Latch[2]
0	X	0	X	X	1	Port IN, Pullup	DDRA5	Pin	Latch[2]
0	X	1	1	X	X	A/D IN	DDRA5	0	Latch[2]
0	X	1	0, 2-7	X	0	Port IN, Hi-Z	DDRA5	Pin	Latch[2]
0	X	1	0, 2-7	X	1	Port IN, Pullup	DDRA5	Pin	Latch[2]
1	X	0	X	0	X	OUT, CMOS	DDRA5	Latch	Latch, Pin
1	0	0	X	1	X	OUT, OD	DDRA5	Latch	Latch, Pin
1	1	0	X	1	0	OUT, OD, Hi-Z	DDRA5	Latch	Latch, Pin
1	1	0	X	1	1	OUT, OD, Pullup	DDRA5	Latch	Latch, Pin
1	X	1	X	0	X	OUT, CMOS	DDRA5	Latch	Latch, Pin
1	0	1	X	1	X	OUT, OD	DDRA5	Latch	Latch, Pin
1	1	1	X	1	0	OUT, OD, Hi-Z	DDRA5	Latch	Latch, Pin
1	1	1	X	1	1	OUT, OD, Pullup	DDRA5	Latch	Latch, Pin

NOTES:

1. X is don't care state.
2. Does not affect input, but stored to data register latch
3. Do not apply analog voltage to this pin unless the I/O pin mode is set to A/D IN. Excessive current may be drawn if this pin is read as a digital input port while analog voltage is applied.

Table 7-5. PA6/RMO I/O Pin Functions

I/O Port		Remote Carrier Output			AWOM	RAH	I/O Pin Modes	Access to DDRA6	Access to Data Register Latch PA6	
DDR	OL	RMPE	RMON	RPOL				Read/Write	Read	Write
0	X	X	X	X	X	0	Port IN, Hi-Z	DDRA6	Pin	Latch[3]
0	X	X	X	X	X	1	Port IN, Pullup	DDRA6	Pin	Latch[3]
1	X	0	X	X	0	X	OUT, Output Latch, CMOS	DDRA6	Latch	Latch, Pin
1	0	0	X	X	1	X	OUT, Output Latch, OD	DDRA6	Latch	Latch, Pin
1	1	0	X	X	1	0	OUT, Output Latch, OD, Hi-Z	DDRA6	Latch	Latch, Pin
1	1	0	X	X	1	1	OUT, Output Latch, OD, Pullup	DDRA6	Latch	Latch, Pin
1	X	1	0	0	0	X	OUT, Remote Idle, CMOS = V_{SS}	DDRA6	Latch	Latch[3]
1	X	1	0	1	0	X	OUT, Remote Idle, CMOS = V_{DD}	DDRA6	Latch	Latch[3]
1	X	1	1	X	0	X	OUT, Remote Carrier, CMOS	DDRA6	Latch	Latch[3]
1	X	1	1	X	1	0	OUT, Remote Carrier, OD/Hi-Z	DDRA6	Latch	Latch[3]
1	X	1	1	X	1	1	OUT, Remote Carrier, OD/Pullup	DDRA6	Latch	Latch[3]

NOTES:

1. OL = output latch
2. X is don't care state.
3. Does not affect input, but stored to data register latch
4. V_{SS}/Hi-Z = output is either V_{SS} (N-ch on) or Hi-Z (N- and P-ch off) depending on clock pulse
5. V_{SS}/Pullup = output is either V_{SS} (N-ch on) or pullup (resistive) depending on clock pulse

Table 7-6. PA7 I/O Pin Functions

DDR	Output Latch	AWOMH	RAH	I/O Pin Modes	Access to DDRA7	Access to Data Register Latch PA7	
					Read/Write	Read	Write
0	X	X	0	Port IN, Hi-Z	DDRA7	Pin	Latch[2]
0	X	X	1	Port IN, Pullup	DDRA7	Pin	Latch[2]
1	X	0	X	OUT, CMOS	DDRA7	Latch	Latch, Pin
1	0	1	X	OUT, OD	DDRA7	Latch	Latch, Pin
1	1	1	0	OUT, OD, Hi-Z	DDRA7	Latch	Latch, Pin
1	1	1	1	OUT, OD, Pullup	DDRA7	Latch	Latch, Pin

NOTES:

1. X is don't care state.
2. Does not affect input, but is stored to the data register latch

7.4 Port B

Port B is an 8-bit bidirectional port that is shared with LCD frontplane drivers as shown in **Figure 7-10**. Each port B pin is controlled by the corresponding bits in a data direction register and a data register. The port B data register is located at address $0001. The port B data direction register (DDRB) is located at address $0001 of the option map. Reset clears the DDRB. The port B data register is unaffected by reset. The LCD frontplane drivers are enabled on reset.

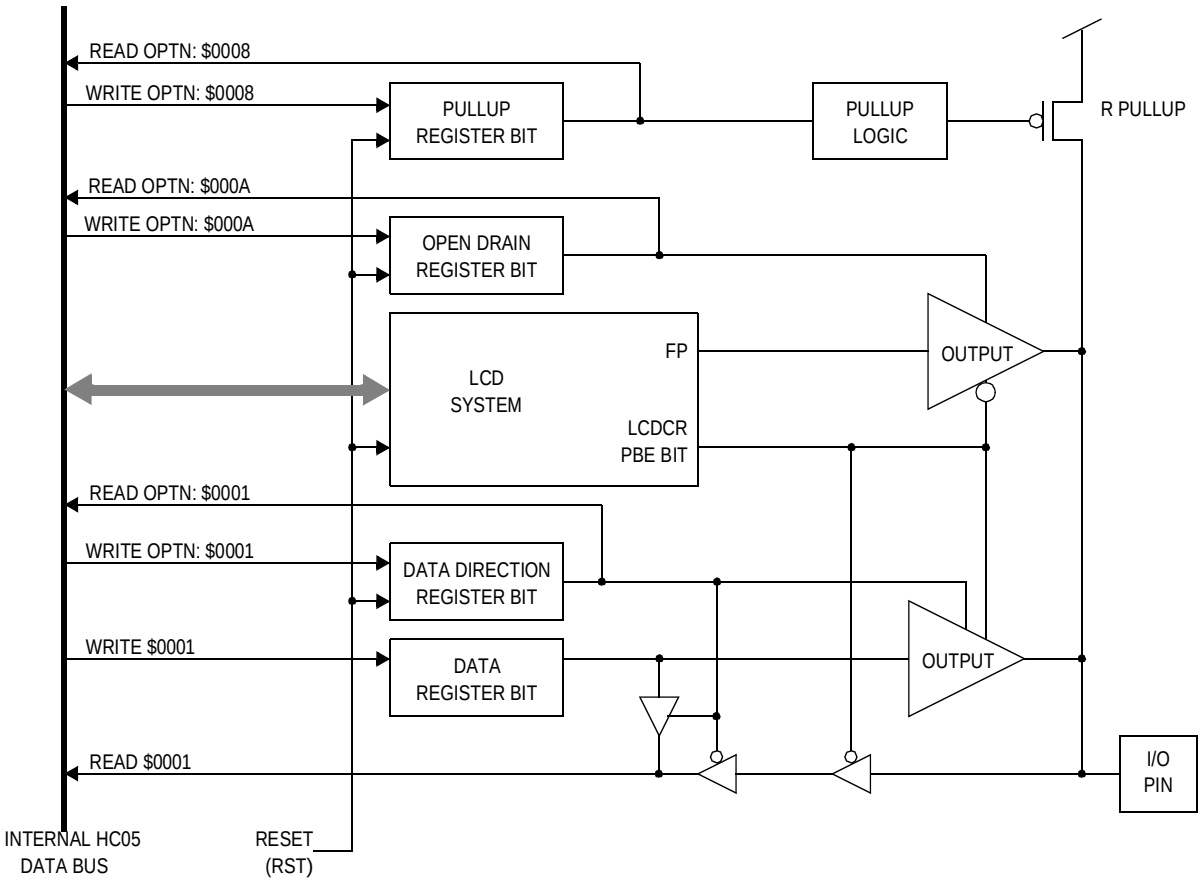


Figure 7-10. Port B0:B7/FP24:FP17 I/O Circuitry

7.4.1 Port B Data Register

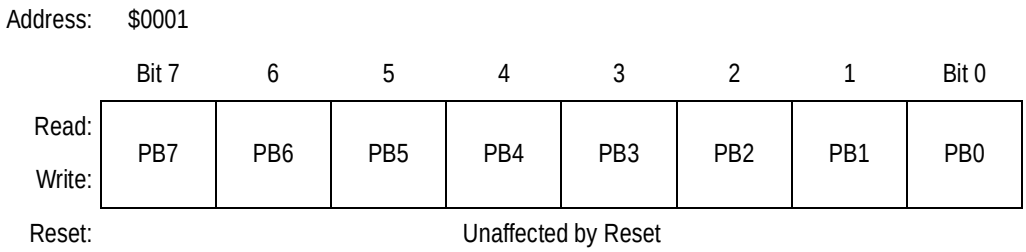


Figure 7-11. Port B Data Register (PORTB)

Each port B I/O pin has a corresponding bit in the port B data register. When a port B pin is programmed as an output, the state of the corresponding data register bit determines the state of the output pin. When a port B pin is programmed as an input, any read of the port B data register will return the logic state of the corresponding I/O pin. The port B data register is unaffected by reset.

7.4.2 Port B Data Direction Register

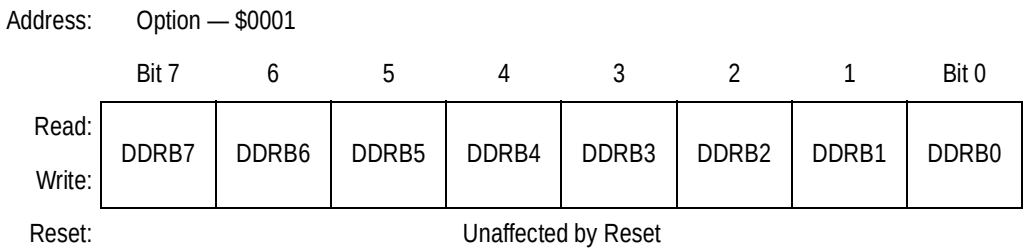


Figure 7-12. Port B Data Direction Register (DDRB)

Each port B I/O pin can be programmed as an input by clearing the corresponding bit in the DDRB or programmed as an output by setting the corresponding bit in the DDRB. The DDRB can be accessed at address $0001 of the option map. The DDRB is cleared by reset.

7.4.3 Port B Pullup Register

Each port B pin may have a software programmable pullup device enabled by the RCR select bits RBH and RBL. The pullup is activated whenever the corresponding bit in the RCR is set. Since reset clears the RCR, all pins will initialize with the pullup devices disabled. See **7.6.6 Resistor Control Register 1**.

NOTE: *Do not turn on port B pullups when LCD is selected for corresponding port pin.*

7.4.4 Port B Wire-ORed Mode Register

Port B bits 0–7 configured for output pins may have software programmable wired-OR mode (open drain) output enabled by the BWOMH and BWOML bits in the WOMR. Since reset clears the WOMR, the wired-OR mode becomes disabled on reset. See **7.6.8 Open Drain Output Control Register**.

7.4.5 I/O Pin Truth Tables

Table 7-7 and **Table 7-8** summarize the input or output and LCD mode programming for port B.

Table 7-7. PB0–PB3/FP24–FP21 I/O Pin Functions

DDR	Output Latch	LCDCR PBEL	BWOML	RBL	I/O Pin Modes	Access to DDRB0–DDRB3	Access to Data Register Latch PB0–PB3	
						Read/Write	Read	Write
X	X	0	X	0	LCD FP Output	DDRB0–DDRB3	0	Latch[2]
X	X	0	X	1	LCD FP Output, Pullup[3]	DDRB0–DDRB3	0	Latch[2]
0	X	1	X	0	Port IN, Hi-Z	DDRB0–DDRB3	Pin	Latch[2]
0	X	1	X	1	Port IN, Pullup	DDRB0–DDRB3	Pin	Latch[2]
1	X	1	0	X	OUT, CMOS	DDRB0–DDRB3	Latch	Latch, Pin
1	0	1	1	X	OUT, OD	DDRB0–DDRB3	Latch	Latch, Pin
1	1	1	1	0	OUT, OD, Hi-Z	DDRB0–DDRB3	Latch	Latch, Pin
1	1	1	1	1	OUT, OD, Pullup	DDRB0–DDRB3	Latch	Latch, Pin

NOTES:
1. X is don't care state.
2. Does not affect input, but is stored to data register latch
3. Do not turn on pullup R (RBL = 1) when using these pins as LCD ports.

Table 7-8. PB4–PB7/FP20–FP17 I/O Pin Functions

DDR	Output Latch	LCDCR PBEH	BWOMH	RBH	I/O Pin Modes	Access to DDRB4–DDRB7	Access to Data Register Latch PB4–PB7	
						Read/Write	Read	Write
X	X	0	X	0	LCD FP Output	DDRB4–DDRB7	0	Latch[2]
X	X	0	X	1	LCD FP Output, Pullup[3]	DDRB4–DDRB7	0	Latch[2]
0	X	1	X	0	Port IN, Hi-Z	DDRB4–DDRB7	Pin	Latch[2]
0	X	1	X	1	Port IN, Pullup	DDRB4–DDRB7	Pin	Latch[2]
1	X	1	0	X	OUT, CMOS	DDRB4–DDRB7	Latch	Latch, Pin
1	0	1	1	X	OUT, OD	DDRB4–DDRB7	Latch	Latch, Pin
1	1	1	1	0	OUT, OD, Hi-Z	DDRB4–DDRB7	Latch	Latch, Pin
1	1	1	1	1	OUT, OD, Pullup	DDRB4–DDRB7	Latch	Latch, Pin

NOTES:
1. X is don't care state.
2. Does not affect input, but is stored to data register latch
3. Do not turn on pullup R (RBH = 1) when using these pins as LCD ports.

7.5 Port C

Port C is a 4-bit I/O port which shares its pins with external interrupt $\overline{\text{IRQ}}$ and the serial peripheral interface (SPI) system as shown in **Figure 7-13** through **Figure 7-16**. Each port C pin is controlled by the corresponding bits in a wired-OR mode register and a pullup register. The port C data register is located at address $0002. The port C pullup register (RCR) is located at address $0009 of the option map. The wired-OR mode register (WOMR) is located at address $000A of the option map. Reset clears the RCR and the WOMR.

The PC0–PC2 pins are shared with the serial peripheral interface (SPI). When the SPI is enabled (SPE = 1), the pins PC0, PC1, and PC2 are configured as serial clock output or input (SCK), serial data output (SDO), and serial data input (SDI) pins, respectively. The direction of the SCK depends on the MSTR bit in the SPCR. When PORTC is read, the pin state is read. See **Table 7-9** through **Table 7-10**. The SCK pin should be at the V_{DD} level before the SPI is enabled.

The PC3 pin is shared with the external interrupt $\overline{\text{IRQ}}$ pin. The $\overline{\text{IRQ}}$ pin has a Schmitt trigger to improve noise immunity. The PC3 pin state can be read any time regardless of the IRQ configurations.

Port C bits 2 and 3, when configured as output ports, are open drain outputs.

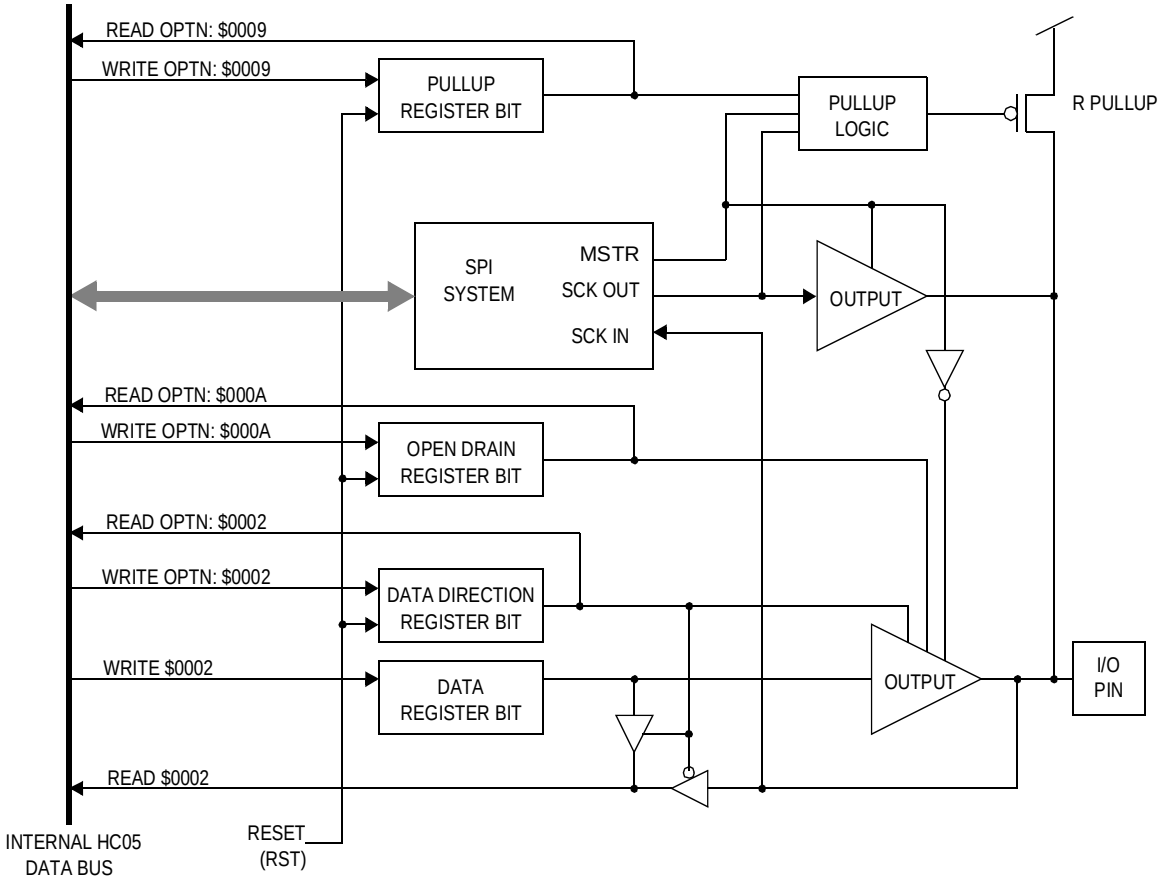


Figure 7-13. Port PC0/SCK Circuitry

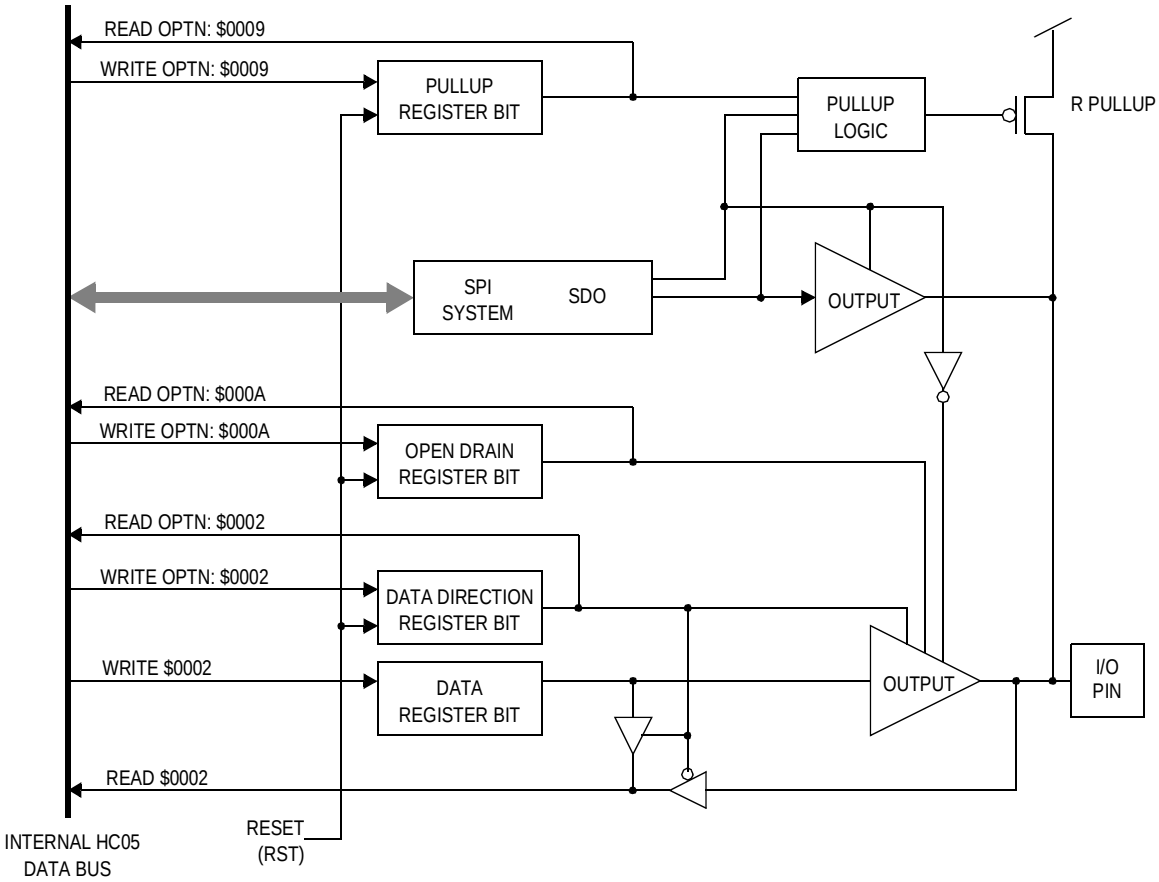


Figure 7-14. PC1/SDO Circuitry

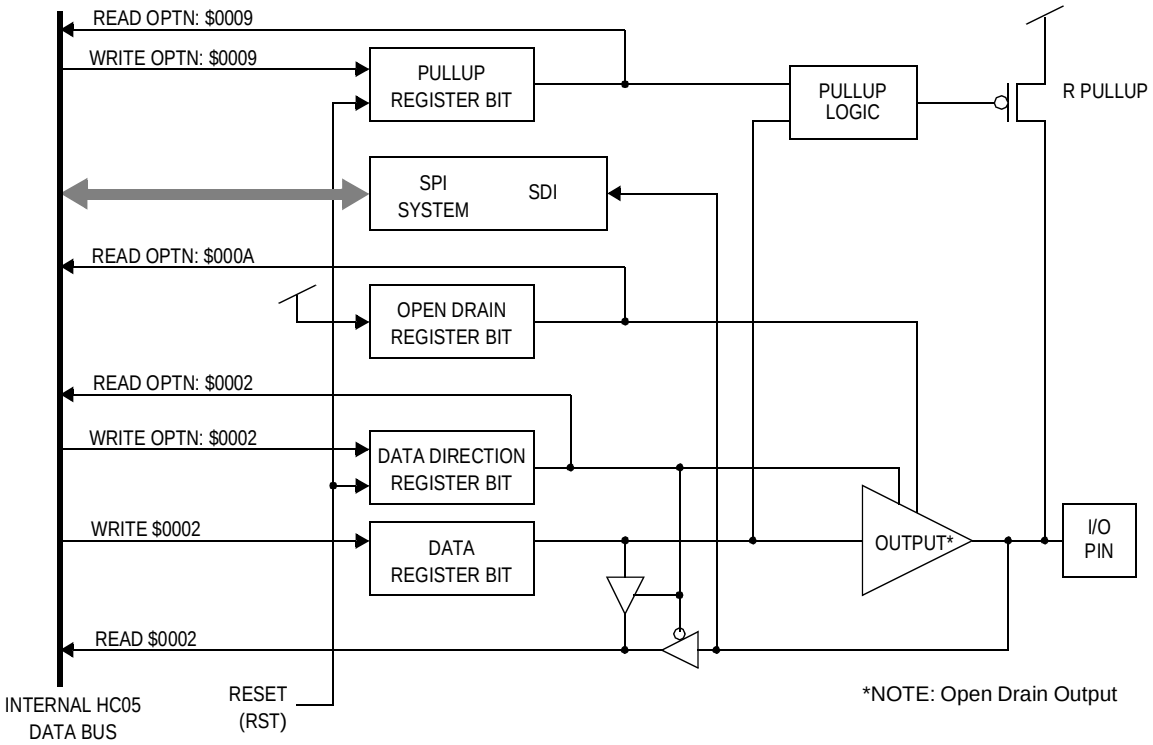


Figure 7-15. PC2/SDI Circuitry

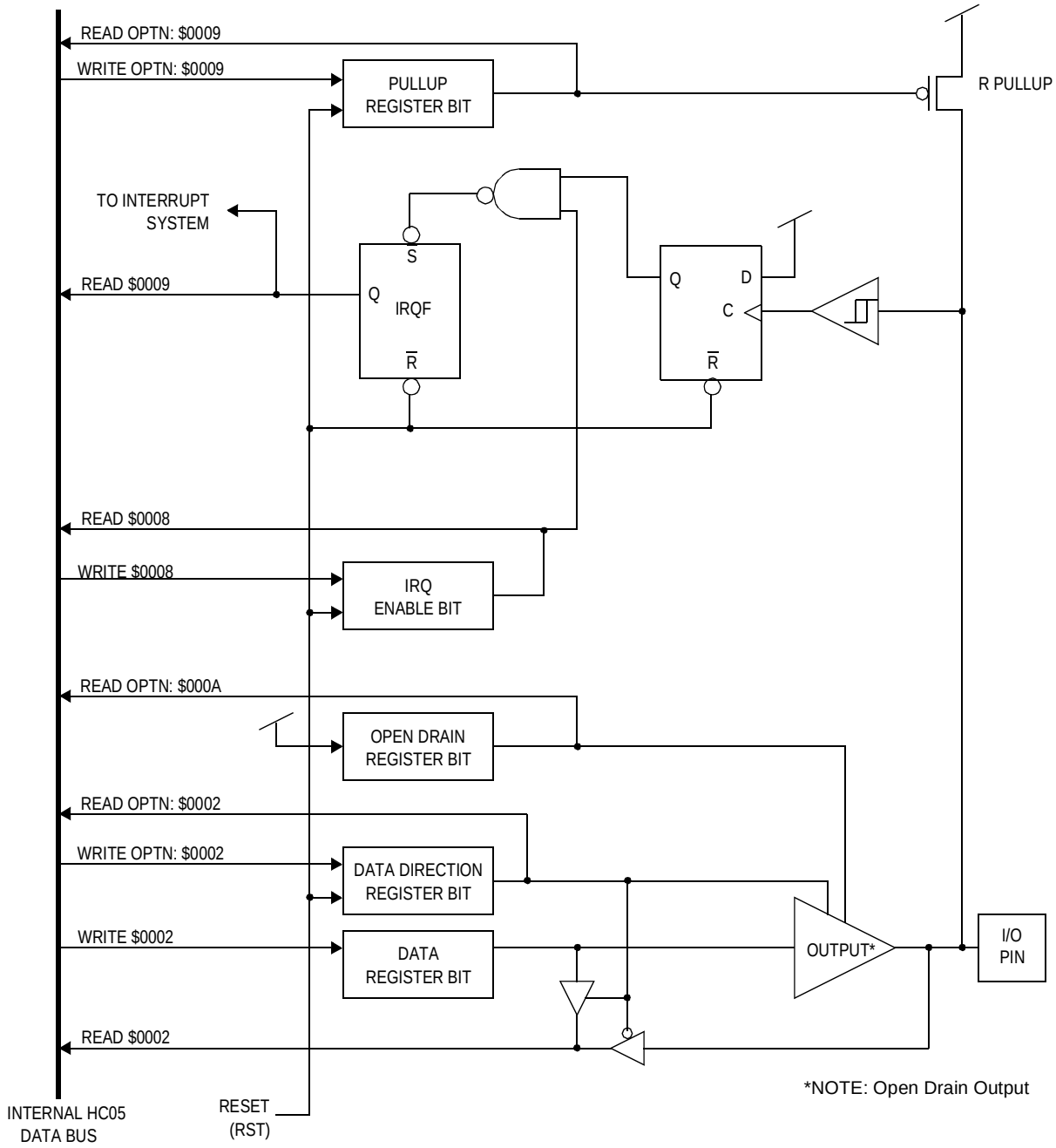


Figure 7-16. PC3/$\overline{\text{IRQ}}$ Circuitry

7.5.1 Port C Data Register

Each port C input pin has a corresponding bit in the port C data register. Regardless of the peripheral configuration, any read of the port C data register will return the logic state of the corresponding I/O pin. The port C data register is unaffected by reset.

Address: $0002

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	0	0	0	0	PC3	PC2	PC1	PC0
Write:								
Reset:	0	0	0	0	U	U	U	U

= Unimplemented U = Unaffected

Figure 7-17. Port C Data Register (PORTC)

7.5.2 Port C Data Direction Register

Each port C I/O pin can be programmed as an input by clearing the corresponding bit in the DDRC or programmed as an output by setting the corresponding bit in the DDRC. The DDRC can be accessed at address $0002 of the option map. The DDRC is cleared by reset.

Address: Option — $0002

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	0	0	0	0	DDRC3	DDRC2	DDRC1	DDRC0
Write:								
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 7-18. Port C Data Direction Register (DDRC)

7.5.3 Port C Pullup Register

Each port C pin can have a software programmable pullup device enabled by the RCR2 select bit RC. The pullup is activated whenever the RC bit in the RCR2 is set. Since reset clears the RCR2, all pins will initialize with the pullup devices disabled. See **7.6.7 Resistor Control Register 2**.

7.5.4 Port C Wired-OR Mode Register

Port C bits 0 and 1 are configured for output pins and can have software programmable wired-OR mode (open drain) output enabled by the CWOM bit in the WOMR. Since reset clears the WOMR, the wired-OR mode is disabled on reset. Port C bits 2 and 3, when configured as an output port, have wired-OR mode output only. See **7.6.8 Open Drain Output Control Register**.

7.5.5 I/O Pin Truth Tables

Table 7-9 through **Table 7-12** summarize the input, pullup, wired-OR mode, and SPI pin programming.

Table 7-9. PC0/SCK I/O Pin Functions

DDR	Output Latch	SPCR SPE	SPCR MSTR	WOMR CWOM	RCR2 RC Bit	I/O Pin Modes	Access to DDRC0	Access to Data Register Latch PC0	
							Read/Write	Read	Write
0	X	0	X	X	0	Port IN, Hi-Z	DDRC0	Pin	Latch[2]
0	X	0	X	X	1	Port IN, Pullup	DDRC0	Pin	Latch[2]
1	X	0	X	0	X	OUT, CMOS	DDRC0	Latch	Latch, Pin
1	0	0	X	1	X	OUT, OD	DDRC0	Latch	Latch, Pin
1	1	0	X	1	0	OUT, OD, Hi-Z	DDRC0	Latch	Latch, Pin
1	1	0	X	1	1	OUT, OD, Pullup	DDRC0	Latch	Latch, Pin
0	X	1	0	X	0	SCK IN, Hi-Z	DDRC0	Pin	Latch[2]
1	X	1	0	X	0	SCK IN, Hi-Z	DDRC0	Latch	Latch[2]
0	X	1	1	0	0	SCK OUT, CMOS, Hi-Z	DDRC0	Pin	Latch[2]
0	X	1	1	0	1	SCK OUT, CMOS, Pullup	DDRC0	Pin	Latch[2]
1	X	1	1	0	0	SCK OUT, CMOS, Hi-Z	DDRC0	Latch	Latch[2]
1	X	1	1	0	1	SCK OUT, CMOS, Pullup	DDRC0	Latch	Latch[2]
0	X	1	1	1	0	SCK OUT, OD, Hi-Z	DDRC0	Pin	Latch[2]
0	X	1	1	1	1	SCK OUT, OD, Pullup	DDRC0	Pin	Latch[2]
1	X	1	1	1	0	SCK OUT, OD, Hi-Z	DDRC0	Latch	Latch[2]
1	X	1	1	1	1	SCK OUT, OD, Pullup	DDRC0	Latch	Latch[2]

NOTES:
1. X is don't care state.
2. Does not affect input, but stored to data register latch

Table 7-10. PC1/SDO I/O Pin Functions

DDR	Output Latch	SPCR SPE	WOMR CWOM	RCR2 RC Bit	I/O Pin Modes	Access to DDRC1	Access to Data Register Latch PC1	
						Read/Write	Read	Write
0	X	0	X	0	Port IN, Hi-Z	DDRC1	Pin	Latch[2]
0	X	0	X	1	Port IN, Pullup	DDRC1	Pin	Latch[2]
1	X	0	0	X	Port OUT, CMOS	DDRC1	Latch	Latch, Pin
1	0	0	1	X	Port OUT, OD	DDRC1	Latch	Latch, Pin
1	1	0	1	0	Port OUT, OD, Hi-Z	DDRC1	Latch	Latch, Pin
1	1	0	1	1	Port OUT, OD, Pullup	DDRC1	Latch	Latch, Pin
0	X	1	0	0	SDO OUT, CMOS, Hi-Z	DDRC1	Pin	Latch[2]
0	X	1	0	1	SDO OUT, CMOS, Pullup	DDRC1	Pin	Latch[2]
1	X	1	0	0	SDO OUT, CMOS, Hi-Z	DDRC1	Latch	Latch[2]
1	X	1	0	1	SDO OUT, CMOS, Pullup	DDRC1	Latch	Latch[2]
0	X	1	1	0	SDO OUT, OD, Hi-Z	DDRC1	Pin	Latch[2]
0	X	1	1	1	SDO OUT, OD, Pullup	DDRC1	Pin	Latch[2]
1	X	1	1	0	SDO OUT, OD, Hi-Z	DDRC1	Latch	Latch[2]
1	X	1	1	1	SDO OUT, OD, Pullup	DDRC1	Latch	Latch[2]

NOTES:
1. X is don't care state.
2. Does not affect input, but stored to data register latch

Table 7-11. PC2/SDI I/O Pin Functions

DDR	Output Latch	SPCR SPE	WOMR CWOM	RCR2 RC Bit	I/O Pin Modes	Access to DDRC2	Access to Data Register Latch PC2	
						Read/Write	Read	Write
0	X	0	X	0	Port IN, Hi-Z	DDRC2	Pin	Latch[2]
0	X	0	X	1	Port IN, Pullup	DDRC2	Pin	Latch[2]
1	0	0	X	X	Port OUT, OD	DDRC2	Latch	Latch, Pin
1	1	0	X	0	Port OUT, OD, Hi-Z	DDRC2	Latch	Latch, Pin
1	1	0	X	1	Port OUT, OD, Pullup	DDRC2	Latch	Latch, Pin
0	X	1	X	0	SDI IN, Hi-Z	DDRC2	Pin	Latch[2]
0	X	1	X	1	SDI IN, Pullup	DDRC2	Pin	Latch[2]
1	X	1	X	0	SDI IN, Hi-Z	DDRC2	Latch	Latch[2]
1	X	1	X	1	SDI IN, Pullup	DDRC2	Latch	Latch[2]

NOTES:
1. X is don't care state.
2. Does not affect input, but stored to data register latch

Table 7-12. PC3/$\overline{\text{IRQ}}$ I/O Pin Functions

DDR	Output Latch	INTCR IRQE	WOMR CWOM	RCR2 RC Bit	I/O Pin Modes	Access to DDRC3	Access to Data Register Latch PC3	
						Read/Write	Read	Write
0	X	0	X	0	Port IN, Hi-Z	DDRC3	Pin	Latch[2]
0	X	0	X	1	Port IN, Pullup	DDRC3	Pin	Latch[2]
0	X	1	X	0	Port IN, Hi-Z, IRQ	DDRC3	Pin	Latch[2]
0	X	1	X	1	Port IN, Pullup, IRQ	DDRC3	Pin	Latch[2]
1	0	0	X	X	Port OUT, OD	DDRC3	Latch	Latch, Pin
1	1	0	X	0	Port OUT, OD, Hi-Z	DDRC3	Latch	Latch, Pin
1	1	0	X	1	Port OUT, OD, Pullup	DDRC3	Latch	Latch, Pin
1	0	1	X	X	Port OUT, OD, IRQ	DDRC3	Latch	Latch, Pin
1	1	1	X	0	Port OUT, OD, Hi-Z, IRQ	DDRC3	Latch	Latch, Pin
1	1	1	X	1	Port OUT, OD, Pullup, IRQ	DDRC3	Latch	Latch, Pin

NOTES:
1. X is don't care state.
2. Does not affect input, but stored to data register latch

7.6 I/O Port Programming

All bidirectional I/O pins can be programmed as inputs or outputs.

7.6.1 Pin Data Direction

The direction of a pin is determined by the state of its corresponding bit in the associated port data direction register (DDR). A pin is configured as an output if its corresponding DDR bit is set to a logic 1. A pin is configured as an input if its corresponding DDR bit is cleared to a logic 0.

The data direction bits DDRA0:DDRA7, DDRB0:DDRB7, and DDRC0:DDRC3 are read/write bits which can be manipulated with read-modify-write instructions. At power-on or reset, all DDRs are cleared, which configures all I/O port pins as input (except port B is configured as an LCD port).

7.6.2 Output Pin

When an I/O pin is programmed as an output pin, the state of the corresponding data register bit will determine the state of the pin. The state of the data register bits can be altered by writing to address $0000 for port A, address $0001 for port B, and address $0002 for port C. Reads of the corresponding data register bit at address $0000 or $0003 will return the state of the data register bit (not the state of the I/O pin itself). Therefore, bit manipulation is possible on all pins programmed as outputs.

7.6.3 Input Pin

When an I/O pin is programmed as an input pin, or for an input-only pin, the state of the pin can be determined by reading the corresponding data register bit. Any writes to the corresponding data register bit for an input-only pin will be ignored.

If the corresponding bit in the pullup register is set, the input pin will have an activated pullup device. Since the pullup register bits are read-write, bit manipulation may be used on these register bits.

7.6.4 I/O Pin Transitions

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.

7.6.5 I/O Pins with Subsystems

An I/O pin that is shared with another subsystem is in general configured as an input pin during reset, except for LCD driver pins. The LCD driver output pins BP0:BP3 and FP0:FP24 are configured to output V_{DD} level during and after reset. See truth tables of each I/O port for more details.

Table 7-13. Port Control Register Bits Summary

Port	Bit	DDR	WOM	Pullup R	Module Control		Module
A	0	DDRA0	1	RAL	KWIE	KWIE0	KWI0
	1	DDRA1	1			KWIE1	KWI1
	2	DDRA2	1			KWIE2	KWI2
	3	DDRA3	1		BZPE, KWIE	BZxx, KWIE3	KWI3/BZ
	4	DDRA4	AWOM	RAH	ADON, EVCE	CH2:CH0/EVxx	AD0/EVI
	5	DDRA5			ADON	CH2:CH0	AD1
	6	DDRA6			RME	TBCLK, RMC4:RMC0	RMO
	7	DDRA7			—	—	—
B	0	DDRB0	BWOML	RBL	LCDE, PBEL	F24B3:F24B0	FP24
	1	DDRB1				F23B3:F23B0	FP23
	2	DDRB2				F22B3:F22B0	FP22
	3	DDRB3				F21B3:F21B0	FP21
	4	DDRB4	BWOMH	RBH	LCDE, PBEH	F20B3:F20B0	FP20
	5	DDRB5				F19B3:F19B0	FP19
	6	DDRB6				F18B3:F18B0	FP18
	7	DDRB7				F17B3:F17B0	FP17
C	0	DDRC0	CWOM	RC	SPE	MSTR, SPR	SCK
	1	DDRC1				DORD, SPR	SDO
	2	DDRC2	1				SDI
	3	DDRC3	1		IRQE	IRQS	IRQ

NOTES:
1. Pullup resistor resistances are typical values with V_{DD} = 3 V. See **Section 15. Electrical Specifications** for more details.
2. Port C bits 2 and 3 are open-drain outputs and do not have CMOS drive capability.

7.6.6 Resistor Control Register 1

Figure 7-19. Resistor Control Register 1 (RCR1)

Bits 7–4 — Reserved

These bits are not used and always read as zero.

RBH — Port B Pullup Resistor (H)

When this bit is set, the pullup resistor is connected to the upper four bits of port B. However, for those pins configured as CMOS output or open-drain output with output of logic low, the pullup resistors are disabled. This bit is cleared on reset.

RBL — Port B Pullup Resistor (L)

When this bit is set, the pullup resistor is connected to the lower four bits of port B. However, for those pins configured as CMOS output or open-drain output with output of logic low, the pullup resistors are disabled. This bit is cleared on reset.

RAH — Port A Pullup Resistor (H)

When this bit is set, the pullup resistor is connected to the upper four bits of port A. However, for those pins configured as CMOS output or open-drain output with output of logic low, the pullup resistors are disabled. This bit is cleared on reset.

RAL — Port A Pullup Resistor (L)

When this bit is set, the pullup resistor is connected to the lower four bits of port A. However, for those pins configured as CMOS output or open-drain output with output of logic low, the pullup resistors are disabled. This bit is cleared on reset.

7.6.7 Resistor Control Register 2

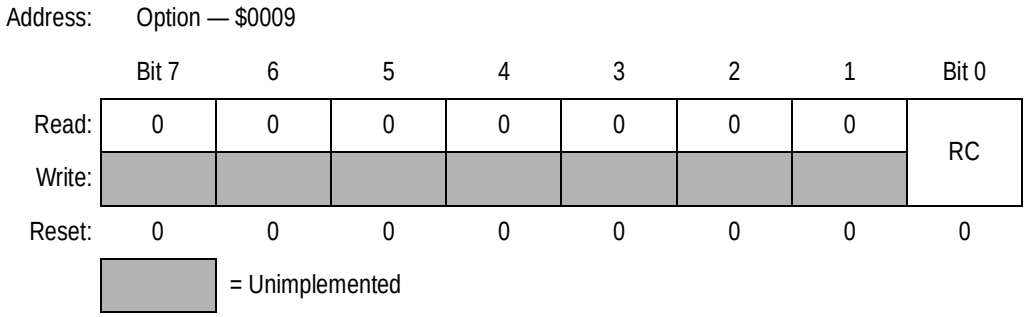


Figure 7-20. Resistor Control Register 2 (RCR2)

Bits 7:1 — Reserved

These bits are not used and always read as zero.

RC — Port C Pullup Resistor

When the RC bit is set, the pullup resistor is connected to all four bits of port C. However, for those pins configured as CMOS output or open-drain output with output of logic low, the pullup resistors are disabled. This bit is cleared on reset.

7.6.8 Open Drain Output Control Register

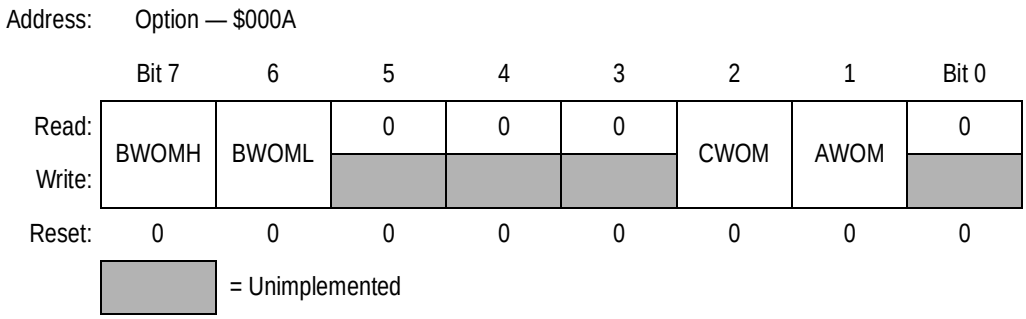


Figure 7-21. Wired-OR Mode Register (WOM)

BWOMH — Port B Open-Drain Mode (H)

When this bit is set, the upper four bits (7–4) of port B pins that are configured as outputs become open-drain outputs. This bit is cleared on reset.

BWOML — Port B Open-Drain Mode (L)

When this bit is set, the lower four bits (3–0) of port B pins that are configured as outputs become open-drain outputs. This bit is cleared on reset.

Bits 5–3 — Reserved

These bits are not used and always read zero.

CWOM — Port C Open-Drain Mode

When this bit is set, port C pins that are configured as outputs become open-drain outputs. This bit is cleared on reset.

AWOM — Port A Open-Drain Mode (High Nibble)

When this bit is set, the upper four bits of port A that are configured as outputs become open-drain outputs. This bit is cleared on reset.

Bit 0 — Reserved

This bit is not used and always returns to zero.

Section 8. Oscillators and Clock

8.1 Contents

8.2 Introduction

The MC68HC05L25 has dual on-chip oscillators for typical 4.0-MHz and 32.768-kHz crystals. Refer to **Figure 8-1**. The clock generated is used by the CPU and by the subsystem modules such as time base and LCD. Refer to **Figure 8-3**.

8.3 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 these conditions:

- Power-on detection
- When FOSCE bit is cleared

8.4 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 two is selected on reset.

Table 8-1. System Bus Frequency Selection

SYS1	SYS0	Divide Ratio	CPU Bus Frequency (Hz)		
			OSC = 4.0 M	OSC = 4.1943 M	XOSC = 32.768 K
0	0	OSC Divided by 2 (Default)	2.0 M	2.0972 M	—
0	1	OSC Divided by 4	1.0 M	1.0486 M	—
1	0	OSC Divided by 64	62.5 k	65.536 k	—
1	1	XOSC Divided by 2	—	—	16.384 k

8.5 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 clearing the FOSCE bit in the MISC register. The effects of restarting the OSC will vary depending on the current state of the MCU, including SYS0–SYS1 and FOSCE bits.

NOTE: *Do not switch the system clock to XOSC (SYS1–SYS0 = 11) when XOSC clock is not available. XOSC clock is available when STUP flag is set.*

Do not switch the system clock to OSC (SYS1–SYS0 = 00, 01, or 10) when OSC clock is not available. OSC clock is available when FTUP flag is set.

8.6 OSC On Line

If the system clock is OSC, FOSCE should remain set. 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 $\overline{\text{IRQ}}$ or $\overline{\text{RESET}}$ re-starts the oscillator. When the POR counter overflows, internal reset is released and execution can begin. The stabilization time will vary between 8064 and 8192 counts.

NOTE: *Exiting STOP with external $\overline{IRQ}$ will always return the MCU to the state as defined by the register definitions prior to executing the STOP instruction.*

8.7 XOSC On Line

If XOSC is the system clock (SYS0–SYS1 = 1–1), OSC can be stopped either by the STOP instruction or by clearing the FOSCE bit.

The suboscillator (XOSC) never stops except during powerdown. This clock can also be used as the clock source of the system clock and time base.

OSC and XOSC pins have options for feedback and damping resistor implementations. These options are set through mask option and can be read through the MOSR register.

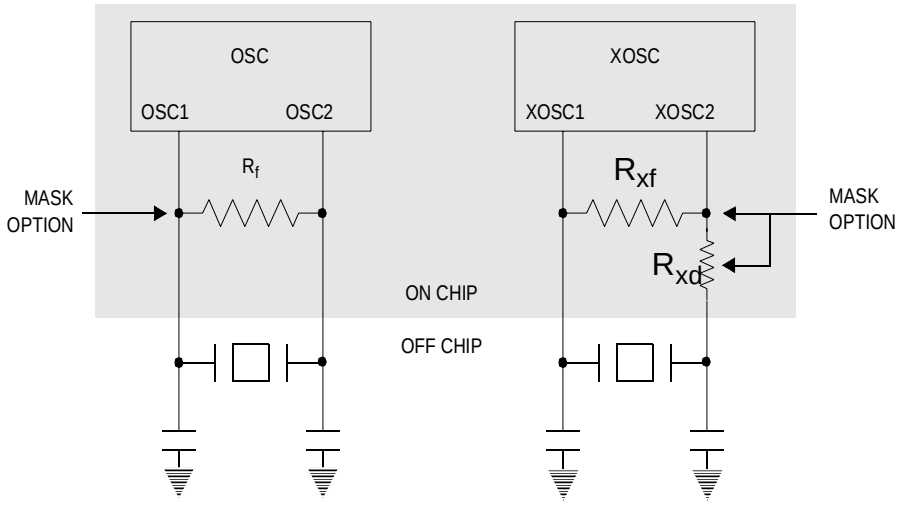


Figure 8-1. OSC1, OSC2, XOSC1, and XOSC2 Mask Options

8.7.1 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 $\overline{\text{IRQ}}$ 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 8064 and 8192 counts.

8.7.2 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.

8.7.3 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, the 7-bit divider and 6-bit POR counter are preset to $0078 (since FOSCE = 0), and execution is halted. Exiting STOP with external $\overline{\text{IRQ}}$ does not re-start the OSC; however, execution begins immediately using XOSC. OSC can be re-started by setting FOSCE, and when the POR counter overflows, FTUP be will set, signaling that OSC is stable and can be used as the system clock. The stabilization time will be 8072 counts.

8.7.4 Unused XOSC

When XOSC is not used, the XOSC1 pin must be connected to the $\overline{\text{RESET}}$ pin to ensure proper initialization of clock circuitry. The XOSC2 pin should be left unconnected. See **Figure 8-2**. Configure time base by setting the TBCLK bit in TBCR1 to receive clock from fast oscillator OSC.

NOTE: *When XOSC is not used, the XOSC1 input pin should be connected to $\overline{\text{RESET}}$ pin to ensure proper initialization of clock circuitry.*

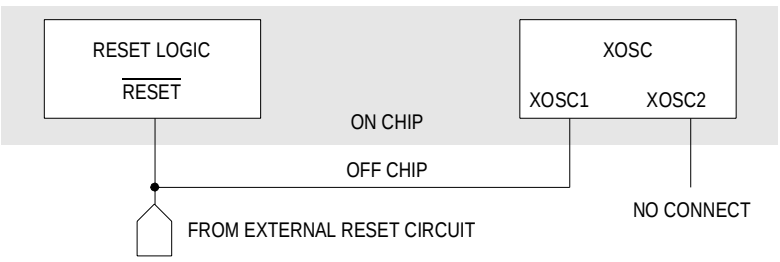


Figure 8-2. Unused XOSC1 Pin

8.8 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 the FTUP flag in the MISC register and initializes the POR counter. Stop mode is exited by $\overline{\text{RESET}}$, $\overline{\text{IRQ}}$, $\overline{\text{KWI}}$, SPI (slave mode), or TBI interrupt.

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

If XOSC is selected as the 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. Wait mode is exited by the $\overline{\text{RESET}}$ and any interrupts.

Table 8-2. CPU Startup Time Requirements

Before RESET or Interrupt			Power-On Reset	External $\overline{\text{RESET}}$	Exit Stop Mode by an Interrupt
CPU Clock Source	CPU	FOSCE			
—	—	—	Delay	—	—
OSC (OSC ON)	RUN	1	—	No Delay	—
OSC (OSC OFF)	RUN	0[1]	—	Delay	—
	STOP	1	—	Delay	Delay
	STOP	0[2]	—	Delay	Delay
XOSC (OSC ON)	RUN	1	—	No Delay	—
XOSC (OSC OFF)	RUN	0	—	Delay	—
	STOP	1	—	Delay	No Delay
	STOP	0	—	Delay	No Delay

NOTES:
1. Do not enter this state.
2. This state does not exist.

NOTE: *Power-on reset is strictly for power-on conditions and does not detect a drop in power.*

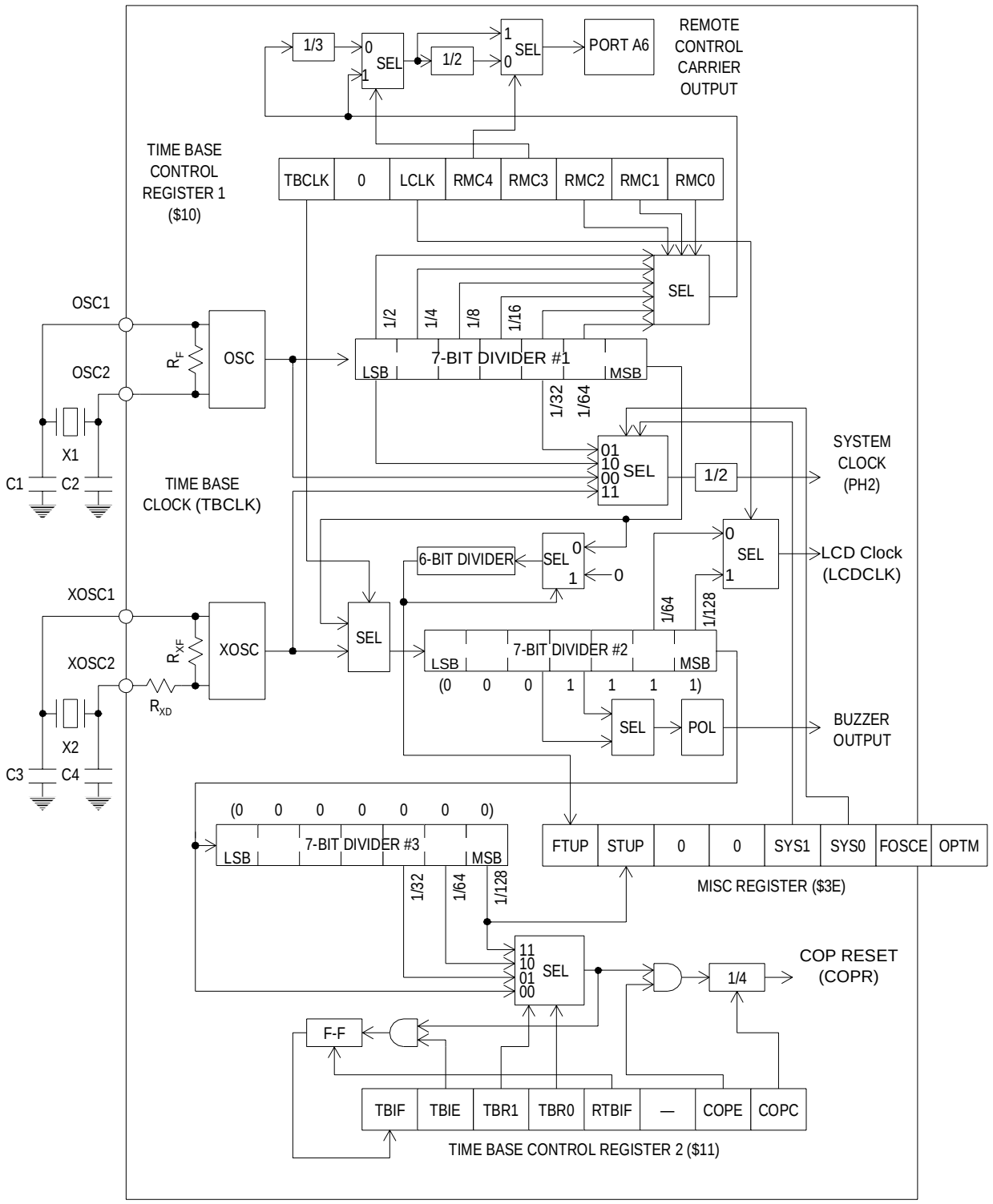


Figure 8-3. Clock Signal Distribution

8.9 XOSC Clock Divider and POD Counter

The XOSC clock divided by a 14-bit counter, also called power-on divider (POD), is used for the system clock. The oscillator clock divided by two is used by the system clock and oscillator clock divided by 64 or 128 is provided for the LCD module.

The overflow of the POD counter is used for setting the STUP bit and releasing power-on reset (POR). The 14-bit divider/POD counter is initialized to $0078 by power-on detection.

8.10 System Clock Control

The system clock (PH2) is provided for CPU and all internal modules.

8.11 XOSC

The oscillator (XOSC) runs continuously after power-up. The XOSC never stops while power is applied.

XOSC pins have options for feedback and damping resistor implementations. These options are set through mask option and may be read through the mask option status register (MOSR). See **1.4 Mask Options**.

8.12 Stop and Wait Modes

Power reduction can be achieved by executing the STOP instruction and halting the CPU. During stop mode, the CPU and all modules except time base are halted. The stop mode is exited by external $\overline{\text{RESET}}$, COP reset, $\overline{\text{IRQ}}$, SPI (slave mode), or TB interrupt. The CPU resumes immediately from stop mode since XOSC never stops oscillating during stop mode.

The CPU clock is halted and the peripheral modules are not affected during wait mode. Wait mode is exited by $\overline{\text{RESET}}$ or any interrupts.

Table 8-3. Recovery Time Requirements

Mode Before Reset or Interrupt	Delay Time After Reset or Interrupt		
	Power-On Reset	COP and External RESET	Exit Stop Mode by Interrupt
Power Off	Delay	—	—
Run	See Note	No Delay	—
Stop/Wait	See Note	No Delay	No Delay

NOTE:
Power-on reset is strictly for power-on conditions and does not detect a drop in power.

8.13 Miscellaneous Register

Address: $003E

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	FTUP	STUP	0	0	SYS1	SYS0	FOSCE	OPTM
Write:								
Reset:	U	U	0	0	0	0	1	0

= Unimplemented U = Unaffected

Figure 8-4. Miscellaneous Register (MISC)

FTUP — OSC Time Up Flag

Power-on detection and clearing FOSCE bit clears this bit. This bit is set by the overflow of the POR counter. A reset does not affect this bit.

Read:

1 = OSC clock available for the system clock
0 = During POR or OSC shut down

STUP — XOSC Time Up Flag

The power-on detection clears this bit. This bit is set after the time base has counted 16,264 clocks. A reset does not affect this bit.

Read:

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

Bits 5 and 4 — Reserved

These bits are not used and always read as zero.

SYS1 and SYS0 — System Clock Select

These two bits select the system clock source. On reset the SYS1 and SYS0 bits are initialized to zero and zero, respectively.

Table 8-4. System Bus Frequency Selection

SYS1	SYS0	Divide Ratio	ϕ2 CPU Bus Frequency (Hz)		
			OSC = 4.0 M	OSC = 4.1943 M	XOSC = 32.768 k
0	0	OSC Divided by 2 (Default)	2.0 M	2.0972 M	—
0	1	OSC Divided by 4	1.0 M	1.0486 M	—
1	0	OSC Divided by 64	62.5 k	65.536 k	—
1	1	XOSC Divided by 2	—	—	16.384 k

NOTE: *Do not switch the system clock to XOSC (SYS1–SYS0 = 11) when the XOSC clock is not available. The XOSC clock is available when the STUP flag is set.*

Do not switch the system clock to OSC (SYS1–SYS0 = 00, 01, or 10) when the OSC clock is not available. OSC clock is available when the FTUP flag is set.

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. This bit is set on reset.

1 = When this bit is set:

1. The main oscillator starts again.
2. The FTUP flag is set by the POR counter overflow (8072 clocks) and OSC is ready to be used as the system and time base clocks.

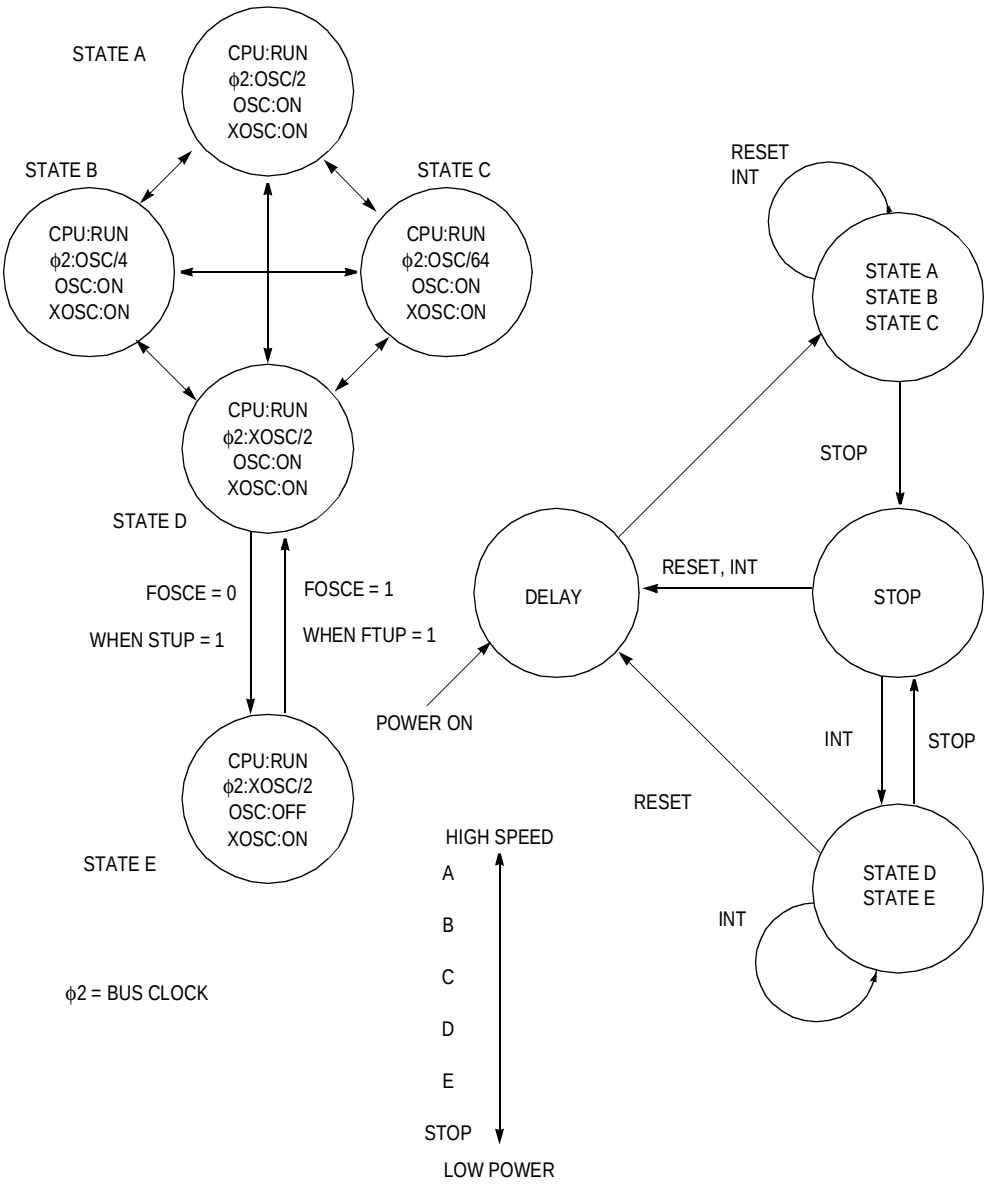
0 = When this bit is cleared:

1. OSC is shut down.
2. The 7-bit dividers at the OSC input and POR counter are initialized to $0078.
3. The FTUP flag is cleared.

OPTM — Option Map Select

The OPTM bit selects one of two register maps at $0000–$000F. This bit is cleared on reset.

1 = Option map selected
0 = Main register map selected



NOTES:
1. When switching from state E to state D, the FTUP bit in the MISC register must be set.
2. When switching from state D to state E, the STUP bit in the MISC register must be set.

Figure 8-5. Clock State and STOP/POD Delay Diagram

Section 9. Time Base

9.1 Contents

9.2 Introduction

Time base is a 14-bit up-counter which is clocked by XOSC.

This 14-bit divider is initialized to $0078 only upon power-on reset (POR). After counting 16,264 clocks, the STUP bit in the MISC register is set. See **Figure 8-4** for more information.

9.3 Time Base Submodules

The clock divided by the time base is used for LCDCLK, STUP, TBI, and COP. The time base clock can be provided either from OSC or XOSC.

9.3.1 LCDCLK

One of four clock frequency combinations can be selected for the LCD clock.

Table 9-1. LCD Clock Frequency

TBCR1		Divide Ratio	LCD Clock Frequency (Hz) (f_{XOSC} = 32.768 kHz)		
TBCLK	LCLK		OSC = 2.0 MHz	OSC = 4.0 MHz	OSC = 4.1943 MHz
0	0	XOSC ÷ 64	512	512	512
0	1	XOSC ÷ 28	256	256	256
1	0	OSC ÷ 8192	244	488	512
1	1	OSC ÷ 16,384	122	244	256

9.3.2 STUP

The time base divider is initialized to $0078 at power-on, and when the count reaches 16,264, the STUP flag in the MISC register is set. Once the STUP flag is set, it is never cleared while power is applied.

9.3.3 TBI

Time base interrupt can be generated every 0.5, 0.25, 0.125, or 0.0039 seconds with a 32.768-kHz crystal at XOSC pins. See **Table 9-2**.

Time base interrupt flag (TBIF) is set every period and an interrupt is requested if the enable bit (TBIE) is set. The clock divided by 128, 4096, 8192, or 16,384 is used to set TBIF, and this clock is selected by the TBR1 and TBR0 bits in the TBCR2 register.

Table 9-2. Time Base Interrupt Frequency

TBCR2		Divide Ratio	Frequency (Hz)		
TBR1	TBR0		OSC = 4.0 MHz	OSC = 4.1943 MHz	XOSC = 32.768 kHz
0	0	TBCLK ÷ 128	244	256	256
0	1	TBCLK ÷ 4096	7.63	8.00	8.00
1	0	TBCLK ÷ 8192	3.81	4.00	4.00
1	1	TBCLK ÷ 16,384	1.91	2.00	2.00

9.3.4 COP

The computer operating properly (COP) watchdog timer on the MC68HC05L25 is controlled by the COPE and COPC bits in the TBCR2 register.

The COP uses the same clock as TBI that is selected by the TBR1 and TBR0 bits. The TBI clock is divided by four and overflow of this divider generates COP timeout reset if the COP enable (COPE) bit is set. The COP timeout reset has the same vector address as power-on and external $\overline{\text{RESET}}$. To prevent the COP timeout, the COP divider is cleared by writing a one to the COP clear (COPC) bit.

When the time base divider is driven by the OSC clock, the clock for the divider is suspended during stop mode or when FOSCE is equal to zero. This may cause stretching of the COP period or no COP timeout reset occurring when there is a processing error. It is recommended that the XOSC clock be used as the clock source for time base to avoid these problems.

When the COP is enabled during stop mode and the time base is driven by the XOSC clock, the divider does not stop counting and the COPC bit must be triggered to prevent the COP timeout. It is recommended that the COP watchdog should be disabled for a system that must have intentional use of the stop mode period longer than the COP timeout period.

Table 9-3. COP Timeout Period

TBCR2		COP Period (ms)					
TBR1	TBR0	OSC = 4.0 MHz		OSC = 4.1943 MHz		XOSC = 32.768 kHz	
		Min	Max	Min	Max	Min	Max
0	0	12.3	16.4	11.7	15.6	11.7	15.6
0	1	393	524	375	500	375	500
1	0	786	1048	750	1000	750	1000
1	1	1573	2097	1500	2000	1500	2000

9.3.5 Remote Control Carrier Generator

The PA6/RMO pin functions as a general-purpose I/O port after reset. The RMPE bit must be set in order to use this port as a remote control carrier output. The RMO outputs idle state is set by the RPOL bit when RMON is cleared. The RMCLK signal selected by RMC4–RMC0 bits is output on the pin when RMON is set.

Table 9-4. Remote Carrier Frequency Selection

RMC4	RMC3	RMC2: RMC0	RMO Duty	Divider	Remote Carrier Frequency on RMO Pin (RMCLK)		
					OSC = 440 kHz	OSC = 3.6 MHz	OSC = 4.0 MHz
0	1	0	50%	1/4	110 kHz	900 kHz	1000 kHz
		1		1/8	55.0 kHz	450 kHz	500 kHz
		2		1/16	27.5 kHz	225 kHz	250 kHz
		3		1/32	13.8 kHz	113 kHz	125 kHz
		4		1/64	6.88 kHz	56.3 kHz	62.5 kHz
		5		1/128	3.44 kHz	28.1 kHz	31.3 kHz
	0	0	50%	1/12	36.7 kHz	300 kHz	333 kHz
		1		1/24	18.3 kHz	150 kHz	167 kHz
		2		1/48	9.17 kHz	75.0 kHz	83.3 kHz
		3		1/96	4.58 kHz	37.5 kHz	41.7 kHz
		4		1/192	2.29 kHz	18.8 kHz	20.8 kHz
		5		1/384	1.15 kHz	9.38 kHz	10.4 kHz
1	1	0	50%	1/2	220 kHz	1800 kHz	2000 kHz
		1		1/4	110 kHz	900 kHz	1000 kHz
		2		1/8	55.0 kHz	450 kHz	500 kHz
		3		1/16	27.5 kHz	225 kHz	250 kHz
		4		1/32	13.8 kHz	113 kHz	125 kHz
		5		1/64	6.88 kHz	56.3 kHz	62.5 kHz
	0	0	33%	1/6	73.3 kHz	600 kHz	667 kHz
		1		1/12	36.7 kHz	300 kHz	333 kHz
		2		1/24	18.3 kHz	150 kHz	167 kHz
		3		1/48	9.17 kHz	75.0 kHz	83.3 kHz
		4		1/96	4.58 kHz	37.5 kHz	41.7 kHz
		5		1/192	2.29 kHz	18.8 kHz	20.8 kHz

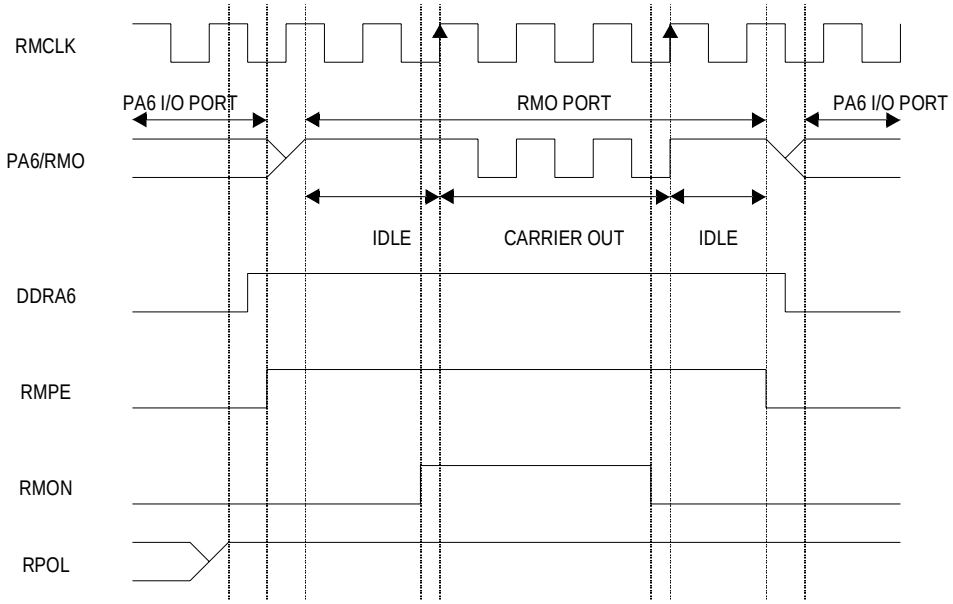


Figure 9-1. Remote Control Carrier Output Port Control (RPOL = 1)

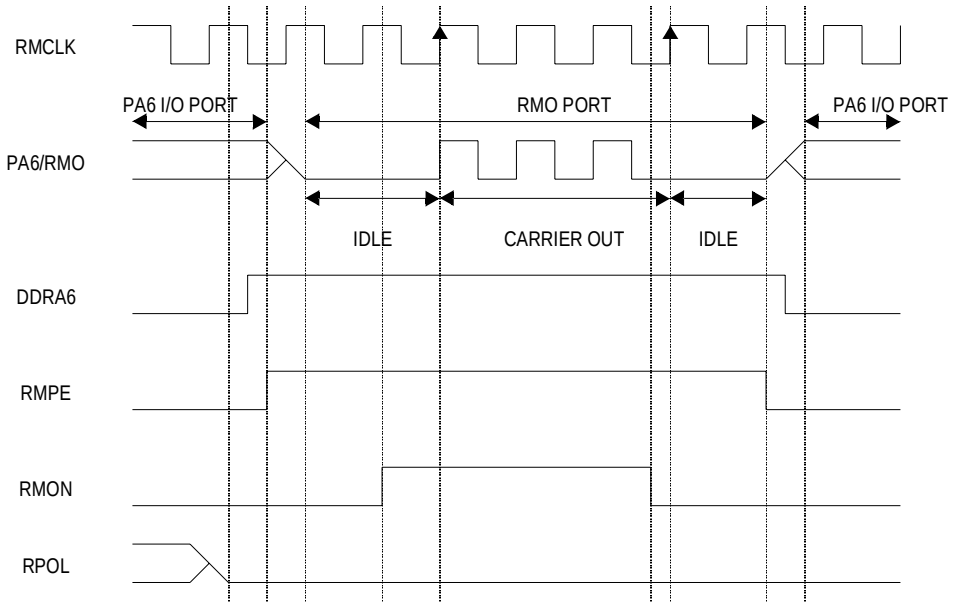


Figure 9-2. Remote Control Carrier Output Port Control (RPOL = 0)

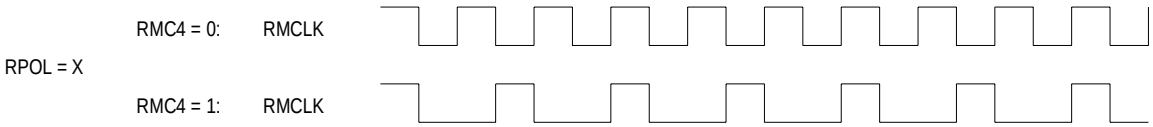


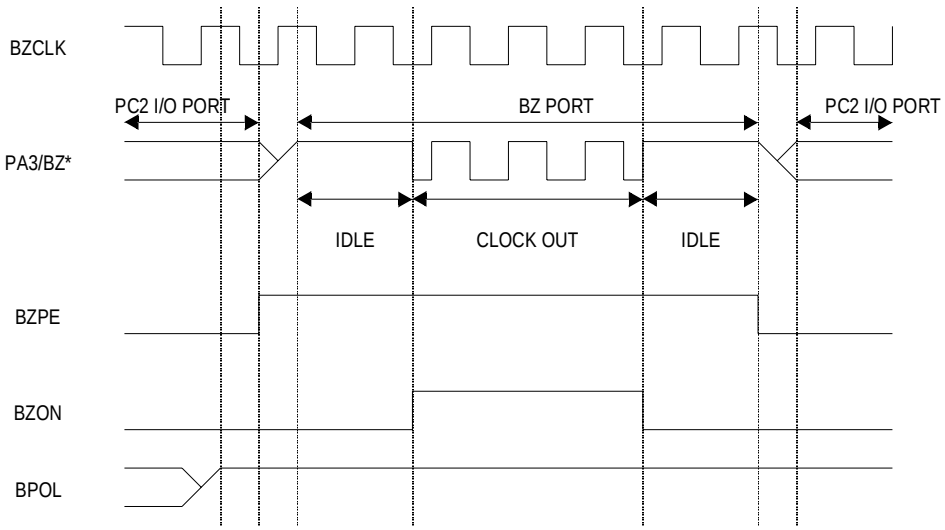
Figure 9-3. Remote Control Carrier Duty Control

9.3.6 Buzzer Tone Generator

The PA3/$\overline{\text{KWI3}}$/BZ pin functions as a general-purpose I/O port after reset. The BZPE bit must be set in order to use this port as buzzer tone output. The BZ outputs idle state is set by the BPOL bit when BZON is cleared. The BZCLK signal selected by the BCLK bit is output on the pin when BZON is set. The BZ output pin is open drain.

Table 9-5. Buzzer Frequency

TBCR1 TBCLK	TBCR3 BCLK	Buzzer Frequency on BZ Pin (BZCLK)			
		f_{OSC} = 2 MHz f_{XOSC} = 32.768 kHz	f_{OSC} = 3.6 MHz f_{XOSC} = 32.768 kHz	f_{OSC} = 4 MHz f_{XOSC} = 32.768 kHz	f_{OSC} = 4.194304 MHz f_{XOSC} = 32.768 kHz
0	0	4096 Hz	4096 Hz	4096 Hz	4096 Hz
0	1	2048 Hz	2048 Hz	2048 Hz	2048 Hz
1	0	Approximately 1953 Hz	Approximately 3516 Hz	Approximately 3906 Hz	4096 Hz
1	1	Approximately 977 Hz	Approximately 1758 Hz	Approximately 1953 Hz	2048 Hz



*The BZ output pin is open drain. The logic 1 shown for the BZ pin is actually a hi-Z state unless it is a pullup.

Figure 9-4. Buzzer Tone Output Control

9.4 Time Base Control Register 1

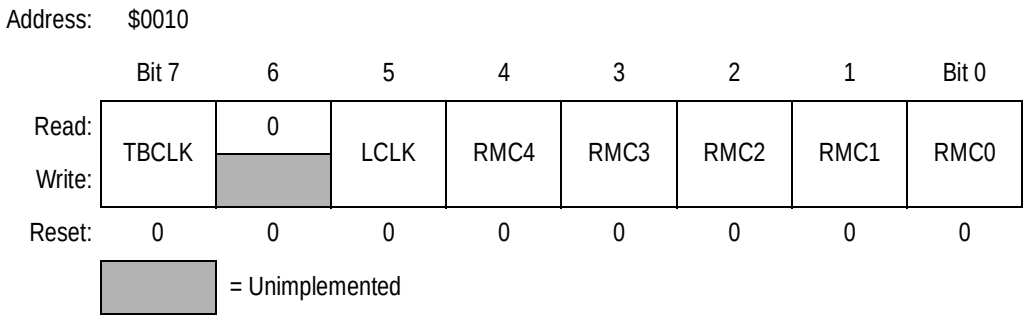


Figure 9-5. Time Base Control Register 1 (TBCR1)

TBCLK — Clock Source

This bit selects the time base clock source and is cleared on reset.

1 = OSC selected for time base clock source
0 = XOSC selected for the time base clock source

Bit 6 — Reserved

This bit is not used and always reads as zero.

LCLK — LCD Clock

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

When TBCLK = 0:

1 = XOSC divide by 128 selected for the LCD clock
0 = XOSC divide by 64 selected for the LCD clock

When TBCLK = 1:

1 = OSC divide by 16,384 selected for the LCD clock
0 = OSC divide by 8192 selected for the LCD clock

RMC4 — Remote Control Generator Divider

This bit selects the remote control carrier duty cycle and is cleared on reset.

1 = 33-67% duty selected if RMC3 = 0
0 = 50-50% duty selected

RMC3:RMC0 — Remote Control Generator Divider

These bits select the remote control carrier frequency. (See **Table 9-4**.) These bits are cleared on reset.

9.5 Time Base Control Register 2

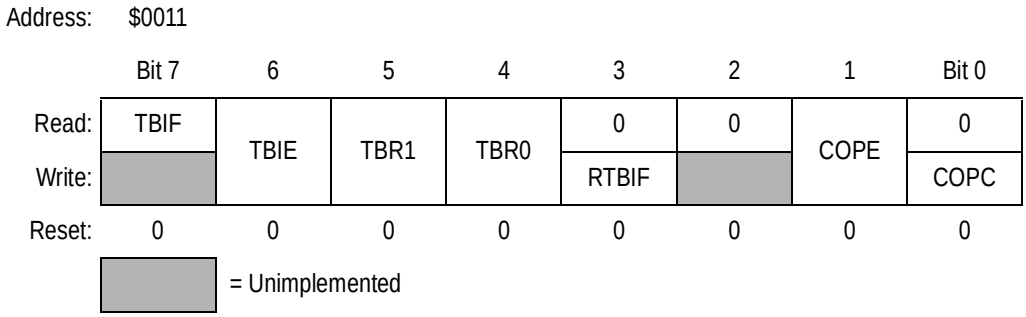


Figure 9-6. Time Base Control Register 2 (TBCR2)

TBIF — Time Base Interrupt Flag

The TBIF bit is set every timeout interval of the time base interrupt. This is a read-only bit and is cleared by writing a one to the RTBIF bit. Reset clears the TBIF bit. The time base interrupt period between reset and first TBIF depends on the time elapsed during reset, since the time base divider is not initialized on reset.

TBIE — Time Base Interrupt Enable

The TBIE bit enables the time base interrupt capability. If TBIF = 1 and TBIE = 1, the time base interrupt is generated.

1 = TB interrupt requested when TBIF = 1
0 = TB interrupt disabled

TBR1–TBR0 — Time Base Interrupt Rate Select

The TBR1 and TBR0 bits select one of four rates for the time base interrupt period. The TBI rate is also related to the COP timeout reset period. See **Table 9-2** and **Table 9-3**. These bits are set to a logical 1 on reset.

RTBIF — Reset Time Base Interrupt Flag

The RTBIF bit is a write-only bit and always reads as zero. Writing a one to this bit clears the TBIF bit and writing zero to this bit has no effect.

1 = Reset TBIF
0 = No effect

Bit 2 — Reserved

This bit is not used and always reads as zero.

COPE — COP Enable

When the COPE bit is one, the COP reset function is enabled. This bit is cleared on reset (including COP timeout reset) and a write to this bit is allowed only once after reset.

1 = COP enabled
0 = COP disabled

COPC — COP Clear

Writing a logical 1 to COPC bit clears the 2-bit divider to prevent COP timeout. (The COP timeout period depends on the TBI rate.) This is a write-only bit and returns to zero when read.

1 = Clear COP timeout divider
0 = No effect

9.6 Time Base Control Register 3

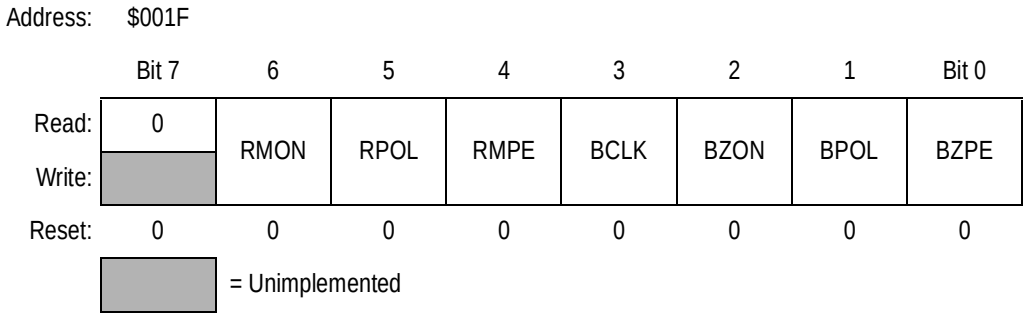


Figure 9-7. Time Base Control Register 3 (TBCR3)

Bit 7 — Reserved

This bit is not used and always reads as zero.

RMON — Remote Control Generator Signal ON

When the port is used as remote control output (RMPE = 1), this bit turns on or off the remote control signal. The idle state is output when cleared and is cleared on reset.

1 = Carrier on
0 = Carrier off; idle state defined by RPOL as output

RPOL — Remote Control Idle Polarity

This bit controls the idle state of the remote control generator output on the port and is cleared on reset.

1 = Remote idle state outputs logic 1.
0 = Remote idle state outputs logic 0.

RMPE — Remote Control Generator Port Output Enable

This bit enables the remote control generator output on the port. The actual remote signal on/off is controlled by RMON bit. This bit is cleared on reset.

1 = Port pin functions as remote control output.
0 = Port pin functions as general I/O port.

BCLK — Buzzer Clock Select

This bit selects the buzzer tone output frequency. This bit is cleared on reset.

1 = OSC/2^{11} or XOSC/2^4 selected for buzzer clock
0 = OSC/2^{10} or XOSC/2^3 selected for buzzer clock

BZON — Buzzer Signal ON

When the port is used as buzzer output (BZPE = 1), BZON turns on the buzzer signal. Reset clears BZON.

1 = Buzzer on
0 = Buzzer off; idle state defined by BPOL as output

BPOL — Buzzer Output Polarity

This bit selects the buzzer output pin's polarity during buzzer idle (standby) period (BZON = 0). When BZE = 0, this bit has no effect. This bit is cleared on reset.

1 = Buzzer idle state outputs logic 1.
0 = Buzzer idle state outputs logic 0.

BZPE — Buzzer Output Port Enable

This bit controls whether the port functions as buzzer output or a general I/O port. The actual buzzer signal on/off is controlled by the BZON bit. (See **Table 7-1**.) This bit is cleared on reset.

1 = Port pin functions as buzzer output.
0 = Port pin functions as general I/O port.

Section 10. Serial Peripheral Interface

10.1 Contents

10.2 Introduction

The serial peripheral interface (SPI) is built into the MC68HC05L25 to transmit or receive synchronous serial data. In this format, the serial clock is not included in the data stream and must be provided as a separate signal.

When the SPI is enabled, reading port C will return the actual pin level.

The MSTR bit selects the source of the serial clock from the internal or the external clock. The internal clock speed is selectable as 1/2 or 1/16 of the system clock.

10.3 Features

- Full Duplex 3-Wire Synchronous Transfers
- Master or Slave Operation
- Bit Rate Selection
- End of Transmission Interrupt
- Data Collision Flag
- Master Mode Maximum Serial Clock Speed at 1/2 the CPU System Clock
- Slave Mode Maximum Serial Clock Speed Up until the CPU System Clock

10.4 Block Diagram

Figure 10-1 illustrates the block diagram of the SPI module.

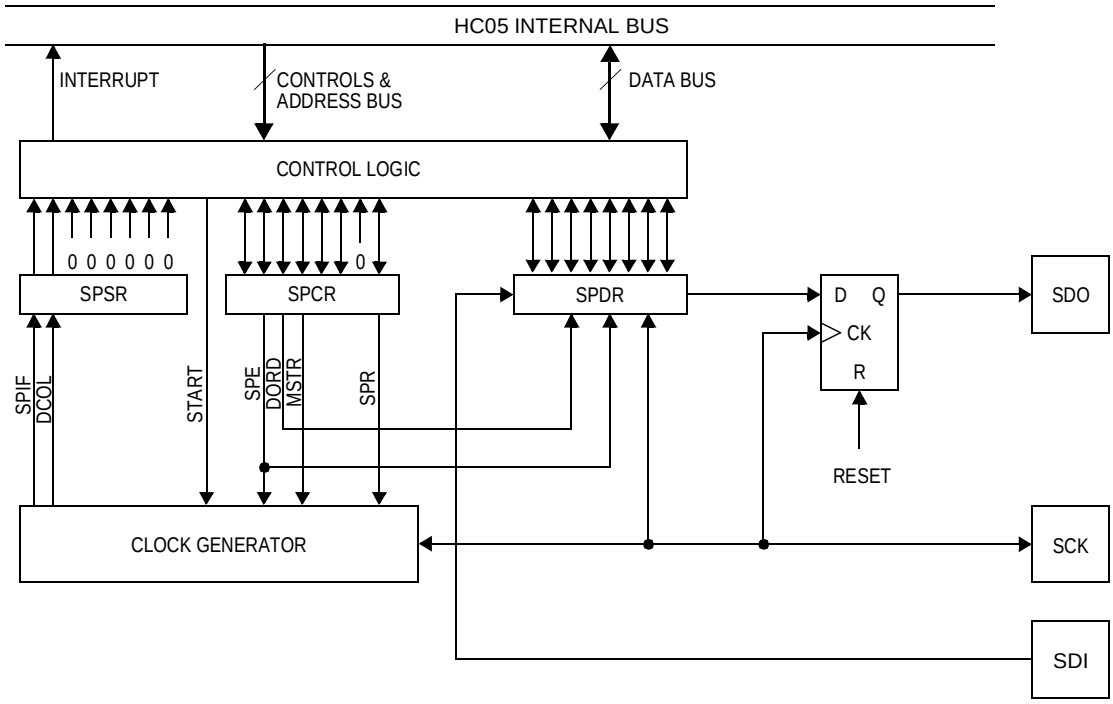


Figure 10-1. SPI Block Diagram

10.4.1 Control

The control logic is an interface to the HC05 internal bus. It generates the clock start signal, when writing to SPDR is detected in master mode. It also generates a flag clear signal and interrupt request to the CPU.

10.4.2 SPDR

The serial peripheral data register (SPDR) is an 8-bit shift register. This register can be read or written by the CPU. It can also change parallel data to serial or vice versa.

10.4.3 SPCR

The serial peripheral control register (SPCR) contains bits SPIE, SPE, DORD, SPR, and MSTR. The description on each bit can be found in **10.6.1 Serial Peripheral Control Register**.

10.4.4 Clock Generator

The clock generator includes a 3-bit serial clock counter. The counter starts after detecting the serial clock and halts after setting SPIF when the counter overflows.

In master mode, this block generates serial clock (SCK) when CPU writes to the data register (SPDR) and the clock rate is selected by SPR bit in the control register (SPCR).

In slave mode, external clock from the SCK pin is used instead of master mode clock, and SPR has no effect.

10.4.5 Others

The SPI does not use the data register of port C. Therefore, regardless of whether the SPI is used, the data register can be read from port C.

10.4.6 Signal Description

The basic signals SDO, SDI, and SCK of SPI are described in the following paragraphs. SCK, SDO, and SDI pins are shared with port C pins PC0, PC1, and PC2, respectively.

10.4.7 Serial Data Out (SDO)

SDO is an output pin. This pin is shared with port C pin PC1. When the SPI is enabled by SPE bit in the SPCR, this pin becomes an output pin. When the SPE is cleared, the pin becomes PC1 and thus becomes an input pin. The state of PC1/SDO can be read any time through PC1 data register.

When the SPI is enabled and PC1/SDO is an output, data output becomes valid at the falling edge of the serial clock.

10.4.8 Serial Data In (SDI)

The SDI pin is multiplexed with a general-purpose I/O pin. This becomes an input-only pin and accepts serial input data when the SPI is enabled.

10.4.9 Serial Clock (SCK)

The SCK pin is used for synchronization of both input and output data streams through SDI and SDO pins. The SCK pin should be at V_{DD} level before SPI is enabled.

The master and slave devices are capable of exchanging a data byte during a sequence of eight clock pulses. Since SCK is generated by the master, slave data transfer is accomplished by synchronization of SCK.

When the MSTR bit in the SPCR is set, SCK becomes an output and the serial clock is supplied to the internal and external systems. When the serial clock is idling, high level is being output. When the bit is a logic 1, the CPU writes data to SPCR and outputs eight clock pulses. After the end of the eighth clock, high level is being output while idling. The clock speed in master mode is one-half the system clock.

When the MSTR bit in the SPCR is cleared, SCK becomes an input and the external system supplies the serial clock while the internal system operates by synchronizing to this clock. After eight serial clocks are input to the SCK pin, the SPIF bit in the SPSR is set and will not receive the next serial clock input until the SPIF bit is cleared. The clock speed in slave mode is dependent upon the speed of the external system and has a maximum speed up till the internal system clock.

10.5 Functional Description

A block diagram of the SPI module is shown in **Figure 10-1**. In the SPI, if the SPE bit (SPI enable) of SPCR is set, bits 0, 1, and 2 of port C will be connected. During this time, bit 0 is used as the SCK (serial clock), bit 1 as the SDO (serial data out), and bit 2 will become SDI. When SPE is a logic zero, SPI system is disabled.

In master mode (MSTR = 1), SCK becomes an output. When the CPU writes data to SPDR, start trigger will be applied from the control logic to the clock generator. The clock generator divides the system clock of the CPU (by 2 or 16) to generate the serial clock which is then output to the SCK pin. This clock is also used in the 3-bit clock counter and 8-bit shift register (SPDR).

In slave mode (MSTR = 0), SCK becomes an input, and the external serial clock is used. Therefore, the internal clock generator will not generate the serial clock. After detecting the external clock, the clock will be used by the 3-bit clock counter and the 8-bit shift register (SPDR) located in the clock generator. The SCK is used to latch incoming data.

In either master or slave mode, the SPIF flag is set after the end of the transmission and if the SPIE bit in the SPCR is set, the interrupt request is sent to the CPU. This interrupt request is accepted when the I mask bit of condition code register (CCR) is a logic zero and is inhibited when the bit is a logic one or until the mask is released. Also, if the SPIE bit is cleared, the interrupt request will not be accepted by the CPU.

To clear the SPIF while it is still set, the SPDR must be read or written after accessing SPSR.

Regardless of the master/slave I/O conditions, the DCOL bit of SPSR will be set when SPDR is accessed while the shift register is operating and while SPSR is not being accessed with SPIF set. DCOL is used to indicate that the data is not being properly read or written into SPDR.

To clear the DCOL flag while it is still set, the SPDR must be read or written after accessing SPSR.

10.6 Register Description

The SPI has three registers: control register (SPCR), status register (SPSR), and data register (SPDR). SPCR and SPDR can be read or written by the CPU, but SPSR can only be read.

10.6.1 Serial Peripheral Control Register

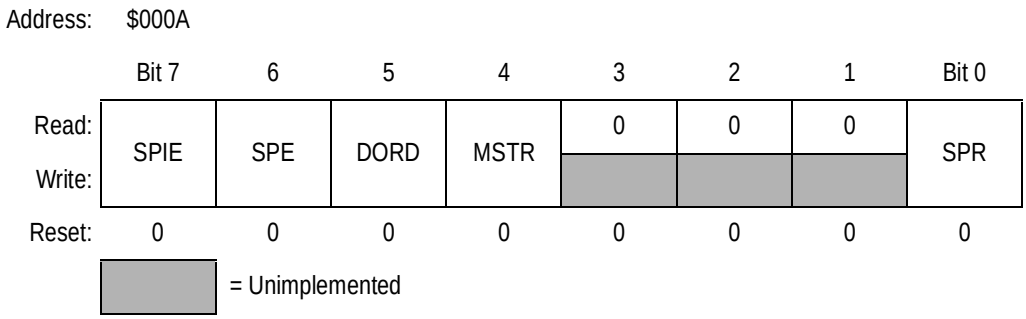


Figure 10-2. SPI Control Register (SPCR)

SPIE — SPI Interrupt Enable

When SPIE (SPI interrupt enable) is set, it allows the occurrence of processor interrupt when SPIF in the SPSR is set. This interrupt request is accepted when the I bit in the CCR is cleared but inhibited when I bit is set. If the interrupt request is sent repeatedly while the I bit and only when SPIE and SPIF are set, the interrupt will occur immediately after the I bit is cleared. Reset clears this bit.

1 = SPI interrupt enabled
0 = SPI interrupt disabled

SPE — SPI Enable

When SPE (SPI enable) is set, it enables the SPI system and connects bit 0 and bit 1 of port C to SCK and SDIO. Clearing SPE initializes the SPI system and disconnects SPI from port C. Reset clears this bit.

1 = SPI enabled
0 = SPI disabled

NOTE: *PC0/SCK should be at V_{DD} level before SPI is enabled. This can be done with an internal or external pullup resistor or by setting DDRC0 = 1 and PC0 = 1 prior to enabling the SPI. Otherwise, the circuit will not initialize correctly.*

DORD — Data Transmission Order

When DORD is set, the data in the 8-bit shift register (SPDR) is shifted in/out from LSB first. When clear, the data is shifted MSB first. Reset clears this bit.

1 = LSB first
0 = MSB first

MSTR — Master Mode Select

This MSTR (master mode select) bit determines whether to output the serial clock internally or input the clock externally. When set, SPI is in master mode and SCK is configured as an output pin. SCK outputs the serial clock when CPU writes data to SPDR. When cleared, the SPI is in slave mode and SCK is configured as an input pin. SCK receives the serial clock externally. Reset clears this bit.

1 = Master mode
0 = Slave mode

Bits 3:1 — Reserved

These bits are reserved and always read as zero.

SPR — SPI Clock Rate Select

This is the clock rate selection bit. When set, the master mode SCK rate is the system clock divided by 16. When clear, the rate system clock is divided by two. Reset clears this bit.

1 = System clock divided by 16
0 = System clock divided by 2

10.6.2 Serial Peripheral Status Register

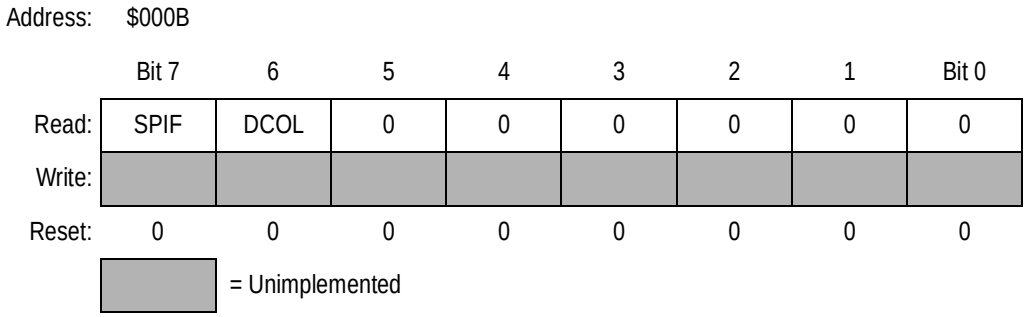


Figure 10-3. SPI Status Register (SPSR)

SPIF — Serial Transfer Complete Flag

SPIF (serial peripheral interface flag) notifies the user that the data transfer between MC68HC05L25 and the external device has been completed. Upon completion of the data transfer, the rising edge of the eighth serial clock pulse sets SPIF. If SPIE in the SPCR is set, the SPI interrupt (SPII) will be generated.

While SPIF is set, all access to the SPDR is inhibited until SPSR is read by the CPU. Also, even if the ninth serial clock is detected, the shift register (SPDR) will not operate.

Clearing the SPIF is accomplished by a software sequence of accessing the SPSR while SPIF is set and followed by the SPDR access. (SPIF and DCOL can be cleared simultaneously.)

Reset clears this bit.

1 = Serial data transfer complete
0 = Serial data transfer in progress

DCOL — Data Collision

DCOL (data collision) notifies the user that an invalid access to the SPDR has been made. This bit is set when an attempt was made to read or write to SPDR while a data transfer was taking place with an external device. When DCOL is set, access to the SPDR becomes invalid. The transfer continues uninterrupted without any effect from the SPDR access. This flag does not generate SPI interrupt. It is read-only.

DCOL is cleared by reading the SPSR with SPIF set followed by a read or write to the SPDR. If the last part of the clearing sequence is done after another transmission has started, DCOL will be set again. (DCOL and SPIF can be cleared simultaneously.)

Reset clears this bit.

1 = Data collision occurred

0 = Data collision did not occur

Bits 5–0 — Reserved

These bits are unused and always read as zero.

10.6.3 SPI Data Register

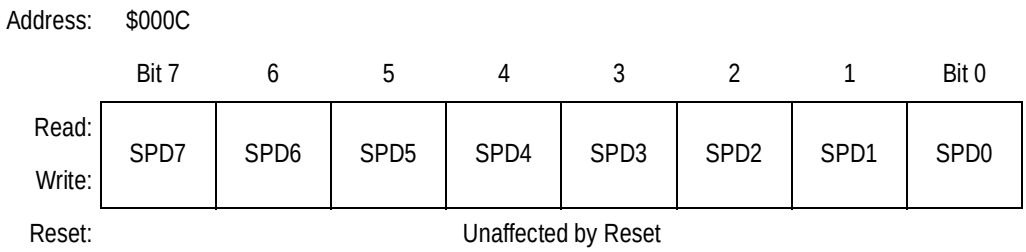


Figure 10-4. SPI Data Register (SPDR)

The SPDR is used to transmit and receive data on the serial bus.

In master mode, a write to SPDR initiates the transmission/reception of data byte. At transfer completion, SPIF status bits are set.

In slave mode, a write to the SPDR will not initiate the serial clock. The serial clock is input to the SCK pin by the external device.

In either master or slave mode, a write to the SPDR is inhibited while this register is shifting (this condition causes DCOL to set) or when SPIF is set without reading SPSR. In this case, even if an access has occurred, the access becomes invalid. Refer to SPIF and DCOL descriptions for more information.

When SPI is not being used, SPDR can be used as a data storage. This byte is not affected by reset.

10.6.4 Timing Diagram

Figure 10-5 illustrates the clock/data timing.

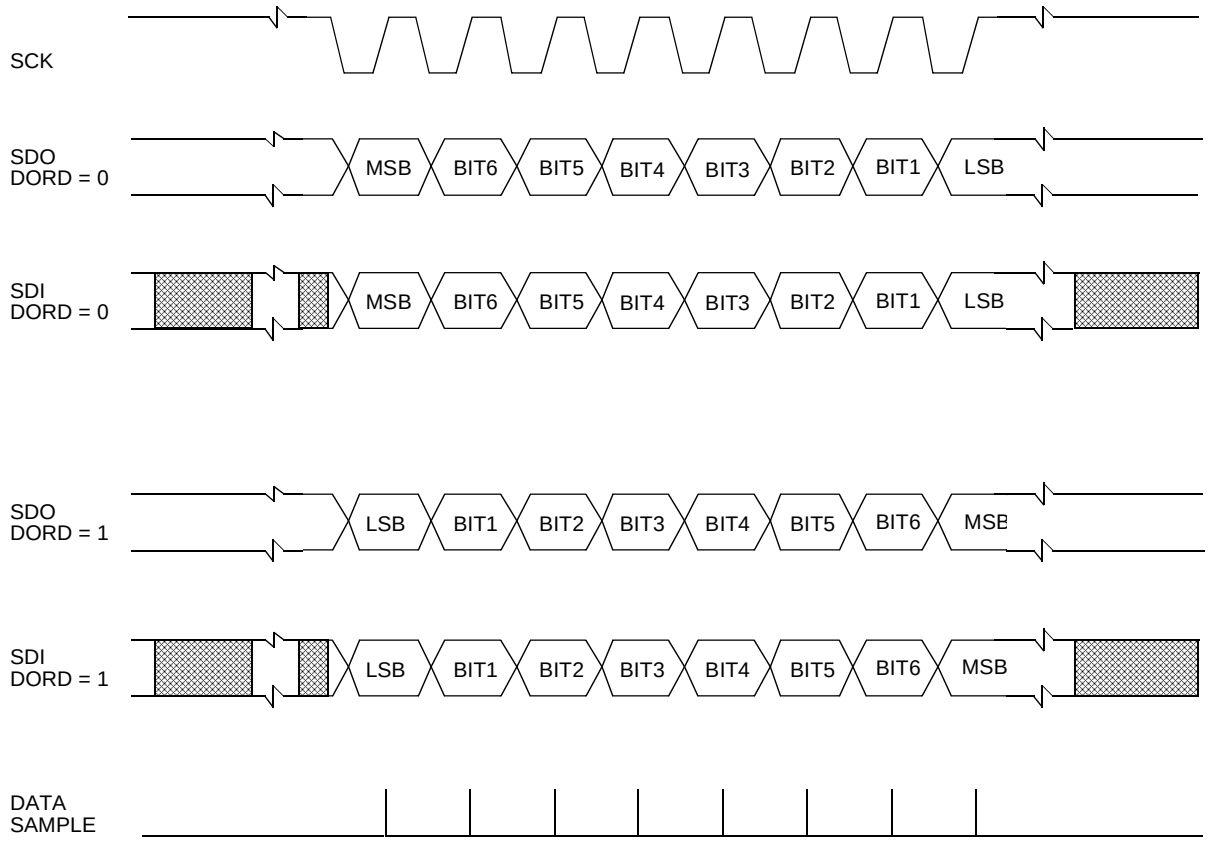


Figure 10-5. Clock/Data Timing

10.6.5 Stop/Wait Condition

The following paragraphs describe stop and wait modes.

10.6.6 Stop Mode

The SPI configured as master mode is not operational during stop mode since the system clock and SPI clock generator are halted. If stop mode occurs while SPI is in progress (transmitting/receiving) and in master mode, the access will halt and remains halted until stop is released.

Due to the static architecture, the previous conditions of SCK and SDIO are preserved during stop mode.

In slave mode, all accesses are possible during stop mode. However, at the end of transmission, interrupt occurs but the SPI will not be set immediately until after the system clock starts operating. (This operation is transparent to the programmer.)

10.6.7 Wait Mode

In wait mode, the CPU halts but will not affect the SPI operation. Therefore, SPI interrupt in master and slave modes can be executed to wake up the CPU.

Section 11. LCD Driver

11.1 Contents

11.2 Introduction

The MC68HC05L25 has 25 or 24 frontplane (FP) and three or four backplane (BP) drivers. The number of drivers for the FP and BP can be selected by software option. The maximum number of segments configurable is either 24 x 4 = 96 or 25 x 3 = 75 segments.

The MC68HC05L25 uses a 1/3 biasing method. The bias voltages are supplied from an external source using the V_{LCD} pin. Voltages V_{LCD1}, V_{LCD2}, and V_{LCD3} are generated internally with resistor divider.

There are 11 bytes of data latch for selection (turned on) or nonselection (turned off) of segments. Each byte consists of two FP drivers and either three or four BP drivers depending on the duty configured. The data latch is available in memory locations $21 through $2D and can be accessed by the CPU using the conventional memory access method (LOAD, STORE, BIT operations, etc.).

The clock which forms the LCD FP and BP waveforms is supplied by the time base module.

11.3 Block Diagram

Figure 11-1 illustrates the block diagram of the LCD module.

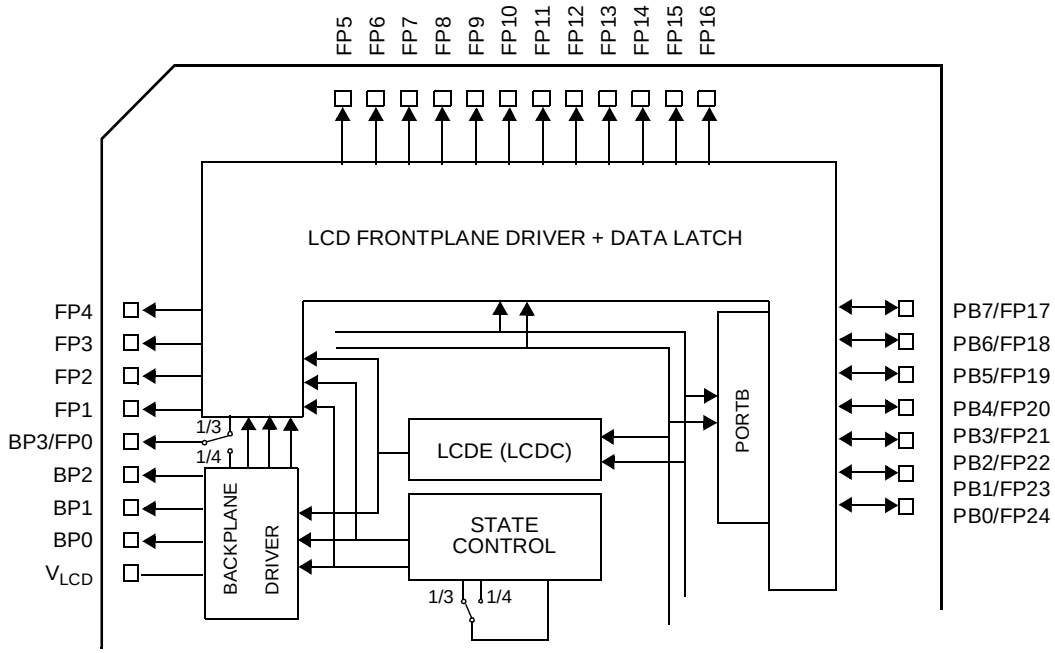


Figure 11-1. LCD Block Diagram

11.4 Functional Description

The following paragraphs provide a functional description of the LCD driver. See **Figure 11-15** for a simplified schematic of the LCD system.

11.4.1 LCD Control Register

Address: $0020

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	LCDE	PBEH	DUTY	PBEL	0	0	FC	LC
Write:								
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 11-2. LCD Control Register

LCDE — LCD Enable

Setting this bit enables the LCD waveforms to appear on the pins. Reset clears this bit.

1 = LCD drivers are operational. Each FP and BP driver outputs the driver waveform specified by the data latch.

0 = LCD drivers are halted. All FP and BP drivers have the same electric potential as V_{DD}. R_{LCD}s are disconnected to reduce DC current.

PBEH — Port B Enable High Nibble

This bit enables the port B I/O bits 4–7 that are multiplexed with frontplane drivers 20–17. Reset clears this bit.

0 = PB4–PB7/FP20–FP17 pins function as port B bits 4–7.

0 = PB4–PB7/FP20–FP17 pins function as LCD frontplane drivers 20–17.

DUTY — Duty Cycle Select

This bit selects the duty cycle of the LCD waveforms between 1/3 duty and 1/4 duty and selects either BP3 or FP0 for the muxed pin. Reset clears this bit.

1 = 1/3 duty cycle is selected and BP3/FP0 pin functions as FP0.

0 = 1/4 duty cycle is selected and BP3/FP0 pin functions as BP3.

PBEL — Port B Enable Low Nibble

This bit enables the port B I/O bits 0–3 that are multiplexed with frontplane drivers 24–21. Reset clears this bit.

1 = PB0–PB3/FP24–FP21 pins function as port B bits 0–3.

0 = PB0–PB3/FP24–FP21 pins function as LCD frontplane drivers 24–21.

Bits 3 and 2 — Reserved

These bits are unused and always read as zero.

FC and LC — Fast Charge and Low Current

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

Table 11-1. R_{LCD} Configuration

FC	LC	Action
X	0	Default value of approximately 160 kΩ per resistor
0	1	Resistor value of approximately 860 kΩ per resistor
1	1	Fast-Charge: For a period of LCDCLK/128 in each frame, the resistor values are reduced to default (value for LC = 0).

11.4.2 Fast Change Option

The R_{LCD} is approximately 160 kΩ at V_{DD} = 3 V by default. This value can be inappropriate for some applications. For those applications that require less DC current drain through the R_{LCD} chain, it may be increased to approximately 860 kΩ at V_{DD} = 3 V by setting the LC bit in LCDCR. Some applications may require the default resistance to drive the capacitive load of the LCD panel, yet do not wish to have the DC current drain of it while the LCD segments are not switching. For a compromise, a fast-charge option is available. The R_{LCD} values are reduced to the default resistance for a fraction of the LCD segment cycle before the LCD segments change, and then are set to low-current mode for the remainder of the LCD cycle frame. The DC current increase is very negligible and will be within a few percent increase of the low-current mode.

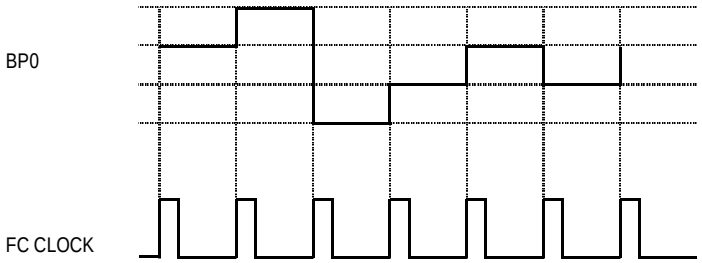


Figure 11-3. BP0/FC Clock Timing

11.4.3 LCD Data Registers

The LCD data latches LDAT1 through LDAT11 maintain the ON/OFF data for the FP and BP segments of the LCD.

Four bits of data latch are assigned to each frontplane driver from address space $21 through $2D as shown in **Figure 11-4**.

When a logic 1 is written to the bits in the data latch, the applicable FP-BP segment turns ON. When a logic 0 is written to the bits, the segment is turned OFF.

The values in the data latches are not initialized and are unknown on reset.

If 1/3 duty is selected, each BP3 bit in the data latches is ignored.

Addr Hex	Register Name		Bit Number Sequence							
			Bit 7	6	5	4	3	2	1	Bit 0
$0020	LCD Control Register (LCDCR)	Read:	LCDE	PBEH	DUTY	PBEL	0	0	FC	LC
		Write:								
$0021	LCD Data Register (LDAT1)	Read:	F1B3	F1B2	F1B1	F1B0	F0B3	F0B2	F0B1	F0B0
		Write:								
$0022	LCD Data Register (LDAT2)	Read:	F3B3	F3B2	F3B1	F3B0	F2B3	F2B2	F2B1	F2B0
		Write:								
$0023	LCD Data Register (LDAT3)	Read:	F5B3	F5B2	F5B1	F5B0	F4B3	F4B2	F4B1	F4B0
		Write:								
$0024	LCD Data Register (LDAT4)	Read:	F7B3	F7B2	F7B1	F7B0	F6B3	F6B2	F6B1	F6B0
		Write:								
$0025	LCD Data Register (LDAT5)	Read:	F9B3	F9B2	F9B1	F9B0	F8B3	F8B2	F8B1	F8B0
		Write:								
$0026	LCD Data Register (LDAT6)	Read:	F11B3	F11B2	F11B1	F11B0	F10B3	F10B2	F10B1	F10B0
		Write:								
$0027	LCD Data Register (LDAT7)	Read:	F13B3	F13B2	F13B1	F13B0	F12B3	F12B2	F12B1	F12B0
		Write:								
$0028	LCD Data Register (LDAT8)	Read:	F15B3	F15B2	F15B1	F15B0	F14B3	F14B2	F14B1	F14B0
		Write:								
$0029	LCD Data Register (LDAT9)	Read:	F17B3	F17B2	F17B1	F17B0	F16B3	F16B2	F16B1	F16B0
		Write:								
$002A	LCD Data Register (LDAT10)	Read:	F19B3	F19B2	F19B1	F19B0	F18B3	F18B2	F18B1	F18B0
		Write:								
$002B	LCD Data Register (LDAT11)	Read:	F21B3	F21B2	F21B1	F21B0	F20B3	F20B2	F20B1	F20B0
		Write:								
$002C	LCD Data Register (LDAT12)	Read:	F23B3	F23B2	F23B1	F23B0	F22B3	F22B2	F22B1	F22B0
		Write:								
$002D	LCD Data Register (LDAT13)	Read:	0	0	0	0	F24B3	F24B2	F24B1	F24B0
		Write:								

= Unimplemented

Figure 11-4. LCD Data Registers (LDAT1–LDAT13)

11.5 Terminal Description

A total of 27 pins is dedicated to the LCD driver.

11.5.1 V_{LCD} Bias Inputs

V_{LCD1} through V_{LCD3} are internal bias voltages for the LCD driver waveforms. V_{LCD3} potential is available externally as the V_{LCD} pin, and a variable resistor for contrast can be placed between V_{LCD} and V_{SS}. See **Figure 11-15**.

The LCD uses the three bias voltages typically as follows:

1. $V_{LCD1} = V_{DD}$ — 1/3 V_{LCDA} (V_{LCDA} is the ON voltage for the LCD modules.)
2. $V_{LCD2} = V_{DD}$ — 2/3 V_{LCDA} (Usually, $V_{LCDA} \leq V_{DD}$ is used.)
3. $V_{LCD3} = V_{DD}$ — $V_{LCDA} = V_{LCD}$ (V_{LCD} is the external pin.)

The three voltages shown above are arranged so that the external voltages will have a $V_{LCD1} > V_{LCD2} > V_{LCD3}$ relationship in a voltage divider configuration.

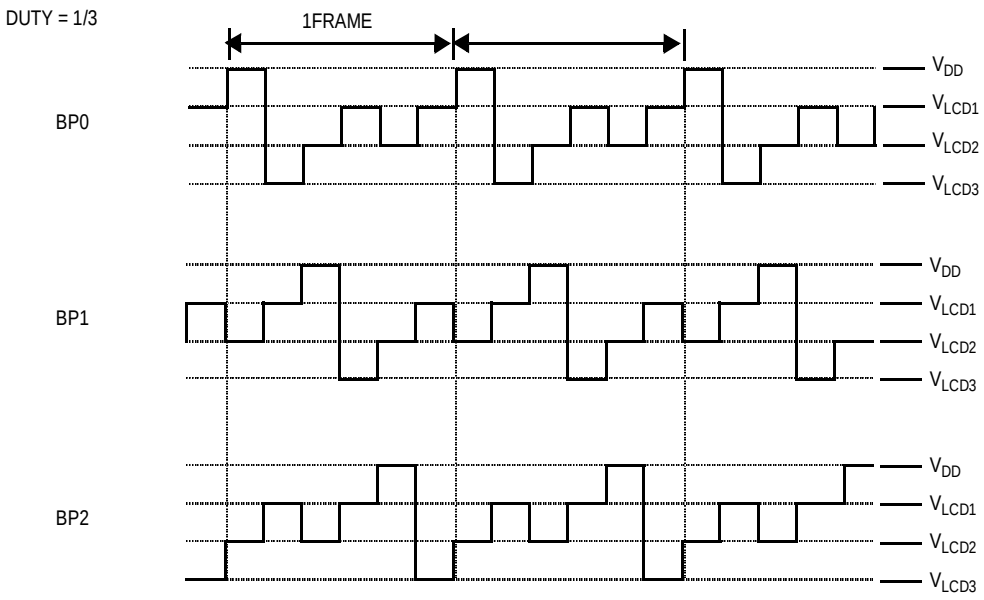
11.5.2 Backplane Drivers (BP0–BP3)

Pins BP0–BP3 are the output terminals for the backplane drivers.

These are connected to the backplane of the LCD panel. Depending on the duty, the waveforms in **Figure 11-5** and **Figure 11-6** appear on the backplane pins.

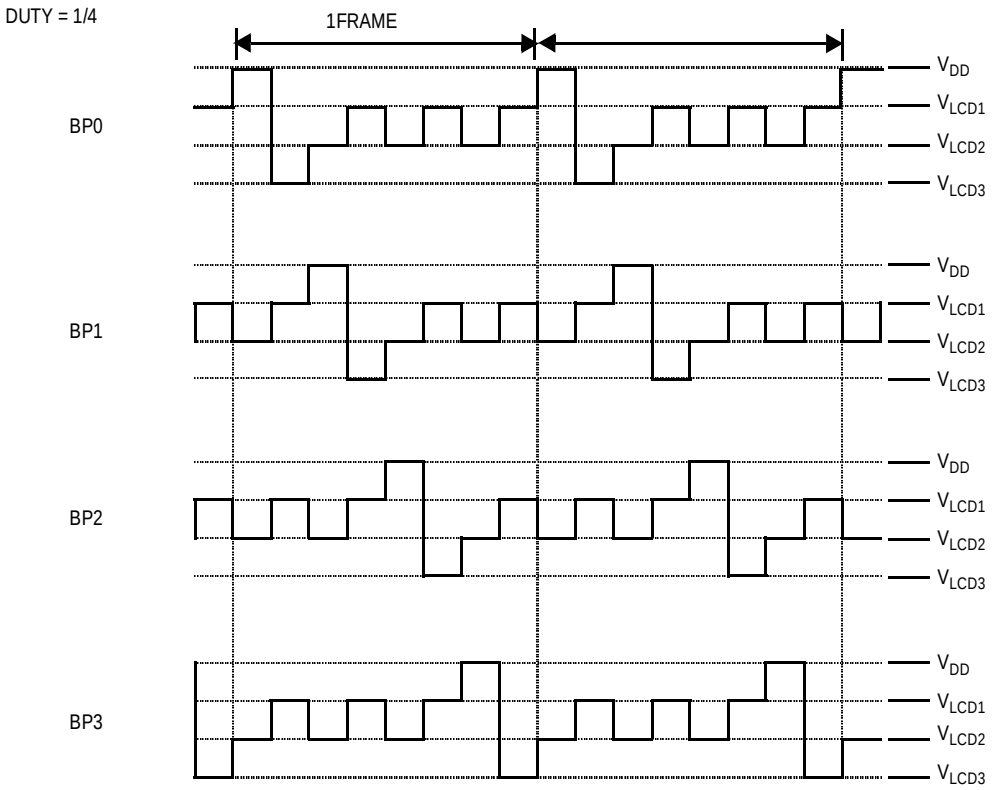
11.5.3 Frontplane Drivers

Pins FP0–FP24 are the output terminals for the frontplane drivers. These are connected to the frontplane of the LCD panel. Depending on the content of the data latch, the waveforms in **Figure 11-7** and **Figure 11-8** appear on the frontplane drivers.



NOTES:
1. BP3 is not used.
2. At 1/3 duty, 1FRAME has three times the cycle of LCD waveform base clock.

Figure 11-5. 1/3 Duty LCD Backplane Driver Waveforms



NOTE: The element which selects or does not select the BP waveforms is as follows.

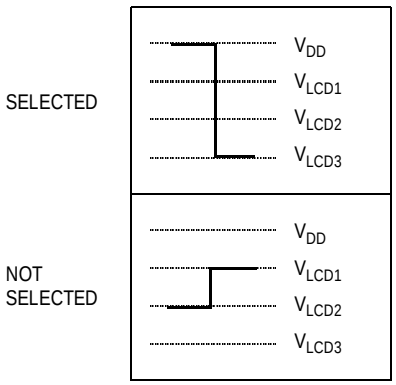


Figure 11-6. 1/4 Duty LCD Backplane Driver Waveforms

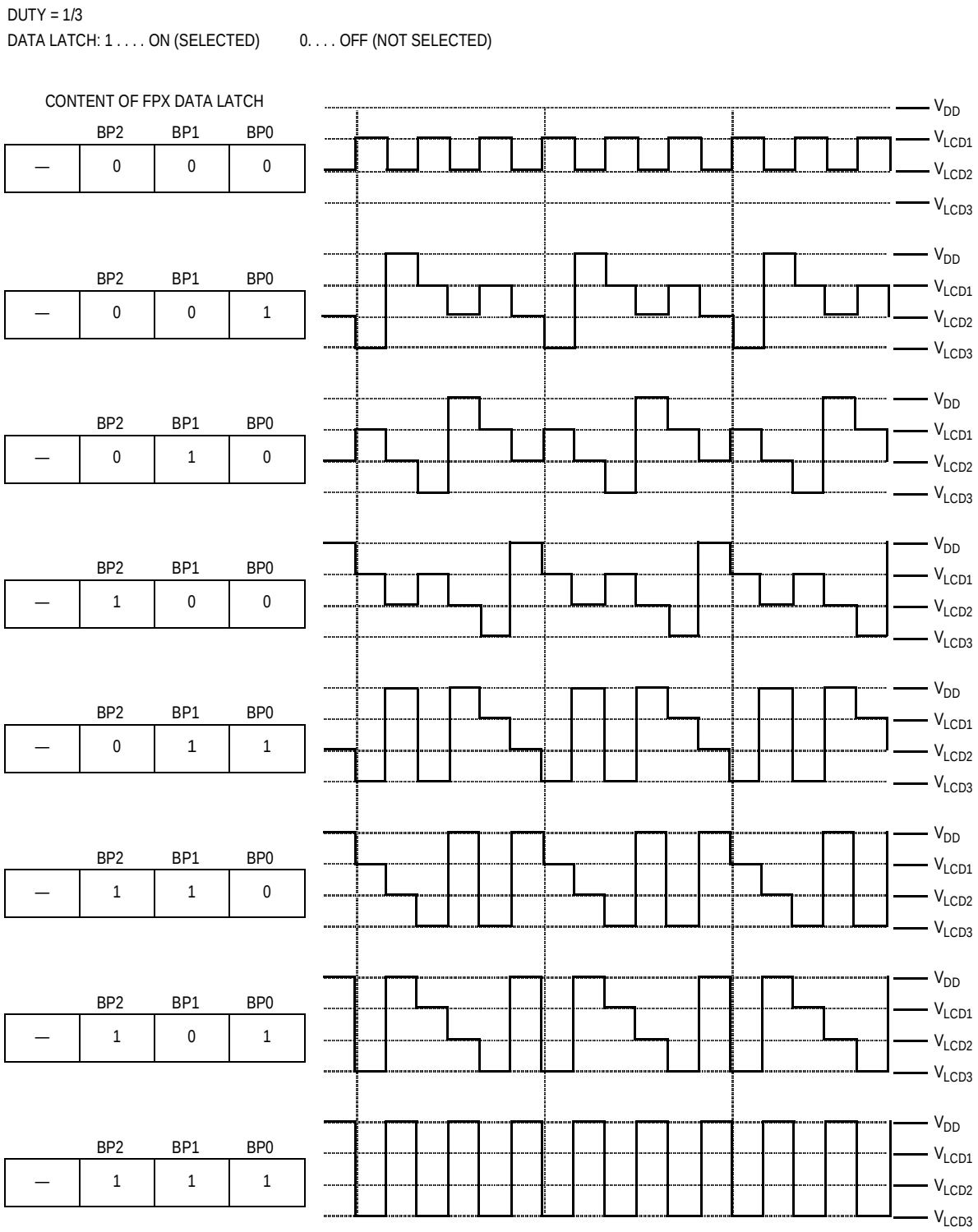
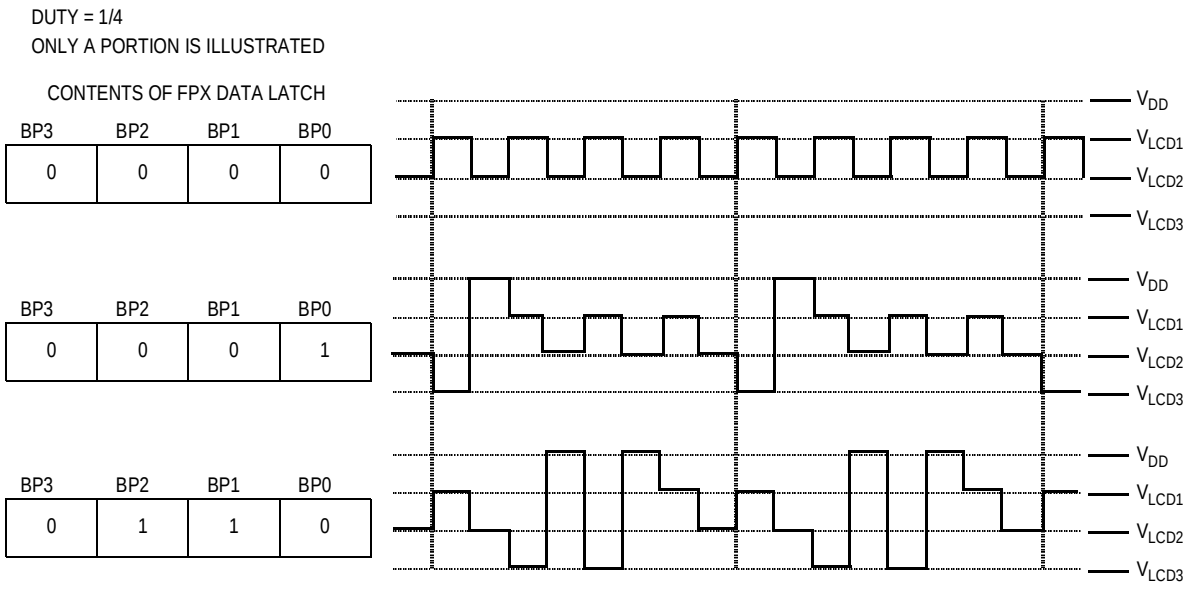


Figure 11-7. 1/3 Duty LCD Frontplane Driver Waveforms



NOTE: The fundamental elements which select or do not select the frontplane waveforms are as follows.

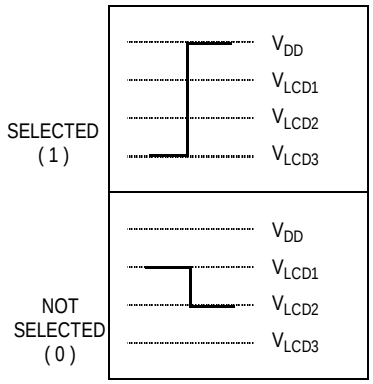


Figure 11-8. 1/4 Duty LCD Frontplane Driver Waveforms

11.6 LCD Connection and LCD Driver Operation

The connection between the MC68HC05L25 and the seven segments of the LCD panel is discussed in the following paragraphs.

Figure 11-9 illustrates a 1/3 duty example.

Pins BP0, BP1, BP2, FP0, FP1, and FP2 output the waveforms illustrated in **Figure 11-10.**

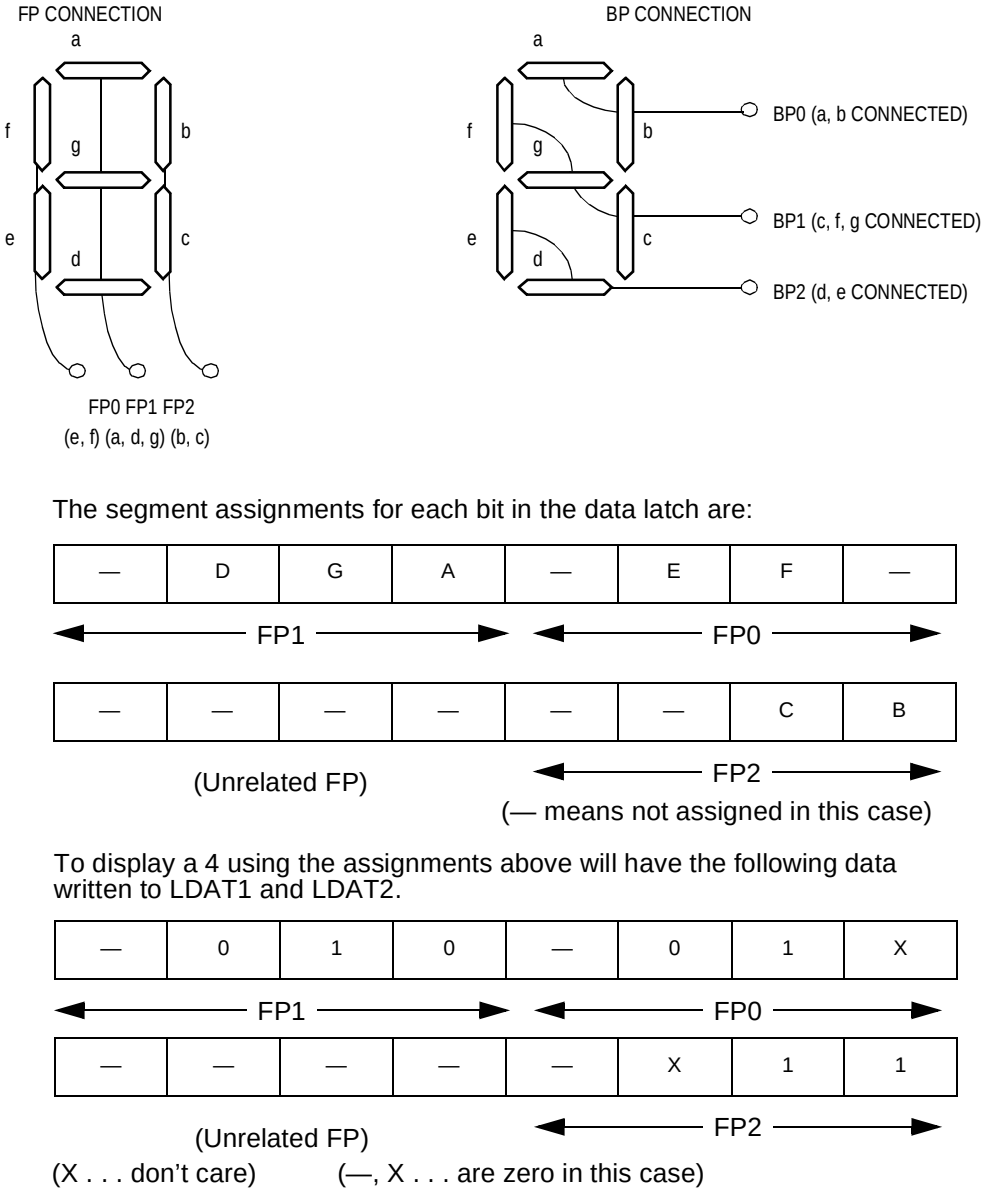


Figure 11-9. 1/3 Duty Example

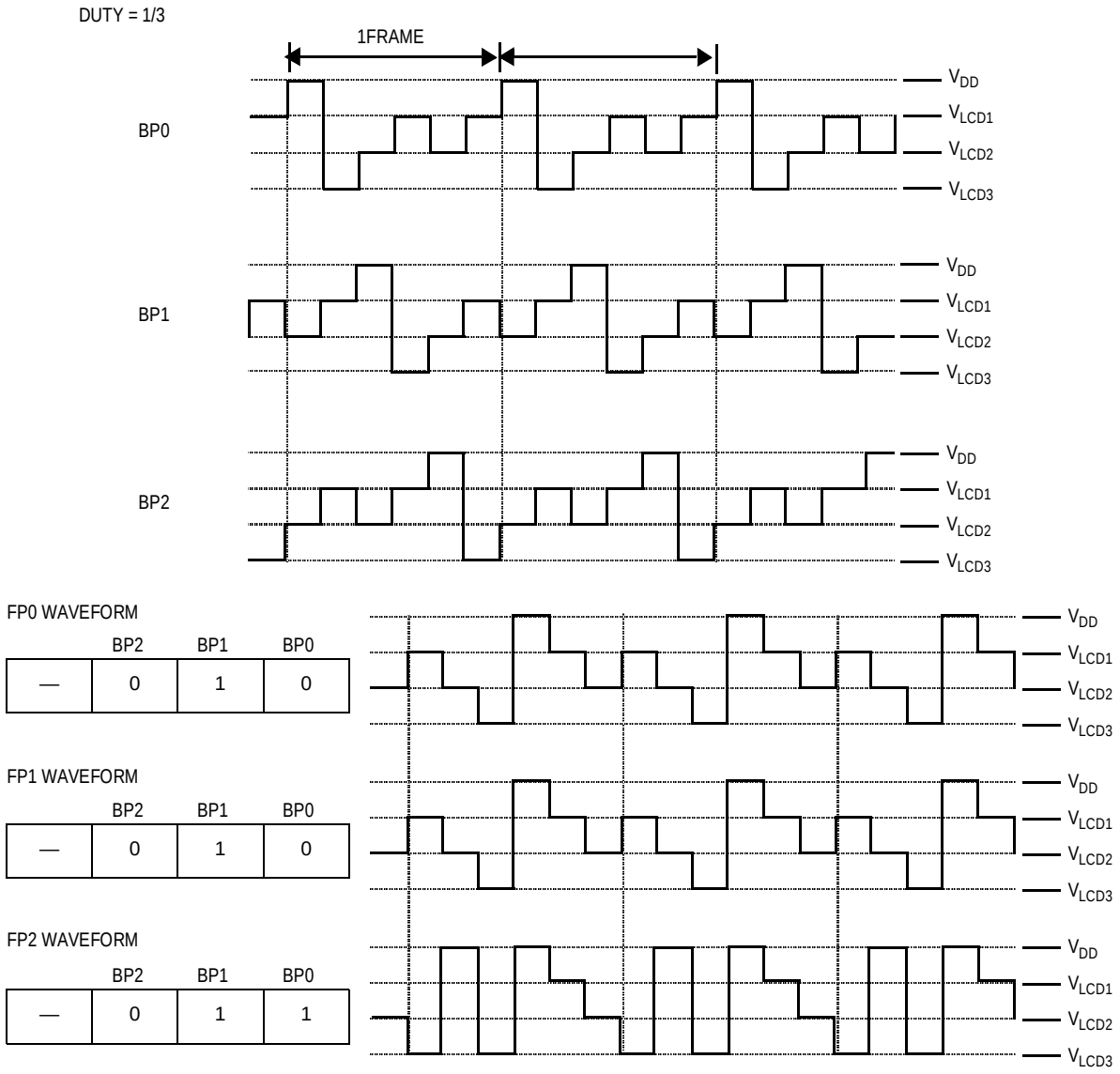


Figure 11-10. BP0–BP2 and FP0–FP2 Output Waveforms

The electric potential waveform for the F segment (between FP0 and BP1) is illustrated in **Figure 11-11**. As shown, the LCD ON voltage (V_{LCD}) of the AC waveform is attained, so the F segment will turn ON.

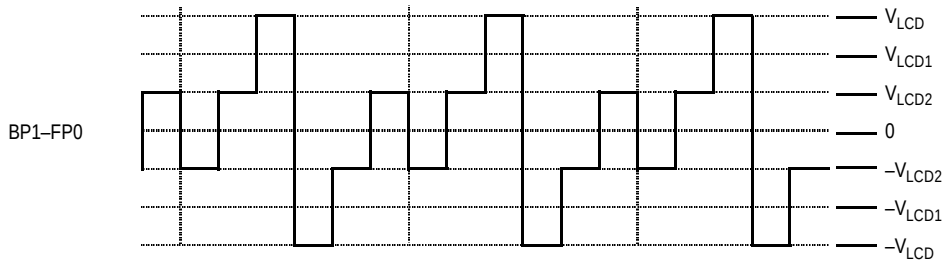


Figure 11-11. F Segment Potential Waveform

The electric potential waveform for the E segment (between FP0 and BP2) is illustrated in **Figure 11-12**. This segment is not turned ON.

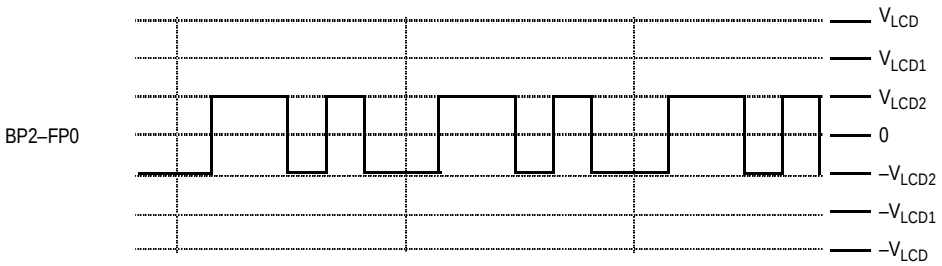


Figure 11-12. E Segment Electric Potential Waveform

The methods described will determine whether to turn ON or turn OFF the LCD segment. The waveform elements which select or do not select the BP and FP waveforms are shown in **Figure 11-13**.

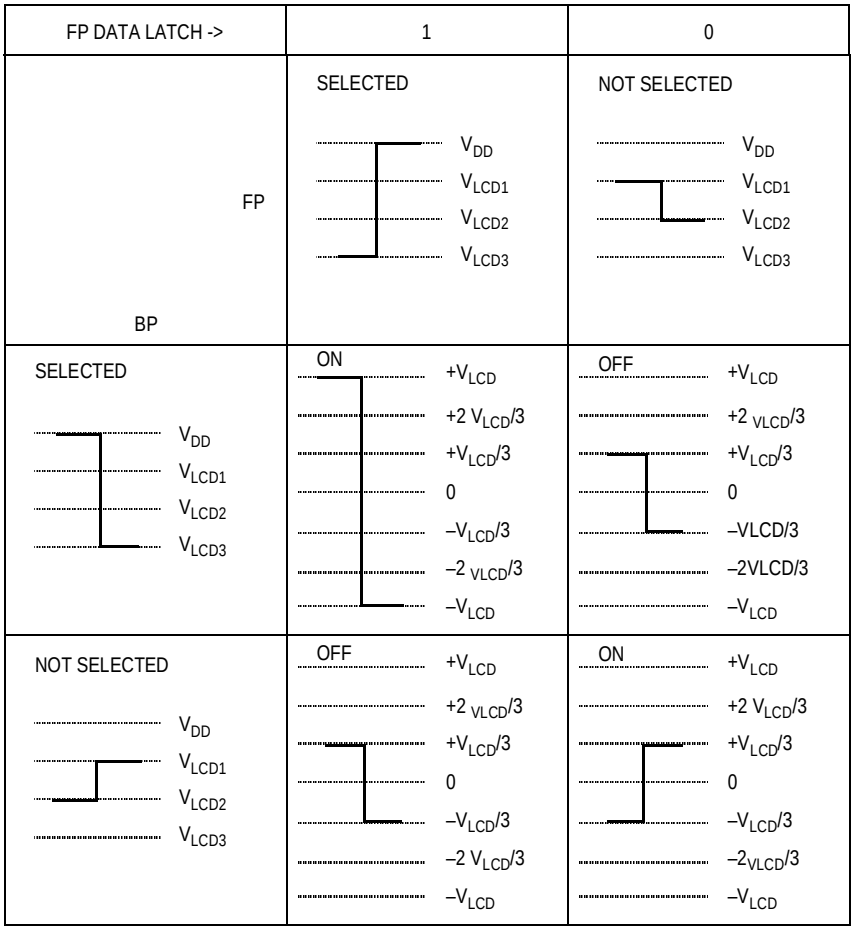


Figure 11-13. Waveform Elements

11.7 LCD Waveform Base Clock and LCD Cycle Frame

The clock which produces the LCD FP and BP output waveforms, the LCD waveform back clock, is generated from the time base module.

The frequency for the LCD waveform base clock can be changed by the time base control register.

11.7.1 Time Base Control Register 1

Address: $0010

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	TBCLK	0	LCLK	RMC4	RMC3	RMC2	RMC1	RMC0
Write:								
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 11-14. Time Base Control Register 1 (TBCR1)

LCLK — LCD Clock

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

When TBCLK = 0:

1 = XOSC divide by 128 is selected for the LCD clock

0 = XOSC divide by 64 is selected for the LCD clock

When TBCLK = 1:

1 = OSC divide by 16,384 is selected for the LCD clock

0 = OSC divide by 8192 is selected for the LCD clock

Table 11-2. LCD Waveform Base Clock Frequency

TBCR1		Divide Ratio	LCD Waveform Base Clock Frequency (Hz) (f_{XOSC} = 32.768 kHz)		
TBCLK	LCLK		OSC = 2.0 MHz	OSC = 4.0 MHz	OSC = 4.1943 MHz
0	0	XOSC ÷ 64	512	512	512
0	1	XOSC ÷ 128	256	256	256
1	0	OSC ÷ 8192	244	488	512
1	1	OSC ÷ 16384	122	244	256

11.7.2 LCD Cycle Frame

The LCD cycle frame with respect to the LCD waveform base clock and duty is

$$(\text{LCD Cycle Frame}) = \frac{1}{(\text{LCD Waveform Base Clock}) \cdot (\text{Duty})}$$

For example, given 1/3 duty and 256 Hz waveform base clock.

$$(\text{LCD Cycle Frame}) = \frac{1}{256 \cdot \frac{1}{3}}$$

$$= 11.72 \text{ ms}$$

11.8 Simplified LCD Schematic

A simplified schematic of the LCD driver is shown in **Figure 11-15**.

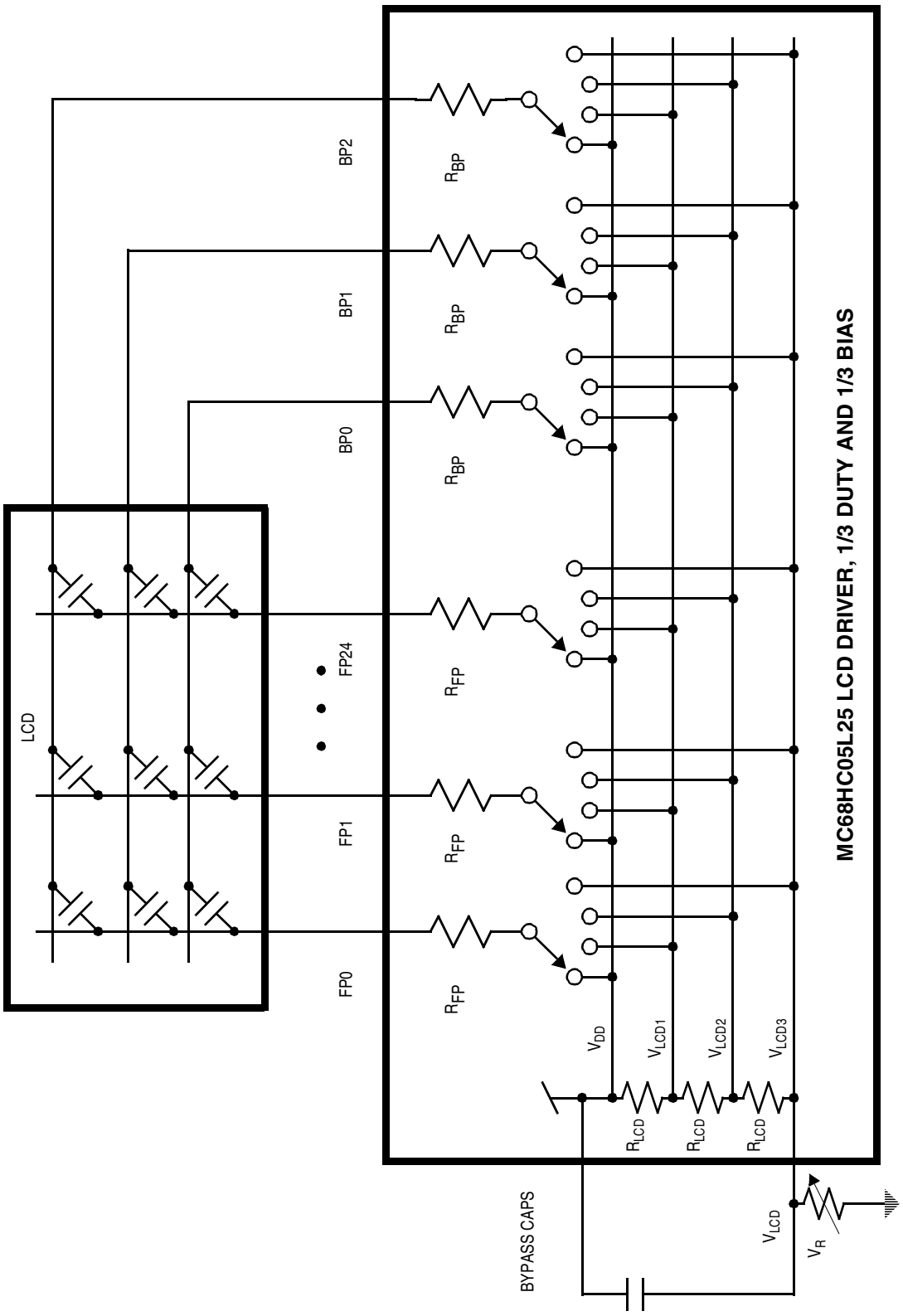


Figure 11-15. Simplified LCD Schematic

Section 12. Analog Subsystem

12.1 Contents

12.2 Introduction

The MC68HC05L25 includes a 2-channel, multiplexed input, 8-bit, successive approximation analog-to-digital (A/D) converter. The A/D subsystem shares its inputs with port A pins PA4 and PA5.

12.3 Analog Section

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

12.4 Ratiometric Conversion

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

12.4.1 V_{REFH}

The reference supply for the A/D converter is tied to V_{DD} internally. The low reference is tied to the V_{SS} pin internally.

12.4.2 Accuracy and Precision

The 8-bit conversion result is accurate to within ±1.5 LSB, including quantization; however, the accuracy of conversions is tested and guaranteed only with external oscillator operation at V_{DD} = 5 V.

12.5 Conversion Process

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

12.6 Digital Section

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

12.6.1 Conversion Times

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

12.6.2 Internal versus External Oscillator

If the MCU PH2 clock frequency is less than 1 MHz (2-MHz external oscillator), the internal RC oscillator (approximately 1.5 MHz) must be used for the A/D converter clock. The internal RC clock is selected by setting the ADRC bit in the ADSC register.

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

1. Since the internal RC oscillator is running asynchronously with respect to the PH2 clock, the conversion complete bit (CC) in register ADSC must be used to determine when a conversion sequence has been completed.
2. Electrical noise will slightly degrade the accuracy of the A/D converter. The A/D converter is synchronized to read voltages during the quiet period of the clock driving it. Since the internal and external clocks are not synchronized, the A/D converter occasionally will measure an input when the external clock is making a transition.
3. If the PH2 clock is 1 MHz or greater (for example, external oscillator 2 MHz or greater and SYS1–SYS0 = 0–0), the internal RC oscillator must be turned off and the external oscillator used as the conversion clock.

12.6.3 Multi-Channel Operation

An input multiplexer allows the A/D converter to select from one of two external analog signals. Port A pins PA4 and PA5 are shared with the inputs to the multiplexer.

NOTE: *Applying analog voltage to an A/D input pin that is not selected (used as a general-purpose digital I/O port) may result in excessive I_{DD}.*

12.7 A/D Subsystem Operation during Wait Modes

The A/D subsystem continues normal operation during wait modes. To decrease power consumption during wait, the ADON and ADRC bits in the A/D status and control register should be cleared if the A/D subsystem is not being used.

12.8 A/D Subsystem Operation during Stop Modes

When stop mode is enabled, execution of the STOP instruction will terminate all A/D subsystem functions. Any pending conversion is aborted. When the oscillator resumes operation upon leaving stop mode, a finite amount of time passes before the A/D subsystem stabilizes sufficiently to provide conversions at its rated accuracy. The delays built into the MC68HC05L25 when coming out of stop mode are sufficient for this purpose. No explicit delays need to be added to the application software.

12.9 A/D Status and Control Register

The ADSC register reports the completion of A/D conversion and provides control over oscillator selection, analog subsystem power, and input channel selection.

Address: $001E

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	CC	ADRC	ADON	0	0	CH2	CH1	CH0
Write:								
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 12-1. A/D Status and Control Register (ADSC)

CC — Conversion Complete

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

1 = A/D conversion sequence completed
0 = A/D subsystem is off or conversion is in progress

ADRC — RC Oscillator Control

When ADRC is set, the A/D subsystem operates from the internal RC oscillator instead of the PH2 clock. The RC oscillator requires a time, $t_{RCON,}$ to stabilize before accurate conversion results can be obtained. See **12.6.2 Internal versus External Oscillator** for more information.

1 = RC OSC on
0 = RC OSC off

ADON — A/D Subsystem On

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

1 = A/D subsystem enabled
0 = A/D subsystem disabled

Bits 4:3 — Reserved

These bits are not used and always read as zero.

CH2:CH0 — Channel Select Bits

Channel select bits CH2, CH1, and CH0 form a 3-bit field which is used to select an input to the A/D converter. Channels 0 and 1 correspond to port A input pins PA4 and PA5. Channels 4–6 are used for reference measurements. In single-chip mode, channels 2, 3, and 7 are reserved. If a conversion is attempted with channel 2, 3, or 7 selected, the result will be undefined. **Table 12-1** lists the inputs selected by bits CH0–CH2.

If the ADON bit is set, and an input from channel 0 or 1 is selected, the corresponding port A pin will not function as a digital port. If the port A data register is read when DDR = 0 while the A/D is on and one of the shared input channels is selected using bits CH0–CH2, the corresponding port A pin will read as a logic 0. If the DDR = 1, the port A data register will read the output latch value. The remaining port A pins will read normally. To digitally read a multiplexed port A pin as an input port, the A/D subsystem must be disabled (ADON = 0) or input channels 2–7 must be selected.

Table 12-1. A/D Multiplexer Input Channel Assignments

Channel	Signal
0	AD0 Port A Bit 4
1	AD1 Port A Bit 5
2	Reserved
3	Reserved
4	$V_{REFH} = V_{DD}$
5	$(V_{REFH} + V_{REFL})/2$
6	$V_{REFL} = V_{SS}$
7	Factory Test

12.10 A/D Conversion Data Register

This register contains the output of the A/D converter.

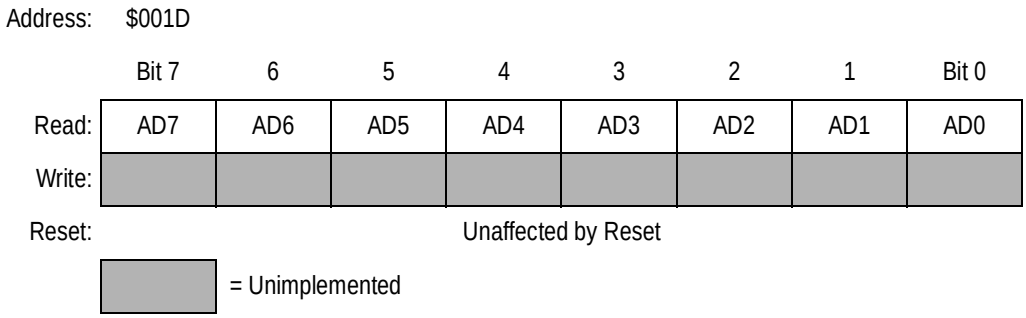


Figure 12-2. A/D Conversion Value Data Register (ADDR)

Section 13. Event Counter

13.1 Contents

13.2 Features

Event counter features include:

- Asynchronous Input up to 6 MHz
- Overflow Interrupt
- Event Count Complete Interrupt
- Variable Gate Generation
- Spike Filter
- Effective 18-Bit Resolution

13.3 Introduction

The event counter consists of a 16-bit counter externally driven from the event counter pin, with input gate generation and filtering circuitry. Average frequency measurements can be made over user specified intervals ranging from 4 to 60 ms (with 2-MHz bus clock). Measurements are continuously repeated at a user specified rate. A maskable and resetable event count complete interrupt and event counter overflow interrupt are available. Using the overflow interrupt, an effective 18-bit count can be achieved. See **Table 13-1**.

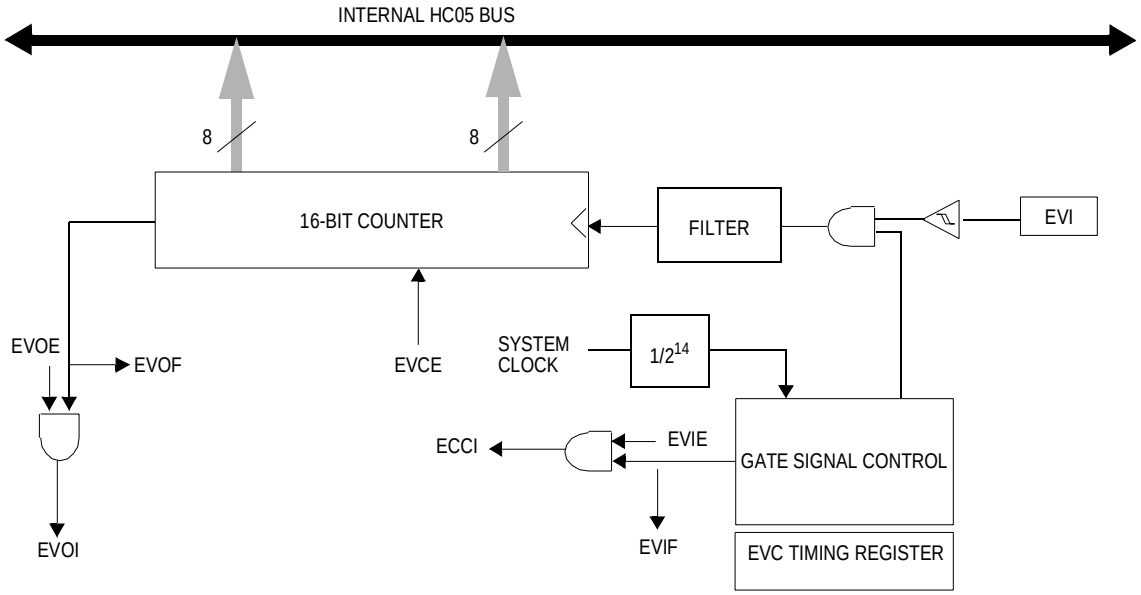


Figure 13-1. Event Counter Block Diagram

13.4 Event Counter Status/Control Register

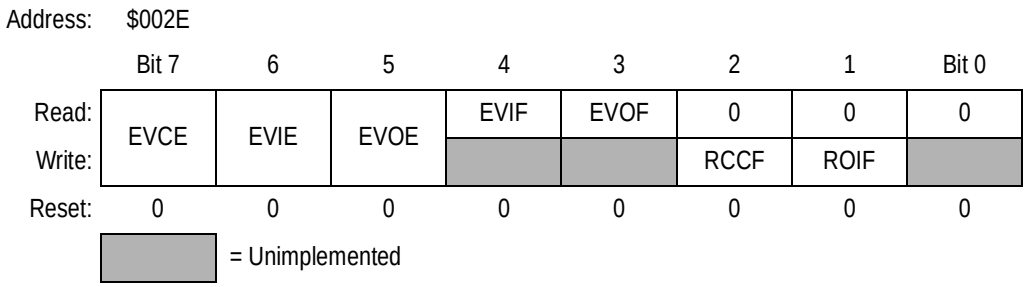


Figure 13-2. Event Counter Status/Control Register (EVSC)

EVCE — Event Counter Enable

1 = Event counter enabled
0 = Event counter disabled

EVIE — Event Counter Complete Interrupt Enable

1 = Event counter complete interrupt enabled
0 = Event counter complete interrupt disabled

EVOE — Event Counter Overflow Enable

1 = Event counter overflow interrupt enabled
0 = Event counter overflow interrupt disabled

EVIF — Event Counter Complete Interrupt Flag (read only)

1 = Flag set when gate delay time expires
0 = Flag cleared when logic 1 is written to ROIF

EVOF — Event Counter Overflow Flag (read only)

1 = Flag set when gate delay time expires
0 = Flag cleared when logic 1 is written to ROIF

RCCF — Reset Count Complete Interrupt Flag (write only)

When a logic 1 is written to this bit, EVIF is cleared. Always reads as zero.

ROIF — Reset Overflow Interrupt Flag (write only)

When a logic 1 is written to this bit, EVOF is cleared. Always reads as zero.

Bit 0 — Reserved

This bit is not used and always reads as zero.

13.5 Event Counter Timing Register

Address: $002F	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	WT3	WT2	WT1	WT0	MT3	MT2	MT1	MT0
Reset:	1	1	1	1	1	1	1	1

Figure 13-3. Event Counter Timing Register (EVTR)

This register controls generation of the gate signal which is used to control the input to the event counter. See **Figure 13-5**.

The value in the event counter timing register determines the length of the measurement and the length of the wait time between measurements. See **Table 13-1** and **Table 13-2**. The measurement time bits, MT3 through MT0, determine the length of time that the input gate on the EVI pin is open. During this time the gate

signal is a logic 1. The wait time bits, WT3 through WT0, determine the length of time that the gate signal is a logic 0. t_{gc} is the length of a unit count. The specification for t_{gc} is found in **Section 15. Electrical Specifications**.

After being enabled, EVCE = 1, the event counter will make measurements continuously. If the event counter timing register is written, the current measurement will be aborted, and a new measurement will be initiated.

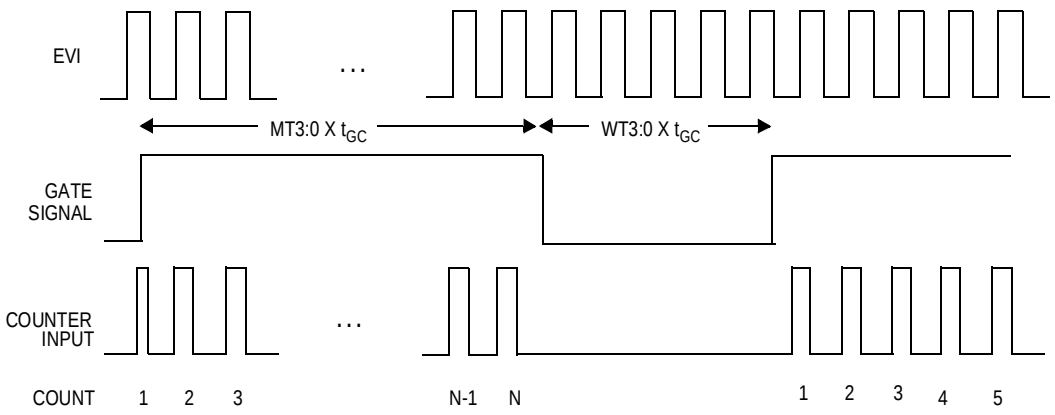


Figure 13-4. Event Counter Input Timing Example

The input to the event counter is the logical AND of the signal on the EVI pin and the internally generated gate signal. The rising edges of the counter input signal are used to generate the events that increment the counter. If the pulse width of the ANDed signal is less than that which the circuitry is capable of detecting, the narrow pulse will not be allowed to pass through the filter.

Table 13-1. Measurement Time Nibble

MT3:MT0	Measurement Time	MT3:MT0	Measurement Time
$X0	0 ms	$X8	31.250 ms
$X1	3.9063 ms	$X9	35.156 ms
$X2	7.8125 ms	$XA	39.063 ms
$X3	11.719 ms	$XB	42.969 ms
$X4	15.625 ms	$XC	46.875 ms
$X5	19.531 ms	$XD	50.781 ms
$X6	23.438 ms	$XE	54.688 ms
$X7	27.344 ms	$XF	58.593 ms

Table 13-2. Wait Time Nibble

WT3:WT0	Wait Time	WT3:WT0	Wait Time
$0X ms	0 ms	$8x	31.250 ms
$1X ms	3.9063 ms	$9x	35.156 ms
$2X ms	7.8125 ms	$Ax	39.063 ms
$3X ms	11.719 ms	$BX	42.969 ms
$4X ms	15.625 ms	$CX	46.875 ms
$5X ms	19.531 ms	$DX	50.781 ms
$6X ms	23.438 ms	$EX	54.688 ms
$7X ms	27.344 ms	$FX	58.593 ms

NOTE: SYS0 and SYS1 = 00, f_{osc} = 4.1943 MHz

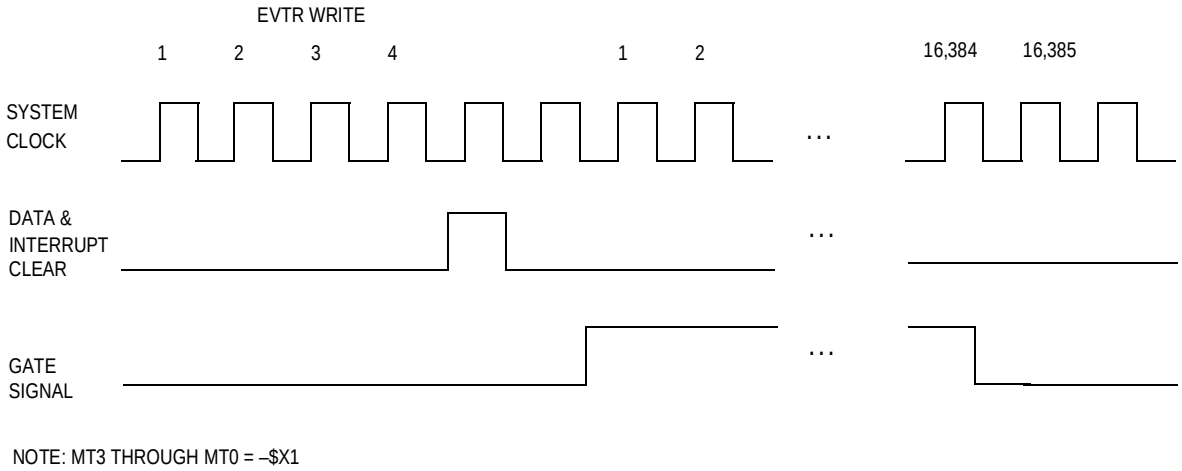


Figure 13-5. Event Counter Gate Signal Timing Example

The above example illustrates the relation of the gate signal to external oscillator clocks for the case of MT3–MT0 = $1.

The beginning of the gate signal can be caused by a write to the event counter timing register or expiration of the wait time. If the event counter timing register is written, the rising edge of the gate signal will occur on the fourth internal processor clock cycle of the write to the event counter timing register. The event counter data registers are cleared on the rising edge of the internal gate signal. The external gate signal rises two clock cycles later.

After MT3–MT0 x t_{gt}, the gate signal will rise, terminating the measurement time. The gate signal will be the same length for all successive measurements.

Unless the start of the gate delay signal and the event counter input signal are externally synchronized, the value of the least significant bit of the event counter data low register may arbitrarily change.

If the fast oscillator, OSC, is disabled, the event counter will not function properly.

13.6 Event Counter Interrupts

The event counter complete interrupt (ECCI) is generated at the falling edge of the gate signal. This interrupt indicates the presence of valid data in the event counter data registers. Since reading the data registers during the measurement time may give invalid results, the CPU must read the data registers before the rising edge of the gate signal. The event counter complete interrupt can be cleared by writing a one to RCCF. The event counter complete interrupt is cleared automatically at the beginning of each measurement.

The event counter overflow interrupt (EVOF) is generated if the count exceeds 65,535, the maximum value of the 16-bit event counter. This interrupt can be used to indicate an invalid measurement or to increase the resolution of the event counter, which will be described later. The event counter overflow interrupt can be cleared by writing a one to ROIF. The event counter overflow interrupt is cleared automatically at the beginning of each measurement.

If an overflow occurs (the counter increments beyond $FFFF), the event counter overflow flag (EVOF) will be set. If EVOE is set, an interrupt will be generated. Following an overflow, the event counter will increment from zero.

The resolution of the event counter can be increased by using the event counter overflow interrupt. If it is a count of more than 65,535 the maximum value of the 16-bit event counter is encountered, the event counter overflow interrupt service routine should note the number of “roll-overs” that occur. The overflow interrupt service routine should not clear the event counter interrupt. In this way, the user can be assured that the correct count has been recorded.

13.7 Event Counter During Wait Mode

The event counter continues to operate in wait mode. If EVOE is set and an event counter overflow interrupt occurs, the processor will exit wait mode. If EVIE is set and an event counter interrupt occurs, the processor will exit wait mode.

13.8 Event Counter During Stop Mode

In stop mode, the event counter is disabled.

13.9 Event Counter Data Registers

Address: $0030

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	BIT15	BIT14	BIT13	BIT12	BIT11	BIT10	BIT9	BIT8
Write:								
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 13-6. Event Counter Data High Register (EVDH)

Address: $0031

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
Write:								
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 13-7. Event Counter Data Low Register (EVDL)

These read-only registers are the basis of all event counter operations. While the gate signal is low, the value of the most recent event count will remain in the event counter data registers. After the event counter interrupt, the result of the event count can be read. If the event counter data registers are read while the gate signal is high, an incorrect value may result.

Section 14. Instruction Set

14.1 Contents

14.2 Introduction

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

14.3 Addressing Modes

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

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

14.3.1 Inherent

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

14.3.2 Immediate

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

14.3.3 Direct

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

14.3.4 Extended

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

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

14.3.5 Indexed, No Offset

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

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

14.3.6 Indexed, 8-Bit Offset

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

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

14.3.7 Indexed,16-Bit Offset

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

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

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

14.3.8 Relative

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

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

14.4 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

14.4.1 Register/Memory Instructions

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

Table 14-1. 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

14.4.2 Read-Modify-Write Instructions

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

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

Table 14-2. Read-Modify-Write Instructions

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

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

14.4.3 Jump/Branch Instructions

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

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

Table 14-3. 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

14.4.4 Bit Manipulation Instructions

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

Table 14-4. Bit Manipulation Instructions

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

14.4.5 Control Instructions

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

Table 14-5. 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

14.5 Instruction Set Summary

Table 14-6. Instruction Set Summary

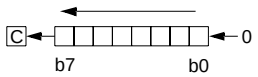
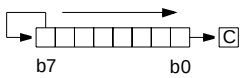
Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			H	I	N	Z	C				
ADC #*opr* ADC *opr* ADC *opr* ADC *opr*,X ADC *opr*,X ADC ,X	Add with Carry	A ← (A) + (M) + (C)	↕	—	↕	↕	↕	IMM DIR EXT IX2 IX1 IX	A9 B9 C9 D9 E9 F9	ii dd hh ll ee ff ff	2 3 4 5 4 3
ADD #*opr* ADD *opr*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

Table 14-6. Instruction Set Summary (Continued)

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

Table 14-6. Instruction Set Summary (Continued)

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

Table 14-6. Instruction Set Summary (Continued)

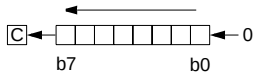
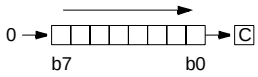
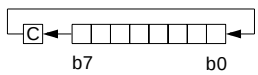
Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			H	I	N	Z	C				
JSR *opr* JSR *opr* JSR *opr*,X JSR *opr*,X JSR ,X	Jump to Subroutine	PC ← (PC) + n (n = 1, 2, or 3) Push (PCL); SP ← (SP) – 1 Push (PCH); SP ← (SP) – 1 PC ← Effective Address	—	—	—	—	—	DIR EXT IX2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

Table 14-6. Instruction Set Summary (Continued)

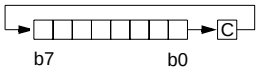
Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			H	I	N	Z	C				
ROR *opr* RORA RORX ROR *opr*,X ROR ,X	Rotate Byte Right through Carry Bit		—	—	↕	↕	↕	DIR INH INH IX1 IX	36 46 56 66 76	dd ff 	5 3 3 6 5
RSP	Reset Stack Pointer	SP ← $00FF	—	—	—	—	—	INH	9C		2
RTI	Return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

Table 14-6. Instruction Set Summary (Continued)

Source Form	Operation	Description	Effect on CCR					Address Mode	Opcode	Operand	Cycles
			H	I	N	Z	C				
TST *opr* TSTA TSTX TST *opr*,X TST ,X	Test Memory Byte for Negative or Zero	(M) – $00	—	—	↕	↕	—	DIR INH INH IX1 IX	3D 4D 5D 6D 7D	dd ff 	4 3 3 5 4
TXA	Transfer Index Register to Accumulator	A ← (X)	—	—	—	—	—	INH	9F		2
WAIT	Stop CPU Clock and Enable Interrupts		—	↕	—	—	—	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 14-7. Opcode Map

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

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

LSB of Opcode in Hexadecimal

MSB / LSB	0
0	5 BRSET0 3 DIR

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

Section 15. Electrical Specifications

15.1 Contents

15.2 Introduction

This section contains electrical specifications for the MC68HC05L25.

15.3 Maximum Ratings

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

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

Rating	Symbol	Value	Unit
Supply Voltage	V_{DD}	–0.3 to +7.0	V
Input Voltage (Normal Digital Level)	V_{IN}	V_{SS} –0.3 to V_{DD} +0.3	V
Current Drain Per Pin Excluding V_{DD} and V_{SS}	I	25	mA
Storage Temperature Range	T_{STG}	–65 to +150	°C

NOTE: Voltages referenced to V_{SS}

NOTE: *This device is not guaranteed to operate properly at the maximum ratings. Refer to* ***15.7 DC Electrical Characteristics (V_{DD} = 3.3 V)*** *and* ***15.8 DC Electrical Characteristics (V_{DD} = 5.0 V)*** *for guaranteed operating conditions.*

15.4 Operating Range

Characteristic	Symbol	Value	Unit
Operating Temperature Range MC68HC05L25 (Standard)	T_A	T_L to T_H 0 to +70	°C

15.5 Thermal Characteristics

Characteristic	Symbol	Value	Unit
Thermal Resistance MC68HC05L25FA (48-pin VQFP) MC68HC05L25PB (52-pin TQFP)	θ_{JA}	 195 126	°C/W

15.6 DC Operating Characteristics

Characteristic	Symbol	Min	Max	Unit
Operating Voltage Internal Operating Frequency f_{OP} = 1.0 MHz Internal Operating Frequency f_{OP} = 2.1 MHz	V_{DD}	 3.0 4.5	 5.5 5.5	V

NOTE: V_{SS} = 0 Vdc, T_A = 0 °C to +7 0°C, unless otherwise noted

15.7 DC Electrical Characteristics (V_{DD} = 3.3 V)

Characteristic	Symbol	Min	Typ	Max	Unit
Output Voltage I_{Load} = 10.0 µA	V_{OL} V_{OH}	— V_{DD} –0.1	— —	0.1 —	V
Output High Voltage (I_{Load} = –0.4 mA) PA4:PA7, PB0:PB7, and PC0:PC1	V_{OH}	V_{DD} –0.8	—	—	V
Output Low Voltage (I_{Load} = 0.8 mA) PA0:PA7, PB0:PB7, and PC0:PC3 (I_{Load} = 20 mA) PA0:PA3	V_{OL}	— —	— —	0.4 1.0	V
Input High Voltage PA0:PA7, PB0:PB7, and PC0:PC3, $\overline{IRQ}$, $\overline{RESET}$, and XOSC1	V_{IH}	0.7 × V_{DD}	—	V_{DD}	V
Input Low Voltage PA0:PA7, PB0:PB7, and PC0:PC3, $\overline{IRQ}$, $\overline{RESET}$, and XOSC1	V_{IL}	V_{SS}	—	0.3 × V_{DD}	V
Supply Current (See Notes) Run (f_{OP} = 1.0 MHz) Wait (f_{OP} = 1.0 MHz) Stop With Time Base Running; LCD Off With All Clocks Halted	I_{DD}	 — — — —	 1.5 1.0 8.0 2.0	 8.0 5.0 20 10	 mA mA µA µA
I/O Ports Hi-Z Leakage Current (Without Individual Pullup Activated) PA0:PA7, PB0:PB7, and PC0:PC3	I_{IL}	—	—	±10	µA
Pullup Current (With Individual Pullup Activated) PA0:PA7 and PB0:PB7 PC0:PC3	I_{IL}	6 20	20 60	60 180	µA
Input Current $\overline{RESET}$, $\overline{IRQ}$, and XOSC1	I_{IN}	—	—	±1	µA
Capacitance Ports (As Input or Output) $\overline{RESET}$, $\overline{IRQ}$	 C_{OUT} C_{IN}	 — —	 — —	 12 8	 pF
Crystal Oscillator Mode Feedback Resistor OSC1 to OSC2 XOSC1 to XOSC2	 R_{OF} R_{XOF}	 1 2.7	 2 5.5	 3 8.2	 MΩ
Crystal Oscillator Mode Damping Resistor on XOSC2	R_{XOD}	160	320	480	kΩ
$\overline{RESET}$ Pin Pullup Resistance	R_{RST}	20	60	120	kΩ

NOTES:
1. V_{DD} = 3.3 Vdc ± 0.3 Vdc, V_{SS} = 0 Vdc, T_A = 0 °C to +70 °C, unless otherwise noted
2. All values shown reflect average measurements.
3. Typical values at midpoint of voltage range, 25 °C only
4. Wait I_{DD}: Only time base active
5. Run (Operating) I_{DD}, wait I_{DD}: Measured using external square wave clock source to OSC1 (f_{OP} = 1.0 MHz); all inputs 0.2 Vdc from rail; no DC loads; less than 50 pF on all outputs; C_L = 20 pF on OSC2
6. Wait and stop I_{DD}: All ports configured as inputs, V_{IL} = 0.2 Vdc, V_{IH} = V_{DD} –0.2 Vdc
7. Wait and stop I_{DD} are affected linearly by the OSC2, XOSC2 capacitance.

15.8 DC Electrical Characteristics (V_{DD} = 5.0 V)

Characteristic	Symbol	Min	Typ	Max	Unit
Output Voltage I_{Load} = 10.0 µA	V_{OL} V_{OH}	— V_{DD} –0.1	— —	0.1 —	V
Output High Voltage (I_{Load} = –0.8 mA) PA4:PA7, PB0:PB7, and PC0:PC1	V_{OH}	V_{DD} –0.8	—	—	V
Output Low Voltage (I_{Load} = 0.8 mA) PA0:PA7, PB0:PB7, and PC0:PC3 (I_{Load} = 20 mA) PA0:PA3	V_{OL}	— —	— —	0.4 0.8	V
Input High Voltage PA0:PA7, PB0:PB7, and PC0:PC3, $\overline{IRQ}$, $\overline{RESET}$, and XOSC1	V_{IH}	0.7 × V_{DD}	—	V_{DD}	V
Input Low Voltage PA0:PA7, PB0:PB7, and PC0:PC3, $\overline{IRQ}$, $\overline{RESET}$, and XOSC1	V_{IL}	V_{SS}	—	0.3 × V_{DD}	V
Supply Current (See Notes) Run (f_{OP} = 2.1 MHz) Wait (f_{OP} = 2.1 MHz) Stop With Time Base Running; LCD Off With All Clocks Halted	I_{DD}	 — — — —	 3.0 2.0 17 3.0	 12 6.0 40 15	 mA mA µA µA
I/O Ports Hi-Z Leakage Current (Without Individual Pullup Activated) PA0:PA7, PB0:PB7, and PC0:PC3	I_{IL}	—	—	±10	µA
Pullup Current (With Individual Pullup Activated) PA0:PA7 and PB0:PB7 PC0:PC3	I_{IH}	10 20	30 60	90 180	µA
Input Current $\overline{RESET}$, $\overline{IRQ}$, and XOSC1	I_{IN}	—	—	±1	µA
Capacitance Ports (As Input or Output) $\overline{RESET}$, $\overline{IRQ}$	C_{OUT} C_{IN}	— —	— —	12 8	pF
Crystal Oscillator Mode Feedback Resistor OSC1 to OSC2 XOSC1 to XOSC2	R_{OF} R_{XOF}	1 2.7	2 5.5	3 8.2	MΩ
Crystal Oscillator Mode Damping Resistor on XOSC2	R_{XOD}	160	320	480	kΩ
$\overline{RESET}$ Pin Pullup Resistance	R_{RST}	10	32	70	kΩ

NOTES:
1. V_{DD} = 5.0 Vdc ± 0.5 Vdc, V_{SS} = 0 Vdc, T_A = 0 °C to +70 °C, unless otherwise noted
2. All values shown reflect average measurements.
3. Typical values at midpoint of voltage range, 25 °C only
4. Wait I_{DD}: Only time base active
5. Run (Operating) I_{DD}, wait I_{DD}: Measured using external square wave clock source to OSC1 (f_{OP} = 2.1 MHz); all inputs 0.2 Vdc from rail; no DC loads; less than 50 pF on all outputs; C_L = 20 pF on OSC2
6. Wait, stop I_{DD}: All ports configured as inputs, V_{IL} = 0.2 Vdc, V_{IH} = V_{DD} –0.2 Vdc
7. Wait, stop I_{DD} is affected linearly by the OSC2, XOSC2 capacitance.

15.9 LCD DC Electrical Characteristics (V_{DD} = 3.0 V, V_{LCD} = 0 V)

Characteristic	Symbol	Min	Typ	Max	Unit
LCD Bias Resistance Default Low-Current Mode	R_{LCD}	80 430	160 860	240 1300	kΩ
Output Current, Backplanes and Frontplanes High-Current State (Default) V_O = 2.85 V V_O = 1.85 V V_O = 1.15 V V_O = 0.15 V	I_{BH}	 –400 –10 1 20	 –260 –2.8 4 50	 –120 –1 10 80	µA
Output Current, Backplanes and Frontplanes Low-Current State V_O = 2.85 V V_O = 1.85 V V_O = 1.15 V V_O = 0.15 V	I_{BL}	 –400 –1 0.1 20	 –260 –0.2 0.6 50	 –120 –0.05 1 80	µA

NOTES:
1. All values shown reflect average measurements. These values are design targets and not characterization results.
2. If the FC option is selected, for time 1/(32 x f_{LCD}) after the backplane or frontplane waveform changes to a new voltage level, the circuit is maintained in the high-current state to allow the load capacitors to charge quickly. Then the circuit is returned to the low-current state until the next voltage level change occurs.

15.10 LCD DC Electrical Characteristics (V_{DD} = 5.0 V, V_{LCD} = 2.0 V)

Characteristic	Symbol	Min	Typ	Max	Unit
LCD Bias Resistance Default Low-Current Mode	R_{LCD}	 80 430	 160 860	 240 1300	kΩ
Output Current, Backplanes and Frontplanes High-Current State (Default) V_O = 4.85 V V_O = 3.85 V V_O = 3.15 V V_O = 2.15 V	I_{BH}	 –660 –10 1 20	 –440 –3.8 4.2 50	 –220 –1 10 80	µA
Output Current, Backplanes and Frontplanes Low-Current State V_O = 4.85 V V_O = 3.85 V V_O = 3.15 V V_O = 2.15 V	I_{BL}	 –660 –1 0.2 20	 –440 –0.2 1.2 50	 –220 –0.05 3 30	µA

NOTES:
1. All values shown reflect average measurements. These values are design targets and not characterization results.
2. If the FC option is selected, for time 1/(32 x f_{LCD}) after the backplane or frontplane waveform changes to a new voltage level, the circuit is maintained in the high-current state to allow the load capacitors to charge quickly. Then the circuit is returned to the low-current state until the next voltage level change occurs.

15.11 A/D Converter Characteristics

Characteristic	Symbol	Min	Max	Units	Comments
Resolution	—	8		Bit	
Absolute Accuracy V_{DD} = 4.5 to 5.5 V V_{DD} = 3.0 to 3.6 V	—	— —	± 1.5 ± 3	LSB LSB	Error includes quantization.
Conversion Range	—	V_{SS}	V_{DD}	V	
ADC On Current Stabilization TIme	t_{ADON}	—	100	µs	
RC Oscillator Stabilization Time	t_{RCON}	—	5	µs	
Conversion Time (Includes Sampling Time) External Clock (MCU System Clock) Internal RC Oscillator (ADRC = 1)	—	32 —	32 32	t_{AD} µs	$t_{AD} = t_{cyc}$ if clock source equals MCU.
Monotonicity	—	Inherent Within Total Error			
Zero Input Reading	—	00	01	Hex	V_{IN} = 0 V
Ratiometric Reading	—	FF	FF	Hex	$V_{IN} = V_{DD}$
Sample Acquisition Time External Clock (MCU System Clock) Internal RC Oscillator (ADRC = 1)	—	12 —	12 12	t_{AD} µs	Source impedances greater than 10 kΩ adversely affect internal RC charging time during input sampling. $t_{AD} = t_{cyc}$ if clock source equals MCU.
Input Capacitance	—	—	8	pF	
Analog Input Voltage	—	V_{SS}	V_{DD}	V	
Input Leakage AD0 and AD1	—	—	± 400	nA	The external system error caused by input leakage current is approximately equal to the product of R source and input current.

NOTES:

1. V_{DD} = 5.0 Vdc ± 10%, V_{SS} = 0 Vdc, T_A = 0 °C to +70 °C, unless otherwise noted

15.12 Control Timing (V_{DD} = 3.3 V)

Characteristic	Symbol	Min	Typ	Max	Unit
Frequency of Operation OSC Crystal Oscillator Option XOSC Crystal Oscillator Option External Clock Source	 f_{OSC} f_{XOSC} f_{OSC}	 — — dc	 — 32.768 —	 2.0 — 2.0	 MHz kHz MHz
Internal Operating Frequency Crystal Oscillator ($f_{osc} \div 2$) External Clock ($f_{osc} \div 2$)	f_{OP}	 — dc	 — —	 1.0 1.0	MHz
Cycle Time ($1 \div f_{OP}$)	t_{CYC}	1 000	—	—	ns
Crystal Oscillator Startup Time (Crystal Oscillator Option)	t_{OXON}	—	—	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}	250	—	—	ns
$\overline{\text{IRQ}}$ Interrupt Pulse Period	t_{ILIL}	Note 2	—	—	t_{CYC}
OSC1 Pulse Width	t	200	—	—	ns
Event Counter Gate Count	t_{GC}	16,384	16,384	16,384	f_{OSC}

NOTES:
1. V_{DD} = 3.3 Vdc ± 0.3 Vdc, V_{SS} = 0 Vdc, T_A = 0 °C to +70 °C, unless otherwise noted
2. 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 21 t_{CYC}.

15.13 Control Timing (V_{DD} = 5.0 V)

Characteristic	Symbol	Min	Typ	Max	Unit
Frequency of Operation OSC Crystal Oscillator Option XOSC Crystal Oscillator Option External Clock Source	 f_{OSC} f_{XOSC} f_{OSC}	 — — dc	 — 32.768 —	 4.2 — 4.2	 MHz kHz MHz
Internal Operating Frequency Crystal Oscillator ($f_{osc} \div 2$) External Clock ($f_{osc} \div 2$)	f_{OP}	 — dc	 — —	 2.1 2.1	 MHz
Cycle Time ($1 \div f_{OP}$)	t_{CYC}	480	—	—	ns
Crystal Oscillator Startup Time (Crystal Oscillator Option)	t_{OXON}	—	—	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 2	—	—	t_{CYC}
OSC1 Pulse Width	t	90	—	—	ns
Event Counter Gate Count	t_{GC}	16,384	16,384	16,384	f_{OSC}

NOTES:
1. V_{DD} = 5.0 Vdc ± 0.5 Vdc, V_{SS} = 0 Vdc, T_A = 0 °C to +70 °C, unless otherwise noted
2. 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 21 t_{CYC}.

Section 16. Mechanical Specifications

The MC68HC05L25 is available in the following packages:

- 48-pin quad flat pack (VQFP)
- 52-pin thin quad flat pack (TQFP)

Package specifications for the MC68HC05L25 were not available at the time of this publication. Contact your local Motorola sales office for the latest information.

Mechanical Specifications

Section 17. Ordering Information

17.1 Contents

17.2 Introduction

This section contains instructions for ordering custom-masked ROM MCUs.

17.3 MCU Ordering Forms

To initiate an order for a ROM-based MCU, first obtain the current ordering form for the MCU from a Motorola representative. Submit the following items when ordering MCUs:

- A current MCU ordering form that is **completely filled out** (Contact your Motorola sales office for assistance.)
- A copy of the customer specification if the customer specification deviates from the Motorola specification for the MCU
- Customer's application program on one of the media listed in **17.4 Application Program Media**

17.4 Application Program Media

Please deliver the application program to Motorola in one of the following media:

- Macintosh®[1] 3 1/2-inch diskette (double-sided 800 K or double-sided high-density 1.4 M)
- MS-DOS®[2] or PC-DOS™[3] 3 1/2-inch diskette (double-sided 720 K or double-sided high-density 1.44 M)
- MS-DOS® or PC-DOS™ 5 1/4-inch diskette (double-sided double-density 360 K or double-sided high-density 1.2 M)

Use positive logic for data and addresses.

When submitting the application program on a diskette, clearly label the diskette with the following information:

- Customer name
- Customer part number
- Project or product name
- File name of object code
- Date
- Name of operating system that formatted diskette
- Formatted capacity of diskette

On diskettes, the application program must be in Motorola's S-record format (S1 and S9 records), a character-based object file format generated by M6805 cross assemblers and linkers.

NOTE: *Begin the application program at the first user ROM location. Program addresses must correspond exactly to the available on-chip user ROM addresses as shown in the memory map. Write $00 in all non-user ROM locations or leave all non-user ROM locations blank. Refer to the current*

1. Macintosh is a registered trademark of Apple Computer, Inc.
2. MS-DOS is a registered trademark of Microsoft Corporation.
3. PC-DOS is a trademark of International Business Machines Corporation.

MCU ordering form for additional requirements. Motorola may request pattern re-submission if non-user areas contain any non-zero code.

If the memory map has two user ROM areas with the same addresses, then write the two areas in separate files on the diskette. Label the diskette with both filenames.

In addition to the object code, a file containing the source code can be included. Motorola keeps this code confidential and uses it only to expedite ROM pattern generation in case of any difficulty with the object code. Label the diskette with the filename of the source code.

17.5 ROM Program Verification

The primary use for the on-chip ROM is to hold the customer's application program. The customer develops and debugs the application program and then submits the MCU order along with the application program.

Motorola inputs the customer's application program code into a computer program that generates a listing verify file. The listing verify file represents the memory map of the MCU. The listing verify file contains the user ROM code and may also contain non-user ROM code, such as self-check code. Motorola sends the customer a computer printout of the listing verify file along with a listing verify form.

To aid the customer in checking the listing verify file, Motorola will program the listing verify file into customer-supplied blank preformatted Macintosh or DOS disks. All original pattern media are filed for contractual purposes and are not returned.

Check the listing verify file thoroughly, then complete and sign the listing verify form and return the listing verify form to Motorola. The signed listing verify form constitutes the contractual agreement for the creation of the custom mask.

17.6 ROM Verification Units (RVUs)

After receiving the signed listing verify form, Motorola manufactures a custom photographic mask. The mask contains the customer's application program and is used to process silicon wafers. The application program cannot be changed after the manufacture of the mask begins. Motorola then produces 10 MCUs, called RVUs, and sends the RVUs to the customer. RVUs are usually packaged in unmarked ceramic and tested to 5 Vdc at room temperature. RVUs are not tested to environmental extremes because their sole purpose is to demonstrate that the customer's user ROM pattern was properly implemented. The 10 RVUs are free of charge with the minimum order quantity. These units are not to be used for qualification or production. RVUs are not guaranteed by Motorola Quality Assurance.

17.7 MC Order Numbers

The following table shows the MC order numbers for the available package types.

MC Order Number	Operating Temperature Range
MC68HC05L25FA	–0° to 70°C
MC68HC05L25PB	–0° to 70°C

NOTES:
FA = 48-pin quad flat pack (VQFP)
PB = 52-pin thin quad flat pack (TQFP)

HOW TO REACH US:

USA/EUROPE/LOCATIONS NOT LISTED:

Motorola Literature Distribution;
P.O. Box 5405, Denver, Colorado 80217
1-303-675-2140 or 1-800-441-2447

JAPAN:

Motorola Japan Ltd.; SPS, Technical Information Center,
3-20-1, Minami-Azabu Minato-ku, Tokyo 106-8573 Japan
81-3-3440-3569

ASIA/PACIFIC:

Motorola Semiconductors H.K. Ltd.;
Silicon Harbour Centre, 2 Dai King Street,
Tai Po Industrial Estate, Tai Po, N.T., Hong Kong
852-26668334

TECHNICAL INFORMATION CENTER:

1-800-521-6274

HOME PAGE:

http://www.motorola.com/semiconductors

MOTOROLA

MC68HC05L25/D

Freescale Semiconductor, Inc.