HC05C0GRS/D
REV 1.2

68HC05C0

SPECIFICATION
(General Release)

CSIC System Design Group
Austin, Texas

TABLE OF CONTENTS

SECTION 1 INTRODUCTION 1

1.1 GENERAL 1
1.2 FEATURES 1
1.3 MASK OPTIONS 2
1.4 SIGNAL DESCRIPTION 3
1.4.1 VDD AND VSS 3
1.4.2 $\overline{\text{IRQ}}$ 3
1.4.3 OSC1, OSC2 3
1.4.4 $\overline{\text{RESET}}$ 3
1.4.5 $\overline{\text{A15}}$,A14-A8 3
1.4.6 AD7:0 3
1.4.7 AS (ADDRESS STROBE)/ $\overline{\text{CS2}}$ 3
1.4.8 $\overline{\text{RD}}$ 4
1.4.9 $\overline{\text{WR}}$ 4
1.4.10 A7-A0/PD7-PD0 4
1.4.11 PB4-PB0/SCK,TDO,RDI,TCAP,TCMP 4
1.4.12 $\overline{\text{CS1}}$/PB5 4
1.4.13 PC3-PC0 4
1.4.14 $\overline{\text{LIR}}$/MODE 5
1.5 DEVICE PINOUT 5

SECTION 2 MEMORY 9

2.1 EXTERNAL MAPPING EXCEPTIONS 11
2.2 RAM 11

SECTION 3 CPU CORE 13

3.1 REGISTERS 13
3.1.1 ACCUMULATOR (A) 14
3.1.2 INDEX REGISTER (X) 14
3.1.3 PROGRAM COUNTER (PC) 14
3.1.4 STACK POINTER (SP) 14
3.1.5 CONDITION CODE REGISTER (CCR) 14
3.2 ADDRESSING MODES 15
3.2.1 IMMEDIATE 15
3.2.2 DIRECT 15
3.2.3 EXTENDED 15
3.2.4 RELATIVE 15
3.2.5 INDEXED, NO OFFSET 16
3.2.6 INDEXED, 8-BIT OFFSET 16
3.2.7 INDEXED, 16-BIT OFFSET 16
3.2.8 BIT SET/CLEAR 16
3.2.9 BIT TEST AND BRANCH 16
3.2.10 INHERENT 17

SECTION 4 INTERRUPTS 19
4.1 HARDWARE CONTROLLED INTERRUPT SEQUENCE 20
4.2 SOFTWARE INTERRUPT (SWI) 20
4.3 EXTERNAL INTERRUPT 20
4.4 16-BIT TIMER INTERRUPT 21
4.5 SERIAL COMMUNICATIONS INTERFACE INTERRUPT 21
4.6 MULTI-FUNCTION TIMER INTERRUPTS 21
4.7 KEYBOARD SCAN INTERRUPT 21

SECTION 5 RESETS 23
5.1 POWER-ON RESET (POR) 23
5.2 COMPUTER OPERATING PROPERLY (COP) RESET 24
5.3 LOW-VOLTAGE RESET 24
5.4 ILLEGAL STOP RESET 24
5.5 $\overline{\text{RESET}}$ PIN 24

SECTION 6 LOW-POWER MODES 25
6.1 STOP 25
6.1.1 STOP RECOVERY 25
6.2 WAIT 25

SECTION 7 INPUT/OUTPUT PORTS 29
7.1 PORT B 29
7.2 PORT C 29
7.3 PORT D 29
7.4 INPUT/OUTPUT PROGRAMMING 30

SECTION 8 SYSTEM CONFIGURATION 33
8.1 EXPANDED BUS MODES 33
8.1.1 MUXED MODE 33
8.1.2 NON -MUXED MODE 33
8.2 CHIP SELECTS 33
8.2.1 PAGE-0 CHIP SELECT TYPE 33
8.2.2 GENERAL - PURPOSE CHIP SELECT TYPE 34
8.2.3 $\overline{\text{A15}}$ CHIP SELECT 34
8.3 CONFIGURATION REGISTER (CNFGR) 34
8.3.1 STPEN - SToP ENable 34
8.3.2 STREC - STop RECovery 34
8.3.3 COPEN - Computer Operating Properly ENable 34
8.3.4 IRV - Internal Read Visibility 34
8.3.5 LIRV - Load Instruction Register Visibility 35
8.3.6 LVREN - Low Voltage Reset ENable 35
8.3.7 CS1P1:CS1P0 - Chip Select 1 Parameters 35
8.4 EXTERNAL INTERRUPT CONTROL/STATUS REGISTER (EICSR) 35
8.4.1 KSF - Keyboard Scan request Flag 36

8.4.2 KSEN - Keyboard Scan ENable 36
8.4.3 Keyboard Scan Acknowledge 36
8.4.4 Keyboard Scan Edge/Level 36
8.4.5 IRQF - Interrupt ReQuest Flag 36
8.4.6 IRQEN - Interrupt ReQuest ENable 36
8.4.7 IRQA - Interrupt ReQuest Acknowledge 37
8.4.8 IRQEL - Interrupt ReQuest Edge/Level 37

SECTION 9 ADDRESS/DATA BUS INTERFACE 39

9.1 MODE SELECTION 39
9.2 MUXED MODE BUS TIMING 39
9.3 NON-MUXED MODE BUS TIMING 40
9.4 CLOCK STRETCHED BUS TIMING 41
9.4.1 MUXED MODE CLOCK STRETCHED READ CYCLE 42
9.4.2 MUXED MODE CLOCK STRETCHED WRITE CYCLE 42
9.4.3 NON-MUXED MODE CLOCK STRETCHED READ CYCLE 43
9.4.4 NON-MUXED MODE CLOCK STRETCHED WRITE CYCLE 43
9.5 BUS CYCLE CHARACTERISTICS 45
9.6 ADDRESS/DATA BUS DURING WAIT AND STOP MODE 45

SECTION 10 MULTI-FUNCTION TIMER 47

10.1 INTRODUCTION 47
10.2 TIMER CONTROL AND STATUS REGISTER (TCSR) 48
10.2.1 TOF - Timer Over Flow 48
10.2.2 RTIF - Real Time Interrupt Flag 48
10.2.3 TOFE - Timer Over Flow Enable 49
10.2.4 RTIE - Real Time Interrupt Enable 49
10.2.5 TOFC - Timer Over Flow Flag Clear 49
10.2.6 RTIFC - Real Time Interrupt Flag Clear 49
10.2.7 RT1:RT0 - Real Time Interrupt Rate Select 49
10.3 TIMER COUNTER REGISTER (TCR) 50
10.4 COMPUTER OPERATING PROPERLY (COP) WATCHDOG RESET 50
10.5 TIMER DURING WAIT MODE 51
10.6 TIMER DURING STOP MODE 51

SECTION 11 16-BIT TIMER 53

11.1 INTRODUCTION 53
11.2 COUNTER 53
11.3 OUTPUT COMPARE REGISTER 55
11.4 INPUT CAPTURE REGISTER 56
11.5 TIMER CONTROL REGISTER (TCR) 57
11.6 TIMER STATUS REGISTER (TSR) 59
11.7 TIMER DURING WAIT MODE 59
11.8 TIMER DURING STOP MODE 60

SECTION 12 SERIAL COMMUNICATIONS INTERFACE 61
12.1 OVERVIEW AND FEATURES 61
12.1.1 SCI TWO-WIRE SYSTEM FEATURES 61
12.1.2 SCI RECEIVER FEATURES 61
12.1.3 SCI TRANSMITTER FEATURES 61
12.2 DATA FORMAT 62
12.3 FUNCTIONAL DESCRIPTION 62
12.4 RECEIVER 65
12.4.1 SCI RECEIVER WAKE-UP FEATURE 65
12.4.1.1 SCI RECEIVER IDLE WAKE-UP 65
12.4.1.2 SCI RECEIVER ADDRESS MARK WAKE-UP 65
12.4.2 RECEIVE DATA IN 65
12.4.3 START BIT DETECTION FOLLOWING A FRAMING ERROR 67
12.5 TRANSMITTER 68
12.5.1 TRANSMIT DATA OUT 68
12.5.2 SCI SYNCHRONOUS TRANSMISSION 68
12.6 REGISTERS 69
12.6.1 SERIAL COMMUNICATIONS DATA REGISTER (SCDAT) 69
12.6.2 SERIAL COMMUNICATIONS CONTROL REGISTER 1 (SCCR1) 69
12.6.3 SERIAL COMMUNICATIONS CONTROL REGISTER 2 (SCCR2) 74
12.6.4 SERIAL COMMUNICATIONS STATUS REGISTER (SCSR) 76
12.6.5 BAUD RATE REGISTER 78
12.7 SCI DURING WAIT MODE 82
12.8 SCI DURING STOP MODE 82

SECTION 13 ELECTRICALS 83
13.1 MAXIMUM RATINGS 83
13.2 THERMAL CHARACTERISTICS 83
13.3 DC ELECTRICAL CHARACTERISTICS 84
13.4 CONTROL TIMING 85

LIST OF FIGURES

Figure 1-1 Block Diagram of the MC68HC05C0 2
Figure 1-2 40-Pin DIP Pinout of the MC68HC05C0 5
Figure 1-3 42-Pin SDIP Pinout of the MC68HC05C0 6
Figure 1-4 44-Pin PLCC Pinout of the MC68HC05C0 7

Figure 2-1 The 64K Memory Map of the MC68HC05C0 9
Figure 2-2 I/O Registers for the MC68HC05C0 10

Figure 3-1 Programming Model 13
Figure 3-2 Stacking Order 13

Figure 4-1 Interrupt Flowchart 22

Figure 5-1 Power-On Reset and $\overline{\text{RESET}}$ 23

Figure 6-1 Stop Recovery Timing Diagram 26
Figure 6-2 STOP/WAIT Flowchart 27

Figure 7-1 Port D Interrupt Option 30
Figure 7-2 Port I/O Circuitry 31

Figure 8-1 Configuration Register 34
Figure 8-2 External Interrupt Control/Status Register 36

Figure 9-1 Muxed Mode Timing 39
Figure 9-2 Non-Muxed Mode Timing 40
Figure 9-3 Muxed Mode Clock Stretched Read Cycle 42
Figure 9-4 Muxed Mode Clock Stretched Write Cycle 42
Figure 9-5 Non-Muxed Mode Clock Stretched Read Cycle 43
Figure 9-6 Non-Muxed Mode Clock Stretched Read Cycle 43

Figure 10-1 Multi-Function Timer Block Diagram 47
Figure 10-2 Timer Control and Status Register (TCSR) 48
Figure 10-3 Timer Counter Register 50

Figure 11-1 Timer Block Diagram 54
Figure 11-2 Timer Counter MSB Register 55
Figure 11-3 Timer Counter LSB Register 55
Figure 11-4 Timer Alternate Counter MSB Register 55
Figure 11-5 Timer Alternate Counter LSB Register 55
Figure 11-6 Timer Output Compare MSB Register 56
Figure 11-7 Timer Output Compare LSB Register 56
Figure 11-8 Timer Input Capture MSB Register 57
Figure 11-9 Timer Input Capture LSB Register 57
Figure 11-10 Timer Control Register 57
Figure 11-11 Timer Status Register 59

Figure 12-1 Data Format 62
Figure 12-2 SCI Block Diagram 64
Figure 12-3 SCI Examples of Start Bit Recognition Technique 66
Figure 12-4 SCI Sampling Technique Used on all Bits 66
Figure 12-5 SCI Artificial Start Following a Framing Error 67
Figure 12-6 SCI Start Bit Following a Break 68
Figure 12-7 Serial Communications Data Register 69
Figure 12-8 Serial Communications Control Register 69
Figure 12-9 SCI Data Clock Timing Diagram (M=0) 73
Figure 12-10 SCI Data Clock Timing Diagram (M=1) 73
Figure 12-11 Serial Communications Control Register 2 74
Figure 12-12 Serial Communications Status Register 76
Figure 12-13 Serial Communications Baud Rate Register 78
Figure 12-14 SCI Rate Generator Block Diagram 81

LIST OF TABLES

Table 4-1 Vector Address for Interrupts and Reset 19

Table 7-1 I/O Pin Functions 30

Table 8-1 Chip Select Types 35

Table 9-1 Bus Modes 39
Table 9-2 Muxed Mode Timing 40
Table 9-3 Non-Muxed Mode Timing 41
Table 9-4 Clock Stretched Muxed Mode Timing 44
Table 9-5 Clock Stretched Non-Muxed Mode Timing 44

Table 10-1 RTI Rates 49
Table 10-2 Minimum COP Reset Times 51

Table 12-1 Method of Receiver Wake-Up 71
Table 12-2 SCI Clock on SCK Pin 72
Table 12-3 Prescaler Highest Baud Rate Frequency Output 79
Table 12-4 Transmit Baud Rate Output for a Given Prescaler Output 80

THIS PAGE INTENTIONALLY LEFT BLANK

SECTION 1 INTRODUCTION

1.1 GENERAL

The MC68HC05C0 HCMOS Microcomputer is a member of the M68HC05 Family.This part is suitable for applications which require an external address and data bus. It provides a mode select for either a muxed or a non-muxed bus, and a clock stretching capability for slower peripherals. This 8-bit microcomputer unit (MCU) contains an on-chip oscillator, CPU, RAM, serial and parallel I/O, Multi-Function Timer, 16-Bit Timer, and a Low-Voltage Reset.

1.2 FEATURES

- HC05 Core
- 40 Pin PDIP, 42 Pin SDIP, and 44 Pin PLCC Packages
- Mode Select for Lower Byte Address/Data Muxed or Non-Muxed
- 4 MHz CPU Bus Rate for Non-Muxed Mode, 2 MHz for Muxed Mode
- 4 Bus Cycle Clock Stretching for Slow Peripherals
- 512 Bytes of On-Chip RAM
- 16-Bit Timer with Input Capture and Output Compare
- Up to 18 Bidirectional I/O Lines with 44 Pin PLCC Package
- Programmable Interrupt and Pullups on 8 I/O Lines (Keyboard Scan) with Enable, Acknowledge, Request, and Edge or Edge and Level Sensitivity
- External interrupt ($\overline{IRQ}$) with Enable, Acknowledge, Request, and Edge or Edge and Level Sensitivity
- Low-Voltage Inhibit to Hold CPU in Reset (LVR)
- Power Saving STOP and WAIT Modes
- Internal Read Visibility and Load Instruction Register Capability for Emulation
- Serial Communications Interface (SCI) with Synchronous Master Transmit Capability
- Up to 3 Chip Selects
- Multi-Function Timer with Selectable Time-out COP Watchdog
- High Current Source and Sink Pin

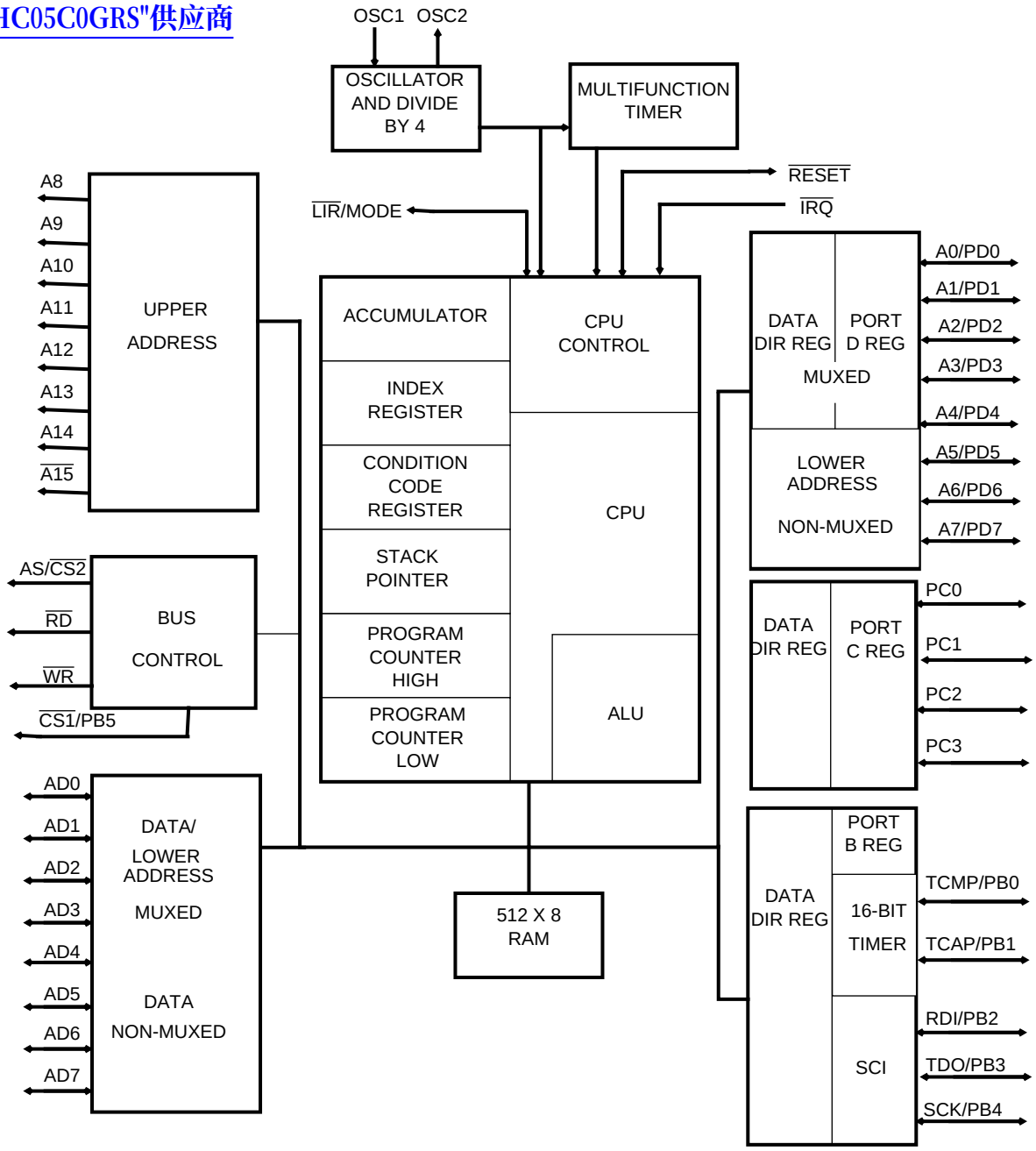


Figure 1-1: Block Diagram of the MC68HC05C0

1.3 MASK OPTIONS

There are no mask options. Instead a Configuration Register and an External Interrupt Control/Status Register are provided. See Section 8.3 CONFIGURATION REGISTER (CNFGR) for more details.

NOTE: When the MCU is used in a noisy environment, it is advisable that the Configuration Register be periodically monitored to ensure the integrity of the programmed options.

1.4 SIGNAL DESCRIPTION

1.4.1 V_{DD} AND V_{SS}

Power is supplied to the microcontroller using these two pins. VDD is the positive supply and VSS is ground.

1.4.2 $\overline{IRQ}$

This active low input-only pin is the external interrupt. The $\overline{IRQ}$ pin contains an internal Schmitt trigger as part of its input to improve noise immunity. An External Interrupt Control/ Status Register provides a one-time writable (at reset) choice of interrupt triggering sensitivity. This register also provides enable, request and acknowledge bits for the $\overline{IRQ}$. See Section 4 INTERRUPTS for more information.

1.4.3 OSC1, OSC2

These pins provide control input for an on-chip clock oscillator circuit. A crystal, a ceramic resonator, or an external signal connects to these pins providing a system clock. The oscillator frequency is four times the internal bus rate.

1.4.4 $\overline{RESET}$

This active low bidirectional control pin is used as an input to reset the MCU to a known start-up state. The $\overline{RESET}$ pin contains an internal Schmitt trigger as part of its input to improve noise immunity. $\overline{RESET}$ is also an open-drain output to indicate that any of four possible internal reset conditions occurred. See Section 5 RESETS for more information.

1.4.5 $\overline{A15}$,A14-A8

These eight dedicated lines constitute the upper address byte. $\overline{A15}$ is active low and divides the memory map into two 32K regions. In this manner, $\overline{A15}$ can be used as a chip select for program memory without the need of an inverter. See Section 9 ADDRESS/ DATA BUS INTERFACE for the timing and a detailed description of the address bus.

1.4.6 AD7:0

These eight dedicated lines constitute the lower address or data byte. In Non-Muxed mode, these eight lines become data bits D7-D0. In Muxed mode, address and data are multiplexed together. See Section 9 ADDRESS/DATA BUS INTERFACE for the timing and a detailed description of the address/data bus.

1.4.7 AS (ADDRESS STROBE)/ $\overline{CS2}$

In Muxed mode, this pin is the active high Address Strobe, which is used to indicate the presence of the lower address byte on the muxed address/data bus. See Section 9 ADDRESS/DATA BUS INTERFACE for the timing and a detailed description of the address strobe. In Non-Muxed mode, this active low pin becomes an active low chip select. See Section 8 SYSTEM CONFIGURATION for more information on the chip selects.

1.4.8 $\overline{RD}$

This active low output pin is used to drive an external peripheral during an external read cycle. It can also indicate an internal read cycle if Internal Read Visibility is selected in the Configuration Register. See Section 9 ADDRESS/DATA BUS INTERFACE for the timing and a detailed description of the $\overline{RD}$ signal.

1.4.9 $\overline{WR}$

This active low output pin is used to drive an external peripheral during an external write cycle. It can also indicate an internal write cycle if Internal Read Visibility is selected in the Configuration Register. See Section 9 ADDRESS/DATA BUS INTERFACE for the timing and a detailed description of the $\overline{WR}$ signal.

1.4.10 A7-A0/PD7-PD0

These eight I/O lines constitute either the lower address byte or Port D. In Non-Muxed mode, they are dedicated to the lower address byte. See Section 9 ADDRESS/DATA BUS INTERFACE for the timing and a detailed description of the address bus. In Muxed mode, they are Port D. The state of any Port D pin is software programmable and all Port D lines are configured as inputs during Reset. See Section 7.4 INPUT/OUTPUT PROGRAMMING for a detailed description of I/O programming. Port D also can be programmed to enable internal pullups and generate an interrupt when any of the 8 I/O lines are pulled low. This can be used as a keyboard scan. See Section 7.3 PORT D for more details.

1.4.11 PB4-PB0/SCK,TDO,RDI,TCAP,TCMP

These five I/O lines constitute Port B and are shared with internal peripherals. The state of any pin is software programmable and all Port B lines are configured as inputs during Reset. See Section 7.4 INPUT/OUTPUT PROGRAMMING for a detailed description of I/O programming. Signals SCK,TDO and RDI of the SCI subsystem are shared with PB4, PB3 and PB2. See Section 12 SERIAL COMMUNICATIONS INTERFACE for a detailed description of the SCI. Signals TCAP and TCMP of the 16-Bit Timer are shared with PB1 and PB0. See Section 11 16-BIT TIMER for a detailed description of the 16-Bit Timer. PB0 has a high current sink and source capability.

1.4.12 $\overline{CS1}$/PB5

This pin can be used as a dedicated chip select or as an additional Port B line (PB5). See Section 8 SYSTEM CONFIGURATION for more information on the chip selects. If it is used as a Port B line, it will function identically to PB4-PB0. See Section 7.4 INPUT/OUTPUT PROGRAMMING for a detailed description of I/O programming.

1.4.13 PC3-PC0

These four I/O lines constitute Port C. The state of any pin is software programmable and all Port C lines are configured as inputs during Reset. See Section 7.4 INPUT/OUTPUT PROGRAMMING for a detailed description of I/O programming. Port C is not available on the 40 Pin PDIP package, and only PC0 and PC2 are available on the 42 Pin SDIP package.

1.4.14 $\overline{\text{LIR}}$/MODE

During Power-On Reset, the MCU latches the state of the MODE pin to select between Muxed or Non-Muxed modes of operation. See Section 9 ADDRESS/DATA BUS INTERFACE for a detailed description of the MODE pin. During normal operation, the Configuration Register can be programmed to make this pin the Load Instruction Register signal. This active low output is used to indicate that a fetch of the next opcode is in progress.

1.5 DEVICE PINOUT

See **Figure 1-2 : 40-Pin DIP Pinout of the MC68HC05C0**, **Figure 1-3 : 42-Pin SDIP Pinout of the MC68HC05C0** and **Figure 1-4 : 44-Pin PLCC Pinout of the MC68HC05C0** for the pinout of the three available packages.

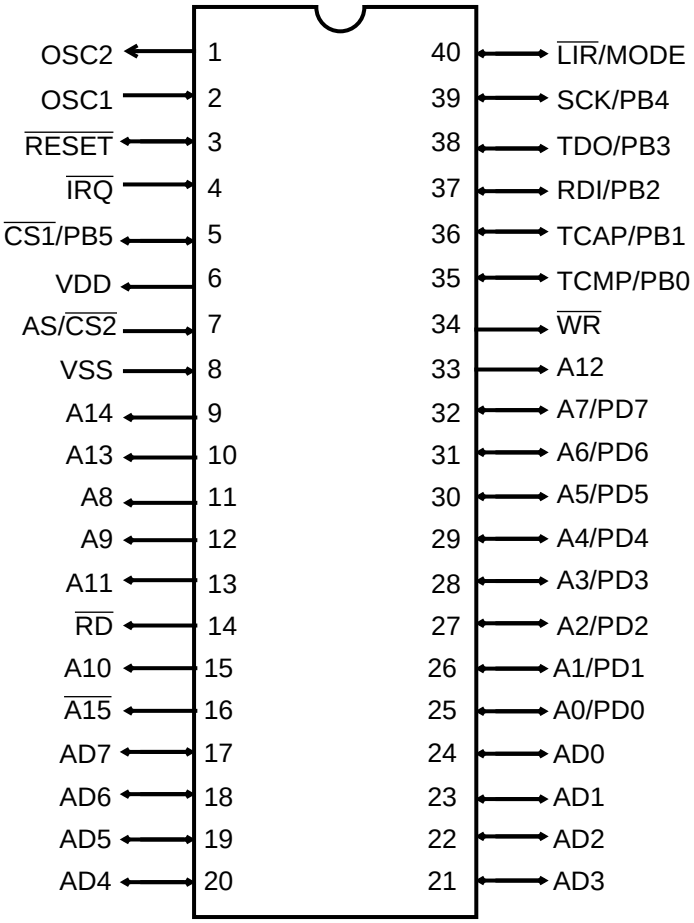


Figure 1-2: 40-Pin DIP Pinout of the MC68HC05C0

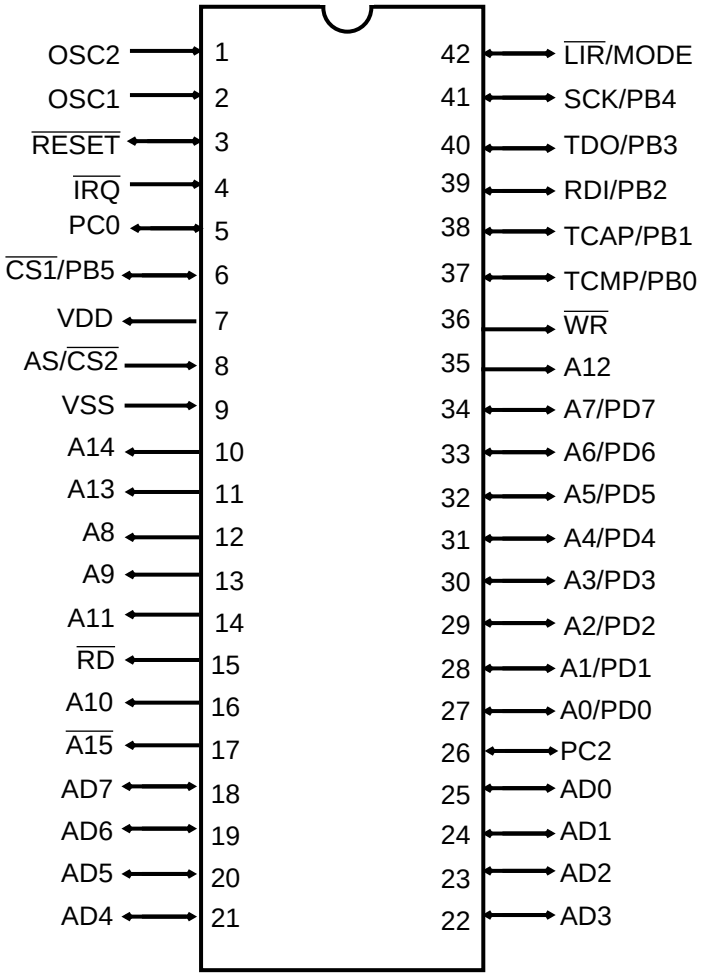


Figure 1-3: 42-Pin SDIP Pinout of the MC68HC05C0

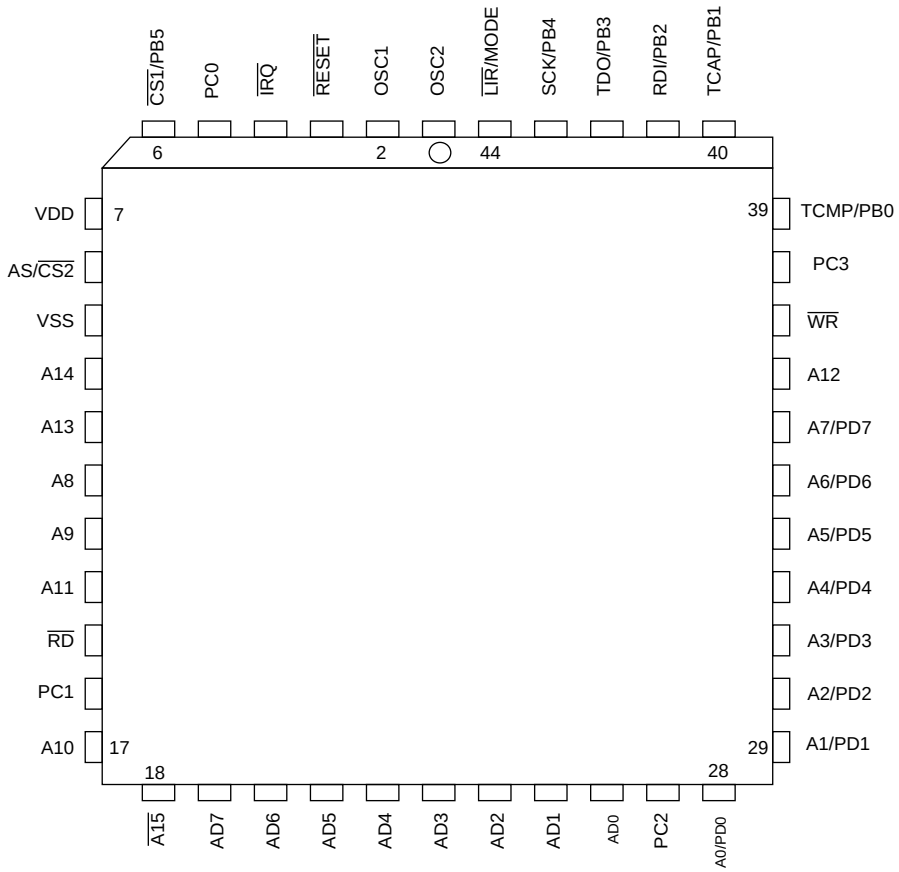
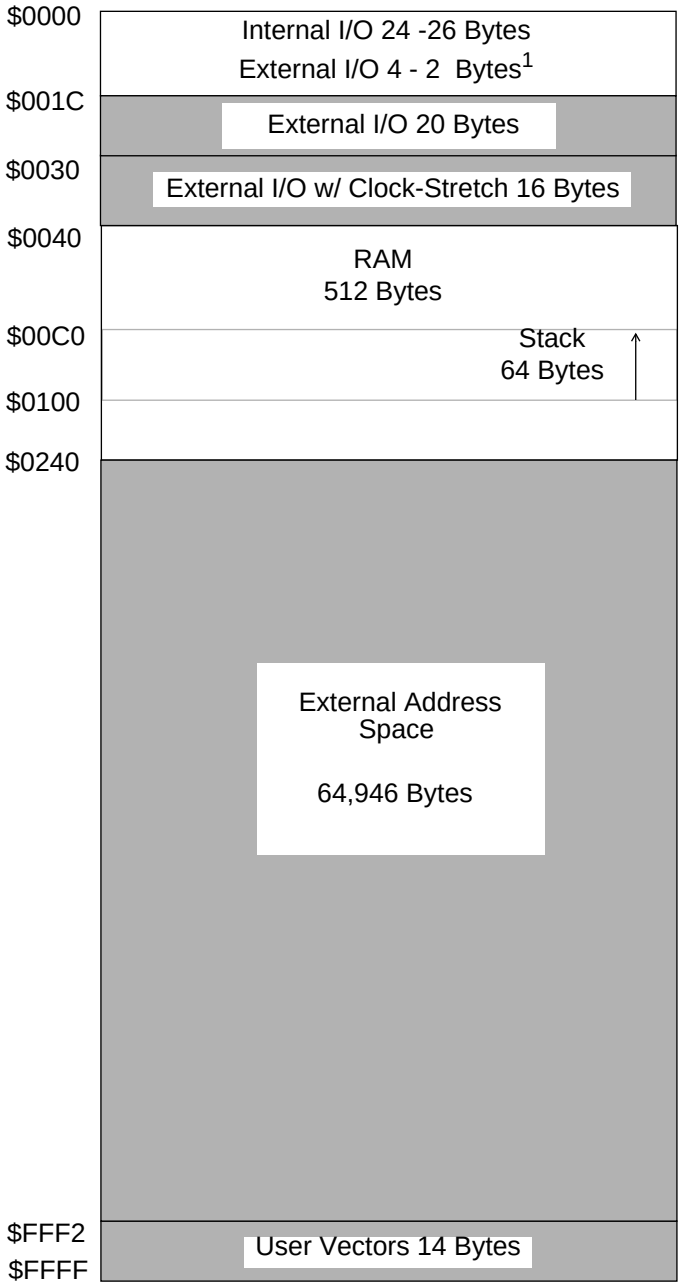


Figure 1-4: 44-Pin PLCC Pinout of the MC68HC05C0

THIS PAGE INTENTIONALLY LEFT BLANK

SECTION 2 MEMORY

The MC68HC05C0 has a 64K byte memory map, consisting of user RAM, I/O and external memory space. See **Figure 2-1 : The 64K Memory Map of the MC68HC05C0** and **Figure 2-2 : I/O Registers for the MC68HC05C0**.



Note [1]: Addresses $00 and $04 are mapped external;
Port D is mapped external in Non-Muxed mode

Figure 2-1: The 64K Memory Map of the MC68HC05C0

ADDRESS 0000 TO 001B	DATA 7	6	5	4	3	2	1	0
$00 EXTERNAL	—	—	—	—	—	—	—	—
$01 PORT B DATA	0	0	DA5	DA4	DA3	DA2	DA1	DA0
$02 PORT C DATA	0	0	0	0	DA3	DA2	DA1	DA0
$03 PORT D DATA	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0
$04 EXTERNAL	—	—	—	—	—	—	—	—
$05 PORT B DDR	0	0	DDR5	DDR4	DDR3	DDR2	DDR1	DDR0
$06 PORT C DDR	0	0	0	0	DDR3	DDR2	DDR1	DDR0
$07 PORT D DDR	DDR7	DDR6	DDR5	DDR4	DDR3	DDR2	DDR1	DDR0
$08 MFT CTRL/STAT	TOF	RTIF	TOFE	RTIE	TOFC	RTIFC	RT1	RT0
$09 MFT CNTR	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0
$0A TIM CTRL	ICIE	OCIE	TOIE	OCM	ICM	0	IEDG	OLVL
$0B TIM STATUS	ICF	OCF	TOF	0	0	0	0	0
$0C TIM ICR (MSB)	DA15	DA14	DA13	DA12	DA11	DA10	DA9	DA8
$0D TIM ICR (LSB)	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0
$0E TIM OCR (MSB)	DA15	DA14	DA13	DA12	DA11	DA10	DA9	DA8
$0F TIM OCR (LSB)	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0
$10 TIM CNTR (MSB)	CNT15	CNT14	CNT13	CNT12	CNT11	CNT10	CNT9	CNT8
$11 TIM CNTR (LSB)	CNT7	CNT6	CNT5	CNT4	CNT3	CNT2	CNT1	CNT0
$12 TIM ACNTR (MSB)	CNT15	CNT14	CNT13	CNT12	CNT11	CNT10	CNT9	CNT8
$13 TIM ACNTR (LSB)	CNT7	CNT6	CNT5	CNT4	CNT3	CNT2	CNT1	CNT0
$14 SCI BAUD	SCP1	SCP0	SCT2	SCT1	SCT0	SCR2	SCR1	SCR0
$15 SCI CTRL1	R8	T8	SCKM	M	WAKE	CPOL	CPHA	LBCL
$16 SCI CTRL2	TIE	TCIE	RIE	ILIE	TE	RE	RWU	SBK
$17 SCI STATUS	TDRE	TC	RDRF	IDLE	OR	NF	FE	—
$18 SCI DATA	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0
$19 CONFIG REG	STPEN	STREC	COPEN	IRV	LIRV	LVREN	CS1P1	CS1P0
$1A EISC REG	KSF	KSEN	KSA	KSEL	IRQF	IRQEN	IRQA	IRQEL
RESERVED	—	—	—	—	—	—	—	—

Figure 2-2: I/O Registers for the MC68HC05C0

2.1 EXTERNAL MAPPING EXCEPTIONS

In the first 28 bytes of the Memory Map, most of the addresses are mapped internal as on-chip peripheral registers. However, addresses $00 and $04 are mapped external, so that an external Port Replacement Unit may be used for Port A. In Non-Muxed mode Port D addresses $03 and $07 are mapped external, because the Port D pins are used as the lower-order address bus.

2.2 RAM

The user RAM consists of 512 bytes from location $0040 to $0240 including the stack area. The stack begins at address $00FF. The stack pointer can access 64 bytes of RAM in the range $00FF to $00C0. See Section 3.1.4 STACK POINTER (SP).

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.

THIS PAGE INTENTIONALLY LEFT BLANK

SECTION 3 CPU CORE

3.1 REGISTERS

The MCU contains five registers as shown in the programming model of **Figure 3-1 : Programming Model**. The interrupt stacking order is shown in **Figure 3-2 : Stacking Order**.

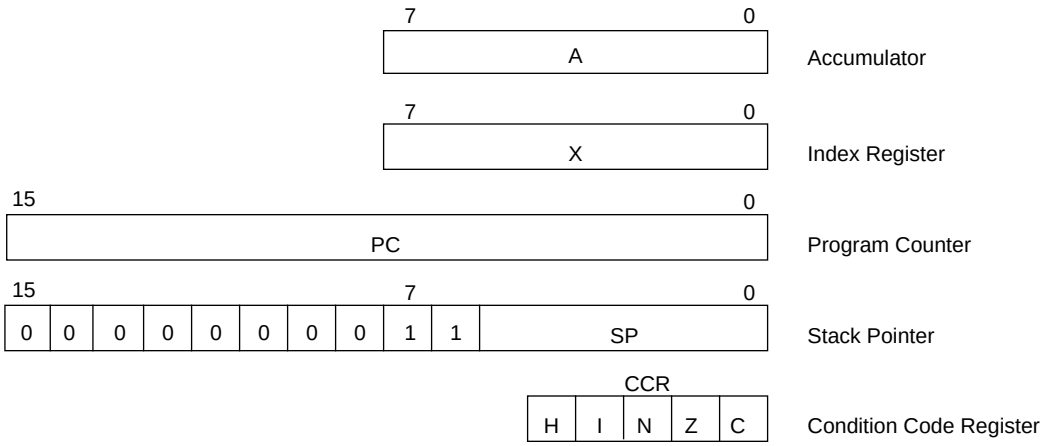
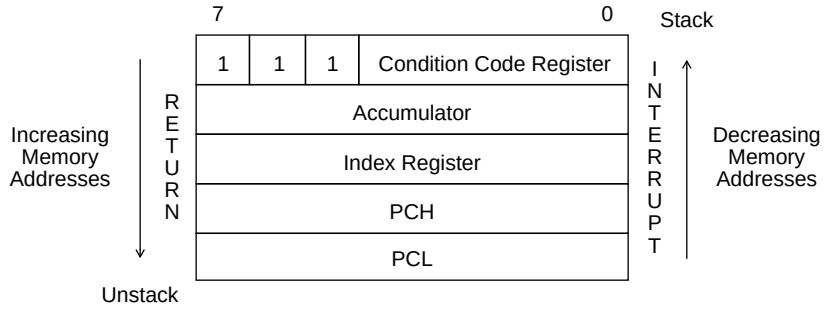


Figure 3-1: Programming Model



Since the Stack Pointer decrements during pushes, the PCL is stacked first, followed by PCH, etc. Pulling from the stack is in the reverse order.

Figure 3-2: Stacking Order

3.1.1 ACCUMULATOR (A)

The accumulator is a general purpose 8-bit register used to hold operands and results of arithmetic calculations or data manipulations.

3.1.2 INDEX REGISTER (X)

The index register is an 8-bit register used for the indexed addressing value to create an effective address. The index register may also be used as a temporary storage area.

3.1.3 PROGRAM COUNTER (PC)

The program counter is a 16-bit register that contains the address of the next byte to be fetched.

3.1.4 STACK POINTER (SP)

The stack pointer contains the address of the next free location on the stack. During Reset or the reset stack pointer (RSP) instruction, the stack pointer is set to location $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 ten most significant bits are permanently set to 0000000011. These ten bits are appended to the six least significant register bits to produce an address within the range of $00FF to $00C0. Subroutines and interrupts may use up to 64 (decimal) locations. If 64 locations are exceeded, the stack pointer wraps around and loses the previously stored information. A subroutine call occupies two locations on the stack; an interrupt uses five locations.

3.1.5 CONDITION CODE REGISTER (CCR)

The condition code register is a 5-bit register in which four bits are used to indicate the results of the instruction just executed, and the fifth bit indicates whether interrupts are masked. These bits can be individually tested by a program, and specific actions can be taken as a result of their state. Each bit is explained in the following paragraphs.

3.1.5.1 Half Carry (H)

This bit is set during ADD and ADC operations to indicate that a carry occurred between bits 3 and 4.

3.1.5.2 Interrupt (I)

When this bit is set, all interrupts with the exception of SWI are masked (disabled). If a non-SWI interrupt occurs while this bit is set, the interrupt is latched and processed as soon as the interrupt bit is cleared. SWI causes an interrupt regardless of the state of this bit.

3.1.5.3 Negative (N)

When set, this bit indicates that the result of the last arithmetic, logical, or data manipulation was negative.

3.1.5.4 Zero (Z)

When set, this bit indicates that the result of the last arithmetic, logical, or data manipulation was zero.

3.1.5.5 Carry/Borrow (C)

When set, this bit indicates that a carry or borrow out of the arithmetic logical unit (ALU) occurred during the last arithmetic operation. This bit is also affected during bit test and branch instructions and during shifts and rotates.

3.2 ADDRESSING MODES

The MCU uses ten different addressing modes to provide the programmer with an opportunity to optimize the code for all situations. The various indexed addressing modes make it possible to locate data tables, code conversion tables, and scaling tables anywhere in the memory space. Short indexed accesses are single byte instructions; the longest instructions (three bytes) permit accessing tables throughout memory. Short and long absolute addressing is also included. One- or two-byte direct addressing instructions access all data bytes in most applications. Extended addressing permits jump instructions to reach all memory.

The term “effective address” (EA) is used in describing the various addressing modes. Effective address is defined as the address at which the argument for an instruction is fetched or stored.

3.2.1 IMMEDIATE

In the immediate addressing mode, the operand is contained in the byte immediately following the opcode byte. The immediate addressing mode is used to access constants that do not change during program execution (e.g., a constant used to initialize a loop counter).

3.2.2 DIRECT

In the direct addressing mode, the effective address of the argument is contained in a single byte following the opcode byte. Direct addressing allows the user to directly address the lowest 256 bytes in memory with a single two-byte instruction.

3.2.3 EXTENDED

In the extended addressing mode, the effective address of the argument is contained in the two bytes following the opcode byte. Instructions with extended addressing mode are capable of referencing arguments anywhere in memory with a single three-byte instruction. When using the Motorola assembler, the user need not specify whether an instruction uses direct or extended addressing. The assembler automatically selects the shortest form of the instruction.

3.2.4 RELATIVE

The relative addressing mode is used only in branch instructions. In relative addressing, the contents of the 8-bit signed offset byte, which is the last byte of the instruction, is added to the PC if, and only if, the branch conditions are true. Otherwise control proceeds to the

next instruction. The span of relative addressing is from -128 to +127 from the address of the next opcode. The programmer need not calculate the offset when using the Motorola assembler, since it calculates the proper offset and checks to see that it is within the span of the branch.

3.2.5 INDEXED, NO OFFSET

In the indexed, no offset addressing mode, the effective address of the argument is contained in the 8-bit index register. This addressing mode can access the first 256 memory locations. These instructions are only one byte long. This mode is often used to move a pointer through a table or to hold the address of a frequently referenced I/O or RAM location.

3.2.6 INDEXED, 8-BIT OFFSET

In the indexed, 8-bit offset addressing mode, the effective address is the sum of the contents of the unsigned 8-bit index register and the unsigned byte following the opcode byte. The addressing mode is useful for selecting the K^{th} element in an n element table. With this two-byte instruction, K would typically be in X with the address of the beginning of the table in the second byte of the instruction. As such, tables may begin anywhere within the first 256 addressable locations and could extend as far as location 510 ($1FE). This is the last location which can be accessed in this way.

3.2.7 INDEXED, 16-BIT OFFSET

In the indexed, 16-bit offset addressing mode, the effective address is the sum of the contents of the unsigned 8-bit index register and the two unsigned bytes following the opcode byte. This addressing mode can be used in a manner similar to indexed, 8-bit offset except that this three-byte instruction allows tables to be anywhere in memory. As with direct and extended addressing, the Motorola assembler determines the shortest form of indexed addressing.

3.2.8 BIT SET/CLEAR

In the bit set/clear addressing mode, the bit to be set or cleared is part of the opcode, and the byte following the opcode specifies the direct addressing of the byte in which the specified bit is to be set or cleared. Any read/write bit in the first 256 locations of memory, including I/O, can be selectively set or cleared with a single two-byte instruction.

3.2.9 BIT TEST AND BRANCH

The bit test and branch addressing mode is a combination of direct addressing and relative addressing. The bit that is to be tested and its condition (set or clear) is included in the opcode. The address of the byte to be tested is in the single byte immediately following the opcode byte. The signed relative 8-bit offset in the third byte is added to the PC if the specified bit is set or cleared in the specified memory location. This single three-byte instruction allows the program to branch based on the condition of any readable bit in the first 256 locations of memory. The span of branching is from -128 to +127 from the address of the next opcode. The state of the tested bit is also transferred to the carry bit of the condition code register.

3.2.10 INHERENT

In the inherent addressing mode, all the information necessary to execute the instruction is contained in the opcode. Operations specifying only the index register and/or accumulator as well as the control instruction with no other arguments are included in this mode. These instructions are one byte long.

THIS PAGE INTENTIONALLY LEFT BLANK

SECTION 4 INTERRUPTS

The MCU can be interrupted six different ways: one maskable external interrupt: $\overline{IRQ}$, four software selectable maskable interrupts: Serial Communications Interface (SCI), 16-Bit Timer (TIM), Multi-Function Timer (MFT), Keyboard Scan (KEY), and the nonmaskable software interrupt instruction (SWI).

If the interrupt mask bit (I bit) of the CCR is set, all maskable interrupts (internal and external) are disabled. Clearing the I bit enables interrupts.

Interrupts cause the processor to save register contents on the stack and to set the interrupt mask (I bit) to prevent additional interrupts. The RTI instruction causes the register contents to be recovered from the stack and normal processing to resume.

Unlike Reset, hardware interrupts do not cause the current instruction execution to be halted, but are considered pending until the current instruction is complete.

NOTE: The current instruction is the one already fetched and being operated on.

When the current instruction is complete, the processor checks all pending hardware interrupts. If interrupts are not masked (CCR I bit clear) the processor proceeds with interrupt processing; otherwise, the next instruction is fetched and executed.

The SWI is executed the same as any other instruction, regardless of the I-bit state.

Table 4-1 lists vector addresses in order of priority for all interrupts including Reset.

Table 4-1: Vector Address for Interrupts and Reset

Register	Flag Name	Interrupts	CPU Interrupt	Vector Address
N/A	N/A	Reset	$\overline{RESET}$	$FFFE-$FFFF
N/A	N/A	Software	SWI	$FFFC-$FFFD
N/A	N/A	External Interrupt	$\overline{IRQ}$	$FFFA-$FFFB
TSR	ICF	16-Bit Timer	TIM	$FFF8-$FFF9
TSR	OCF	16-Bit Timer	TIM	$FFF8-$FFF9
TSR	TOF	16-Bit Timer	TIM	$FFF8-$FFF9
SCSR	TDRE	Serial Comm. Interface	SCI	$FFF6-$FFF7
SCSR	TC	Serial Comm. Interface	SCI	$FFF6-$FFF7
SCSR	OR/RDRF	Serial Comm. Interface	SCI	$FFF6-$FFF7
SCSR	IDLE	Serial Comm. Interface	SCI	$FFF6-$FFF7
TCSR	TOF	Multi-Function Timer	MFT	$FFF4-$FFF5
TCSR	RTIF	Multi-Function Timer	MFT	$FFF4-$FFF5
N/A	N/A	Keyboard Scan	KEY	$FFF2-$FFF3

4.1 HARDWARE CONTROLLED INTERRUPT SEQUENCE

The following three functions (RESET, STOP, and WAIT) are not in the strictest sense an interrupt; however, they are acted upon in a similar manner. Flowcharts for hardware interrupts are shown in **Figure 4-1 : Interrupt Flowchart**, and for STOP and WAIT in **Figure 6-2 : STOP/WAIT Flowchart**. A discussion is provided below.

1. RESET - Five different reset mechanisms can cause the MCU to vector to its starting address which is specified by the contents of memory locations $FFFE and $FFFF. The I bit in the condition code register is also set. Much of the MCU is configured to a known state during Reset. **See Section 5 RESETS** for a description of reset mechanisms.
2. STOP - The STOP instruction causes the oscillator to be turned off and the processor to “sleep” until an external interrupt ($\overline{IRQ}$), a KEY interrupt or Reset occurs.
3. WAIT - The WAIT instruction causes all processor clocks to stop, but leaves the timer clock running. This “rest” state of the processor can be cleared by Reset, an external interrupt ($\overline{IRQ}$), a SCI interrupt, a TIM interrupt, a KEY interrupt, or a MFT interrupt.

4.2 SOFTWARE INTERRUPT (SWI)

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

4.3 EXTERNAL INTERRUPT

The interrupt request is latched immediately following the falling edge of $\overline{IRQ}$. It is then serviced as specified by the contents of $FFFA and $FFFB.

Either a level and edge-sensitive trigger option, or an edge-sensitive-only trigger option is one-time writable (at reset) in the External Interrupt Control/Status Register. In addition, this register also provides enable, request and acknowledge bits for the $\overline{IRQ}$. The enable bit (IRQEN) allows the $\overline{IRQ}$ interrupt to be masked without having to set the I bit. The request bit (IRQF) indicates that an $\overline{IRQ}$ request is pending. The $\overline{IRQ}$ acknowledge bit (IRQA) allows a pending $\overline{IRQ}$ interrupt to be cleared without having to enter the $\overline{IRQ}$ interrupt service routine. See Section 8.3 CONFIGURATION REGISTER (CNFGR) for more details.

NOTE: The internal interrupt latch is automatically cleared in the first part of the interrupt service routine; therefore, one external interrupt pulse could be latched and serviced as soon as the I bit is cleared.

4.4 16-BIT TIMER INTERRUPT

There are three16-Bit Timer interrupt flags that cause a timer interrupt whenever they are set and enabled. The interrupt flags and enable bits are located in the Timer Status Register (TSR) and the Timer Control Register (TCR) respectively. Any of these interrupts will vector to the same interrupt service routine, located at the address specified by the contents of memory location $FFF8 and $FFF9. See Section 11 16-BIT TIMER for more information on the 16-Bit Timer interrupts.

4.5 SERIAL COMMUNICATIONS INTERFACE INTERRUPT

There are four Serial Communications Interface (SCI) interrupt flags that will cause an interrupt whenever they are set and enabled. The interrupt flags and enable bits are located in the SCI Status Register (SCSR) and the SCI Control Register2 (SCCR2) respectively. Any of these interrupts will vector to the interrupt service routine located at the address specified by the contents of memory locations $FFF6 and $FFF7. See Section 12 SERIAL COMMUNICATIONS INTERFACE for more information on the SCI interrupts.

4.6 MULTI-FUNCTION TIMER INTERRUPTS

There are two Multi-Function Timer (MFT) interrupt flags that will cause an interrupt whenever they are set and enabled. The interrupt flags and enable bits are located in the MFT Control and Status Register (TCSR). Either of these interrupts will vector to the same interrupt service routine, located at the address specified by the contents of memory locations $FFF4 and $FFF5. See Section 10 MULTI-FUNCTION TIMER for more information on MFT interrupts.

4.7 KEYBOARD SCAN INTERRUPT

Port D can be configured to enable internal pullups and generate an interrupt when any of the 8 I/0 Lines are pulled low.

Either a level and edge-sensitive trigger option, or an edge-sensitive-only trigger option is one-time writable (at reset) in the External Interrupt Control/Status Register. In addition, this register also provides enable, request and acknowledge bits for the Keyboard Scan Interrupt (KEY). A KEY interrupt will vector to the interrupt service routine, located at the address specified by the contents of memory locations $FFF2 and $FFF3. See Section 8.3 CONFIGURATION REGISTER (CNFGR) for more details.

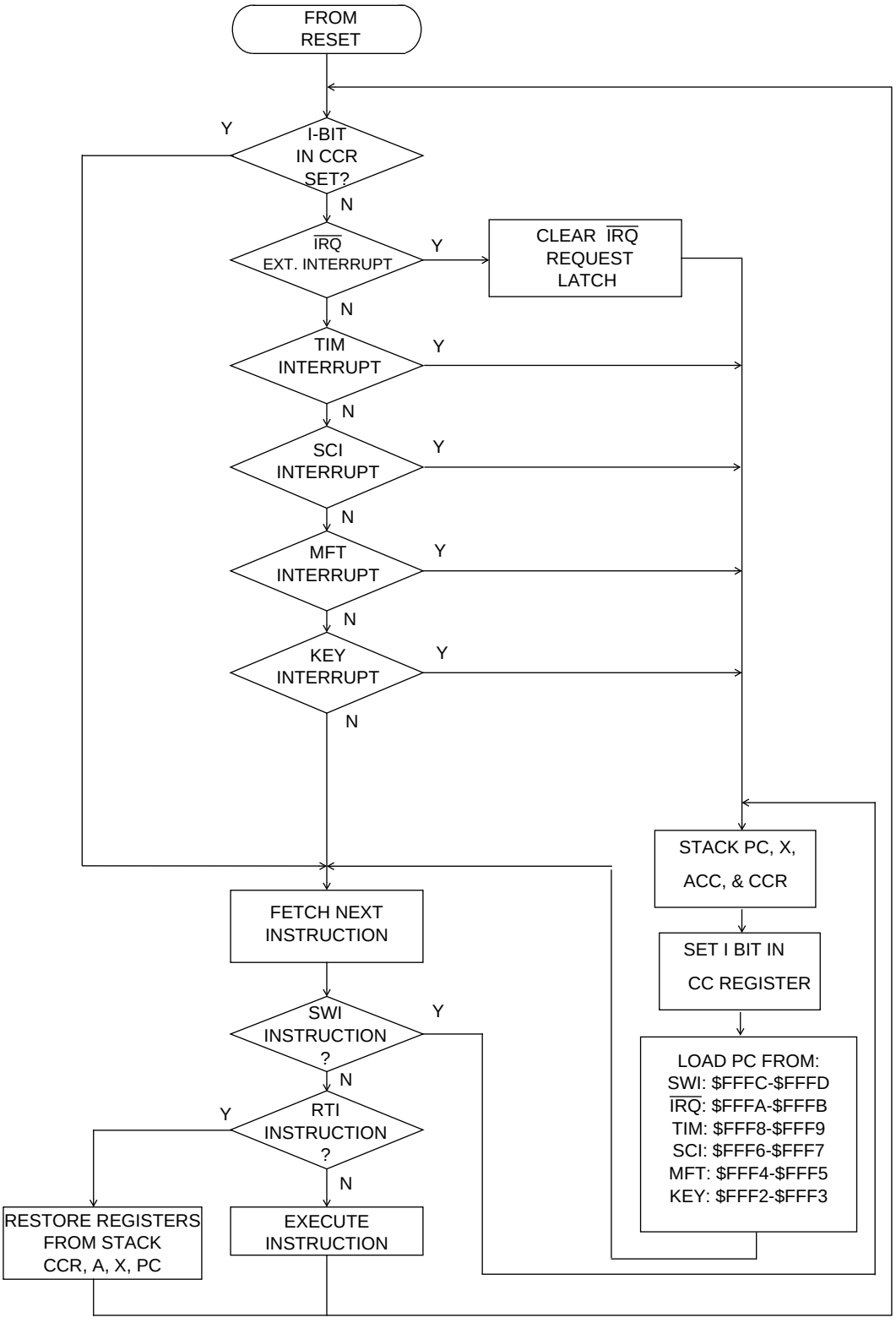


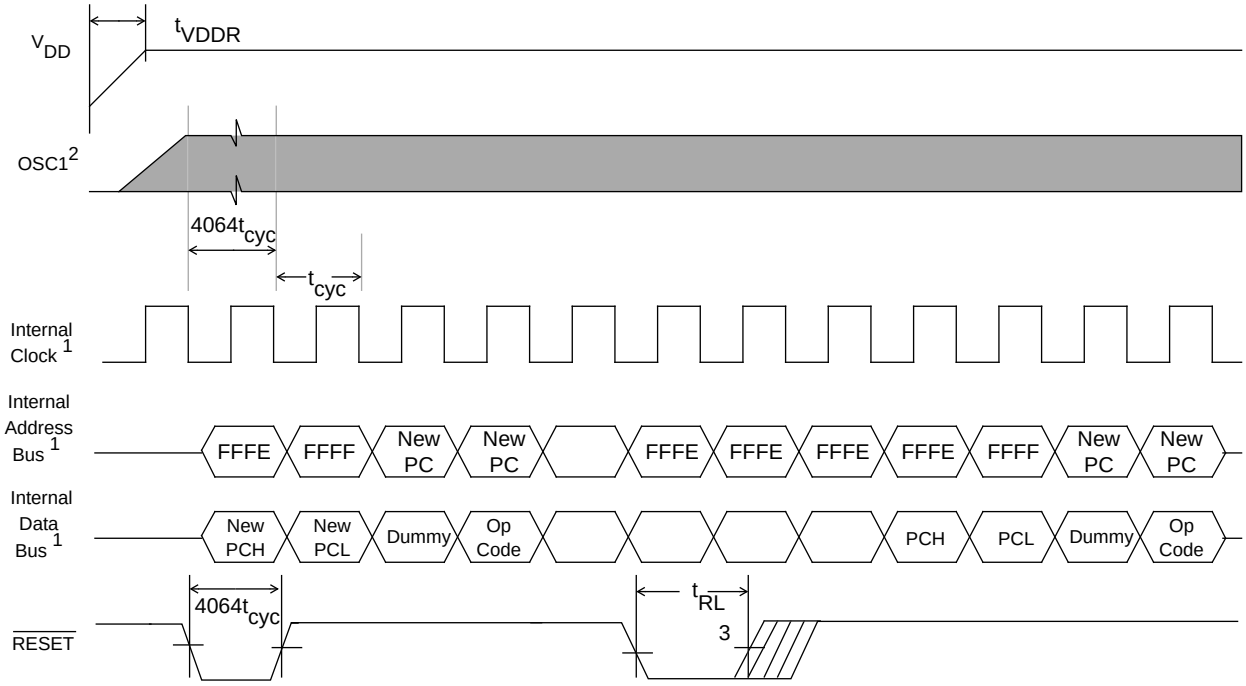
Figure 4-1: Interrupt Flowchart

SECTION 5 RESETS

The MCU can be reset five ways. There are four internal reset conditions: the initial power-on reset (POR) function, a COP watchdog-timer reset, a low-voltage reset (LVR), and an illegal STOP reset, and one external reset condition: an active low input to the $\overline{\text{RESET}}$ pin. The four internal conditions cause the $\overline{\text{RESET}}$ pin to be driven low for a minimum of eight processor clock cycles (t_{cyc}) to allow external peripherals to reset. Any reset will cause the operating mode and the Muxed/Non-Muxed mode to be relatched on the rising edge or $\overline{\text{RESET}}$.

5.1 POWER-ON RESET (POR)

An internal reset is generated on power-up to allow the internal clock generator to stabilize. The power-on reset is strictly for power turn-on conditions and should not be used to detect a drop in the power supply voltage. There is a 4064 internal processor clock cycle (t_{cyc}) oscillator stabilization delay after the oscillator becomes active. If the $\overline{\text{RESET}}$ pin is held low at the end of this 4064 cycle delay, the MCU will remain in the reset condition until $\overline{\text{RESET}}$ is driven high.



NOTES:
1. Internal timing signal and internal bus information not available externally.
2. OSC1 line is not meant to represent frequency. It is only used to represent time.
3. The next rising edge of the internal processor clock following the rising edge of $\overline{\text{RESET}}$ initiates the reset sequence.

Figure 5-1: Power-On Reset and $\overline{\text{RESET}}$

5.2 COMPUTER OPERATING PROPERLY (COP) RESET

The MCU contains a watchdog timer that when enabled, automatically times out if not reset (cleared) within a specific time by a program reset sequence. If the COP watchdog timer is allowed to time out, a reset is generated to reset the MCU and external peripheral. See Section 10 MULTI-FUNCTION TIMER, for more information on the COP Watchdog timer.

5.3 LOW-VOLTAGE RESET

The internal low voltage (LVR) reset is generated when the supply voltage to the V_{DD} pin falls below a nominal 3.50 VDC. The LVR threshold is not intended to be an accurate and stable trip point, but is intended to assure that the CPU will be held in reset when the V_{DD} supply voltage is below reasonable operating limits. An enable bit (LVREN) is provided in the Configuration Register to activate the LVR. If the LVR is not desired, or if the part is used in a low voltage application, the LVR can be left disabled.

The low voltage reset will be activated as long as the low voltage reset condition exists. This will also hold the MCU and external peripherals in reset. When the low voltage condition ends, the low voltage reset will remain active for an additional 4064 cycles. If the LVR is tripped only for a very short time, the LVR reset signal will still last at least eight processor clock cycles (t_{cyc}). If any other reset function is active at the end of the LVR reset signal, the $\overline{\text{RESET}}$ signal will remain low until the other reset condition(s) end.

5.4 ILLEGAL STOP RESET

The Configuration Register provides a one-time writable bit to enable/disable STOP. If STOP mode is disabled and a STOP instruction is executed, an illegal STOP reset occurs. This is to protect against runaway code which could inadvertently execute STOP, thereby preventing a COP time-out to occur.

5.5 $\overline{\text{RESET}}$ PIN

The MCU is reset when a logic zero is applied to the $\overline{\text{RESET}}$ input for a period of one and one-half machine cycles (t_{cyc}).

SECTION 6 LOW-POWER MODES

6.1 STOP

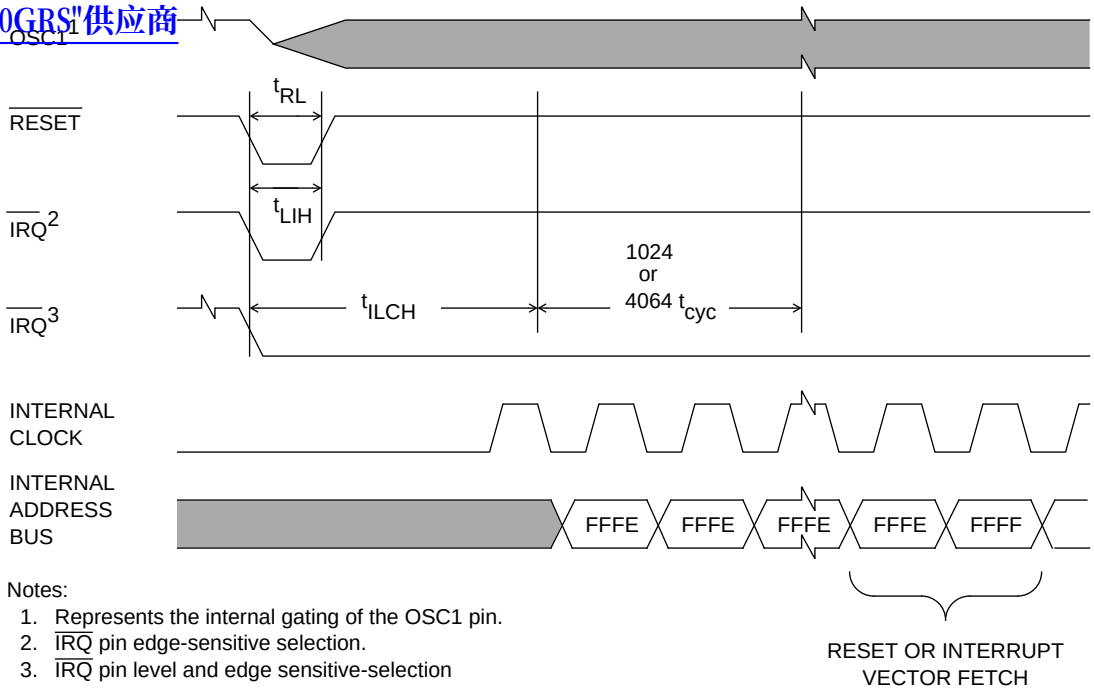
The STOP instruction places the MCU in its lowest power consumption mode. In STOP mode, the internal oscillator is turned off, halting all internal processing, including both timers (Multi-Function Timer with COP and 16-Bit Timer).

When the CPU enters STOP Mode the interrupt flags (TOF and RTIF) and the interrupt enable bits (TOFE and RTIE) in the TCSR are cleared by internal hardware to remove any pending timer interrupt requests and to disable any further timer interrupts. Execution of the STOP instruction automatically clears the I-bit in the Condition Code Register and sets the IRQE enable bit in the External Interrupt Control/Status Register so that the IRQ external interrupt is enabled. All other registers, including the other bits in the TCSR, and memory remain unaltered. All input/output lines remain unchanged. The processor can be brought out of the STOP mode only by an external interrupt, KEY interrupt, or $\overline{\text{RESET}}$.

6.1.1 STOP RECOVERY

A STOP recovery time of 1024 (Fast) or 4064 (Normal) is selectable in the Configuration Register. The crystal or ceramic resonator must assume stable condition before the selected stop recovery time elapses, under all possible operating conditions. This will ensure correct general operation of the 68HC05C0 when STOP mode is exited. See **Figure 6-1 : Stop Recovery Timing Diagram**.

NOTE: The contents of the Configuration Register are reset by any of the five reset conditions. Therefore, if $\overline{\text{RESET}}$ is used to bring the part out of STOP mode, the STOP recovery will always be 4064 cycles, even if the fast (1024) recovery time was selected in the Configuration Register.



Notes:
1. Represents the internal gating of the OSC1 pin.
2. $\overline{IRQ}$ pin edge-sensitive selection.
3. $\overline{IRQ}$ pin level and edge sensitive-selection

Figure 6-1: Stop Recovery Timing Diagram

6.2 WAIT

The WAIT instruction places the MCU in a low-power consumption mode, but the WAIT mode consumes more power than the STOP mode. All CPU action is suspended in WAIT mode. The oscillator, and the timers and SCI if enabled, remain active. Any interrupt or any form of Reset will cause the MCU to exit the WAIT mode.

During the WAIT mode, the I bit in the CCR is cleared to enable interrupts. The IRQE bit in the Configuration Register is set to allow an IRQ interrupt to force the MCU out of WAIT. All other registers, memory, and input/output lines remain in their previous state.

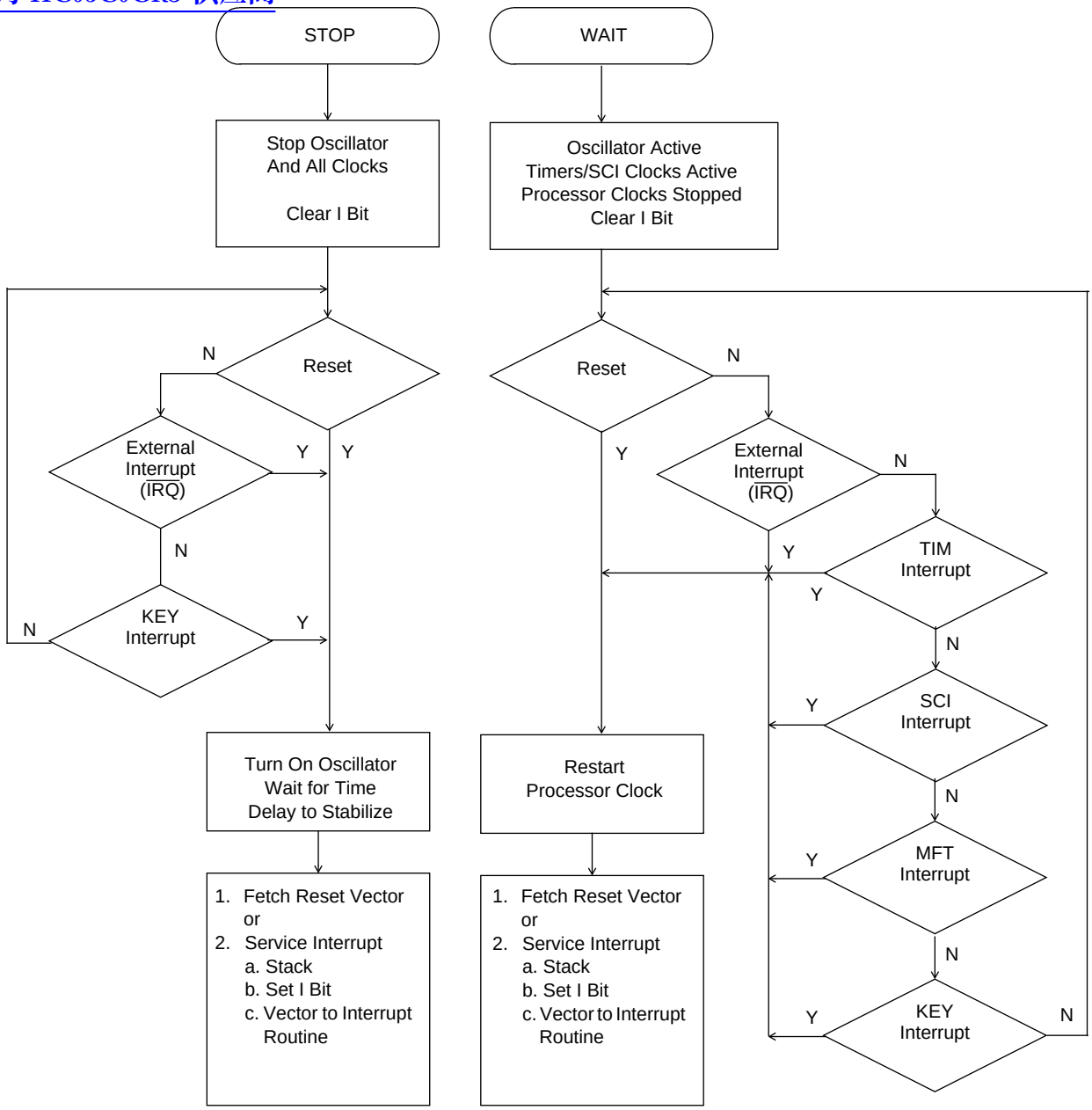


Figure 6-2: STOP/WAIT Flowchart

THIS PAGE INTENTIONALLY LEFT BLANK

SECTION 7 INPUT/OUTPUT PORTS

The MC68HC05C0 has 18 lines arranged as one 8-bit I/O port (Port D), one 6-bit I/O port (Port B), and one 4-bit I/O port (Port C). The I/O ports are programmable as either inputs or outputs under software control of the data direction registers. There is no internal Port A, although an external port replacement unit may be used.

NOTE: To avoid a glitch on the output pins, write data to the I/O Port Data Register before writing a one to the corresponding Data Direction Register.

7.1 PORT B

Port B is a 6-bit bidirectional port which shares its pins with the 16-Bit Timer subsystem, the SCI subsystem and $\overline{CS1}$. See Section 11 16-BIT TIMER and **Section 12, SERIAL COMMUNICATIONS INTERFACE** for a detailed description of these subsystems. See Section 8.2 CHIP SELECTS for more details on $\overline{CS1}$. The address of the Port B data register is $01 and the data direction register (DDR) is at address $05. Reset does not affect the data register, but clears the data direction register, thereby returning the ports to inputs. Writing a one to a DDR bit sets the corresponding port bit to output mode. PB0 has a high current sink and source capability.

7.2 PORT C

Port C is a 4-bit bidirectional port which does not share any of its pins with other subsystems. The address of the Port C data register is $02 and the data direction register (DDR) is at address $06. Reset does not affect the data register, but clears the data direction register, thereby returning the ports to inputs. Writing a one to a DDR bit sets the corresponding port bit to output mode. Port C is not available on the 40 pin PDIP package, and only PC0 and PC2 are available on the 42 pin SDIP package.

7.3 PORT D

Port D is an 8-bit bidirectional port which becomes the lower address byte during Non-Muxed mode. See Section 9 ADDRESS/DATA BUS INTERFACE for a detailed description. The address of the Port D data register is $03 and the data direction register (DDR) is at address $07. Reset does not affect the data register, but clears the data direction register, thereby returning the ports to inputs. Writing a one to a DDR bit sets the corresponding port bit to output mode. When Port D cannot be used as general purpose I/O (Non-Muxed Mode), addresses $03 and $07 are mapped externally. Port D can also be programmed to enable internal pullups and generate an interrupt when any of the 8 I/O lines are pulled low. This requires that the Keyboard Scan is enabled (KSEN) in the External Interrupt Control/ Status Register, and the DDR for any interrupt causing Port D bit is configured as an input. See **Figure 7-1 : Port D Interrupt Option**.

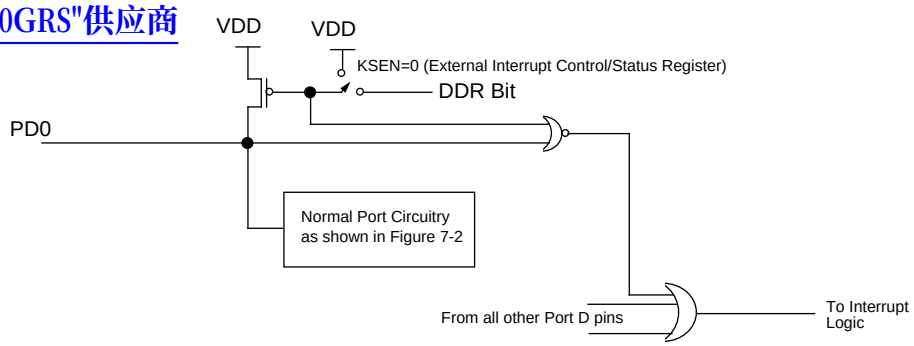


Figure 7-1: Port D Interrupt Option

7.4 INPUT/OUTPUT PROGRAMMING

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

During Reset, all DDRs are cleared, which configure all port pins as inputs. The data direction registers are capable of being written to or read by the processor. During the programmed output state, a read of the data register actually reads the value of the output data latch and not the I/O pin. See **Figure 7-2 : Port I/O Circuitry** and **Table 7-1 : I/O Pin Functions**.

Table 7-1: I/O Pin Functions

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

R/$\overline{W}$ is an internal signal.

NOTE: Port lines shared with subsystems will have their direction controlled by the subsystem if enabled, instead of the port line Data Direction Register. However even with the subsystem enabled, during a read of the port line, the Data Direction Register will still control from where data is read according to **Table 7-1** above (R/$\overline{W}$ = 1).

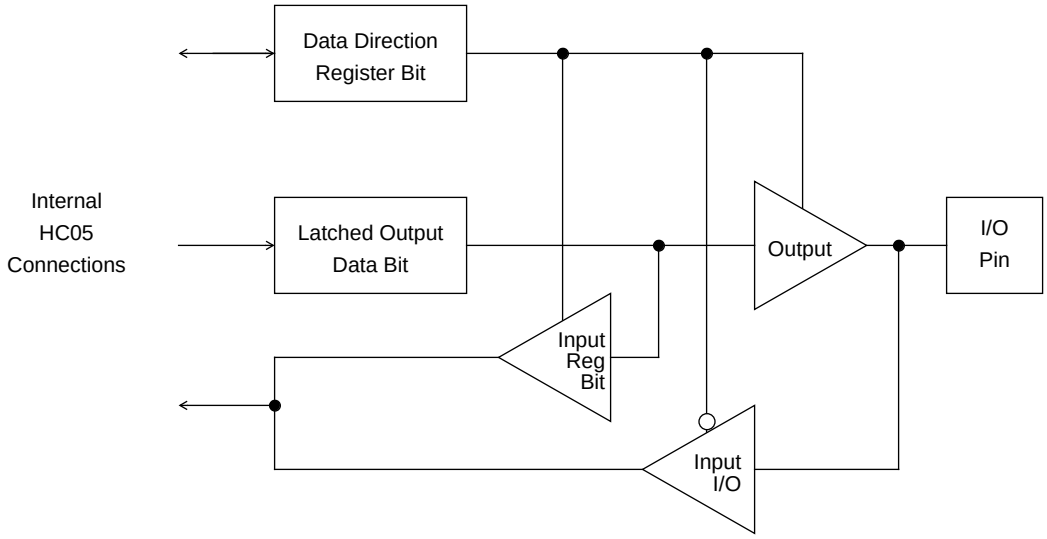


Figure 7-2: Port I/O Circuitry

THIS PAGE INTENTIONALLY LEFT BLANK

SECTION 8 SYSTEM CONFIGURATION

This section describes all the system options. Described are expanded bus modes, chip selects, a Configuration Register, and the External Interrupt Control/Status Register. The Configuration Register includes internal read visibility, load instruction register visibility, clock stretching, COP and Low-Voltage Reset enables, chip-select type and enable, STOP recovery time and enable. The External Interrupt Control/Status Register contains request, enable, acknowledge, and edge-level sensitivity options for both the external interrupt and the KEY interrupt.

8.1 EXPANDED BUS MODES

There are two expanded bus modes available on the MC68HC05C0. The bus mode is specified by the value on the $\overline{\text{LIR}}$/MODE pin. See Section 9.1 MODE SELECTION for more information on selecting the bus mode.

8.1.1 MUXED MODE

In Muxed Mode, the lower order address lines (A7-A0) are muxed with the data lines (D7-D0) on pins 17 - 24 (40 Pin DIP). An address strobe (AS) is available on pin 7 (40 Pin DIP) for external latching of the address. In this configuration, Port D comprises pins 25 - 32 (40 Pin DIP).

8.1.2 NON -MUXED MODE

In Non-Muxed mode, the lower order address lines (A7-A0) are available on pins 25 - 32 (40 Pin DIP) and the lower order data lines (D7-D0) are available on pins 17 - 24 (40 Pin DIP). Pin 7 (40 Pin DIP) is available as an additional chip select ($\overline{\text{CS2}}$).

8.2 CHIP SELECTS

In Non-Muxed Mode, both $\overline{\text{CS1}}$ and $\overline{\text{CS2}}$ are active chip selects. In Muxed Mode, only $\overline{\text{CS1}}$ is available ($\overline{\text{CS2}}$ is shared with AS). In muxed mode the type for $\overline{\text{CS1}}$ can be configured by the user, as specified in the Configuration Register bits CS1P1:0. In Non-Muxed mode $\overline{\text{CS1}}$ and $\overline{\text{CS2}}$ default to a Page-0 chip select type and General Purpose chip select type, respectively. PB5 defaults to a general purpose port pin out of reset. Therefore $\overline{\text{CS1}}$ must be enabled in either mode. The $\overline{\text{A15}}$ chip select is always present.

8.2.1 PAGE-0 CHIP SELECT TYPE

Addresses in page 0 which are mapped external are selected by this chip select type. See Section 2 MEMORY. When the Page-0 chip select type is chosen by CS1P1:0, clock stretching can also be enabled or disabled. Clock stretching will stretch accesses in the address range $0030 through $003F. See Section 9 ADDRESS/DATA BUS INTERFACE for more information on clock stretching.

8.2.2 GENERAL - PURPOSE CHIP SELECT TYPE

The General-Purpose chip select type will select the16-KByte address range from address $4000-$7FFF.

8.2.3 $\overline{A15}$ CHIP SELECT

The $\overline{A15}$ chip select is address bit 15 inverted. This splits the memory map into 32K halves. Inversion of this address bit allows it to be used directly as an active low chip select for program memory with no additional external logic.

8.3 CONFIGURATION REGISTER (CNFGR)

The Configuration Register allows the user to program various system parameters.

		7	6	5	4	3	2	1	0
CNFGR $19	RD / WR	STPEN	STREC	COPEN	IRV	LIRV	LVREN	CS1P1	CS1P0
	RST	0	1	1	0	0	0	1	1

Figure 8-1: Configuration Register

8.3.1 STPEN - SToP ENable

A STOP Enable bit value of logic one will enable STOP mode; a logic zero disables STOP mode. With STOP mode disabled, execution of the Stop instruction will cause the MCU to reset. STPEN is one-time writable. In order to write a new value, the part must first be reset by any of the five reset conditions.

8.3.2 STREC - STop RECovery

The STOP recovery time is the number of bus cycles that elapse after STOP mode is exited and before program operation resumes. See **Figure 6-1 : Stop Recovery Timing Diagram** for more information on STOP recovery. STREC is one-time writable.

0 = 1024 cycle recovery

1 = 4064 cycle recovery

8.3.3 COPEN - Computer Operating Properly ENable

A logic one for the COPEN bit will enable the COP watchdog timer feature. See Section 10.4 COMPUTER OPERATING PROPERLY (COP) WATCHDOG RESET for more information on the COP. COPEN is one-time writable.

8.3.4 IRV - Internal Read Visibility

A logic one in this bit position will turn on internal read visibility. During debugging, internal read visibility should be on. $\overline{RD}$ and $\overline{WR}$ are active for all accesses during IRV to allow creation of an ECLK. Enabled chip selects are active as well, and all internal bus activity is externally visible. In normal user operation, however, IRV should be off. In this case $\overline{RD}$, $\overline{WR}$, and chip selects are not active during internal accesses, preventing possible bus

contention. During internal accesses when IRV is off, the external address pins are driven to the value of the last accessed external address, and the data pins (or muxed address/data pins) are driven with the last external data. Therefore, code operating out of internal RAM will not cause any external bus activity. This is important in reducing power consumption and EMI emissions.

8.3.5 LIRV - Load Instruction Register Visibility

A logic one indicates that $\overline{\text{LIR}}$ will be driven as output on the $\overline{\text{LIR}}$/MODE pin after reset. In this case the user must ensure that the logic level on this pin, used to set bus mode, is no longer actively driven after the rising edge of $\overline{\text{Reset}}$. If LIRV will always be a logic zero, $\overline{\text{LIR}}$/MODE may be hardwired to the desired mode logic level. See Section 9 ADDRESS/DATA BUS INTERFACE for more information on setting the mode. IRV must be set in addition to LIRV for the $\overline{\text{LIR}}$ functionality.

8.3.6 LVREN - Low Voltage Reset ENable

A logic one enables the Low-Voltage Reset (LVR). See Section 5.3 LOW-VOLTAGE RESET for more information.

8.3.7 CS1P1:CS1P0 - Chip Select 1 Parameters

$\overline{\text{CS1}}$ shares a pin with the general purpose port bit PB5. The pin type and chip select type for $\overline{\text{CS1}}$ are determined by the value of the Chip Select Parameter bits as shown in the table below. If $\overline{\text{CS1}}$ is enabled, and the Page-0 type is chosen, clock stretching may be enabled for the address range $0030 through $003F. See Section 9 ADDRESS/DATA BUS INTERFACE for more information on clock stretching.

Table 8-1: Chip Select Types

CS1P1:CS1P0	PIN TYPE	CS1 TYPE	STRETCHING
00	Chip Select $\overline{\text{CS1}}$	Page-0	Disabled
01	Chip Select $\overline{\text{CS1}}$	Page-0	Enabled
10	Chip Select $\overline{\text{CS1}}$	General Purpose	❇
11	Port PB5	❇	❇

8.4 EXTERNAL INTERRUPT CONTROL/STATUS REGISTER (EICSR)

The External Interrupt Control/Status Register provides control and status bits for the external interrupt and the KEY interrupt.

		7	6	5	4	3	2	1	0
EICSR $1A	RD	KSF	KSEN	0	KSEL	IRQF	IRQEN	0	IRQEL
	WR			KSA				IRQA	
	RST	0	0	0	0	0	1	0	0

Figure 8-2: External Interrupt Control/Status Register

8.4.1 KSF - Keyboard Scan request Flag

If the keyboard scan request flag is set, this indicates that a KEY request is pending. This flag can only be set if KSEN is logic one. Writing to this bit has no effect. The KSF flag is cleared by writing to the keyboard scan acknowledge (KSA). While servicing a KEY interrupt, a pending interrupt request flag can be cleared by writing a logic one to the KSA bit. In this case, if edge-level sensitive triggering is enabled, the KSEN should be cleared as well in order to prevent re-entering the interrupt service routine.

8.4.2 KSEN - Keyboard Scan ENable

The keyboard scan feature is included to reduce external keyboard scan hardware. A logic one enables the keyboard scan feature, which will generate a CPU interrupt if any of the lines on the Port D pins are pulled low when their corresponding data registers are configured as inputs. KSEN must be set in order to allow the keyboard wake-up to exit STOP or WAIT mode.

8.4.3 Keyboard Scan Acknowledge

Writing a logic one to the KSA bit clears the keyboard scan latch. If KSEL is zero, KSF will be cleared as well. Writing a zero has no effect. This bit always reads as zero.

8.4.4 Keyboard Scan Edge/Level

A logic one indicates edge and level sensitive triggering for a keyboard interrupt request. A logic zero indicates falling-edge triggering only. This bit is one-time writable; in order to write a new value, the part must first be externally reset.

8.4.5 IRQF - Interrupt ReQuest Flag

If the interrupt request flag is set, this indicates that a $\overline{IRQ}$ request is pending. The flag can be set regardless of the state of IRQEN. Writing to this bit has no effect. The IRQF is automatically cleared when the $\overline{IRQ}$ vector is fetched. While servicing an $\overline{IRQ}$ interrupt, a pending interrupt request flag can be cleared by writing a logic one to the IRQA bit. In this case, if edge-level sensitive triggering is enabled, the IRQEN should be cleared as well in order to prevent re-entering the interrupt service routine.

8.4.6 IRQEN - Interrupt ReQuest ENable

A logic one for IRQEN enables the IRQF to initiate an $\overline{IRQ}$ interrupt sequence. If IRQEN is a logic zero, the interrupt sequence cannot be generated, thus allowing the $\overline{IRQ}$ interrupt to be masked without having to set the I bit to disable all interrupts. Execution of the STOP

or WAIT instructions causes the IRQEN bit to be set in order to allow the external $\overline{IRQ}$ to exit these modes.

8.4.7 IRQA - Interrupt ReQuest Acknowledge

Writing a logic one to the IRQA bit clears the $\overline{IRQ}$ latch. If IRQEL is zero, IRQF will be cleared as well. This allows a pending IRQ interrupt and/or flag to be cleared without having to enter the $\overline{IRQ}$ service routine. Writing a zero has no effect. This bit always reads as zero.

8.4.8 IRQEL - Interrupt ReQuest Edge/Level

A logic one indicates edge and level sensitive triggering for the $\overline{IRQ}$ interrupt; a logic zero indicates negative-edge triggering only. This bit is one-time writable between resets; in order to write a new value, the part must first be externally reset.

THIS PAGE INTENTIONALLY LEFT BLANK

SECTION 9 ADDRESS/DATA BUS INTERFACE

Bus timing is determined by the chosen expanded bus mode, and whether clock stretching is active. The internal bus frequency is a divide-by-four of the oscillator frequency OSC1.

9.1 MODE SELECTION

The 68HC05C0 has two possible expanded bus modes: Muxed Mode and Non-Muxed Mode. In Muxed Mode, the data is multiplexed with the lower order 8 address bits. In Non-Muxed Mode, address and data have individual, separate output. The mode is determined by the logic value on the $\overline{\text{LIR}}$/MODE pin at the rising edge of $\overline{\text{RESET}}$ as shown below.

Table 9-1: Bus Modes

MODE PIN	BUS MODE
0	Non-Muxed
1	Muxed

Once the mode is established at the rising edge of reset, this pin can be used as $\overline{\text{LIR}}$. See Section 8 SYSTEM CONFIGURATION. If $\overline{\text{LIR}}$ is not desired, the MODE pin may be hardwired. $\overline{\text{LIR}}$ always follows the same timing as the upper address byte A15:A8.

9.2 MUXED MODE BUS TIMING

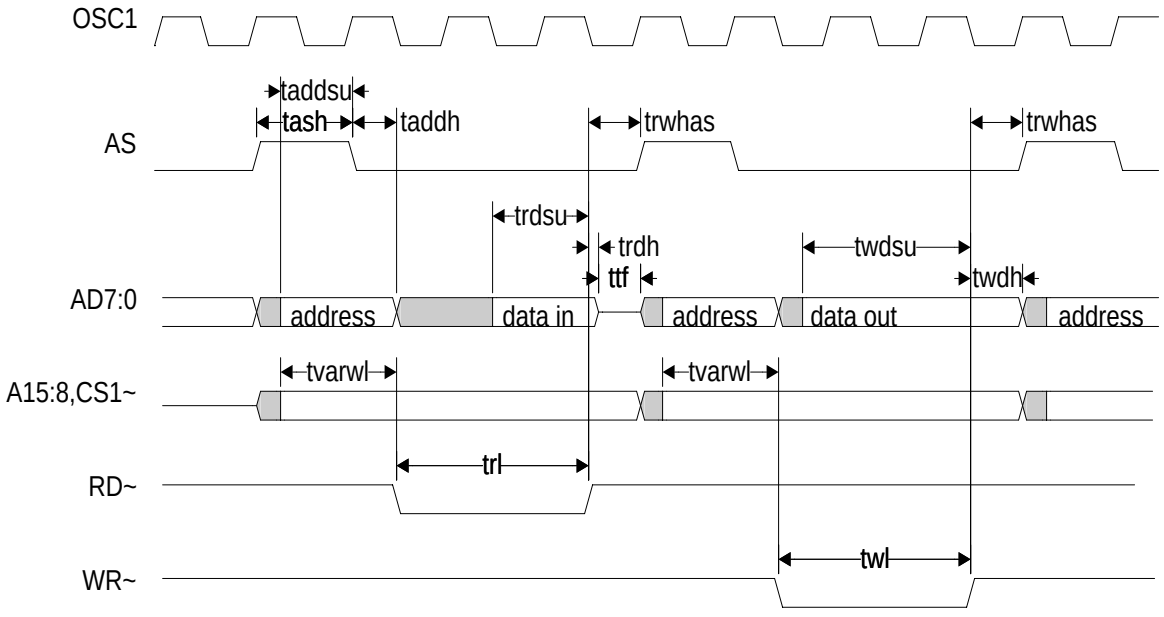


Figure 9-1: Muxed Mode Timing

Table 9-2: Muxed Mode Timing

symbol	characteristic	bus time in nanoseconds		
		min	typ	max
tash	address strobe high pulse width	TBD	2*Tosc/TBD	TBD
taddsu	address strobe setup time	TBD	2*Tosc/TBD	TBD
taddh	address strobe hold time	TBD	Tosc/TBD	TBD
tvarwl	address valid to rd~ or wr~ fall	TBD	3*Tosc/TBD	TBD
trl	rd~ low pulse width	TBD	4*Tosc/TBD	TBD
trdsu	read data setup time to rd~ rise	TBD	10ns/TBD	TBD
trdh	read data hold time after rd~ rise	TBD	0/TBD	TBD
ttf	time to float for external peripherals	TBD	Tosc/TBD	TBD
twl	wr~ low pulse width	TBD	4*Tosc/TBD	TBD
twdsu	write data setup time to wr~ rise	TBD	4*Tosc/TBD	TBD
twdh	write data hold time after wr~ rise	TBD	Tosc/TBD	TBD

Tosc is one-half oscillator period
(Bus cycle = 4x oscillator period)

9.3 NON-MUXED MODE BUS TIMING

In Non-Muxed Mode, bus AD7:0 contains only the bidirectional 8 bit data bus.

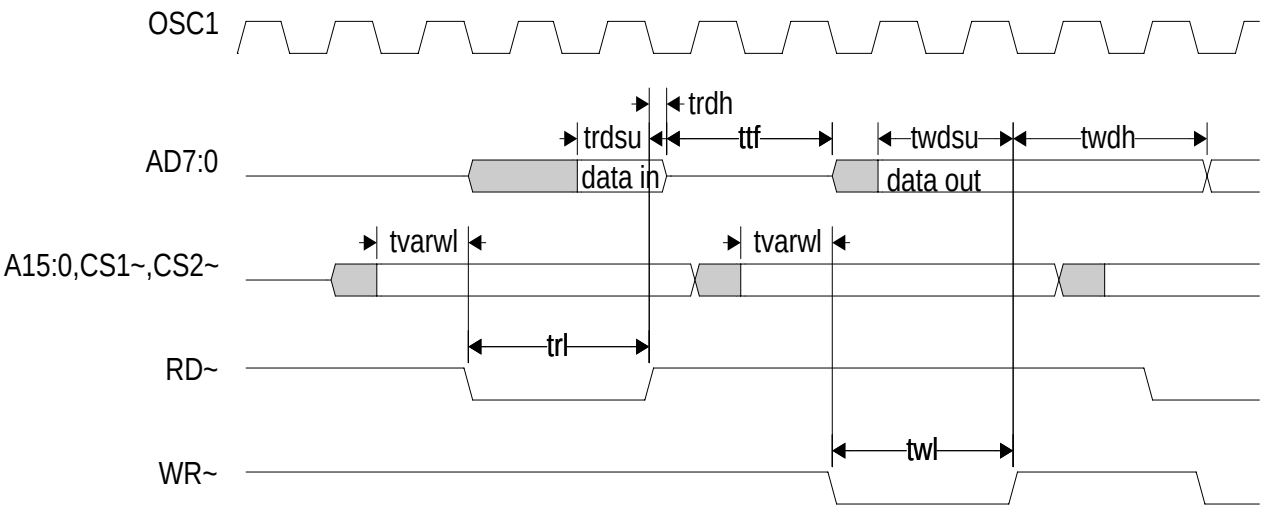


Figure 9-2: Non-Muxed Mode Timing

Table 9-3: Non-Muxed Mode Timing

symbol	characteristic	bus time in nanoseconds		
		min	typ	max
tvarwl	address valid to rd~ or wr~ fall	TBD	3*Tosc/TBD	TBD
trl	rd~ low pulse width	TBD	4*Tosc/TBD	TBD
trdsu	read data setup time to rd~ rise	TBD	10ns/TBD	TBD
trdh	read data hold time after rd~ rise	TBD	0/TBD	TBD
ttf	time to float for external peripherals	TBD	4*Tosc/TBD	TBD
twl	wr~ low pulse width	TBD	4*Tosc/TBD	TBD
twdsu	write data setup time to wr~ rise	TBD	4*Tosc/TBD	TBD
twdh	write data hold time after wr~ rise	TBD	4*Tosc/TBD	TBD

Tosc is one-half oscillator period
(Bus cycle = 4x oscillator period)

9.4 CLOCK STRETCHED BUS TIMING

If $\overline{CS1}$ is configured as a Page-0 chip select type, clock stretching can be enabled to accommodate slow external peripherals. See Section 8.3.7 CS1P1:CS1P0 - Chip Select 1 Parameters. In either muxed or non-muxed mode, the access/write time, time to float and data hold time are increased with clock stretching as shown in the timing diagrams below. In these diagrams, OSC1 is the oscillator clock. The internal bus frequency is a divide-by-four of the oscillator frequency.

9.4.1 Muxed Mode Clock Stretched Read Cycle

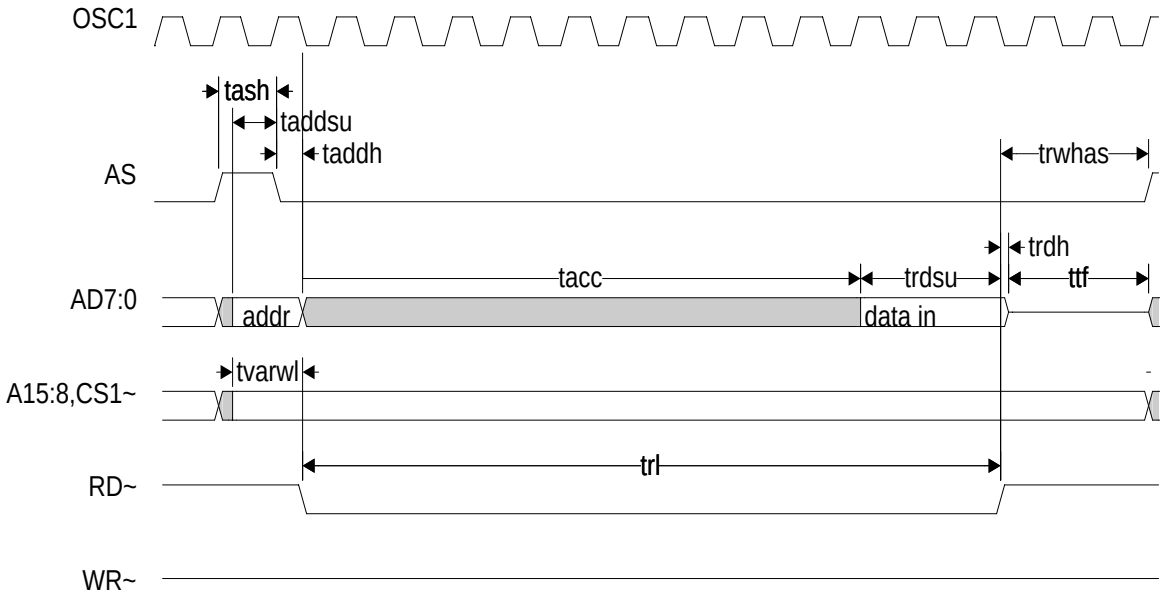


Figure 9-3: Muxed Mode Clock Stretched Read Cycle

9.4.2 Muxed Mode Clock Stretched Write Cycle

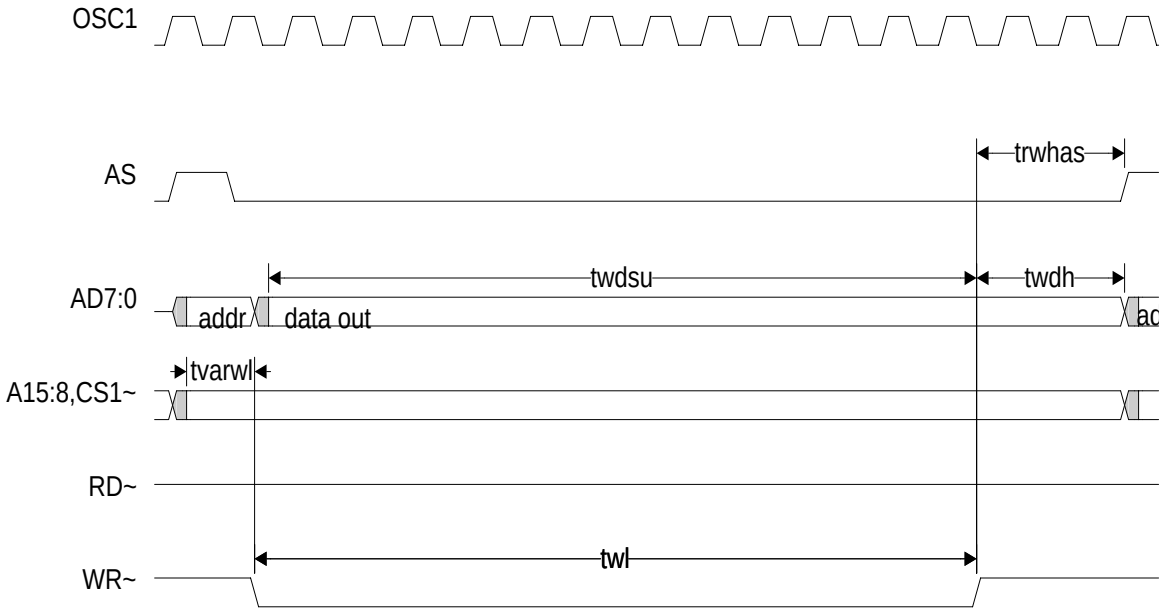


Figure 9-4: Muxed Mode Clock Stretched Write Cycle

9.4.3 Non-Muxed Mode Clock Stretched Read Cycle

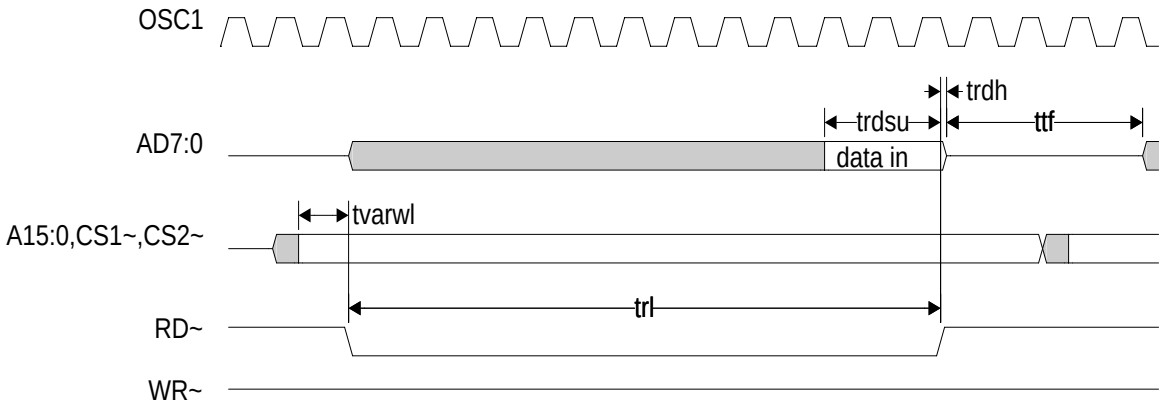


Figure 9-5: Non-Muxed Mode Clock Stretched Read Cycle

9.4.4 Non-Muxed Mode Clock Stretched Write Cycle

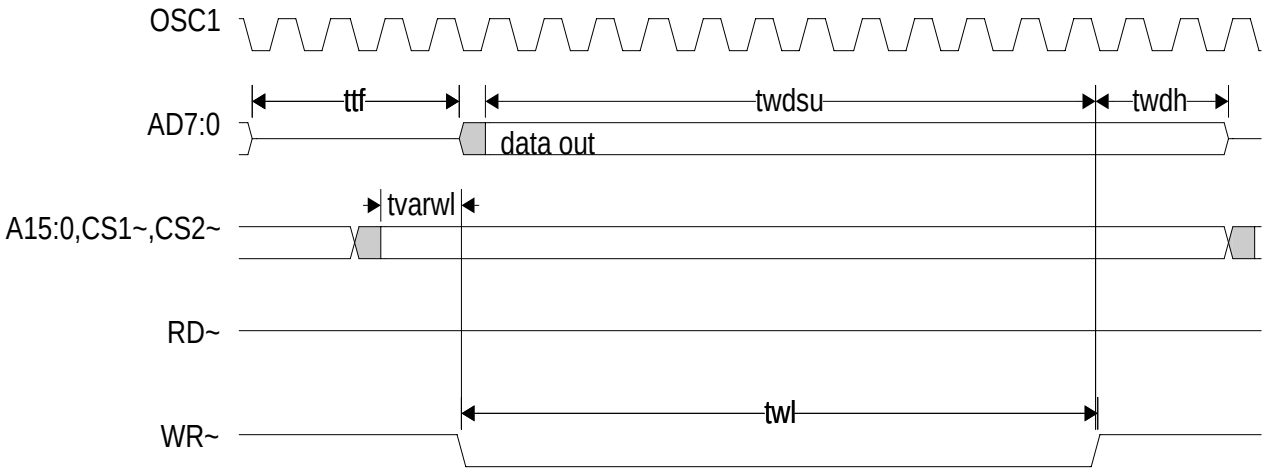


Figure 9-6: Non-Muxed Mode Clock Stretched Write Cycle

Table 9-4: Clock Stretched Muxed Mode Timing

symbol	characteristic	bus time in nanoseconds		
		min	typ	max
tash	address strobe high pulse width	TBD	2*Tosc/TBD	TBD
taddsu	address strobe setup time	TBD	2*Tosc/TBD	TBD
taddh	address strobe hold time	TBD	Tosc/TBD	TBD
tvarwl	address valid to rd~ or wr~ fall	TBD	3*Tosc/TBD	TBD
trl	rd~ low pulse width	TBD	24*Tosc/TBD	TBD
trdsu	read data setup time to rd~ rise	TBD	10ns/TBD	TBD
trdh	read data hold time after rd~ rise	TBD	0/TBD	TBD
ttf	time to float for external peripherals	TBD	5*Tosc/TBD	TBD
twl	wr~ low pulse width	TBD	24*Tosc/TBD	TBD
twdsu	write data setup time to wr~ rise	TBD	24*Tosc/TBD	TBD
twdh	write data hold time after wr~ rise	TBD	5*Tosc/TBD	TBD

Tosc is one-half oscillator period
(Bus cycle = 4x oscillator period)

Table 9-5: Clock Stretched Non-Muxed Mode Timing

symbol	characteristic	bus time in nanoseconds		
		min	typ	max
tvarwl	address valid to rd~ or wr~ fall	TBD	3*Tosc/TBD	TBD
trl	rd~ low pulse width	TBD	24*Tosc/TBD	TBD
trdsu	read data setup time to rd~ rise	TBD	10ns/TBD	TBD
trdh	read data hold time after rd~ rise	TBD	0/TBD	TBD
ttf	time to float for external peripherals	TBD	8*Tosc/TBD	TBD
twl	wr~ low pulse width	TBD	24*Tosc/TBD	TBD
twdsu	write data setup time to wr~ rise	TBD	24*Tosc/TBD	TBD
twdh	write data hold time after wr~ rise	TBD	8*Tosc/TBD	TBD

Tosc is one-half oscillator period
(Bus cycle = 4x oscillator period)

9.5 BUS CYCLE CHARACTERISTICS

There are a few points to be made regarding bus activity which are independent of mode or cycle stretching. First, as in all 68HC05 microcontrollers, any write cycle is preceded by a “dummy” read cycle. Although this is transparent to the user with other 68HC05 parts, the 68HC05C0 expanded bus reveals this phenomena. Another note of interest is that if IRV is not enabled, internal bus activity does not cause changes on the external address or data pins, or the $\overline{RD}$, $\overline{WR}$ or chip select outputs. See Section 8.3.4 IRV - Internal Read Visibility for more information on IRV. The reason for this behavior is to reduce unnecessary external switching which contributes to power consumption and noise. For the same reasons, effective address calculation bus cycles which force the internal address bus to $XXFE are not reflected externally, unless IRV is enabled.

9.6 ADDRESS/DATA BUS DURING WAIT AND STOP MODE

The CPU halts during WAIT or STOP mode, which suspends bus activity. The address bus will hold the state of the address following the WAIT or STOP instruction. If the part is operating in Muxed Mode, the muxed address/data bus is driven with the lower order address following the WAIT or STOP instruction. If the part is operating in Non-Muxed Mode, the data bus will be driven to the opcode of the instruction following the WAIT or STOP instruction. When the processor exits WAIT or STOP mode, the bus activity will resume.

NOTE: When executing a program out of external memory, $\overline{RD}$ will remain low during STOP mode and will become high again at the beginning of the STOP recovery. This implies that the external memory will be driving data for the duration of STOP. If it is not desired that the external memory be active for the entire STOP duration, the STOP instruction should be executed out of internal RAM with internal read visibility disabled.

THIS PAGE INTENTIONALLY LEFT BLANK

SECTION 10 MULTI-FUNCTION TIMER

10.1 INTRODUCTION

This timer is a 15-stage multi-function ripple counter. The features include Timer Over Flow, Power-On Reset (POR), Real Time Interrupt, and COP Watchdog Timer.

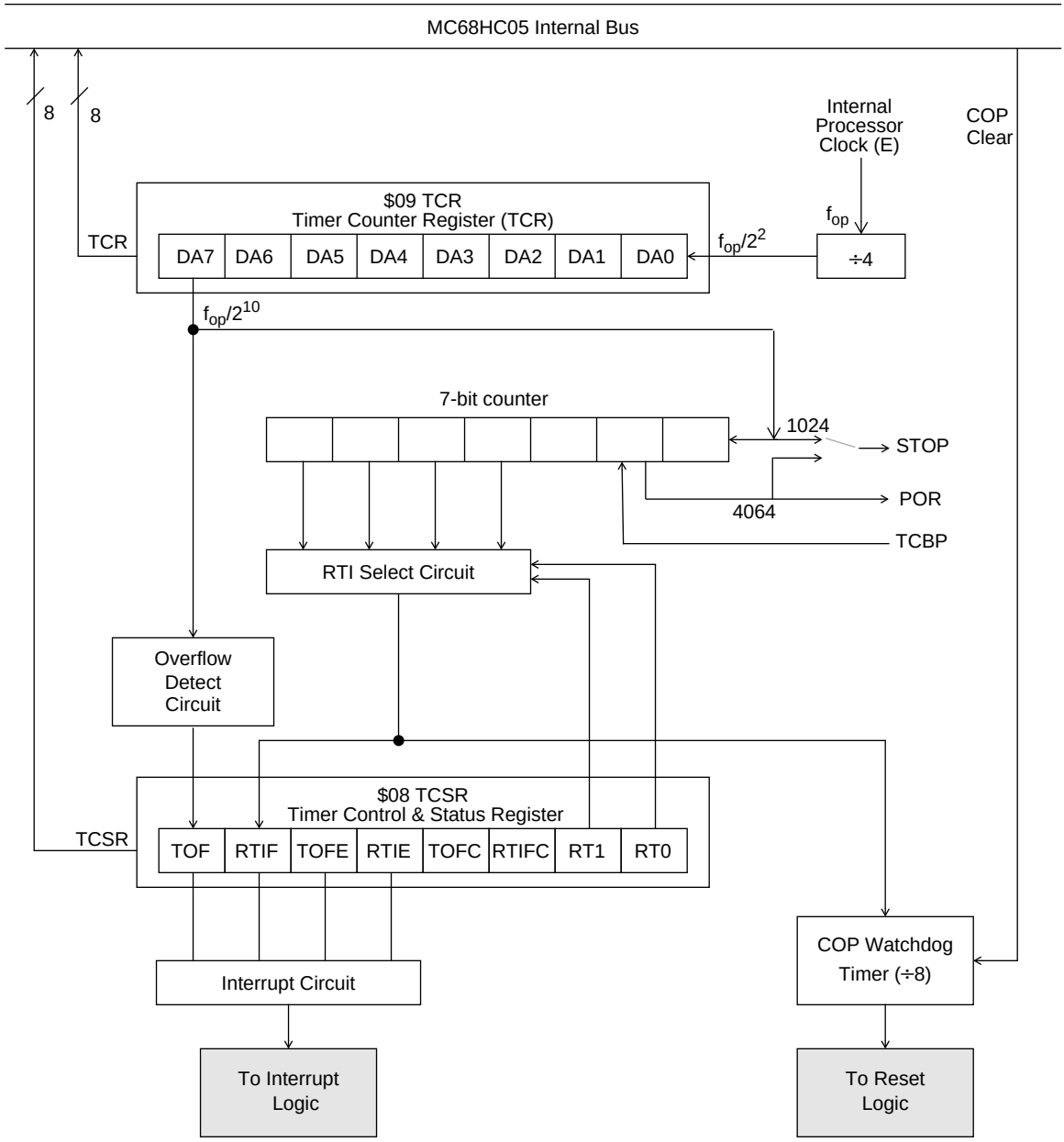


Figure 10-1: Multi-Function Timer Block Diagram

As seen in **Figure 10-1 : Multi-Function Timer Block Diagram**, the Timer begins with a fixed divide-by-four prescaler. This signal drives an 8-bit ripple counter. The value of this 8-bit ripple counter can be read by the CPU at any time by accessing the Timer Counter Register (TCR).

A timer overflow function is implemented on the last stage of this counter, giving a possible periodic interrupt at the rate of $f_{op}/1024$.

Two additional stages produce the Timer Counter Bypass circuitry, at $f_{op}/4064$, (available only in Test Mode) in the timer chain.

This circuit is followed by two more stages, with the resulting clock ($f_{op}/16384$) driving the Real Time Interrupt circuit. The RTI circuit consists of three divider stages with a 1 of 4 selector.

The output of the RTI circuit is further divided by eight to drive the COP Watchdog Timer circuit.

The RTI rate selector bits, the RTI and TOF enable bits, the RTI and TOF flags and clearing mechanisms are located in the Timer Control and Status Register.

The STOP mode recovery timing is selected from the $f_{op}/4064$ and $f_{op}/1024$ signals. The POR function uses the $f_{op}/4064$.

10.2 TIMER CONTROL AND STATUS REGISTER (TCSR)

The TCSR contains the timer interrupt flag, the timer interrupt enable bits, the timer interrupt clearing mechanisms and the real time interrupt rate select bits. **Figure 10-2 : Timer Control and Status Register (TCSR)** shows the value of each bit in the TCSR when coming out of reset.

	7	6	5	4	3	2	1	0
TCSR $08 RD	TOF	RTIF	TOFE	RTIE	TOFC	RTIFC	RT1	RT0
TCSR $08 WR								
RST	0	0	0	0	0	0	1	1

Figure 10-2: Timer Control and Status Register (TCSR)

10.2.1 TOF - Timer Over Flow

TOF is a read-only status bit and is set when the 8-bit ripple counter rolls over from $FF to $00. A CPU interrupt request will be generated if TOFE is set. Clearing the TOF is done by writing a logic 1 to TOFC. Writing to TOF has no effect. A reset clears TOF.

10.2.2 RTIF - Real Time Interrupt Flag

The Real Time Interrupt circuit consists of a three stage divider and a 1 of 4 selector. The clock frequency that drives the RTI circuit is E/2**14 (or E/16384) with three additional divider stages giving a maximum interrupt period of 65.5 milliseconds at a bus rate of 2 MHz. RTIF is a read-only status bit and is set when the output of the chosen (1 of 4

selection) stage goes active. A CPU interrupt request will be generated if RTIE is set. Clearing the RTIF is done by writing a '1' to RTIFC. Writing to RTIF has no effect. Reset clears RTIF.

10.2.3 TOFE - Timer Over Flow Enable

When this bit is set, a CPU interrupt request is generated when the TOF bit is set. When this bit is clear, the TOF flag is prevented from generating an interrupt request. A reset clears this bit.

10.2.4 RTIE - Real Time Interrupt Enable

When this bit is set, a CPU interrupt request is generated when the RTIF bit is set. When this bit is clear, the RTIF flag is prevented from generating an interrupt request. A reset clears this bit.

10.2.5 TOFC - Timer Over Flow Flag Clear

TOF is cleared by writing a logic 1 to TOFC. Writing a logic 0 to TOFC has no effect on TOF. This bit always reads as logic 0.

10.2.6 RTIFC - Real Time Interrupt Flag Clear

RTIF is cleared by writing a logic 1 to RTIFC. Writing a logic 0 to RTIFC has no effect on RTIF. This bit always reads as logic 0.

10.2.7 RT1:RT0 - Real Time Interrupt Rate Select

These two bits select one of four taps from the Real Time Interrupt circuit. **Table 10-1** shows the available interrupt rates with a variety of oscillator frequencies. A reset sets these two bits which selects the lowest periodic rate and gives the maximum time in which to alter these bits if necessary. Care should be taken when altering RT0 and RT1 if the time-out period is imminent or uncertain. If the selected tap is modified during a cycle in which the counter is switching, an RTIF could be missed or an additional one could be generated. To avoid problems, the COP should be cleared before changing RTI taps.

The encoding for these two bits is shown below.

Table 10-1: RTI Rates

	RTI RATES AT f_{osc} FREQUENCY SPECIFIED:	
RT1:RT0	**8.0 MHz**	**16.0 MHz**
00	8.192 ms	4.096 ms
01	16.384 ms	8.192 ms
10	32.768 ms	16.384 ms
11	65.536 ms	32.768 ms

10.3 TIMER COUNTER REGISTER (TCR)

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

TCR $09

	7	6	5	4	3	2	1	0
RD	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0
WR								
RST	0	0	0	0	0	0	0	0

Figure 10-3: Timer Counter Register

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

10.4 COMPUTER OPERATING PROPERLY (COP) WATCHDOG RESET

The COP watchdog timer function is implemented on this device by using the output of the RTI circuit and further dividing it by eight.

The COP is enabled out of reset. The COP can be disabled by writing a logic 0 to the COPEN bit in the Configuration Register.

If the COP circuit times out, an internal reset is generated and the normal reset vector is fetched. $\overline{\text{RESET}}$ is also driven low to reset external peripherals. Preventing a COP time-out is done by writing a logic 0 to bit 0 of address $FFFF at the minimum reset rate. When the COP is cleared, only the final divide by eight stage (output of the RTI) is cleared.

The minimum COP reset rates are listed in **Table 10-2** below. Because it is not readily possible to determine the state of the divider chain ahead of the COP circuit, the COP should be reset within a period equivalent to **seven** real-time interrupts, rather than the eight that one might expect.

Table 10-2: Minimum COP Reset Times

	MINIMUM COP RESET AT f_{osc} FREQUENCY SPECIFIED:	
RT1:RT0	**8.0 MHz**	**16.0 MHz**
00	57.344 ms	28.672 ms
01	114.688 ms	57.344 ms
10	229.376 ms	114.688 ms
11	458.752 ms	229.376 ms

10.5 TIMER DURING WAIT MODE

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

10.6 TIMER DURING STOP MODE

The timer is cleared when going into STOP mode. In addition, the interrupt flags (TOF and RTIF) and the interrupt enable bits (TOFE and RTIE) in the TCSR are cleared by internal hardware to remove any pending timer interrupt requests and to disable any further timer interrupts.When STOP is exited by an external interrupt or an external $\overline{\text{RESET}}$, the internal oscillator will resume, followed by an internal processor oscillator stabilization delay (stop recovery time). The timer is then cleared and operation resumes.

THIS PAGE INTENTIONALLY LEFT BLANK

SECTION 11 16-BIT TIMER

11.1 INTRODUCTION

The timer consists of a 16-bit, software-programmable counter driven by a fixed divide-by-four prescaler. This timer can be used for many purposes, including input waveform measurements while simultaneously generating an output waveform. Pulse widths can vary from several microseconds to many seconds. See **Figure 11-1** below for a block diagram of the 16-Bit Timer.

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

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

11.2 COUNTER

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

The double-byte, free-running counter can be read from either of two locations, $10-$11 (counter register) or $12-$13 (counter alternate register). A read from only the least significant byte (LSB) of the free-running counter ($11, $13) receives the count value at the time of the read. If a read of the free-running counter or counter alternate register first addresses the most significant byte (MSB) ($10, $12), the LSB ($11, $13) is transferred to a buffer. This buffer value remains fixed after the first MSB read, even if the user reads the MSB several times. This buffer is accessed when reading the free-running counter or counter alternate register LSB ($11 or $13) and, thus, completes a read sequence of the total counter value. In reading either the free-running counter or counter alternate register, if the MSB is read, the LSB must also be read to complete the sequence.

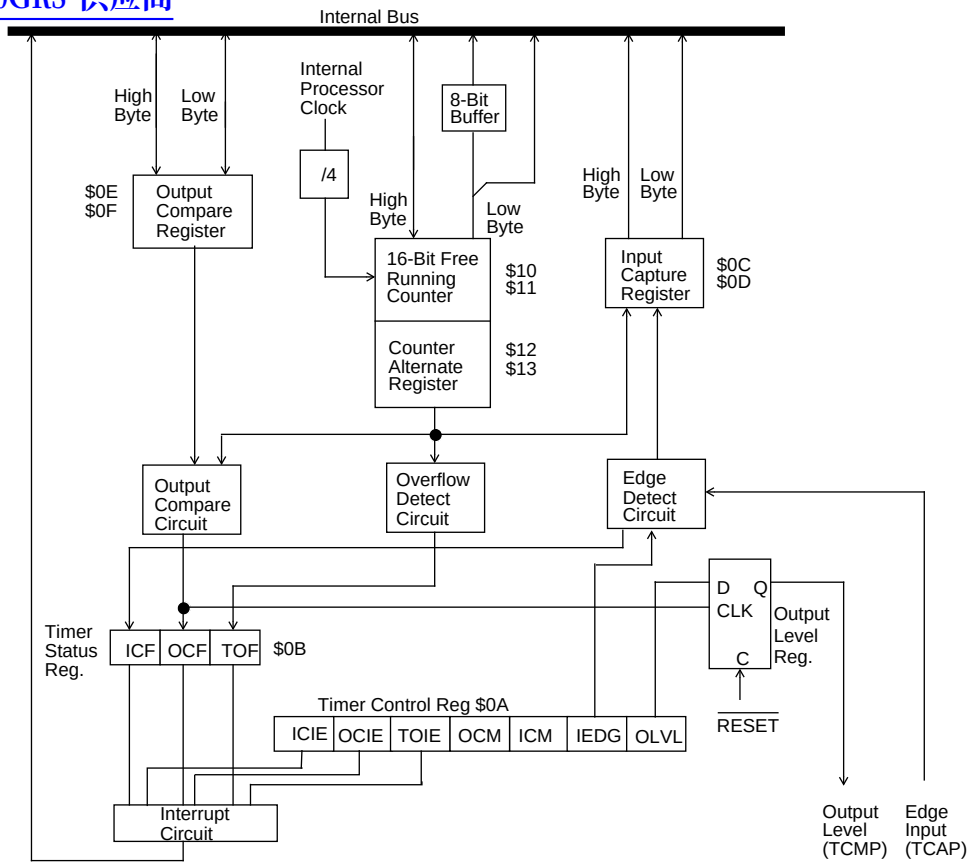


Figure 11-1: Timer Block Diagram

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

The free-running counter is configured to $FFFC (to facilitate testing TOF) during reset and is always a read-only register. During a power-on reset, the counter is also preset to $FFFC and begins running after the oscillator start-up delay. Because the free-running counter is 16 bits preceded by a fixed divided-by-four prescaler, the value in the free-running counter repeats every 262,144 internal bus clock cycles. When the counter rolls over from $FFFF to $0000, the TOF bit is set and an interrupt will result if the timer overflow interrupt enable bit (TOIE) is set.

CNTR MSB $10

	7	6	5	4	3	2	1	0
RD	CNT15	CNT14	CNT13	CNT12	CNT11	CNT10	CNT9	CNT8
WR								
RST	1	1	1	1	1	1	1	1

Figure 11-2: Timer Counter MSB Register

CNTR LSB $11

	7	6	5	4	3	2	1	0
RD	CNT7	CNT6	CNT5	CNT4	CNT3	CNT2	CNT1	CNT0
WR								
RST	1	1	1	1	1	1	0	0

Figure 11-3: Timer Counter LSB Register

ACNTR MSB $12

	7	6	5	4	3	2	1	0
RD	CNT15	CNT14	CNT13	CNT12	CNT11	CNT10	CNT9	CNT8
WR								
RST	1	1	1	1	1	1	1	1

Figure 11-4: Timer Alternate Counter MSB Register

ACNTR LSB $13

	7	6	5	4	3	2	1	0
RD	CNT7	CNT6	CNT5	CNT4	CNT3	CNT2	CNT1	CNT0
WR								
RST	1	1	1	1	1	1	0	0

Figure 11-5: Timer Alternate Counter LSB Register

11.3 OUTPUT COMPARE REGISTER

The 16-bit output compare register is made up of two 8-bit registers at locations $0E (MSB) and $0F (LSB). The output compare register is used for several purposes, such as indicating when a period of time has elapsed. All bits are readable and writable and are not altered by the timer hardware or reset. If the compare function is not needed, the two bytes of the output compare register can be used as storage locations.

The output compare register contents are continually compared with the contents of the free-running counter, and if a match is found, the output compare flag (OCF) bit is set and the output level (OLVL) bit is clocked to an output level register. The output compare register values and the output level bit should be changed after each successful

comparison to establish a new elapsed timeout. An interrupt can also accompany a successful output compare provided the interrupt enable bit (OCIE) is set.

After a processor write cycle to the output compare register containing the MSB ($0E), the output compare function is inhibited until the LSB ($0F) is also written. The user must write both bytes (locations) if the MSB is written first. A write made only to the LSB ($0F) will not inhibit the compare function. The free-running counter is updated every four internal bus clock cycles. The minimum time required to update the output compare register is a function of the program rather than the internal hardware.

The processor can write to either byte of the output compare register without affecting the other byte. The output level (OLVL) bit is clocked to the output level register even if the output compare flag (OCF) is still set from a previous compare.

OCR MSB $0E

	7	6	5	4	3	2	1	0
RD WR	DA15	DA14	DA13	DA12	DA11	DA10	DA9	DA8
RST	U	U	U	U	U	U	U	U

Figure 11-6: Timer Output Compare MSB Register

OCR LSB $0F

	7	6	5	4	3	2	1	0
RD WR	DA15	DA14	DA13	DA12	DA11	DA10	DA9	DA8
RST	U	U	U	U	U	U	U	U

Figure 11-7: Timer Output Compare LSB Register

11.4 INPUT CAPTURE REGISTER

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

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

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

After a read of the input capture register ($0C) MSB, the counter transfer is inhibited until the LSB ($0D) is also read. This characteristic causes the time used in the input capture software routine and its interaction with the main program to determine the minimum pulse period.

A read of the input capture register LSB ($0D) does not inhibit the free-running counter transfer since they occur on opposite edges of the internal bus clock.

ICR MSB $0C

	7	6	5	4	3	2	1	0
RD / WR	DA15	DA14	DA13	DA12	DA11	DA10	DA9	DA8
RST	U	U	U	U	U	U	U	U

Figure 11-8: Timer Input Capture MSB Register

ICR LSB $0D

	7	6	5	4	3	2	1	0
RD / WR	DA15	DA14	DA13	DA12	DA11	DA10	DA9	DA8
RST	U	U	U	U	U	U	U	U

Figure 11-9: Timer Input Capture LSB Register

11.5 TIMER CONTROL REGISTER (TCR)

The TCR is a read/write register containing seven control bits. Three bits control interrupts associated with the timer status register flags ICF, OCF and TOF.

TCR $0A

	7	6	5	4	3	2	1	0
RD / WR	ICIE	OCIE	TOIE	OCM	ICM	0	IEDG	OLVL
RST	0	0	0	0	0	0	0	0

Figure 11-10: Timer Control Register

ICIE - Input Capture Interrupt Enable

1 = Interrupt enabled
0 = Interrupt disabled

OCIE - Output Compare Interrupt Enable

1 = Interrupt enabled
0 = Interrupt disabled

TOIE - Timer Overflow Interrupt Enable

1 = Interrupt enabled
0 = Interrupt disabled

OCM- Output Compare Mode

This bit allows Port B bit 0 to become the Timer Output Level (TCMP pin) regardless of the state of the Port B bit 0 DDR bit. Note that the output compare functionality will still be active when OCM is clear; however, TCMP will not be available externally.

1 = Port B bit 0 = TCMP
0 = Port B bit 0 = General Purpose I/O

ICM- Input Capture Mode

This bit allows Port B bit 1 to become the Timer Edge Input (TCAP pin) regardless of the state of the Port B bit 1 DDR bit. TCAP has hysteresis to improve noise immunity. Note that the input capture functionality will still be active when ICM is clear; however, the port will no longer be TCAP. Therefore, care must be taken if port B bit 1 is a general purpose I/O since it can still cause ICF to be set if the right transition occurs on the port pin (according to IEDG). This implies that port B bit 1 can be made an output and manually toggled by software to cause an input capture interrupt to occur if ICIE is set.

1 = Port B bit 1 = TCAP
0 = Port B bit 1 = General Purpose I/O

IEDG - Input Edge

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

1 = Positive edge
0 = Negative edge

Reset does not affect the IEDG bit (unaffected).

OLVL - Output Level

Value of output level is clocked into output level register by the next successful output compare and will appear on the TCMP pin. The OCM should be set for the TCMP to be visible externally.

1 = High output
0 = Low output

Bit 2 - Not used

Always reads zero

11.6 TIMER STATUS REGISTER (TSR)

The TSR is a read-only register containing three status flag bits.

TSR $0B

	7	6	5	4	3	2	1	0
RD	ICF	OCF	TOF					
WR								
RST	0	0	0	0	0	0	0	0

Figure 11-11: Timer Status Register

ICF - Input Capture Flag

1 = Flag set when selected polarity edge is sensed by input capture edge detector
0 = Flag cleared when TSR and input capture low register ($0D) are accessed

OCF - Output Compare Flag

1 = Flag set when output compare register contents match the free-running counter contents
0 = Flag cleared when TSR and output compare low register ($0F) are accessed

TOF - Timer Overflow Flag

1 = Flag set when free-running counter transition from $FFFF to $0000 occurs
0 = Flag cleared when TSR and counter low register ($11) are accessed

Bits 0-4 - Not used

Always read zero

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

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

1. The timer status register is read or written when TOF is set, and
2. The LSB of the free-running counter is read but not for the purpose of servicing the flag.

The counter alternate register at address $12 and $13 contains the same value as the free-running counter (at address $10 and $11); therefore, this alternate register can be read at any time without affecting the timer overflow flag in the timer status register.

11.7 TIMER DURING WAIT MODE

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

11.8 TIMER DURING STOP MODE

In the STOP mode, the timer stops counting and holds the last count value if STOP is exited by an interrupt. If RESET is used, the counter is forced to $FFFC. During STOP, if at least one valid input capture edge occurs at the TCAP pin, the input capture detect circuit is armed. This does not set any timer flags nor wake up the MCU, but when the MCU does wake up, there is an active input capture flag and data from the first valid edge that occurred during the STOP mode. If RESET is used to exit STOP mode, then no input capture flag or data remains, even if a valid input capture edge occurred.

SECTION 12 SERIAL COMMUNICATIONS INTERFACE

12.1 OVERVIEW AND FEATURES

The SCI on the MC68HC05C0 is a full duplex UART type asynchronous system. The SCI uses standard non-return-to-zero (NRZ) format (one start bit, eight or nine data bits, and one stop bit). An on-chip baud rate generator derives standard baud rate frequencies from the MCU oscillator. Both the transmitter and the receiver are double buffered so back-to-back characters can be handled easily, even if the CPU is delayed in responding to the completion of an individual character.The SCI transmitter and receiver are functionally independent but use the same data format and baud rate prescalers.

12.1.1 SCI TWO-WIRE SYSTEM FEATURES:

- Standard NRZ (mark/space) format
- Advanced error detection method includes noise detection for noise duration of up to one-sixteenth bit time
- Full-duplex operation (Simultaneous Transmit and Receive)
- Software programmable for one of 32 different baud rates
- Different baud rates possible for transmit and receive
- Software selectable word length (eight or nine bit words)
- Separate transmitter and receiver enable bits
- Interrupt or polled operation
- Four separate enable bits available for interrupt control

12.1.2 SCI RECEIVER FEATURES:

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

12.1.3 SCI TRANSMITTER FEATURES:

- Transmit data register empty flag
- Transmit complete flag
- Send break
- Transmitter clock for synchronous transmission; phase and polarity are software programmable

A system must include both RDI (receive data in) and TDO (transmit data out) signals.

12.2 DATA FORMAT

Receive data in (RDI) is the serial data that is transferred to the internal data bus from the receive data input pin (RDI). Transmit data out (TDO) is the serial data that is transferred from the internal bus to the transmit data output pin (TDO). The non-return-to-zero (NRZ) data format shown in **Figure 12-1**is used and must meet the following criteria:

- A high level indicates a logic one and a low level indicates a logic zero.
- The idle line is brought to a logic one state prior to transmission/reception of a character.
- A frame consists of a start bit, a character of eight or nine data bits, and a stop bit.
- A start bit (logic zero) is used to indicate the start of a frame.
- A stop bit (logic one) is used to indicate the end of a frame.
- The data is transmitted and received least significant bit first.
- A break is defined as the transmission or reception of a low (logic zero) for at least one complete frame time.
- A preamble is defined as the transmission or reception of a high (logic one) for one complete frame time after software enables the transmitter.

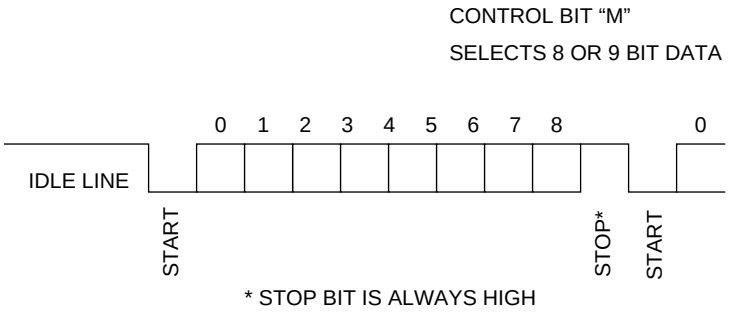


Figure 12-1: Data Format

12.3 FUNCTIONAL DESCRIPTION

A block diagram of the SCI is shown in **Figure 12-2**. The user has option bits in serial control register 1 (SCCR1) to select the “wake-up” method (WAKE bit) and data word length (M bit) of the SCI. Serial communications control register 2 (SCCR2) provides control bits which individually enable/disable the transmitter or receiver (TE and RE, respectively), enable system interrupts (TIE, TCIE, RIE, ILIE), provide the wake-up enable bit (RWU) and the send break code bit (SBK). Control bits in the baud rate register (SCP1:0, SCT2:0, SCR2:0) allow the user to select different baud rates for the transmitter and receiver.

Data transmission is initiated by a write to the serial communications data register (SCDAT). Provided the transmitter is enabled, data stored in the SCDAT is transferred to the transmit data shift register. This transfer of data sets the transmit data register empty flag (TDRE) in the SCI status register (SCSR) and may generate an interrupt if the transmit interrupt is enabled. The transfer of data to the transmit data shift register is synchronized with the bit rate clock. All data is transmitted least significant bit first. Upon completion of data transmission, the transmission complete flag (TC) in the SCSR is set (provided no pending data, preamble or break is to be sent), and an interrupt may be generated if the transmit complete interrupt is enabled. The TC bit will also be set if the transmitter is disabled, and the data, preamble or break (in the transmit data shift register) has been sent. This will also generate an interrupt if the transmission complete interrupt enable bit (TCIE) is set. If the transmitter is disabled in the middle of a transmission, that character will be completed before the transmitter gives up control of the TDO pin.

The data transfer from the input serial shift register to the SCDAT is synchronized by the receiver bit rate clock. The receive data register full flag bit (RDRF) in the SCSR is set to indicate that a data byte has been transferred from the input serial shift register to the SCDAT, which can cause an interrupt if the receiver interrupt is enabled. The OR (overrun), NF (noise), or FE (framing) error flags in the SCSR may be set if data reception errors occurred. When SCDAT is read, it contains the last data byte received, provided that the receiver is enabled.

An idle line interrupt is generated if the idle line interrupt is enabled and the IDLE bit (which detects idle line transmission) in SCSR is set. This allows a receiver that is not in the wake-up mode to detect the end of a message or the preamble of a new message, or to resynchronize with the transmitter. A valid character must be received before the idle line condition or the IDLE bit will not be set and idle line interrupt will not be generated.

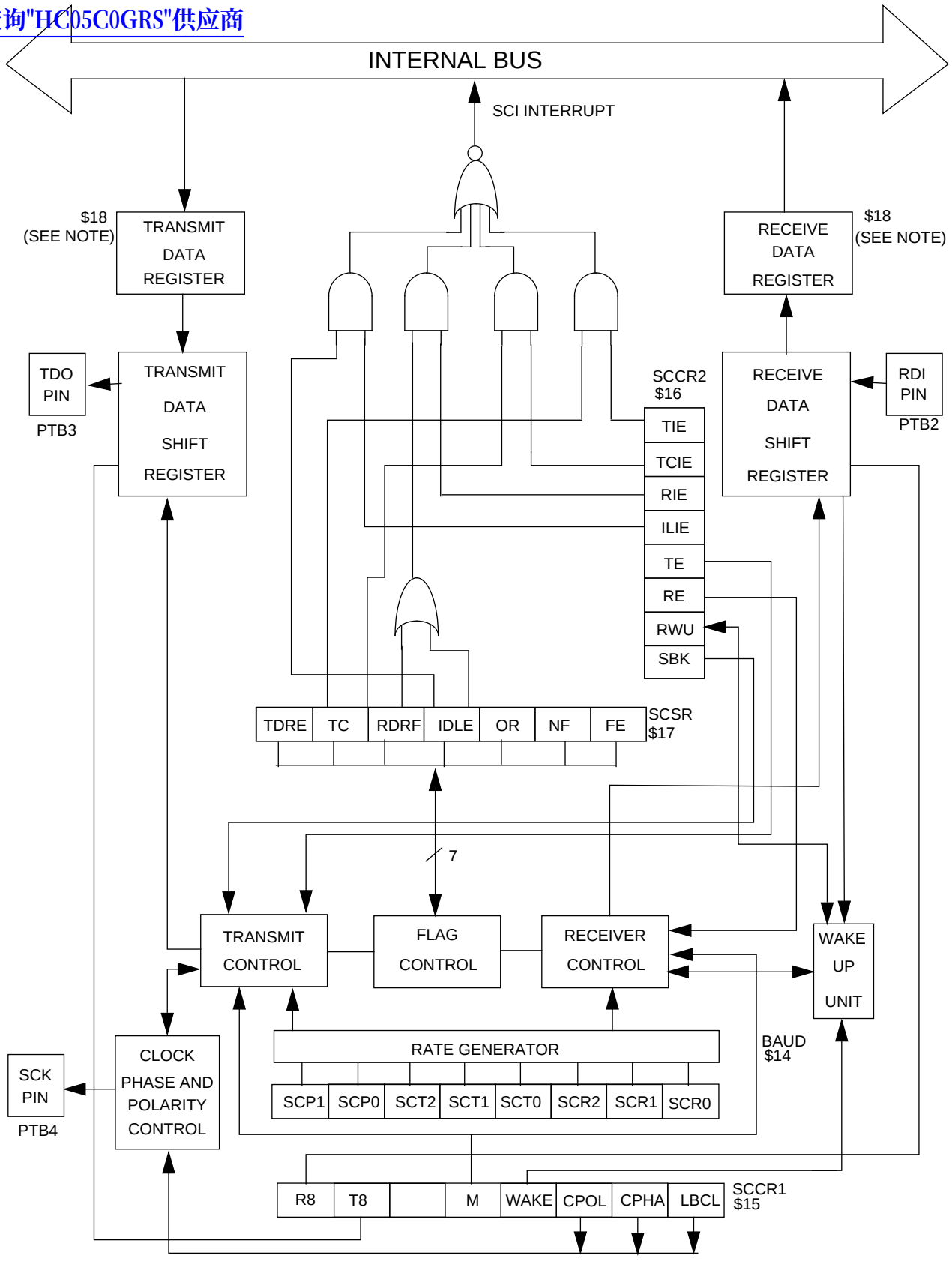


Figure 12-2: SCI Block Diagram

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

12.4 RECEIVER

This section describes the receiver wake-up features, the receiver functionality including the sampling of the bits, and the condition of a start-bit following a framing error.

12.4.1 SCI RECEIVER WAKE-UP FEATURE

In a typical multiprocessor configuration, the software protocol will usually identify the address(es) at the beginning of the message. In order to permit uninterested MPU's to ignore the remainder of the message, a wake-up feature is included whereby all further SCI receiver flag (and interrupt) processing can be inhibited until certain conditions are detected on the line. To enable the wake-up feature the RWU bit in SCCR1 must be set. See Section 12.6.2 SERIAL COMMUNICATIONS CONTROL REGISTER 1 (SCCR1).

12.4.1.1 SCI RECEIVER IDLE WAKE-UP

The receiver idle wake-up feature is included whereby all further SCI receiver flag (and interrupt) processing can be inhibited until its data line returns to the idle state.

An SCI receiver is re-enabled by an idle string of 10 (M=0) or 11 (M=1) consecutive ones. Software for the transmitter must provide the required idle string between consecutive messages and prevent it from occurring within messages.

The wake-up method is selected by the WAKE bit. See Section 12.6.2 SERIAL COMMUNICATIONS CONTROL REGISTER 1 (SCCR1).

12.4.1.2 SCI RECEIVER ADDRESS MARK WAKE-UP

A second wake-up method is provided in lieu of the idle string discussed in the previous section. This method allows the user to insert a logic one in the most significant bit of the transmit data word (address byte) which will wake up sleeping receivers. In software, receivers can then compare the received address to determine if they need to listen. If not they can be put back to sleep and ignore incoming data until the next address mark.

12.4.2 RECEIVE DATA IN

Receive data in (RDI) is the serial data which is transferred from the input pin via the SCI to the receive data register (RDR). The RDI pin contains an internal Schmitt trigger to improve noise immunity. While waiting for a start bit, the receiver samples the input at a rate 16 times higher than the set baud rate. This increased period is referred to as the RT period. When the input (idle) line is detected low, it is tested for three more sample times at intervals of 2RT. See **Figure 12-3 : SCI Examples of Start Bit Recognition Technique** (all data samples are at RT clock edges). If at least two of these three samples detect a logic low, a valid start bit is assumed to be detected. If in two or more samples, a logic high is detected, the start bit will be rejected (the line is assumed to be idle), and the next zero detected will be checked the same way.

The receive clock generator is controlled by the baud rate register (See Section 12.6.5 BAUD RATE REGISTER); however, the SCI is synchronized by the start bit independent of the transmitter.

Once a valid start bit is detected, the start bit, each data bit, and the stop bit are each sampled three more times at intervals of 8RT, 9RT and 10RT (1RT is the position where the bit is expected to start). See **Figure 12-4 : SCI Sampling Technique used on all Bits**. The value of the bit is determined by voting logic which takes the value of a majority of samples. A noise flag is set if all three samples on a valid start bit, data bit, or stop bit do not agree. A noise flag is also set if the start verification samples do not agree.

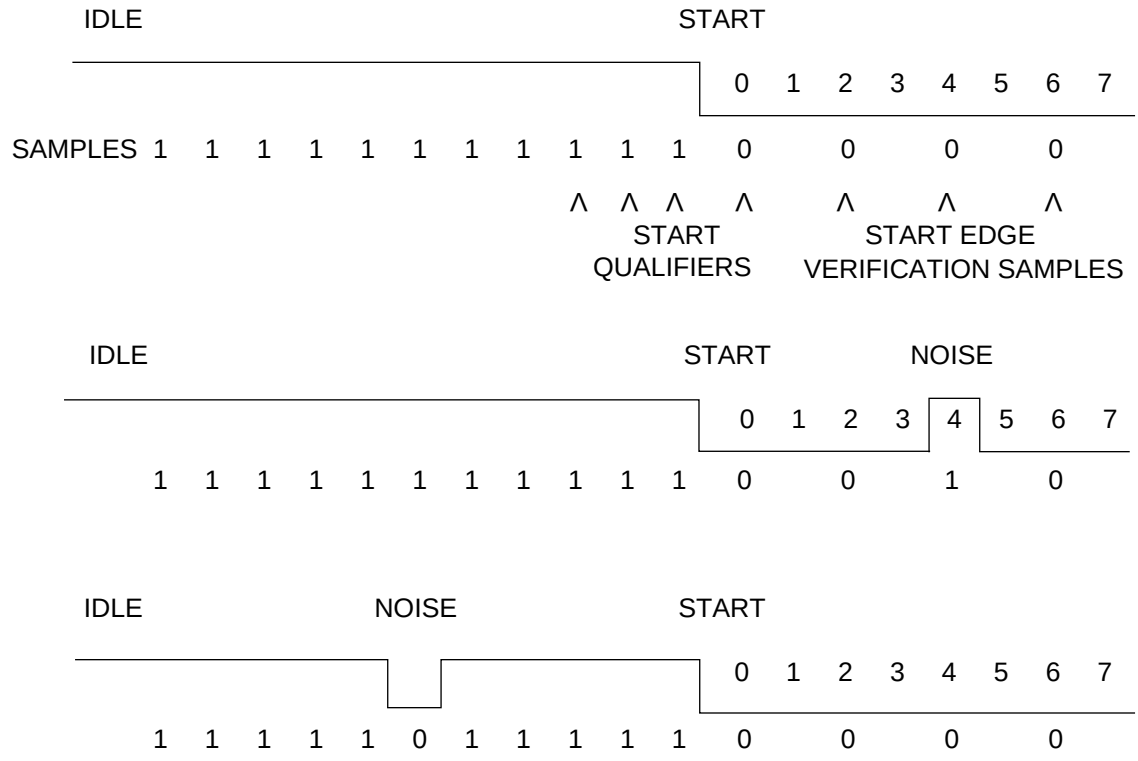


Figure 12-3: SCI Examples of Start Bit Recognition Technique

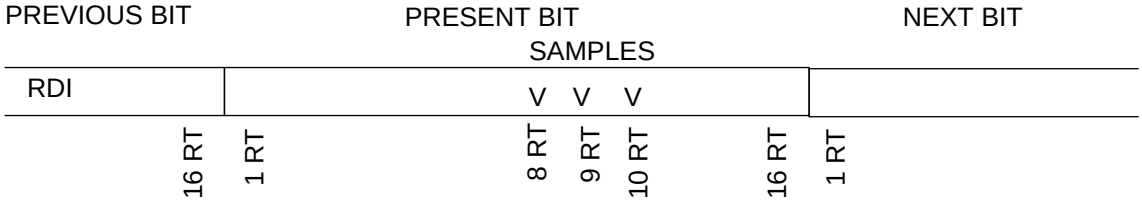


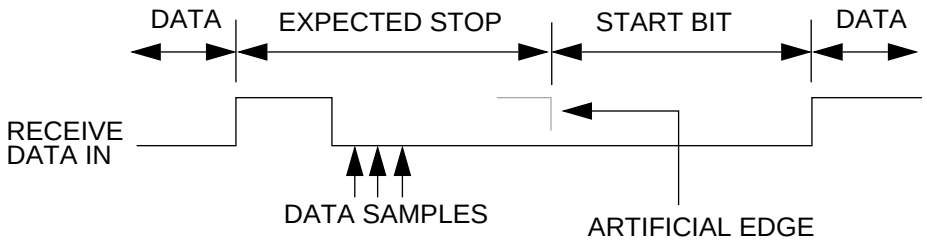
Figure 12-4: SCI Sampling Technique used on all Bits

12.4.3 START BIT DETECTION FOLLOWING A FRAMING ERROR

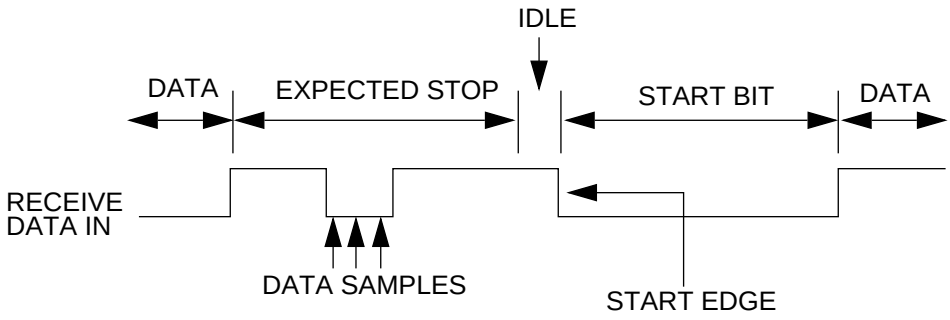
A framing error occurs if the bit stream word boundaries are not synchronized with the receiver bit counter. If there has been a framing error without detection of a break (10 zeros for 8-bit format or 11 zeros for 9-bit format), the circuit continues to operate as if there actually was a stop bit. In this case the start edge may be placed artificially. See **Figure 12-5 : SCI Artificial Start Following a Framing Error** example (a).

The last bit received in the data shift register (which is to be interpreted as a stop bit) will be inverted to a logic one, and the three logic one start qualifiers (shown in **Figure 12-3**) are forced into the sample shift register during the interval when detection of a start bit is anticipated. This creates a falling edge at the appropriate time for a start bit, which will re-synchronize the data and receiver. Example (b) does not require an artificial start because the line returns high, therefore the start bit is created without intervention.

If the receiver detects that a break produced the framing error (RDRF=1, FE=1, receiver data register = $00), the start bit will not be artificially induced and the receiver must actually detect a logic one before the start bit can be recognized (See **Figure 12-6 : SCI Start Bit Following a Break**).



(a) LINE IS LOW AT END OF STOP BIT - SO START EDGE IS ARTIFICIALLY PLACED



(b) LINE IS HIGH AT END OF STOP BIT - WAIT FOR REAL START EDGE

Figure 12-5: SCI Artificial Start Following a Framing Error

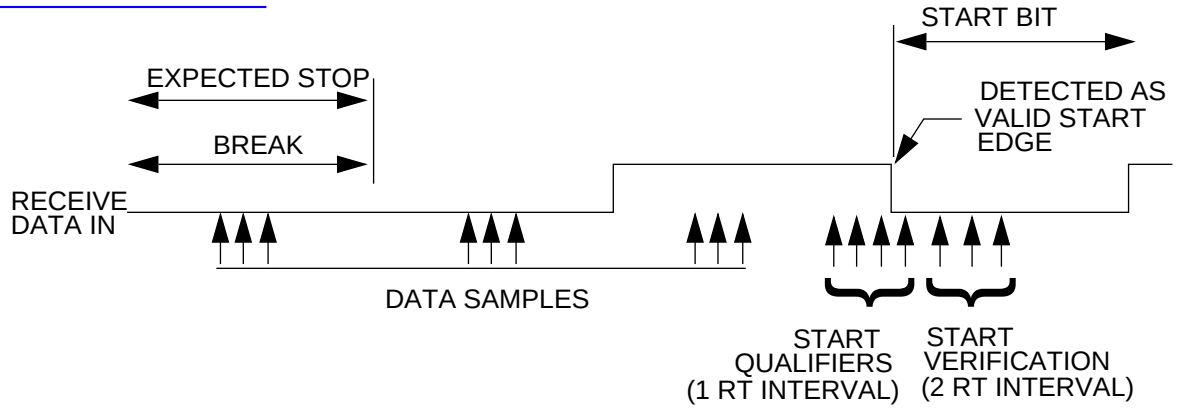


Figure 12-6: SCI Start Bit Following a Break

12.5 TRANSMITTER

This section describes the transmit data out and the synchronous transmit capability.

12.5.1 TRANSMIT DATA OUT

Transmit data out (TDO) is the serial data which is transferred from the transmit data register (TDR) via the SCI to the output pin. The data format is shown in **Figure 12-1**.The transmitter generates a bit time by using a derivative of the RT clock, thus producing a transmission rate equal to one-sixteenth that of the receiver sample clock (assuming the same baud rate is selected for both receiver and transmitter).

12.5.2 SCI SYNCHRONOUS TRANSMISSION

The SCI transmitter allows a one-way synchronous transmission. The SCK pin is the clock output of the SCI transmitter. The SCK pin is inactive when the transmitter is sending start and stop bits. In addition, the SCK pin is inactive during idle, preamble, and send break conditions.

The LBCL bit in SCCR1 controls whether or not the SCK clock is active during the last valid data bit (address mark). The CPOL bit in SCCR1 selects the SCK clock polarity, and the CPHA bit in SCCR1 selects the phase. See **Figure 12-9 : SCI Data Clock Timing Diagram (M=0)** and **Figure 12-10 : SCI Data Clock Timing Diagram (M=1)** for more information.

These options allow the SCI to control serial peripherals which consist of shift registers, without losing any functionality of the SCI transmitter. These options do not affect the SCI receiver, which is independent of the transmitter.

NOTE: The LBCL, CPOL, and CPHA bits must be selected before the transmitter is enabled to ensure that the clocks function correctly. In addition, these bits should only be changed with the transmitter disabled.

12.6 REGISTERS

There are five different registers used in the SCI: SCDAT, SCCR1, SCCR2, SCSR and BAUD. The internal configuration of these registers is discussed in the following paragraphs.

12.6.1 SERIAL COMMUNICATIONS DATA REGISTER (SCDAT)

		7	6	5	4	3	2	1	0
SCDR $18	RD	DA7	DA6	DA5	DA4	DA2	DA2	DA1	DA0
	WR								
	RST	U	U	U	U	U	U	U	U

Figure 12-7: Serial Communications Data Register

The Serial Communications Data Register (SCDAT) performs two functions in the serial communications interface. It acts as the read-only receive data register when it is read and as the write-only transmit data register when it is written. **Figure 12-2**shows this register as two separate registers, namely: the receiver data register (RDR) and the transmit data register (TDR).

The TDR provides the parallel interface from the internal data bus to the transmit shift register. The RDR provides the interface from the receive shift register to the internal data bus.

12.6.2 SERIAL COMMUNICATIONS CONTROL REGISTER 1 (SCCR1)

The SCI Control Register 1 (SCCR1) contains control bits to determine word length, select wake-up method, and control the options to output the transmitter clock for synchronous transmissions.

		7	6	5	4	3	2	1	0
SCCR1 $15	RD	R8	T8	SCKM	M	WAKE	CPOL	CPHA	LBCL
	WR								
	RST	0	0	0	0	0	0	0	0

Figure 12-8: Serial Communications Control Register

R8 - Receive Data Bit 8

This bit is the ninth serial data bit received when the SCI system is configured for nine data bit operation (M = 1). The most significant bit (bit 8) of the received character is transferred into this bit at the same time as the remaining eight bits (bits 0 – 7) are transferred from the serial receive shifter to the SCI receive data register. A write to this bit has no effect. In addition, If M=0 a read from this bit is undefined.

T8 - Transmit Data Bit 8

This bit is the ninth data bit to be transmitted when the SCI system is configured for nine data bit operation (M = 1). When the eight low order bits (bits 0 – 7) of a transmit character are transferred from the SCI data register to the serial transmit shift register, this bit (bit 8) is transferred to the ninth bit position of the shifter. This bit is read/writable anytime, but only has an effect when M=1.

SCKM - Serial Clock Mode

When set, this bit allows Port B bit 4 to become the SCK pin, provided the transmitter is enabled (TE is set in SCCR2) or a transmission is in progress when TE is written to a zero. If this bit is clear, or if this bit is set but the transmitter is disabled and no transmission is in progress, control of this port will be returned to the Port B bit 4 data direction register.

1 = Port B bit 4 = SCK (if TE or transmission in progress)
0 = Port B bit 4 = General Purpose I/O

M - Mode (SCI character word length)

The M bit controls the character length for both the transmitter and receiver at the same time. The 9th data bit is most commonly used as an extra stop bit or in conjunction with the “address mark” wake-up method. It can also be used as a parity bit.

1= one start bit, nine data bits, one stop bit
0= one start bit, eight data bits, one stop bit

NOTE: The Mode bit should not be changed in the middle of a transmission or reception.

WAKE - Wake-Up Select

This bit allows the user to select the method for receiver “wake-up”. If the WAKE bit is clear, an idle line condition will “wake up” the receiver. If the WAKE bit is set, the system acknowledges an address bit (most significant bit).

The address bit “wake-up” is dependent on both the WAKE bit and the M bit as shown in **Table 12-1**.

NOTE: The receiver will not acknowledge any wake-up feature unless the RWU bit in control register 2 (SCCR2) is set.

Table 12-1: Method of Receiver Wake-Up

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

CPOL - Clock Polarity

This bit allows the user to select the polarity of the clock to the SCK pin. This bit works in conjunction with the CPHA bit to produce the desired clock-data relation.

1= Steady state high outside the transmission window
0= Steady state low outside the transmission window

NOTE: The CPOL bit should not be changed when the transmitter is enabled.

CPHA - Clock Phase

This bit allows the user to select the phase of the clock to be sent to the SCK pin. This bit works in conjunction with the CPOL bit to produce the desired clock-data relation.

1= SCK line activated at the beginning of the data bit
0 = SCK line activated in the middle of the data bit

NOTE: The CPHA bit should not be changed when the transmitter is enabled.

LBCL - Last Bit Clock

This bit allows the user to select whether or not the clock associated with the last data bit transmitted (MSB) is output to the SCK pin.

1= Clock is active on last data bit
0= Clock is inactive on last data bit

The last data bit is the eighth or ninth bit, depending on the state of the M bit. Refer to **Table 12-2** to determine the number of clock pulses transmitted.

NOTE: The LBCL bit should not be changed when the transmitter is enabled.

Table 12-2: SCI Clock on SCK Pin

M Bit	Data Format	LBCL Bit	Number of Clocks on SCK pin
0	8 bit	0	7
0	8 bit	1	8
1	9 bit	0	8
1	9 bit	1	9

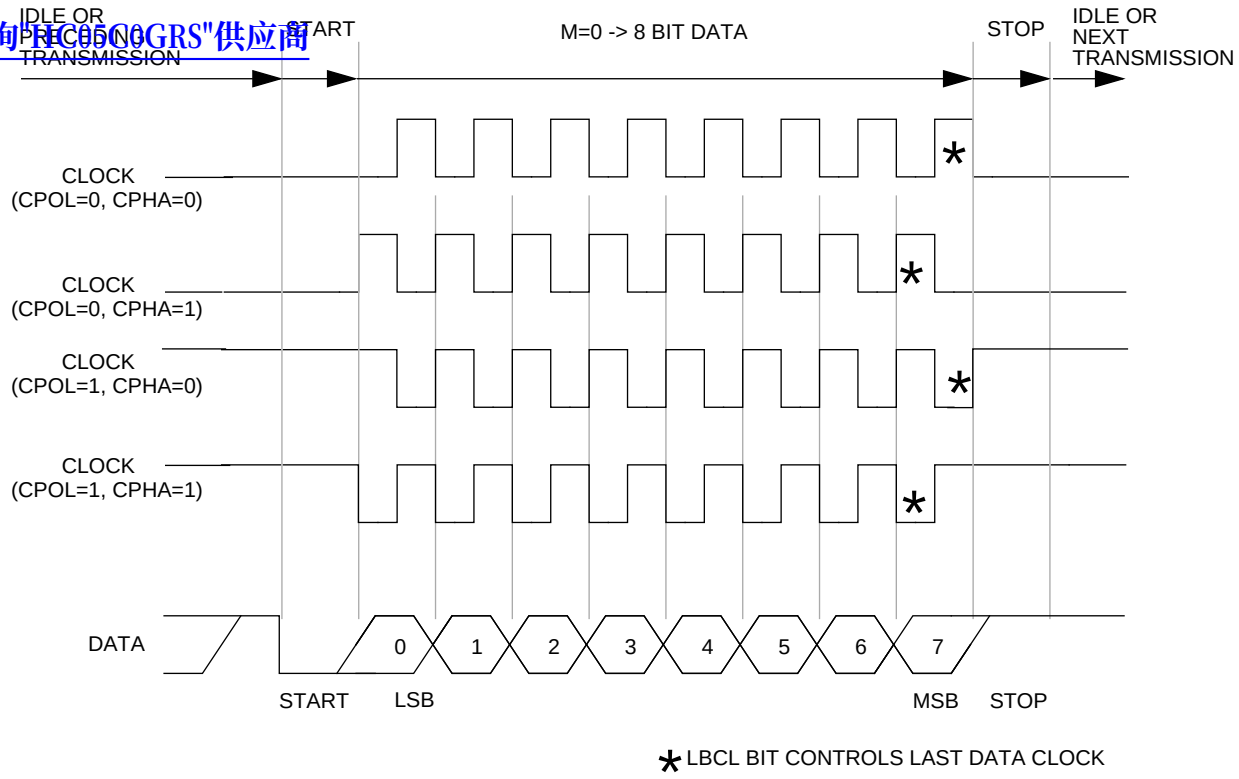


Figure 12-9: SCI Data Clock Timing Diagram (M=0)

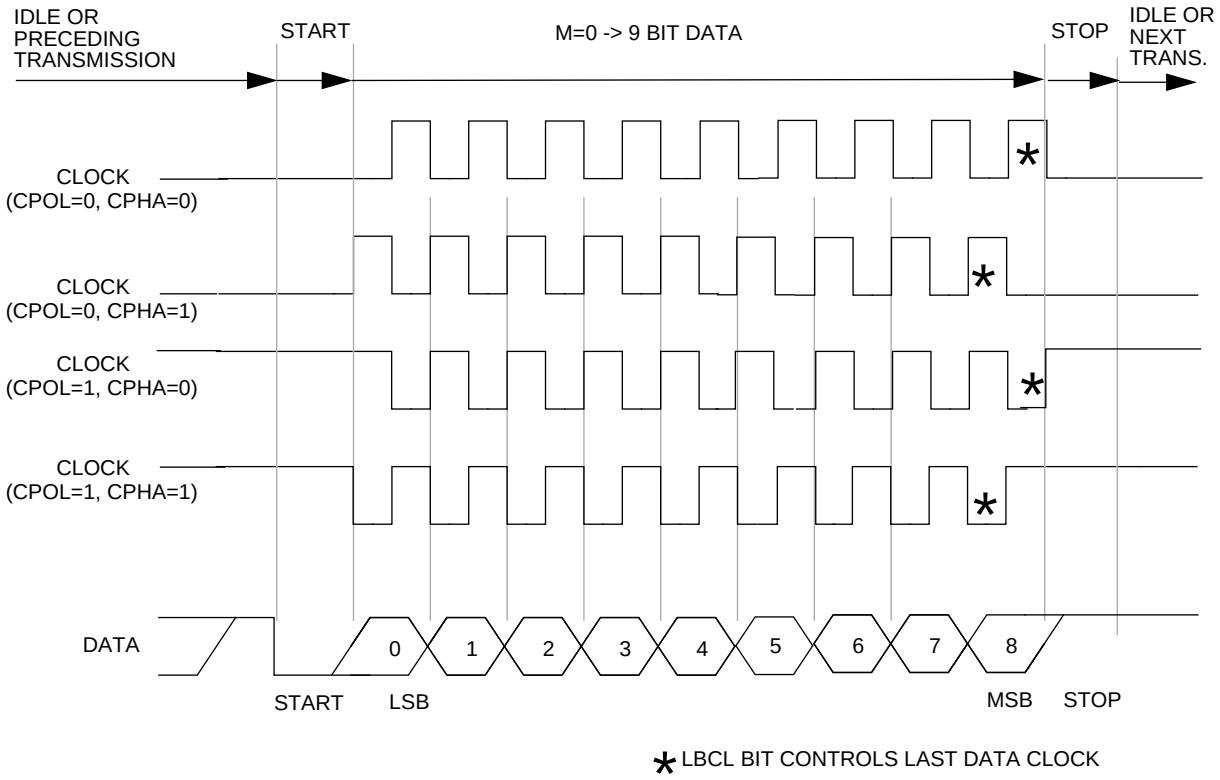


Figure 12-10: SCI Data Clock Timing Diagram (M=1)

12.6.3 SERIAL COMMUNICATIONS CONTROL REGISTER 2 (SCCR2)

The SCI Control Register 2 (SCCR2) provides control bits which individually enable/disable the transmitter or receiver, enable the system interrupts, provide the wake-up enable bit, and provide a "send break code" bit. All bits are read/writable. The flags are described in **Section 12.6.4, SERIAL COMMUNICATIONS STATUS REGISTER (SCSR)**.

		7	6	5	4	3	2	1	0
SCCR2 $16	RD / WR	TIE	TCIE	RIE	ILIE	TE	RE	RWU	SBK
	RST	0	0	0	0	0	0	0	0

Figure 12-11: Serial Communications Control Register 2

TIE- Transmit Interrupt Enable

1= TDRE interrupt enabled
0= TDRE interrupt disabled

TCIE - Transmit Complete Interrupt Enable

1= TC interrupt enabled
0 = TC interrupt disabled

RIE - Receiver Interrupt Enable

1= RDRF or OR (with RDRF) interrupt is enabled
0= RDRF and OR (with RDRF) interrupts disabled

ILIE - Idle Line Interrupt Enable

1= Idle interrupt disabled
0= Idle interrupt disabled

TE- Transmit Enable

When the transmit enable bit is set, the transmit shift register output is applied to the TDO line and the corresponding clocks are applied to the SCK pin. Depending on the state of control bit M (SCCR1), a preamble of 10 (M = 0) or 11 (M = 1) consecutive ones is transmitted when software sets the TE bit from a cleared state.

If a transmission is in progress, and TE is written to a zero, then the transmitter will wait until after the present byte has been transmitted before releasing the TDO and SCK (if enabled) pins to their Data Direction Register control.

If the TE bit has been cleared and then set before the current byte has been transmitted, the transmitter will wait until that byte is transmitted and will then initiate transmission of a new preamble. After the preamble, the line will then be idle (high) or begin transmitting the next byte if new data has been loaded.

After loading the last byte in the serial communications data register and receiving the TDRE flag, the user can clear TE. Transmission of the last byte will then be completed before the transmitter gives up control of the TDO and SCK (if enabled) pins. While the transmitter is active, the data direction register control for Port B bit 3 is overridden and the line is forced to be the TDO pin. Port B bit 4 is also overridden and forced to be the SCK pin if SCKM is set in SCCR1.

1= Transmitter enabled
0= Transmitter disabled

RE- Receiver Enable

When the receiver enable bit is set, the receiver is enabled. When RE is clear, the receiver is disabled and all of the status bits associated with the receiver (RDRF, IDLE, OR, NF and FE) are inhibited. While the receiver is enabled, the data direction register control for Port B bit 2 is overridden and the line is forced to be the RDI pin.

1= Receiver enabled
0= Receiver disabled

RWU - Receiver Wake-Up

When the receiver wake-up bit is set by the user software, it puts the receiver to sleep and enables the wake-up function. If the WAKE bit in SCCR1 is cleared, RWU is automatically cleared by the SCI logic after receiving 10 (M = 0) or 11 (M = 1) consecutive ones. If the WAKE bit in SCCR1 is set, RWU is automatically cleared by the SCI logic after receiving a data word whose MSB is set. See **Table 12-1**.

1= Receiver wake-up enabled
0= Receiver wake-up disabled

SBK - Send Break

If the send break bit is toggled set and cleared, the transmitter sends 10 (M = 0) or 11 (M = 1) zeros and then reverts to idle. If SBK remains set, the transmitter will continually send whole blocks of zeros (sets of 10 or 11) until cleared. At the completion of the break code, the transmitter sends at least logic one to guarantee recognition of a valid start bit. If the transmitter is currently empty and idle, setting and clearing SBK is likely to queue two character times of break because the first break transfers almost immediately to the shift register and the second is then queued into the parallel transmit buffer.

If during a data byte transmission, the TE and SBK bits are toggled, or if SBK is toggled followed by TE being toggled, a break character will not be sent. Only a preamble will be

transmitted, followed by the next character if ready. If however, the TE is toggled first, followed by the SBK being toggled, the break character will be transmitted. In that condition, a preamble character, followed by a break character, followed by the next character if ready, will be sent.

1= Send break code enabled
0= Send break code disabled

12.6.4 SERIAL COMMUNICATIONS STATUS REGISTER (SCSR)

The SCI Status Register (SCSR) provides inputs to the interrupt logic circuits for generation of the SCI system interrupt. In addition, a noise flag bit and framing error bit are contained here.

SCSR $17

	7	6	5	4	3	2	1	0
RD	TDRE	TC	RDRF	IDLE	OR	NF	FE	
WR								
RST	1	1	0	0	0	0	0	—

Figure 12-12: Serial Communications Status Register

TDRE - Transmit Data Register Empty Flag

This bit is set when the byte in the transmit data register is transferred to the transmit serial shift register. TDRE clear indicates that the transfer has not yet occurred and a write to the transmit data register (SCDAT) will overwrite the previous value.

TDRE can only be cleared by first accessing the SCSR with TDRE set, followed by writing to the transmit data register (SCDAT).

New data will not be transmitted unless the SCSR register is read before writing to the transmit data register (SCDAT) to clear TDRE.

TC - Transmit Complete Flag

This bit is set at the end of a data frame preamble or break condition if one of the following conditions is true:

1. TE=1, TDRE=1, and no pending data, preamble, or break is to be transmitted
2. TE=0, and the data, preamble, or break (in the transmit shift register) has been transmitted

The TC bit can only be cleared by first accessing the SCSR with TC set, followed by writing to the transmit data register (SCDAT). The TC bit, unlike the TDRE bit, does not inhibit the transmitter function in any way.

RDRF - Receive Data Register Full Flag

This bit is set when the contents of the receiver serial shift register are transferred to the receiver data register (SCDAT). The RDRF bit can only be cleared by first accessing the SCSR with RDRF set, followed by a read of the receiver data register (SCDAT).

If multiple errors are detected in any one received word, the NF, FE, and RDRF bits will be affected as appropriate during the same clock cycle.

IDLE - Idle Line Detected Flag

This bit is set when a receiver idle line is detected (the receipt of a minimum of ten/eleven consecutive 1's). The minimum number of consecutive 1's needed is 10 (M=0) or 11 (M=1). This allows a receiver that is not in wake-up mode to detect the end of a message, detect the preamble of a new message, or to re-synchronize with the transmitter.

The IDLE bit can only be cleared by first accessing the SCSR with IDLE set, followed by a read of the receiver data register (SCDAT).

Once cleared, IDLE will not be set again until after RDRF has been set (until after the line has been active and becomes idle again). Also, the IDLE bit will not be set by an idle line when the receiver wakes up from idle wake-up mode (RWU=1 and WAKE=0).

When the receiver is first enabled, any initial idle or preamble received will not set the IDLE flag. The IDLE will not be set until after RDRF has been set.

OR - Overrun Error Flag

This bit is set when a new byte is ready to be transferred from the receiver shift register to the receiver data register (SCDAT) and the receive data register is already full (RDRF bit is set). Data transfer is then inhibited until this bit is cleared. Data in the receive data register is valid in this case, but additional data received during an overrun condition (including the byte causing the overrun) will be lost.

The OR bit can only be cleared by first accessing the SCSR with OR set, followed by a read of the receiver data register (SCDAT).

NF - Noise Error Flag

This bit is set if there is noise on a "valid" start bit, any of the data bits, or on the stop bit. It is not set by noise on the idle line nor by invalid start bits. If there is noise, the NF bit is not set until the RDRF bit is set.

The NF bit represents the status of the byte in the receiver data register (SCDAT). For the byte being received (shifted in) there will also be a "working" noise flag which will be transferred to the NF bit when the serial data is loaded into the receive data register. The

NF bit does not generate an interrupt because the RDRF bit gets set with NF and RDRF can be used to generate the interrupt.

The NF bit can only be cleared by first accessing the SCSR with NF set, followed by a read of the receive data register (SCDAT).

FE - Framing Error Flag

This bit is set when the word boundaries in the bit stream are not synchronized with the receiver bit counter (generated by the reception of a logic zero bit where a stop bit was expected). The received byte is still transferred to the received data register. The FE bit is not set until the RDRF bit is set.

The FE bit does not generate an interrupt because the RDRF bit gets set with FE and RDRF can be used to generate the interrupt.

If the received byte causes both a framing error and overrun error, the processor will only recognize the overrun error. The framing error flag inhibits further transfer of data into the receive data register (SCDAT) until it is cleared.

The FE bit can only be cleared by first accessing the SCSR with FE set, followed by a read of the receive data register (SCDAT).

12.6.5 BAUD RATE REGISTER

The Baud Rate Register selects the SCI transmitter and receiver baud rates. All bits are read/writable.

NOTE: The Baud Rate Register should not be manipulated while the transmitter or the receiver is enabled.

SCBR $14

	7	6	5	4	3	2	1	0
RD / WR	SCP1	SCP0	SCT2	SCT1	SCT0	SCR2	SCR1	SCR0
RST	0	0	1	1	1	1	1	1

Figure 12-13: Serial Communications Baud Rate Register

SCP1:0 - SCI Prescaler Bits

The SCP1 and SCP0 bits in the baud rate register are used to select a common prescaler to increase the range of standard baud rates controlled by the SCR2:0 bits for the receiver and the SCT2:0 bits for the transmitter. Reset clears these bits thereby causing the default prescaler to be a divide by one. Refer to **Table 12-3** for a list of highest transmission baud rates with various crystal frequencies.

SCT2:0 - SCI Transmit Baud Rate Selection Bits

The SCT2, SCT1 and SCT0 bits in the baud rate register are used to select the baud rate of the SCI transmitter. They further divide down the clock from the prescaler stage. See **Figure 12-14 : SCI Rate Generator Block Diagram**. These bits are undefined after reset. Refer to **Table 12-4** for example baud rates from selected prescaler frequency outputs.

SCR2:0 - SCI Receiver Baud Rate Selection Bits

The SCR2, SCR1 and SCR0 bits in the baud rate register are used to select the baud rate of the SCI receiver. They further divide down the clock from the prescaler stage. See **Figure 12-14 : SCI Rate Generator Block Diagram**. These bits are undefined after reset. Refer to **Table 12-4** for example baud rates from selected prescaler frequency outputs. Note that the final frequencies shown in **Table 12-4** are baud rates. The receiver clock will be 16 times higher to accommodate bit sampling.

Table 12-3: Prescaler Highest Baud Rate Frequency Output

SCP Bit		Clock* Divided By	Crystal Frequency MHz				
1	0		16.0	8.0	4.9152	4.0	3.6864
0	0	1	250.0 kHz	125.0 kHz	76.80 kHz	62.50 kHz	57.60 kHz
0	1	3	83.333 kHz	41.666 kHz	25.60 kHz	20.833 kHz	19.20 kHz
1	0	4	62.50 kHz	31.25 kHz	19.20 kHz	15.625 kHz	14.40 kHz
1	1	13	19.20 kHz	9600 Hz	5.907 kHz	4800 Hz	4430 Hz

* Refers to internal processor clock

NOTE: The divided frequencies shown in **Table 12-3** represent the highest baud rates that can be obtained by a specific crystal frequency and using only prescaler division. Lower baud rates may be obtained by providing a further division using the SCI rate select bits as shown below for some representative prescaler outputs.

Table 12-4: Transmit Baud Rate Output for a Given Prescaler Output

SCR2:0/SCT2:0 Bits			Divided By	Representative Highest Prescaler Baud Rate Output			
2	1	0		250.0 kHz	76.80 kHz	19.20 kHz	9600 Hz
0	0	0	1	250.0 kHz	76.80 kHz	19.20 kHz	9600 Hz
0	0	1	2	125.0 kHz	38.40 kHz	9600 Hz	4800 Hz
0	1	0	4	62.50 kHz	19.20 kHz	4800 Hz	2400 Hz
0	1	1	8	31.25 kHz	9600 Hz	2400 Hz	1200 Hz
1	0	0	16	15.62 kHz	4800 Hz	1200 Hz	600 Hz
1	0	1	32	7.81 kHz	2400 Hz	600 Hz	300 Hz
1	1	0	64	3.90 kHz	1200 Hz	300 Hz	150 Hz
1	1	1	128	1.95 kHz	600 Hz	150 Hz	75 Hz

NOTE: **Table 12-4** illustrates how the SCI select bits can be used to provide lower transmitter or receiver baud rates by further dividing the prescaler output frequency. The four examples are only representative samples. In all cases, the baud rates shown correspond to the transmit clock, and the receive clock is 16 times higher in frequency than the actual baud rate.

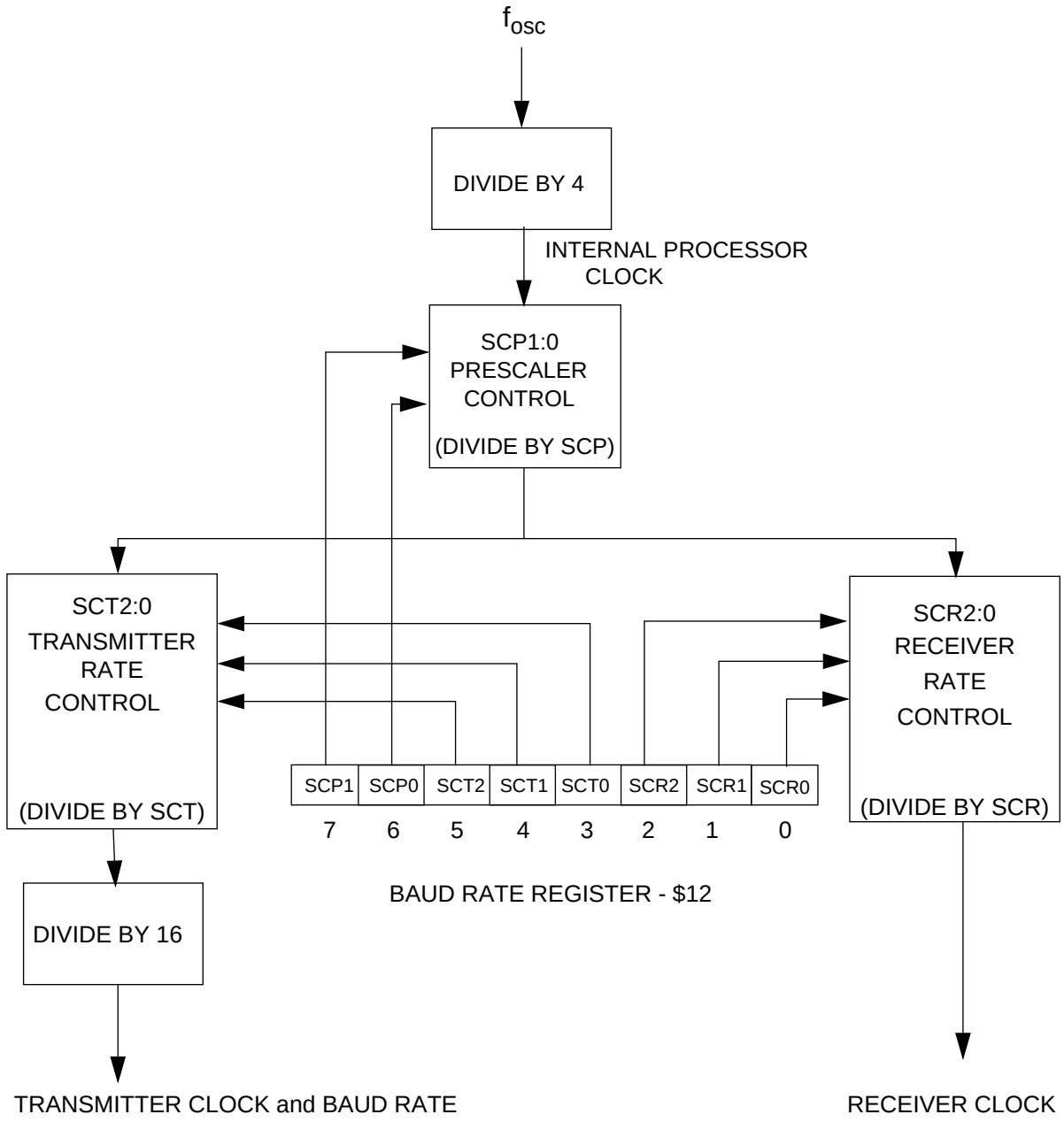


Figure 12-14: SCI Rate Generator Block Diagram

As an example of baud rate generation, suppose a 9600 Hz baud rate is desired from a 4.9152 MHz oscillator crystal. The prescaler bits (SCP1:0) could be set for either a divide-by-one or a divide-by-four. If a divide-by-four prescaler is used, then SCR2:0 and SCT2:0 must be set for divide-by-two. If instead a divide-by-one prescaler is used, then SCR2:0 and SCT2:0 must be set for a divide-by-eight.

12.7 SCI DURING WAIT MODE

The SCI system is not affected by the WAIT mode and continues regular operation. For low power consumption the SCI should be disabled when not in use. Any valid SCI interrupt will cause WAIT mode to be exited.

12.8 SCI DURING STOP MODE

If the receiver is receiving data when the STOP instruction is executed, received data sampling is stopped (baud rate generator stops) and the rest of the data is lost. When STOP mode is entered, the baud rate generator which drives the receiver and transmitter is shut down. This essentially stops all SCI activity. The receiver is unable to receive, and the transmitter is unable to transmit. If the STOP instruction is executed during a transmitter transfer, that transfer is halted.

When STOP mode is exited, that particular transmission resumes (if the exit is the result of an external interrupt).

NOTE: For the above reasons, all SCI transactions should be inactive when the STOP instruction is executed.

SECTION 13 ELECTRICALS

13.1 MAXIMUM RATINGS

(Voltages referenced to VSS)

Rating	Symbol	Value	Unit
Supply Voltage	V_{DD}	-0.3 to +7.0	V
Input Voltage	V_{in}	V_{SS} - 0.3 to V_{DD} + 0.3	V
$\overline{IRQ}$ Pin	Vin	V_{SS} - 0.3 to $2 \times V_{DD}$ + 0.3	V
Current Drain Per Pin Excluding V_{DD} and V_{SS}	I	25	mA
Operating Temperature Range MC68HC05C0	T_A	T_L to T_H 0 to +70	°C
Storage Temperature Range	T_{stg}	-65 to +150	°C

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

13.2 THERMAL CHARACTERISTICS

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

13.3 DC ELECTRICAL CHARACTERISTICS

(V_{DD} = 5.0 V_{dc} ±10%, V_{SS} = 0 V_{dc}, T_A = 0°C to +70°C, unless otherwise noted)

Characteristic[1]	Symbol	Min	Typ	Max	Unit
Output voltage I_{Load} = 10.0 µA I_{Load} = -10.0 µA	 V_{OL} V_{OH}	 — V_{DD}-0.1	 — —	 0.1 —	 V
Output High Voltage (I_{Load} = -0.8 mA) PB0-5, PC0-3, PD0-7,$\overline{RD}$,$\overline{WR}$,AS/$\overline{CS2}$ $\overline{A15}$,A14-A8,AD7-0,$\overline{LIR}$/MODE (I_{Load} = -5.0 mA) PB0	 V_{OH} V_{OH}	 V_{DD}-0.8 V_{DD}-0.8	 — —	 — —	 V V
Output Low Voltage (I_{Load} = 1.6 mA) PB0-5, PC0-3, PD0-7,$\overline{RD}$,$\overline{WR}$,AS/$\overline{CS2}$ $\overline{A15}$,A14-A8,AD7-0,$\overline{LIR}$/MODE,$\overline{RESET}$ (I_{Load} = 20 mA) PB0	 V_{OL} V_{OL}	 — —	 — —	 0.4 0.6	 V V
Input High Voltage PB0-5,PC0-3, PD0-7, $\overline{IRQ}$,$\overline{RESET}$,OSC1, AD7-0,$\overline{LIR}$/MODE	V_{IH}	0.7×V_{DD}	—	V_{DD}	V
Input Low Voltage PB0-5,PC0-3, PD0-7, $\overline{IRQ}$,$\overline{RESET}$,OSC1, AD7-0,$\overline{LIR}$/MODE	V_{IL}	V_{SS}	—	0.2×V_{DD}	V
Supply Current (see Notes) Run [3] Wait [2] [4] [6] Stop with LVI disabled/enabled [5] 25°C 0°C to +70°C	 I_{DD} I_{DD} I_{DD} I_{DD}	 — — — —	 TBD TBD TBD/TBD TBD/TBD	 TBD TBD TBD/TBD TBD/TBD	 mA mA µA µA
I/O Ports Hi-Z Leakage Current PB0-5, PC0-3, PD0-7,AD0-7	I_{OZ}	—	—	10	µA
Input Current $\overline{RESET}$, $\overline{IRQ}$, OSC1,PB0-5,PC0-3,PD0-7,AD0-7	I_{in}	—	—	1	µA
Capacitance Ports (as Input or Output) $\overline{RESET}$, $\overline{IRQ}$, OSC1,PB0-5,PC0-3,PD0-7,AD0-7	 C_{out} C_{in}	 — —	 — —	 12 8	 pF pF

NOTES:

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

13.4 CONTROL TIMING

(V_{DD} = 5.0 V_{dc} ±10%, V_{SS} = 0 V_{dc}, T_A = 0°C to +70°C, unless otherwise noted)

Characteristic	Symbol	Min	Max	Unit
Frequency of Operation Crystal Option External Clock Option	 f_{osc} f_{osc}	 — dc	 16.0 16.0	 MHz MHz
Internal Operating Frequency Crystal (f_{osc} ÷ 4) External Clock (f_{osc} ÷ 4)	 f_{op} f_{op}	 — dc	 4.0 4.0	 MHz MHz
Cycle Time	t_{cyc}	250	—	ns
Crystal Oscillator Start-up Time	t_{OXOV}	—	100	ms
Stop Recovery Start-up Time (Crystal Oscillator)	t_{ILCH}	—	100	ms
RESET* Pulse Width	t_{RL}	1.5	—	t_{cyc}
Interrupt Pulse Width Low (Edge-Triggered)	t_{ILIH}	125	—	ns
Interrupt Pulse Period	t_{ILIL}	*	—	t_{cyc}
OSC1 Pulse Width	t_{OH},t_{OL}	60	—	ns

* The minimum period T_{ILIL} should not be less than the number of cycle times it takes to execute the interrupt service routine plus 21 t_{cyc}.

THIS PAGE INTENTIONALLY LEFT BLANK