HC05

MC68HC05BD3
MC68HC705BD3
MC68HC05BD5

TECHNICAL
DATA

GENERAL DESCRIPTION 1

PIN DESCRIPTION AND I/O PORTS 2

MEMORY AND REGISTERS 3

RESETS AND INTERRUPTS 4

MULTI-FUNCTION TIMER 5

PULSE WIDTH MODULATION 6

M-BUS SERIAL INTERFACE 7

SYNC SIGNAL PROCESSOR 8

CPU CORE AND INSTRUCTION SET 9

LOW POWER MODES 10

OPERATING MODES 11

ELECTRICAL SPECIFICATIONS 12

MECHANICAL SPECIFICATIONS 13

MC68HC705BD3 14

MC68HC05BD5 15

1 GENERAL DESCRIPTION

2 PIN DESCRIPTION AND I/O PORTS

3 MEMORY AND REGISTERS

4 RESETS AND INTERRUPTS

5 MULTI-FUNCTION TIMER

6 PULSE WIDTH MODULATION

7 M-BUS SERIAL INTERFACE

8 SYNC SIGNAL PROCESSOR

9 CPU CORE AND INSTRUCTION SET

10 LOW POWER MODES

11 OPERATING MODES

12 ELECTRICAL SPECIFICATIONS

13 MECHANICAL SPECIFICATIONS

14 MC68HC705BD3

15 MC68HC05BD5

MC68HC05BD3
MC68HC705BD3
MC68HC05BD5

High-density complementary metal oxide semiconductor (HCMOS) microcontroller unit

Conventions

Register and bit mnemonics are defined in the paragraphs describing them.

An overbar is used to designate an active-low signal, eg: $\overline{\text{RESET}}$.

Unless otherwise stated, blank cells in a register diagram indicate that the bit is either unused or reserved; shaded cells indicate that the bit is not described in the following paragraphs; ‘u’ is used to indicate an undefined state (on reset).

CUSTOMER FEEDBACK QUESTIONNAIRE (MC68HC05BD3D/H)

Motorola wishes to continue to improve the quality of its documentation. We would welcome your feedback on the publication you have just received. Having used the document, please complete this card (or a photocopy of it, if you prefer).

1. How would you rate the quality of the document? Check one box in each category.

	Excellent			Poor		Excellent			Poor
Organization	❑	❑	❑	❑	Tables	❑	❑	❑	❑
Readability	❑	❑	❑	❑	Table of contents	❑	❑	❑	❑
Understandability	❑	❑	❑	❑	Index	❑	❑	❑	❑
Accuracy	❑	❑	❑	❑	Page size/binding	❑	❑	❑	❑
Illustrations	❑	❑	❑	❑	Overall impression	❑	❑	❑	❑

Comments: _____

2. What is your intended use for this document? If more than one option applies, please rank them (1, 2, 3).

Selection of device for new application ❑
System design ❑
Training purposes ❑
Other ❑ Please specify: _____

3. How well does this manual enable you to perform the task(s) outlined in question 2?

Completely			Not at all
❑	❑	❑	❑

Comments: _____

4. How easy is it to find the information you are looking for?

Easy			Difficult
❑	❑	❑	❑

Comments: _____

5. Is the level of technical detail in the following sections sufficient to allow you to understand how the device functions?

		Too little detail				Too much detail
SECTION 1	GENERAL DESCRIPTION	❑	❑	❑	❑	❑
SECTION 2	PIN DESCRIPTION AND I/O PORTS	❑	❑	❑	❑	❑
SECTION 3	MEMORY AND REGISTERS	❑	❑	❑	❑	❑
SECTION 4	RESETS AND INTERRUPTS	❑	❑	❑	❑	❑
SECTION 5	MULTI-FUNCTION TIMER	❑	❑	❑	❑	❑
SECTION 6	PULSE WIDTH MODULATION	❑	❑	❑	❑	❑
SECTION 7	M-BUS SERIAL INTERFACE	❑	❑	❑	❑	❑
SECTION 8	SYNC SIGNAL PROCESSOR	❑	❑	❑	❑	❑
SECTION 9	CPU CORE AND INSTRUCTION SET	❑	❑	❑	❑	❑
SECTION 10	LOW POWER MODES	❑	❑	❑	❑	❑
SECTION 11	OPERATING MODES	❑	❑	❑	❑	❑
SECTION 12	ELECTRICAL SPECIFICATIONS	❑	❑	❑	❑	❑
SECTION 13	MECHANICAL SPECIFICATIONS	❑	❑	❑	❑	❑
SECTION 14	MC68HC705BD3	❑	❑	❑	❑	❑
SECTION 15	MC68HC05BD5	❑	❑	❑	❑	❑

Comments: _____

6. Have you found any errors? If so, please comment: _____

7. From your point of view, is anything missing from the document? If so, please say what: _____

– Cut along this line to remove –

8. How could we improve this document? _____

_____

_____

9. How would you rate Motorola's documentation?

	Excellent			Poor
– In general	❑	❑	❑	❑
– Against other semiconductor suppliers	❑	❑	❑	❑

10. Which semiconductor manufacturer provides the best technical documentation? _____

11. Which company (in any field) provides the best technical documentation? _____

12. How many years have you worked with microprocessors?

Less than 1 year ❑ 1–3 years ❑ 3–5 years ❑ More than 5 years ❑

– Second fold back along this line –

By air mail
Par avion

FIX STAMP HERE

– First fold back along this line –

– Cut along this line to remove –

Motorola Semiconductors H.K. Ltd.,
13/F, Prosperity Centre,
77-81 Container Port Road,
Kwai Chung, N.T.,
HONG KONG.

F.A.O. HKG CSIC Technical Publications
(re: MC68HC05BD3D/H)

FAX: (852) 2485-0548

MOTOROLA
Semiconductor Products Sector
Asia Pacific Group

– Third fold back along this line –

13. Currently there is some discussion in the semiconductor industry regarding a move towards providing data sheets in electronic form. If you have any opinion on this subject, please comment. _____

_____

14. We would be grateful if you would supply the following information (at your discretion), or attach your card.

Name: _____ Phone No: _____

Position: _____ FAX No: _____

Department: _____

Company: _____

Address: _____

_____

or helping us improve our documentation,
echnical Publications , Motorola Semiconductors H.K. Ltd., Hong Kong.

– Finally, tuck this edge into opposite flap –

TABLE OF CONTENTS

Paragraph Number	TITLE	Page Number
	1 GENERAL DESCRIPTION	
1.1	Features	1-1
	2 PIN DESCRIPTION AND I/O PORTS	
2.1	PIN DESCRIPTIONS	2-1
2.2	Pin Assignments	2-2
2.3	INPUT/OUTPUT PORTS	2-3
2.3.1	Port A	2-3
2.3.2	Port B	2-3
2.3.3	Port C	2-4
2.3.4	Port D	2-4
2.3.5	Input/Output Programming	2-4
2.3.6	Port C and D Configuration Registers	2-5
	3 MEMORY AND REGISTERS	
3.1	Registers	3-1
3.2	RAM (MC68HC05BD3)	3-1
3.3	RAM (MC68HC705BD3/MC68HC05BD5)	3-1
3.4	ROM (MC68HC05BD3)	3-2
3.5	ROM (MC68HC05BD5)	3-2
3.6	EPROM (MC68HC705BD3)	3-2
3.7	Bootstrap ROM	3-2

4
RESETS AND INTERRUPTS

4.1 RESETS 4-1
4.1.1 Power-On Reset (POR) 4-1
4.1.2 $\overline{\text{RESET}}$ Pin 4-1
4.1.3 Illegal Address (ILADR) Reset 4-2
4.1.4 Computer Operating Properly (COP) Reset 4-2
4.2 INTERRUPTS 4-3
4.2.1 Non-maskable Software Interrupt (SWI) 4-3
4.2.2 Maskable Hardware Interrupts 4-5
4.2.2.1 External Interrupt ($\overline{\text{IRQ}}$) 4-5
4.2.2.2 Sync Signal Processor Interrupt 4-7
4.2.2.3 M-Bus Interrupts 4-7
4.2.2.4 Multi-Function Timer Interrupts 4-8

5
MULTI-FUNCTION TIMER

5.1 MFT Counter Register 5-1
5.2 MFT Control and Status Register 5-1
5.3 COP Watchdog 5-2

6
PULSE WIDTH MODULATION

6.1 PWM Registers 6-1
6.2 General Operation 6-1

7
M-BUS SERIAL INTERFACE

7.1 M-Bus Interface Features 7-1
7.2 M-Bus Protocol 7-2
7.2.1 START Signal 7-3
7.2.2 Slave Address Transmission 7-3
7.2.3 Data Transfer 7-4
7.2.4 Repeated START Signal 7-4
7.2.5 STOP Signal 7-4
7.2.6 Arbitration Procedure 7-4
7.2.7 Clock Synchronization 7-5
7.2.8 Handshaking 7-5
7.3 M-Bus Registers 7-5

7.3.1 M-Bus Address Register (MADR) 7-6
7.3.2 M-Bus Frequency Register (MFDR) 7-6
7.3.3 M-Bus Control Register (MCR) 7-7
7.3.4 M-Bus Status Register (MSR) 7-8
7.3.5 M-Bus Data I/O Register (MDR) 7-9
7.4 Programming Considerations 7-11
7.4.1 Initialization 7-11
7.4.2 Generation of a START Signal and the First Byte of Data Transfer 7-11
7.4.3 Software Responses after Transmission or Reception of a Byte 7-11
7.4.4 Generation of the STOP Signal 7-12
7.4.5 Generation of a Repeated START Signal 7-13
7.4.6 Slave Mode 7-13
7.4.7 Arbitration Lost 7-13

8
SYNC SIGNAL PROCESSOR

8.1 Functional Blocks 8-1
8.1.1 Polarity Correction 8-1
8.1.1.1 Separate Vertical Sync Input 8-2
8.1.1.2 Separate Horizontal Or Composite Sync Input 8-3
8.1.2 Sync Detection 8-3
8.1.3 Free-running Pseudo Sync Signal Generator 8-4
8.1.4 Sync Separation 8-4
8.1.5 Vertical Sync Pulse Reshaper 8-5
8.1.6 Sync Signal Counters 8-5
8.2 VSYNC Interrupt 8-5
8.3 Registers 8-7
8.3.1 Sync Signal Control & Status Register (SSCSR) 8-7
8.3.2 Vertical Frequency Registers (VFRS) 8-9
8.3.3 Line Frequency Registers (LFRs) 8-9
8.3.4 Sync Signal Control Register (SSCR) 8-10
8.3.5 Horizontal Sync Period Width Register (HPWR) 8-10
8.4 System Operation 8-11

9
CPU CORE AND INSTRUCTION SET

9.1 Registers 9-1
9.1.1 Accumulator (A) 9-1
9.1.2 Index register (X) 9-2
9.1.3 Program counter (PC) 9-2
9.1.4 Stack pointer (SP) 9-2
9.1.5 Condition code register (CCR) 9-2

9.2 Instruction set 9-3
9.2.1 Register/memory Instructions 9-4
9.2.2 Branch instructions 9-4
9.2.3 Bit manipulation instructions 9-4
9.2.4 Read/modify/write instructions 9-4
9.2.5 Control instructions 9-4
9.2.6 Tables 9-4
9.3 Addressing modes 9-11
9.3.1 Inherent 9-11
9.3.2 Immediate 9-11
9.3.3 Direct 9-11
9.3.4 Extended 9-12
9.3.5 Indexed, no offset 9-12
9.3.6 Indexed, 8-bit offset 9-12
9.3.7 Indexed, 16-bit offset 9-12
9.3.8 Relative 9-13
9.3.9 Bit set/clear 9-13
9.3.10 Bit test and branch 9-13

10
LOW POWER MODES

10.1 STOP Mode 10-1
10.2 WAIT Mode 10-1
10.3 COP Watchdog Timer Considerations 10-2

11
OPERATING MODES

11.1 User Mode (Normal Operation) 11-2
11.2 Self-Check Mode 11-2
11.3 Bootstrap Mode 11-4

12
ELECTRICAL SPECIFICATIONS

12.1 Maximum Ratings 12-1
12.2 Thermal Characteristics 12-1
12.3 DC Electrical Characteristics 12-2
12.4 Control Timing 12-3
12.5 M-Bus Timing 12-4
12.6 Sync Signal Processor Timing 12-5

Paragraph Number | TITLE | Page Number

13
MECHANICAL SPECIFICATIONS

13.1 42-Pin SDIP Package (Case 858-01) 13-1
13.2 40-Pin DIP Package (Case 711-03) 13-1

14
MC68HC705BD3

14.1 Features 14-1
14.2 Memory Map 14-1
14.3 EPROM Programming 14-1
14.3.1 Programming Control Register (PCR) 14-3
14.3.2 EPROM Programming Sequence 14-3
14.4 DC Electrical Characteristics 14-4

15
MC68HC05BD5

15.1 Features 15-1
15.2 Memory Map 15-1
15.3 DC Electrical Characteristics 15-3

THIS PAGE LEFT BLANK INTENTIONALLY

LIST OF FIGURES

Figure Number	TITLE	Page Number
1-1	MC68HC05BD3/MC68HC705BD3/MC68HC05BD5 Block Diagram	1-2
2-1	Pin Assignment for 40-pin DIP Package	2-2
2-2	Pin Assignment for 42-pin SDIP Package	2-3
2-3	Parallel Port I/O Circuitry	2-6
3-1	Memory Map	3-3
4-1	Power-On Reset and $\overline{\text{RESET}}$ Timing	4-2
4-2	Interrupt Stacking Order	4-4
4-3	External Interrupt Circuit and Timing	4-6
6-1	8-Bit PWM Output Waveforms	6-2
7-1	M-Bus Interface Block Diagram	7-2
7-2	M-Bus Transmission Signal Diagram	7-3
7-3	Clock Synchronization	7-5
7-4	Flowchart of M-Bus Interrupt Routine	7-10
8-1	Sync Signal Processor Block Diagram	8-2
8-2	Sync Signal Polarity Correction	8-3
8-3	Sync Separator	8-4
8-4	Sync Signal Counters Block Diagram	8-6
8-5	Vertical Frequency Counter Timing	8-6
8-6	Typical Monitor System Operation	8-12
9-1	Programming model	9-1
9-2	Stacking order	9-2
10-1	WAIT Flowchart	10-2
11-1	Flowchart of Mode Entering	11-1
11-2	Self-Check Mode Timing	11-2
11-3	MC68HC05BD3 Self-Test Circuit	11-3
12-1	M-Bus Timing	12-4
14-1	MC68HC705BD3 Memory Map	14-2
15-1	MC68HC05BD5 Memory Map	15-2

THIS PAGE LEFT BLANK INTENTIONALLY

LIST OF TABLES

Table Number	TITLE	Page Number
2-1	I/O Pin Functions	2-4
2-2	Configuration for PC6 and PC7	2-5
3-1	Register Outline	3-4
4-1	Reset/Interrupt Vector Addresses	4-4
5-1	COP Reset and RTI Rates	5-3
7-1	M-Bus Prescaler	7-6
8-1	Vertical Frame Frequencies	8-9
9-1	MUL instruction	9-5
9-2	Register/memory instructions	9-5
9-3	Branch instructions	9-6
9-4	Bit manipulation instructions	9-6
9-5	Read/modify/write instructions	9-7
9-6	Control instructions	9-7
9-7	Instruction set	9-8
9-8	M68HC05 opcode map	9-10
11-1	Mode Selection	11-2
11-2	Self-Check Report	11-4
12-1	DC Electrical Characteristics for MC68HC05BD3	12-2
12-2	Control Timing	12-3
12-3	M-Bus Interface Input Signal Timing	12-4
12-4	M-Bus Interface Output Signal Timing	12-4
12-5	Sync Signal Processor Timing	12-5
14-1	MC68HC705BD3 Programming Boards	14-1
14-2	DC Electrical Characteristics for MC68HC705BD3	14-4
15-1	DC Electrical Characteristics for MC68HC05BD5	15-3

THIS PAGE LEFT BLANK INTENTIONALLY

1
GENERAL DESCRIPTION

The MC68HC05BD3 HCMOS microcontroller is a member of the M68HC05 Family of low-cost single-chip microcontrollers. This 8-bit microcontroller unit (MCU) contains an on-chip oscillator, CPU, RAM, ROM, parallel I/O capability with pins programmable as input or output, M-Bus serial interface system (I^2C), Pulse Width Modulator, Multi-Function Timer, and Sync Signal Processor. These features make it particularly suitable as a multi-sync computer monitor MCU.

The MC68HC05BD5 is functionally equivalent to MC68HC05BD3, with increase RAM and ROM. The MC68HC705BD3 is an EPROM version of the MC68HC05BD5. All references to the MC68HC05BD3 apply equally to the MC68HC705BD3 and MC68HC05BD5, unless otherwise stated. *References specific to the MC68HC705BD3 are italicized in the text and also, for quick reference, they are summarized in Section 14.* References to MC68HC05BD5 are summarized in Section 15.

1.1 Features

- Fully static chip design featuring the industry standard 8-bit M68HC05 core
- Power saving Wait mode
- 128 bytes of on-chip RAM for MC68HC05BD3
  256 bytes for *MC68HC705BD3* and MC68HC05BD5
- 3.75 K-bytes of user ROM for MC68HC05BD3
  7.75 K-bytes of user ROM for MC68HC05BD5
  7.75 K-bytes of user EPROM for MC68HC705BD3
- 24 bidirectional I/O lines: 14 dedicated and 10 multiplexed I/O lines.
  4 of the 14 dedicated I/O lines and 6 of the MUXed I/O lines have +10V open-drain O/Ps.
- 16 channels PWM outputs: 8 dedicated, +10V open-drain PWM channels and
  8 multiplexed with I/O lines of which 6 of them have +10V open-drain outputs.
- M-Bus Serial Interface (I^2C-bus†)
- Multi-Function Timer (MFT) with Periodic Interrupt
- COP watchdog reset

† I^2C-bus is a proprietary Philips interface bus

- Horizontal and Vertical Sync Signal Processor
- Self-check mode
- Available in 40-pin DIP and 42-pin SDIP packages

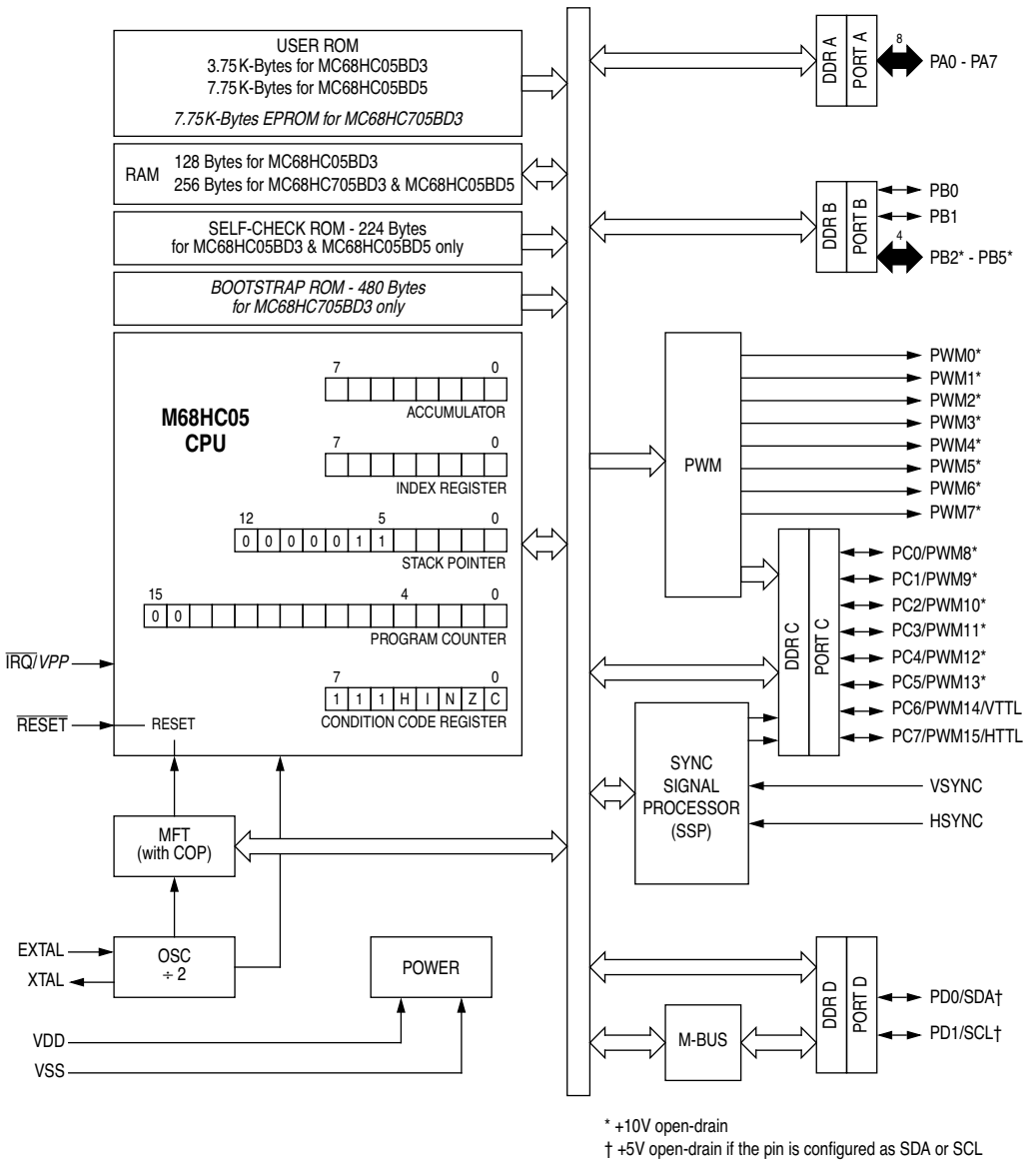


Figure 1-1 MC68HC05BD3/MC68HC705BD3/MC68HC05BD5 Block Diagram

2
PIN DESCRIPTION AND I/O PORTS

This section provides a description of the functional pins and I/O programming of the MC68HC05BD3 microcontroller.

2.1 PIN DESCRIPTIONS

PIN NAME	40-pin DIP PIN No.	42-pin SDIP PIN No.	DESCRIPTION
VDD, VSS	5, 6	5, 7	Power is supplied to the MCU using these two pins. VDD is power and VSS is ground.
$\overline{IRQ}$/*VPP*	15	16	In the user mode this pin an external hardware interrupt $\overline{IRQ}$. It is software programmable to provide two choices of interrupt triggering sensitivity. These options are: 1) negative edge-sensitive triggering only, or 2) both negative edge-sensitive and level sensitive triggering. *In the bootstrap mode on the MC68HC705BD3, this is the EPROM programming voltage input pin.*
$\overline{RESET}$	4	4	The active low $\overline{RESET}$ input is not required for start-up, but can be used to reset the MCU internal state and provide an orderly software start-up procedure.
XTAL, EXTAL	7, 8	8, 9	These pins provide connections to the on-chip oscillator. The oscillator can be driven by a AT-crystal circuit or a ceramic resonator with a frequency of 4.2 MHz. EXTAL may also be driven by an external oscillator if an external crystal/resonator circuit is not used. See Figure 11-3 for values of crystal circuit external components.
PA0-PA7	23-16	24-17	These eight I/O lines comprise port A. The state of any pin is software programmable. All port A lines are configured as input during power on or external reset.
PB0-PB5	14-9	15-10	These six I/O lines comprise port B. The state of any pin is software programmable. All port B lines are configured as input during power on or external reset. PB2 to PB5 are +10V open-drain pins.
PC0/PWM8 to PC5/PWM13	26-31	27-32	These six port C pins are +10V open-drain type. The state of any pin is software programmable. All port C lines are configured as input during power on or external reset. These pins become PWM output channels 8 to 13 by setting the appropriate bits in Configuration register 1 ($0A).

PIN NAME	40-pin DIP PIN No.	42-pin SDIP PIN No.	DESCRIPTION
PC6/PWM14/VTTL, PC7/PWM15/HTTL	32, 33	33, 34	These two port C I/O lines are shared with the PWM and Sync Signal Processor. Configuration for use are set by the Configuration register 1 ($0A) and Configuration register 2 ($0B).
PD0/SDA, PD1/SCL	24, 25	25, 26	These two port D I/O lines are shared with the M-Bus lines SDA and SCL. When configured as M-Bus lines in Configuration register 2 ($0B), these pins become +5V open-drain pins.
PWM0 to PWM7	3-1, 38-34	3-1, 40-38, 36, 35	These eight pins are dedicated for the 8-bit PWM channel 0 to 7.
HSYNC, VSYNC	39, 40	41, 42	These two pins are for video sync signals input from the host computer. The polarity of the input signals can either be positive or negative. These two pins contain internal Schmitt triggers as part of their inputs to improve noise immunity

2.2 Pin Assignments

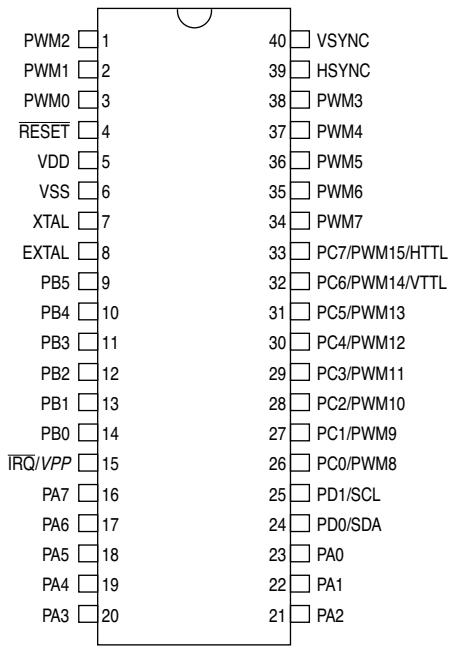


Figure 2-1 Pin Assignment for 40-pin DIP Package

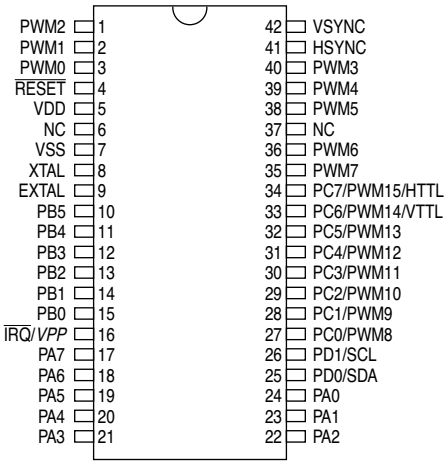


Figure 2-2 Pin Assignment for 42-pin SDIP Package

2.3 INPUT/OUTPUT PORTS

In the User Mode there are 24 bidirectional I/O lines arranged as 4 I/O ports (Port A, B, C, and D). The individual bits in these ports are programmable as either inputs or outputs under software control by the data direction registers (DDRs). Also, if enabled by software, Port C and D will have additional functions as PWM outputs, M-Bus I/O and Sync Signal Processor outputs.

2.3.1 Port A

Port A is an 8-bit bidirectional port which does not share any of its pins with other subsystems. The Port A data register is at $00 and the data direction register (DDR) is at $04. Reset does not affect the data register, yet 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.

2.3.2 Port B

Port B is a 6-bit bidirectional port which does not share any of its pins with other subsystems. PB2 to PB5 are +10V open-drain port pins. The Port B data register is at $01 and the data direction register (DDR) is at $05. Reset does not affect the data register, yet 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.

2.3.3 Port C

Port C is an 8-bit bidirectional port which shares pins with PWM and SSP subsystem. See Section 6 for a detailed description of PWM and Section 8 for a detailed description of SSP. These pins are configured to PWM output when the corresponding bits in the Configuration register 1 are set, except PC6 and PC7. PC6 and PC7 are configured to SSP outputs when the corresponding bits in the Configuration register 2 are set. The Port C data register is at $02 and the data direction register (DDR) is at $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.

2.3.4 Port D

Port D is a 2-bit bidirectional port which shares pins with M-Bus subsystem. See Section 7 for a detailed description of M-Bus. These pins are configured to the corresponding functions when the corresponding bits in the Configuration register 2 are set. They are +5V open-drain I/O pins when used as M-Bus I/O. The Port D data register is at $03 and the data direction register (DDR) is at $07. Reset does not affect the data register, yet 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.

2.3.5 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. A pin is configured as an input if its corresponding DDR bit is cleared.

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

Table 2-1 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 an output mode. The output data latch is read.

2.3.6 Port C and D Configuration Registers

Port C and Port D are shared with PWM, M-Bus and SSP. The configuration registers at $0A and $0B are used to configure those I/O pins. They are default to zero after power-on reset. Setting these bits will set the corresponding pins to the corresponding functions, except PC6 and PC7. For example, setting SCL and SDA bits of register $0B will configure Port D pins 1 and 0 as M-Bus pins, regardless of DDR1 and DDR0 settings.

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Configuration Register 1	$000A	PWM15	PWM14	PWM13	PWM12	PWM11	PWM10	PWM9	PWM8	0000 0000

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Configuration Register 2	$000B	HTTL	VTTL					SCL	SDA	0000 0000

PC7 and PC6 are shared with both PWM and SSP. When HTTL and VTTL in $000B are set, PC7 and PC6 are configured as HTTL and VTTL outputs respectively, regardless of the status of PWM15 and PWM14 in $000A. That is, HTTL and VTTL settings override PWM15 and PWM14 settings.

Table 2-2 Configuration for PC6 and PC7

PWM15	HTTL	Result of PC7
0	0	PC7
0	1	HTTL
1	0	PWM15
1	1	HTTL

PWM14	VTTL	Result of PC6
0	0	PC6
0	1	VTTL
1	0	PWM14
1	1	VTTL

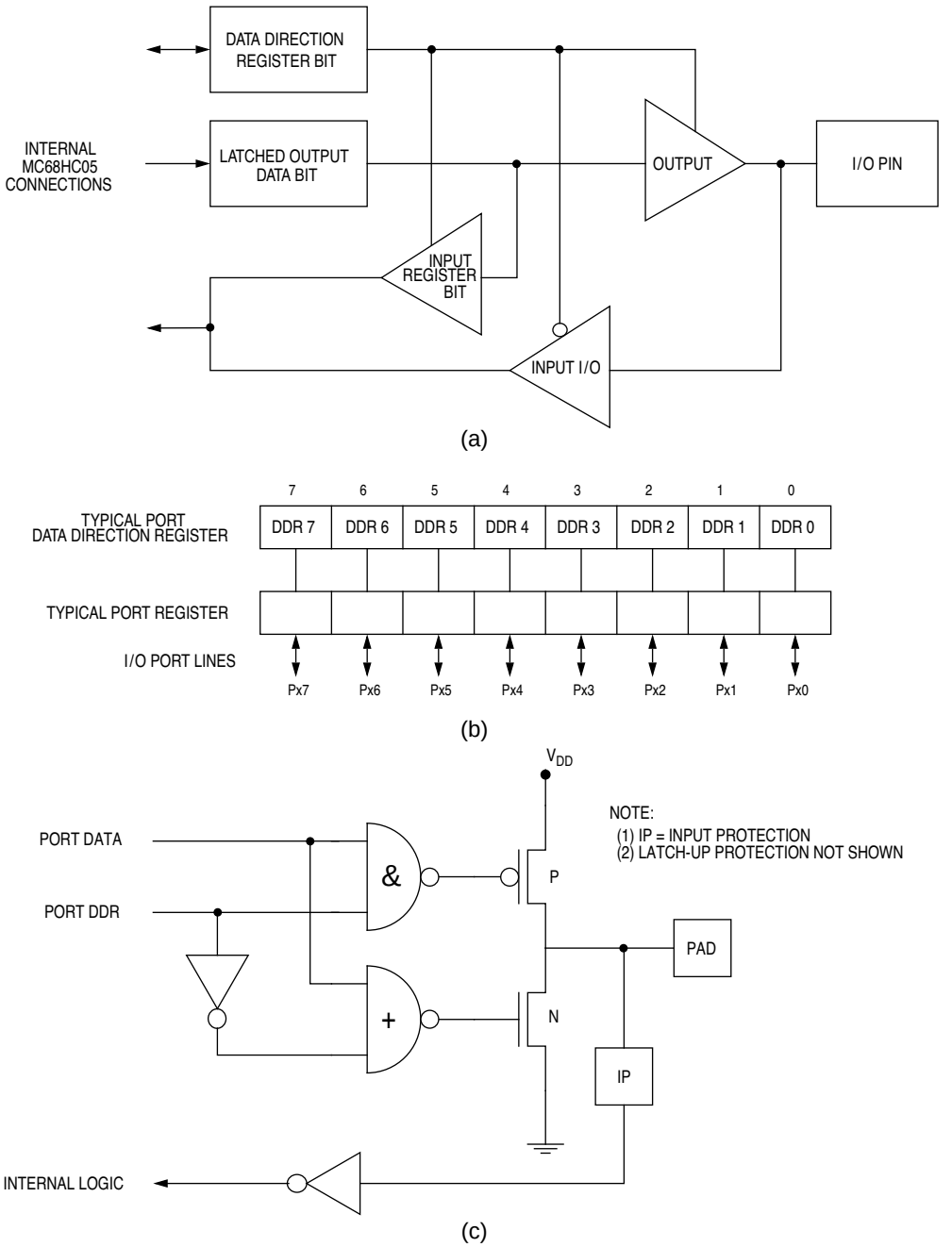


Figure 2-3 Parallel Port I/O Circuitry

3
MEMORY AND REGISTERS

The MC68HC05BD3/MC68HC705BD3/MC68HC05BD5 has a 16K-byte memory map consisting of registers, user ROM/*EPROM*, user RAM, self-check/*bootstrap* ROM, and I/O as shown in Figure 3-1.

3.1 Registers

All the I/O, control and status registers of the MC68HC05BD3 are contained within the first 48-byte block of the memory map (address $0000 to $002F).

3.2 RAM (MC68HC05BD3)

The user RAM consists of 128 bytes of memory, from $0080 to $00FF. This is shared with a 64 byte stack area. The stack begins at $00FF and counts down to $00C0.

3.3 RAM (MC68HC705BD3/MC68HC05BD5)

The user RAM consists of 256 bytes of memory, from $0080 to $017F. This is shared with a 64 byte stack area. The stack begins at $00FF and counts down to $00C0.

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

3.4 ROM (MC68HC05BD3)

The user ROM consists of 3.75K-bytes of memory, from $3000 to $3EFF.

3

3.5 ROM (MC68HC05BD5)

The user ROM consists of 7.75K-bytes of memory, from $2000 to $3EFF.

3.6 EPROM (MC68HC705BD3)

The user EPROM consists of 7.75K-bytes of memory, from $2000 to $3EFF.

3.7 Bootstrap ROM

This is available on the MC68HC705BD3 device only. It stores the on-chip program for programming the user EPROM.

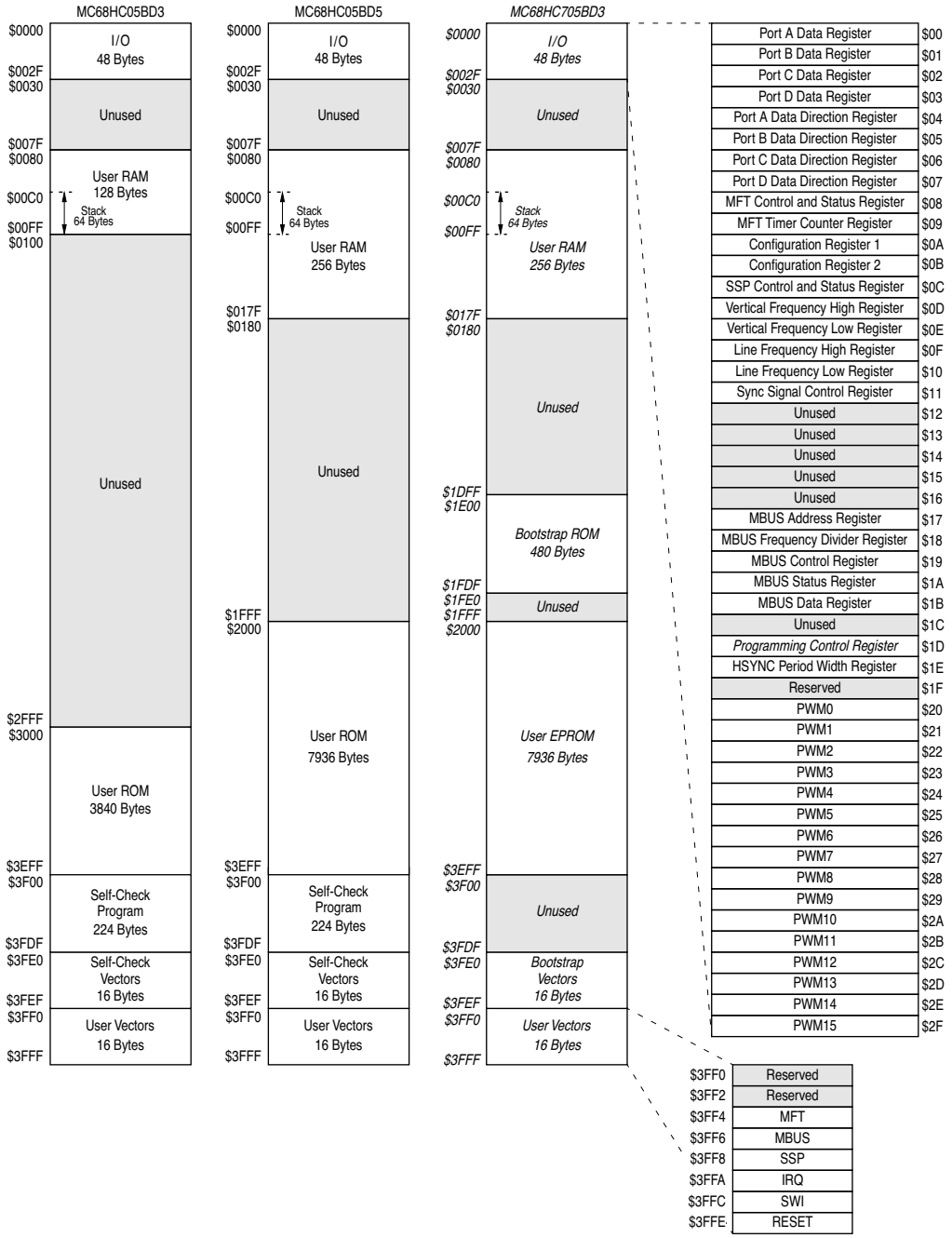


Figure 3-1 Memory Map

Table 3-1 Register Outline

Register Name	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Port A data	$0000	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0	undefined
Port B data	$0001			PB5	PB4	PB3	PB2	PB1	PB0	undefined
Port C data	$0002	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0	undefined
Port D data	$0003							PD1	PD0	undefined
Port A data direction	$0004	DDRA7	DDRA6	DDRA5	DDRA4	DDRA3	DDRA2	DDRA1	DDRA0	0000 0000
Port B data direction	$0005			DDRB5	DDRB4	DDRB3	DDRB2	DDRB1	DDRB0	--00 0000
Port C data direction	$0006	DDRC7	DDRC6	DDRC5	DDRC4	DDRC3	DDRC2	DDRC1	DDRC0	0000 0000
Port D data direction	$0007							DDRD1	DDRD0	---- --00
MFT control and status	$0008	TOF	RTIF	TOFIE	RTIE	IRQN		RT1	RT0	0000 0-11
MFT timer counter	$0009	MFTCR7	MFTCR6	MFTCR5	MFTCR4	MFTCR3	MFTCR2	MFTCR1	MFTCR0	0000 0000
Configuration 1	$000A	PWM15	PWM14	PWM13	PWM12	PWM11	PWM10	PWM9	PWM8	0000 0000
Configuration 2	$000B	HTTL	VTTL					SCL	SDA	00-- --00
SSP control and status	$000C	VPOL	HPOL	VDET	HDET	SOUT	INSRTB	FOUT	VSIN	0000 0000
Vertical frequency high	$000D	0	0	0	VF12	VF11	VF10	VF9	VF8	0000 0000
Vertical frequency low	$000E	VF7	VF6	VF5	VF4	VF3	VF2	VF1	VF0	0000 0000
Line frequency high	$000F	HOVER	0	0	0	LF11	LF10	LF9	LF8	0000 0000
Line frequency low	$0010	LF7	LF6	LF5	LF4	LF3	LF2	LF1	LF0	0000 0000
Sync signal control	$0011	VSIE	0	0	0	0	0	0	0	0000 0000
Unused	$0012									
Unused	$0013									
Unused	$0014									
Unused	$0015									
Unused	$0016									
MBUS address	$0017	MAD7	MAD6	MAD5	MAD4	MAD3	MAD2	MAD1		0000 000-
MBUS frequency divider	$0018				FD4	FD3	FD2	FD1	FD0	---0 0000
MBUS control	$0019	MEN	MIEN	MSTA	MTX	TXAK				0000 0---
MBUS status	$001A	MCF	MASS	MBB	MAL		SRW	MIF	RXAK	1000 -001
MBUS data	$001B	MD7	MD6	MD5	MD4	MD3	MD2	MD1	MD0	undefined
Unused	$001C									
Programming Control	$001D							ELAT	PGM	---- --00
HSYNC period width	$001E	HPWR7	HPWR6	HPWR5	HPWR4	HPWR3	HPWR2	HPWR1	HPWR0	0000 0000
Reserved	$001F									

Table 3-1 Register Outline

Register Name	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
0PWM	$0020	0PWM4	0PWM3	0PWM2	0PWM1	0PWM0	0BRM2	0BRM1	0BRM0	0000 0000
1PWM	$0021	1PWM4	1PWM3	1PWM2	1PWM1	1PWM0	1BRM2	1BRM1	1BRM0	0000 0000
2PWM	$0022	2PWM4	2PWM3	2PWM2	2PWM1	2PWM0	2BRM2	2BRM1	2BRM0	0000 0000
3PWM	$0023	3PWM4	3PWM3	3PWM2	3PWM1	3PWM0	3BRM2	3BRM1	3BRM0	0000 0000
4PWM	$0024	4PWM4	4PWM3	4PWM2	4PWM1	4PWM0	4BRM2	4BRM1	4BRM0	0000 0000
5PWM	$0025	5PWM4	5PWM3	5PWM2	5PWM1	5PWM0	5BRM2	5BRM1	5BRM0	0000 0000
6PWM	$0026	6PWM4	6PWM3	6PWM2	6PWM1	6PWM0	6BRM2	6BRM1	6BRM0	0000 0000
7PWM	$0027	7PWM4	7PWM3	7PWM2	7PWM1	7PWM0	7BRM2	7BRM1	7BRM0	0000 0000
8PWM	$0028	8PWM4	8PWM3	8PWM2	8PWM1	8PWM0	8BRM2	8BRM1	8BRM0	0000 0000
9PWM	$0029	9PWM4	9PWM3	9PWM2	9PWM1	9PWM0	9BRM2	9BRM1	9BRM0	0000 0000
10PWM	$002A	10PWM4	10PWM3	10PWM2	10PWM1	10PWM0	10BRM2	10BRM1	10BRM0	0000 0000
11PWM	$002B	11PWM4	11PWM3	11PWM2	11PWM1	11PWM0	11BRM2	11BRM1	11BRM0	0000 0000
12PWM	$002C	12PWM4	12PWM3	12PWM2	12PWM1	12PWM0	12BRM2	12BRM1	12BRM0	0000 0000
13PWM	$002D	13PWM4	13PWM3	13PWM2	13PWM1	13PWM0	13BRM2	13BRM1	13BRM0	0000 0000
14PWM	$002E	14PWM4	14PWM3	14PWM2	14PWM1	14PWM0	14BRM2	14BRM1	14BRM0	0000 0000
15PWM	$002F	15PWM4	15PWM3	15PWM2	15PWM1	15PWM0	15BRM2	15BRM1	15BRM0	0000 0000

THIS PAGE LEFT BLANK INTENTIONALLY

4
RESETS AND INTERRUPTS

4.1 RESETS

The MC68HC05BD3 can be reset in four ways: by the initial power-on reset function, by an active low input to the $\overline{\text{RESET}}$ pin, by an opcode fetch from an illegal address, and by a COP watchdog timer reset. Any of these resets will cause the program to go to its starting address, specified by the contents of memory locations $3FFE and $3FFF, and cause the interrupt mask of the Condition Code register to be set.

4.1.1 Power-On Reset (POR)

The power-on reset occurs when a positive transition is detected on the supply voltage, V_{DD}. The power-on reset is used strictly for power-up conditions, and should not be used to detect any drops in the power supply voltage. There is no provision for a power-down reset. The power-on circuitry provides for a 4064 tcyc delay from the time that the oscillator becomes active. If the external $\overline{\text{RESET}}$ pin is low at the end of the 4064 tcyc time out, the processor remains in the reset condition until $\overline{\text{RESET}}$ goes high. The user must ensure that V_{DD} has risen to a point where the MCU can operate properly prior to the time the 4064 POR cycles have elapsed. If there is doubt, the external $\overline{\text{RESET}}$ pin should remain low until such time that V_{DD} has risen to the minimum operating voltage specified.

4.1.2 $\overline{\text{RESET}}$ Pin

The $\overline{\text{RESET}}$ input pin is used to reset the MCU to provide an orderly software start-up procedure. When using the external reset, the $\overline{\text{RESET}}$ pin must stay low for a minimum of 1.5tcyc. The $\overline{\text{RESET}}$ pin contains an internal Schmitt Trigger as part of its input to improve noise immunity.

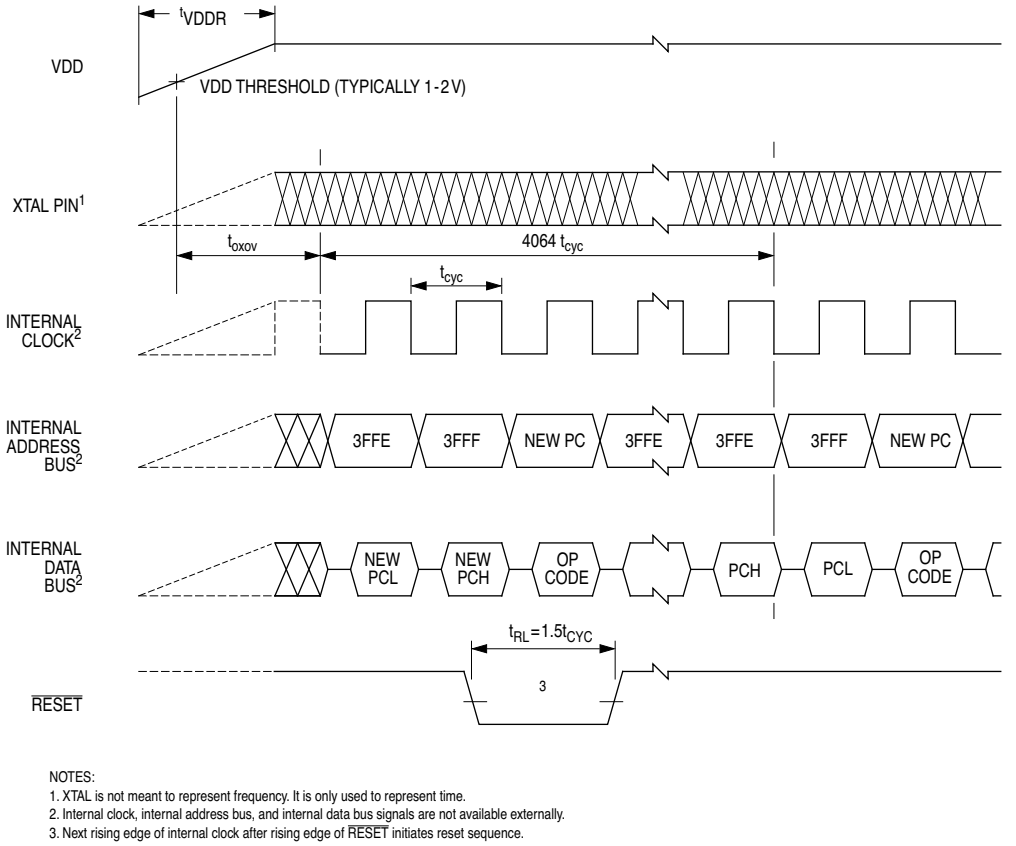


Figure 4-1 Power-On Reset and $\overline{\text{RESET}}$ Timing

4.1.3 Illegal Address (ILADR) Reset

The MCU monitors all opcode fetches. If an illegal address space is accessed during an opcode fetch, an internal reset is generated. Illegal address spaces consist of all unused locations within the memory map and the I/O registers (see Figure 3-1). Because the internal reset signal is used, the MCU comes out of an ILADR reset in the same operating mode it was in when the opcode was fetched.

4.1.4 Computer Operating Properly (COP) Reset

The MCU contains a watchdog timer that automatically times out if not reset (cleared) within a specific amount of time by a program reset sequence.

Note: COP time-out is prevented by periodically writing a “0” to bit 0 of address $3FF0.

If the watchdog timer is allowed to time-out, an internal reset is generated to reset the MCU. Because the internal reset signal is used, the MCU comes out of a COP reset in the same operating mode it was in when the COP time-out was generated.

The COP reset function is always enabled.

See Section 5.3 for more information on the COP watchdog timer.

4.2 INTERRUPTS

The MCU can be interrupted by different sources – four maskable hardware interrupt and one non-maskable software interrupt:

- External signal on the $\overline{IRQ}$ pin
- Multi-Function Timer (MFT)
- M-Bus Interface (MBUS)
- Sync Signal Processor (SSP)
- 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 the 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. 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.

Table 4-1 shows the relative priority of all the possible interrupt sources.

4.2.1 Non-maskable Software Interrupt (SWI)

The software interrupt (SWI) is an executable instruction and a non-maskable interrupt: it is execute regardless of the state of the I-bit in the CCR. If the I-bit is zero (interrupt enabled), SWI is executed after interrupts that were pending when the SWI was fetched, but before interrupts generated after the SWI was fetched. The SWI interrupt service routine address is specified by the contents of memory locations $3FFC and $3FFD.

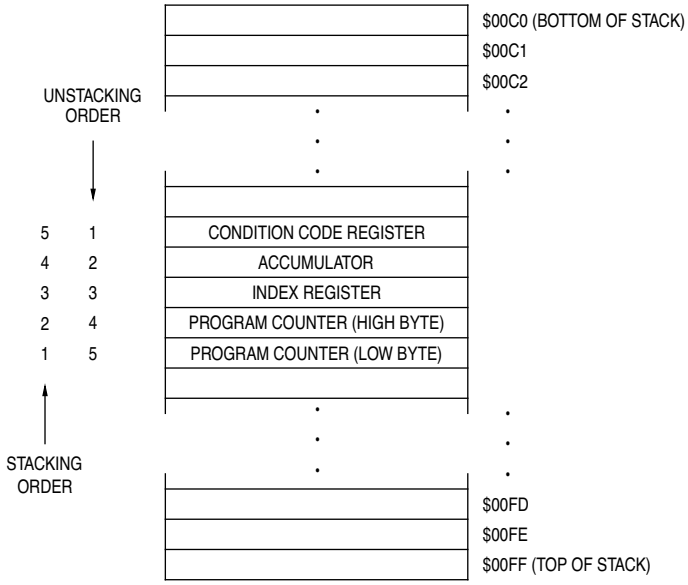


Figure 4-2 Interrupt Stacking Order

Table 4-1 Reset/Interrupt Vector Addresses

Register	Flag Name	Interrupt	CPU Interrupt	Vector Address	Priority
–	–	Reset	$\overline{\text{RESET}}$	$3FFE-$3FFF	highest
–	–	Software	SWI	$3FFC-$3FFD	
–	–	External Interrupt	$\overline{\text{IRQ}}$	$3FFA-$3FFB	
SSCR	–	VSYNC	SSP	$3FF8-$3FF9	
MSR	MIF	M-Bus	MBUS	$3FF6-$3FF7	
MFTCSR	TOF	Timer Overflow	MFT	$3FF4-$3FF5	
	RTIF	Real Time Interrupt			
–	–	–	–	$3FF2-$3FF3	
–	–	–	–	$3FF0-$3FF1	lowest

4.2.2 Maskable Hardware Interrupts

If the interrupt mask bit (I-bit) of the CCR is set, all maskable interrupts (internal and external) are masked. Clearing the I-bit allows interrupt processing to occur.

Note: The internal interrupt latch is 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.

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

The external interrupt $\overline{IRQ}$ can be software configured for “negative-edge” or “negative-edge and level” sensitive triggering by the IRQN bit in the Multi-Function TImer Control and Status register.

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
MFT Control and Status	$0008	TOF	RTIF	TOFIE	RTIE	IRQN		RT1	RT0	0000 0-11

IRQN

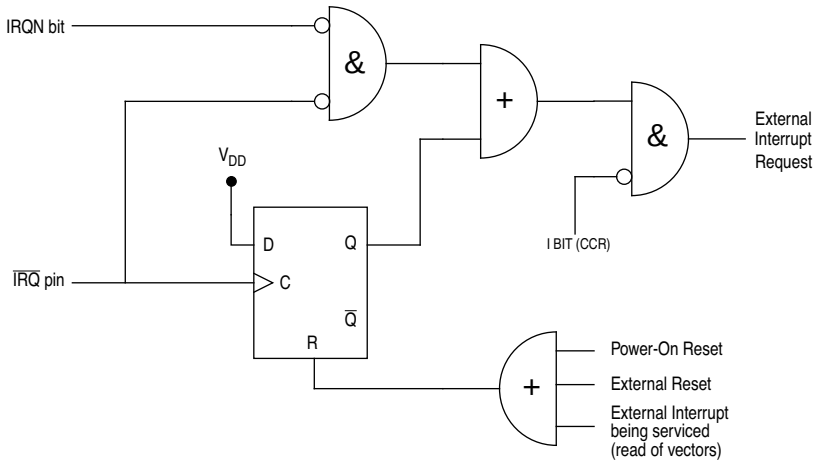
1 (set) – Negative edge triggering for $\overline{IRQ}$ only

0 (clear) – Level and negative edge triggering for $\overline{IRQ}$

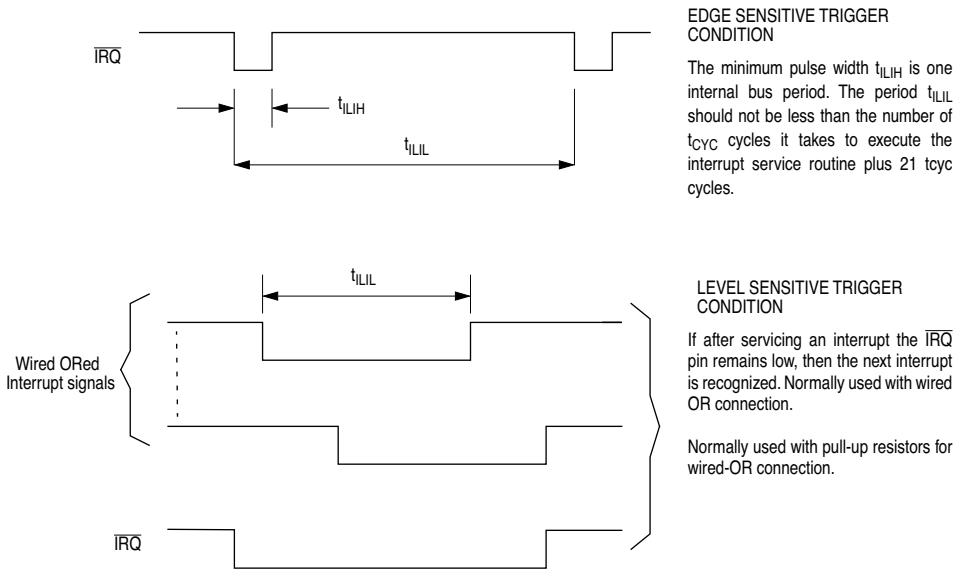
When the signal of the external interrupt pin, $\overline{IRQ}$, satisfies the condition selected, an external interrupt occurs. The actual processor interrupt is generated only if the interrupt mask bit of the condition code register is also cleared. When the interrupt is recognized, the current state of the processor is pushed onto the stack and the interrupt mask bit in the condition code register is set. This masks further interrupts until the present one is serviced. The service routine address is specified by the contents $3FFA & $3FFB.

The interrupt logic recognizes negative edge transitions and pulses (special case of negative edges) on the external interrupt line. Figure 4-3 shows both a block diagram and timing for the interrupt line ($\overline{IRQ}$) to the processor. The first method is used if pulses on the interrupt line are spaced far enough apart to be serviced. The minimum time between pulses is equal to the number of cycles required to execute the interrupt service routine plus 21 cycles. Once a pulse occurs, the next pulse should not occur until the MCU software has exited the routine (an RTI occurs). The second configuration shows several interrupt lines wired-OR to perform the interrupt at the processor. Thus, if the interrupt lines remain low after servicing one interrupt, the next interrupt is recognized.

Note: The internal interrupt latch is cleared in the first part of the service routine; therefore, one (and only one) external interrupt pulse could be latched during t_{ILIL} and serviced as soon as the I bit is cleared.



(a) Interrupt Function Diagram



(b) Interrupt Mode Diagram

Figure 4-3 External Interrupt Circuit and Timing

4.2.2.2 Sync Signal Processor Interrupt

The VSYNC interrupt is generated by the Sync Signal Processor (SSP) after a vertical sync pulse is detected as described in Section 8. The interrupt enable bit, VSIE, for the VSYNC interrupt is located at bit 7 of Sync Signal Control register (SSCR) at $0011. The I-bit in the CCR must be cleared in order for the VSYNC interrupt to be enabled. This interrupt will vector to the interrupt service routine located at the address specified by the contents of $3FF8 and $3FF9. The VSYNC interrupt latch will be cleared automatically by fetching of these vectors.

Refer to Section 8 for detailed description of Sync Signal Processor.

4.2.2.3 M-Bus Interrupts

M-Bus interrupt is enabled when the M-Bus Interrupt Enable bit (MIEN) of M-Bus Control register is set, provided the interrupt mask bit of the Condition Code register is cleared. The interrupt service routine address is specified by the contents of memory location $3FF6 and $3FF7.

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
M-Bus Status Register	$001A	MCF	MAAS	MBB	MAL		SRW	MIF	RXAK	1000 0001

MIF - M-Bus Interrupt

1 (set) – An M-Bus interrupt has occurred.

0 (clear) – An M-Bus interrupt has not occurred.

When this bit is set, an interrupt is generated to the CPU if MIEN is set. This bit is set when one of the following events occurs:

1) Completion of one byte of data transfer. It is set at the falling edge of the 9th clock - MCF set.
2) A match of the calling address with its own specific address in slave mode - MAAS set.
3) A loss of bus arbitration - MAL set.

This bit must be cleared by software in the interrupt routine.

MCF - Data Transfer Complete

1 (set) – A byte transfer has been completed.

0 (clear) – A byte is being transfer.

MAAS - Addressed as Slave

1 (set) – Currently addressed as a slave.

0 (clear) – Not currently addressed.

Then CPU needs to check the SRW bit and set its MTX bit accordingly. Writing to the M-Bus Control register clears this bit.

MAL - Arbitration Lost

1 (set) – Lost arbitration in master mode.

0 (clear) – No arbitration lost.

Refer to Section 7 for detailed description of M-Bus Interface.

4.2.2.4 Multi-Function Timer Interrupts

There are two interrupt sources, TOF and RTIF bits of Multi-Function Timer Control and Status Register. The interrupt service routine address is specified by the contents of memory location $3FF4 and $3FF5.

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
MFT Control and Status Register	$0008	TOF	RTIF	TOFIE	RTIE	IRQN		RT1	RT0	0000 0011

TOF - Timer Overflow

1 (set) – 8-bit ripple timer overflow has occurred.

0 (clear) – No 8-bit ripple timer overflow has occurred.

This bit is set when the 8-bit ripple counter overflows from $FF to $00; a timer overflow interrupt will occur, if TOFIE is set. TOF is cleared by writing a “0” to the bit.

RTIF - Real Time Interrupt Flag

1 (set) – A real time interrupt has occurred.

0 (clear) – A real time interrupt has not occurred.

The clock frequency that drives the RTI circuit is E/16384, giving a maximum interrupt period of 8.19 ms at a bus clock rate of 2 MHz. A CPU interrupt request will be generated if RTIE is set. RTIF is cleared by writing a “0” to the bit.

Refer to Section 5 for detailed description of Multi-Function Timer.

5
MULTI-FUNCTION TIMER

The MFT provides miscellaneous functions to the MC68HC05BD3 MCU. It includes a timer overflow function, real-time interrupt, and COP watchdog. The external interrupt ($\overline{IRQ}$) triggering option is also set by this module's MFT Control and Status Register.

The clock base for this module is derived from the bus clock divided by four. For a 2MHz E (CPU) clock, the clock base is 0.5MHz. This clock base is then divided by an 8-stage ripple counter to generate the timer overflow. Timer overflow rate is thus E/1024. The output of this 8-stage ripple counter then drives a 4-stage divider to generate real time interrupt. Hence, the clock base for real time interrupt is E/16384. Real time interrupt rate is selected by RT0 and RT1 bits of MFT Control and Status register. The interrupt rates are E/16384, (E/16384)/2, (E/16384)/4, and (E/16384)/8. The selected real time interrupt rate is then divided by 8 to generate COP reset.

5.1 MFT Counter Register

The MFT counter register (MFTCR) can be read at location $0009. It is cleared by reset.

5.2 MFT Control and Status Register

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
MFT Control and Status Register	$1C	TOF	RTIF	TOFIE	RTIE	IRQN		RT1	RT0	0000 0011

Register bit definitions:

TOF - Timer Overflow

1 (set) – 8-bit ripple timer overflow has occurred.

0 (clear) – No 8-bit ripple timer overflow has occurred.

This bit is set when the 8-bit ripple counter overflows from $FF to $00; a timer overflow interrupt will occur, if TOFIE (bit 5) is set. TOF is cleared by writing a “0” to the bit.

RTIF - Real Time Interrupt Flag

1 (set) – A real time interrupt has occurred.

0 (clear) – A real time interrupt has not occurred.

When RTIF is set, a CPU interrupt request is generated if RITE is set. The clock frequency that drives the RTI circuit is E/16384 giving a maximum interrupt period of 8.19ms at a bus rate of 2MHz. RTIF is cleared by writing a “0” to the bit.

TOFIE - Timer Overflow Interrupt Enable

1 (set) – TOF interrupt is enabled.

0 (clear) – TOF interrupt is disabled.

RTIE - Real Time Interrupt Enable

1 (set) – Real time interrupt is enabled.

0 (clear) – Real time interrupt is disabled.

IRQN - $\overline{\text{IRQ}}$ Pin Trigger Option

1 (set) – Negative edge triggering for $\overline{\text{IRQ}}$ only

0 (clear) – Level and negative edge triggering for $\overline{\text{IRQ}}$

RT1, RT0 - Rate Select for COP watchdog and RTI

See Section 5.3 on watchdog reset.

5.3 COP Watchdog

The COP (Computer Operating Properly) watchdog timer function is implemented by using the output of the Multi-Function Timer counter. The minimum COP reset rates are controlled by RT0 and RT1 of MFT Control and Status register. If the COP circuit times out, an internal reset is generated and the reset vector is fetched (at $3FFE & $3FFF). Preventing a COP time-out is achieved by writing a “0” to bit 0 of address $3FF0. The COP counter has to be cleared periodically by software with a period less than COP reset rate. The COP watchdog timer is always enabled and continues to count in Wait mode.

Table 5-1 COP Reset and RTI Rates

RT1	RT0	Minimum COP reset period		RTI period	
		COP	E clock = 2MHz	RTI	E clock = 2MHz
0	0	E/16384/7/1	57.344ms	E/16384/1	8.192ms
0	1	E/16384/7/2	114.688ms	E/16384/2	16.384ms
1	0	E/16384/7/4	229.376ms	E/16384/4	32.768ms
1	1	E/16384/7/8	458.752ms	E/16384/8	65.536ms
Note:	RT0 and RT1 should only be changed immediately after COP watchdog timer has been reset.				

THIS PAGE LEFT BLANK INTENTIONALLY

6
PULSE WIDTH MODULATION

The MC68HC05BD3 has 16 PWM channels. Channel 0 to 7 are dedicated PWM channels. Channel 8 to 15 are shared with port C I/O pins, and are selected by the respective bits in Configuration register 1. PWM channels 0 to 13 are +10V open-drain type; therefore a pull-up resistor is required at each of the pins.

6.1 PWM Registers

Each PWM channel has an 8-bit register which contains a 5-bit PWM in the MSB portion and a 3-bit binary rate multiplier (BRM) in the LSB portion. The PWM channel data registers are located from $20 to $2F.

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
0PWM	$0020	0PWM4	0PWM3	0PWM2	0PWM1	0PWM0	0BRM2	0BRM1	0BRM0	0000 0000
:						:				
15PWM	$002F	15PWM4	15PWM3	15PWM2	15PWM1	15PWM0	15BRM2	15BRM1	15BRM0	0000 0000

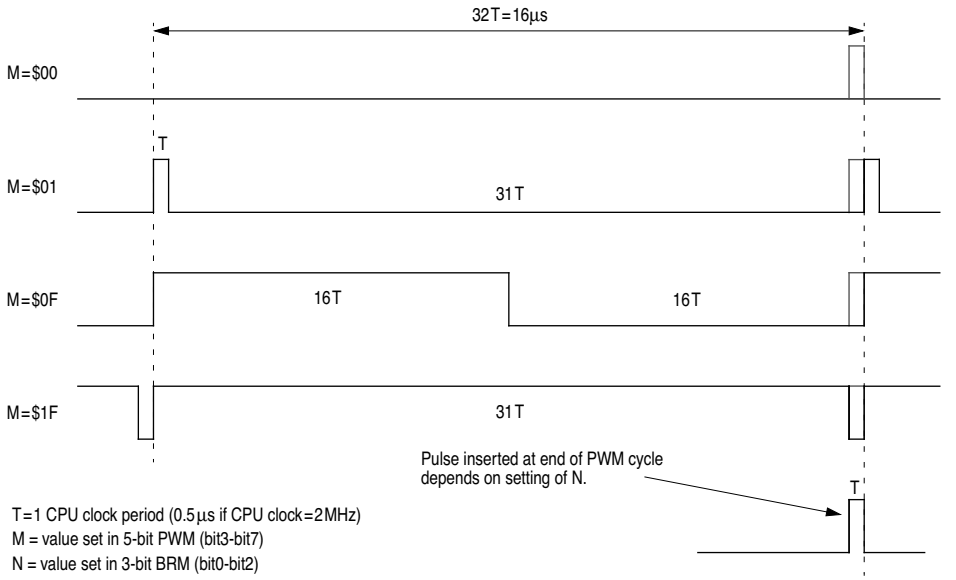
6.2 General Operation

The value programmed in the 5-bit PWM portion will determine the pulse length of the output. The clock to the 5-bit PWM portion is the E clock and the repetition rate of the output is hence 62.5 KHz at 2 MHz E clock.

The 3-bit BRM will generate a number of narrow pulses which are equally distributed among an 8-PWM-cycle. The number of pulses generated is equal to the number programmed in the 3-bit BRM portion. Example of the waveforms are shown in Figure 6-1.

Combining the 5-bit PWM together with the 3-bit BRM, the average duty cycle at the output will be (M+N/8)/32, where M is the content of the 5-bit PWM portion, and N is the content of the 3-bit BRM portion. Using this mechanism, a true 8-bit resolution PWM is achieved.

The value of each PWM Data Register is continuously compared with the content of an internal counter to determine the state of each PWM channel output pin. Double buffering is not used in this PWM design.



N	PWM cycles where pulses are inserted in a 8-cycle frame	Number of inserted pulses in a 8-cycle frame
xx1	4	1
x1x	2, 6	2
1xx	1, 3, 5, 7	4

Figure 6-1 8-Bit PWM Output Waveforms

7
M-BUS SERIAL INTERFACE

M-Bus (Motorola Bus) is a two-wire, bidirectional serial bus which provides a simple, efficient way for data exchange between devices. It is fully compatible with the I^2C bus standard. This two-wire bus minimizes the interconnection between devices and eliminates the need for address decoders; resulting in less PCB traces and economic hardware structure. This bus is suitable for applications requiring communications in a short distance among a number of devices. The maximum data rate is 100 Kbit/s. The maximum communication length and number of devices that can be connected are limited by a maximum bus capacitance of 400 pF.

The M-Bus system is a true multi-master bus, including arbitration to prevent data collision if two or more masters intend to control the bus simultaneously. It may be used for rapid testing and alignment of end products via external connections to an assembly-line computer.

7.1 M-Bus Interface Features

- Compatible with I^2C bus standard
- Multi-master operation
- 32 software programmable serial clock frequencies
- Software selectable acknowledge bit
- Interrupt driven byte-by-byte data transfer
- Arbitration lost driven interrupt with automatic mode switching from master to slave
- Calling address identification interrupt
- Generate/detect the start, stop and acknowledge signals
- Repeated START signal generation
- Bus busy detection

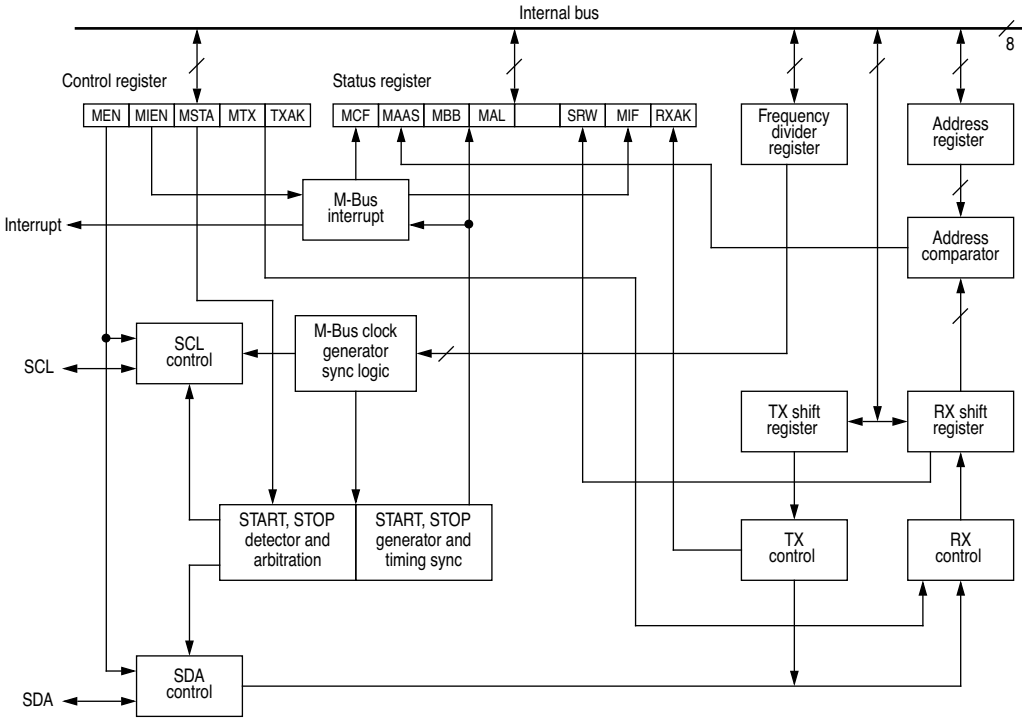


Figure 7-1 M-Bus Interface Block Diagram

7.2 M-Bus Protocol

Normally, a standard communication is composed of four parts,

1) START signal,
2) slave address transmission,
3) data transfer, and
4) STOP signal.

They are described briefly in the following sections and illustrated in Figure 7-2.

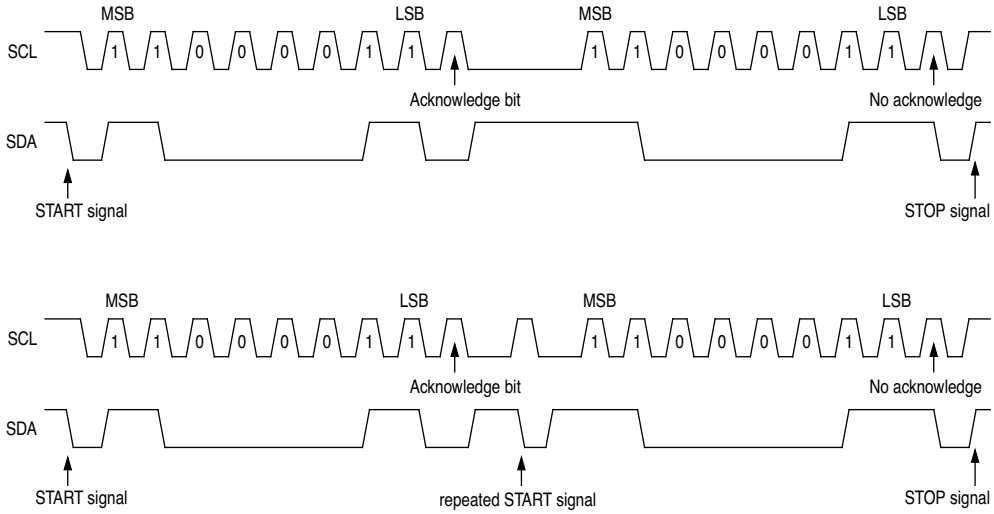


Figure 7-2 M-Bus Transmission Signal Diagram

7.2.1 START Signal

When the bus is free, i.e., no master device is occupying the bus (both SCL and SDA lines are at logic high), a master may initiate communication by sending a START signal. As shown in Figure 7-2, a START signal is defined as a high to low transition of SDA while SCL is high. This signal denotes the beginning of a new data transfer (each data transfer may contain several bytes of data) and wakes up all slaves.

7.2.2 Slave Address Transmission

The first byte of data transfer immediately following the START signal is the slave address transmitted by the master. This is a seven bits long calling address followed by a R/$\overline{W}$ bit. The R/$\overline{W}$ bit dictates the slave of the desired direction of data transfer.

Only the slave with matched address will respond by sending back an acknowledge bit by pulling the SDA low at the 9th clock; see Figure 7-2.

7.2.3 Data Transfer

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

Each data byte is 8 bits long. Data can be changed only when SCL is low and must be held stable when SCL is high as shown in Figure 7-2. One clock pulse is for one bit of data transfer, MSB is transferred first. Each data byte has to be followed by an acknowledge bit. Hence, one complete data byte transfer requires 9 clock pulses.

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

If the master receiver does not acknowledge the slave transmitter after one byte transmission, it means an “end of data” to the slave. The slave shall release the SDA line for the master to generate STOP or START signal.

7.2.4 Repeated START Signal

As shown in Figure 7-2, a repeated START signal is to generate a START signal without first generating a STOP signal to terminate the communication. This is used by the master to communicate with another slave or with the same slave in a different mode (transmit/receive mode) without releasing the bus.

7.2.5 STOP Signal

The master can terminate the communication by generating a STOP signal to free the bus. However, the master may generate a START signal followed by a calling command without generating a STOP signal first. This is called repeat START. A STOP signal is defined as a low to high transition of SDA while SCL is at a logical high; see Figure 7-2.

7.2.6 Arbitration Procedure

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

7.2.7 Clock Synchronization

Since wire-AND logic is performed on the SCL line, a high to low transition on SCL line will affect the devices connected to the bus. The devices start counting their low period and once a device's clock has gone low, it will hold the SCL line low until the clock high state is reached. However, the change of low to high in this device clock may not change the state of the SCL line, if another device clock is still in its low period. Therefore synchronized clock SCL will be held low by the device which releases SCL to a logic high in the last place. Devices with shorter low periods enter a high wait state during this time (see Figure 7-3). When all devices concerned have counted off their low period, the synchronized clock SCL line will be released and go high. All of them will start counting their high periods. The first device to complete its high period will again pull the SCL line low.

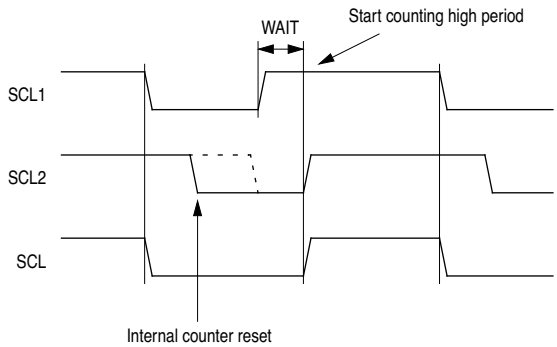


Figure 7-3 Clock Synchronization

7.2.8 Handshaking

The clock synchronization mechanism can be used as a handshake in data transfer. Slave device may hold the SCL low after completion of one byte transfer (9 bits). In such case, it will halt the bus clock and force the master clock in a wait state until the slave releases the SCL line.

7.3 M-Bus Registers

There are five registers used in the M-Bus interface, these are discussed in the following paragraphs.

7.3.1 M-Bus Address Register (MADR)

Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
$0017	MAD7	MAD6	MAD5	MAD4	MAD3	MAD2	MAD1		0000 0000

MAD1-MAD7 are the slave address bits of the M-Bus module.

7.3.2 M-Bus Frequency Register (MFDR)

Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
$0018				FD4	FD3	FD2	FD1	FD0	0000 0000

FD0-FD4 are used for clock rate selection. The serial bit clock frequency is equal to the CPU clock divided by the divider shown in Table 7-1.

7

Table 7-1 M-Bus Prescaler

FD4	FD3	FD2	FD1	FD0	DIVIDER	FD4	FD3	FD2	FD1	FD0	DIVIDER
0	0	0	0	0	22	1	0	0	0	0	352
0	0	0	0	1	24	1	0	0	0	1	384
0	0	0	1	0	28	1	0	0	1	0	448
0	0	0	1	1	34	1	0	0	1	1	544
0	0	1	0	0	44	1	0	1	0	0	704
0	0	1	0	1	48	1	0	1	0	1	768
0	0	1	1	0	56	1	0	1	1	0	896
0	0	1	1	1	68	1	0	1	1	1	1088
0	1	0	0	0	88	1	1	0	0	0	1408
0	1	0	0	1	96	1	1	0	0	1	1536
0	1	0	1	0	112	1	1	0	1	0	1792
0	1	0	1	1	136	1	1	0	1	1	2176
0	1	1	0	0	176	1	1	1	0	0	2816
0	1	1	0	1	192	1	1	1	0	1	3072
0	1	1	1	0	224	1	1	1	1	0	3584
0	1	1	1	1	272	1	1	1	1	1	4352

For a 4MHz external crystal operation (2MHz internal operating frequency), the serial bit clock frequency of M-Bus ranges from 460Hz to 90,909Hz.

7.3.3 M-Bus Control Register (MCR)

Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
$0019	MEN	MIEN	MSTA	MTX	TXAK				0000 0000

Register bit definitions:

MEN - M-Bus Enable

1 (set) – M-Bus interface system enabled.

0 (clear) – M-Bus interface system disabled.

MIEN - M-Bus Interrupt Enable

1 (set) – M-Bus interrupt enabled.

0 (clear) – M-Bus interrupt disabled.

This bit enables the MIF (in MSR) for M-Bus interrupts.

MSTA - Master/Slave Select

1 (set) – M-Bus is set for master mode operation.

0 (clear) – M-Bus is set for slave mode operation.

Upon reset, this bit is cleared. When this bit is changed from 0 to 1, a START signal is generated on the bus, and the master mode is selected. When this bit is changed from 1 to 0, a STOP signal is generated and the operation mode changes from master to slave. In master mode, a bit clear immediately followed by a bit set of this bit generates a repeated START signal without generating a STOP signal.

MTX - Transmit/Receive Mode Select

1 (set) – M-Bus is set for transmit mode.

0 (clear) – M-Bus is set for receive mode.

TXAK - Acknowledge Enable

1 (set) – Do not send acknowledge signal.

0 (clear) – Send acknowledge signal at 9th clock bit.

If cleared, an acknowledge signal will be sent out to the bus at the 9th clock bit after receiving one byte of data. If set, no acknowledge signal response. This is an active low control bit.

7.3.4 M-Bus Status Register (MSR)

Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
$001A	MCF	MAAS	MBB	MAL		SRW	MIF	RXAK	1000 0001

The MIF and MAL bits are software clearable; while the other bits are read only.

MCF - Data Transfer Complete

1 (set) – A byte transfer has been completed.

0 (clear) – A byte is being transfer.

When MCF is set, the MIF (M-Bus interrupt) bit is also set. An M-Bus interrupt is generated if the MIEN bit is set.

MAAS - Addressed as Slave

1 (set) – Currently addressed as a slave.

0 (clear) – Not currently addressed.

This MAAS bit is set when its own specific address (M-Bus Address register) matches the calling address. When MAAS is set, the MIF (M-Bus interrupt) bit is also set. An interrupt is generated if the MIEN bit is set. Then CPU needs to check the SRW bit and set its MTX bit accordingly. Writing to the M-Bus Control register clears this bit.

MBB - Bus Busy

1 (set) – M-Bus busy.

0 (clear) – M-Bus idle.

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

MAL - Arbitration Lost

1 (set) – Lost arbitration in master mode.

0 (clear) – No arbitration lost.

This arbitration lost flag is set when the M-Bus master loses arbitration during a master transmission mode. When MAL is set, the MIF (M-Bus interrupt) bit is also set. This bit must be cleared by software.

SRW - Slave R/W Select

1 (set) – Read from slave, from calling master

0 (clear) – Write to slave from calling master.

When MAAS is set, the R/W command bit of the calling address sent from the master is latched into this SRW bit. By checking this bit, the CPU can then select slave transmit/receive mode by configuring MTX bit of the M-Bus Control register.

MIF - M-Bus Interrupt

1 (set) – An M-Bus interrupt has occurred.

0 (clear) – An M-Bus interrupt has not occurred.

When this bit is set, an interrupt is generated to the CPU if MIEN is set. This bit is set when one of the following events occurs:

1) Completion of one byte of data transfer. It is set at the falling edge of the 9th clock - MCF set.
2) A match of the calling address with its own specific address in slave mode - MAAS set.
3) A loss of bus arbitration - MAL set.

This bit must be cleared by software in the interrupt routine.

RXAK - Receive Acknowledge

1 (set) – No acknowledgment signal detected.

0 (clear) – Acknowledgment signal detected after 8 bits data transmitted.

If cleared, it indicates an acknowledge signal has been received after the completion of 8 bits data transmission on the bus. If set, no acknowledge signal has been detected at the 9th clock. This is an active low status flag.

7.3.5 M-Bus Data I/O Register (MDR)

Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
$001B	MD7	MD6	MD5	MD4	MD3	MD2	MD1	MD0	uuuu uuuu

In master transmit mode, data written into this register is sent to the bus automatically, with the most significant bit out first. In master receive mode, reading of this register initiates receiving of the next byte data. In slave mode, the same function applies after it has been addressed.

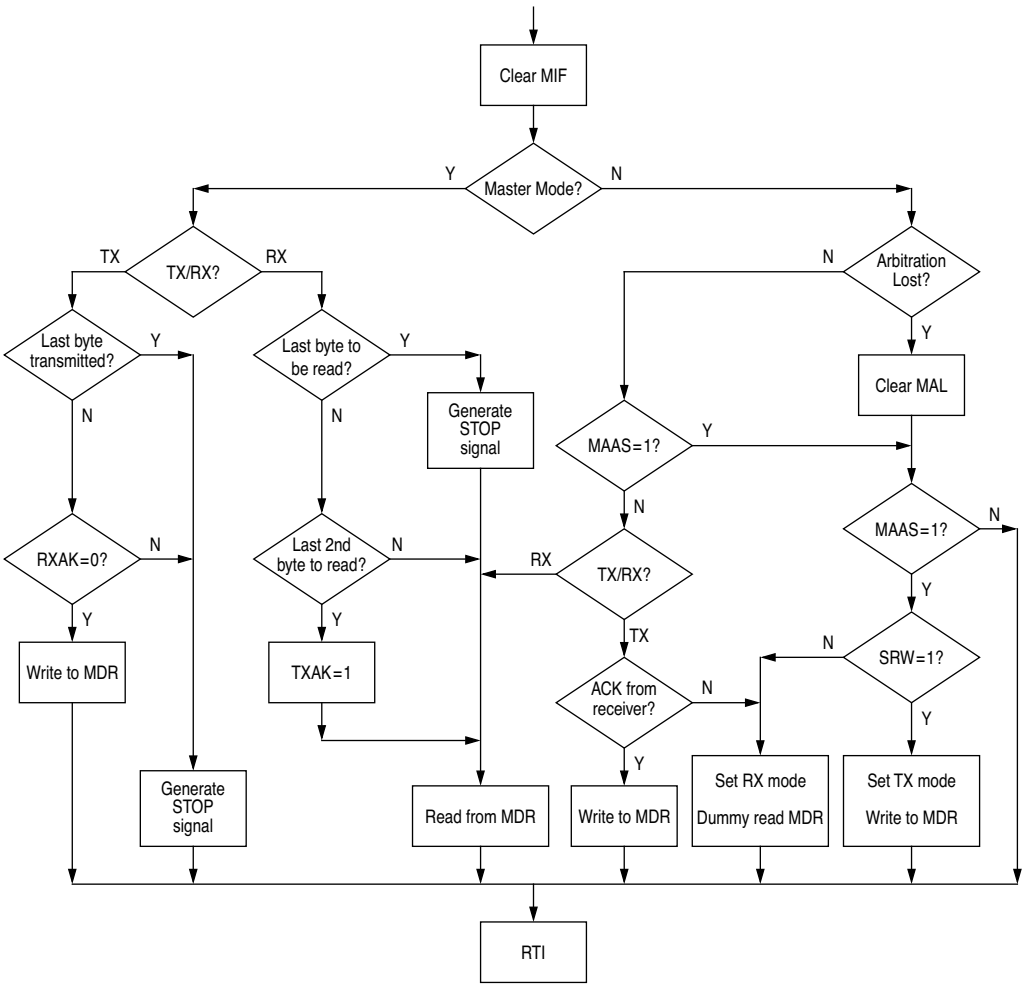


Figure 7-4 Flowchart of M-Bus Interrupt Routine

7.4 Programming Considerations

7.4.1 Initialization

Reset will put the M-Bus Control register to its default status. Before the interface can be used to transfer serial data, the following initialization procedure must be carried out.

1) Update Frequency Divider Register (MFDR) to select an SCL frequency.
2) Update M-Bus Address Register (MADR) to define its own slave address.
3) Set MEN bit of M-Bus Control Register (MCR) to enable the M-Bus interface system.
4) Modify the bits of M-Bus Control Register (MCR) to select Master/Slave mode, Transmit/Receive mode, interrupt enable or not.

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

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

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

7.4.3 Software Responses after Transmission or Reception of a Byte

Upon the completion of the transmission or reception of a data byte, the data transferring bit (MCF) will be set, indicating one byte communication has been finished. The M-Bus interrupt bit (MIF) will also be set to generate an M-Bus interrupt if the interrupt is enabled. Software must clear the

MIF bit in the interrupt routine first. The MCF bit can be cleared by reading the M-Bus Data I/O Register (MDR) in receive mode or writing to the MDR in transmit mode. Software may serve the M-Bus I/O in the main program by monitoring the MIF bit if the interrupt is disabled. The following is an example of a software response by a master in transmit mode in the interrupt routine (see Figure 7-4).

```
ISR          BCLR   1,MSR          ; CLEAR THE MIF FLAG
             BRCLR  5,MCR,SLAVE    ; CHECK THE MSTA FLAG,
                                   ; BRANCH IF SLAVE MODE
             BRCLR  4,MCR,RECEIVE  ; CHECK THE MODE FLAG,
                                   ; BRANCH IF IN RECEIVE MODE
             BRSET  0,MSR,END      ; CHECK ACK FROM RECEIVER
                                   ; IF NO ACK, END OF
                                   ; TRANSMISSION
TRANSMIT     LDA    DATABUF        ; GET THE NEXT BYTE OF DATA
             STA    MDR            ; TRANSMIT THE DATA
```

7.4.4 Generation of the STOP Signal

A data transfer ends with a STOP signal generated by the master device. A master in transmit mode can simply generate a STOP signal after all the data have been transmitted. The following is an example showing how a STOP condition is generated by a master in transmit mode.

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

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

```
MASR   DEC    RXCNT
       BEQ    ENMASR        ; LAST BYTE TO BE READ
       LDA    RXCNT
       DECA                 ; CHECK LAST 2ND BYTE TO BE READ
       BNE    NXMAR         ; NOT LAST ONE OR LAST SECOND
```

```
LAMAR  BSET   3,MCR          ; LAST SECOND, DISABLE ACK
                             ; TRANSMITTING
       BRA    NXMAR
ENMASR BCLR   5,MCR          ; LAST ONE, GENERATE 'STOP'
                             ; SIGNAL
NXMAR  LDA    MDR            ; READ DATA AND STORE
       STA    RXBUF
       RTI
```

7.4.5 Generation of a Repeated START Signal

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

```
RESTART      BCLR   5,MCR          ; ANOTHER START (RESTART) IS
             BSET   5,MCR          ; GENERATED BY THESE TWO
                                   ; CONSECUTIVE INSTRUCTIONS
             LDA    #CALLING       ; GET THE CALLING ADDRESS
             STA    MDR            ; TRANSMIT THE CALLING
                                   ; ADDRESS
```


7.4.6 Slave Mode

In the slave service routine, the master addressed as slave bit (MAAS) should be tested to check if a calling of its own address has been received (Figure 7-4). If MAAS is set, software should set the transmit/receive mode select bit (MTX bit of MCR) according to the R/$\overline{W}$ command bit (SRW). Writing to the MCR clears the MAAS automatically. A data transfer may then be initiated by writing to MDR or a dummy read from MDR.

In the slave transmit routine, the received acknowledge bit (RXAK) must be tested before transmitting the next byte of data. RXAK, if set indicates the end of data signal from the master receiver, the slave transmitter must then switch from transmit mode to receive mode by software and a dummy read must follow to release the SCL line so that the master can generate a STOP signal.

7.4.7 Arbitration Lost

If more than one master want to acquire the bus simultaneously, only one master can win and the others will lose arbitration. The losing device immediately switches to slave receive mode by M-Bus hardware. Its data output to the SDA line is stopped, but internal transmit clock still runs until the end of the data byte transmission. An interrupt occurs when this dummy byte transmission

is accomplished with MAL=1 and MSTA=0. If one master attempts to start transmission while the bus is being controlled by another master, the transmission will be inhibited; the MSTA bit will be changed from 1 to 0 without generating STOP condition; an interrupt will be generated and the MAL bit set to indicate that the attempt to acquire the bus has failed. Considering these cases, the slave service routine should test the MAL bit first, and software should clear the MAL bit if it is set.

8
SYNC SIGNAL PROCESSOR

The functions of the SSP include polarity correction, sync separation, sync pulse reshaper, sync pulse detectors, horizontal line counter, vertical frequency counter, and free running signals generator. In addition, interrupt can be generated for each vertical frame at a user specified horizontal line number.

The processor accepts either composite or separate sync inputs.

For separate sync inputs, the HTTL and VTTL outputs are identical to the incoming horizontal sync with negative sync polarity. As for composite sync input, reassembled horizontal sync pulses can be inserted during the vertical sync period. The VTTL output is triggered by the leading edge of the incoming vertical sync pulse, and the sync pulse will be widened by 9.5µs.

Both HSYNC and VSYNC inputs have internal filter to improve noise immunity. Any pulse that is shorter than an internal bus clock period, will be regarded as a glitch, and will be ignored.

Note: All quoted timings in this section are based on the assumption that the internal bus frequency is 2MHz, i.e. t_{CYC}=0.5µs.

8.1 Functional Blocks

The architecture of the Sync Signal Processor is shown in Figure 8-1. Each of the functional blocks are described in the following paragraphs.

8.1.1 Polarity Correction

The polarity correction block of the sync signal processor accepts the input sync signals (HSYNC/VSYNC) and converts them to negative polarity signals, regardless of the polarity of the inputs. The following describes the methodologies used in polarity correction.

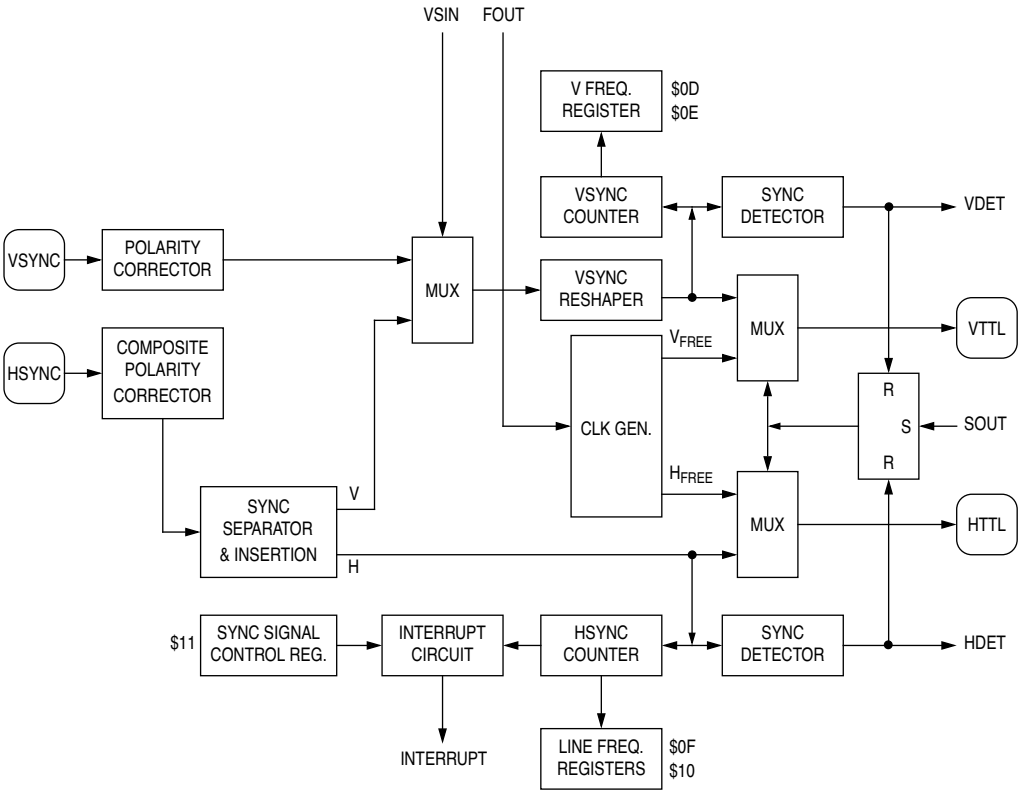


Figure 8-1 Sync Signal Processor Block Diagram

8.1.1.1 Separate Vertical Sync Input

To test the polarity of the input sync signal, the duration of the low pulse is examined. If the low period is longer than a specific value (512µs or $1024t_{CYC}$), as in the case of positive polarity input sync, the input sync will be inverted before output. For negative polarity input sync signal, it is anticipated that the duration of the low pulse would be shorter than the specific value, and the input sync signal passes through to the output without inversion.

This polarity correction is a continuous process, and the error margin is equal to the maximum permissible sync pulse width specified (512µs or $1024t_{CYC}$). At power-up or system reset, negative polarity at input is assumed.

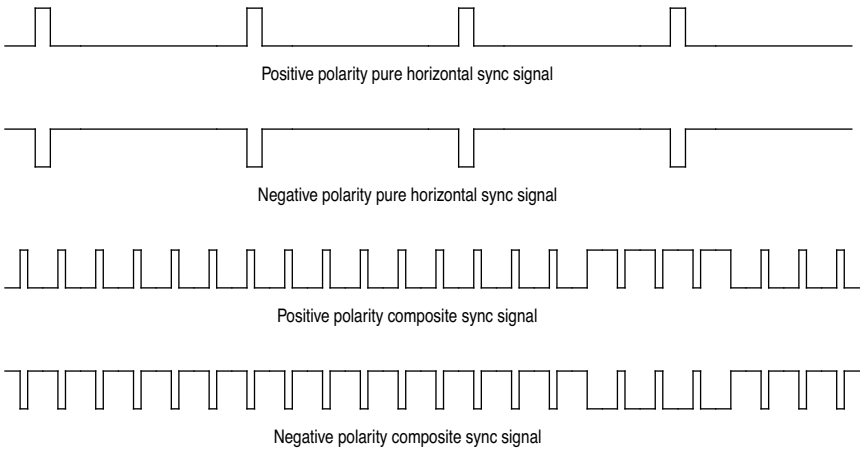


Figure 8-2 Sync Signal Polarity Correction

8.1.1.2 Separate Horizontal Or Composite Sync Input

Since the input at HSYNC can be either a pure horizontal sync signal or a composite sync signal, different methodologies are used in polarity correction.

Unlike the polarity correction for VSYNC, both the high pulse and low pulse of the sync signal at HSYNC are examined. If the pulse, either active high or low, is longer than a certain period (8µs or 16 t_{CYC}), it will be regarded as a long pulse. If there are 8 consecutive low long pulses, the input sync signal will be confirmed as a positive polarity sync signal, and will be inverted. If there are 8 consecutive high long pulses, it will be confirmed as a negative polarity sync signal.

The operation of this module is also continuous, and the error margin is equal to the period of the pre-set number (default is 8) of horizontal sync pulses. At power-up or system reset, negative polarity at input is assumed.

8.1.2 Sync Detection

The sync detector determines whether the incoming sync signal is active. Both sync high and low pulse widths must be within the specific values to be regarded as active. HDET and VDET flags will be set if the HSYNC and VSYNC signals are active, respectively.

8.1.3 Free-running Pseudo Sync Signal Generator

If either HSYNC or VSYNC is absent, a free-running sync signal generator will be enabled. It generates a pseudo vertical sync at 63.5Hz (1/(tcyc x 31488)) and a pseudo horizontal sync at either 48.8KHz (1/(t_{CYC} x 41)) or 62.5KHz (1/(t_{CYC} x 32)), depending on the status of FOUT. This set of free running sync signals replaces the inactive sync signals at the inputs and will be fed to the VTTL and HTTL pins if the pins are selected for VTTL and HTTL function.

8.1.4 Sync Separation

Figure 8-3 is a block diagram of the Sync Separator which includes the duration counters for the high and low pulses, a counter for the number of valid horizontal sync pulses, a register to hold the number of horizontal lines per frame, a logic block for horizontal and vertical sync pulse separation, a comparator, and a sync pulse insertion circuit.

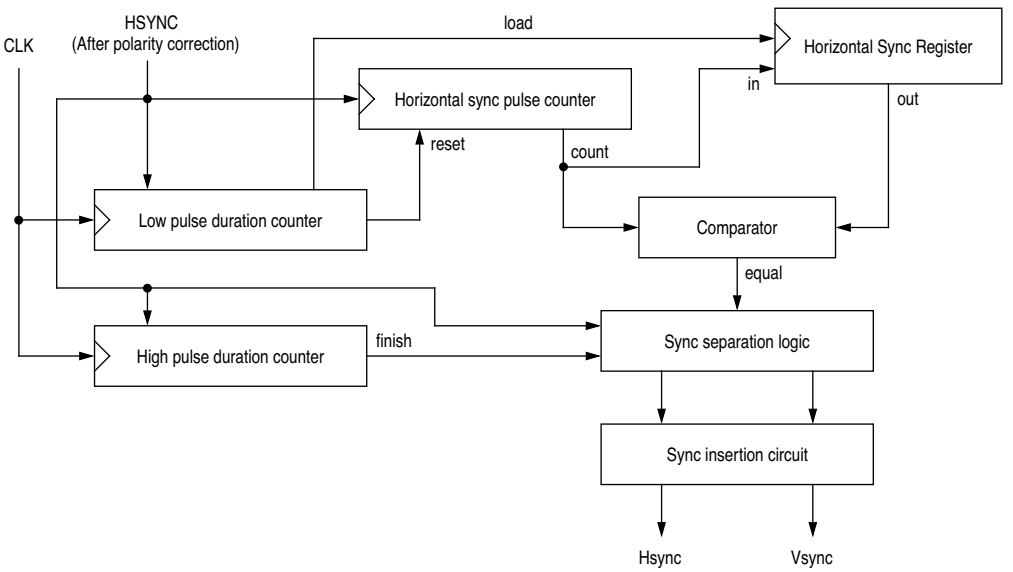


Figure 8-3 Sync Separator

The Low pulse duration counter examines the low pulse width of the incoming composite sync signal. If it is within the horizontal sync pulse limit (8µs or 16 t_{CYC}), a horizontal sync pulse is detected and the horizontal sync pulse counter is advanced. If the low pulse is wider than the limit,

a vertical sync pulse is detected, and the content of the Horizontal sync pulse counter is loaded into the Horizontal Sync Register before the Low Pulse Duration Counter is reset.

Comparator compares the values of the Horizontal Sync Pulse Counter and Horizontal Sync Register, and gives the equal signal to the Sync Separation Logic.

High Pulse Duration Counter examines the high pulse width of the incoming composite sync signal. If it is longer than a specific value (8µs or 16 t_{CYC}), the vertical sync pulse has finished and finish signal will be given to the Sync Separation Logic.

Sync Separation Logic passes the composite sync signal to the Hsync output, until there is an "equal" signal from the comparator. The Hsync output will then output a reassembled waveform by the Sync Insertion Circuit to emulate the HSYNC pulses, and the Vsync output is set to low at the coming falling edge of the composite signal. After the finish signal has been sensed, the Vsync output is fixed to high, and the Hsync output follows the composite sync input again.

8.1.5 Vertical Sync Pulse Reshaper

For separate sync inputs, the vertical sync pulse width VTTL equals to the incoming vertical sync input. For composite sync input, the Sync Pulse Reshaper widens the VTTL pulse width by 9.5µs.

8.1.6 Sync Signal Counters

There are two counters (horizontal line counter and vertical frequency counter) to count the number of horizontal sync pulses and the number of system clock cycles between two vertical sync pulses. These two data can be read by the CPU to check the signal frequencies and can be used to determine the video mode. Figure 8-4 shows a more detailed block diagram of these counters. The 13-bit vertical frequency register encompasses vertical frequency range from approximately 15Hz to 125KHz. Figure 8-5 shows the vertical frequency counter timing. It indicates that there will be ±1 count error on the reading from the register for the same vertical frequency.

8.2 VSYNC Interrupt

The Sync Signal Processor will generate interrupts to the CPU if the VSYNC Interrupt Enable (VSIE) bit is set, and the I-bit in the Condition Code Register (CCR) is cleared. The interrupt will occur at each leading edge of VSYNC.

The interrupt vector address is at $3FF8-$3FF9, and the interrupt latch is cleared automatically by fetching of the interrupt vectors.

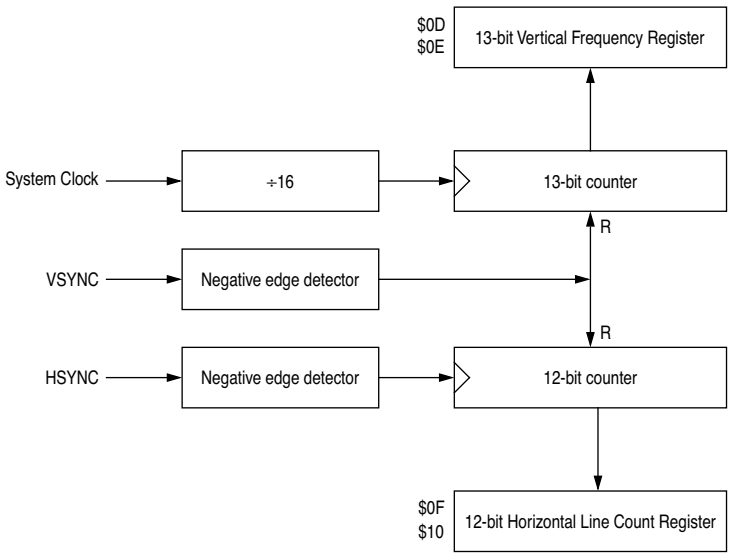
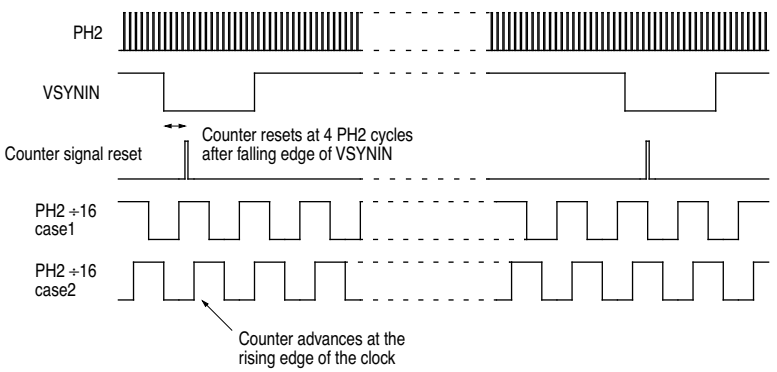


Figure 8-4 Sync Signal Counters Block Diagram



1. The value of the counter will be loaded into the register before it is reset.
2. The Vertical Frequency Counter is clocked by a PH2 ÷16 clock.
3. Because of the asynchronous nature between PH2 and VSYNIN, the register will have one more count in case 2 than in case 1.

Figure 8-5 Vertical Frequency Counter Timing

8.3 Registers

There are seven registers associated with the Sync Signal Processor, these are described below.

8.3.1 Sync Signal Control & Status Register (SSCSR)

Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
$000C	VPOL	HPOL	VDET	HDET	SOUT	INSRTB	FOUT	VSIN	0000 0000

VPOL - Vertical Sync Input Polarity

1 (set) – VSYNC input is positive polarity.

0 (clear) – VSYNC input is negative polarity.

Vertical Sync Input Polarity flag indicates the polarity of the incoming signal at the VSYNC input.

HPOL - Horizontal Sync Input Polarity

1 (set) – HSYNC input is positive polarity.

0 (clear) – HSYNC input is negative polarity.

Horizontal Sync Input Polarity flag indicates the polarity of the incoming signal at the HSYNC input.

VDET - Vertical Sync Signal Detect

1 (set) – An active vertical sync is detected at VSYNC input.

0 (clear) – No vertical sync signal at VSYNC input; use internal generated Vsync for VTTL.

Vertical Sync Signal Detect flag, if set, indicates an active input vertical sync signal has been detected. If cleared, it indicates there is no active signal, and the VTTL will output the internally generated Vsync signal. An active vertical sync signal is defined as:

VDET = (VSYNC pulse width < 480µs or $960t_{CYC}$)·(VSYNC period < 65.5ms or $131x10^3t_{CYC}$)

HDET - Horizontal Sync Signal Detect

1 (set) – An active horizontal sync is detected at HSYNC input.

0 (clear) – No horizontal sync signal at HSYNC input; use internal generated Hsync for HTTL.

Horizontal Sync Signal Detect flag, if set, indicates an active input horizontal sync signal has been detected. If cleared, it indicates there is no active signal, and the HTTL will output the internally generated Hsync signal. An active horizontal sync signal is defined as:

HDET=(HSYNC pulse width < 8µs or $16t_{CYC}$)·(9µs or $18t_{CYC}$ < HSYNC period < 128µs or $256t_{CYC}$)

SOUT - Sync Output Select

1 (set) – Use processed VSYNC and HSYNC inputs for VTTL and HTTL.

0 (clear) – Use internally generated sync signals for VTTL and HTTL.

When cleared, the outputs to VTTL and HTTL are the internally generated signals. When set, the outputs are the processed input signals. This bit can only be set if both VDET and HDET are logic 1's, and will be cleared automatically if VDET or HDET is not logic "1". Reset clears this bit.

INSRTB - Hsync Insertion Bit

1 (set) – No inserted pulses. Hsync remains high state during the vertical sync pulse.

0 (clear) – For composite sync inputs, emulated sync pulses will be inserted into the Hsync signal during the vertical sync pulse.

For separate sync inputs, when this Hsync Insertion bit is cleared, sync pulses will continue to be the Hsync signal during the Vertical Sync Pulse. For composite sync input, when this Hsync Insertion bit is cleared, emulated sync pulses will be inserted into the Hsync signal during the Vertical Sync Pulse. In both cases, when this bit is set, there will be no inserted pulses, and the Hsync signal will be high during the Vertical Sync Pulse. Reset clears this bit.

FOUT - Internal Hsync Frequency Select

1 (set) – 63.5Hz and 62.5KHz for VTTL and HTTL outputs respectively if internally generated sync signals are selected.

0 (clear) – 63.5Hz and 48.8KHz for VTTL and HTTL outputs respectively if internally generated sync signals are selected.

This bit selects the frequency of the free running Hsync signal to HTTL pin if SOUT bit is cleared. When FOUT is set, 63.5Hz and 62.5KHz signals are output to VTTL and HTTL, respectively. When FOUT is cleared, 63.5Hz and 48.8KHz signals are output instead. Reset clears this bits.

VSIN - Vsync Input Source

This bit selects the source of the input sync signals. Reset clears this bits.

1 (set) – Separated sync signals through VSYNC and HSYNC inputs.

0 (clear) – Composite sync signal through HSYNC input

8.3.2 Vertical Frequency Registers (VFRS)

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
VFHR	$000D				VF12	VF11	VF10	VF9	VF8	0000 0000
VFLR	$000E	VF7	VF6	VF5	VF4	VF3	VF2	VF1	VF0	0000 0000

This 13-bit read only register pair contains information of the vertical frame frequency. An internal counter counts the number of internal clocks between two VSYNC pulses. The counted value will then be transferred to this register. The data corresponds to the period of one vertical frame. This register can be read to determine if the frame frequency is valid, and to determine the video mode. However, the data is not valid if VDET bit is cleared.

The frame frequency is calculated by 1/(VFR±1 x 8µs) or 1/(VFR±1 x 16t_{CYC}).

The table below shows examples for the Vertical Frequency Register, all VFR numbers are in hexadecimal**.**

Table 8-1 Vertical Frame Frequencies

VFR	Min. Freq.	Max. Freq.
$03C0	130.07	130.34
$03C1	129.94	130.21
$03C2	129.80	130.07
$04E2	99.92	100.08
$04E3	99.84	100.00
$04E4	99.76	99.92
$06F9	69.99	70.07
$06FA	69.95	70.03
$06FB	69.91	69.99

VFR	Min. Freq.	Max. Freq.
$0823	59.98	60.04
$0824	59.95	60.01
$0825	59.92	59.98
$09C4	49.98	50.02
$09C5	49.96	50.00
$09C6	49.94	49.98
$1FFD	15.262	15.266
$1FFE	15.260	15.264
$1FFF	15.258	15.262

8.3.3 Line Frequency Registers (LFRs)

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
LFHR	$000F	HOVER				LF11	LF10	LF9	LF8	0000 0000
LFLR	$0010	LF7	LF6	LF5	LF4	LF3	LF2	LF1	LF0	0000 0000

This 12-bit read only register pair contains the number of horizontal lines in each vertical frame. An internal line counter counts the horizontal sync pulses between two vertical sync pulses. The counted value will be transferred to this register pair. HOVER bit will be set if the incoming horizontal sync pulses between two vertical sync pulses are more than 4096 or there is no vertical

sync input. The data can be read to determine if the line frequency is valid and to determine the video mode. However, the data is not valid if HDET or VDET bit is cleared or HOVER bit is set. User has to determine whether the incoming signal is separate sync or composite sync. If composite sync signal is input, the actual number of horizontal lines is the value in LFR plus one; because the internal line counter that counts the horizontal sync pulses is rising-edge triggering. If the incoming signal is a composite signal, one horizontal line counting is missed.

8.3.4 Sync Signal Control Register (SSCR)

Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
$0011	VSIE								0000 0000

This is a read/write register. Interrupt will be generated at the leading edge of VSYNC if the VSIE bit is set, I bit in CCR is cleared. The VSYNC interrupt vectors are at $3FF8 and $3FF9, and the interrupt latch is cleared after the interrupt vectors have been fetched.

VSIE - Vsync Interrupt Enable

This bit enables and disables the Vsync interrupt.

1 (set) – Vsync interrupt enabled.

0 (clear) – Vsync interrupt disabled.

8.3.5 Horizontal Sync Period Width Register (HPWR)

Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
$001E	HPWR7	HPWR6	HPWR5	HPWR4	HPWR3	HPWR2	HPWR1	HPWR0	0000 0000

This 8-bit read only register contains the period of incoming horizontal sync signal. It is sampled by t_{CYC} so the horizontal period is equal to HPWR x 0.5µs if t_{CYC} is at 2MHz. As the incoming horizontal sync signal is asynchronous to the system clock, the SSP is designed so that the maximum counting error of HPWR is –2. User should use the LFR to calculate the HSYNC frequency if very accurate frequency detection is needed. If HPWR overflows, the HDET in SSCSR will be cleared. Therefore the minimum valid HSYNC is $256t_{CYC}$, i.e. 7.8125KHz if t_{CYC} equals to 2MHz.

Note: It is not guaranteed that the HPWR counting is correct for the first HSYNC period after the trailing edge of VSYNC.

8.4 System Operation

The incoming signals can be either separate HSYNC and VSYNC or composite sync through HSYNC input. Polarity correction is performed before the sync signals go any further into the system. The sync pulse detection block continuously monitors the signals to see if the signals are active. If the signals are not active, the circuit switches to output the internally generated sync signals. This will protect the circuits behind from being damaged by inactive signals.

A typical monitor system operation is summarized in Figure 8-6.

Note: User is required to check the HDET and VDET at VSIN=0 first. If either or both are not detected, user then set VSIN=1 to check HDET and VDET. It is because if the incoming signal is a valid composite signal, HDET and VDET are both read 1 even VSIN=1.

Note: Each time if VDET is not detected when VSIN=1, user needs to clear VSIN to check VDET. If VDET is still not detected, user then set VSIN to check them again to decide what mode it is.

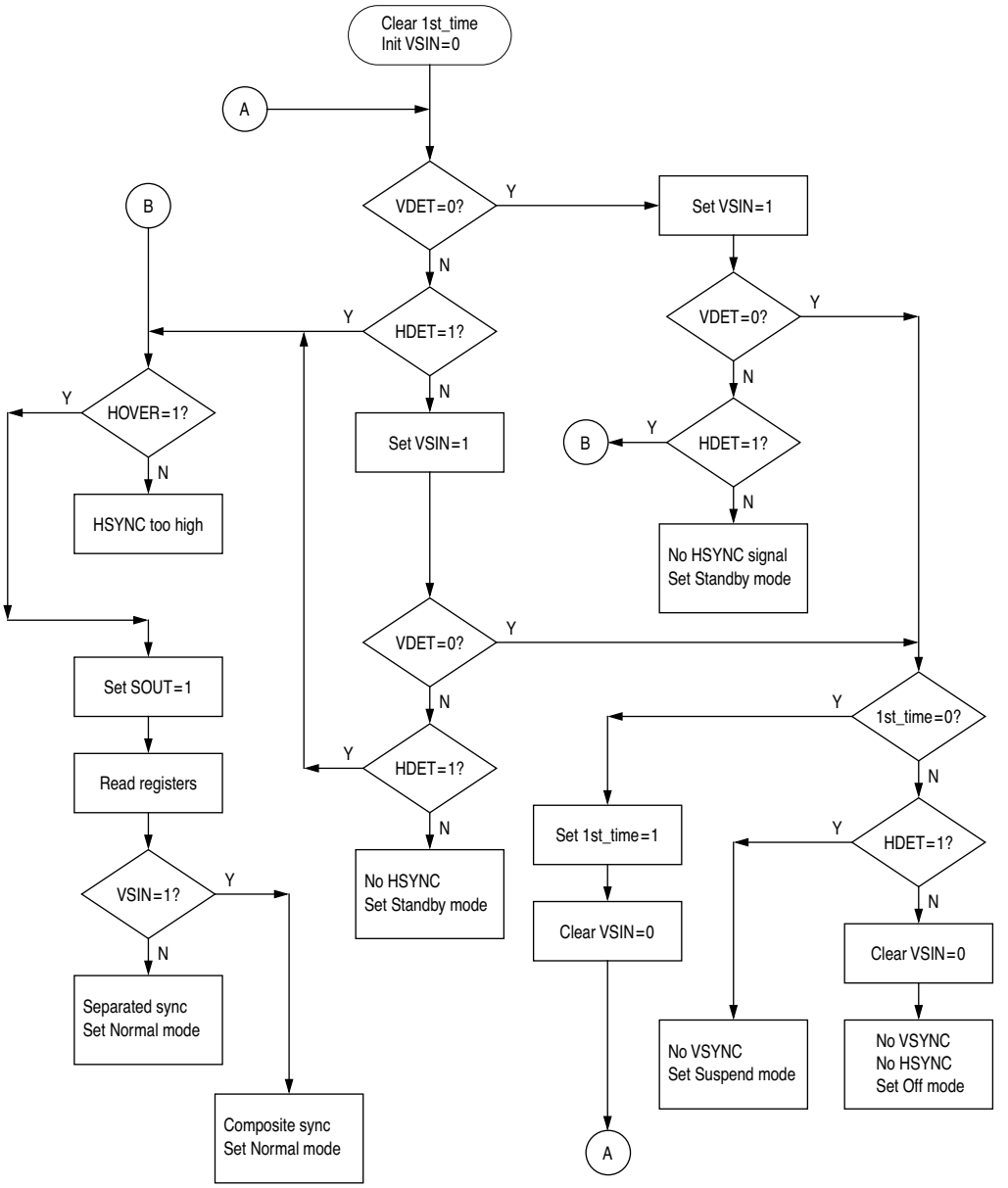


Figure 8-6 Typical Monitor System Operation

9
CPU CORE AND INSTRUCTION SET

This section provides a description of the CPU core registers, the instruction set and the addressing modes of the MC68HC05BD3.

9.1 Registers

The MCU contains five registers, as shown in the programming model of Figure 9-1. The interrupt stacking order is shown in Figure 9-2.

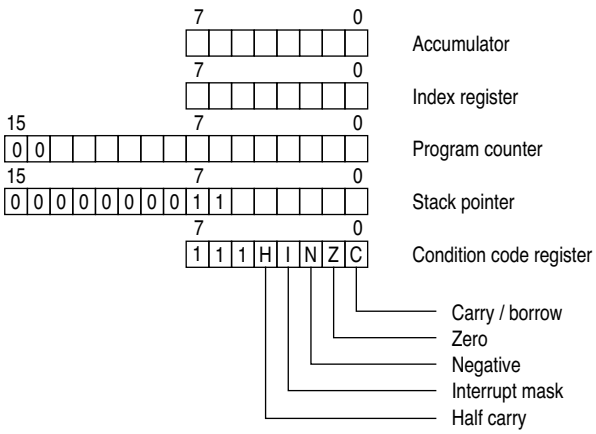


Figure 9-1 Programming model

9.1.1 Accumulator (A)

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

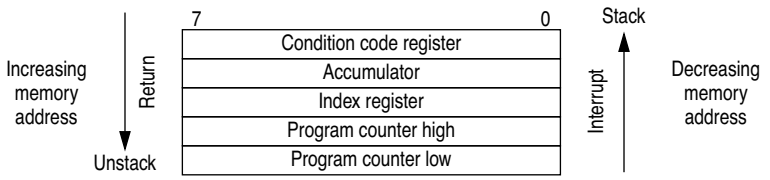


Figure 9-2 Stacking order

9.1.2 Index register (X)

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

9.1.3 Program counter (PC)

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

9.1.4 Stack pointer (SP)

The stack pointer is a 16-bit register, which contains the address of the next free location on the stack. During an MCU 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 $00C0 to $00FF. Subroutines and interrupts may use up to 64 (decimal) locations. If 64 locations are exceeded, the stack pointer wraps around and overwrites the previously stored information. A subroutine call occupies two locations on the stack; an interrupt uses five locations.

9.1.5 Condition code register (CCR)

The CCR 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.

Half carry (H)

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

Interrupt (I)

When this bit is set, all maskable interrupts are masked. If an interrupt occurs while this bit is set, the interrupt is latched and remains pending until the interrupt bit is cleared.

Negative (N)

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

Zero (Z)

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

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.

9.2 Instruction set

The MCU has a set of 62 basic instructions. They can be grouped into five different types as follows:

- Register/memory
- Read/modify/write
- Branch
- Bit manipulation
- Control

The following paragraphs briefly explain each type. All the instructions within a given type are presented in individual tables.

This MCU uses all the instructions available in the M146805 CMOS family plus one more: the unsigned multiply (MUL) instruction. This instruction allows unsigned multiplication of the contents of the accumulator (A) and the index register (X). The high-order product is then stored in the index register and the low-order product is stored in the accumulator. A detailed definition of the MUL instruction is shown in Table 9-1.

9.2.1 Register/memory Instructions

Most of these instructions use two operands. The first operand is either the accumulator or the index register. The second operand is obtained from memory using one of the addressing modes. The jump unconditional (JMP) and jump to subroutine (JSR) instructions have no register operand. Refer to Table 9-2 for a complete list of register/memory instructions.

9.2.2 Branch instructions

These instructions cause the program to branch if a particular condition is met; otherwise, no operation is performed. Branch instructions are two-byte instructions. Refer to Table 9-3.

9.2.3 Bit manipulation instructions

The MCU can set or clear any writable bit that resides in the first 256 bytes of the memory space (page 0). All port data and data direction registers, timer and serial interface registers, control/status registers and a portion of the on-chip RAM reside in page 0. An additional feature allows the software to test and branch on the state of any bit within these locations. The bit set, bit clear, bit test and branch functions are all implemented with single instructions. For the test and branch instructions, the value of the bit tested is also placed in the carry bit of the condition code register. Refer to Table 9-4.

9.2.4 Read/modify/write instructions

These instructions read a memory location or a register, modify or test its contents, and write the modified value back to memory or to the register. The test for negative or zero (TST) instruction is an exception to this sequence of reading, modifying and writing, since it does not modify the value. Refer to Table 9-5 for a complete list of read/modify/write instructions.

9.2.5 Control instructions

These instructions are register reference instructions and are used to control processor operation during program execution. Refer to Table 9-6 for a complete list of control instructions.

9.2.6 Tables

Tables for all the instruction types listed above follow. In addition there is a complete alphabetical listing of all the instructions (see Table 9-7), and an opcode map for the instruction set of the M68HC05 MCU family (see Table 9-8).

Table 9-1 MUL instruction

Operation	**X:A ← X*A**			
Description	Multiplies the eight bits in the index register by the eight bits in the accumulator and places the 16-bit result in the concatenated accumulator and index register.			
Condition codes	H : Cleared I : Not affected N : Not affected Z : Not affected C : Cleared			
Source	MUL			
Form	Addressing mode	Cycles	Bytes	Opcode
	Inherent	11	1	$42

Table 9-2 Register/memory instructions

		Addressing modes																	
		Immediate			**Direct**			**Extended**			**Indexed (no offset)**			**Indexed (8-bit offset)**			**Indexed (16-bit offset)**		
Function	**Mnemonic**	**Opcode**	**# Bytes**	**# Cycles**	**Opcode**	**# Bytes**	**# Cycles**	**Opcode**	**# Bytes**	**# Cycles**	**Opcode**	**# Bytes**	**# Cycles**	**Opcode**	**# Bytes**	**# Cycles**	**Opcode**	**# Bytes**	**# Cycles**
Load A from memory	LDA	A6	2	2	B6	2	3	C6	3	4	F6	1	3	E6	2	4	D6	3	5
Load X from memory	LDX	AE	2	2	BE	2	3	CE	3	4	FE	1	3	EE	2	4	DE	3	5
Store A in memory	STA				B7	2	4	C7	3	5	F7	1	4	E7	2	5	D7	3	6
Store X in memory	STX				BF	2	4	CF	3	5	FF	1	4	EF	2	5	DF	3	6
Add memory to A	ADD	AB	2	2	BB	2	3	CB	3	4	FB	1	3	EB	2	4	DB	3	5
Add memory and carry to A	ADC	A9	2	2	B9	2	3	C9	3	4	F9	1	3	E9	2	4	D9	3	5
Subtract memory	SUB	A0	2	2	B0	2	3	C0	3	4	F0	1	3	E0	2	4	D0	3	5
Subtract memory from A with borrow	SBC	A2	2	2	B2	2	3	C2	3	4	F2	1	3	E2	2	4	D2	3	5
AND memory with A	AND	A4	2	2	B4	2	3	C4	3	4	F4	1	3	E4	2	4	D4	3	5
OR memory with A	ORA	AA	2	2	BA	2	3	CA	3	4	FA	1	3	EA	2	4	DA	3	5
Exclusive OR memory with A	EOR	A8	2	2	B8	2	3	C8	3	4	F8	1	3	E8	2	4	D8	3	5
Arithmetic compare A with memory	CMP	A1	2	2	B1	2	3	C1	3	4	F1	1	3	E1	2	4	D1	3	5
Arithmetic compare X with memory	CPX	A3	2	2	B3	2	3	C3	3	4	F3	1	3	E3	2	4	D3	3	5
Bit test memory with A (logical compare)	BIT	A5	2	2	B5	2	3	C5	3	4	F5	1	3	E5	2	4	D5	3	5
Jump unconditional	JMP				BC	2	2	CC	3	3	FC	1	2	EC	2	3	DC	3	4
Jump to subroutine	JSR				BD	2	5	CD	3	6	FD	1	5	ED	2	6	DD	3	7

Table 9-3 Branch instructions

		Relative addressing mode		
Function	**Mnemonic**	**Opcode**	**# Bytes**	**# Cycles**
Branch always	BRA	20	2	3
Branch never	BRN	21	2	3
Branch if higher	BHI	22	2	3
Branch if lower or same	BLS	23	2	3
Branch if carry clear	BCC	24	2	3
(Branch if higher or same)	(BHS)	24	2	3
Branch if carry set	BCS	25	2	3
(Branch if lower)	(BLO)	25	2	3
Branch if not equal	BNE	26	2	3
Branch if equal	BEQ	27	2	3
Branch if half carry clear	BHCC	28	2	3
Branch if half carry set	BHCS	29	2	3
Branch if plus	BPL	2A	2	3
Branch if minus	BMI	2B	2	3
Branch if interrupt mask bit is clear	BMC	2C	2	3
Branch if interrupt mask bit is set	BMS	2D	2	3
Branch if interrupt line is low	BIL	2E	2	3
Branch if interrupt line is high	BIH	2F	2	3
Branch to subroutine	BSR	AD	2	6

Table 9-4 Bit manipulation instructions

		Addressing modes					
		Bit set/clear			Bit test and branch		
Function	**Mnemonic**	**Opcode**	**# Bytes**	**# Cycles**	**Opcode**	**# Bytes**	**# Cycles**
Branch if bit n is set	BRSET n (n=0–7)				2•n	3	5
Branch if bit n is clear	BRCLR n (n=0–7)				01+2•n	3	5
Set bit n	BSET n (n=0–7)	10+2•n	2	5			
Clear bit n	BCLR n (n=0–7)	11+2•n	2	5			

Table 9-5 Read/modify/write instructions

		Addressing modes														
		Inherent (A)			Inherent (X)			Direct			Indexed (no offset)			Indexed (8-bit offset)		
Function	Mnemonic	Opcode	# Bytes	# Cycles	Opcode	# Bytes	# Cycles	Opcode	# Bytes	# Cycles	Opcode	# Bytes	# Cycles	Opcode	# Bytes	# Cycles
Increment	INC	4C	1	3	5C	1	3	3C	2	5	7C	1	5	6C	2	6
Decrement	DEC	4A	1	3	5A	1	3	3A	2	5	7A	1	5	6A	2	6
Clear	CLR	4F	1	3	5F	1	3	3F	2	5	7F	1	5	6F	2	6
Complement	COM	43	1	3	53	1	3	33	2	5	73	1	5	63	2	6
Negate (two's complement)	NEG	40	1	3	50	1	3	30	2	5	70	1	5	60	2	6
Rotate left through carry	ROL	49	1	3	59	1	3	39	2	5	79	1	5	69	2	6
Rotate right through carry	ROR	46	1	3	56	1	3	36	2	5	76	1	5	66	2	6
Logical shift left	LSL	48	1	3	58	1	3	38	2	5	78	1	5	68	2	6
Logical shift right	LSR	44	1	3	54	1	3	34	2	5	74	1	5	64	2	6
Arithmetic shift right	ASR	47	1	3	57	1	3	37	2	5	77	1	5	67	2	6
Test for negative or zero	TST	4D	1	3	5D	1	3	3D	2	4	7D	1	4	6D	2	5
Multiply	MUL	42	1	11												

Table 9-6 Control instructions

		Inherent addressing mode		
Function	Mnemonic	Opcode	# Bytes	# Cycles
Transfer A to X	TAX	97	1	2
Transfer X to A	TXA	9F	1	2
Set carry bit	SEC	99	1	2
Clear carry bit	CLC	98	1	2
Set interrupt mask bit	SEI	9B	1	2
Clear interrupt mask bit	CLI	9A	1	2
Software interrupt	SWI	83	1	10
Return from subroutine	RTS	81	1	6
Return from interrupt	RTI	80	1	9
Reset stack pointer	RSP	9C	1	2
No-operation	NOP	9D	1	2
Stop	STOP	8E	1	2
Wait	WAIT	8F	1	2

Table 9-7 Instruction set

Mnemonic	Addressing modes										Condition codes				
	INH	IMM	DIR	EXT	REL	IX	IX1	IX2	BSC	BTB	H	I	N	Z	C
ADC											◊	•	◊	◊	◊
ADD											◊	•	◊	◊	◊
AND											•	•	◊	◊	•
ASL											•	•	◊	◊	◊
ASR											•	•	◊	◊	◊
BCC											•	•	•	•	•
BCLR											•	•	•	•	•
BCS											•	•	•	•	•
BEQ											•	•	•	•	•
BHCC											•	•	•	•	•
BHCS											•	•	•	•	•
BHI											•	•	•	•	•
BHS											•	•	•	•	•
BIH											•	•	•	•	•
BIL											•	•	•	•	•
BIT											•	•	◊	◊	•
BLO											•	•	•	•	•
BLS											•	•	•	•	•
BMC											•	•	•	•	•
BMI											•	•	•	•	•
BMS											•	•	•	•	•
BNE											•	•	•	•	•
BPL											•	•	•	•	•
BRA											•	•	•	•	•
BRN											•	•	•	•	•
BRCLR											•	•	•	•	◊
BRSET											•	•	•	•	◊
BSET											•	•	•	•	•
BSR											•	•	•	•	•
CLC											•	•	•	•	0
CLI											•	0	•	•	•
CLR											•	•	0	1	•
CMP											•	•	◊	◊	◊

Address mode abbreviations

BSC	Bit set/clear	IMM	Immediate
BTB	Bit test & branch	IX	Indexed (no offset)
DIR	Direct	IX1	Indexed, 1 byte offset
EXT	Extended	IX2	Indexed, 2 byte offset
INH	Inherent	REL	Relative

Not implemented

Condition code symbols

H	Half carry (from bit 3)	◊	Tested and set if true, cleared otherwise
I	Interrupt mask	•	Not affected
N	Negate (sign bit)	?	Load CCR from stack
Z	Zero	0	Cleared
C	Carry/borrow	1	Set

Table 9-7 Instruction set (Continued)

Mnemonic	Addressing modes										Condition codes				
	INH	IMM	DIR	EXT	REL	IX	IX1	IX2	BSC	BTB	H	I	N	Z	C
COM											•	•	◊	◊	1
CPX											•	•	◊	◊	◊
DEC											•	•	◊	◊	•
EOR											•	•	◊	◊	•
INC											•	•	◊	◊	•
JMP											•	•	•	•	•
JSR											•	•	•	•	•
LDA											•	•	◊	◊	•
LDX											•	•	◊	◊	•
LSL											•	•	◊	◊	◊
LSR											•	•	0	◊	◊
MUL											0	•	•	•	0
NEG											•	•	◊	◊	◊
NOP											•	•	•	•	•
ORA											•	•	◊	◊	•
ROL											•	•	◊	◊	◊
ROR											•	•	◊	◊	◊
RSP											•	•	•	•	•
RTI											?	?	?	?	?
RTS											•	•	•	•	•
SBC											•	•	◊	◊	◊
SEC											•	•	•	•	1
SEI											•	1	•	•	•
STA											•	•	◊	◊	•
STOP											•	0	•	•	•
STX											•	•	◊	◊	•
SUB											•	•	◊	◊	◊
SWI											•	1	•	•	•
TAX											•	•	•	•	•
TST											•	•	◊	◊	•
TXA											•	•	•	•	•
WAIT											•	0	•	•	•

Address mode abbreviations

BSC	Bit set/clear	IMM	Immediate
BTB	Bit test & branch	IX	Indexed (no offset)
DIR	Direct	IX1	Indexed, 1 byte offset
EXT	Extended	IX2	Indexed, 2 byte offset
INH	Inherent	REL	Relative

Not implemented

Condition code symbols

H	Half carry (from bit 3)	◊	Tested and set if true, cleared otherwise
I	Interrupt mask	•	Not affected
N	Negate (sign bit)	?	Load CCR from stack
Z	Zero	0	Cleared
C	Carry/borrow	1	Set

Table 9-8 M68HC05 opcode map

	Bit manipulation		Branch	Read/modify/write					Control			Register/memory						
	BTB	BSC	REL	DIR	INH	INH	IX1	IX	INH	INH	IMM	DIR	EXT	IX2	IX1	IX		
High / Low	0 0000	1 0001	2 0010	3 0011	4 0100	5 0101	6 0110	7 0111	8 1000	9 1001	A 1010	B 1011	C 1100	D 1101	E 1110	F 1111	High	Low
0 0000	BRSET0 5 3 BTB 2	BSET0 5 BSC 2	BRA 3 REL 2	NEG 5 DIR 1	NEGA 3 INH 1	NEGX 3 INH 2	NEG 6 IX1 1	NEG 5 IX 1	RTI 9 INH 1		SUB 2 IMM 2	SUB 3 DIR 2	SUB 4 EXT 3	SUB 5 IX2 2	SUB 4 IX1 1	SUB 3 IX 1	0	0000
1 0001	BRCLR0 5 3 BTB 2	BCLR0 5 BSC 2	BRN 3 REL 2						RTS 6 INH 1		CMP 2 IMM 2	CMP 3 DIR 2	CMP 4 EXT 3	CMP 5 IX2 2	CMP 4 IX1 1	CMP 3 IX 1	1	0001
2 0010	BRSET1 5 3 BTB 2	BSET1 5 BSC 2	BHI 3 REL 2		MUL 11 1 INH						SBC 2 IMM 2	SBC 3 DIR 2	SBC 4 EXT 3	SBC 5 IX2 2	SBC 4 IX1 1	SBC 3 IX 1	2	0010
3 0011	BRCLR1 5 3 BTB 2	BCLR1 5 BSC 2	BLS 3 REL 2	COM 5 DIR 1	COMA 3 INH 1	COMX 3 INH 2	COM 6 IX1 1	COM 5 IX 1	SWI 10 INH 1		CPX 2 IMM 2	CPX 3 DIR 2	CPX 4 EXT 3	CPX 5 IX2 2	CPX 4 IX1 1	CPX 3 IX 1	3	0011
4 0100	BRSET2 5 3 BTB 2	BSET2 5 BSC 2	BCC 3 REL 2	LSR 5 DIR 1	LSRA 3 INH 1	LSRX 3 INH 2	LSR 6 IX1 1	LSR 5 IX 1			AND 2 IMM 2	AND 3 DIR 2	AND 4 EXT 3	AND 5 IX2 2	AND 4 IX1 1	AND 3 IX 1	4	0100
5 0101	BRCLR2 5 3 BTB 2	BCLR2 5 BSC 2	BCS 3 REL 2								BIT 2 IMM 2	BIT 3 DIR 2	BIT 4 EXT 3	BIT 5 IX2 2	BIT 4 IX1 1	BIT 3 IX 1	5	0101
6 0110	BRSET3 5 3 BTB 2	BSET3 5 BSC 2	BNE 3 REL 2	ROR 5 DIR 1	RORA 3 INH 1	RORX 3 INH 2	ROR 6 IX1 1	ROR 5 IX 1			LDA 2 IMM 2	LDA 3 DIR 2	LDA 4 EXT 3	LDA 5 IX2 2	LDA 4 IX1 1	LDA 3 IX 1	6	0110
7 0111	BRCLR3 5 3 BTB 2	BCLR3 5 BSC 2	BEQ 3 REL 2	ASR 5 DIR 1	ASRA 3 INH 1	ASRX 3 INH 2	ASR 6 IX1 1	ASR 5 IX 1	TAX 2 INH 1			STA 4 DIR 2	STA 5 EXT 3	STA 6 IX2 2	STA 5 IX1 1	STA 4 IX 1	7	0111
8 1000	BRSET4 5 3 BTB 2	BSET4 5 BSC 2	BHCC 3 REL 2	LSL 5 DIR 1	LSLA 3 INH 1	LSLX 3 INH 2	LSL 6 IX1 1	LSL 5 IX 1	CLC 2 INH 1		EOR 2 IMM 2	EOR 3 DIR 2	EOR 4 EXT 3	EOR 5 IX2 2	EOR 4 IX1 1	EOR 3 IX 1	8	1000
9 1001	BRCLR4 5 3 BTB 2	BCLR4 5 BSC 2	BHCS 3 REL 2	ROL 5 DIR 1	ROLA 3 INH 1	ROLX 3 INH 2	ROL 6 IX1 1	ROL 5 IX 1	SEC 2 INH 1		ADC 2 IMM 2	ADC 3 DIR 2	ADC 4 EXT 3	ADC 5 IX2 2	ADC 4 IX1 1	ADC 3 IX 1	9	1001
A 1010	BRSET5 5 3 BTB 2	BSET5 5 BSC 2	BPL 3 REL 2	DEC 5 DIR 1	DECA 3 INH 1	DECX 3 INH 2	DEC 6 IX1 1	DEC 5 IX 1	CLI 2 INH 1		ORA 2 IMM 2	ORA 3 DIR 2	ORA 4 EXT 3	ORA 5 IX2 2	ORA 4 IX1 1	ORA 3 IX 1	A	1010
B 1011	BRCLR5 5 3 BTB 2	BCLR5 5 BSC 2	BMI 3 REL 2						SEI 2 INH 1		ADD 2 IMM 2	ADD 3 DIR 2	ADD 4 EXT 3	ADD 5 IX2 2	ADD 4 IX1 1	ADD 3 IX 1	B	1011
C 1100	BRSET6 5 3 BTB 2	BSET6 5 BSC 2	BMC 3 REL 2	INC 5 DIR 1	INCA 3 INH 1	INCX 3 INH 2	INC 6 IX1 1	INC 5 IX 1	RSP 2 INH 1			JMP 2 DIR 2	JMP 3 EXT 3	JMP 4 IX2 2	JMP 3 IX1 1	JMP 2 IX 1	C	1100
D 1101	BRCLR6 5 3 BTB 2	BCLR6 5 BSC 2	BMS 3 REL 2	TST 4 DIR 1	TSTA 3 INH 1	TSTX 3 INH 2	TST 5 IX1 1	TST 4 IX 1	NOP 2 INH 1	BSR 6 REL 2	JSR 5 DIR 2	JSR 6 EXT 3	JSR 7 IX2 2	JSR 6 IX1 1	JSR 5 IX 1	D	1101	
E 1110	BRSET7 5 3 BTB 2	BSET7 5 BSC 2	BIL 3 REL 2						STOP 2 INH 1		LDX 2 IMM 2	LDX 3 DIR 2	LDX 4 EXT 3	LDX 5 IX2 2	LDX 4 IX1 1	LDX 3 IX 1	E	1110
F 1111	BRCLR7 5 3 BTB 2	BCLR7 5 BSC 2	BIH 3 REL 2	CLR 5 DIR 1	CLRA 3 INH 1	CLRX 3 INH 2	CLR 6 IX1 1	CLR 5 IX 1	WAIT 2 INH 1	TXA 2 INH 2		STX 4 DIR 2	STX 5 EXT 3	STX 6 IX2 2	STX 5 IX1 1	STX 4 IX 1	F	1111

Abbreviations for address modes and registers

BSC	Bit set/clear	IX	Indexed (no offset)
BTB	Bit test and branch	IX1	Indexed, 1 byte (8-bit) offset
DIR	Direct	IX2	Indexed, 2 byte (16-bit) offset
EXT	Extended	REL	Relative
INH	Inherent	A	Accumulator
IMM	Immediate	X	Index register

Not implemented

Legend

9.3 Addressing modes

Ten different addressing modes provide programmers with the flexibility to optimize their 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) enable access to tables throughout memory. Short absolute (direct) and long absolute (extended) addressing are 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 locations.

The term ‘effective address’ (EA) is used in describing the various addressing modes. The effective address is defined as the address from which the argument for an instruction is fetched or stored. The ten addressing modes of the processor are described below. Parentheses are used to indicate ‘contents of’ the location or register referred to. For example, (PC) indicates the contents of the location pointed to by the PC (program counter). An arrow indicates ‘is replaced by’ and a colon indicates concatenation of two bytes. For additional details and graphical illustrations, refer to the ***M6805 HMOS/M146805 CMOS Family Microcomputer/ Microprocessor User's Manual*** or to the ***M68HC05 Applications Guide***.

9.3.1 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 or accumulator, as well as the control instruction, with no other arguments are included in this mode. These instructions are one byte long.

9.3.2 Immediate

In the immediate addressing mode, the operand is contained in the byte immediately following the opcode. 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).

$$EA = PC+1;\ PC \leftarrow PC+2$$

9.3.3 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.

$$EA = (PC+1);\ PC \leftarrow PC+2$$
$$\text{Address bus high} \leftarrow 0;\ \text{Address bus low} \leftarrow (PC+1)$$

9.3.4 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 short form of the instruction.

EA = (PC+1):(PC+2); PC ← PC+3
Address bus high ← (PC+1); Address bus low ← (PC+2)

9.3.5 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 RAM or I/O location.

EA = X; PC ← PC+1
Address bus high ← 0; Address bus low ← X

9.3.6 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. Therefore the operand can be located anywhere within the lowest 511 memory locations. This addressing mode is useful for selecting the mth element in an n element table.

EA = X+(PC+1); PC ← PC+2
Address bus high ← K; Address bus low ← X+(PC+1)
where K = the carry from the addition of X and (PC+1)

9.3.7 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. This address 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.

EA = X+[(PC+1):(PC+2)]; PC ← PC+3
Address bus high ← (PC+1)+K; Address bus low ← X+(PC+2)
where K = the carry from the addition of X and (PC+2)

9.3.8 Relative

The relative addressing mode is only used in branch instructions. In relative addressing, the contents of the 8-bit signed byte (the offset) following the opcode are 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 –126 to +129 from the opcode address. 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.

EA = PC+2+(PC+1); PC ← EA if branch taken;
otherwise EA = PC ← PC+2

9.3.9 Bit set/clear

In the bit set/clear addressing mode, the bit to be set or cleared is part of the opcode. The byte following the opcode specifies the address 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.

EA = (PC+1); PC ← PC+2
Address bus high ← 0; Address bus low ← (PC+1)

9.3.10 Bit test and branch

The bit test and branch addressing mode is a combination of direct addressing and relative addressing. The bit 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 (EA1). The signed relative 8-bit offset in the third byte (EA2) 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 branch is from –125 to +130 from the opcode address. The state of the tested bit is also transferred to the carry bit of the condition code register.

EA1 = (PC+1); PC ← PC+2
Address bus high ← 0; Address bus low ← (PC+1)
EA2 = PC+3+(PC+2); PC ← EA2 if branch taken;
otherwise PC ← PC+3

THIS PAGE LEFT BLANK INTENTIONALLY

10
LOW POWER MODES

The MC68HC05BD3 has only one low-power operating mode–the Wait Mode. The WAIT instruction provides the only mode that reduces the power required for the MCU by stopping CPU internal clock. The STOP instruction is not implemented in its normal sense. The STOP instruction will be interpreted as the NOP instruction by the CPU if it is ever encountered. The flow of the WAIT mode is shown in Figure 10-1.

10.1 STOP Mode

Stop mode is not implemented on the MC68HC05BD3. The STOP instruction will be treated and executed as a NOP instruction. Therefore, the I-bit in the Condition Code register will not be cleared.

10.2 WAIT Mode

In the WAIT mode the internal processor clock is halted, suspending all processor and internal bus activities. Other Internal clocks remain active, permitting interrupts to be generated from the Multi-Function Timer, M-Bus Interface, and the Sync Signal Processor, or a reset to be generated from the COP watchdog timer. The timer may be used to generate a periodic exit from the WAIT mode. Execution of the WAIT instruction automatically clears the I-bit in the Condition Code register, so that any hardware interrupt can wake up the MCU. All other registers, memory, and input/output lines remain in their previous states.

10.3 COP Watchdog Timer Considerations

The COP watchdog timer is always enabled in MC68HC05BD3. It will reset the MCU when it times out. For a system that must have intentional uses of the WAIT Mode, care must be taken to prevent such situations from happening during normal operations by arranging timely interrupts to reset the COP watchdog timer.

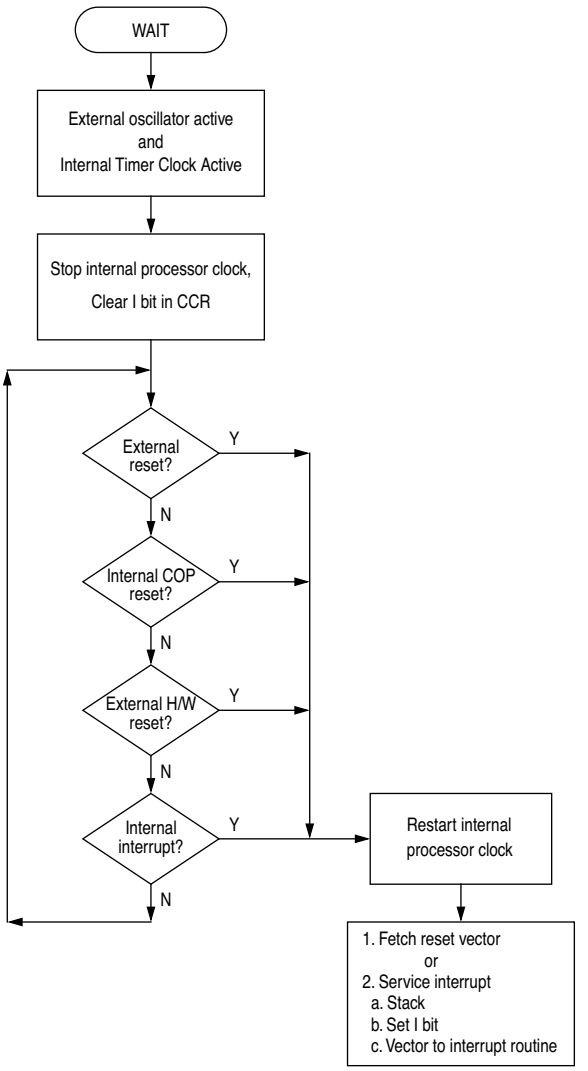


Figure 10-1 WAIT Flowchart

11
OPERATING MODES

The MC68HC05BD3/MC68HC05BD5/*MC68HC705BD3* MCU has two modes of operation, the User Mode and the Self-Check/*Bootstrap* Mode. Figure 11-1 shows the flowchart of entry to these two modes, and Table 11-1 shows operating mode selection.

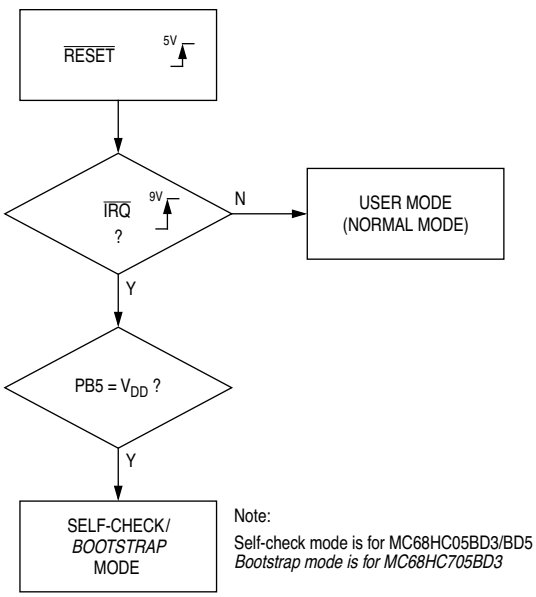


Figure 11-1 Flowchart of Mode Entering

Table 11-1 Mode Selection

$\overline{\text{RESET}}$	$\overline{\text{IRQ}}$	PB5	MODE
5V	V_{SS} to V_{DD}	V_{SS} to V_{DD}	USER
5V	9V +9V Rising Edge*	V_{DD}	SELF-CHECK/ *BOOTSTRAP*
* Minimum hold time should be 2 clock cycles, after that it can be used as a normal $\overline{\text{IRQ}}$ function pin.			

11.1 User Mode (Normal Operation)

The normal operating mode of the MC68HC05BD3/MC68HC05BD5/*MC68HC705BD3* is the user mode. The user mode will be entered if the $\overline{\text{RESET}}$ line is brought low, and the $\overline{\text{IRQ}}$ pin is within its normal operational range (V_{SS} to V_{DD}), the rising edge of the $\overline{\text{RESET}}$ will cause the MCU to enter the user mode.

11.2 Self-Check Mode

The self-check mode is provided on the MC68HC05BD3 and MC68HC05BD5 for the user to check device functions with an on-chip self-check program masked at location $3F00 to $3FDF under minimum hardware support. The hardware is shown in Figure 11-3. Figure 11-2 is the criteria to enter self-check mode, where PB5's condition is latched within first two clock cycles after the rising edge of the reset. PB5 can then be used for other purposes. After entering the self-check mode, CPU branches to the self-check program and carries out the self-check. Self-check is a repetitive test, i.e. if all parts are checked to be good, the CPU will repeat the self-check again. Therefore, the LEDs attached to Port B will be flashing if the device is good; else the combination of LEDs' on-off pattern can show what part of the device is suspected to be bad. Table 11-2 lists the LEDs' on-off patterns and their corresponding indications.

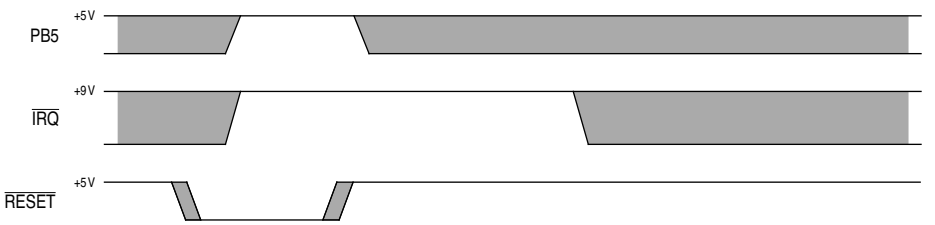


Figure 11-2 Self-Check Mode Timing

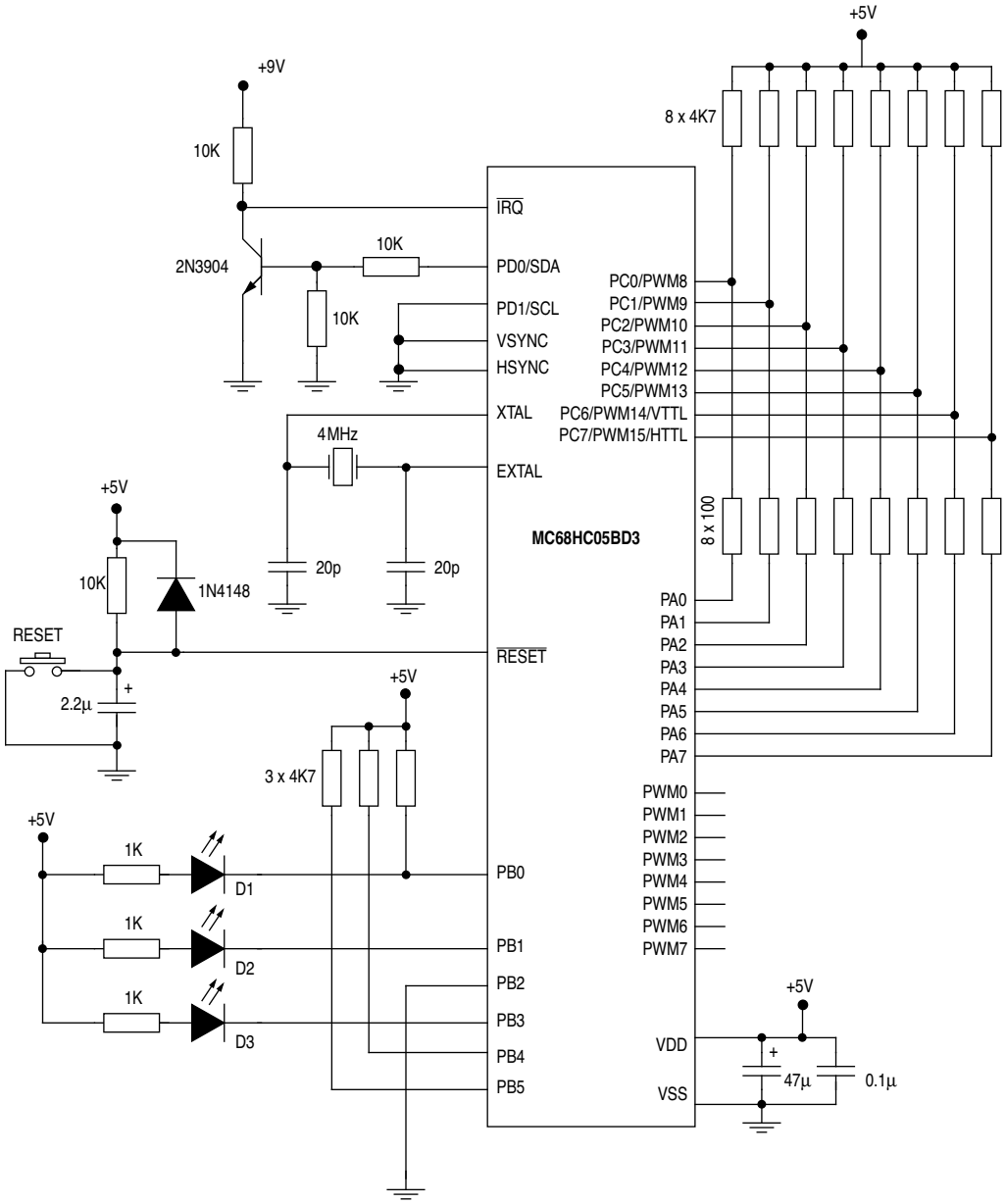


Figure 11-3 MC68HC05BD3 Self-Test Circuit

Table 11-2 Self-Check Report

PB3	PB1	PB0	REMARKS
Flashing			O.K. (self-check is on-going)
1	1	0	Bad I/O
1	0	1	BAD RAM
1	0	0	BAD ROM
0	1	1	BAD $\overline{IRQ}$

1=LED off, 0=LED on

11.3 Bootstrap Mode

The bootstrap mode is provided in the EPROM part (MC68HC705BD3) as a mean of self-programming its EPROM with minimal circuitry. It is entered on the rising edge of $\overline{RESET}$ if $\overline{IRQ}$ pin is at 1.8V_{DD} and PB5 is at logic one. $\overline{RESET}$ must be held low for 4064 cycles after POR (power-on reset) or for a time t_{RL} for any other reset. The user EPROM consists of 7.75K-bytes, from location $2000 to $3EFF.

Refer to Section 15 for further details on MC68HC705BD3.

12
ELECTRICAL SPECIFICATIONS

This section contains the electrical specifications for MC68HC05BD3.

12.1 Maximum Ratings

(Voltages referenced to V_{SS})

RATINGS	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
$\overline{IRQ}$	V_{in}	V_{SS}–0.3 to 2xV_{DD}+0.3	V
Current Drain per pin excluding V_{DD} and V_{SS}	I_D	25	mA
Operating Temperature	T_A	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 should be taken to avoid application of any voltage higher than the maximum rated voltages to this high impedance circuit. For proper operation it is recommended that Vin and Vout be constrained to the range $V_{SS} \le (V_{in} \text{ or } V_{out}) \le V_{DD}$. Reliability of operation is enhanced if unused inputs are connected to an appropriate logic voltage level (e.g. either V_{SS} or V_{DD}).

12.2 Thermal Characteristics

CHARACTERISTICS	SYMBOL	VALUE	UNIT
Thermal resistance - Plastic 40-pin DIP package - Plastic 42-pin SDIP package	 θ_{JA} θ_{JA}	 60 60	 °C/W °C/W

12.3 DC Electrical Characteristics

Table 12-1 DC Electrical Characteristics for MC68HC05BD3

(V_{DD}=5.0Vdc ±10%, V_{SS}=0Vdc, temperature range=0 to 70°C)

CHARACTERISTICS	SYMBOL	MINIMUM	TYPICAL	MAXIMUM	UNIT
Output voltage I_{LOAD} = –10µA I_{LOAD} = +10µA	 V_{OH} V_{OL}	 V_{DD}-0.1 –	 – –	 – 0.1	 V V
Output high voltage (I_{LOAD}=–5mA) PA0-PA7, PB0-PB1, PC6-PC7, PD0-PD1	V_{OH}	V_{DD}-0.8	–	–	V
Output low voltage (I_{LOAD}=+5mA) PA0-PA7, PB0-PB5, PC0-PC7, PD0-PD1, PWM0-PWM7	V_{OL}	–	–	0.4	V
Input high voltage PA0-PA7, PB0-PB5, PC0-PC7, PD0-PD1, $\overline{IRQ}$, $\overline{RESET}$, EXTAL VSYNC, HSYNC (TTL level)	V_{IH}	0.7xV_{DD} 2.0	– –	V_{DD} V_{DD}	V
Input low voltage PA0-PA7, PB0-PB5, PC0-PC7, PD0-PD1, $\overline{IRQ}$, $\overline{RESET}$, EXTAL VSYNC, HSYNC (TTL level)	V_{IL}	V_{SS} V_{SS}	– –	0.2xV_{DD} 0.8	V
Supply current: Run Wait	I_{DD}	– –	4.5 0.73	20 8	mA mA
I/O ports high-Z leakage current PA0-PA7, PB0-PB5, PC0-PC7, PD0-PD1, PWM0-PWM7	I_{IL}	–	–	±10	µA
Input current $\overline{IRQ}$, $\overline{RESET}$, EXTAL, VSYNC, HSYNC	I_{IN}	–	–	1	µA
Capacitance ports (as input or output), $\overline{RESET}$, $\overline{IRQ}$, EXTAL, XTAL, HSYNC, VSYNC	C_{OUT} C_{IN}	– –	– –	12 8	pF pF

Notes:

(1) All values shown reflect average measurements.

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

(3) Wait I_{DD}: only timer system and SSP is active.

(4) Run (operating) I_{DD}, Wait I_{DD}: measured using external square wave clock source to EXTAL (f_{OSC}=4.2MHz), all inputs 0.2 Vdc from rail; no dc loads, less than 50pF on all outputs, C_L=20pF on EXTAL.

(5) Wait I_{DD}: all ports configured as inputs, V_{IL}=0.2 Vdc, V_{IH}=V_{DD} – 0.2 Vdc.

(6) Wait I_{DD} is affected linearly by the EXTAL capacitance.

12.4 Control Timing

Table 12-2 Control Timing

(V_{DD}=5.0Vdc ±10%, V_{SS}=0Vdc, temperature range=0 to 70°C)

CHARACTERISTICS	SYMBOL	MINIMUM	MAXIMUM	UNIT
Frequency of operation Crystal option External clock option	f_{OSC}	– dc	4.2 4.2	MHz MHz
Internal operating frequency (f_{OSC}/2) Crystal External clock	 f_{OP} f_{OP}	 – dc	 2.1 2.1	 MHz MHz
Processor cycle time	t_{CYC}	480	–	ns
Crystal oscillator start-up time	t_{OXOV}	–	100	ms
External RESET pulse width	t_{RL}	1.5	–	t_{CYC}
Watchdog RESET output pulse width	t_{DOGL}	1.5	–	t_{CYC}
Watchdog time-out	t_{DOG}	114688	917504	t_{CYC}
Interrupt pulse width (edge-triggered)	t_{ILIH}	125	–	ns
Interrupt pulse period	t_{ILIL}	–(1)	–	t_{CYC}
EXTAL pulse width	t_{OH}, t_{OL}	100	–	t_{CYC}

Notes:

(1) The minimum period t_{ILIL} should not be less than the number of cycles it takes to execute the interrupt service routine plus 21 t_{CYC}.

12.5 M-Bus Timing

Table 12-3 M-Bus Interface Input Signal Timing

(V_{DD}=5.0Vdc ±10%, V_{SS}=0Vdc, temperature range=0 to 70°C)

PARAMETER	SYMBOL	MINIMUM	MAXIMUM	UNIT
START condition hold time	$t_{HD.STA}$	2	–	t_{CYC}
Clock low period	t_{LOW}	4.7	–	t_{CYC}
Clock high period	t_{HIGH}	4	–	t_{CYC}
Data set-up time	$t_{SU.DAT}$	250	–	ns
Data hold time	$t_{HD.DAT}$	0	–	t_{CYC}
START condition set-up time (for repeated START condition only)	$t_{SU.STA}$	2	–	t_{CYC}
STOP condition set-up time	$t_{SU.STO}$	2	–	t_{CYC}

Table 12-4 M-Bus Interface Output Signal Timing

(V_{DD}=5.0Vdc ±10%, V_{SS}=0Vdc, temperature range=0 to 70°C)

PARAMETER	SYMBOL	MINIMUM	MAXIMUM	UNIT
START condition hold time	$t_{HD.STA}$	8	–	t_{CYC}
Clock low period	t_{LOW}	11	–	t_{CYC}
Clock high period	t_{HIGH}	11	–	t_{CYC}
SDA/SCL rise time (see note 1)	t_R	–	1	µs
SDA/SCL fall time (see note 1)	t_F	–	300	ns
Data set-up time	$t_{SU.DAT}$	$t_{LOW} - t_{CYC}$	–	ns
Data hold time	$t_{HD.DAT}$	1	–	t_{CYC}
START condition set-up time (for repeated START condition only)	$t_{SU.STA}$	10	–	t_{CYC}
STOP condition set-up time	$t_{SU.STO}$	10	–	t_{CYC}

Note:

1. With 200pF loading on the SDA/SCL pins

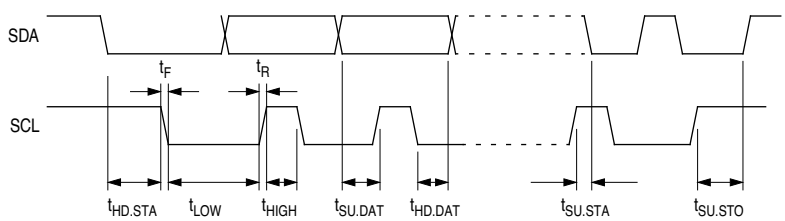


Figure 12-1 M-Bus Timing

12.6 Sync Signal Processor Timing

Table 12-5 Sync Signal Processor Timing

(V_{DD}=5.0Vdc ±10%, V_{SS}=0Vdc, temperature range=0 to 70°C)

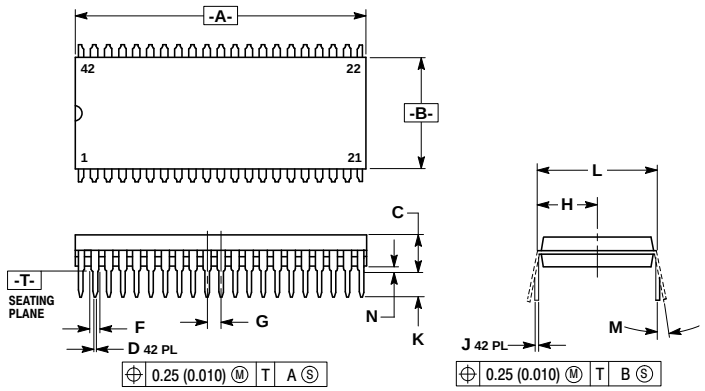
PARAMETER	SYMBOL	MINIMUM	MAXIMUM	UNIT
VSYNC input sync pulse	$t_{VI.SP}$	1		t_{CYC}
HSYNC input sync pulse (except for composite sync input)	$t_{HI.SP}$	1		t_{CYC}
VTTL output sync pulse width for separate sync input	$t_{VO.SP}$	same as input		
VTTL output sync pulse width for composite sync input	$t_{VO.CO}$	input + 9.5µs	input + 10µs	
HTTL output sync pulse width	t_{HO}	same as input		
Free-running VTTL output sync pulse (SOUT clear)	$t_{FVO.SP}$	128		t_{CYC}
Free-running VTTL output period (SOUT clear)	t_{FVO}	31488		t_{CYC}
Free-running HTTL output sync pulse (SOUT clear)	$t_{FHO.SP}$	4		t_{CYC}
Free-running HTTL output period (SOUT clear)	t_{FHO}	41/32		t_{CYC}
Inserted HTTL sync pulse (INSRT cleared)	$t_{IHI.SP}$	4		t_{CYC}
Inserted HTTL period error (INSRT cleared)	$t_{IHI.ER}$	–	1	t_{CYC}
VSYNC to VTTL delay (8pF loading)	t_{VDD}	30	40	ns
HSYNC to HTTL delay (8pF loading)	t_{HHD}	30	40	ns
HSYNC to VTTL delay (composite sync)	t_{HVD}	30	40	ns

THIS PAGE LEFT BLANK INTENTIONALLY

13
MECHANICAL SPECIFICATIONS

This section provides the mechanical dimension for the 42-pin SDIP and 40-pin DIP packages for the MC68HC05BD3.

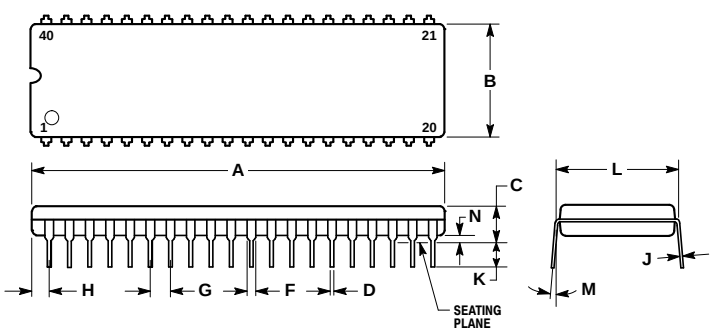
13.1 42-Pin SDIP Package (Case 858-01)



NOTES:
1. DIMENSIONS AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: INCH.
3. DIMENSION L TO CENTER OF LEAD WHEN FORMED PARALLEL.
4. DIMENSIONS A AND B DO NOT INCLUDE MOLD FLASH. MAXIMUM MOLD FLASH 0.25 (0.010).

	INCHES		MILLIMETERS	
DIM	MIN	MAX	MIN	MAX
A	1.435	1.465	36.45	37.21
B	0.540	0.560	13.72	14.22
C	0.155	0.200	3.94	5.08
D	0.014	0.022	0.36	0.56
F	0.032	0.046	0.81	1.17
G	0.070 BSC		1.778 BSC	
H	0.300 BSC		7.62 BSC	
J	0.008	0.015	0.20	0.38
K	0.115	0.135	2.92	3.43
L	0.600 BSC		15.24 BSC	
M	0°	15°	0°	15°
N	0.020	0.040	0.51	1.02

13.2 40-Pin DIP Package (Case 711-03)



NOTES:
1. POSITIONAL TOLERANCE OF LEADS (D), SHALL BE WITHIN 0.25 mm (0.010) AT MAXIMUM MATERIAL CONDITION, IN RELATION TO SEATING PLANE AND EACH OTHER.
2. DIMENSION L TO CENTER OF LEADS WHEN FORMED PARALLEL.
3. DIMENSION B DOES NOT INCLUDE MOLD FLASH.

	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	51.69	52.45	2.035	2.065
B	13.72	14.22	0.540	0.560
C	3.94	5.08	0.155	0.200
D	0.36	0.56	0.014	0.022
F	1.02	1.52	0.040	0.060
G	2.54 BSC		0.100 BSC	
H	1.65	2.16	0.065	0.085
J	0.20	0.38	0.008	0.015
K	2.92	3.43	0.115	0.135
L	15.24 BSC		0.600 BSC	
M	0°	15°	0°	15°
N	0.51	1.02	0.020	0.040

THIS PAGE LEFT BLANK INTENTIONALLY

14
MC68HC705BD3

The MC68HC705BD3 is functionally equivalent to MC68HC05BD3, but with increased RAM size to 256 bytes and the user ROM is replaced by an 7.75K-bytes user EPROM (located from $2000 to $3EFF). The entire MC68HC05BD3 data sheet applies to the MC68HC705BD3, with exceptions outlined in the section.

14.1 Features

- Functionally equivalent to MC68HC05BD3
- 256 bytes on-chip RAM
- 7.75K-bytes user EPROM

14.2 Memory Map

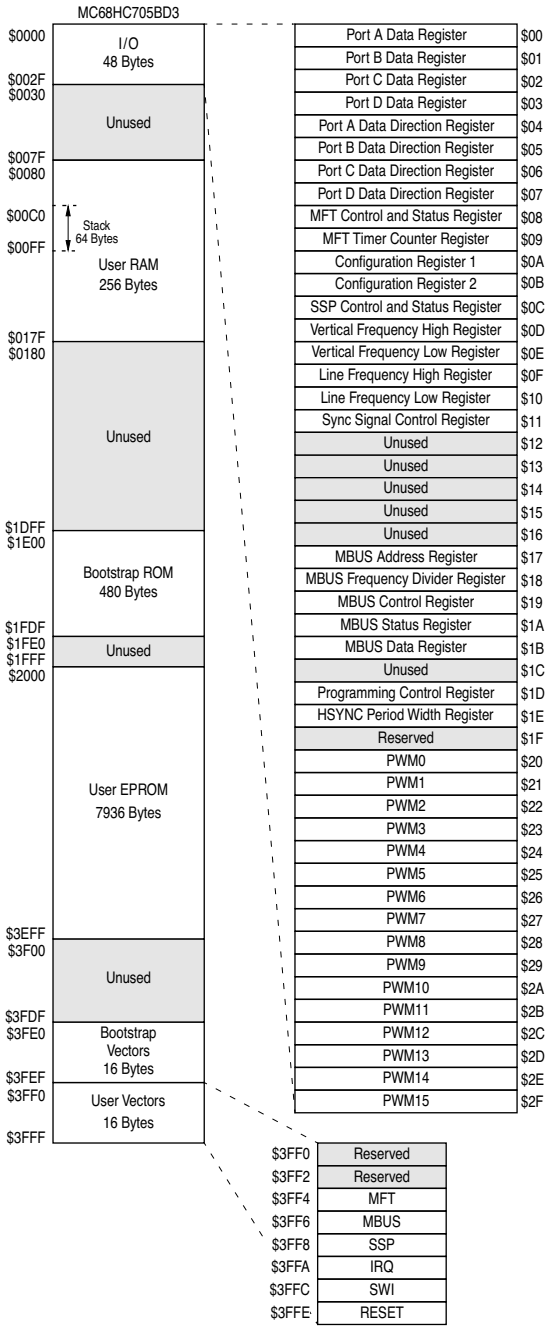
Figure 14-1 shows the memory map for the MC68HC705BD3.

14.3 EPROM Programming

The following programming boards are available from Motorola for programming the on-chip EPROM in the MC68HC705BD3.

Table 14-1 MC68HC705BD3 Programming Boards

Platform board			Adapter	
For programming a single device at a time:	M68HC705UPGMR	+	M68UPA05BD3P40	for 40-pin PDIP
For programming up to 10 devices at a time:	M68HC705UGANG		M68UPA05BD3B42	for 42-pin SDIP


MC68HC705BD3
$0000
I/O
48 Bytes
$002F
$0030
Unused
$007F
$0080
$00C0
Stack
64 Bytes
$00FF
User RAM
256 Bytes
$017F
$0180
Unused
$1DFF
$1E00
Bootstrap ROM
480 Bytes
$1FDF
$1FE0
Unused
$1FFF
$2000
User EPROM
7936 Bytes
$3EFF
$3F00
Unused
$3FDF
$3FE0
Bootstrap
Vectors
16 Bytes
$3FEF
$3FF0
User Vectors
16 Bytes
$3FFF
Port A Data Register $00
Port B Data Register $01
Port C Data Register $02
Port D Data Register $03
Port A Data Direction Register $04
Port B Data Direction Register $05
Port C Data Direction Register $06
Port D Data Direction Register $07
MFT Control and Status Register $08
MFT Timer Counter Register $09
Configuration Register 1 $0A
Configuration Register 2 $0B
SSP Control and Status Register $0C
Vertical Frequency High Register $0D
Vertical Frequency Low Register $0E
Line Frequency High Register $0F
Line Frequency Low Register $10
Sync Signal Control Register $11
Unused $12
Unused $13
Unused $14
Unused $15
Unused $16
MBUS Address Register $17
MBUS Frequency Divider Register $18
MBUS Control Register $19
MBUS Status Register $1A
MBUS Data Register $1B
Unused $1C
Programming Control Register $1D
HSYNC Period Width Register $1E
Reserved $1F
PWM0 $20
PWM1 $21
PWM2 $22
PWM3 $23
PWM4 $24
PWM5 $25
PWM6 $26
PWM7 $27
PWM8 $28
PWM9 $29
PWM10 $2A
PWM11 $2B
PWM12 $2C
PWM13 $2D
PWM14 $2E
PWM15 $2F
$3FF0 Reserved
$3FF2 Reserved
$3FF4 MFT
$3FF6 MBUS
$3FF8 SSP
$3FFA IRQ
$3FFC SWI
$3FFE RESET

14.3.1 Programming Control Register (PCR)

The EPROM Programming Control register controls the actual programming of the EPROM in the MC68HC705BD3.

Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
$001D							ELAT	PGM	---- --00

ELAT - EPROM Latch Control

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

0 (clear) – EPROM address and data bus configured for normal reads.

PGM - EPROM Program Command

1 (set) – Programming power switched on to EPROM array. If ELAT≠1 then PGM=0.

0 (clear) – Programming power switched off to EPROM array.

14.3.2 EPROM Programming Sequence

Programming the EPROM of the MC68HC705BD3 is as follows:

1) Set the ELAT bit.
2) Write the data to be programmed to the address to be programmed.
3) Set the PGM bit.
4) Delay for the appropriate amount of time.
5) Clear the PGM and the ELAT bits.

The last action may be carried out in a single CPU write operation. It is important to remember that an external programming voltage must be applied to the V_{PP} pin while programming, but should be equal to V_{DD} during normal operation.

Example shows address $2000 is programmed with $00.

```
CLR    PCR          ;reset PCR
LDX    #$00         ;load index register with 00
BSET   1,PCR        ;set ELAT bit
LDA    #$00         ;load data=00 in to A
STA    $2000,X      ;latch data and address
```

```
BSET    0,PCR           ;program
JSR     DELAY           ;call delay subroutine
CLR     PCR             ;reset PCR
```

14.4 DC Electrical Characteristics

Table 14-2 DC Electrical Characteristics for MC68HC705BD3

(V_{DD}=5.0Vdc ±10%, V_{SS}=0Vdc, temperature range=0 to 70°C)

CHARACTERISTICS	SYMBOL	MINIMUM	TYPICAL	MAXIMUM	UNIT
Output voltage I_{LOAD} = –10µA I_{LOAD} = +10µA	 V_{OH} V_{OL}	 V_{DD}-0.1 –	 – –	 – 0.1	 V V
Output high voltage (I_{LOAD}=–5mA) PA0-PA7, PB0-PB1, PC6-PC7, PD0-PD1	V_{OH}	V_{DD}-0.8	–	–	V
Output low voltage (I_{LOAD}=+5mA) PA0-PA7, PB0-PB5, PC0-PC7, PD0-PD1, PWM0-PWM7	V_{OL}	–	–	0.4	V
Input high voltage PA0-PA7, PB0-PB5, PC0-PC7, PD0-PD1, $\overline{IRQ}$, $\overline{RESET}$, EXTAL VSYNC, HSYNC (TTL level)	V_{IH}	0.7xV_{DD} 2.0	– –	V_{DD} V_{DD}	V
Input low voltage PA0-PA7, PB0-PB5, PC0-PC7, PD0-PD1, $\overline{IRQ}$, $\overline{RESET}$, EXTAL VSYNC, HSYNC (TTL level)	V_{IL}	V_{SS} V_{SS}	– –	0.2xV_{DD} 0.8	V
Supply current: Run Wait	I_{DD}	– –	6 2	20 8	mA mA
I/O ports high-Z leakage current PA0-PA7, PB0-PB5, PC0-PC7, PD0-PD1, PWM0-PWM7	I_{IL}	–	–	±10	µA
Input current $\overline{IRQ}$, $\overline{RESET}$, EXTAL, VSYNC, HSYNC	I_{IN}	–	–	1	µA
Capacitance ports (as input or output), $\overline{RESET}$, $\overline{IRQ}$, EXTAL, XTAL, HSYNC, VSYNC	C_{OUT} C_{IN}	– –	– –	12 8	pF pF

Notes:

(1) All values shown reflect average measurements.

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

(3) Wait I_{DD}: only timer system and SSP is active.

(4) Run (operating) I_{DD}, Wait I_{DD}: measured using external square wave clock source to EXTAL (f_{OSC}=4.2MHz), all inputs 0.2 Vdc from rail; no dc loads, less than 50pF on all outputs, C_L=20pF on EXTAL.

(5) Wait I_{DD}: all ports configured as inputs, V_{IL}=0.2 Vdc, V_{IH}=V_{DD} – 0.2 Vdc.

(6) Wait I_{DD} is affected linearly by the EXTAL capacitance.

15
MC68HC05BD5

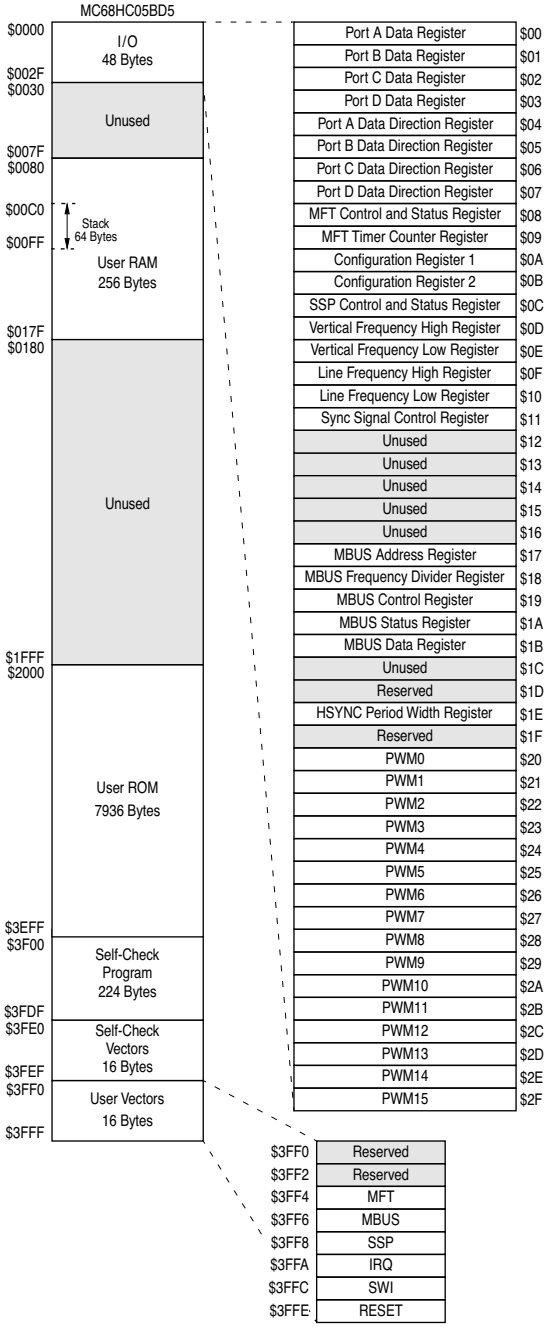
The MC68HC05BD5 is functionally equivalent to MC68HC05BD3, but with increased RAM size of 256 bytes and ROM size of 7.75K-bytes. The entire MC68HC05BD3 data sheet applies to the MC68HC05BD5, with exceptions outlined in the section.

15.1 Features

- Functionally equivalent to MC68HC05BD3
- 256 bytes on-chip RAM
- 7.75K-bytes user ROM

15.2 Memory Map

Figure 15-1 shows the memory map for the MC68HC05BD5.


MC68HC05BD5
$0000
I/O
48 Bytes
$002F
$0030
Unused
$007F
$0080
$00C0
Stack
64 Bytes
$00FF
User RAM
256 Bytes
$017F
$0180
Unused
$1FFF
$2000
User ROM
7936 Bytes
$3EFF
$3F00
Self-Check
Program
224 Bytes
$3FDF
$3FE0
Self-Check
Vectors
16 Bytes
$3FEF
$3FF0
User Vectors
16 Bytes
$3FFF
Port A Data Register $00
Port B Data Register $01
Port C Data Register $02
Port D Data Register $03
Port A Data Direction Register $04
Port B Data Direction Register $05
Port C Data Direction Register $06
Port D Data Direction Register $07
MFT Control and Status Register $08
MFT Timer Counter Register $09
Configuration Register 1 $0A
Configuration Register 2 $0B
SSP Control and Status Register $0C
Vertical Frequency High Register $0D
Vertical Frequency Low Register $0E
Line Frequency High Register $0F
Line Frequency Low Register $10
Sync Signal Control Register $11
Unused $12
Unused $13
Unused $14
Unused $15
Unused $16
MBUS Address Register $17
MBUS Frequency Divider Register $18
MBUS Control Register $19
MBUS Status Register $1A
MBUS Data Register $1B
Unused $1C
Reserved $1D
HSYNC Period Width Register $1E
Reserved $1F
PWM0 $20
PWM1 $21
PWM2 $22
PWM3 $23
PWM4 $24
PWM5 $25
PWM6 $26
PWM7 $27
PWM8 $28
PWM9 $29
PWM10 $2A
PWM11 $2B
PWM12 $2C
PWM13 $2D
PWM14 $2E
PWM15 $2F
$3FF0 Reserved
$3FF2 Reserved
$3FF4 MFT
$3FF6 MBUS
$3FF8 SSP
$3FFA IRQ
$3FFC SWI
$3FFE RESET

15.3 DC Electrical Characteristics

Table 15-1 DC Electrical Characteristics for MC68HC05BD5

(V_{DD}=5.0Vdc ±10%, V_{SS}=0Vdc, temperature range=0 to 70°C)

CHARACTERISTICS	SYMBOL	MINIMUM	TYPICAL	MAXIMUM	UNIT
Output voltage I_{LOAD} = –10µA I_{LOAD} = +10µA	 V_{OH} V_{OL}	 V_{DD}-0.1 –	 – –	 – 0.1	 V V
Output high voltage (I_{LOAD}=–5mA) PA0-PA7, PB0-PB1, PC6-PC7, PD0-PD1	V_{OH}	V_{DD}-0.8	–	–	V
Output low voltage (I_{LOAD}=+5mA) PA0-PA7, PB0-PB5, PC0-PC7, PD0-PD1, PWM0-PWM7	V_{OL}	–	–	0.4	V
Input high voltage PA0-PA7, PB0-PB5, PC0-PC7, PD0-PD1, $\overline{IRQ}$, $\overline{RESET}$, EXTAL VSYNC, HSYNC (TTL level)	V_{IH}	0.7xV_{DD} 2.0	– –	V_{DD} V_{DD}	V
Input low voltage PA0-PA7, PB0-PB5, PC0-PC7, PD0-PD1, $\overline{IRQ}$, $\overline{RESET}$, EXTAL VSYNC, HSYNC (TTL level)	V_{IL}	V_{SS} V_{SS}	– –	0.2xV_{DD} 0.8	V
Supply current: Run Wait	I_{DD}	– –	7 1.3	20 8	mA mA
I/O ports high-Z leakage current PA0-PA7, PB0-PB5, PC0-PC7, PD0-PD1, PWM0-PWM7	I_{IL}	–	–	±10	µA
Input current $\overline{IRQ}$, $\overline{RESET}$, EXTAL, VSYNC, HSYNC	I_{IN}	–	–	1	µA
Capacitance ports (as input or output), $\overline{RESET}$, $\overline{IRQ}$, EXTAL, XTAL, HSYNC, VSYNC	C_{OUT} C_{IN}	– –	– –	12 8	pF pF

Notes:

(1) All values shown reflect average measurements.

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

(3) Wait I_{DD}: only timer system and SSP is active.

(4) Run (operating) I_{DD}, Wait I_{DD}: measured using external square wave clock source to EXTAL (f_{OSC}=4.2MHz), all inputs 0.2 Vdc from rail; no dc loads, less than 50pF on all outputs, C_L=20pF on EXTAL.

(5) Wait I_{DD}: all ports configured as inputs, V_{IL}=0.2 Vdc, V_{IH}=V_{DD} – 0.2 Vdc.

(6) Wait I_{DD} is affected linearly by the EXTAL capacitance.

THIS PAGE LEFT BLANK INTENTIONALLY

GENERAL DESCRIPTION 1

PIN DESCRIPTION AND I/O PORTS 2

MEMORY AND REGISTERS 3

RESETS AND INTERRUPTS 4

MULTI-FUNCTION TIMER 5

PULSE WIDTH MODULATION 6

M-BUS SERIAL INTERFACE 7

SYNC SIGNAL PROCESSOR 8

CPU CORE AND INSTRUCTION SET 9

LOW POWER MODES 10

OPERATING MODES 11

ELECTRICAL SPECIFICATIONS 12

MECHANICAL SPECIFICATIONS 13

MC68HC705BD3 14

MC68HC05BD5 15

1 GENERAL DESCRIPTION

2 PIN DESCRIPTION AND I/O PORTS

3 MEMORY AND REGISTERS

4 RESETS AND INTERRUPTS

5 MULTI-FUNCTION TIMER

6 PULSE WIDTH MODULATION

7 M-BUS SERIAL INTERFACE

8 SYNC SIGNAL PROCESSOR

9 CPU CORE AND INSTRUCTION SET

10 LOW POWER MODES

11 OPERATING MODES

12 ELECTRICAL SPECIFICATIONS

13 MECHANICAL SPECIFICATIONS

14 MC68HC705BD3

15 MC68HC05BD5

Home Page:
www.freescale.com
email:
support@freescale.com
USA/Europe or Locations Not Listed:
Freescale Semiconductor
Technical Information Center, CH370
1300 N. Alma School Road
Chandler, Arizona 85224
(800) 521-6274
480-768-2130
support@freescale.com
Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
support@freescale.com
Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku
Tokyo 153-0064, Japan
0120 191014
+81 2666 8080
support.japan@freescale.com
Asia/Pacific:
Freescale Semiconductor Hong Kong Ltd.
Technical Information Center
2 Dai King Street
Tai Po Industrial Estate,
Tai Po, N.T., Hong Kong
+800 2666 8080
support.asia@freescale.com
For Literature Requests Only:
Freescale Semiconductor
Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
(800) 441-2447
303-675-2140
Fax: 303-675-2150
LDCForFreescaleSemiconductor
@hibbertgroup.com

RoHS-compliant and/or Pb- free versions of Freescale products have the functionality and electrical characteristics of their non-RoHS-compliant and/or non-Pb- free counterparts. For further information, see http://www.freescale.com or contact your Freescale sales representative.

For information on Freescale.s Environmental Products program, go to http://www.freescale.com/epp.

Information in this document is provided solely to enable system and software implementers to use Freescale Semiconductor products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document.
Freescale Semiconductor reserves the right to make changes without further notice to any products herein. Freescale Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale Semiconductor assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters which may be provided in Freescale Semiconductor data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including “Typicals” must be validated for each customer application by customer’s technical experts. Freescale Semiconductor does not convey any license under its patent rights nor the rights of others. Freescale Semiconductor products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Freescale Semiconductor product could create a situation where personal injury or death may occur. Should Buyer purchase or use Freescale Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold Freescale Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Freescale Semiconductor was negligent regarding the design or manufacture of the part.

freescale™
semiconductor