



**SYM53C875  
PCI-SCSI I/O  
Processor with  
UltraSCSI**

**DATA MANUAL  
REVISION 2.0**

***Fast-20***

---

The product described in this publication is a product of Symbios Logic Inc.

SCRIPTS is a trademark and TolerANT is a registered trademark of Symbios Logic Inc.

UltraSCSI is the term used by the SCSI Trade Association to describe Fast-20 SCSI, as documented in the SCSI-3 Fast-20 Parallel Interface standard, X3.277-199X.

It is the policy of Symbios Logic to improve products as new technology, components, software, and firmware become available. Symbios Logic, therefore, reserves the right to change specifications without notice.

The products in this manual are not intended for use in life-support appliances, devices, or systems. Use of these products in such applications without the written consent of the appropriate Symbios Logic officer is prohibited

Copyright ©1995, 1996  
By Symbios Logic Inc.  
All Rights Reserved  
Printed in U.S.A.

We use comments from our readers to improve Symbios product literature. Please e-mail any comments regarding technical documentation to [pubs@symbios.com](mailto:pubs@symbios.com).

---

## Preface

This manual assumes some prior knowledge of current and proposed SCSI and PCI standards. For background information, please contact:

**ANSI**

11 West 42nd Street  
New York, NY 10036  
(212) 642-4900  
Ask for document number X3.131-1994 (SCSI-2)

**Global Engineering Documents**

15 Inverness Way East  
Englewood, CO 80112  
(800) 854-7179 or (303) 792-2181 (outside U.S.)  
Ask for document number X3.131-1994 (SCSI-2) or X3.253 (*SCSI-3 Parallel Interface*)

**ENDL Publications**

14426 Black Walnut Court  
Saratoga, CA 95070  
(408) 867-6642  
Document names: *SCSI Bench Reference*, *SCSI Encyclopedia*

**Prentice Hall**

Englewood Cliffs, NJ 07632  
(201) 767-5937  
Ask for document number ISBN 0-13-796855-8, *SCSI: Understanding the Small Computer System Interface*

**Symbios Logic Electronic Bulletin Board**

(719) 573-3562

**SCSI Electronic Bulletin Board**

(719) 574-0424

**Symbios Logic Internet Anonymous FTP Site**

ftp.symbios.com (204.131.200.1)  
Directory: /pub/ncrchips/scsi

**Symbios Logic World Wide Web Home Page**

<http://www.symbios.com>

**PCI Special Interest Group**

P.O. Box 10470  
Portland, OR 97214  
(800) 433-5177; (503) 797-4201 (International); FAX (503) 234-6762

**Symbios Logic PCI-SCSI Programming Guide**

---

## Revision Record

---

Page No.	Date	Remarks
n/a	6/95	Rev 1.0
1-2, 1-4, 1-5, 2-3, 2-12, 2-13, 2-14, 2-17, 3-3, 3-7, 3-12, 3-13, 4-15, 5-10, 5-11, 5-15, 5-16, 5-21, 5-23, 5-35, 5-38, 5-48, 5-49, 5-51, 6-6, 7-2, 7-4, 7-9, 7-32, 7-33, 7-34	3/96	Rev 2.0 Fast-20 changed to UltraSCSI throughout, except for name of Fast-20 Enable bit; added SCSI clock doubler

---



---

# Contents

Preface .....	i
Revision Record .....	ii
List of Figures .....	xiii
List of Tables .....	xv

## Chapter 1

### Introduction

What is Covered in This Manual .....	1-1
General Description .....	1-1
Package and Feature Options .....	1-2
Benefits of UltraSCSI .....	1-2
TolerANT Technology .....	1-3
SYM53C875 Benefits Summary .....	1-3
SCSI Performance .....	1-3
PCI Performance .....	1-4
Integration .....	1-4
Ease of Use .....	1-4
Flexibility .....	1-5
Reliability .....	1-5
Testability .....	1-5

## Chapter 2

### Functional Description

SCSI Core .....	2-1
DMA Core .....	2-1
SCRIPTS Processor .....	2-1
Internal SCRIPTS RAM .....	2-2
SDMS: The Total SCSI Solution .....	2-2
Designing an UltraSCSI System .....	2-2
Using the SCSI Clock Doubler .....	2-3
Prefetching SCRIPTS Instructions .....	2-3
Op Code Fetch Burst Capability .....	2-4
External Memory Interface .....	2-4
PCI Cache Mode .....	2-5
Load/Store Instructions .....	2-5
3.3 Volt/5 Volt PCI Interface .....	2-5
Additional Access to General Purpose Pins .....	2-5

JTAG Boundary Scan Testing .....	2-6
Big and Little Endian Support .....	2-6
Loopback Mode .....	2-7
Parity Options .....	2-7
DMA FIFO .....	2-10
Data Paths .....	2-11
Asynchronous SCSI Send .....	2-11
Synchronous SCSI Send .....	2-11
Asynchronous SCSI Receive .....	2-12
Synchronous SCSI Receive .....	2-12
SCSI Bus Interface .....	2-13
Differential Mode .....	2-13
8-bit/16-bit SCSI and the Differential Interface .....	2-14
Terminator Networks .....	2-16
(Re)Select During (Re)Selection .....	2-18
Synchronous Operation .....	2-18
Determining the Data Transfer Rate .....	2-18
SCNTL3 Register, bits 6–4 (SCF2–0) .....	2-19
SCNTL3 Register, bits 2–0 (CCF2–0) .....	2-19
SXFER Register, bits 7–5 (TP2–0) .....	2-19
Achieving Optimal SCSI Send Rates .....	2-19
UltraSCSI Synchronous Data Transfers .....	2-19
Interrupt Handling .....	2-21
Polling and Hardware Interrupts .....	2-21
Registers .....	2-21
ISTAT .....	2-21
SIST0 and SIST1 .....	2-21
DSTAT .....	2-21
SIEN0 and SIEN1 .....	2-21
DIEN .....	2-22
DCNTL .....	2-22
Fatal vs. Non-Fatal Interrupts .....	2-22
Masking .....	2-22
Stacked Interrupts .....	2-23
Halting in an Orderly Fashion .....	2-23
Sample Interrupt Service Routine .....	2-24
Chained Block Moves .....	2-24
Wide SCSI Send Bit .....	2-24
Wide SCSI Receive Bit .....	2-25
SWIDE Register .....	2-25
SODL Register .....	2-25
Chained Block Move SCRIPTS Instruction .....	2-25

**Chapter 3**

**PCI Functional Description**

PCI Addressing ..... 3-1

PCI Bus Commands and Functions Supported ..... 3-1

PCI Cache Mode ..... 3-3

    Support for PCI Cache Line Size Register ..... 3-3

    Selection of Cache Line Size ..... 3-3

    Alignment ..... 3-3

    Memory Move Misalignment ..... 3-4

    Memory Write and Invalidate Command ..... 3-4

        Multiple Cache Line Transfers ..... 3-4

        Latency ..... 3-5

        PCI Target Retry ..... 3-5

        PCI Target Disconnect ..... 3-5

    Memory Read Line Command ..... 3-5

    Memory Read Multiple Command ..... 3-6

        Burst Size Selection ..... 3-6

        Read Multiple with Read Line Enabled ..... 3-6

    Unsupported PCI Commands ..... 3-6

Configuration Registers ..... 3-7

    Register 00h

        Vendor ID

        Read Only ..... 3-8

    Register 02h

        Device ID

        Read Only ..... 3-8

    Register 04h

        Command

        Read/Write ..... 3-8

    Register 06h

        Status

        Read/Write ..... 3-10

    Register 08h

        Revision ID

        Read Only ..... 3-11

    Register 09h

        Class Code

        Read Only ..... 3-11

    Register 0Ch

        Cache Line Size

        Read/Write ..... 3-11

    Register 0Dh

        Latency Timer

        Read/Write ..... 3-12

    Register 0Eh

        Header Type

        Read Only ..... 3-12

Register 10h	
Base Address Zero (I/O)	
Read/Write .....	3-12
Register 14h	
Base Address One (Memory)	
Read/Write .....	3-12
Register 18h	
RAM Base Address	
Read/Write .....	3-12
Register 2C-2F	
Subsystem Data	
Read/Write .....	3-12
Register 30h	
Expansion ROM Base Address	
Read/Write .....	3-12
Register 34-35h	
General Purpose Pin Control	
Read/Write .....	3-13
Register 3Ch	
Interrupt Line	
Read/Write .....	3-13
Register 3Dh	
Interrupt Pin	
Read Only .....	3-13
Register 3Eh	
Min_Gnt	
Read Only .....	3-13
Register 3Fh	
Max_Lat	
Read Only .....	3-13

**Chapter 4**

**Signal Descriptions**

53C875, 53C875J Power and Ground Pins .....	4-5
53C875N Power and Ground Pins .....	4-6
53C875JB Power and Ground Pins .....	4-6
System Pins .....	4-8
Address and Data Pins .....	4-9
Interface Control Pins .....	4-11
Arbitration Pins .....	4-12
Error Reporting Pins .....	4-12
SCSI Pins .....	4-13
Additional Interface Pins .....	4-15
External Memory Interface Pins .....	4-17
JTAG Pins .....	4-18

**Chapter 5****Operating Registers**

Register 00 (80)	
SCSI Control Zero (SCNTL0)	
Read/Write .....	5-5
Register 01 (81)	
SCSI Control One (SCNTL1)	
Read/Write .....	5-7
Register 02 (82)	
SCSI Control Two (SCNTL2)	
Read/Write .....	5-9
Register 03 (83)	
SCSI Control Three (SCNTL3)	
Read/Write .....	5-10
Register 04 (84)	
SCSI Chip ID (SCID)	
Read/Write .....	5-12
Register 05 (85)	
SCSI Transfer (SXFER)	
Read/Write .....	5-12
Register 06 (86)	
SCSI Destination ID (SDID)	
Read/Write .....	5-14
Register 07 (87)	
General Purpose (GPREG)	
Read/Write .....	5-15
SDMS Use of GPIO Pins .....	5-15
Register 08 (88)	
SCSI First Byte Received (SFBR)	
Read/Write .....	5-16
Register 09 (89)	
SCSI Output Control Latch (SOCL)	
Read /Write .....	5-16
Register 0A (09)	
SCSI Selector ID (SSID)	
Read Only .....	5-17
Register 0B (8B)	
SCSI Bus Control Lines (SBCL)	
Read Only .....	5-17
Register 0C (8C)	
DMA Status (DSTAT)	
Read Only .....	5-18
Register 0D (8D)	
SCSI Status Zero (SSTAT0)	
Read Only .....	5-19
Register 0E (8E)	
SCSI Status One (SSTAT1)	
Read Only .....	5-20

- Register 0F (8F)
  - SCSI Status Two (SSTAT2)  
(Read Only) ..... 5-21
- Registers 10-13 (90-93)
  - Data Structure Address (DSA)  
Read/Write ..... 5-22
- Register 14 (94)
  - Interrupt Status (ISTAT)  
(Read/Write) ..... 5-23
- Register 18 (98)
  - Chip Test Zero (CTEST0)  
Read/Write ..... 5-25
- Register 19 (99)
  - Chip Test One (CTEST1)  
Read Only ..... 5-25
- Register 1A (9A)
  - Chip Test Two (CTEST2)  
Read/Write ..... 5-26
- Register 1B (9B)
  - Chip Test Three (CTEST3)  
Read/Write ..... 5-27
- Registers 1C-1F (9C-9F)
  - Temporary (TEMP)  
Read/Write ..... 5-28
- Register 20 (A0)
  - DMA FIFO (DFIFO)  
Read/Write ..... 5-28
- Register 21 (A1)
  - Chip Test Four (CTEST4)  
Read/Write ..... 5-29
- Register 22 (A2)
  - Chip Test Five (CTEST5)  
Read/Write ..... 5-30
- Register 23 (A3)
  - Chip Test Six (CTEST6)  
Read/Write ..... 5-31
- Registers 24-26 (A4-A6)
  - DMA Byte Counter (DBC)  
Read/Write ..... 5-32
- Register 27 (A7)
  - DMA Command (DCMD)  
Read/Write ..... 5-32
- Registers 28-2B (A8-AB)
  - DMA Next Address (DNAD)  
Read/Write ..... 5-33
- Registers 2C-2F (AC-AF)
  - DMA SCRIPTS Pointer (DSP)  
Read/Write ..... 5-33
- Registers 30-33 (B0-B3)
  - DMA SCRIPTS Pointer Save (DSPTS)  
Read/Write ..... 5-34



Registers 34-37 (B4-B7)	
Scratch Register A (SCRATCH A)	
Read/Write .....	5-34
Register 38 (B8)	
DMA Mode (DMODE)	
Read/Write .....	5-35
Register 39 (B9)	
DMA Interrupt Enable (DIEN)	
Read/Write .....	5-36
Register 3A (BA)	
Scratch Byte Register (SBR)	
Read/Write .....	5-37
Register 3B (BB)	
DMA Control (DCNTL)	
Read/Write .....	5-38
Register 3C-3F (BC-BF)	
Adder Sum Output (ADDER)	
Read Only .....	5-39
Register 40 (C0)	
SCSI Interrupt Enable Zero (SIEN0)	
Read/Write .....	5-39
Register 41 (C1)	
SCSI Interrupt Enable One (SIEN1)	
Read/Write .....	5-40
Register 42 (C2)	
SCSI Interrupt Status Zero (SIST0)	
Read Only .....	5-41
Register 43 (C3)	
SCSI Interrupt Status One (SIST1)	
Read Only .....	5-42
Register 44 (C4)	
SCSI Longitudinal Parity (SLPAR)	
Read/Write .....	5-43
Register 45 (C5)	
SCSI Wide Residue (SWIDE)	
Read/Write .....	5-44
Register 46 (C6)	
Memory Access Control (MACNTL)	
Read/Write .....	5-44
Register 47 (C7)	
General Purpose Pin Control (GPCNTL)	
Read/Write .....	5-45
Register 48 (C8)	
SCSI Timer Zero (STIME0)	
Read /Write .....	5-45
Register 49 (C9)	
SCSI Timer One (STIME1)	
Read/Write .....	5-46
Register 4A (CA)	
Response ID Zero (RESPID0)	
Read/Write .....	5-48

Register 4B (CB)  
 Response ID One (RESPID1)  
 Read/Write ..... 5-48

Register 4C (CC)  
 SCSI Test Zero (STEST0)  
 Read Only ..... 5-48

Register 4D (CD)  
 SCSI Test One (STEST1)  
 Read/Write ..... 5-49  
 Doubling the SCSI CLK Frequency ..... 5-50

Register 4E (CE)  
 SCSI Test Two (STEST2)  
 Read/Write ..... 5-50

Register 4F (CF)  
 SCSI Test Three (STEST3)  
 Read/Write ..... 5-51

Register 50-51 (D0-D1)  
 SCSI Input Data Latch (SIDL)  
 Read Only ..... 5-53

Registers 54-55 (D4-D5)  
 SCSI Output Data Latch (SODL)  
 Read/Write ..... 5-53

Registers 58-59 (D8-D9)  
 SCSI Bus Data Lines (SBDL)  
 Read Only ..... 5-54

Registers 5C-5F (DC-DF)  
 Scratch Register B (SCRATCHB)  
 (Read/Write) ..... 5-54

Registers 60h-7Fh (E0h-FFh)  
 Scratch Registers C-J  
 (SCRATCHC-SCRATCHJ)  
 Read/Write ..... 5-55

**Chapter 6**

**Instruction Set of the I/O Processor**

SCSI SCRIPTS ..... 6-1  
 Sample Operation ..... 6-2

Block Move Instructions ..... 6-4  
 First Dword ..... 6-4  
 Second Dword ..... 6-7

I/O Instructions ..... 6-8  
 First Dword ..... 6-8  
 Second Dword ..... 6-12

Read/Write Instructions ..... 6-13  
 First Dword ..... 6-13  
 Second Dword ..... 6-14  
 Read-Modify-Write Cycles ..... 6-14  
 Move to/from SFBR Cycles ..... 6-14

Transfer Control Instructions .....	6-16
First Dword .....	6-16
Second Dword .....	6-20
Memory Move Instructions .....	6-20
Read/Write System Memory from a Script .....	6-21
Second Dword .....	6-21
Third Dword .....	6-21
Load and Store Instructions .....	6-23
First Dword .....	6-23
Second Dword .....	6-24
<b>Chapter 7</b>	
<b>Electrical Characteristics</b>	
DC Characteristics .....	7-1
TolerANT Technology Electrical Characteristics .....	7-6
AC Characteristics .....	7-9
PCI and External Memory Interface Timing Diagrams .....	7-11
Timing diagrams included in this section .....	7-11
Target Cycles .....	7-11
Initiator Cycles .....	7-11
External Memory Cycles .....	7-11
PCI and External Memory Interface Timings .....	7-34
SCSI Timings .....	7-36
<b>Appendix A</b>	
<b>Register Summary</b>	
<b>Appendix B</b>	
<b>Mechanical Drawings</b>	
<b>Appendix C</b>	
<b>External Memory Interface Diagram Examples</b>	
Index .....	Index-1



---

# List of Figures

Figure 1-1: SYM53C875 External Memory Interface .....	1-6
Figure 1-2: SYM53C875 Chip Block Diagram .....	1-6
Figure 2-1: DMA FIFO Sections .....	2-10
Figure 2-2: SYM53C875 Host Interface Data Paths .....	2-12
Figure 2-3: SYM53C875 Differential Wiring Diagram .....	2-15
Figure 2-4: Regulated Termination .....	2-17
Figure 2-5: Determining the Synchronous Transfer Rate .....	2-20
Figure 2-6: Block Move and Chained Block Move Instructions .....	2-26
Figure 3-1: PCI Configuration Register Map .....	3-7
Figure 3-2: Command Register Layout .....	3-9
Figure 3-3: Status Register Layout .....	3-11
Figure 4-1: SYM53C875 Pin Diagram .....	4-1
Figure 4-2: SYM53C875J Pin Diagram .....	4-2
Figure 4-3: SYM53C875N Pin Diagram .....	4-3
Figure 4-4: SYM53C875JB Pin Diagram (Top View) .....	4-4
Figure 4-5: SYM53C875 Functional Signal Grouping .....	4-7
Figure 5-1: SYM53C875 Register Address Map .....	5-4
Figure 6-1: SCRIPTS Overview .....	6-3
Figure 6-2: Block Move Instruction Register .....	6-5
Figure 6-3: I/O Instruction Register .....	6-9
Figure 6-4: Read/Write Instruction Register .....	6-13
Figure 6-5: Transfer Control Instruction .....	6-17
Figure 6-6: Memory Move Instructions .....	6-22
Figure 6-7: Load and Store Instruction Format .....	6-24
Figure 7-1: Rise and Fall Time Test Conditions .....	7-7
Figure 7-2: SCSI Input Filtering .....	7-7
Figure 7-3: Hysteresis of SCSI Receiver .....	7-8
Figure 7-4: Input Current as a Function of Input Voltage .....	7-8
Figure 7-5: Output Current as a Function of Output Voltage .....	7-8
Figure 7-6: Clock Timing .....	7-9
Figure 7-7: Reset Input .....	7-10

Figure 7-8: Interrupt Output	7-10
Figure 7-9: PCI Configuration Register Read	7-12
Figure 7-10: PCI Configuration Register Write	7-13
Figure 7-11: Operating Register/SCRIPTS RAM read	7-14
Figure 7-12: Operating Register/SCRIPTS RAM write	7-15
Figure 7-13: External Memory Read	7-16
Figure 7-14: External Memory Write	7-17
Figure 7-15: Op Code Fetch, non-burst	7-18
Figure 7-16: Burst Op Code Fetch	7-19
Figure 7-17: Back to Back Read	7-20
Figure 7-18: Back to Back Write	7-21
Figure 7-19: Burst Read	7-22
Figure 7-20: Burst Write	7-23
Figure 7-21: Read Cycle, Normal/Fast Memory (>64 KB), single-byte access	7-24
Figure 7-22: Write Cycle, Normal/Fast Memory (>64 KB), single-byte access	7-25
Figure 7-23: Read Cycle, Normal/Fast Memory ( $\geq 128$ KB), multiple-byte access	7-26
Figure 7-24: Write Cycle, Normal/Fast Memory ( $\geq 128$ KB), multiple-byte access	7-27
Figure 7-25: Read Cycle, Slow Memory (>64 KB)	7-28
Figure 7-26: Write Cycle, Slow Memory (>64 KB)	7-29
Figure 7-27: Read Cycle, Normal/Fast Memory ( $\leq 64$ K)	7-30
Figure 7-28: Write Cycle, Normal/Fast Memory ( $\leq 64$ K)	7-31
Figure 7-29: Read Cycle, Slow Memory ( $\leq 64$ K)	7-32
Figure 7-30: Write Cycle, Slow Memory ( $\leq 64$ K)	7-33
Figure 7-31: Initiator Asynchronous Send	7-36
Figure 7-32: Initiator Asynchronous Receive	7-36
Figure 7-33: Target Asynchronous Send	7-37
Figure 7-34: Target Asynchronous Receive	7-37
Figure 7-35: Initiator and Target Synchronous Transfers	7-38
Figure B-1: SYM53C875, SYM 53C875J Mechanical Drawing	B-1
Figure B-2: SYM53C875N Mechanical Drawing	B-2
Figure B-3: SYM53C875JB (169-pin BGA) Mechanical Drawing	B-3
Figure C-1: 16 K Interface With 200 ns Memory	C-1
Figure C-2: 64 K Interface with 150 ns Memory	C-2
Figure C-3: 256 K Interface With 150 ns Memory	C-3
Figure C-4: 512 K Interface With 150 ns Memory	C-4



---

## List of Tables

Table 2-1: External Memory Support . . . . .	2-4
Table 2-2: Bits Used for Parity Control and Generation . . . . .	2-8
Table 2-3: SCSI Parity Control . . . . .	2-9
Table 2-4: SCSI Parity Errors and Interrupts . . . . .	2-9
Table 3-1: PCI Bus Commands and Encoding Types . . . . .	3-2
Table 4-1: 53C875, 53C875J Power and Ground Pins . . . . .	4-5
Table 4-2: 53C875N Power and Ground Pins . . . . .	4-6
Table 4-3: 53C875JB Power and Ground Pins . . . . .	4-6
Table 4-4: System Pins . . . . .	4-8
Table 4-5: Address and Data Pins . . . . .	4-9
Table 4-6: Interface Control Pins . . . . .	4-11
Table 4-7: Arbitration Pins . . . . .	4-12
Table 4-8: Error Reporting Pins . . . . .	4-12
Table 4-9: SCSI Pins . . . . .	4-13
Table 4-10: Additional Interface Pins . . . . .	4-15
Table 4-11: External Memory Interface Pins . . . . .	4-17
Table 4-12: JTAG Pins (53C875J/53C875N/53C875JB Only) . . . . .	4-18
Table 5-1: Operating Register Addresses and Descriptions . . . . .	5-1
Table 5-2: Examples of Synchronous Transfer Periods for SCSI-1 Transfer Rates . . . . .	5-13
Table 5-3: Example Transfer periods for fast SCSI-2 and Fast-20 transfer rates . . . . .	5-13
Table 6-1: Read/Write Instructions . . . . .	6-15
Table 7-1: Absolute Maximum Stress Ratings . . . . .	7-1
Table 7-2: Operating Conditions . . . . .	7-1
Table 7-3: SCSI Signals - SD(15-0)/, SDP(1-0)/, SREQ/ SACK/ . . . . .	7-2
Table 7-4: SCSI Signals - SMSG, SI_O/, SC_D/, SATN/, SBSY/, SSEL/, SRST/ . . . . .	7-2
Table 7-5: Input Signals - CLK, SCLK, GNT/, IDSEL, RST/, TESTIN, DIFFSENS, BIG_LIT/ . . . . .	7-2
Table 7-6: Capacitance . . . . .	7-3
Table 7-7: Output Signal - MAC/_TESTOUT, REQ/ . . . . .	7-3
Table 7-8: Output Signals - IRQ/, SDIR(15-0), SDIRP0, SDIRP1, BSYDIR, SELDIR, RSTDIR, TGS, IGS, MAS/(1-0), MCE/, MOE/, MWE/ . . . . .	7-3

Table 7-9: Output Signal - SERR/ .....	7-3
Table 7-10: Bidirectional Signals - AD(31-0), C_BE/(3-0), FRAME/, IRDY/, TRDY/, DEVSEL/, STOP/, PERR/, PAR, BYTEPAR3-0 (53C875N only).....	7-4
Table 7-11: Bidirectional Signals - GPIO0_FETCH/, GPIO1_MASTER/, GPIO2_MAS2/, GPIO3, GPIO4 .....	7-4
Table 7-12: Bidirectional Signals - MAD(7-0) .....	7-5
Table 7-13: Input Signals —TDI, TMS, TCK (53C875J,53C875JB, 53C875N only) .....	7-5
Table 7-14: Output Signal — TDO (53C875, 53C875JB, 53C875N only) .....	7-5
Table 7-15: TolerANT Technology Electrical Characteristics .....	7-6
Table 7-16: SYM53C875 PCI and External Memory Interface Timings .....	7-34
Table 7-17: SCSI-1 transfers (Single-Ended, 5.0 MB/s).....	7-38
Table 7-18: SCSI-1 Transfers (Differential, 4.17 MB/s).....	7-39
Table 7-19: SCSI-2 Fast Transfers (10.0 MB/s (8-bit transfers) or 20.0 MB/s (16-bit transfers), 40 MHz clock) .....	7-39
Table 7-20: SCSI-2 Fast Transfers (10.0 MB/s (8-bit transfers) or 20.0 MB/s (16-bit transfers), 50 MHz clock) .....	7-40
Table 7-21: UltraSCSI Single-Ended Transfers (20.0 MB/s (8-bit transfers) or 40.0 MB/s (16-bit transfers), 80 MHz clock) .....	7-40
Table 7-22: UltraSCSI Differential Transfers (20.0 MB/s (8-bit transfers) or 40.0 MB/s (16-bit transfers), 80 MHz clock) .....	7-41

## Chapter 1 Introduction

---

### What is Covered in This Manual

This manual provides reference information on the SYM53C875 PCI-SCSI I/O Processor. It is intended for system designers and programmers who are using this device to design a SCSI port for PCI-based personal computers, workstations, or embedded applications.

This chapter includes general information about the SYM53C875 and other members of the SYM53C8XX family of PCI-SCSI I/O Processors. Chapter 2 describes the main functional areas of the chip in more detail, including the interfaces to the SCSI bus. Chapter 3 describes the chip's connection to the PCI bus, including the PCI commands and configuration registers supported. Chapter 4 contains the pin diagrams and definitions of each signal. Chapter 5 describes each bit in the operating registers, organized by address. Chapter 6 defines all of the SCSI SCRIPTS instructions that are supported by the SYM53C875. Chapter 7 contains the electrical characteristics and AC timings for the chip. The appendixes contain a register summary, a mechanical drawing of the SYM53C875, and several example interface drawings to connect the SYM53C875 to an external ROM.

This data manual assumes the user is familiar with the current and proposed standards for SCSI and PCI. For additional background information on these topics, please refer to the list of reference materials provided in the Preface of this document.

---

### General Description

The SYM53C875 PCI-SCSI I/O Processor brings high-performance I/O solutions to host adapter, workstation, and general computer designs, making it easy to add SCSI to any PCI system. It provides a local memory bus for local storage of the device's BIOS ROM in flash memory or standard EPROMs. Most versions of the SYM53C875 also support Big and Little Endian byte addressing to accommodate a variety of data configurations. The SYM53C875 supports programming of local FLASH memory for updates to BIOS or SCRIPTS programs.

The SYM53C875 is a pin-for-pin replacement for the SYM53C825 PCI-SCSI I/O processor, with added support for the SCSI-3 UltraSCSI standard as well as other new features. Some software enhancements are needed to take advantage of the new features and UltraSCSI transfer rates supported by the SYM53C875. The SYM53C875 performs UltraSCSI transfers or fast 8- or 16-bit SCSI transfers in single-ended or differential mode, and improves performance by optimizing PCI bus utilization. A system diagram showing the connections of the SYM53C875 with an external ROM or flash memory is pictured in Figure 1-1. A block diagram of the SYM53C875 is pictured in Figure 1-2.

The SYM53C875 integrates a high-performance SCSI core, a PCI bus master DMA core, and the Symbios Logic SCSI SCRIPTS™ processor to meet the flexibility requirements of SCSI-3 and UltraSCSI standards. It is designed to implement multi-threaded I/O algorithms with a minimum of

processor intervention, solving the protocol overhead problems of previous intelligent and non-intelligent adapter designs.

The SYM53C875 is fully supported by the Symbios SCSI Device Management System (SDMS™), a software package that supports the Advanced SCSI Protocol Interface (ASPI) and the ANSI Common Access Method (CAM). SDMS provides BIOS and driver support for hard disk, tape, removable media products, and CD-ROM under the major PC operating systems.

---

## Package and Feature Options

The SYM53C875 is available in three versions with different packaging and feature options. The SYM53C875 is packaged in a 160-pin plastic quad flat pack. The SYM53C875J is identical to the SYM53C875 with additional pins that support JTAG boundary scan testing. The JTAG boundary scan signals replace the TESTIN, MAC/\_TESTOUT, BIG\_LIT/, and SDIRP1 pins. The SYM53C875N includes all of the signals in the SYM53C875, with the additions of the JTAG pins and four additional signals for extended parity checking and generation. The SYM53C875JB is identical to the SYM53C875J, but is packaged in a 169-pin ball grid array.

---

## Benefits of UltraSCSI

UltraSCSI is an extension of the SCSI-3 standard that expands the bandwidth of the SCSI bus and allows faster synchronous SCSI transfer rates. When enabled, UltraSCSI performs 20 megatransfers during an I/O operation, which results in approximately doubling the synchronous transfer rates of fast SCSI-2. The SYM53C875 can perform 8-bit, UltraSCSI synchronous transfers as fast as 20 MB/s. This advantage is most noticeable in heavily loaded systems, or large-block size applications such as video on-demand and image processing.

One advantage of UltraSCSI is that it significantly improves SCSI bandwidth while preserving existing hardware and software investments. The SYM53C875 is compatible with all existing SYM53C825 and SYM53C825A software; the only changes required are to enable the chip to perform synchronous negotiations for UltraSCSI rates. The SYM53C875 can use the same board socket as an SYM53C825, with the addition of an 80MHz SCLK or enabling the internal SCSI clock doubler to provide the correct frequency when transferring synchronous SCSI data at 50 nanosecond transfer rates. Some changes to existing cabling or system designs may be needed to maintain signal integrity at UltraSCSI synchronous transfer rates. These design issues are discussed in Chapter 2.

---

## TolerANT Technology

The SYM53C875 features TolerANT® technology, which includes active negation on the SCSI drivers and input signal filtering on the SCSI receivers. Active negation causes the SCSI Request, Acknowledge, Data, and Parity signals to be actively driven high rather than passively pulled up by terminators. Active negation is enabled by setting bit 7 in the STEST3 register.

TolerANT receiver technology improves data integrity in unreliable cabling environments, where other devices would be subject to data corruption. TolerANT receivers filter the SCSI bus signals to eliminate unwanted transitions, without the long signal delay associated with RC-type input filters. This improved driver and receiver technology helps eliminate double clocking of data, the single biggest reliability issue with SCSI operations. TolerANT input signal filtering is a built in feature of the SYM53C875 and all Symbios Logic fast SCSI devices. On the SYM53C875, the user may select a filtering period of 30 or 60 ns, with bit 1 in the STEST2 register.

The benefits of TolerANT include increased immunity to noise when the signal is going high, better performance due to balanced duty cycles, and improved fast SCSI transfer rates. In addition, TolerANT SCSI devices do not cause glitches on the SCSI bus at power up or power down, so other devices on the bus are also protected from data corruption. TolerANT is compatible with both the Alternative One and Alternative Two termination schemes proposed by the American National Standards Institute.

---

## SYM53C875 Benefits Summary

---

### SCSI Performance

- Includes 4KB internal RAM for SCRIPTS instruction storage
- Performs wide, UltraSCSI synchronous transfers as fast as 40 MB/s
- SCSI synchronous offset increased from 8 to 16 levels
- Supports variable block size and scatter/gather data transfers.
- Performs sustained memory-to-memory DMA transfers faster than 47 MB/s (@ 33 MHz)
- Minimizes SCSI I/O start latency
- Performs complex bus sequences without interrupts, including restore data pointers
- Reduces ISR overhead through a unique interrupt status reporting method
- Performs fast and wide SCSI bus transfers in single-ended and differential mode
  - 10 MB/s asynchronous (20 MB/s with UltraSCSI)
  - 20 MB/s synchronous (40 MB/s with UltraSCSI)
- Load and Store SCRIPTS instruction increases performance of data transfers to and from chip registers
- Supports target disconnect and later reconnect with no interrupt to the system processor
- Supports multi-threaded I/O algorithms in SCSI SCRIPTS with fast I/O context switching
- Expanded Register Move instruction supports additional arithmetic capability

---

## PCI Performance

- Complies with PCI 2.1 specification
- Bursts 2, 4, 8, 16, 32, 64, or 128 dwords across PCI bus
- Supports 32-bit word data bursts with variable burst lengths.
- Pre-fetches up to 8 dwords of SCRIPTS instructions
- Bursts SCRIPTS op code fetches across the PCI bus
- Performs zero wait-state bus master data bursts faster than 110 MB/s (@ 33 MHz)
- Supports PCI Cache Line Size register
- Supports PCI Write and Invalidate, Read Line, and Read Multiple commands

---

## Integration

- 3.3V/5V PCI interface
- Full 32-bit PCI DMA bus master
- Can be used as a third-party PCI bus DMA controller by using Memory to Memory Move instructions
- High performance SCSI core
- Integrated SCRIPTS processor

---

## Ease of Use

- Up to one megabyte of add-in memory support for BIOS and SCRIPTS storage
- Direct PCI-to-SCSI connection
- Reduced SCSI development effort
- Easily adapted to the Advanced SCSI Protocol Interface (ASPI) or the ANSI Common Access Method (CAM), with SDMS software.
- Compiler-compatible with existing SYM53C7XX and SYM53C8XX family SCRIPTS
- Direct connection to PCI, and SCSI single-ended and differential buses
- Development tools and sample SCSI SCRIPTS available
- Maskable and pollable interrupts
- Wide SCSI, A or P cable, and up to 16 devices supported
- Three programmable SCSI timers: Select/Reselect, Handshake-to-Handshake, and General Purpose. The time-out period is programmable from 100  $\mu$ s to greater than 25.6 seconds
- SDMS software for complete PC-based operating system support
- Support for relative jumps
- SCSI Selected As ID bits for responding with multiple IDs



---

## Flexibility

- High level programming interface (SCSI SCRIPTS)
- Programs local memory bus FLASH memory
- Big/Little Endian support
- Selectable 88- or 536-byte DMA FIFO for backward compatibility
- Tailored SCSI sequences execute from main system RAM or internal SCRIPTS RAM
- Flexible programming interface to tune I/O performance or to adapt to unique SCSI devices
- Support for changes in the logical I/O interface definition
- Low level access to all registers and all SCSI bus signals
- Fetch, Master, and Memory Access control pins
- Separate SCSI and system clocks
- SCSI clock doubler bits enable UltraSCSI transfer rates with a 40 MHz SCSI clock
- Selectable IRQ pin disable bit
- 32 additional scratch pad registers
- Ability to route system clock to SCSI clock
- Latch-up protection greater than 150 mA
- Voltage feed through protection (minimum leakage current through SCSI pads)
- 25% of pins are power and ground
- Power and ground isolation of I/O pads and internal chip logic
- TolerANT technology provides:
  - Active negation of SCSI Data, Parity, Request, and Acknowledge signals for improved fast SCSI transfer rates.
  - Input signal filtering on SCSI receivers improves data integrity, even in noisy cabling environments.
- JTAG Boundary scan support (SYM, 53C875J, 53C875JB, 53C875N only)
- Extended PCI parity checking and generation (SYM53C875N only)
- Extended SCSI parity checking

---

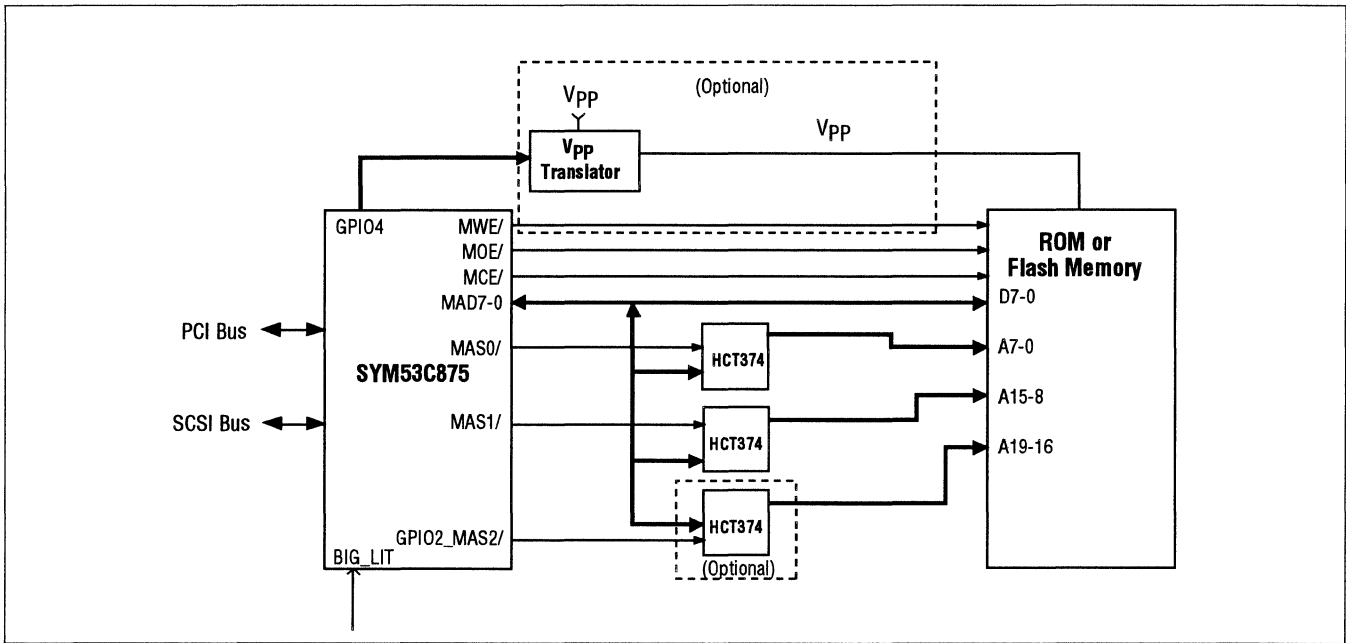
## Testability

- All SCSI signals accessible through programmed I/O
- SCSI loopback diagnostics
- SCSI bus signal continuity checking
- Support for single-step mode operation
- Test mode (AND tree) to check pin continuity to the board (most package options)
- JTAG Boundary scan support (SYM, 53C875J, 53C875JB, 53C875N only)

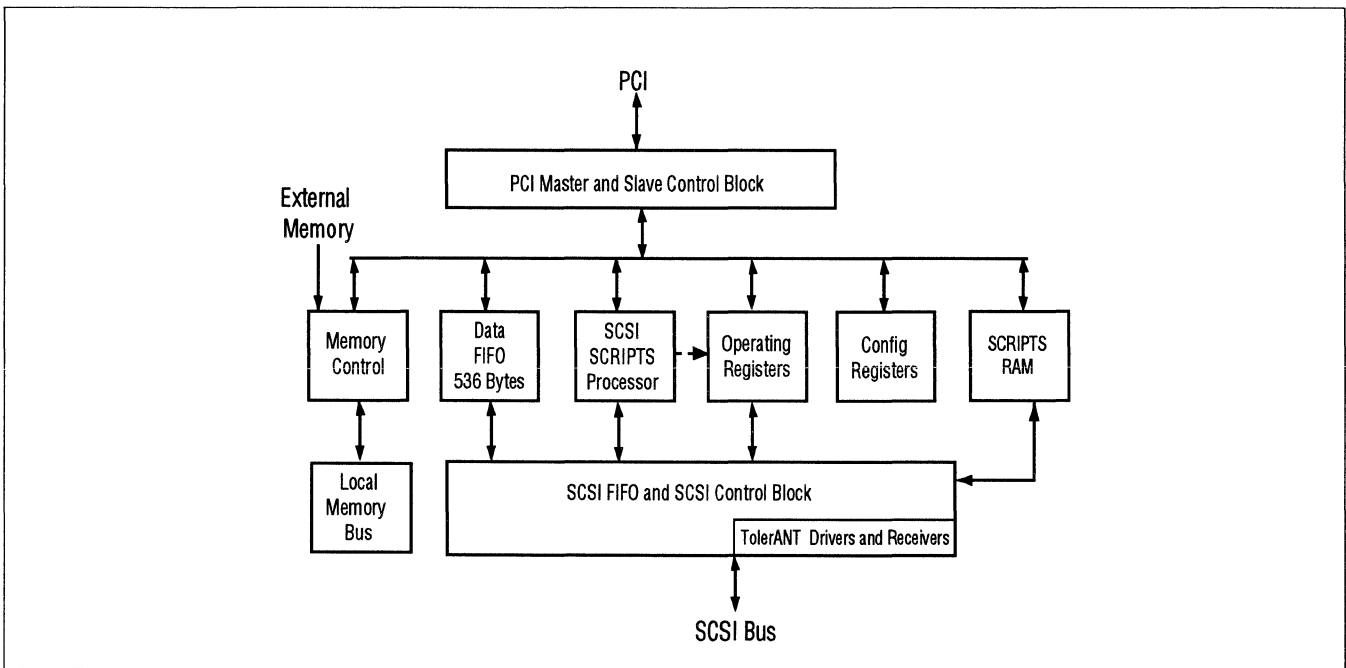
---

## Reliability

- 2 KV ESD protection on SCSI signals
- Typical 300 mV SCSI bus hysteresis
- Protection against bus reflections due to impedance mismatches
- Controlled bus assertion times (reduces RFI, improves reliability, and eases FCC certification)



**Figure 1-1: SYM53C875 External Memory Interface**



**Figure 1-2: SYM53C875 Chip Block Diagram**

---

## Chapter 2

# Functional Description

The SYM53C875 is composed of three functional blocks: the SCSI Core, the DMA Core, and the SCRIPTS Processor. The SYM53C875 is fully supported by the SCSI Device Management System (SDMS), a complete software package that supports the Symbios Logic product line of SCSI processors and controllers.

---

## SCSI Core

The SCSI core supports the 8- or 16-bit data bus. It supports UltraSCSI synchronous transfer rates up to 40 MB/s, SCSI synchronous transfer rates up to 20 MB/s, and asynchronous transfer rates up to 10 MB/s on a 16-bit wide SCSI bus. The SCSI core can be programmed with SCSI SCRIPTS, making it easy to “fine tune” the system for specific mass storage devices or SCSI-3 requirements.

The SCSI core offers low-level register access or a high-level control interface. Like first generation SCSI devices, the SYM53C875 SCSI core can be accessed as a register-oriented device. The ability to sample and/or assert any signal on the SCSI bus can be used in error recovery and diagnostic procedures. In support of loopback diagnostics, the SCSI core may perform a self-selection and operate as both an initiator and a target.

The SYM53C875 SCSI core is controlled by the integrated SCRIPTS processor through a high-level logical interface. Commands controlling the SCSI core are fetched out of the main host memory or local memory. These commands instruct the SCSI core to Select, Reselect, Disconnect, Wait for a Disconnect, Transfer Information, Change Bus Phases and, in general, implement all aspects of the SCSI protocol. The SCRIPTS processor is a special high-speed processor optimized for SCSI protocol.

---

## DMA Core

The DMA core is a bus master DMA device that attaches directly to the industry standard PCI Bus. The DMA core is tightly coupled to the SCSI core through the SCRIPTS processor, which supports uninterrupted scatter/gather memory operations.

The SYM53C875 supports 32-bit memory and automatically supports misaligned DMA transfers. A 536-byte FIFO allows the SYM53C875 to support 2, 4, 8, 16, 32, 64, or 128 longword bursts across the PCI bus interface.

---

## SCRIPTS Processor

The SCSI SCRIPTS processor allows both DMA and SCSI commands to be fetched from host memory or internal SCRIPTS RAM. Algorithms written in SCSI SCRIPTS control the actions of the SCSI and DMA cores and are executed from 32-bit system RAM. The SCRIPTS processor executes complex SCSI bus sequences independently of the host CPU.

The SCRIPTS processor can begin a SCSI I/O operation in approximately 500 ns. This compares with 2-8 ms required for traditional intelligent host adapters. Algorithms may be designed to tune SCSI bus performance, to adjust to new bus device types (such as scanners, communication gateways, etc.), or to incorporate changes in the SCSI-2 or SCSI-3 logical bus definitions without sacrificing I/O performance. SCSI SCRIPTS are hardware-independent, so they can be used interchangeably on any host or CPU system bus.

---

## Internal SCRIPTS RAM

The SYM53C875 has 4 KB (1024 x 32 bits) of internal, general purpose RAM. The RAM is designed for SCRIPTS program storage, but is not limited to this type of information. When the chip fetches SCRIPTS instructions or Table Indirect information from the internal RAM, these fetches remain internal to the chip and do not use the PCI bus. Other types of access to the RAM by the SYM53C875 use the PCI bus, as if they were external accesses. The MAD5 pin enables the 4K internal RAM. To disable the internal RAM, connect a 4.7K  $\Omega$  resistor between the MAD5 pin and  $V_{SS}$ .

The RAM can be relocated by the PCI system BIOS anywhere in 32-bit address space. The RAM Base Address register in PCI configuration space contains the base address of the internal RAM. This register is similar to the ROM Base Address register in PCI configuration space. To simplify loading of SCRIPTS instructions, the base address of the RAM will appear in the SCRATCHB register when bit 3 of the CTEST2 register is set. The RAM is byte-accessible from the PCI bus and will be visible to any bus-mastering device on the bus. External accesses to the RAM (i.e., by the CPU) follow the same timing sequence as a standard slave register access, except that the target wait states required will drop from 5 to 3.

A complete set of development tools is available for writing custom drivers with SCSI SCRIPTS. For more information on the SCSI SCRIPTS instructions supported by the SYM53C875, see Chapter 6.

---

## SDMS: The Total SCSI Solution

For users who do not need to develop custom drivers, Symbios Logic provides a total SCSI solution in PC environments with the SCSI Device Management System (SDMS). SDMS provides BIOS driver support for hard disk, tape, and removable media peripherals for the major PC-based operating systems.

SDMS includes a SCSI BIOS to manage all SCSI functions related to the device. It also provides a series of SCSI device drivers that support most major operating systems. SDMS supports a multi-threaded I/O application programming interface (API) for user-developed SCSI applications. SDMS supports both the ASPI and CAM SCSI software specifications.

---

## Designing an UltraSCSI System

Migrating an existing single-ended SCSI design from SCSI-2 to UltraSCSI requires minor software modifications as well as consideration for some hardware design guidelines. Since UltraSCSI is based on existing SCSI standards, it can use existing software programs as long as the software is able to negotiate for UltraSCSI synchronous transfer rates.

In the area of hardware, the primary area of concern in single-ended systems is to maintain signal integrity at high data transfer rates. To assure reliable operation at UltraSCSI transfer speeds, follow the system design parameters recommended in the SCSI-3 Fast-20 Parallel Interface standard, which is available from the SCSI BBS referenced at the beginning of this manual. Chapter 7 contains UltraSCSI timing information. In addition to the

guidelines in the draft standard, make the following software and hardware adjustments to accommodate UltraSCSI transfers:

- Set the Fast-20 Enable bit to enable UltraSCSI transfers.
- Set the TolerANT Enable bit, bit 7 in the STEST3 register whenever the Fast-20 Enable bit is set.
- Do not extend the SREQ/SACK filtering period with STEST2 bit 1.
- Use an 80 MHz SCSI clock or enable the SCSI clock doubler using bits 2 and 3 of the STEST1 register. Set the Halt SCSI clock (HSC) bit in STEST3 before switching to the doubled SCSI clock.

---

### Using the SCSI Clock Doubler

The SYM53C875 can double the frequency of a 40-50 MHz SCSI clock, allowing the system to perform UltraSCSI transfers in systems that do not have 80 MHz clock input. This option is user-selectable with bit settings in the STEST1, STEST3, and SCNTL3 registers. At power-on or reset, the doubler is disabled and powered down. Follow these steps to use the clock doubler:

1. Set the SCLK Doubler Enable bit (STEST1, bit 3)
2. Wait 20  $\mu$ s
3. Halt the SCSI clock by setting the Halt SCSI Clock bit (STEST3 bit 5)
4. Set the clock conversion factor using the SCF and CCF fields in the SCNTL3 register
5. Set the SCLK Doubler Select bit (STEST1, bit 2)
6. Clear the Halt SCSI Clock bit

---

## Prefetching SCRIPTS Instructions

When enabled (by setting the Prefetch Enable bit in the DCNTL register), the prefetch logic in the SYM53C875 fetches 8 dwords of instructions. The prefetch logic automatically determines the maximum burst size that it can perform, based on the burst length as determined by the values in the DMODE register. If the unit cannot perform bursts of at least four dwords, it will disable itself. While the SYM53C875 is prefetching SCRIPTS instructions, the PCI Cache Line Size register value does not have any effect and the Read Line, Read Multiple, and Write and Invalidate commands will not be used.

The SYM53C875 may flush the contents of the prefetch unit under certain conditions, listed below, to ensure that the chip always operates from the most current version of the software. When one of these conditions apply, the contents of the prefetch unit are flushed automatically.

1. On every Memory Move instruction. The Memory Move instruction is often used to place modified code directly into memory. To make sure that the chip executes all recent modifications, the prefetch unit flushes its contents and loads the modified code every time a instruction is issued. To avoid inadvertently flushing the prefetch unit contents, use the No Flush option for all Memory Move operations that do not modify code within the next 8 dwords. For more information on this instruction, refer to Chapter 6.
2. On every Store instruction. The Store instruction may also be used to place modified code directly into memory. To avoid inadvertently flushing the prefetch unit

contents, use the No Flush option for all Store operations that do not modify code within the next 8 dwords.

3. On every write to the DSP.
4. On all Transfer Control instructions when the transfer conditions are met. This is necessary because the next instruction to be executed is not the sequential next instruction in the prefetch unit.
5. When the Pre-Fetch Flush bit (DCNTL bit 6) is set. The unit flushes whenever this bit is set. The bit is self-clearing.

---

### Op Code Fetch Burst Capability

Setting the Burst Op Code Fetch Enable bit in the DMODE register (38h) causes the SYM53C875 to burst in the first two longwords of all instruction fetches. If the instruction is a memory-to-memory move, the third longword will be accessed in a separate ownership. If the instruction is an indirect type, the additional longword will be accessed in a subsequent bus ownership. If the instruction is a table indirect Block Move, the SYM53C875 will use two accesses to obtain the four longwords required, in two bursts of two longwords each.

Note: this feature is only useful if pre-fetching is disabled.

---

## External Memory Interface

The SYM53C875 supports up to one megabyte of external memory in binary increments from 16 KB, to allow the use of expansion ROM for add-in PCI cards. The device also supports flash ROM updates through the add-in interface and the GPIO4 pin (used to control  $V_{PP}$ , the power supply for programming external memory). This interface is designed for low-speed operations such as downloading instruction code from ROM; it is not intended for dynamic activities such as executing instructions.

System requirements include the SYM53C875, two or three external 8-bit address holding registers (HCT273 or HCT374), and the appropriate memory device. The 4.7 K $\Omega$  pull-down resistors on the MAD bus require HC or HCT external components to be used. If in-system flash ROM updates are required, a 7406 (high voltage open collector inverter), an MTD4P05, and several passive components are also needed. The memory size and speed is determined by pull-down resistors on the 8-bit bidirectional memory bus at power up. The SYM53C875 senses this bus shortly after the release of the Reset signal and configures the ROM Base Address register and the memory cycle state machines for the appropriate conditions.

The external memory interface works with a variety of ROM sizes and speeds. An example set of interface drawings is in Appendix C.

The SYM53C875 supports a variety of sizes and speeds of expansion ROM, using pull-down resistors on the MAD(3-0) pins. The encoding of pins MAD(3-1) allows the user to define how much external memory is available to the SYM53C875. Table 2-1 shows the memory space associated with the possible values of MAD(3-1). The MAD(3-1) pins are fully defined in Chapter 4, "Signal Descriptions."

**Table 2-1: External Memory Support**

MAD(3-1)	Available Memory Space
000	16 KB
001	32 KB
010	64 KB
011	128 KB
100	256 KB
101	512 KB
110	1024 KB
111	no external memory present



To use one of the configurations mentioned above in a host adapter board design, put 4.7 K $\Omega$  pull-down resistors on the MAD pins corresponding to the available memory space. For example, to connect to a 32 KB external ROM, use pull-downs on MAD(3) and MAD(2). If the external memory interface is not used, then no external resistors are necessary since there are internal pull-ups on the MAD bus. The internal pull-up resistors are disabled when external pull-down resistors are detected, to reduce current drain.

The SYM53C875 allows the system to determine the size of the available external memory using the Expansion ROM Base Address register in PCI configuration space. For more information on how this works, refer to the PCI specification or the Expansion ROM Base Address register description in Chapter 3.

MAD(0) is the slow ROM pin. When pulled down, it enables two extra clock cycles of data access time to allow use of slower memory devices. The external memory interface also supports updates to flash memory. The 12 volt power supply for flash memory,  $V_{PP}$ , is enabled and disabled with the GPIO4 pin and the GPIO4 control bit. For more information on the GPIO4 pin, refer to Chapter 4.

---

## PCI Cache Mode

The SYM53C875 supports the PCI specification for an 8-bit Cache Line Size register located in PCI configuration space. The Cache Line Size register provides the ability to sense and react to non-aligned addresses corresponding to cache line boundaries. In conjunction with the Cache Line Size register, the PCI commands Read Line, Read Multiple, and Write and Invalidate are each software enabled or disabled to allow the user full flexibility in using these commands. For more information on PCI cache mode operations, refer to Chapter 3.

---

## Load/Store Instructions

The SYM53C875 supports the Load/Store instruction type, which simplifies the movement of data between memory and the internal chip registers. It also enables the SYM53C875 to transfer bytes to addresses relative to the DSA register. For more information on the Load and Store instructions, refer to Chapter 6.

---

## 3.3 Volt/5 Volt PCI Interface

The SYM53C875 can attach directly to a 3.3 Volt or a 5 Volt PCI interface, due to separate  $V_{DD}$  pins for the PCI bus drivers. This allows the devices to be used on the universal board recommended by the PCI Special Interest Group.

---

## Additional Access to General Purpose Pins

The SYM53C875 can access the GPIO0 and GPIO1 general purpose pins through register bits in the PCI configuration space, instead of using the GPCNTL register in the operating register space to control these pins. In the Symbios Logic SDMS software, the configuration bits control pins as the clock and data lines, respectively.

To access the GPIO1-0 pins through the configuration space, connect a 4.7 K $\Omega$  resistor between the MAD(7) pin and  $V_{SS}$ . MAD(7) contains an internal pull-up that is sensed shortly after chip reset. If the pin is sensed high, GPIO1-0 access is disabled; if it is low, GPIO1-0 access is enabled. Additionally, if GPIO1-0 access has been enabled through the MAD(7) pin and if GPIO0 and/or GPIO1 are sensed low after chip reset, GPIO1-0 access will be disabled. If GPIO1-0 access through configuration space is enabled, the GPIO0 and GPIO1 pins cannot be controlled from the GPCNTL and GPREG registers, but will be observable from the GPREG register. When GPIO1-0 access is enabled, the Serial Interface Control register at configuration addresses 34-35h controls the GPIO0 and GPIO1 pins. For more

information on GPIO1-0 access, refer to the Serial Interface Control register description in Chapter 3. For more information on the GPIO1-0 pins, see Chapter 4.

---

## JTAG Boundary Scan Testing

The SYM 53C875J/53C875N/53C875JB include support for JTAG boundary scan testing in accordance with the IEEE 1149.1 specification, with one exception that is discussed in this section. The device can accept all required boundary scan instructions, as well as the optional CLAMP, HIGHZ, and IDCODE instructions.

The SYM 53C875J/53C875N/53C875JB use an 8-bit instruction register to support all boundary scan instructions. The data registers included in the device are the Boundary Data register, the IDCODE register, and the Bypass register. The device can handle a 10 MHz TCK frequency for TDO and TDI.

Due to design constraints, the RST/ pin (System Reset) will always tri-state the SCSI pins when it is asserted. This action cannot be controlled by the boundary scan logic, and thus is not compliant with the specification. There are two solutions that resolve this issue:

1. Use the RST/ pin as a boundary scan compliance pin. When the pin is deasserted, the device is boundary scan compliant and when asserted, the device is non-compliant. To maintain compliance, the RST/ pin must be driven high.
2. When RST/ is asserted during boundary scan testing, the expected output on the SCSI pins must be a high-z condition, and not what is contained in the boundary scan data registers for the SCSI pin output cells.

Because of package limitations, the SYM53C875J/53C875JB replaces the TESTIN, MAC/\_TESTOUT, BIG\_LIT/, and SDIRP1 signals with the JTAG boundary scan signals. The SYM53C875N includes support for these signals in addition to the JTAG pins.

---

## Big and Little Endian Support

The SYM53C875/53C875N supports both Big and Little Endian byte ordering through pin selection. The 53C875J/53C875JB operates in Little Endian mode only (the BIG\_LIT pin is replaced by one of the JTAG boundary scan signals). In Big Endian mode, the first byte of an aligned SCSI-to-PCI transfer will be routed to lane three and succeeding transfers will be routed to descending lanes. This mode of operation also applies to data transfers over the add-in ROM interface. The byte of data accessed at location 0000h from memory is routed to lane three, and the data at location 0003h is routed to byte lane 0. In Little Endian mode, the first byte of an aligned SCSI to PCI transfer will be routed to lane zero and succeeding transfers will be routed to ascending lanes. This mode of operation also applies to the add-in ROM interface. The byte of data accessed at location 0000h from memory is routed to lane zero, and the data at location 0003h is routed to byte lane 3.

The Big\_Lit pin gives the SYM53C875 the flexibility of operating with either Big or Little Endian byte orientation. Internally, in either mode, the actual byte lanes of the DMA FIFO and registers are not modified. The SYM53C875 supports slave accesses in Big or Little Endian mode.

When a dword is accessed, no repositioning of the individual bytes is necessary since dwords are addressed by the address of the least significant byte. SCRIPTS always uses dwords in 32-bit systems, so compatibility is maintained between systems using different byte orientations. When less than a dword is accessed, individual bytes must be repositioned. Internally, the SYM53C875 adjusts the byte control logic of the DMA FIFO and register decodes to access the appropriate byte lanes.

The registers will always appear on the same byte lane, but the address of the register will be repositioned.

Big/Little Endian mode selection has the most effect on individual byte access. Internally, the SYM53C875 adjusts the byte control logic of the DMA FIFO and register decodes to enable the appropriate byte lane. The registers will always appear on the same byte lane, but the address of the register will be repositioned.

Data to be transferred between system memory and the SCSI bus always starts at address zero and continues through address 'n' - there is no byte ordering in the chip. The first byte in from the SCSI bus goes to address 0, the second to address 1, etc. Going out onto the SCSI bus, address zero is the first byte out on the SCSI bus, address 1 is the second byte, etc. The only difference is that in a Little Endian system, address 0 will be on byte lane 0, and in Big Endian mode address zero will be on byte lane 3.

Correct SCRIPTS will be generated if the SCRIPTS compiler is run on a system that has the same byte ordering as the target system. Any SCRIPTS patching in memory must patch the instruction with the byte ordering that the SCRIPTS processor expects.

Software drivers for the SYM53C875 should access registers by their logical name (i.e., SCNTL0) rather than by their address. The logical name should be equated to the register's Big Endian address in Big Endian mode (SCNTL0 = 03h), and its Little Endian address in Little Endian Mode (SCNTL0 = 00h). This way, there is no change to the software when moving from one mode to the other; only the equate statement setting the operating modes needs to be changed.

Addressing of registers from within a SCRIPTS instruction is independent of bus mode. Internally, the SYM53C875 always operates in Little Endian mode.

---

## Loopback Mode

The SYM53C875 loopback mode allows testing of both initiator and target functions and, in effect, lets the chip communicate with itself. When the Loopback Enable bit is set in the STEST1 register, the SYM53C875 allows control of all SCSI signals, whether the SYM53C875 is operating in initiator or target mode. For more information on this mode of operation, refer to the *SYM53C8XX Family Programming Guide*.

---

## Parity Options

The SYM53C875 implements a flexible parity scheme that allows control of the parity sense, allows parity checking to be turned on or off, and has the ability to deliberately send a byte with bad parity over the SCSI bus to test parity error recovery procedures. Table 2-1 defines the bits that are involved in parity control and observation. Table 2-2 describes the parity control function of the Enable Parity Checking and Assert SCSI Even Parity bits in the SCNTL0 register. Table 2-3 describes the options available when a parity error occurs.

The SYM53C875N has four additional parity pins for checking incoming data on the PCI bus. These pins are assigned to each byte of the PCI address/data bus, and work in addition to the PAR (PCI parity) pin. In PCI master read or slave write operations, each byte of incoming data on the PCI bus is checked against its corresponding parity line, in addition to the normal parity checking against the PCI PAR signal. In PCI master write or slave read operations, parity is generated for each byte. This extra parity checking is always enabled for the SYM53C875N. The host system must support these pins. This feature is not register selectable. A parity error on any Byte Parity pin for PCI master read or slave write operations will cause a fatal DMA interrupt; SCRIPTS will stop running. This interrupt can be masked with the EBPE Interrupt Enable bit, bit 1 in the DIEN register. These additional parity pins in no way affect the generation or checking of the PCI specified parity line.

**Table 2-2: Bits Used for Parity Control and Generation**

<b>Bit Name</b>	<b>Location</b>	<b>Description</b>
Assert SATN/ on Parity Errors	SCNTL0, Bit 1	Causes the SYM53C875 to automatically assert SATN/ when it detects a parity error while operating as an initiator.
Enable Parity Checking	SCNTL0, Bit 3	Enables the SYM53C875 to check for parity errors. The SYM53C875 checks for odd parity. This bit also checks for parity errors on the four additional parity pins on the SYM53C875N.
Assert Even SCSI Parity	SCNTL1, Bit 2	Determines the SCSI parity sense generated by the SYM53C875 to the SCSI bus.
Disable Halt on SATN/ or a Parity Error (Target Mode Only)	SCNTL1, Bit 5	Causes the SYM53C875 not to halt operations when a parity error is detected in target mode.
Enable Parity Error Interrupt	SIEN0, Bit 0	Determines whether the SYM53C875 will generate an interrupt when it detects a SCSI parity error.
Parity Error	SIST0, Bit 0	This status bit is set whenever the SYM53C875 has detected a parity error on the SCSI bus.
Status of SCSI Parity Signal	SSTAT0, Bit 0	This status bit represents the active high current state of the SCSI SDP0 parity signal.
SCSI SDP1 Signal	SSTAT2, Bit 0	This bit represents the active high current state of the SCSI SDP1 parity signal.
Latched SCSI Parity	SSTAT 2, Bit 3 and SSTAT1, Bit 3	These bits reflect the SCSI odd parity signal corresponding to the data latched into the SIDL register
Master Parity Error Enable	CTEST4, Bit 3	Enables parity checking during master data phases.
Master Data Parity Error	DSTAT, Bit 6	Set when the SYM53C875 as a master detects that a target device has signalled a parity error during a data phase.
Master Data Parity Error Interrupt Enable	DIEN, Bit 6	By clearing this bit, a Master Data Parity Error will not cause IRQ/ to be asserted, but the status bit will be set in the DSTAT register.
Extended Byte Parity Error Interrupt Enable (53C875N only)	DIEN, Bit 1	By clearing this bit, an Extended Byte Parity Error will not cause IRQ/ to be asserted, but the status bit will be set in the DSTAT register.

**Table 2-3: SCSI Parity Control**

EPC	AESP	Description
0	0	Will not check for parity errors. Parity is generated when sending SCSI data. Asserts odd parity when sending SCSI data.
0	1	Will not check for parity errors. Parity is generated when sending SCSI data. Asserts even parity when sending SCSI data.
1	0	Checks for odd parity on SCSI data received. Parity is generated when sending SCSI data. Asserts odd parity when sending SCSI data.
1	1	Checks for odd parity on SCSI data received. Parity is generated when sending SCSI data. Asserts even parity when sending SCSI data.

*Key:*

*EPC = Enable Parity Checking (bit 3 SCNTL0)*

*ASEP = Assert SCSI Even Parity (bit 2 SCNTL1)*

*This table only applies when the Enable Parity Checking bit is set.*

**Table 2-4: SCSI Parity Errors and Interrupts**

DHP	PAR	Description
0	0	Will halt when a parity error occurs in target or initiator mode and will NOT generate an interrupt.
0	1	Will halt when a parity error occurs in target mode and will generate an interrupt in target or initiator mode.
1	0	Will not halt in target mode when a parity error occurs until the end of the transfer. An interrupt will not be generated.
1	1	Will not halt in target mode when a parity error occurs until the end of the transfer. An interrupt will be generated.

*Key: DHP = Disable Halt on SATN/ or Parity Error (bit 5 SCNTL1) PAR = Parity Error (bit 0 SIEN0)*

## DMA FIFO

The DMA FIFO is 4 bytes wide by 134 transfers deep. The DMA FIFO is illustrated in Figure 2-1. To assure compatibility with older products in the SYM53C8XX family, the user may set the DMA FIFO size to 88 bytes by clearing the DMA FIFO Size bit, bit 5 in the CTEST5 register.

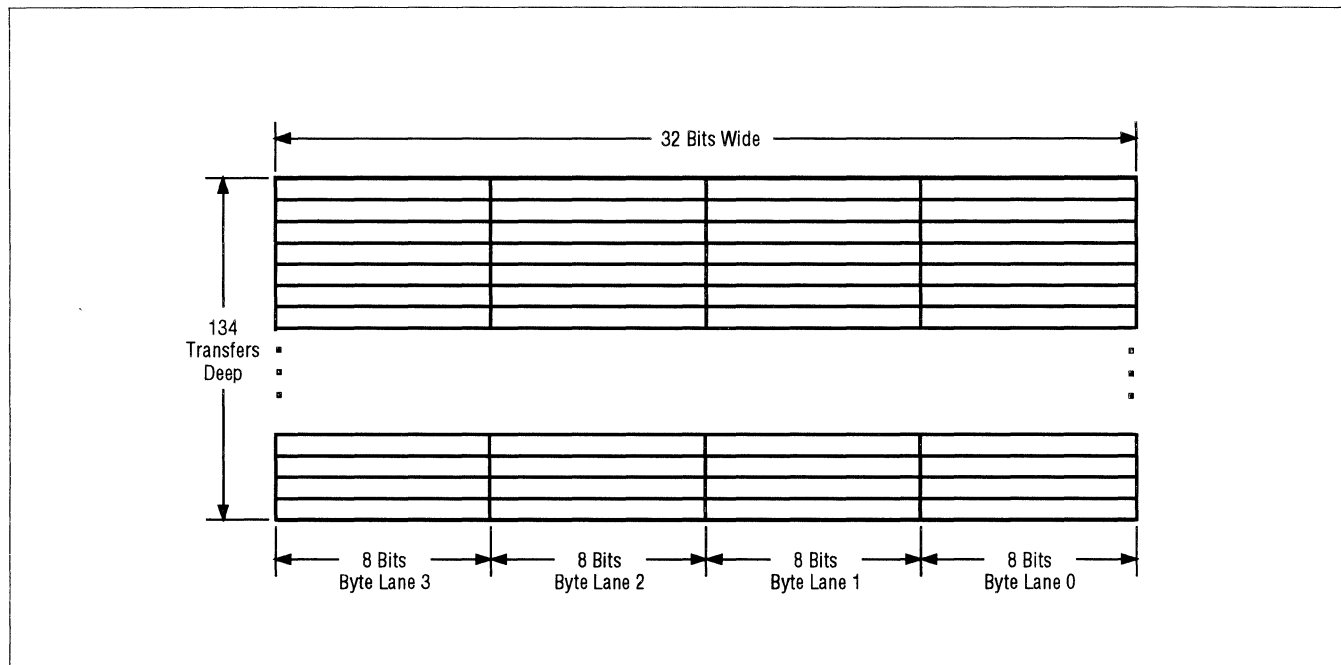


Figure 2-1: DMA FIFO Sections

## Data Paths

The data path through the SYM53C875 is dependent on whether data is being moved into or out of the chip, and whether SCSI data is being transferred asynchronously or synchronously.

Figure 2-2 shows how data is moved to/from the SCSI bus in each of the different modes.

The following steps determine if any bytes remain in the data path when the chip halts an operation:

### Asynchronous SCSI Send

1. If the DMA FIFO size is set to 88 bytes, look at the DFIFO and DBC registers and calculate if there are bytes left in the DMA FIFO. To make this calculation, subtract the seven least significant bits of the DBC register from the 7-bit value of the DFIFO register. AND the result with 7Fh for a byte count between zero and 88.

If the DMA FIFO size is set to 536 bytes (using bit 5 of the CTEST5 register), subtract the 10 least significant bits of the DBC register from the 10-bit value of the DMA FIFO Byte Offset Counter, which consists of bits 1-0 in the CTEST5 register and bits 7-0 of the DMA FIFO register. AND the result with 3FFh for a byte count between 0 and 536.

2. Read bit 5 in the SSTAT0 and SSTAT2 registers to determine if any bytes are left in the SODL register. If bit 5 is set in the SSTAT0 or SSTAT2, then the least significant byte or the most significant byte in the SODL register is full, respectively. Checking this bit also reveals bytes left in the SODL register from a Chained Move operation with an odd byte count.

### Synchronous SCSI Send

1. If the DMA FIFO size is set to 88 bytes, look at the DFIFO and DBC registers and calculate if there are bytes left in the DMA FIFO. To make this calculation, subtract the seven least significant bits of the DBC register from the 7-bit value of the DFIFO register. AND the result with 7Fh for a byte count between zero and 88.

If the DMA FIFO size is set to 536 bytes (using bit 5 of the CTEST5 register), subtract the 10 least significant bits of the DBC register from the 10-bit value of the DMA FIFO Byte Offset Counter, which consists of bits 1-0 in the CTEST5 register and bits 7-0 of the DMA FIFO register. AND the result with 3FFh for a byte count between 0 and 536.

2. Read bit 5 in the SSTAT0 and SSTAT2 registers to determine if any bytes are left in the SODL register. If bit 5 is set in the SSTAT0 or SSTAT2, then the least significant byte or the most significant byte in the SODL register is full, respectively. Checking this bit also reveals bytes left in the SODL register from a Chained Move operation with an odd byte count.
3. Read bit 6 in the SSTAT0 and SSTAT2 registers to determine if any bytes are left in the SODR register. If bit 6 is set in the SSTAT0 or SSTAT2, then the least significant byte or the most significant byte in the SODR register is full, respectively.

### Asynchronous SCSI Receive

1. If the DMA FIFO size is set to 88 bytes, look at the DFIFO and DBC registers and calculate if there are bytes left in the DMA FIFO. To make this calculation, subtract the seven least significant bits of the DBC register from the 7-bit value of the DFIFO register. AND the result with 7Fh for a byte count between 0 and 88.

If the DMA FIFO size is set to 536 bytes (using bit 5 of the CTEST5 register), subtract the 10 least significant bits of the DBC register from the 10-bit value of the DMA FIFO Byte Offset Counter, which consists of bits 1-0 in the CTEST5 register and bits 7-0 of the DMA FIFO register. AND the result with 3FFh for a byte count between 0 and 536.

2. Read bit 7 in the SSTAT0 and SSTAT2 register to determine if any bytes are left in the SIDL register. If bit 7 is set in the SSTAT0 or SSTAT2, then the least significant byte or the most significant byte is full, respectively.
3. If any wide transfers have been performed using the Chained Move instruction, read the Wide SCSI Receive bit (SCNTL2, bit 0) to determine whether a byte is left in the SWIDE

register.

### Synchronous SCSI Receive

1. If the DMA FIFO size is set to 88 bytes, subtract the seven least significant bits of the DBC register from the 7-bit value of the DFIFO register. AND the result with 7Fh for a byte count between 0 and 88.

If the DMA FIFO size is set to 536 bytes (using bit 5 of the CTEST5 register), subtract the 10 least significant bits of the DBC register from the 10-bit value of the DMA FIFO Byte Offset Counter, which consists of bits 1-0 in the CTEST5 register and bits 7-0 of the DMA FIFO register. AND the result with 3FFh for a byte count between 0 and 536.

2. Read bits 7-4 of the SSTAT1 register and bit 4 of the SSTAT2 register, the binary representation of the number of valid bytes in the SCSI FIFO, to determine if any bytes are left in the SCSI FIFO.
3. If any wide transfers have been performed using the Chained Move instruction, read the Wide SCSI Receive bit (SCNTL2, bit 0) to determine whether a byte is left in the SWIDE register.

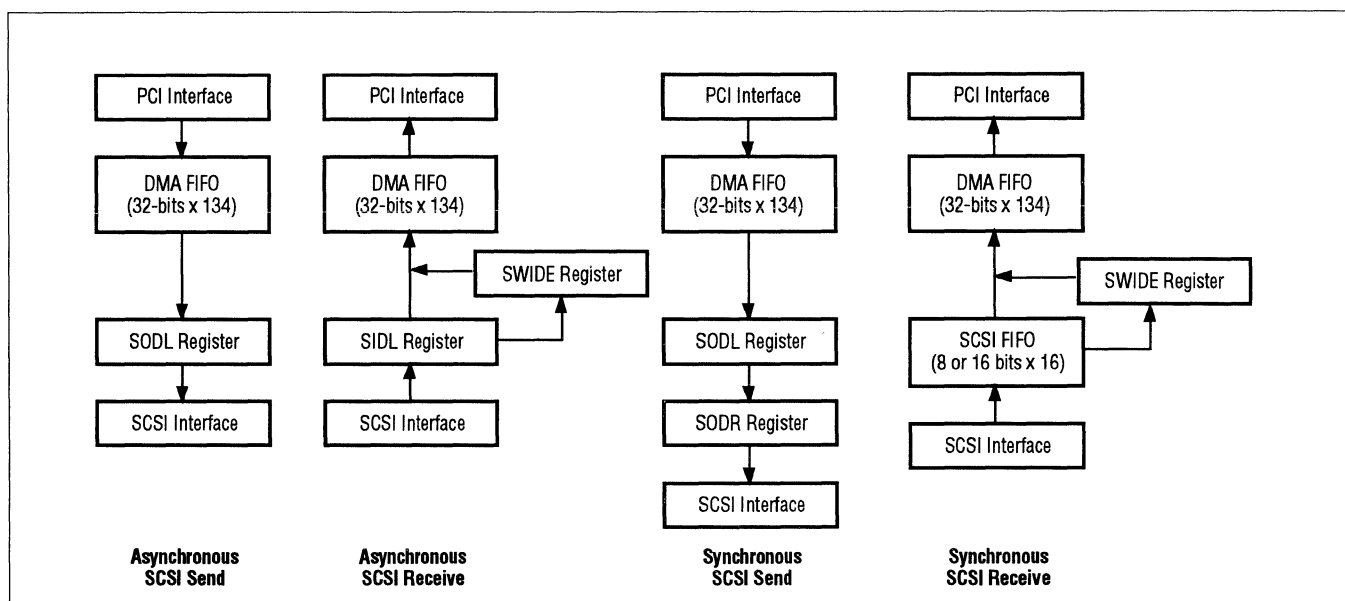


Figure 2-2: SYM53C875 Host Interface Data Paths



## SCSI Bus Interface

The SYM53C875 supports both single-ended and differential operation.

All SCSI signals are active low. The SYM53C875 contains the single-ended output drivers and can be connected directly to the SCSI bus. Each output is isolated from the power supply to ensure that a powered-down SYM53C875 has no effect on an active SCSI bus (CMOS “voltage feed-through” phenomena). TolerANT technology provides signal filtering at the inputs of SREQ/ and SACK/ to increase immunity to signal reflections.

### Differential Mode

In differential mode, the SDIR (15-0), SDIRP (1-0), IGS, TGS, RSTDIR, BSYDIR, and SELDIR signals control the direction of external differential-pair transceivers. The SYM53C875 is placed in differential mode by setting the DIF bit, bit 5 of the STEST2 register (4Eh) . Setting this bit tri-states the BSY/, SEL/, and RST/ pads so they can be used as pure input pins. In addition to the standard SCSI lines, the following signals are used during differential operation by the SYM53C875:

Signal	Function
BSYDIR, SELDIR, RSTDIR	Active high signals used to enable the differential drivers as outputs for SCSI signals BSY/, SEL/, and RST/, respectively
SDIR(15-0), SDIRP(1-0)	Active high signals used to control direction of the differential drivers for SCSI data and parity lines, respectively
IGS	Active high signal used to control direction of the differential driver for initiator group signals ATN/ and ACK/
TGS	Active high signal used to control direction of the differential drivers for target group signals MSG/, C/D/, I/O/, and REQ/
DIFFSENS	Input to the SYM53C875 used to detect the presence of a single-ended device on a differential system. If a logical zero is detected on this pin, then it is assumed that a single-ended device is on the bus and all SCSI outputs will be tri-stated to avoid damage to the transceiver.

See Figure 2-3 for an example differential wiring diagram, in which the SYM53C875 is connected to the TI SN75976A differential transceiver. The recommended value of the pull-up resistor on the REQ/, ACK/, MSG/, C/D/, I/O/, ATN/, SD0-7/, and SDP0/ lines is 680  $\Omega$  when the Active Negation portion of Symbios LogicTolerANT technology is not enabled. When TolerANT is enabled, the recommended resistor value on the REQ/, ACK/, SD7-0/, and SDP0/ signals is 1.5 K $\Omega$ . The electrical characteristics of these pins change when TolerANT is enabled, permitting a higher resistor value.

To interface the SYM53C875 to the 75LBC976, connect the DIR pins, as well as IGS and TGS, of the SYM53C875 directly to the transceiver enables (nDE/RE/). These signals control the direction of the channels on the 75LBC976.

The SCSI bi-directional control and data pins (SD7-0/, SDP0/, REQ/, ACK/, MSG/, I\_O/, C\_D/, and ATN/) of the SYM53C875 connect to the bi-directional data pins (nA) of the 75LBC976 with a pull-up resistor. The three remaining pins, SEL/, BSY/, and RST/ are connected to the 75LBC976 with a pull-down resistor. The pull-down resistors are required when the pins (nA) of the 75LBC976 are configured as inputs. When the data pins are inputs, the resistors provide a bias voltage to both the SYM53C87 pins (SEL/, BSY/, and RST/) and the 75LBC976 data pins. Because the SEL/, BSY/, and RST/ pins on the SYM53C875 are inputs

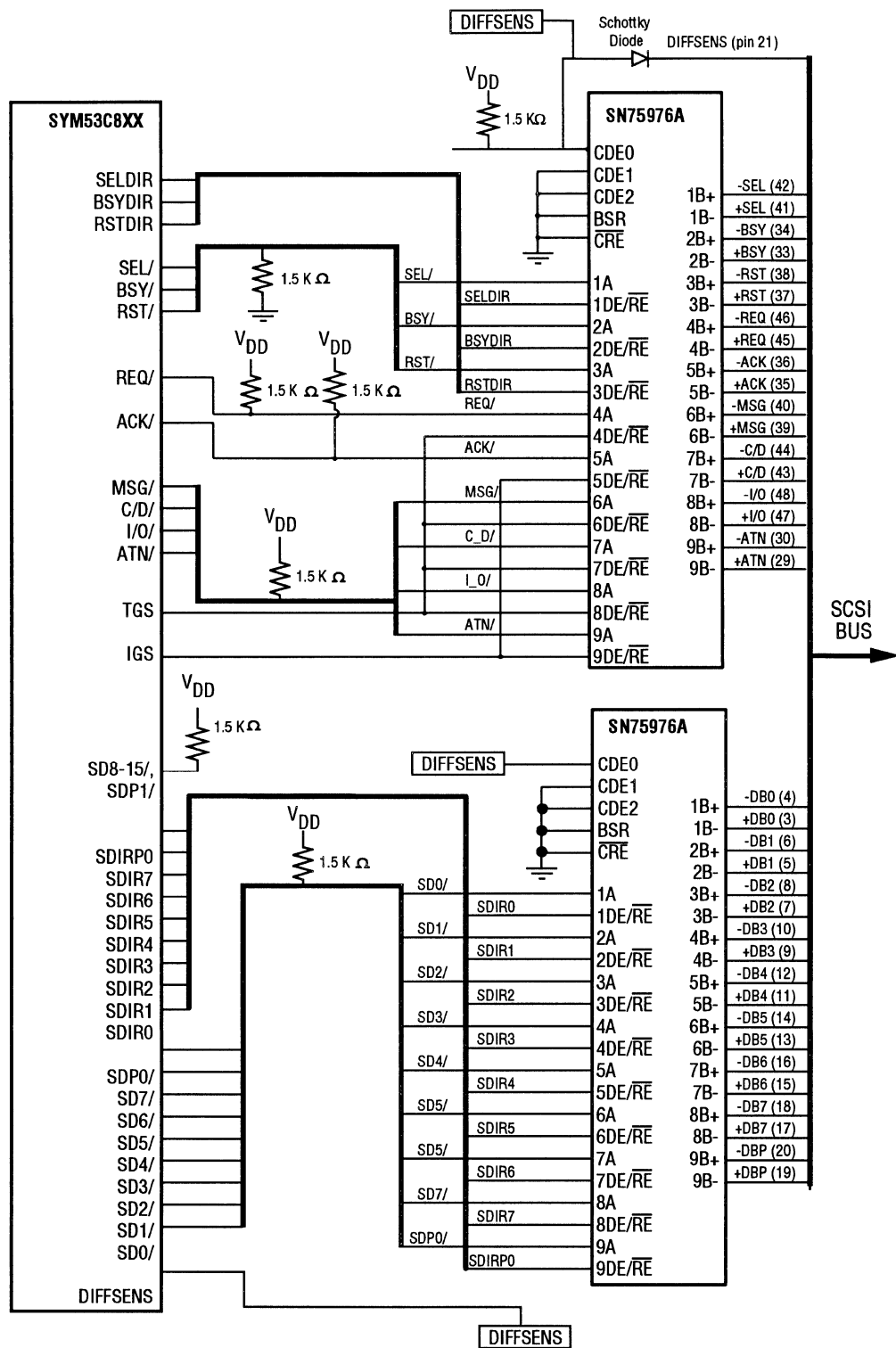
only, this configuration allows for the SEL/, BSY/, and RST/ SCSI signals to be asserted on the SCSI bus. The differential pairs on the SCSI bus are reversed when connected to the 75LBC976, due to the active low nature of the SCSI bus.

The pull-up value should be no lower than the transceiver  $I_{OL}$  can tolerate, but not so high as to cause RC timing problems.

Note: the TISN75976A differential transceiver must be used to achieve UltraSCSI transfer rates.

### 8-bit/16-bit SCSI and the Differential Interface

In an 8-bit SCSI bus, the SD15-8 pins on the SYM53C875 should be pulled up with a 1.5 K $\Omega$  resistor or terminated like the rest of the SCSI bus lines. This is very important, as errors may occur during reselection if these lines are left floating. In the SYM53C875J and SYM53C875JB, the SDIRP1 pin is replaced by the TCK JTAG signal. If the device is used in a wide differential system, use the SDIRP0 pin to control the direction of the differential transceiver for both the SP0 and SP1 signals. The SDIRP0 signal is capable of driving both direction inputs from a transceiver.



Note: Use the SN75976A to achieve UltraSCSI transfer rates.

---

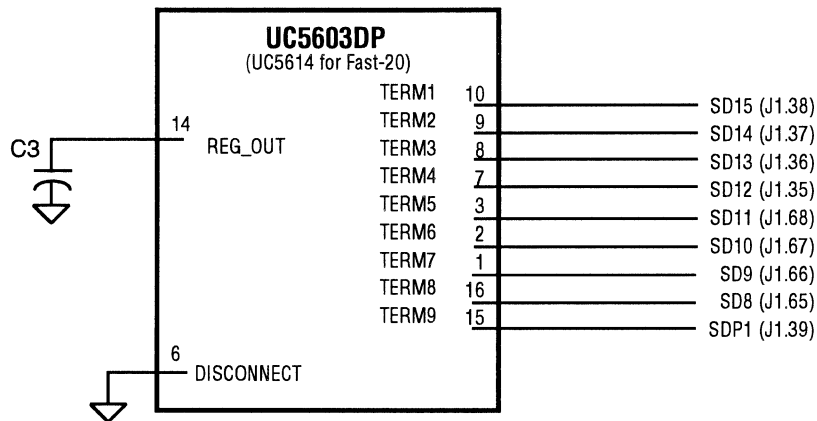
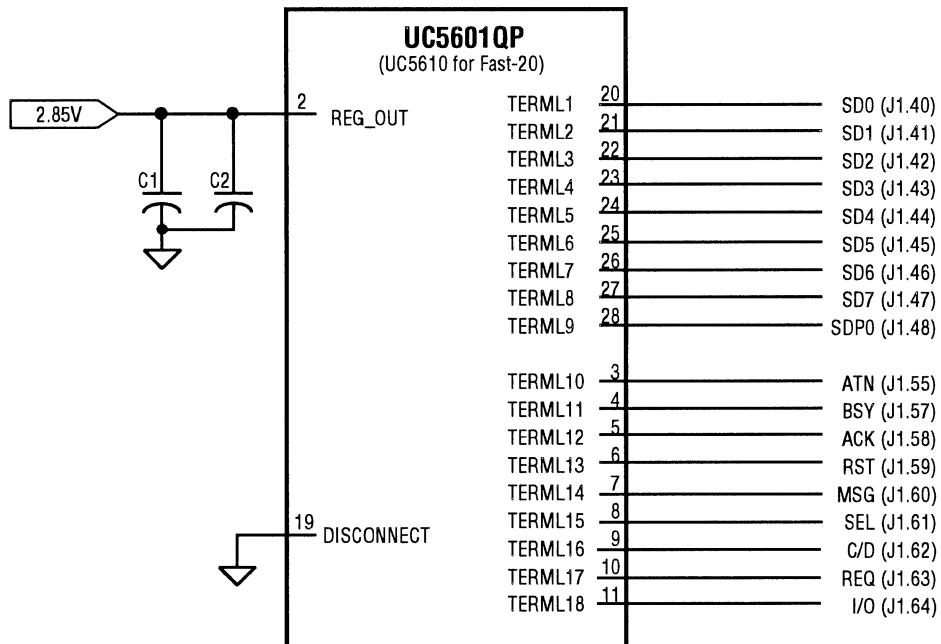
## Terminator Networks

The terminator networks provide the biasing needed to pull signals to an inactive voltage level, and to match the impedance seen at the end of the cable with the characteristic impedance of the cable. Terminators must be installed at the extreme ends of the SCSI chain, and only at the ends; no system should ever have more or less than two terminators installed and active. SCSI host adapters should provide a means of accommodating terminators. The terminators should be socketed, so that if not needed they may be removed, or there should be a means of disabling them with software.

Single-ended cables can use a 220  $\Omega$  pull-up to the terminator power supply (Term-Power) line and a 330  $\Omega$  pull-down to Ground. Because of the high-performance nature of the SYM53C875, Regulated (or Active) termination is recommended. Figure 2-4 shows a Unitrode active terminator. For additional information, refer to the SCSI-2 Specification. TolerANT active negation can be used with either termination network.

**Note:** if the SYM53C875 is to be used in a design with only an 8-bit SCSI bus, all 16 data lines still must be terminated or pulled high.

**Note:** active termination is required for UltraSCSI synchronous transfers.



Key	
C1	10 $\mu$ F SMT
C2	0.1 $\mu$ F SMT
C3	2.2 $\mu$ F SMT
J1	68-pin, high density "P" connector

**Figure 2-4: Regulated Termination**

---

### **(Re)Select During (Re)Selection**

In multi-threaded SCSI I/O environments, it is not uncommon to be selected or reselected while trying to perform selection/reselection. This situation may occur when a SCSI controller (operating in initiator mode) tries to select a target and is reselected by another. The Select SCRIPTS instruction has an alternate address to which the SCRIPTS will jump when this situation occurs. The analogous situation for target devices is being selected while trying to perform a reselection.

Once a change in operating mode occurs, the initiator SCRIPTS should start with a Set Initiator instruction or the target SCRIPTS should start with a Set Target instruction. The Selection and Reselection Enable bits (SCID bits 5 and 6, respectively) should both be asserted so that the SYM53C875 may respond as an initiator or as a target. If only selection is enabled, the SYM53C875 cannot be reselected as an initiator. There are also status and interrupt bits in the SIST0 and SIEN0 registers, respectively, indicating that the SYM53C875 has been selected (bit 5) and reselected (bit 4).

---

## **Synchronous Operation**

The SYM53C875 can transfer synchronous SCSI data in both initiator and target modes. The SXFER register controls both the synchronous offset and the transfer period. It may be loaded by the CPU before SCRIPTS execution begins, from within SCRIPTS via a Table Indirect I/O instruction, or with a Read-Modify-Write instruction.

The SYM53C875 can receive data from the SCSI bus at a synchronous transfer period as short as 50 ns, regardless of the transfer period used to send data. The SYM53C875 can receive data at one-fourth of the divided SCLK frequency. Depending on the SCLK frequency, the negotiated transfer period, and the synchronous clock divider, the SYM53C875 can send synchronous data at intervals as short as 50 ns for UltraSCSI, 100 ns for fast SCSI and 200 ns for SCSI-1.

---

### **Determining the Data Transfer Rate**

Synchronous data transfer rates are controlled by bits in two different registers of the SYM53C875. A brief description of the bits is provided below. Figure 2-5 illustrates the clock division factors used in each register, and the role of the register bits in determining the transfer rate.

### SCNTL3 Register, bits 6–4 (SCF2–0)

The SCF2-0 bits select the factor by which the frequency of SCLK is divided before being presented to the synchronous SCSI control logic. The output from this divider controls the rate at which data can be received; this rate must not exceed 80 MHz. The receive rate of synchronous SCSI data is 1/4 of the SCF divider output. For example, if SCLK is 80MHz and the SCF value is set to divide by two, then the maximum rate at which data can be received is 10 MHz ( $80/(2*4) = 10$ ).

### SCNTL3 Register, bits 2–0 (CCF2–0)

The CCF2-0 bits select the factor by which the frequency of SCLK is divided before being presented to the asynchronous SCSI core logic. This divider must be set according to the input clock frequency in the table.

### SXFER Register, bits 7–5 (TP2–0)

The TP2-0 divider bits determine the SCSI synchronous transfer period when sending synchronous SCSI data in either initiator or target mode. This value further divides the output from the SCF divider.

### Achieving Optimal SCSI Send Rates

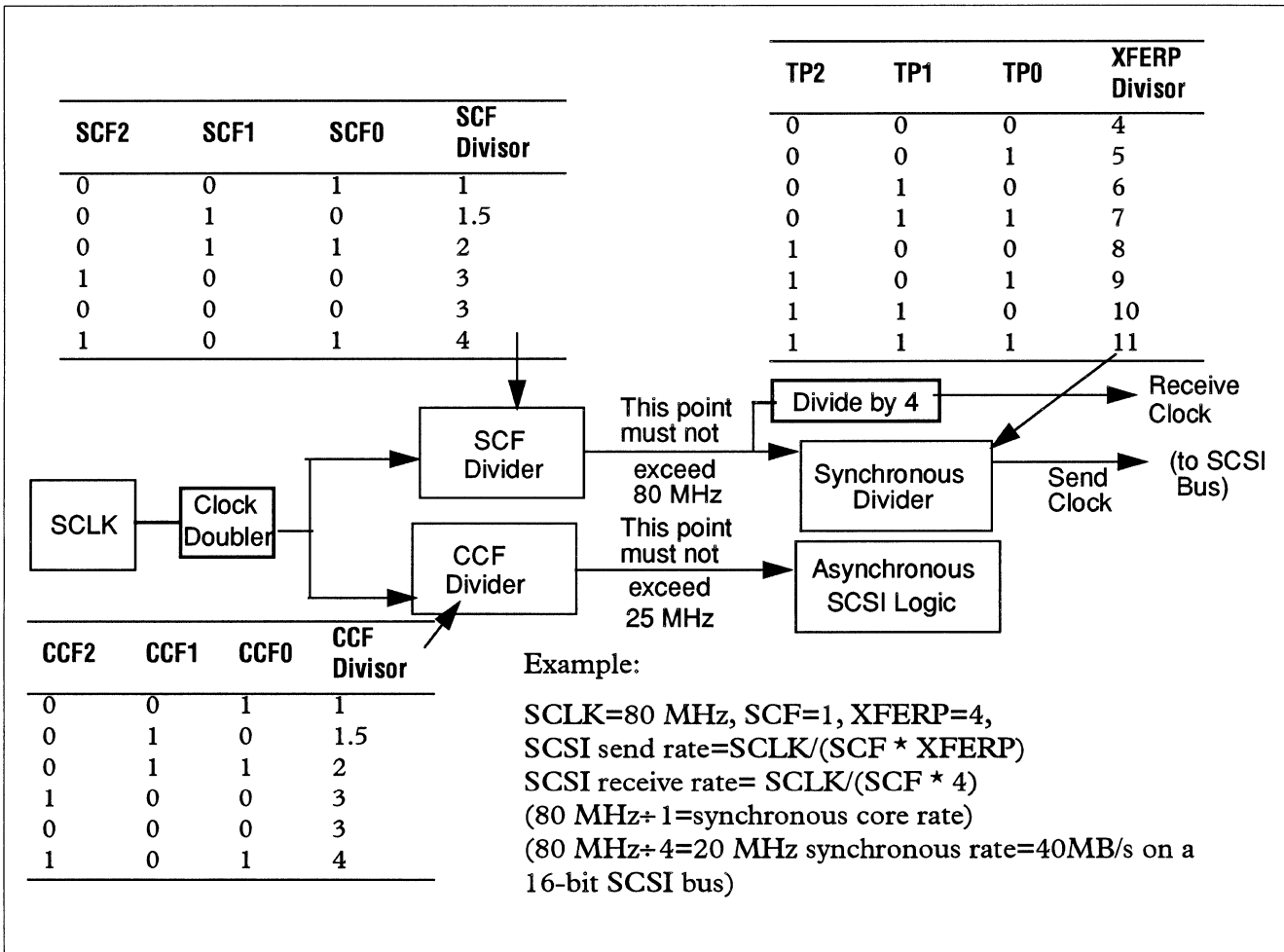
To achieve optimal synchronous SCSI send timings, the SCF divisor value should be set high, to divide the clock as much as possible before presenting the clock to the TP divider bits in the SXFER register. The TP2-0 divider value should be as low as possible. For example, with an 80 MHz clock to achieve a 10 MB/s send rate, the SCF bits can be set to divide by 1 and the TP bits to divide by 8; or the SCF bits can be set to divide by 2 and the TP bits set to divide by 2. Use the second option to achieve optimal SCSI timings.

---

## UltraSCSI Synchronous Data Transfers

UltraSCSI is simply an extension of current Fast SCSI-2 synchronous transfer specifications. It allows synchronous transfer periods to be negotiated down as low as 50 ns, which is half the 100 ns period allowed under Fast SCSI-2. This will allow a maximum transfer rate of 40 MB/s on a 16-bit SCSI bus. The SYM53C875 requires an 80MHz SCSI clock input to perform UltraSCSI transfers. In addition, the following bit values affect the chip's ability to support UltraSCSI synchronous transfer rates:

1. Clock Conversion Factor bits, SCNTL3 register bits 2-0 and Synchronous Clock Conversion Factor bits, SCNTL3 register bits 6-4. These fields now support a value of 101 (binary), allowing the SCLK frequency to be divided down by 4. This allows systems using an 80 MHz clock or the internal clock doubler to operate at Fast SCSI-2 transfer rates as well as UltraSCSI rates, if needed.
2. Fast-20 Mode Enable bit, SCNTL 3 register bit 7. Setting this bit enables UltraSCSI synchronous transfers in systems that have an 80MHz clock or use the internal SCSI clock doubler.



**Figure 2-5: Determining the Synchronous Transfer Rate**



## Interrupt Handling

The SCRIPTS processor in the SYM53C875 performs most functions independently of the host microprocessor. However, certain interrupt situations must be handled by the external microprocessor. This section explains all aspects of interrupts as they apply to the SYM53C875.

### Polling and Hardware Interrupts

The external microprocessor is informed of an interrupt condition by polling or hardware interrupts. Polling means that the microprocessor must continually loop and read a register until it detects a bit set that indicates an interrupt. This method is the fastest, but it wastes CPU time that could be used for other system tasks. The preferred method of detecting interrupts in most systems is hardware interrupts. In this case, the SYM53C875 will assert the Interrupt Request (IRQ/) line that will interrupt the microprocessor, causing the microprocessor to execute an interrupt service routine. A hybrid approach would use hardware interrupts for long waits, and use polling for short waits.

### Registers

The registers in the SYM53C875 that are used for detecting or defining interrupts are the ISTAT, SIST0, SIST1, DSTAT, SIEN0, SIEN1, DCNTL, and DIEN.

#### ISTAT

The ISTAT is the only register that can be accessed as a slave during SCRIPTS operation, therefore it is the register that is polled when polled interrupts are used. It is also the first register that should be read when the IRQ/ pin has been asserted in association with a hardware interrupt. The INTF (Interrupt on the Fly) bit should be the first interrupt serviced. It must be written to one to be cleared. This interrupt must be cleared before servicing any other interrupts. If the SIP bit in the ISTAT register is set, then a SCSI-type interrupt has occurred and the SIST0 and SIST1 registers

should be read. If the DIP bit in the ISTAT register is set, then a DMA-type interrupt has occurred and the DSTAT register should be read. SCSI-type and DMA-type interrupts may occur simultaneously, so in some cases both SIP and DIP may be set.

#### SIST0 and SIST1

The SIST0 and SIST1 registers contain the SCSI-type interrupt bits. Reading these registers will determine which condition or conditions caused the SCSI-type interrupt, and will clear that SCSI interrupt condition. If the SYM53C875 is receiving data from the SCSI bus and a fatal interrupt condition occurs, the SYM53C875 will attempt to send the contents of the DMA FIFO to memory before generating the interrupt. If the SYM53C875 is sending data to the SCSI bus and a fatal SCSI interrupt condition occurs, data could be left in the DMA FIFO. Because of this the DMA FIFO Empty (DFE) bit in DSTAT should be checked. If this bit is clear, set the CLF (Clear DMA FIFO) and CSF (Clear SCSI FIFO) bits before continuing. The CLF bit is bit 2 in CTEST3. The CSF bit is bit 1 in STEST3.

#### DSTAT

The DSTAT register contains the DMA-type interrupt bits. Reading this register will determine which condition or conditions caused the DMA-type interrupt, and will clear that DMA interrupt condition. Bit 7 in DSTAT, DFE, is purely a status bit; it will not generate an interrupt under any circumstances and will not be cleared when read. DMA interrupts will flush neither the DMA nor SCSI FIFOs before generating the interrupt, so the DFE bit in the DSTAT register should be checked after any DMA interrupt. If the DFE bit is clear, then the FIFOs must be cleared by setting the CLF (Clear DMA FIFO) and CSF (Clear SCSI FIFO) bits, or flushed by setting the FLF (Flush DMA FIFO) bit.

#### SIEN0 and SIEN1

The SIEN0 and SIEN1 registers are the interrupt enable registers for the SCSI interrupts in SIST0 and SIST1.

## DIEN

The DIEN register is the interrupt enable register for DMA interrupts in DSTAT.

## DCNTL

When bit 1 in this register is set, the IRQ/ pin will not be asserted when an interrupt condition occurs. The interrupt is not lost or ignored, but merely masked at the pin. Clearing this bit when an interrupt is pending will immediately cause the IRQ/ pin to assert. As with any register other than ISTAT, this register cannot be accessed except by a SCRIPTS instruction during SCRIPTS execution.

## Fatal vs. Non-Fatal Interrupts

A fatal interrupt, as the name implies, always causes SCRIPTS to stop running. All non-fatal interrupts become fatal when they are enabled by setting the appropriate interrupt enable bit. Interrupt masking will be discussed later in this section. All DMA interrupts (indicated by the DIP bit in ISTAT and one or more bits in DSTAT being set) are fatal.

Some SCSI interrupts (indicated by the SIP bit in the ISTAT and one or more bits in SIST0 or SIST1 being set) are non-fatal. When the SYM53C875 is operating in Initiator mode, only the Function Complete (CMP), Selected (SEL), Reselected (RSL), General Purpose Timer Expired (GEN), and Handshake to Handshake Timer Expired (HTH) interrupts are non-fatal. When operating in Target mode CMP, SEL, RSL, Target mode: SATN/ active (M/A), GEN, and HTH are non-fatal. Refer to the description for the Disable Halt on a Parity Error or SATN/ active (Target Mode Only) (DHP) bit in the SCNTL1 register to configure the chip's behavior when the SATN/ interrupt is enabled during Target mode operation. The Interrupt on the Fly interrupt is also non-fatal, since SCRIPTS can continue when it occurs.

The reason for non-fatal interrupts is to prevent SCRIPTS from stopping when an interrupt occurs that does not require service from the CPU. This

prevents an interrupt when arbitration is complete (CMP set), when the SYM53C875 has been selected or reselected (SEL or RSL set), when the initiator has asserted ATN (target mode: SATN/ active), or when the General Purpose or Handshake to Handshake timers expire. These interrupts are not needed for events that occur during high-level SCRIPTS operation.

## Masking

Masking an interrupt means disabling or ignoring that interrupt. Interrupts can be masked by clearing bits in the SIEN0 and SIEN1 (for SCSI interrupts) registers or DIEN (for DMA interrupts) register. How the chip will respond to masked interrupts depends on: whether polling or hardware interrupts are being used; whether the interrupt is fatal or non-fatal; and whether the chip is operating in Initiator or Target mode.

If a non-fatal interrupt is masked and that condition occurs, SCRIPTS will not stop, the appropriate bit in the SIST0 or SIST1 will still be set, the SIP bit in the ISTAT will not be set, and the IRQ/ pin will not be asserted. See the section on non-fatal vs. fatal interrupts for a list of the non-fatal interrupts.

If a fatal interrupt is masked and that condition occurs, then SCRIPTS will still stop, the appropriate bit in the DSTAT, SIST0, or SIST1 register will be set, and the SIP or DIP bits in the ISTAT will be set, but the IRQ/ pin will not be asserted.

When the chip is initialized, enable all fatal interrupts if you are using hardware interrupts. If a fatal interrupt is disabled and that interrupt condition occurs, SCRIPTS will halt and the system will never know it unless it times out and checks the ISTAT after a certain period of inactivity.

If you are polling the ISTAT instead of using hardware interrupts, then masking a fatal interrupt will make no difference since the SIP and DIP bits in the ISTAT inform the system of interrupts, not the IRQ/ pin. Masking an interrupt after IRQ/ is asserted will not cause IRQ/ to be deasserted.

## Stacked Interrupts

The SYM53C875 stacks interrupts if they occur one after the other. If the SIP or DIP bits in the ISTAT register are set (first level), then there is already at least one pending interrupt, and any future interrupts will be stacked in extra registers behind the SIST0, SIST1, and DSTAT registers (second level). When two interrupts have occurred and the two levels of the stack are full, any further interrupts will set additional bits in the extra registers behind SIST0, SIST1, and DSTAT. When the first level of interrupts are cleared, all the interrupts that came in afterward will move into the SIST0, SIST1, and DSTAT. After the first interrupt is cleared by reading the appropriate register, the IRQ/ pin will be deasserted for a minimum of three CLKs; the stacked interrupt(s) will move into the SIST0, SIST1, or DSTAT; and the IRQ/ pin will be asserted once again.

Since a masked non-fatal interrupt will not set the SIP or DIP bits, interrupt stacking will not occur. A masked, non-fatal interrupt will still post the interrupt in SIST0, but will not assert the IRQ/ pin. Since no interrupt is generated, future interrupts will move right into the SIST0 or SIST1 instead of being stacked behind another interrupt. When another condition occurs that generates an interrupt, the bit corresponding to the earlier masked non-fatal interrupt will still be set.

A related situation to interrupt stacking is when two interrupts occur simultaneously. Since stacking does not occur until the SIP or DIP bits are set, there is a small timing window in which multiple interrupts can occur but will not be stacked. These could be multiple SCSI interrupts (SIP set), multiple DMA interrupts (DIP set), or multiple SCSI and multiple DMA interrupts (both SIP and DIP set).

As previously mentioned, DMA interrupts will not attempt to flush the FIFOs before generating the interrupt. It is important to set either the Clear

DMA FIFO (CLF) and Clear SCSI FIFO (CSF) bits if a DMA interrupt occurs and the DMA FIFO Empty (DFE) bit is not set. This is because any future SCSI interrupts will not be posted until the DMA FIFO is clear of data. These 'locked out' SCSI interrupts will be posted as soon as the DMA FIFO is empty.

## Halting in an Orderly Fashion

When an interrupt occurs, the SYM53C875 will attempt to halt in an orderly fashion.

- If the interrupt occurs in the middle of an instruction fetch, the fetch will be completed, except in the case of a Bus Fault. Execution will not begin, but the DSP will point to the next instruction since it is updated when the current instruction is fetched.
- If the DMA direction is a write to memory and a SCSI interrupt occurs, the SYM53C875 will attempt to flush the DMA FIFO to memory before halting. Under any other circumstances only the current cycle will be completed before halting, so the DFE bit in DSTAT should be checked to see if any data remains in the DMA FIFO.
- SCSI SREQ/SACK handshakes that have begun will be completed before halting.
- The SYM53C875 will attempt to clean up any outstanding synchronous offset before halting.
- In the case of Transfer Control Instructions, once instruction execution begins it will continue to completion before halting.
- If the instruction is a JUMP/CALL WHEN/IF <phase>, the DSP will be updated to the transfer address before halting.
- All other instructions may halt before completion.

---

## Sample Interrupt Service Routine

The following is a sample of an interrupt service routine for the SYM53C875. It can be repeated if polling is used, or should be called when the IRQ/ pin is asserted if hardware interrupts are used.

1. Read ISTAT.
2. If the INTF bit is set, it must be written to a one to clear this status.
3. If only the SIP bit is set, read SIST0 and SIST1 to clear the SCSI interrupt condition and get the SCSI interrupt status. The bits in the SIST0 and SIST1 tell which SCSI interrupt(s) occurred and determine what action is required to service the interrupt(s).
4. If only the DIP bit is set, read the DSTAT to clear the interrupt condition and get the DMA interrupt status. The bits in the DSTAT will tell which DMA interrupt(s) occurred and determine what action is required to service the interrupt(s).
5. If both the SIP and DIP bits are set, read SIST0, SIST1, and DSTAT to clear the SCSI and DMA interrupt condition and get the interrupt status. If using 8-bit reads of the SIST0, SIST1, and DSTAT registers to clear interrupts, insert a 12 CLK delay between the consecutive reads to ensure that the interrupts clear properly. Both the SCSI and DMA interrupt conditions should be handled before leaving the ISR. It is recommended that the DMA interrupt be serviced before the SCSI interrupt, because a serious DMA interrupt condition could influence how the SCSI interrupt is acted upon.
6. When using polled interrupts, go back to step 1 before leaving the interrupt service routine, in case any stacked interrupts moved in when the first interrupt was cleared. When using hardware interrupts, the IRQ/ pin will be asserted again if there are any stacked interrupts. This should cause the system to re-enter the interrupt service routine.

---

## Chained Block Moves

Since the SYM53C875 has the capability to transfer 16-bit wide SCSI data, a unique situation occurs when dealing with odd bytes. The chained move (CHMOV) SCRIPTS instruction along with the Wide SCSI Send (WSS) and Wide SCSI Receive (WSR) bits in the SCNTL2 register are used to facilitate these situations. The Chained Block Move instruction is illustrated in Figure 2-6.

---

### Wide SCSI Send Bit

The WSS bit is set whenever the SCSI core is sending data (Data Out for initiator or Data In for target) and the core detects a partial transfer at the end of a chained Block Move SCRIPTS instruction (this flag will not be set if a normal Block Move instruction is used). Under this condition, the SCSI core does not send the low-order byte of the last partial memory transfer across the SCSI bus. Instead, the low-order byte is temporarily stored in the lower byte of the SODL register and the WSS flag is set. The hardware uses the WSS flag to determine what behavior must occur at the start of the next data send transfer. When the WSS flag is set at the start of the next transfer, the first byte (the high-order byte) of the next data send transfer is “married” with the stored low-order byte in the SODL register; and the two bytes are sent out across the bus, regardless of the type of Block Move instruction (normal or chained). The flag is automatically cleared when the “married” word is sent. The flag can alternately be cleared through SCRIPTS or by the microprocessor. Additionally, this bit can be used by the microprocessor or SCRIPTS for error detection and recovery purposes.

---

## Wide SCSI Receive Bit

The WSR bit is set whenever the SCSI core is receiving data (Data In for initiator or Data Out for target) and the core detects a partial transfer at the end of a block move or chained block move SCRIPTS instruction. When WSR is set, the high order byte of the last SCSI bus transfer is not transferred to memory. Instead, the byte is temporarily stored in the SWIDE register. The hardware uses the WSR bit to determine what behavior must occur at the start of the next data receive transfer. The bit is automatically cleared at the start of the next data receive transfer. The bit can alternatively be cleared by the microprocessor or through SCRIPTS. The bit can also be used by the microprocessor or SCRIPTS for error detection and recovery purposes.

---

## SWIDE Register

This register is used to store data for partial byte data transfers. For receive data, the SWIDE register holds the high-order byte of a partial SCSI transfer which has not yet been transferred to memory. This stored data may be a residue byte (and therefore ignored) or it may be valid data that will be transferred to memory at the beginning of the next Block Move instruction.

---

## SODL Register

For send data, the low-order byte of the SODL register holds the low-order byte of a partial memory transfer which has not yet been transferred across the SCSI bus. This stored data is usually “married” with the first byte of the next data send transfer, and both bytes are sent across the SCSI bus at the start of the next data send block move command.

---

## Chained Block Move SCRIPTS Instruction

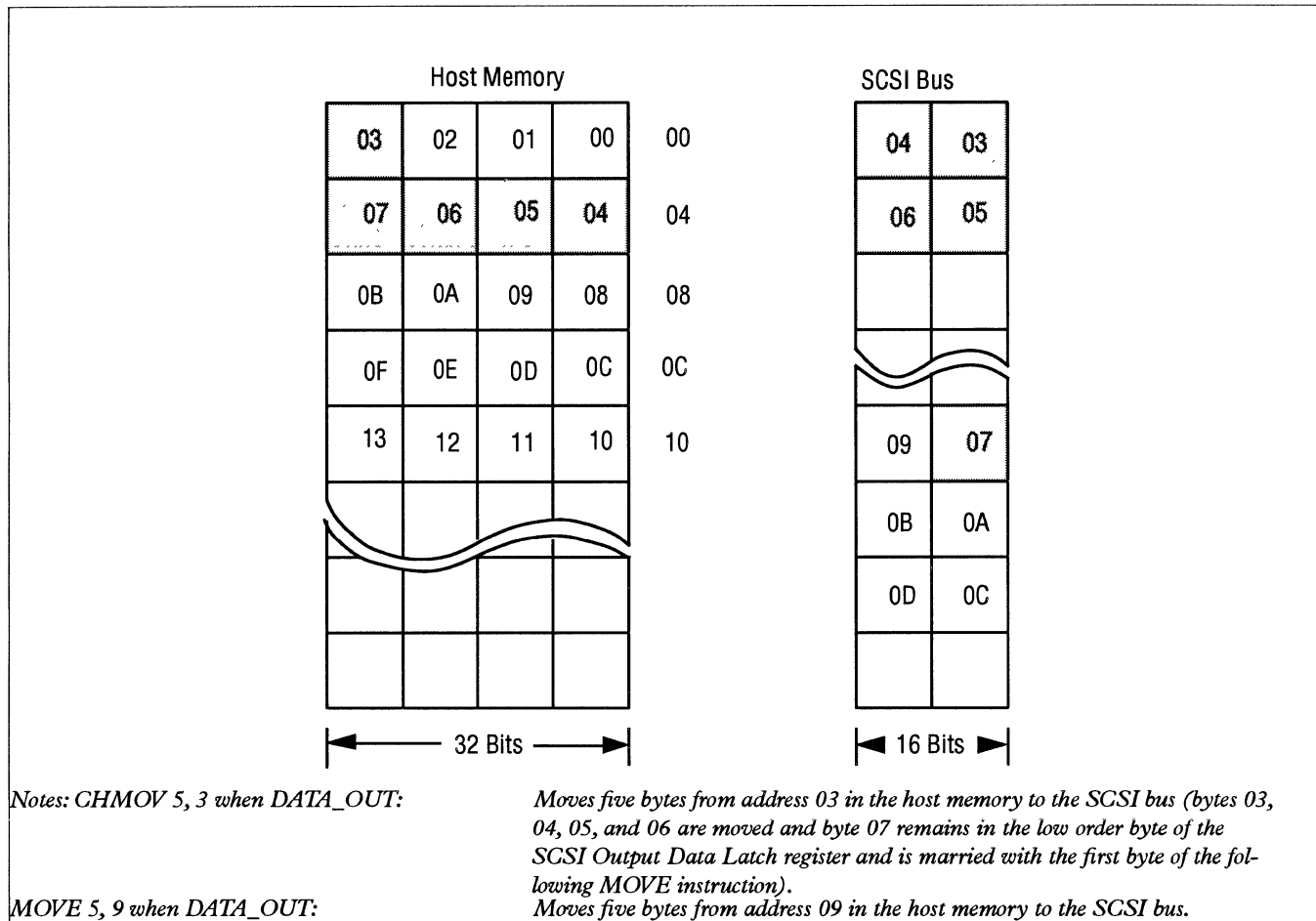
A chained Block Move SCRIPTS instruction is primarily used to transfer consecutive data send or data receive blocks. Using the chained block move instruction facilitates partial receive transfers and allows correct partial send behavior without additional op code overhead. Behavior of the chained Block Move instruction varies slightly for sending and receiving data.

For receive data (Data In for initiator or Data Out for target), a chained Block Move instruction indicates that if a partial transfer occurred at the end of the instruction, the WSR flag is set. The high order byte of the last SCSI transfer is stored in the SWIDE register rather than transferred to memory. The contents of the SWIDE register should be the first byte transferred to memory at the start of the chained block move data stream. Since the byte count always represents data transfers to/from memory (as opposed to the SCSI bus), the byte transferred out of the SWIDE register is one of the bytes in the byte count. If the WSR bit is clear when a receive data chained Block Move instruction is executed, the data transfer occurs similar to that of the regular block move instruction. Whether the WSR bit is set or clear, when a normal block move instruction is executed, the contents of the SWIDE register will be ignored and the transfer takes place normally. For “N” consecutive wide data receive Block Move instructions, the 2nd through the Nth Block Move instructions should be chained block moves.

For send data (Data Out for initiator or Data In for target), a chained Block Move instruction indicates that if a partial transfer terminates the chained block move instruction, the last low-order byte (the partial memory transfer) should be stored in the lower byte of the SODL register and not sent across the SCSI bus. Without the chained block move instruction, the last low-order byte would be sent across the SCSI bus. The starting byte count represents data bytes transferred from memory but not to the SCSI bus when a partial transfer exists.

For example, if the instruction is an Initiator chained Block Move Data Out of five bytes (and WSS is not previously set), five bytes will be transferred out of memory to the SCSI core, four bytes will be transferred from the SCSI core across the SCSI bus, and one byte will be temporarily stored in the lower byte of the SODL register waiting to be married with the first byte of the next block move instruction. Regardless of whether a chained Block Move or normal Block Move instruction is used, if the WSS bit is set at the start of a data send

command, the first byte of the data send command is assumed to be the high-order byte and is “married” with the low-order byte stored in the stored in the lower byte of the SODL register before the two bytes are sent across the SCSI bus. For “N” consecutive wide data send Block Move commands, the first through the (Nth - 1) Block Move instructions should be Chained Block Moves.



**Figure 2-6: Block Move and Chained Block Move Instructions**

## Chapter 3

# PCI Functional Description

---

## PCI Addressing

There are three types of PCI-defined address space:

- Configuration space
- Memory space
- I/O space

Configuration space is a contiguous 256 x 8-bit set of addresses dedicated to each “slot” or “stub” on the bus. Decoding  $C\_BE/(3-0)$  determines if a PCI cycle is intended to access configuration register space. The  $IDSEL$  bus signal is a “chip select” that allows access to the configuration register space only. A configuration read/write cycle without  $IDSEL$  will be ignored. The eight lower order addresses are used to select a specific 8-bit register.  $AD(10-8)$  are decoded as well, but they must be zero or the  $SYM53C875$  will not respond. According to the PCI specification,  $AD(10-8)$  are to be used for multifunction devices. The host processor uses the PCI configuration space to initialize the  $SYM53C875$ .

The lower 128 bytes of the  $SYM53C875$  configuration space holds system parameters while the upper 128 bytes map into the  $SYM53C875$  operating registers. For all PCI cycles except configuration cycles, the  $SYM53C875$  registers are located on the 256-byte block boundary defined by the base address assigned through the configured register. The  $SYM53C875$  operating registers are available in both the upper and lower 128-byte portions of the 256-byte space selected.

At initialization time, each PCI device is assigned a base address (in the case of the  $SYM53C875$ , the upper 24 bits of the address are selected) for mem-

ory accesses and I/O accesses. On every access, the  $SYM53C875$  compares its assigned base addresses with the value on the Address/Data bus during the PCI address phase. If there is a match of the upper 24 bits, the access is for the  $SYM53C875$  and the low order eight bits define the register to be accessed. A decode of  $C\_BE/(3-0)$  determines which registers and what type of access is to be performed.

PCI defines memory space as a contiguous 32-bit memory address that is shared by all system resources, including the  $SYM53C875$ . Base Address Register One determines which 256-byte memory area this device will occupy.

PCI defines I/O space as a contiguous 32-bit I/O address that is shared by all system resources, including the  $SYM53C875$ . Base Address Register Zero determines which 256-byte I/O area this device will occupy.

---

## PCI Bus Commands and Functions Supported

Bus commands indicate to the target the type of transaction the master is requesting. Bus commands are encoded on the  $C\_BE/(3-0)$  lines during the address phase. PCI bus command encoding and types appear in Table 3-1.

The I/O Read command is used to read data from an agent mapped in I/O address space. All 32 address bits are decoded.

The I/O Write command is used to write data to an agent when mapped in I/O address space. All 32 address bits are decoded.

**PCI Bus Commands and Functions Supported**

The Memory Read, Memory Read Multiple, and Memory Read Line commands are used to read data from an agent mapped in memory address space. All 32 address bits are decoded.

The Memory Write and Memory Write and Invalidate commands are used to write data to an agent when mapped in memory address space. All 32 address bits are decoded.

**Table 3-1: PCI Bus Commands and Encoding Types**

<b>C_BE(3-0)</b>	<b>Command Type</b>	<b>Supported as Master</b>	<b>Supported as Slave</b>
0000	Special Interrupt Acknowledge	No	No
0001	Special Cycle	No	No
0010	I/O Read Cycle	Yes	Yes
0011	I/O Write Cycle	Yes	Yes
0100	Reserved		n/a
0101	Reserved		n/a
0110	Memory Read	Yes	Yes
0111	Memory Write	Yes	Yes
1000	Reserved		n/a
1001	Reserved		n/a
1010	Configuration Read	No	Yes
1011	Configuration Write	No	Yes
1100	Memory Read Multiple	Yes**	No (defaults to 0110)
1101	Dual Address Cycle	No	No
1110	Memory Read Line	Yes*	No (defaults to 0110)
1111	Memory Write and Invalidate	Yes***	No (defaults to 0111)

\*This operation is selectable by bit 3 in the DMODE operating register

\*\*This operation is selectable by bit 2 in the DMODE operating register

\*\*\*This operation is selectable by bit 0 in the CTEST3 operating register



## PCI Cache Mode

The SYM53C875 supports the PCI specification for an 8-bit Cache Line Size register located in PCI configuration space. The Cache Line Size register provides the ability to sense and react to non-aligned addresses corresponding to cache line boundaries. In conjunction with the Cache Line Size register, the PCI commands Read Line, Read Multiple, and Write and Invalidate are each software enabled or disabled to allow the user full flexibility in using these commands.

### Support for PCI Cache Line Size Register

The SYM53C875 supports the PCI specification for an 8-bit Cache Line Size register in PCI configuration space; it can sense and react to non-aligned addresses corresponding to cache line boundaries.

### Selection of Cache Line Size

The cache logic will select a cache line size based on the values for the burst size in the DMODE register and the PCI Cache Line Size register.

**Note:** the SYM53C875 will not automatically use the value in the PCI Cache Line Size register as the cache line size value. The chip scales the value of the Cache Line Size register down to the nearest binary burst size allowed by the chip (2, 4, 8, 16, 32, 64, or 128), compares this value to the burst size defined by the values of the DMODE register and bit 2 of the CTEST5 register, then selects the smallest as the value for the cache line size. The SYM53C875 will use this value for all burst data transfers.

### Alignment

The SYM53C875 uses the calculated line size value to monitor the current address for alignment to the cache line size. When it is not aligned, the chip attempts to align to the cache boundary by using a “smart aligning” scheme. This means that it will attempt to use the largest burst size possible that is less than the cache line size, to reach the cache boundary quickly with no overflow. This process is a stepping mechanism that will step up to the highest possible burst size based on the current address.

The stepping process begins at a 4-dword boundary. The SYM53C875 will first try to align to a 4-dword boundary (0x0000, 0x0008, 0x0010, etc.) by using single dword transfers (no bursting). Once this boundary has been reached the chip will evaluate the current alignment to various burst sizes allowed, and will select the largest possible as the next burst size, while not exceeding the cache line size. The chip will then issue this burst, and re-evaluate the alignment to various burst sizes, again selecting the largest possible while not exceeding the cache line size, as the next burst size. This stepping process continues until the chip reaches the cache line size boundary or runs out of data. Once a cache line boundary is reached, the chip will use the cache line size as the burst size from then on, except in the case of multiples (explained below). The alignment process is finished at this point.

### Example: Cache Line Size - 16, Current Address = 0x01

The chip is not aligned to a 4-dword cache boundary (the stepping threshold), so it issues four single-dword transfers (the first is a 3-byte transfer). At address 0x10, the chip is aligned to a 4-dword boundary, but not aligned to any higher burst size boundaries that are less than the cache line size. So, the part will issue a burst of 4. At this point, the address is 0x20, and the chip will evaluate that it is aligned not only to a 4-dword boundary, but also to an 8-dword boundary. It will select the highest, 8, and burst 8 dwords. At this point, the address is 0x40, which is a cache line size boundary. Alignment stops, and the burst size from then on is switched to 16.

## Memory Move Misalignment

The SYM53C875 will not operate in a cache alignment mode when a Memory Move instruction type is issued and the read and write addresses are different distances from the nearest cache line boundary. For example, if the read address is 0x21F and the write address is 0x42F, and the cache line size is eight (8), the addresses are byte aligned, but they are not the same distance from the nearest cache boundary. The read address is 1 byte from the cache boundary 0x220 and the write address is 17 bytes from the cache boundary 0x440. In this situation, the chip will not align to cache boundaries and will operate as an SYM53C825.

## Memory Write and Invalidate Command

The Memory Write and Invalidate command is identical to the Memory Write command, except that it additionally guarantees a minimum transfer of one complete cache line; i.e., the master intends to write all bytes within the addressed cache line in a single PCI transaction unless interrupted by the target. This command requires implementation of the PCI Cache Line Size register at address 0Ch in PCI configuration space. The SYM53C875 enables Memory Write and Invalidate cycles when bit 0 in the CTEST3 register (WRIE) and bit 4 in the PCI Command register are set. This will cause Memory Write and Invalidate commands to be issued when the following conditions are met:

1. The CLSE bit, WRIE bit, and PCI configuration Command register, bit 4 must be set.
2. The cache line size register must contain a legal burst size (2, 4, 8, 16, 32, 64, or 128) value AND that value must be less than or equal to the DMODE burst size.
3. The chip must have enough bytes in the DMA FIFO to complete at least one full cache line burst.

4. The chip must be aligned to a cache line boundary.

When these conditions have been met, the SYM53C875 will issue a Write and Invalidate command instead of a Memory Write command during all PCI write cycles.

## Multiple Cache Line Transfers

The Write and Invalidate command can write multiple cache lines of data in a single bus ownership. The chip issues a burst transfer as soon as it reaches a cache line boundary. The size of the transfer will not automatically be the cache line size, but rather a multiple of the cache line size as allowed for in the Revision 2.1 of the PCI specification. The logic will select the largest multiple of the cache line size based on the amount of data to transfer, with the maximum allowable burst size being that determined from the DMODE Burst Size bits and CTEST 5, bit 2. If multiple cache line size transfers are not desired, the DMODE burst size can be set to exactly the cache line size and the chip will only issue single cache line transfers.

After each data transfer, the chip re-evaluates the burst size based on the amount of remaining data to transfer and again selects the highest possible multiple of the cache line size, no larger than the DMODE burst size. The most likely scenario of this scheme is that the chip will select the DMODE burst size after alignment, and issue bursts of this size. The burst size will, in effect, throttle down toward the end of a long Memory Move or Block Move transfer until only the cache line size burst size is left; the chip will finish the transfer with this burst size.

## Latency

In accordance with the PCI specification, the chip's latency timer will be ignored when issuing a Write and Invalidate command such that when a latency time-out has occurred, the SYM53C875 will continue to transfer up until a cache line boundary. At that point, the chip will relinquish the bus, and finish the transfer at a later time using another bus ownership. If the chip is transferring multiple cache lines it will continue to transfer until the next cache boundary is reached.

## PCI Target Retry

During a Write and Invalidate transfer, if the target device issues a retry (STOP with no TRDY, indicating that no data was transferred), the SYM53C875 will relinquish the bus and immediately try to finish the transfer on another bus ownership. The chip will issue another Write and Invalidate command on the next ownership, in accordance with the PCI specification.

## PCI Target Disconnect

During a Write and Invalidate transfer, if the target device issues a disconnect the SYM53C875 will relinquish the bus and immediately try to finish the transfer on another bus ownership. The chip will not issue another Write and Invalidate command on the next ownership unless the address is aligned.

## Memory Read Line Command

This command is identical to the Memory Read command, except that it additionally indicates that the master intends to fetch a complete cache line. This command is intended to be used with bulk sequential data transfers where the memory system and the requesting master might gain some performance advantage by reading up to a cache line boundary rather than a single memory cycle. The Read Line Mode function that exists in the previous SYM53C8XX chips has been modified in the SYM53C875 to reflect the PCI cache line size reg-

ister specifications. The functionality of the Enable Read Line bit (bit 3 in DMODE) has been modified to more resemble the Write and Invalidate mode in terms of conditions that must be met before a Read Line command will be issued. However, the Read Line option will operate exactly like the previous SYM53C8XX chips when cache mode has been disabled by a CLSE bit reset or when certain conditions exist in the chip (explained below).

The Read Line mode is enabled by setting bit 3 in the DMODE register. If cache mode has been disabled, Read Line commands will be issued on every read data transfer, except op code fetches, as in previous SYM53C8XX chips.

If cache mode has been enabled, a Read Line command will be issued on all read cycles, except op code fetches, when the following conditions have been met:

1. The CLSE and Enable Read Line bits must be set.
2. The Cache Line Size register must contain a legal burst size value (2, 4, 8, 16, 32, 64, or 128) AND that value must be less than or equal to the DMODE burst size.
3. The number of bytes to be transferred at the time a cache boundary has been reached must be equal to or greater than the DMODE burst size.
4. The chip must be aligned to a cache line boundary.

When these conditions have been met, the chip will issue a Read Line command instead of a Memory Read during all PCI read cycles. Otherwise, it will issue a normal Memory Read command.

---

## Memory Read Multiple Command

This command is identical to the Memory Read command except that it additionally indicates that the master may intend to fetch more than one cache line before disconnecting. The SYM53C875 supports PCI Read Multiple functionality and will issue Read Multiple commands on the PCI bus when the Read Multiple Mode is enabled. This mode is enabled by setting bit 2 of the DMODE register (ERMP). If cache mode has been enabled, a Read Multiple command will be issued on all read cycles, except op code fetches, when the following conditions have been met:

1. The CLSE and ERMP bits must be set.
2. The Cache Line Size register must contain a legal burst size value (2, 4, 8, 16, 32, 64, or 128) AND that value must be less than or equal to the DMODE burst size.
3. The number of bytes to be transferred at the time a cache boundary has been reached must be at least twice the full cache line size.
4. The chip must be aligned to a cache line boundary.

When these conditions have been met, the chip will issue a Read Multiple command instead of a Memory Read during all PCI read cycles.

## Burst Size Selection

The Read Multiple command reads in multiple cache lines of data in a single bus ownership. The number of cache lines to be read is a multiple of the cache line size as allowed for in the Revision 2.1 of the PCI specification. The logic will select the largest multiple of the cache line size based on the amount of data to transfer, with the maximum allowable burst size being that determined from the DMODE Burst Size bits and CTEST 5, bit 2.

## Read Multiple with Read Line Enabled

When both the Read Multiple and Read Line modes have been enabled, the Read Line command will not be issued if the above conditions are met. Instead, a Read Multiple command will be issued, even though the conditions for Read Line have been met.

If the Read Multiple mode is enabled and the Read Line mode has been disabled, Read Multiple commands will still be issued if the Read Multiple conditions are met.

## Unsupported PCI Commands

The SYM53C875 does not respond to reserved commands, special cycle, dual address cycle, or interrupt acknowledge commands as a slave. It will never generate these commands as a master.

## Configuration Registers

The Configuration registers are accessible only by system BIOS during PCI configuration cycles, and are not available to the user at any time. No other cycles, including SCRIPTS operations, can access these registers. The lower 128 bytes hold configuration data while the upper 128 bytes hold the SYM53C875 operating registers, which are described in Chapter Five, “Operating Registers.” These registers can be accessed by SCRIPTS or the host processor.

Note: the configuration register descriptions provide general

information only, to indicate which PCI configuration addresses are supported in the SYM53C875. Table 3-2 shows the PCI configuration registers implemented by the SYM53C875. Addresses 40h through 7Fh are not defined.

All PCI-compliant devices, such as the SYM53C875, must support the Vendor ID, Device ID, Command, and Status Registers. Support of other PCI-compliant registers is optional. In the SYM53C875, registers that are not supported are not writable and will return all zeroes when read. Only those registers and bits that are currently supported by the SYM53C875 are described in this chapter. For more detailed information on PCI registers, please see the PCI Specification.

31	16	15	0	
Device ID = 000Fh		Vendor ID = 1000h		00h
Status		Command		04h
Class Code = 010000h			Rev ID = 0Xh	08h
Not Supported	Header Type	Latency Timer	Cache Line Size	0Ch
Base Address Zero (I/O) <sup>1</sup>				10h
Base Address One (Memory) <sup>2</sup>				14h
RAM Base Address <sup>3</sup>				18h
Not Supported				1Ch
Not Supported				20h
Not Supported				24h
Reserved				28h
Subsystem Data Register				2Ch
Expansion ROM Base Address <sup>4</sup>				30h
Reserved	Reserved	General Purpose Pin Control <sup>5</sup>		34h
Reserved				38h
Max_Lat	Min_Gnt	Interrupt Pin	Interrupt Line	3Ch

Figure 3-1: PCI Configuration Register Map

<sup>1</sup>I/O Base is supported

<sup>2</sup>Memory Base is supported

Note: Addresses 40h to 7Fh are not defined. All unsupported registers are not writable and will return all zeroes when read. Reserved registers will also return zeroes when read.

<sup>3</sup>This register powers up enabled and can be disabled by pull-down resistors on the MAD5 pin

<sup>4</sup>If expansion memory is enabled through pull-down resistors on the MAD(7-0) bus.

<sup>5</sup>Both of these registers default to Reserved and are enabled through pull-down resistors on the MAD(7-6) pins at reset.

---

**Register 00h**

Vendor ID

Read Only

This field identifies the manufacturer of the device. The Symbios Logic Vendor ID is 1000h.

---

**Register 02h**

Device ID

Read Only

This field identifies the particular device. The SYM53C875 device ID is 000Fh.

---

**Register 04h**

Command

Read/Write

The Command Register, illustrated in Figure 3-2, provides coarse control over a device's ability to generate and respond to PCI cycles. When a zero is written to this register, the SYM53C875 is logically disconnected from the PCI bus for all accesses except configuration accesses.

In the SYM53C875, bits 3 through 5 and bit 7 and 9 are not implemented. Bits 10 through 15 are reserved.

**Bits 15-9 Reserved****Bit 8 SERR/ Enable**

This bit enables the SERR/ driver. SERR/ is disabled when this bit is clear. The default value of this bit is zero. This bit and bit 6 must be set to report address parity errors.

**Bit 7 Reserved****Bit 6 Enable Parity Error Response**

This bit allows the SYM53C875 to detect parity errors on the PCI bus and report these errors to the system. Only data parity checking is enabled. The SYM53C875 always generates parity for the PCI bus.

**Bit 5 Reserved****Bit 4 Write and Invalidate Mode**

This bit, when set, will cause Memory Write and Invalidate cycles to be issued on the PCI bus after certain conditions have been met. For more information on these conditions, refer to the section "Memory Write and Invalidate Command". To enable Write and Invalidate Mode, bit 0 in the CTEST3 register (operating register set) must also be set.

**Bit 3 Reserved****Bit 2 Enable Bus Mastering**

This bit controls the SYM53C875's ability to act as a master on the PCI bus. A value of zero disables the device from generating PCI bus master accesses. A value of one allows the SYM53C875 to behave as a bus master. The SYM53C875 must be a bus master in order to fetch SCRIPTS instructions and transfer data.

**Bit 1 Enable Memory Space**

This bit controls the SYM53C875's response to Memory Space accesses. A value of zero disables the device response. A value of one allows the SYM53C875 to respond to Memory Space accesses at the address specified by Base Address One.

**Bit 0 Enable I/O Space**

This bit controls the SYM53C875's response to I/O space accesses. A value of zero disables the response. A value of one allows the SYM53C875 to respond to I/O space accesses at the address specified in Base Address Zero.

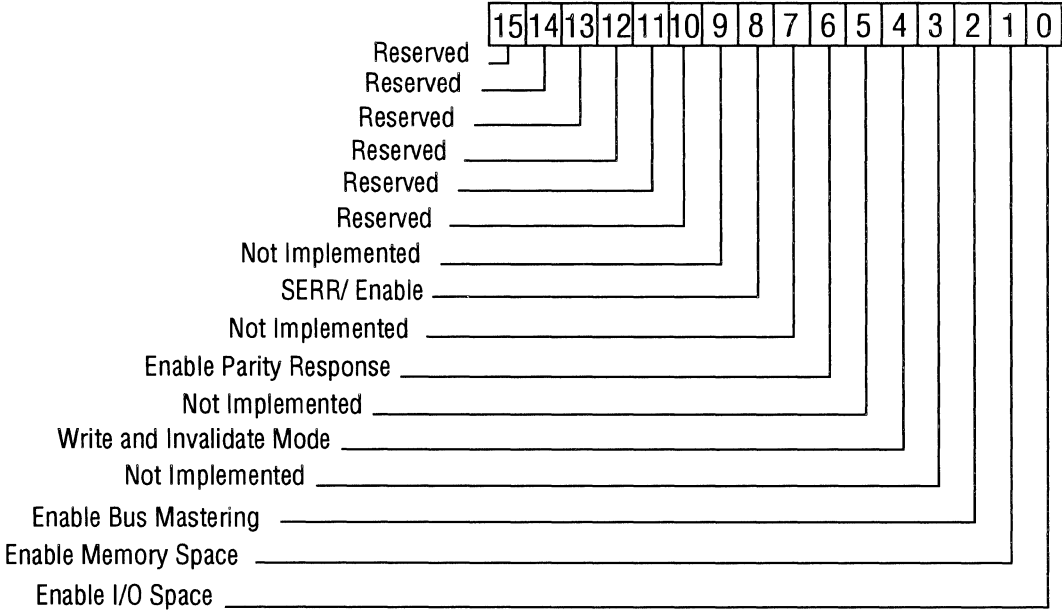


Figure 3-2: Command Register Layout

---

**Register 06h**  
**Status**  
**Read/Write**

The Status Register, illustrated in Figure 3-3, is used to record status information for PCI bus-related events.

In the SYM53C875, bits 0 through 4 are reserved and bits 5, 6, 7, and 11 are not implemented by the SYM53C875.

Reads to this register behave normally. Writes are slightly different in that bits can be cleared, but not set. A bit is reset whenever the register is written, and the data in the corresponding bit location is a one. For instance, to clear bit 15 and not affect any other bits, write the value 8000h to the register.

**Bit 15 Detected Parity Error (from Slave)**

This bit will be set by the SYM53C875 whenever it detects a data parity error, even if parity error handling is disabled.

**Bit 14 Signaled System Error**

This bit is set whenever a device asserts the SERR/ signal.

**Bit 13 Master Abort (from Master)**

This bit should be set by a master device whenever its transaction (except for Special Cycle) is terminated with master-abort. All master devices should implement this bit.

**Bit 12 Received Target Abort (from Master)**

This bit should be set by a master device whenever its transaction is terminated with a target abort. All master devices should implement this bit.

**Bit 11 Reserved**

**Bits 10-9 DEVSEL/Timing**

These bits encode the timing of DEVSEL/.

These are encoded as 00b for fast, 01b for medium, 10b for slow with 11b reserved.

These bits are read-only and should indicate the slowest time that a device asserts DEVSEL/ for any bus command except Configuration Read and Configuration Write. In the SYM53C875, 01b is supported.

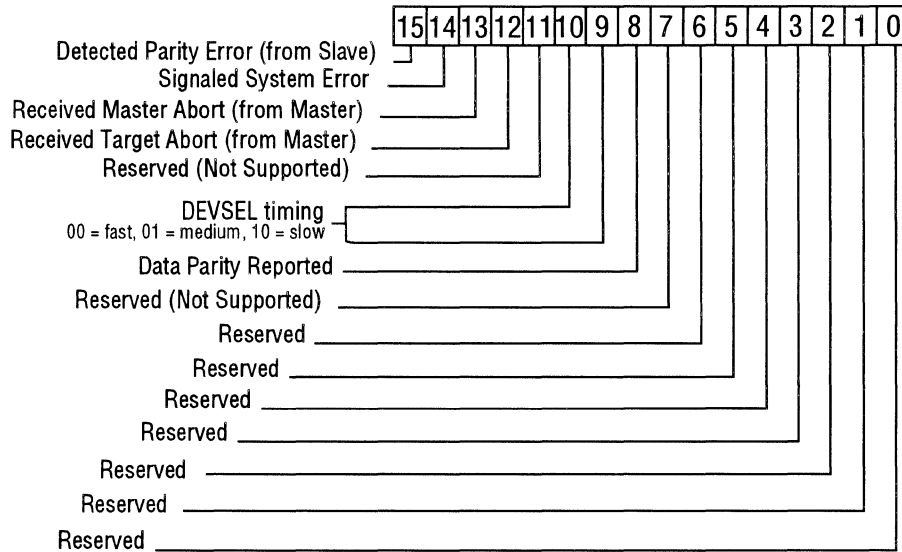
**Bit 8 Data Parity Reported**

This bit is set when the following three conditions are met: 1) The bus agent asserted PERR/ itself or observed PERR/ asserted; 2)

The agent setting this bit acted as the bus master for the operation in which the error occurred; 3) The Parity Error Response bit in the Command register is set.

**Bits 7-0 Reserved**





**Figure 3-3: Status Register Layout**

**Register 08h**  
**Revision ID**  
**Read Only**

This register specifies device and revision identifiers. In the SYM53C875, the upper nibble will be 0000b. The lower nibble reflects the current revision level of the device. It should have the same value as the Chip Revision Level bits in the CTEST3 register.

**Register 09h**  
**Class Code**  
**Read Only**

This register is used to identify the generic function of the device. The upper byte of this register is a base class code, the middle byte is a subclass code, and the lower byte identifies a specific register-level programming interface. The value of this register is 010000h, which indicates a SCSI controller.

**Register 0Ch**  
**Cache Line Size**  
**Read/Write**

This register specifies the system cache line size in units of 32-bit words. Cache mode is enabled and disabled by the Cache Line Size Enable (CLSE) bit, bit 7 in the DCNTL register. Setting this bit causes the SYM53C875 to align to cache line boundaries before allowing any bursting, except during Memory Moves in which the read and write addresses are not aligned to a burst size boundary. For more information on this register, see the section "Support for PCI Cache Line Size Register".

---

**Register 0Dh**  
**Latency Timer**  
**Read/Write**

The Latency Timer register specifies, in units of PCI bus clocks, the value of the Latency Timer for this PCI bus master. The SYM53C875 supports this timer. All eight bits are writable, allowing latency values of 0-255 PCI clocks. Use the following equation to calculate an optimum latency value for the SYM53C875:

$$\text{Latency} = 2 + (\text{Burst Size} * (\text{typical wait states} + 1)).$$

Values greater than optimum are also acceptable.

---

**Register 0Eh**  
**Header Type**  
**Read Only**

This register identifies the layout of bytes 10h through 3Fh in configuration space and also whether or not the device contains multiple functions. The value of this register is 00h.

---

**Register 10h**  
**Base Address Zero (I/O)**  
**Read/Write**

This 32-bit register has bit zero hardwired to one. Bit 1 is reserved and must return a zero on all reads, and the other bits are used to map the device into I/O space.

---

**Register 14h**  
**Base Address One (Memory)**  
**Read/Write**

This register has bit 0 hardwired to zero. For detailed information on the operation of this register, refer to the PCI Specification.

---

**Register 18h**  
**RAM Base Address**  
**Read/Write**

This register holds the memory base address of the 4 KB internal RAM. The user can read this register through the SCRATCHB register in the operating register set when bit 3 of the CTEST2 register is set.

---

**Register 2C-2F**  
**Subsystem Data**  
**Read/Write**

This register supports subsystem identification. It is byte-addressable and powers up as all zeroes. To write to this register, connect a 4.7 K $\Omega$  resistor between the MAD(6) pin and V<sub>SS</sub>. The MAD(6) pin contains an internal pull-up and will be sensed shortly after the deassertion of Chip Reset. If sensed high, the register will always return zeros; if sensed low, the register can be written with non-zero values.

---

**Register 30h**  
**Expansion ROM Base Address**  
**Read/Write**

This four-byte register handles the base address and size information for expansion ROM. It functions exactly like the Base Address Zero and Base Address One registers, except that the encoding of the bits is different. The upper 21 bits correspond to the upper 21 bits of the expansion ROM base address.

The Expansion ROM Enable bit, bit 0, is the only bit defined in this register. This bit is used to control whether or not the device accepts accesses to its expansion ROM. When the bit is set, address decoding is enabled, and a device can be used with or without an expansion ROM depending on the system configuration. To access the external memory interface, the Memory Space bit in the Command register must also be set.

The host system detects the size of the external memory by first writing the Expansion ROM Base Address register with all ones and then reading back the register. The SYM53C875 will respond with zeros in all don't care locations. The ones in the remaining bits represent the binary version of the external memory size. For example, to indicate an external memory size of 32 KB, this register, when written with ones and read back, will return ones in the upper 17 bits.

---

### Register 34-35h

#### General Purpose Pin Control

Read/Write

This register is enabled if the SYM53C875 is configured to access a serial EEPROM. If serial EEPROM access is enabled, the least significant bit at address 34h will be set, indicating that the SYM53C875 is serial EEPROM-capable. Register 35h will contain the direction control and bit values associated with the serial clock and data, defined below.

#### Bit 5 GPIO1 Control

This bit contains the direction control for the GPIO1 pin. This bit powers up clear, indicating that GPIO1 is an input. Setting this direction bit will cause GPIO1 to become an output.

#### Bit 4 GPIO0 Control

This bit contains the direction control for the GPIO0 pin. This bit powers up clear, indicating that GPIO0 is an input. Setting this direction bit will cause GPIO0 to become an output.

#### Bit 1 GPIO1 Value

This bit contains the value which will be available on GPIO1.

#### Bit 0 GPIO0 Value

This bit contains the value which will be available on GPIO0.

All bits in registers 34h and 35h will be cleared if GPIO1-0 access is disabled..

---

### Register 3Ch

#### Interrupt Line

Read/Write

This register is used to communicate interrupt line routing information. POST software will write the routing information into this register as it initiates and configures the system. The value in this register tells which input of the system interrupt controller(s) the device's interrupt pin has been connected to. Values in this register are specified by system architecture.

---

### Register 3Dh

#### Interrupt Pin

Read Only

This register tells which interrupt pin the device uses. Its value is set to 01h, for the INTA/ signal.

---

### Register 3Eh

#### Min\_Gnt

Read Only

---

### Register 3Fh

#### Max\_Lat

Read Only

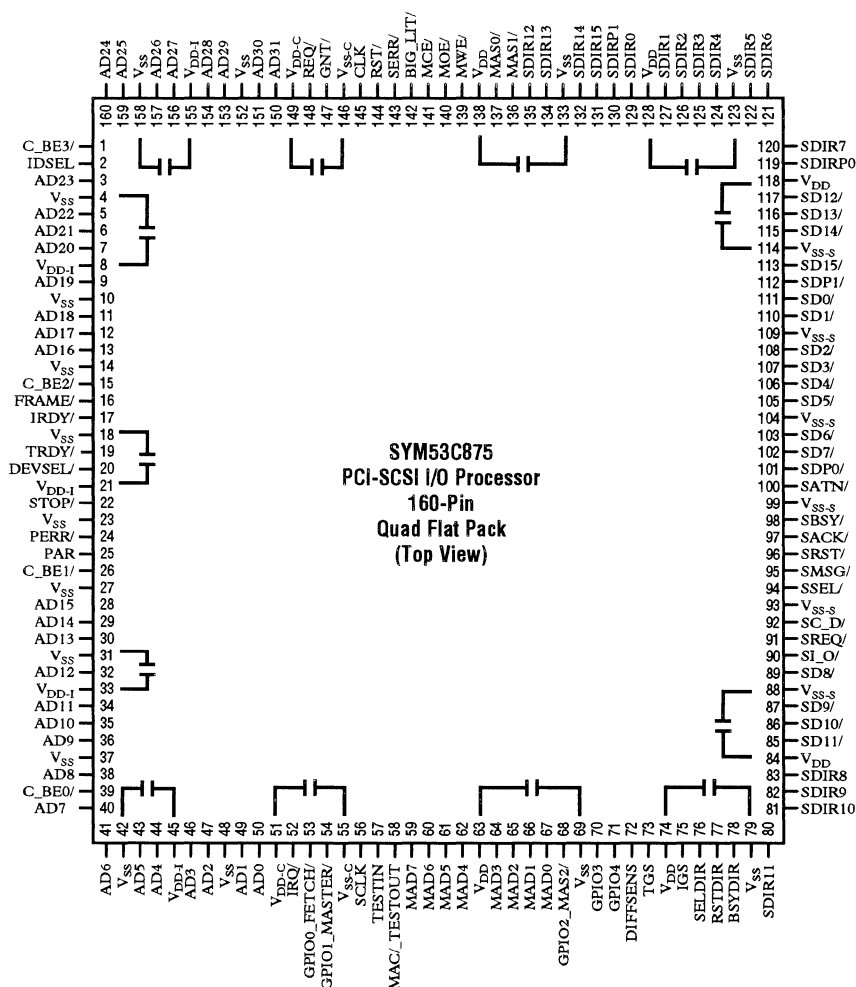
These registers are used to specify the desired settings for Latency Timer values. Min\_Gnt is used to specify how long a burst period the device needs. Max\_Lat is used to specify how often the device needs to gain access to the PCI bus. The value specified in these registers is in units of 0.25 microseconds. Values of zero indicate that the device has no major requirements for the settings of Latency Timers. The SYM53C875 sets the Min\_Gnt register to 11h and the Max\_Lat register to 40h.



## Chapter 4

# Signal Descriptions

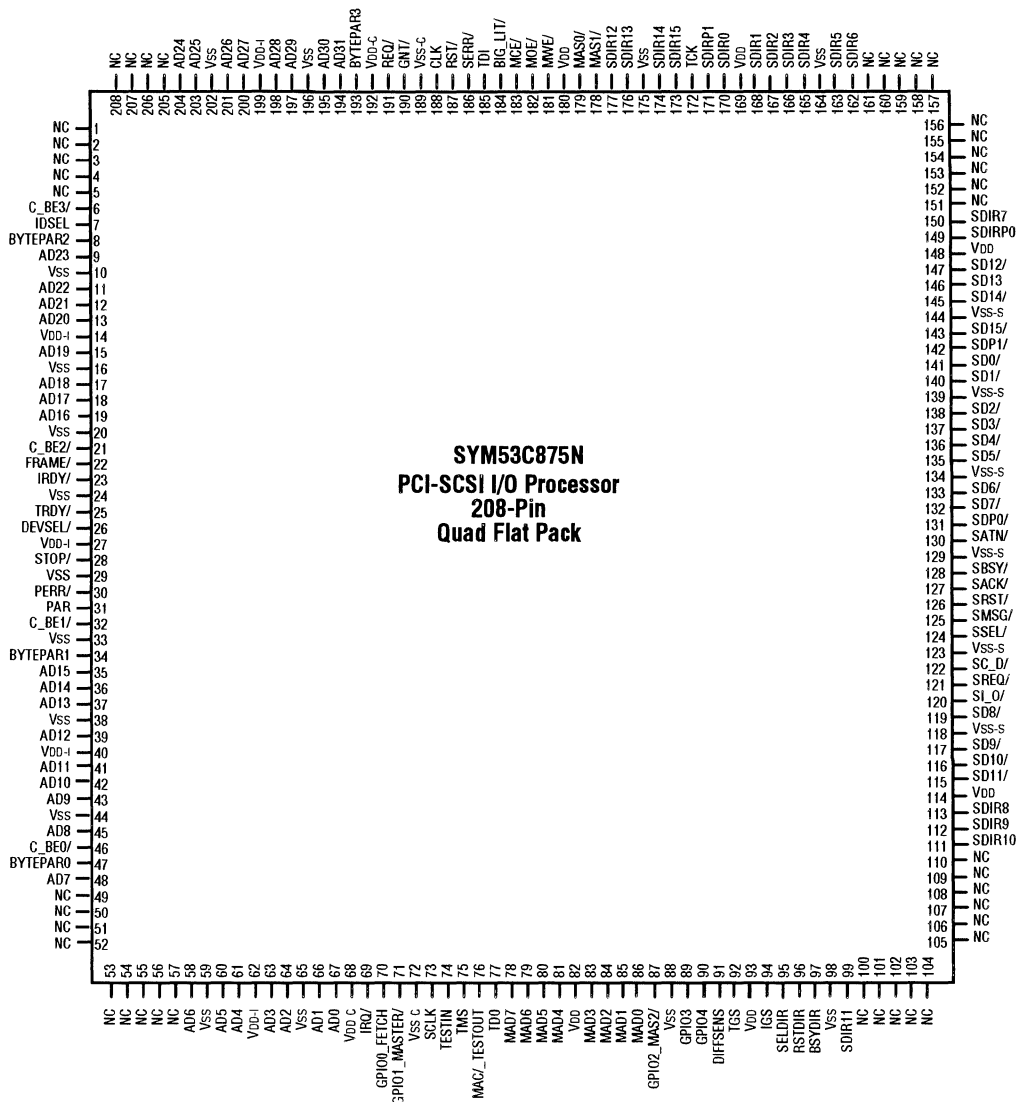
This chapter presents the SYM53C875 pin configuration and signal definitions using tables and illustrations. Figure 4-1 through Figure 4-4 are the pin diagrams for all versions of the SYM53C875 and Figure 4-5 is the functional signal grouping. The pins definitions are presented in Table 4-2 through Table 4-12. The SYM53C875 is a pin-for-pin replacement for the SYM53C825.



The decoupling capacitor arrangement shown above is recommended to maximize the benefits of the internal split ground system. Capacitor values between 0.01 and 0.1 $\mu$ F should provide adequate noise isolation. Because of the number of high current drivers on the SYM53C875, a multi-layer PC board with power and ground planes is required.

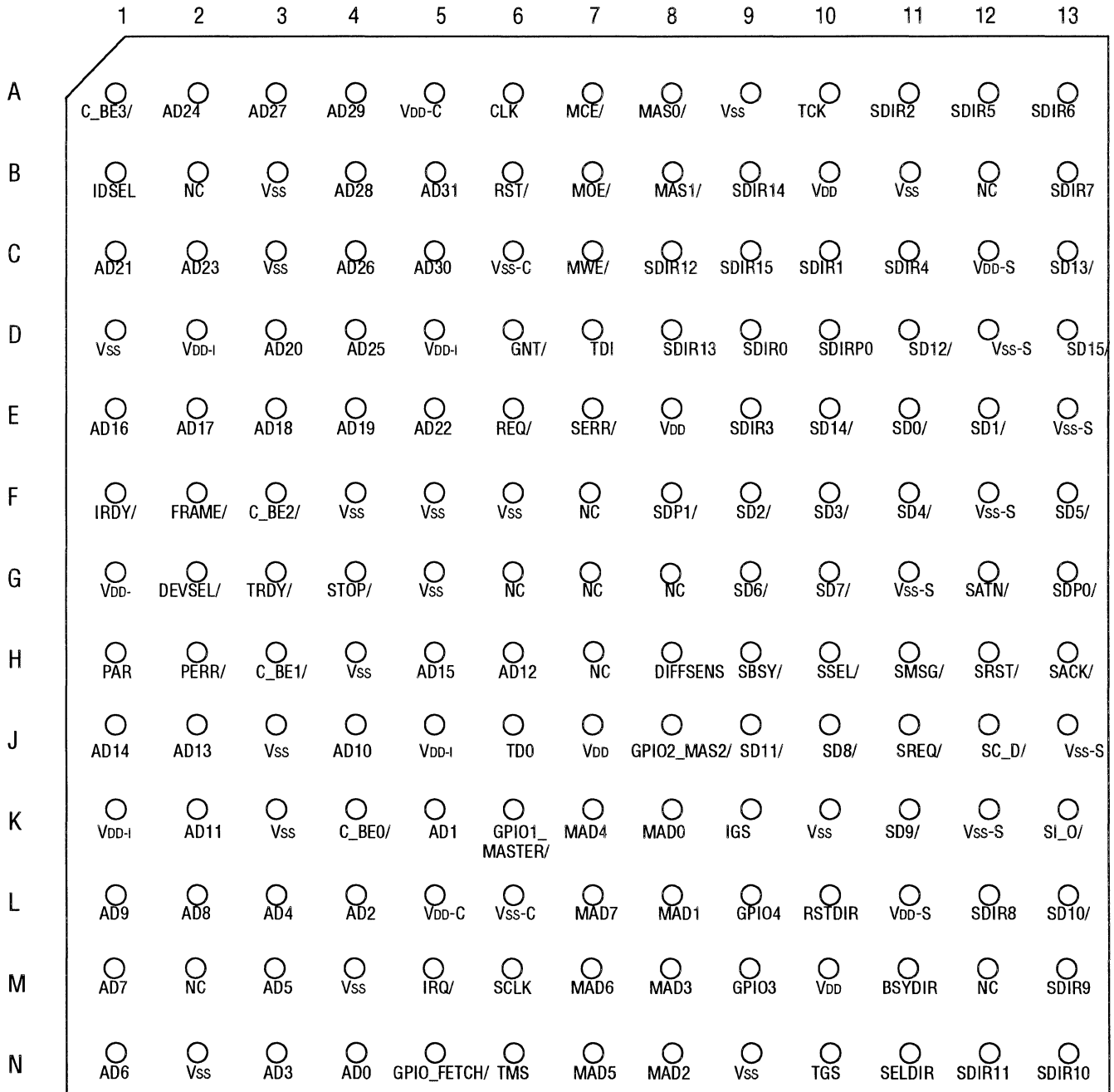
**Figure 4-1: SYM53C875 Pin Diagram**





The decoupling capacitor arrangement shown above is recommended to maximize the benefits of the internal split ground system. Capacitor values between 0.01 and 0.1µF should provide adequate noise isolation. Because of the number of high current drivers on the SYM53C875, a multi-layer PC board with power and ground planes is required.

**Figure 4-3: SYM53C875N Pin Diagram**



Note: Pins F7, G6, G7, G8, and H7 are connected to the die pad.

**Figure 4-4: SYM53C875JB Pin Diagram (Top View)**



The PCI/SCSI pin definitions are organized into the following functional groups: System, Address/Data, Interface Control, Arbitration, Error Reporting, SCSI, and Optional Interface. A slash (/) at the end of the signal name indicates that the active state occurs when the signal is at a low voltage. When the slash is absent, the signal is active at a high voltage.

There are four signal type definitions:

- I            Input, a standard input-only signal
- O            Totem Pole Output, a standard output driver
- T/S         Tri-State, a bi-directional, tri-state input/output pin
- S/T/S       Sustained Tri-state, an active low tri-state signal owned and driven by one and only one agent at a time

**Table 4-1: 53C875, 53C875J Power and Ground Pins**

Symbol	Pin No.	Description
$V_{SS}$	4, 10, 14, 18, 23, 27, 31, 37, 42, 48, 69, 79, 123, 133, 152, 158	Power supplies to the PCI I/O pins
$V_{DD}$	63, 74, 84, 118, 128, 138	Power supplies to the PCI I/O pins
$V_{DD-I^*}$	8, 21, 33, 45, 155	$V_{DD}$ pad for PCI bus drivers
$V_{SS-S}$	88, 93, 99, 104, 109, 114	Power supplies to the SCSI bus I/O pins
$V_{SS-C}$	55, 146	Power supplies to the internal logic core
$V_{DD-C}$	51, 149	Power supplies to the internal logic core

*\*These pins can accept a  $V_{DD}$  source of 3.3 or 5 Volts. All other  $V_{DD}$  pins must be supplied 5 Volts.*

**Table 4-2: 53C875N Power and Ground Pins**

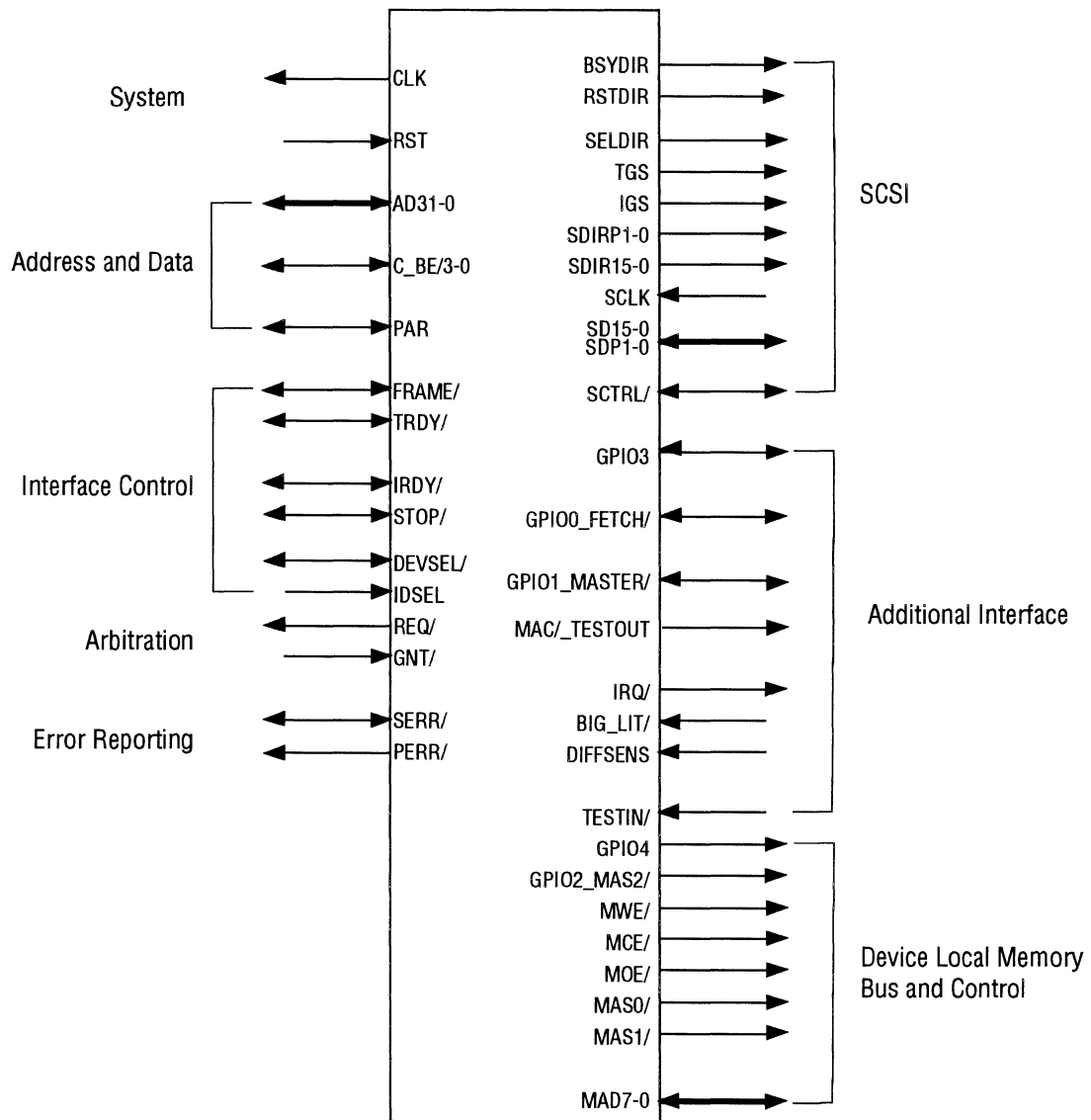
Symbol	Pin No.	Description
$V_{SS}$	10, 16, 20, 24, 29, 33, 38, 44, 59, 65, 88, 98, 164, 175, 196, 202	Power supplies to the PCI I/O pins
$V_{DD}$	82, 93, 148, 169, 180	Power supplies to the PCI I/O pins
$V_{DD-I^*}$	14, 27, 62, 199	$V_{DD}$ pad for PCI bus drivers
$V_{SS-S}$	118, 123, 129, 134, 139, 144	Power supplies to the SCSI bus I/O pins
$V_{SS-C}$	72, 189	Power supplies to the internal logic core
$V_{DD-C}$	68, 192	Power supplies to the internal logic core

*\*These pins can accept a  $V_{DD}$  source of 3.3 or 5Volts. All other  $V_{DD}$  pins must be supplied 5Volts.*

**Table 4-3: 53C875JB Power and Ground Pins**

Symbol	Pin No.	Description
$V_{SS}$	A9, B3, B11, C3, D1, F4, F5, F6, G5, H4, J3, K3, K10, M4, N2, N9	Power supplies to the PCI I/O pins
$V_{DD}$	B10, E8, J7, M10	Power supplies to the PCI I/O pins
$V_{DD-I^*}$	D2, D5, G1, J5, K1	$V_{DD}$ pad for PCI bus drivers
$V_{SS-S}$	D12, E13, F12, G11, J13, K12	Power supplies to the SCSI bus I/O pins
$V_{DD-S}$	C12, L11	Power supplies to the SCSI bus I/O pins
$V_{SS-C}$	L6, C6	Power supplies to the internal logic core
$V_{DD-C}$	A5, L5	Power supplies to the internal logic core

*\*These pins can accept a  $V_{DD}$  source of 3.3 or 5Volts. All other  $V_{DD}$  pins must be supplied 5Volts.*



**Figure 4-5: SYM53C875 Functional Signal Grouping**

Table 4-4: System Pins

Symbol	Pin No. 53C875, 53C875J/ 53C875N/ 53C875JB	Type	Description
CLK	145/188/A6	I	Clock provides timing for all transactions on the PCI bus and is an input to every PCI device. All other PCI signals are sampled on the rising edge of CLK, and other timing parameters are defined with respect to this edge. This clock can optionally be used as the SCSI core clock; however, the SYM53C875 will not be able to achieve fast SCSI transfer rates.
RST/	144/187/B6	I	Reset forces the PCI sequencer of each device to a known state. All t/s and s/t/s signals are forced to a high impedance state, and all internal logic is reset. The RST/ input is synchronized internally to the rising edge of CLK. The CLK input must be active while RST/ is active to properly reset the device.

Table 4-5: Address and Data Pins

Symbol	Pin No.	Type	Description
AD(31-0)	53C875, 53C875J: 150, 151, 153, 154, 156, 157, 159, 160, 3, 5, 6, 7, 9, 11, 12, 13, 28, 29, 30, 32, 34, 35, 36, 38, 40, 41, 43, 44, 46, 47, 49, 50 53C875N: 194, 195, 197, 198, 200, 201, 203, 204, 9, 11, 12, 13, 15, 17, 18, 19, 35, 36, 37, 39, 41, 42, 43, 45, 48, 58, 60, 61, 63, 64, 66, 67 53C875JB: B5, C5, A4, B4, A3, C4, D4, A2, C2, E5, C1, D3, E4, E3, E2, E1, H5, J1, J2, H6, K2, J4, L1, L2, M1, N1, M3, L3, N3, L4, K5, N4	T/S	Physical longword address and data are multiplexed on the same PCI pins. During the first clock of a transaction, AD(31-0) contain a physical address. During subsequent clocks, AD(31-0) contain data. A bus transaction consists of an address phase, followed by one or more data phases. PCI supports both read and write bursts. AD(7-0) define the least significant byte, and AD(31-24) define the most significant byte.

**Table 4-5: Address and Data Pins (Continued)**

Symbol	Pin No.	Type	Description
C_BE(3-0)/	53C875, 53C875J:1, 15, 26, 39 53C875N: 6, 21, 32, 46 53C875JB: A1,F3, H3, K4	T/S	Bus command and byte enables are multiplexed on the same PCI pins. During the address phase of a transaction, C_BE(3-0)/ define the bus command. During the data phase, C_BE(3-0)/ are used as byte enables. The byte enables determine which byte lanes carry meaningful data. C_BE(0)/ applies to byte 0, and C_BE(3)/ to byte 3.
PAR	53C875, 53C875J: 25 53C875N: 31 53C875JB: H1	T/S	Parity is the even parity bit that protects the AD(31-0) and C_BE(3-0)/ lines. During address phase, both the address and command bits are covered. During data phase, both data and byte enables are covered.
BYTEPAR3-0 (53C875N only)	193, 8, 34, 47	T/S	When the PCI Byte Parity pins are enabled, the SYM53C875N checks each byte of incoming data on the PCI bus against its corresponding parity line, in addition to the normal parity checking against the PCI PAR signal. This extra parity checking/generation is not register selectable; it is always enabled for the SYM53C875N. A parity error on any Byte Parity pin for PCI master read or slave write operations will cause a fatal DMA interrupt; SCRIPTS will stop running. This interrupt can be masked with the Extended Byte Parity Enable bit, bit 1 of the DIEN register.

Table 4-6: Interface Control Pins

Symbol	Pin No. 53C875, 53C875J/ 53C875N/ 53C875JB	Type	Description
FRAME/	16/22/F2	S/T/S	Cycle Frame is driven by the current master to indicate the beginning and duration of an access. FRAME/ is asserted to indicate a bus transaction is beginning. While FRAME/ is asserted, data transfers continue. When FRAME/ is deasserted, the transaction is in the final data phase or the bus is idle.
TRDY/	19/25/G3	S/T/S	Target Ready indicates the target agent's (selected device's) ability to complete the current data phase of the transaction. TRDY/ is used with IRDY/. A data phase is completed on any clock when both TRDY/ and IRDY/ are sampled asserted. During a read, TRDY/ indicates that valid data is present on AD(31-0). During a write, it indicates the target is prepared to accept data. Wait cycles are inserted until both IRDY/ and TRDY/ are asserted together.
IRDY/	17/23/F1	S/T/S	Initiator Ready indicates the initiating agent's (bus master's) ability to complete the current data phase of the transaction. This signal is used with TRDY/. A data phase is completed on any clock when both IRDY/ and TRDY/ are sampled asserted. During a write, IRDY/ indicates that valid data is present on AD(31-0). During a read, it indicates the master is prepared to accept data. Wait cycles are inserted until both IRDY/ and TRDY/ are asserted together.
STOP/	22/28/G4	S/T/S	Stop indicates that the selected target is requesting the master to stop the current transaction.
DEVSEL/	20/26/G2	S/T/S	Device Select indicates that the driving device has decoded its address as the target of the current access. As an input, it indicates to a master whether any device on the bus has been selected.
IDSEL	2/7/B1	I	Initialization Device Select is used as a chip select in place of the upper 24 address lines during configuration read and write transactions.

**Table 4-7: Arbitration Pins**

Symbol	Pin No. 53C875, 53C875J/ 53C875N/ 53C875JB	Type	Description
REQ/	148/191/E6	O	Request indicates to the arbiter that this agent desires use of the PCI bus. This is a point-to-point signal. Every master has its own REQ/.
GNT/	147/190/D6	I	Grant indicates to the agent that access to the PCI bus has been granted. This is a point-to-point signal. Every master has its own GNT/.

**Table 4-8: Error Reporting Pins**

Symbol	Pin No. 53C875, 53C875J/ 53C875N/ 53C875JB	Type	Description
PERR/	24/30/H2	S/T/S	Parity Error may be pulsed active by an agent that detects a data parity error. PERR/ can be used by any agent to signal data corruptions.
SERR/	143/186/E7	O	This open drain output pin is used to report address parity errors. On detection of a PERR/ pulse, the central resource may generate a non-maskable interrupt to the host CPU, which often implies the system will be unable to continue operation once error processing is complete.



Table 4-9: SCSI Pins

Symbol	Pin No. 53C875, 53C875J/ 53C875N/ 53C875JB	Type	Description
SCLK	56/73/M6	I	SCLK is used to derive all SCSI-related timings. The speed of this clock is determined by the application's requirements; in some applications SCLK may be sourced internally from the PCI bus clock (CLK). If SCLK is internally sourced, then the SCLK pin should be tied low.
SD(15-0)/, SDP(1-0)/	53C875, 53C875J: 113, 115, 116, 117, 85, 86, 87, 89, 102, 103, 105, 106, 107, 108, 110, 111, 112, 101 53C875N: 143, 145, 146, 147, 115, 116, 117, 119, 132, 133, 135, 136, 137, 138, 140, 141, 142, 131 53C875JB: D13, E10, C13, D11, J9, L13 K11, J10, G10, G9, F13, F11, F10, F9, E12, E11, F8, G13	I/O	SCSI Data includes the following data lines and parity signals: SD(15-0)/ (16-bit SCSI data bus), and SDP(1-0)/(SCSI data parity bits).
SCTRL/	53C875, 53C875J: 92, 90, 95, 91, 97, 98, 100, 96, 94 53C875N: 122, 120, 125, 121, 127, 128, 130, 126, 124 53C875JB: J12, K13, H11, J11, H13, H9, G12, H12, H10	I/O	SCSI Control includes the following signals: SC_D/ SCSI phase line, command/data SI_O/ SCSI phase line, input/output SMSG/ SCSI phase line, message SREQ/ Data handshake signal from target device SACK/ Data handshake signal from initiator device SBSY/ SCSI bus arbitration signal, busy SATN/ SCSI Attention, the initiator is requesting a message out phase SRST/ SCSI bus reset SSEL/ SCSI bus arbitration signal, select device

Table 4-9: SCSI Pins (Continued)

Symbol	Pin No. 53C875, 53C875J/ 53C875N/ 53C875JB	Type	Description
SDIR(15-0)	53C875, 53C875J: 131, 132, 134, 135, 80, 81, 82, 83, 120, 121, 122, 124, 125, 126, 127, 129  53C875N: 173, 174, 176, 177, 99, 111, 112, 113, 150, 162, 163, 165, 166, 167, 168, 170  53C875JB: C9, B9, D8, C8, N12, N13, M13, L12, B13, A13, A12, C11, E9, A11, C10, D9	O	Driver direction control for SCSI data lines
SDIRP(1-0) (SDIRP1 not available on 53C875J, 53C875JB)	130, 119;NA, 119/171, 149 / NA,D10	O	Driver direction control for SCSI parity signals In the SYM53C875J and SYM53C875JB, this pin is replaced by the TCK JTAG signal. If the device is used in a wide differential system, use the SDIRP0 pin to control the direction of the differential transceiver for both the SP0 and SP1 signals. The SDIRP0 signal is capable of driving both direction inputs from a transceiver.
SELDIR	76/95/N11	O	Driver enable control for SCSI SEL/ signal
RSTDIR	77/96/L10	O	Driver enable control for SCSI RST/ signal
BSYDIR	78/97/M11	O	Driver enable control for SCSI BSY/ signal
IGS	75/94/K9	O	Direction control for initiator driver group
TGS	73/92/N10	O	Direction control for target driver group

Table 4-10: Additional Interface Pins

Symbol	Pin No. 53C875, 53C875J/ 53C875N/53C875JB	Type	Description
TESTIN (Not available on 53C875J, 53C875JB)	57, NA/74/NA	I	Test In. When this pin is driven low, the SYM53C875 connects all inputs and outputs to an “AND tree.” The SCSI control signals and data lines are not connected to the “AND tree.” The output of the “AND tree” is connected to the Test Out pin. This allows manufacturers to verify chip connectivity and determine exactly which pins are not properly attached. When the TESTIN pin is driven low, internal pull-ups are enabled on all input, output, and bidirectional pins, all outputs and bidirectional signals will be tri-stated, and the MAC/_TESTOUT pin will be enabled. Connectivity can be tested by driving one of the SYM53C875 pins low. The MAC/_TESTOUT pin should respond by also driving low.
GPIO0_ FETCH/	53/70/N5	I/O	General Purpose I/O pin. Optionally, when driven low, indicates that the next bus request will be for an op code fetch. This pin powers up as a general purpose input.  This pin has two specific purposes in the Symbios SDMS software. SDMS uses it to toggle SCSI device LEDs, turning on the LED whenever the SYM53C875 is on the SCSI bus. SDMS drives this pin low to turn on the LED, or drives it high to turn off the LED. This signal can also be used as data I/O for serial EEPROM access. In this case it is used with the GPIO0 pin, which serves as a clock, and the pin can be controlled from PCI configuration register 35h or observed from the GPREG operating register, at address 07h.
GPIO1_ MASTER/	54/71/K6	I/O	General purpose I/O pin. Optionally, when driven low, indicates that the SYM53C875 is bus master. This pin powers up as a general purpose input.  Symbios SDMS software supports use of this signal in serial EEPROM applications, when enabled, in combination with the GPIO0 pin. When this signal is used as a clock for serial EEPROM access, the GPIO1 pin serves as data, and the pin is controlled from PCI configuration register 35h, or observed from the GPREG operating register at address 07h.
GPIO4-3	71, 70/90, 89/L9, M9	I/O	General purpose I/O pins. GPIO4 powers up as an output. It can be used as the enable line for $V_{PP}$ , the 12 Volt power supply to the external flash memory interface. GPIO3 powers up as an input.  Symbios SDMS software uses GPIO3 to detect a differential board. If the pin is pulled low externally, the board will be configured by SDMS as a differential board. If it is pulled high or left floating, SDMS will configure it as a single-ended board. The Symbios PCI to SCSI host adapters use the GPIO4 pin in the process of flashing a new SDMS ROM.

**Table 4-10: Additional Interface Pins (Continued)**

Symbol	Pin No. 53C875, 53C875J/ 53C875N/53C875JB	Type	Description
DIFFSENS	72/91/H8	I	The Differential Sense pin detects the presence of a single-ended device on a differential system. When external differential transceivers are used and a zero is detected on this pin, all chip SCSI outputs will be tri-stated to avoid damage to the transceivers. This pin should be tied high during single-ended operation. The normal value of this pin is 1.
MAC/_ TESTOUT  (Not available on 53C875, 53C875JB)	58, NA/76/NA	T/S	Memory Access Control. This pin can be programmed to indicate local or system memory accesses (non-PCI applications). It is also used to test the connectivity of the SYM53C875 signals using an “AND tree” scheme. The MAC/_TESTOUT pin is only driven as the Test Out function when the TESTIN/ pin is driven low.
IRQ/	52/69/M5	O	Interrupt. This signal, when asserted low, indicates that an interrupting condition has occurred and that service is required from the host CPU. The output drive of this pin is programmed as either open drain with an internal weak pull-up or, optionally, as a totem pole driver. Refer to the description of DCNTL Register, bit 3, for additional information.
BIG_LIT/  (Not available on 53C875J, 53C875JB)	142, NA/184/NA	I	Big_Little Endian Select. When this pin is driven low, the SYM53C875 will route the first byte of an aligned SCSI to PCI transfer to byte lane zero of the PCI bus and subsequent bytes received will be routed to ascending lanes. An aligned PCI to SCSI transfer will route PCI byte lane zero onto the SCSI bus first, and transfer ascending byte lanes in order. When this pin is driven high, the SYM53C875 will route the first byte of an aligned SCSI-to-PCI transfer to byte lane three of the PCI bus and subsequent bytes received will be routed to descending lanes. An aligned PCI-to-SCSI transfer will route PCI byte lane three onto the SCSI bus first and transfer descending byte lanes in order. This mode of operation also applies to the external memory interface. When this pin is driven in Little Endian mode and the chip is performing a read from external memory, the byte of data accessed at location 00000h will be routed to PCI byte lane zero and the data accessed at location 00003h will be routed to PCI byte lane three. When the chip is performing a write to flash memory, PCI byte lane zero will be routed to location 00000h and ascending byte lanes will be routed to subsequent memory locations. When this pin is driven in Big Endian mode and the chip is performing a read from external memory, the byte of data accessed at location 00000h will be routed to PCI byte lane three and the data accessed at location 00003h will be routed to byte lane zero. When the chip is performing a write to flash memory, PCI byte lane three will be routed to location 00000h and descending byte lanes will be routed to subsequent memory locations.

Table 4-11: External Memory Interface Pins

Symbol	Pin No. 53C875, 53C875J/ 53C875N/53C875JB	Type	Description
MAS0/	137/179/A8	O	Memory Address Strobe 0. This pin is used to latch in the least significant address byte of an external EPROM or flash memory. Since the SYM53C875 moves addresses eight bits at a time, this pin connects to the clock of an external bank of flip-flops which are used to assemble up to a 20-bit address for the external memory.
MAS1/	136/178/B8	O	Memory Address Strobe 1. This pin is used to latch in the address byte corresponding to address bits 15-8 of an external EPROM or flash memory. Since the SYM53C875 moves addresses eight bits at a time, this pin connects to the clock of an external bank of flip-flops which assemble up to a 20-bit address for the external memory.
MAD7-0	53C875, 53C875J: 59, 60, 61, 62, 64, 65, 66, 67 53C875N: 78, 79, 80, 81, 83, 84, 85, 86 53C875JB: L7, M7, N7, K7, M8, N8, L8, K8	I/O	Memory Address/Data Bus. This bus is used in conjunction with the memory address strobe pins and external address latches to assemble up to a 20-bit address for an external EPROM or flash memory. This bus will put out the most significant byte first and finish with the least significant bits. It is also used to write data to a flash memory or read data into the chip from external EPROM/flash memory.  The MAD5 pin enables the 4K internal RAM on the SYM53C875. To disable the internal RAM, connect a 4.7 K $\Omega$ resistor between the MAD5 pin and V <sub>SS</sub> . Connecting a 4.7 K $\Omega$ resistor between the MAD6 pin and V <sub>SS</sub> enables the Subsystem Data register in PCI configuration space, a read/write register that supports subsystem identification. Connecting a 4.7 K $\Omega$ resistor between MAD7 and V <sub>SS</sub> enables the Serial Control register in PCI configuration space, a 2-byte register that controls the GPIO1-0 pins when serial EEPROM access is enabled.
MWE/	139/181/C7	O	Memory Write Enable. This pin is used as a write enable signal to an external flash memory.
MOE/	140/182/B7	O	Memory Output Enable. This pin is used as an output enable signal to an external EPROM or flash memory during read operations.
MCE/	141/183/A7	O	Memory Chip Enable. This pin is used as a chip enable signal to an external EPROM or flash memory device
GPIO2_ MAS2/	68 /87/J8	I/O	General Purpose I/O pin. Optionally, this pin is used as a Memory Address Strobe 2 if an external memory with more than 16 bits of addressing is specified by the pull-down resistors at power up and bit 0 in the expansion ROM Base Address register is set.

Table 4-12: JTAG Pins 53C875J/53C875N/53C875JB Only)

Symbol	Pin No. 53C875J/53C875N/ 53C875JB	Type	Description
TCK	130/172/A10		Test Clock pin for JTAG boundary scan.
TMS	57/75/N6		Test Mode Select pin for JTAG boundary scan.
TDI	142/185/D7		Test Data In pin for JTAG boundary scan.
TDO	58/77/J6		Test Data Out pin for JTAG boundary scan.

## Chapter 5

# Operating Registers

This section contains descriptions of all SYM53C875 operating registers. Table 5-1 summarizes the SYM53C875 operating register set. Figure 5-1, the register map, lists registers by operating and configuration addresses. The terms “set” and “assert” are used to refer to bits that are programmed to a binary one. Similarly, the terms “deassert,” “clear” and “reset” are used to refer to bits that are programmed to a binary zero. Any bits marked as reserved should always be written to zero; mask all information read from them. Reserved bit functions may be changed at any time. Unless otherwise indicated, all bits in registers are active high, that is, the feature is enabled by setting the

bit. The bottom row of every register diagram shows the default register values, which are enabled after the chip is powered on or reset.

Note: the only register that the host CPU can access while the SYM53C875 is executing SCRIPTS is the ISTAT register; attempts to access other registers will interfere with the operation of the chip. However, all operating registers are accessible with SCRIPTS. All read data is synchronized and stable when presented to the PCI bus.

Note: the SYM53C875 cannot fetch SCRIPTS instructions from the operating register space. Instructions must be fetched from system memory or the internal SCRIPTS RAM.

**Table 5-1: Operating Register Addresses and Descriptions**

Memory or I/O Address	Config Memory or I/O	Read/Write	Label	Description
00	80	R/W	SCNTL0	SCSI Control 0
01	81	R/W	SCNTL1	SCSI Control 1
02	82	R/W	SCNTL2	SCSI Control 2
03	83	R/W	SCNTL3	SCSI Control 3
04	84	R/W	SCID	SCSI Chip ID
05	85	R/W	SXFER	SCSI Transfer
06	86	R/W	SDID	SCSI Destination ID
07	87	R/W	GPREG	General Purpose Bits
08	88	R/W	SFBR	SCSI First Byte Received
09	89	R/W	SOCL	SCSI Output Control Latch
0A	8A	R	SSID	SCSI Selector ID
0B	8B	R/W	SBCL	SCSI Bus Control Lines
0C	8C	R	DSTAT	DMA Status
0D	8D	R	SSTAT0	SCSI Status 0
0E	8E	R	SSTAT1	SCSI Status 1
0F	8F	R	SSTAT2	SCSI Status 2

**Table 5-1: Operating Register Addresses and Descriptions (Continued)**

<b>Memory or I/O Address</b>	<b>Config Memory or I/O</b>	<b>Read/Write</b>	<b>Label</b>	<b>Description</b>
10-13	90-93	R/W	DSA	Data Structure Address
14	94	R/W	ISTAT	Interrupt Status
18	98	R/W	CTEST0	Reserved
19	99	R/W	CTEST1	Chip Test 1
1A	9A	R	CTEST2	Chip Test 2
1B	9B	R	CTEST3	Chip Test 3
1C-1F	9C-9F	R/W	TEMP	Temporary Register
20	A0	R/W	DFIFO	DMA FIFO
21	A1	R/W	CTEST4	Chip Test 4
22	A2	R/W	CTEST5	Chip Text 5
23	A3	R/W	CTEST6	Chip Test 6
24-26	A4-A6	R/W	DBC	DMA Byte Counter
27	A7	R/W	DCMD	DMA Command
28-2B	A8-AB	R/W	DNAD	DMA Next Address for Data
2C-2F	AC-AF	R/W	DSP	DMA SCRIPTS Pointer
30-33	B0-B3	R/W	DSPS	DMA SCRIPTS Pointer Save
34-37	B4-B7	R/W	SCRATCHA	General Purpose Scratch Pad A
38	B8	R/W	DMODE	DMA Mode
39	B9	R/W	DIEN	DMA Interrupt Enable
3A	BA	R/W	SBR	Scratch Byte Register
3B	BB	R/W	DCNTL	DMA Control
3C-3F	BC-BF	R	ADDER	Sum output of internal adder
40	C0	R/W	SIEN0	SCSI Interrupt Enable 0
41	C1	R/W	SIEN1	SCSI Interrupt Enable 1
42	C2	R	SIST0	SCSI Interrupt Status 0
43	C3	R	SIST1	SCSI Interrupt Status 1
44	C4	R/W	SLPAR	SCSI Longitudinal Parity
45	C5	R	SWIDE	SCSI Wide Residue Data
46	C6	R/W	MACNTL	Memory Access Control
47	C7	R/W	GPCNTL	General Purpose Control
48	C8	R/W	STIME0	SCSI Timer 0
49	C9	R/W	STIME1	SCSI Timer 1
4A	CA	R/W	RESPID0	Response ID 0
4B	CB	R/W	RESPID1	Response ID 1



**Table 5-1: Operating Register Addresses and Descriptions (Continued)**

<b>Memory or I/O Address</b>	<b>Config Memory or I/O</b>	<b>Read/Write</b>	<b>Label</b>	<b>Description</b>
4C	CC	R	STEST0	SCSI Test 0
4D	CD	R	STEST1	SCSI Test 1
4E	CE	R/W	STEST2	SCSI Test 2
4F	CF	R/W	STEST3	SCSI Test 3
50-51	D0-D1	R	SIDL	SCSI Input Data Latch
52-53	D2-D3		Reserved	
54-55	D4-D5	R/W	SODL	SCSI Output Data Latch
56-57	D6-D7		Reserved	
58-59	D8-D9	R	SBDL	SCSI Bus Data Lines
5A-5B	DA-DB		Reserved	
5C-5F	DC-DF	R/W	SCRATCHB	General Purpose Scratch Pad B
60-7F	E0-FF	R/W	SCRATCHC-J	General Purpose Scratch Pad C-J

				Mem I/O	Config
SCNTL3	SCNTL2	SCNTL1	SCNTL0	00	80
GPREG	SDID	SXFER	SCID	04	84
SBCL	SSID	SOCL	SFBR	08	88
SSTAT2	SSTAT1	SSTAT0	DSTAT	0C	8C
DSA				10	90
RESERVED			ISTAT	14	94
CTEST3	CTEST2	CTEST1	RESERVED	18	98
TEMP				1C	9C
CTEST6	CTEST5	CTEST4	DFIFO	20	A0
DCMD	DBC			24	A4
DNAD				28	A8
DSP				2C	AC
DSPS				30	B0
SCRATCH A				34	B4
DCNTL	SBR	DIEN	DMODE	38	B8
ADDER				3C	BC
SIST1	SIST0	SIEN1	SIEN0	40	C0
GPCNTL	MACNTL	SWIDE	SLPAR	44	C4
RESPID1	RESPID0	STIME1	STIME0	48	C8
STEST3	STEST2	STEST1	STEST0	4C	CC
RESERVED		SIDL		50	D0
RESERVED		SODL		54	D4
RESERVED		SBDL		58	D8
SCRATCH B				5C	DC
SCRATCHC				60	E0
SCRATCHD				64	E4
SCRATCHE				68	E8
SCRATCHF				6C	EC
SCRATCHG				70	F0
SCRATCHH				74	F4
SCRATCHI				78	F8
SCRATCHJ				7F	FF

Figure 5-1: SYM53C875 Register Address Map

**Register 00 (80)**  
**SCSI Control Zero (SCNTL0)**  
 Read/Write

ARB1	ARB0	START	WATN	EPC	RES	AAP	TRG
7	6	5	4	3	2	1	0
Default>>>							
1	1	0	0	0	X	0	0

**Bit 7 ARB1 (Arbitration Mode bit 1)**

**Bit 6 ARB0 (Arbitration Mode bit 0)**

ARB1	ARB0	Arbitration Mode
0	0	Simple arbitration
0	1	Reserved
1	0	Reserved
1	1	Full arbitration, selection/reselection

**Simple Arbitration**

1. The SYM53C875 waits for a bus free condition to occur.
2. It asserts SBSY/ and its SCSI ID (contained in the SCID register) onto the SCSI bus. If the SSEL/ signal is asserted by another SCSI device, the SYM53C875 will deassert SBSY/, deassert its ID and set the Lost Arbitration bit (bit 3) in the SSTAT0 register.
3. After an arbitration delay, the CPU should read the SBDL register to check if a higher priority SCSI ID is present. If no higher priority ID bit is set, and the Lost Arbitration bit is not set, the SYM53C875 has won arbitration.
4. Once the SYM53C875 has won arbitration, SSEL/ must be asserted via the SOCL for a bus clear plus a bus settle delay (1.2 μs) before a low level selection can be performed.

**Full Arbitration, Selection/Reselection**

1. The SYM53C875 waits for a bus free condition.
2. It asserts SBSY/ and its SCSI ID (the highest priority ID stored in the SCID register) onto the SCSI bus.
3. If the SSEL/ signal is asserted by another SCSI device or if the SYM53C875 detects a higher priority ID, the SYM53C875 will deassert BSY, deassert its ID, and wait until the next bus free state to try arbitration again.
4. The SYM53C875 repeats arbitration until it wins control of the SCSI bus. When it has won, the Won Arbitration bit is set in the SSTAT0 register, bit 2.
5. The SYM53C875 performs selection by asserting the following onto the SCSI bus: SSEL/, the target's ID (stored in the SDID register), and the SYM53C875's ID (stored in the SCID register).
6. After a selection is complete, the Function Complete bit is set in the SIST0 register, bit 6.
7. If a selection time-out occurs, the Selection Time-Out bit is set in the SIST1 register, bit 2.

**Bit 5 START (Start Sequence)**

When this bit is set, the SYM53C875 will start the arbitration sequence indicated by the Arbitration Mode bits. The Start Sequence bit is accessed directly in low-level mode; during SCSI SCRIPTS operations, this bit is controlled by the SCRIPTS processor. An arbitration sequence should not be started if the connected (CON) bit in the SCNTL1 register, bit 4, indicates that the SYM53C875 is already connected to the SCSI bus. This bit is automatically cleared when the arbitration sequence is complete. If a sequence is aborted, bit 4 in the SCNTL1 register should be checked to verify that the SYM53C875 did not connect to the SCSI bus.

**Bit 4 WATN (Select with SATN/ on a Start Sequence)**

When this bit is set and the SYM53C875 is in initiator mode, the SATN/ signal will be asserted during SYM53C875 selection of a SCSI target device. This is to inform the target that the SYM53C875 has a message to send. If a selection time-out occurs while attempting to select a target device, SATN/ will be deasserted at the same time SSEL/ is deasserted. When this bit is clear, the SATN/ signal will not be asserted during selection. When executing SCSI SCRIPTS, this bit is controlled by the SCRIPTS processor, but it may be set manually in low level mode.

**Bit 3 EPC (Enable Parity Checking)**

When this bit is set, the SCSI data bus is checked for odd parity when data is received from the SCSI bus in either initiator or target mode. Parity is also checked as data goes from the SCSI FIFO to the DMA FIFO. If a parity error is detected, bit 0 of the SIST0 register is set and an interrupt may be generated.

If the SYM53C875 is operating in initiator mode and a parity error is detected, SATN/ can optionally be asserted, but the transfer continues until the target changes phase. When this bit is cleared, parity errors are not reported.

When these bits are set in the SYM53C875N, the chip again checks inbound SCSI parity at the SCSI FIFO—DMA FIFO interface after the data has passed through the SCSI FIFO. The parity bits are not passed through the DMA FIFO, but parity is generated before the data is sent out on the PCI bus.

**Bit 2 Reserved****Bit 1 AAP (Assert SATN/ on Parity Error)**

When this bit is set, the SYM53C875 automatically asserts the SATN/ signal upon detection of a parity error. SATN/ is only asserted in initiator mode. The SATN/ signal is asserted before deasserting SACK/ during the byte

transfer with the parity error. The Enable Parity Checking bit must also be set for the SYM53C875 to assert SATN/ in this manner. A parity error is detected on data received from the SCSI bus.

If the Assert SATN/ on Parity Error bit is cleared or the Enable Parity Checking bit is cleared, SATN/ will not be automatically asserted on the SCSI bus when a parity error is received.

**Bit 0 TRG (Target Mode)**

This bit determines the default operating mode of the SYM53C875. The user must manually set target or initiator mode. This can be done using the SCRIPTS language (SET TARGET or CLEAR TARGET). When this bit is set, the chip is a target device by default. When this bit is cleared, the SYM53C875 is an initiator device by default.

**CAUTION:**

Writing this bit while not connected may cause the loss of a selection or reselection due to the changing of target or initiator modes.

**Register 01 (81)**  
**SCSI Control One (SCNTL1)**  
 Read/Write

EXC	ADB	DHP	CON	RST	AESP	IARB	SST
7	6	5	4	3	2	1	0
Default>>>							
0	0	0	0	0	0	0	0

**Bit 7 EXC (Extra Clock Cycle of Data Setup)**

When this bit is set, an extra clock period of data setup is added to each SCSI send data transfer. The extra data setup time can provide additional system design margin, though it will affect the SCSI transfer rates. Clearing this bit disables the extra clock cycle of data setup time. Setting this bit only affects SCSI send operations.

**Bit 6 ADB (Assert SCSI Data Bus)**

When this bit is set, the SYM53C875 drives the contents of the SCSI Output Data Latch Register (SODL) onto the SCSI data bus. When the SYM53C875 is an initiator, the SCSI I/O signal must be inactive to assert the SODL contents onto the SCSI bus. When the SYM53C875 is a target, the SCSI I/O signal must be active for the SODL contents to be asserted onto the SCSI bus. The contents of the SODL register can be asserted at any time, even before the SYM53C875 is connected to the SCSI bus. This bit should be cleared when executing SCSI SCRIPTS. It is normally used only for diagnostics testing or operation in low level mode.

**Bit 5 DHP (Disable Halt on Parity Error or ATN) (Target Only)**

The DHP bit is only defined for target mode. When this bit is cleared, the SYM53C875 halts the SCSI data transfer when a parity error is detected or when the SATN/ signal is asserted. If SATN/ or a parity error is received in the middle of a data transfer, the SYM53C875 may transfer up to three additional bytes before halting to synchronize between internal core

cells. During synchronous operation, the SYM53C875 transfers data until there are no outstanding synchronous offsets. If the SYM53C875 is receiving data, any data residing in the DMA FIFO is sent to memory before halting.

When this bit is set, the SYM53C875 does not halt the SCSI transfer when SATN/ or a parity error is received.

**Bit 4 CON (Connected)**

This bit is automatically set any time the SYM53C875 is connected to the SCSI bus as an initiator or as a target. It is set after the SYM53C875 successfully completes arbitration or when it has responded to a bus initiated selection or reselection. This bit is also set after the chip wins simple arbitration when operating in low level mode. When this bit is clear, the SYM53C875 is not connected to the SCSI bus.

The CPU can force a connected or disconnected condition by setting or clearing this bit. This feature would be used primarily during loopback mode.

**Bit 3 RST (Assert SCSI RST/ Signal)**

Setting this bit asserts the SRST/ signal. The SRST/ output remains asserted until this bit is cleared. The 25  $\mu$ s minimum assertion time defined in the SCSI specification must be timed out by the controlling microprocessor or a SCRIPTS loop.

**Bit 2 AESP (Assert Even SCSI Parity (force bad parity))**

When this bit is set, the SYM53C875 asserts even parity. It forces a SCSI parity error on each byte sent to the SCSI bus from the SYM53C875. If parity checking is enabled, then the SYM53C875 checks data received for odd parity. This bit is used for diagnostic testing and should be clear for normal operation. It can be used to generate parity errors to test error handling functions.

**Bit 1 IARB (Immediate Arbitration)**

Setting this bit causes the SCSI core to immediately begin arbitration once a Bus Free phase is detected following an expected SCSI disconnect. This bit is useful for multi-threaded applications. The ARB1-0 bits in SCNTL0 should be set for full arbitration and selection before setting this bit.

Arbitration will be re-tried until won. At that point, the SYM53C875 will hold BSY and SEL asserted, and wait for a select or reselect sequence to be requested. The Immediate Arbitration bit will be reset automatically when the selection or reselection sequence is completed, or times out.

An unexpected disconnect condition will clear IARB without attempting arbitration. See the SCSI Disconnect Unexpected bit (SCNTL2, bit 7) for more information on expected versus unexpected disconnects.

An immediate arbitration sequence can be aborted. First, the Abort bit in the ISTAT register should be set. Then one of two things will eventually happen:

1. The Won Arbitration bit (SSTAT0 bit 2) will be set. In this case, the Immediate Arbitration bit needs to be reset. This will complete the abort sequence and disconnect the SYM53C875 from the SCSI bus. If it is not acceptable to go to Bus Free phase immediately following the arbitration phase, a low level selection may be performed instead.
2. The abort will complete because the SYM53C875 loses arbitration. This can be detected by the Immediate Arbitration bit being cleared. The Lost Arbitration bit (SSTAT0 bit 3) should not be used to detect this condition. No further action needs to be taken in this case.

**Bit 0 SST (Start SCSI Transfer)**

This bit is automatically set during SCRIPTS execution, and should not be used. It causes the SCSI core to begin a SCSI transfer, including SREQ/SACK handshaking. The determination of whether the transfer is a send or receive is made according to the value written to the I/O bit in SOCL. This bit is self-resetting. It should not be set for low level operation.

**CAUTION:**

Writing to this register while not connected may cause the loss of a selection/reselection by resetting the Connected bit.

**Register 02 (82)**  
**SCSI Control Two (SCNTL2)**  
 Read/Write

SDU	CHM	SLPMD	SLPHBEN	WSS	VUE0	VUE1	WSR
7	6	5	4	3	2	1	0
Default>>>							
0	0	0	0	0	0	0	0

**Bit 7 SDU (SCSI Disconnect Unexpected)**

This bit is valid in initiator mode only. When this bit is set, the SCSI core is not expecting the SCSI bus to enter the Bus Free phase. If it does, an unexpected disconnect error will be generated (see the Unexpected Disconnect bit in the SIST0 register, bit 2). During normal SCRIPTS mode operation, this bit is set automatically whenever the SCSI core is reselected, or successfully selects another SCSI device.

The SDU bit should be reset with a register write (MOVE 0X00 TO SCNTL2) before the SCSI core expects a disconnect to occur, normally prior to sending an Abort, Abort Tag, Bus Device Reset, Clear Queue or Release Recovery message, or before deasserting SACK/ after receiving a Disconnect command or Command Complete message.

**Bit 6 CHM (Chained Mode)**

This bit determines whether or not the SCSI core is programmed for chained SCSI mode. This bit is automatically set by the Chained Block Move (CHMOV) SCRIPTS instruction and is automatically cleared by the Block Move SCRIPTS instruction (MOVE).

Chained mode is primarily used to transfer consecutive wide data blocks. Using chained mode facilitates partial receive transfers and allows correct partial send behavior. When this bit is set and a data transfer ends on an odd byte boundary, the SYM53C875 will store the last byte in the SCSI Wide Residue Data Register during a receive operation, or in the SCSI Output Data Latch register during a send operation. This byte will be combined with the first byte from the subsequent transfer so that a wide transfer can be completed.

For more information, see the “Chained Mode” section in Chapter 2, “Functional Description.”

**Bit 5 SLPMD (SLPAR Mode Bit)**

If this bit is clear, the SLPAR register functions like the SYM53C825. If this bit is set, the SLPAR register reflects the high or low byte of the SLPAR word, depending on the state of SCNTL2 bit 4. It also allows a seed value to be written to the SLPAR register.

**Bit 4 SLPHBEN (SLPAR High Byte Enable)**

If this bit is clear, the low byte of the SLPAR word is present in the SLPAR register. If this bit is set, the high byte of the SLPAR word is present in the SLPAR register.

**Bit 3 WSS (Wide SCSI Send)**

When read, this bit returns the value of the Wide SCSI Send (WSS) flag. Asserting this bit will clear the WSS flag. This clearing function is self-resetting.

When the WSS flag is high following a wide SCSI send operation, the SCSI core is holding a byte of “chain” data in the SODL register. This data will become the first low-order byte sent when married with a high-order byte during a subsequent data send transfer.

Performing a SCSI receive operation will clear this bit. Also, performing any non-wide transfer will clear this bit.

**Bit 2 VUE0 (Vendor Unique Enhancements bit 0)**

This bit is a read only value indicating whether the group code field in the SCSI instruction is standard or vendor unique. If reset, the bit indicates standard group codes; if set, the bit indicates vendor unique group codes. The value in this bit is reloaded at the beginning of all asynchronous target receives. The default for this bit is reset.

**Bit 1 VUE1 (Vendor Unique Enhancements bit 1)**

This bit is used to disable the automatic byte count reload during Block Move instructions in the command phase. If this bit is reset, the device will reload the Block Move byte count if the first byte received is one of the standard group codes. If this bit is set, the device will not reload the Block Move byte count, regardless of the group code.

**Bit 0 WSR (Wide SCSI Receive)**

When read, this bit returns the value of the Wide SCSI Receive (WSR) flag. Setting this bit clears the WSR flag. This clearing function is self-resetting.

The WSR flag indicates that the SCSI core received data from the SCSI bus, detected a possible partial transfer at the end of a chained or non-chained block move command, and temporarily stored the high-order byte in the SWIDE register rather than passing the byte out the DMA channel. The hardware uses the WSR status flag to determine what behavior must occur at the start of the next data receive transfer. When the flag is set, the stored data in SWIDE may be “residue” data, valid data for a subsequent data transfer, or overrun data. The byte may be read as normal data by starting a data receive transfer.

Performing a SCSI send operation will clear this bit. Also, performing any non-wide transfer will clear this bit.

**Register 03 (83)  
SCSI Control Three (SCNTL3)  
Read/Write**

F20	SCF2	SCF1	SCF0	EWS	CCF2	CCF1	CCF0
7	6	5	4	3	2	1	0
Default>>>							
0	0	0	0	0	0	0	0

**Bit 7 F20 (Fast-20 Enable)**

Setting this bit enables UltraSCSI synchronous SCSI transfers in systems that have an 80 MHz clock. The default value of this bit is 0. This bit should remain cleared in systems that have a 40MHz clock, unless the SCSI clock doubler is used to increase the SCLK frequency to at least 80 MHz.

When this bit is set, the signal filtering period for SREQ/ and SACK/ automatically changes to 15 ns, regardless of the value of the Extend REQ/ACK Filtering bit in the STEST2 register.

**Bits 6-4 SCF2-0 ( Synchronous Clock Conversion Factor)**

These bits select a factor by which the frequency of SCLK is divided before being presented to the synchronous SCSI control logic. They should be written to the same value as the Clock Conversion Factor bits below unless fast SCSI operation is desired. See the SCSI Transfer (SXFER) register description for examples of how the SCF bits are used to calculate synchronous transfer periods. See the table under the description of bits 7-5 of the SXFER register for the valid combinations.

**Note:** to migrate from a Fast SCSI-2 system with a 40MHz clock, divide the clock by a factor of two or more to achieve the same synchronous transfer rate in a system with an 80MHz clock.

**Note:** for additional information on how the synchronous transfer rate is determined, refer to Chapter 2.



**Bit 3 EWS (Enable Wide SCSI)**

When this bit is clear, all information transfer phases are assumed to be eight bits, transmitted on SD7-0/, SDP0/. When this bit is asserted, data transfers are done 16 bits at a time, with the least significant byte on SD7-0/, SDP/ and the most significant byte on SD15-8/, SDP1/. Command, Status, and Message phases are not affected by this bit.

Clearing this bit will also clear the Wide SCSI Receive bit in the SWIDE register.

**Bits 2-0 CCF2-0 (Clock Conversion Factor)**

These bits select a factor by which the frequency of SCLK is divided before being presented to the SCSI core. The synchronous portion of the SCSI core can be run at a different clock rate for fast SCSI, using the Synchronous Clock Conversion Factor bits. The bit encoding is displayed in the table below. All other combinations are reserved and should never be used.

SCF2 CCF2	SCF1 CCF1	SCF0 CCF0	Factor Frequency	SCSI Clock (MHz)
0	0	0	SCLK/3	50.01-75
0	0	1	SCLK/1	16.67-25
0	1	0	SCLK/1.5	25.01-37.5
0	1	1	SCLK/2	37.51-50
1	0	0	SCLK/3	50.01-75
1	0	1	SCLK/4	75.01-80.00
1	1	0	Reserved	
1	1	1	Reserved	

Note: it is important that these bits be set to the proper values to guarantee that the SYM53C875 meets the SCSI timings as defined by the ANSI specification.

Note: for additional information on how the synchronous transfer rate is determined, refer to Chapter 2, "Functional Description."

Note: to migrate from a Fast SCSI-2 system with a 40MHz clock, divide the clock by a factor of two or more to achieve the same synchronous transfer rate in a system with an 80MHz clock.

Note: if the SCSI clock doubler is enabled, use the desired frequency after doubling to determine the conversion factor.

**Register 04 (84)**  
**SCSI Chip ID (SCID)**  
 Read/Write

RES	RRE	SRE	RES	ENC3	ENC2	ENC1	ENC0
7	6	5	4	3	2	1	0
Default>>>							
X	0	0	X	0	0	0	0

**Bit 7 Reserved**

**Bit 6 RRE (Enable Response to Reselection)**

When this bit is set, the SYM53C875 is enabled to respond to bus-initiated reselection at the chip ID in the RESPID0 and RESPID1 registers. Note that the SYM53C875 will not automatically reconfigure itself to initiator mode as a result of being reselected.

**Bit 5 SRE (Enable Response to Selection)**

When this bit is set, the SYM53C875 is able to respond to bus-initiated selection at the chip ID in the RESPID0 and RESPID1 registers. Note that the SYM53C875 will not automatically reconfigure itself to target mode as a result of being selected.

**Bit 4 Reserved**

**Bits 3-0 Encoded Chip SCSI ID, bits 3-0**

These bits are used to store the SYM53C875 encoded SCSI ID. This is the ID which the chip will assert when arbitrating for the SCSI bus. The IDs that the SYM53C875 will respond to when being selected or reselected are configured in the RESPID0 and RESPID1 registers. The priority of the 16 possible IDs, in descending order is:

Highest								Lowest							
7	6	5	4	3	2	1	0	15	14	13	12	11	10	9	8

**Register 05 (85)**  
**SCSI Transfer (SXFER)**  
 Read/Write

TP2	TP1	TP0	M04	M03	M02	M01	M00
7	6	5	4	3	2	1	0
Default>>>							
0	0	0	0	0	0	0	0

Note: when using Table Indirect I/O commands, bits 7-0 of this register will be loaded from the I/O data structure.

Note: for additional information on how the synchronous transfer rate is determined, refer to Chapter 2, "Functional Description."

**Bits 7-5 TP2-0 (SCSI Synchronous Transfer Period)**

These bits determine the SCSI synchronous transfer period used by the SYM53C875 when sending synchronous SCSI data in either initiator or target mode. These bits control the programmable dividers in the chip.

Note: for UltraSCSI transfers, the ideal transfer period is 4, and 5 is acceptable. Setting the transfer period to a value greater than 5 is not recommended.

TP2	TP1	TP0	XFERP
0	0	0	4
0	0	1	5
0	1	0	6
0	1	1	7
1	0	0	8
1	0	1	9
1	1	0	10
1	1	1	11

The synchronous transfer period the SYM53C875 should use when transferring SCSI data is determined as in this example. The SYM53C875 is connected to a hard disk which can transfer data at 10 MB/s synchronously. The SYM53C875's SCLK is running at 40 MHz. The synchronous transfer period (SXFERP) is found as follows:

$$\text{SXFERP} = \text{Period}/\text{SSCP} + \text{ExtCC}$$

$$\text{Period} = 1 + \text{Frequency} = 1 + 10 \text{ MB/s} = 100 \text{ ns}$$

$$\text{SSCP} = 1 + \text{SSCF} = 1 + 40 \text{ MHz} = 25 \text{ ns}$$

(This SCSI synchronous core clock is determined in SCNTL3 bits 6-4

ExtCC = 1 if SCNTL1 bit 7 is asserted and the SYM53C875 is sending data.

ExtCC = 0 if the SYM53C875 is receiving data)

$$\text{SXFERP} = 100 + 25 = 4$$

**Key:**

SXFERP = Synchronous transfer period

SSCP = SCSI Synchronous core period

SSCF = SCSI Synchronous core frequency

ExtCC = Extra clock cycle of data setup

**Table 5-2: Examples of Synchronous Transfer Periods for SCSI-1 Transfer Rates**

CLK (MHz)	SCSI CLK + SCNTL3 bits 6-4	XFERP	Synch. Transfer Period (ns)	Synch. Send Rate (MB/s)	Synch. Receive Rate (MB/s)
80	+4	4	200	5	5
80	+4	5	250	4	5
66.67	+3	4	180	5.55	5.55
66.67	+3	5	225	4.44	5.55
50	+2	4	160	6.25	6.25
50	+2	5	200	5	6.25
40	+2	4	200	5	5
37.50	+1.5	4	160	6.25	6.25
33.33	+1.5	4	180	5.55	5.55

**Table 5-2: Examples of Synchronous Transfer Periods for SCSI-1 Transfer Rates**

CLK (MHz)	SCSI CLK + SCNTL3 bits 6-4	XFERP	Synch. Transfer Period (ns)	Synch. Send Rate (MB/s)	Synch. Receive Rate (MB/s)
25	+1	4	160	6.25	6.25
20	+1	4	200	5	5
16.67	+1	4	240	4.17	4.17

**Table 5-3: Example Transfer periods for fast SCSI-2 and Fast-20 transfer rates**

CLK (MHz)	SCSI CLK + SCNTL3 bits 6-4	XFERP	Synch. Transfer Period (ns)	Synch. Send Rate (MB/s)	Synch. Receive Rate (MB/s)
80	+1	4	50	20.0	20.0
80	+2	4	100	10.0	10.0
66.67	+1.5	4	90	11.11	11.11
66.67	+1.5	5	112.5	8.88	11.11
50	+1	4	80	12.5	12.5
50	+1	5	100	10.0	12.5
40	+1	4	100	10.0	10.0
37.50	+1	4	106.67	9.375	9.375
33.33	+1	4	120	8.33	8.33
25	+1	4	160	6.25	6.25
20	+1	4	200	5	5
16.67	+1	4	240	4.17	4.17

**Bits 4-0 MO4-MO0 (Max SCSI Synchronous Offset)**

These bits describe the maximum SCSI synchronous offset used by the SYM53C875 when transferring synchronous SCSI data in either initiator or target mode. The following table describes the possible combinations and their relationship to the synchronous data offset used by the SYM53C875. These bits determine the SYM53C875's method of transfer for Data In and Data Out phases only; all other information transfers will occur asynchronously.

M04	M03	M02	M01	M00	Synchronous Offset
0	0	0	0	0	0-Asynchronous
0	0	0	0	1	1
0	0	0	1	0	2
0	0	0	1	1	3
0	0	1	0	0	4
0	0	1	0	1	5
0	0	1	1	0	6
0	0	1	1	1	7
0	1	0	0	0	8
0	1	0	0	1	9
0	1	0	1	0	10
0	1	0	1	1	11
0	1	1	0	0	12
0	1	1	0	1	13
0	1	1	1	0	14
0	1	1	1	1	15
1	0	0	0	0	16
1	X	X	X	1	Reserved
1	X	X	1	X	Reserved
1	X	1	X	X	Reserved
1	1	X	X	X	Reserved

**Register 06 (86)**

**SCSI Destination ID (SDID)**

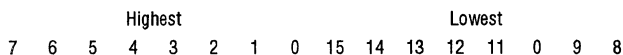
Read/Write

RES	RES	RES	RES	ENC3	ENC2	ENC1	ENC0
7	6	5	4	3	2	1	0
Default>>>							
X	X	X	X	0	0	0	0

**Bits 7-4 Reserved**

**Bits 3-0 Encoded Destination SCSI ID**

Writing these bits sets the SCSI ID of the intended initiator or target during SCSI reselection or selection phases, respectively. When executing SCRIPTS, the SCRIPTS processor writes the destination SCSI ID to this register. The SCSI ID is defined by the user in a SCRIPTS SELECT or RESELECT instruction. The value written should be the binary-encoded ID value. The priority of the 16 possible IDs, in descending order, is:



## Register 07 (87)

### General Purpose (GPREG)

#### Read/Write

RES	RES	RES	GPIO4	GPIO3	GPIO2	GPIO1	GPIO0
7	6	5	4	3	2	1	0
Default>>>							
X	X	X	0	X	X	X	X

#### Bits 7-5 Reserved

#### Bits 4-0 GPIO4-GPIO0 (General Purpose)

These bits can be programmed through the GPCNTL Register to become inputs, outputs or to perform special functions. As an output, these pins can be used to enable or disable external terminators. These signals can also be programmed as live inputs and sensed through a SCRIPTS Register to Register Move Instruction. GPIO(3-0) default as inputs and GPIO4 defaults as an output pin. When configured as inputs, an internal pull-up is enabled.

GPIO4 can be used to enable or disable  $V_{PP}$ , the 12-volt power supply to the external flash memory. This bit powers up with the power to the external memory disabled.

The GPIO1-0 signals can also be controlled from PCI configuration register 35h. They may be read, but not controlled, from this register.

#### SDMS Use of GPIO Pins

Symbios Logic SDMS software uses GPIO3 to detect a differential board. If the pin is pulled low externally, the board will be configured by SDMS as a differential board. If it is pulled high or left floating, SDMS will configure it as a single-ended board. The Symbios PCI to SCSI host adapters use the GPIO4 pin in the process of flashing a new SDMS ROM.

Symbios SDMS software uses the GPIO0 pin to toggle SCSI device LEDs, turning on the LED whenever the SYM53C875 is on the SCSI bus. SDMS drives this pin low to turn on the LED, or drives it high to turn off the LED.

SDMS uses the GPIO1-0 pins to support serial EEPROM access. When serial EEPROM access is enabled, GPIO1 is used as a clock and GPIO0 is used as data. The pins are controlled from PCI configuration register 35h. They may be read, but not controlled, from this register.

**Register 08 (88)**  
**SCSI First Byte Received (SFBR)**  
 Read/Write

1B7	1B6	1B5	1B4	1B3	1B2	1B1	1B0
7	6	5	4	3	2	1	0
Default>>>							
0	0	0	0	0	0	0	0

This register contains the first byte received in any asynchronous information transfer phase. For example, when the SYM53C875 is operating in initiator mode, this register contains the first byte received in the Message In, Status, and Data In phases.

When a Block Move instruction is executed for a particular phase, the first byte received is stored in this register— even if the present phase is the same as the last phase. The first byte received value for a particular input phase is not valid until after a MOVE instruction is executed.

This register is also the accumulator for register read-modify-writes with the SFBR as the destination. This allows bit testing after an operation.

The SFBR is not writable via the CPU, and therefore not by a Memory Move. The Load instruction may not be used to write to this register. However, it can be loaded via SCRIPTS Read/Write operations. To load the SFBR with a byte stored in system memory, the byte must first be moved to an intermediate SYM53C875 register (such as the SCRATCH register), and then to the SFBR.

This register will also contain the state of the lower eight bits of the SCSI data bus during the selection phase if the COM bit in the DCNTL register is clear.

**Register 09 (89)**  
**SCSI Output Control Latch (SOCL)**  
 Read /Write

REQ	ACK	BSY	SEL	ATN	MSG	C/D	I/O
7	6	5	4	3	2	1	0
Default>>>							
0	0	0	0	0	0	0	0

**Bit 7 REQ(Assert SCSI REQ/ Signal)**

**Bit 6 ACK(Assert SCSI ACK/ Signal)**

**Bit 5 BSY(Assert SCSI BSY/ Signal)**

**Bit 4 SEL(Assert SCSI SEL/ Signal)**

**Bit 3 ATN(Assert SCSI ATN/ Signal)**

**Bit 2 MSG(Assert SCSI MSG/ Signal)**

**Bit 1 C/D(Assert SCSI C\_D/ Signal)**

**Bit 0 I/O(Assert SCSI I\_O/ Signal)**

This register is used primarily for diagnostic testing or programmed I/O operation. It is controlled by the SCRIPTS processor when executing SCSI SCRIPTS. SOCL should only be used when transferring data via programmed I/O. Some bits are set (1) or reset (0) when executing SCSI SCRIPTS. Do not write to the register once the SYM53C875 starts executing normal SCSI SCRIPTS.

**Register 0A (09)**  
**SCSI Selector ID (SSID)**  
**Read Only**

VAL	RES	RES	RES	ENID3	ENID2	ENID1	ENID0
7	6	5	4	3	2	1	0
Default>>>							
0	X	X	X	0	0	0	0

**Bit 7 VAL (SCSI Valid)**

If VAL is asserted, the two SCSI IDs were detected on the bus during a bus-initiated selection or reselection, and the encoded destination SCSI ID bits below are valid. If VAL is deasserted, only one ID was present and the contents of the encoded destination ID are meaningless.

**Bits 6-4 Reserved**

**Bits 3-0 Encoded Destination SCSI ID**

Reading the SSID register immediately after the SYM53C875 has been selected or reselected returns the binary-encoded SCSI ID of the device that performed the operation. These bits are invalid for targets that are selected under the single initiator option of the SCSI-1 specification. This condition can be detected by examining the VAL bit above.

**Register 0B (8B)**  
**SCSI Bus Control Lines (SBCL)**  
**Read Only**

REQ	ACK	BSY	SEL	ATN	MSG	C/D	I/O
7	6	5	4	3	2	1	0
Default>>>							
X	X	X	X	X	X	X	X

**Bit 7 REQ (SREQ/ Status)**

**Bit 6 ACK (SACK/ Status)**

**Bit 5 BSY (SBSY/ Status)**

**Bit 4 SEL (SSEL/ Status)**

**Bit 3 ATN (SATN/ Status)**

**Bit 2 MSG (SMSG/ Status)**

**Bit 1 C/D (SC\_D/ Status)**

**Bit 0 I/O (SI\_O/ Status)**

When read, this register returns the SCSI control line status. A bit will be set when the corresponding SCSI control line is asserted. These bits are not latched; they are a true representation of what is on the SCSI bus at the time the register is read. The resulting read data is synchronized before being presented to the PCI bus to prevent parity errors from being passed to the system. This register can be used for diagnostics testing or operation in low level mode.

## Register 0C (8C) DMA Status (DSTAT) Read Only

DFE	MDPE	BF	ABRT	SSI	SIR	RES	IID
7	6	5	4	3	2	1	0
Default>>>							
1	0	0	0	0	0	X	0

Reading this register will clear any bits that are set at the time the register is read, but will not necessarily clear the register because additional interrupts may be pending (the SYM53C875 stacks interrupts). The DIP bit in the ISTAT register will also be cleared. DMA interrupt conditions may be individually masked through the DIEN register.

When performing consecutive 8-bit reads of the DSTAT, SIST0 and SIST1 registers (in any order), insert a delay equivalent to 12 CLK periods between the reads to ensure that the interrupts clear properly. See Chapter 2, “Functional Description,” for more information on interrupts.

### Bit 7 DFE (DMA FIFO Empty)

This status bit is set when the DMA FIFO is empty. It may be used to determine if any data resides in the FIFO when an error occurs and an interrupt is generated. This bit is a pure status bit and will not cause an interrupt.

### Bit 6 MDPE (Master Data Parity Error)

This bit is set when the SYM53C875 as a master detects a data parity error, or a target device signals a parity error during a data phase. This bit is completely disabled by the Master Parity Error Enable bit (bit 3 of CTEST4).

### Bit 5 BF (Bus Fault)

This bit is set when a PCI bus fault condition is detected. A PCI bus fault can only occur when the SYM53C875 is bus master, and is defined as a cycle that ends with a Bad Address or Target Abort Condition.

### Bit 4 ABRT (Aborted)

This bit is set when an abort condition occurs. An abort condition occurs when a software abort command is issued by setting bit 7 of the ISTAT register.

### Bit 3 SSI (Single Step Interrupt)

If the Single-Step Mode bit in the DCNTL register is set, this bit will be set and an interrupt generated after successful execution of each SCRIPTS instruction.

### Bit 2 SIR (SCRIPTS Interrupt Instruction Received)

This status bit is set whenever an Interrupt instruction is evaluated as true.

### Bit 1 EBPI (Extended Byte Parity Error Interrupt) (53C875N only)

This bit is set whenever the SYM53C875 detects a parity error on one of the four additional parity pins on the SYM53C875N.

### Bit 0 IID (Illegal Instruction Detected)

This status bit will be set any time an illegal or reserved instruction op code is detected, whether the SYM53C875 is operating in single-step mode or automatically executing SCSI SCRIPTS. Any of the following conditions during instruction execution will also cause this bit to be set:

1. The SYM53C875 is executing a Wait Disconnect instruction and the SCSI REQ line is asserted without a disconnect occurring.
2. A Block Move instruction is executed with 000000h loaded into the DBC register, indicating that zero bytes are to be moved.
3. During a Transfer Control instruction, the Compare Data (bit 18) and Compare Phase (bit 17) bits are set in the DBC register while the SYM53C875 is in target mode.



4. During a Transfer Control instruction, the Carry Test bit (bit 21) is set and either the Compare Data (bit 18) or Compare Phase (bit 17) bit is set.
5. A Transfer Control instruction is executed with the reserved bit 22 set.
6. A Transfer Control instruction is executed with the Wait for Valid phase bit (bit 16) set while the chip is in target mode.
7. A Load/Store instruction is issued with the memory address mapped to the operating registers of the chip, not including ROM or RAM.
8. A Load/Store instruction is issued when the register address is not aligned with the memory address
9. A Load/Store instruction is issued with bit 5 in the DCMD register clear or bits 3 or 2 set.
10. A Load/Store instruction when the count value in the DBC register is not set at 1 to 4.
11. A Load/Store instruction attempts to cross a dword boundary.
12. A Memory Move instruction is executed with one of the reserved bits in the DCMD register set.
13. A Memory Move instruction is executed with the source and destination addresses not byte-aligned.

**Register 0D (8D)**  
**SCSI Status Zero (SSTAT0)**  
**Read Only**

ILF	ORF	OLF	AIP	LOA	WOA	RST	SDP0/
7	6	5	4	3	2	1	0

Default>>>  
 0 0 0 0 0 0 0 0

**Bit 7 ILF (SIDL Least Significant Byte Full)**

This bit is set when the least significant byte in the SCSI Input Data Latch register (SIDL) contains data. Data is transferred from the SCSI bus to the SCSI Input Data Latch register before being sent to the DMA FIFO and then to the host bus. The SIDL register contains SCSI data received asynchronously. Synchronous data received does not flow through this register.

**Bit 6 ORF (SODR Least Significant Byte Full)**

This bit is set when the least significant byte in the SCSI Output Data Register (SODR, a hidden buffer register which is not accessible) contains data. The SODR register is used by the SCSI logic as a second storage register when sending data synchronously. It cannot be read or written by the user. This bit can be used to determine how many bytes reside in the chip when an error occurs.

**Bit 5 OLF (SODL Least Significant Byte Full)**

This bit is set when the least significant byte in the SCSI Output Data Latch (SODL) contains data. The SODL register is the interface between the DMA logic and the SCSI bus. In synchronous mode, data is transferred from the host bus to the SODL register, and then to the SCSI Output Data Register (SODR, a hidden buffer register which is not accessible) before being sent to the SCSI bus. In asynchronous mode, data is transferred from the host bus to the SODL register, and then to the SCSI bus. The SODR buffer register is not

used for asynchronous transfers. This bit can be used to determine how many bytes reside in the chip when an error occurs.

**Bit 4 AIP (Arbitration in Progress)**

Arbitration in Progress (AIP = 1) indicates that the SYM53C875 has detected a Bus Free condition, asserted BSY, and asserted its SCSI ID onto the SCSI bus.

**Bit 3 LOA (Lost Arbitration)**

When set, LOA indicates that the SYM53C875 has detected a bus free condition, arbitrated for the SCSI bus, and lost arbitration due to another SCSI device asserting the SEL/ signal.

**Bit 2 WOA (Won Arbitration)**

When set, WOA indicates that the SYM53C875 has detected a Bus Free condition, arbitrated for the SCSI bus and won arbitration. The arbitration mode selected in the SCNTL0 register must be full arbitration and selection for this bit to be set.

**Bit 1 RST/ (SCSI RST/ Signal)**

This bit reports the current status of the SCSI RST/ signal, and the RST signal (bit 6) in the ISTAT register. This bit is not latched and may be changing when read.

**Bit 0 SDP0/ (SCSI SDP0/ Parity Signal)**

This bit represents the active high current status of the SCSI SDP0/ parity signal. This signal is not latched and may be changing as it is read.

**Register 0E (8E)  
SCSI Status One (SSTAT1)  
Read Only**

FF3	FF2	FF1	FF0	SDPOL	MSG	C/D	I/O
7	6	5	4	3	2	1	0
Default>>>							
0	0	0	0	X	X	X	X

**Bits 7-4 FF3-FF0 (FIFO Flags)**

These four bits, along with SSTAT2 bit 4, define the number of bytes or words that currently reside in the SYM53C875's SCSI synchronous data FIFO. These bits are not latched and they will change as data moves through the FIFO.

FF4 (SSTAT2 bit 4)	FF3	FF2	FF1	FF0	Bytes or Words in the SCSI FIFO
0	0	0	0	0	0
0	0	0	0	1	1
0	0	0	1	0	2
0	0	0	1	1	3
0	0	1	0	0	4
0	0	1	0	1	5
0	0	1	1	0	6
0	0	1	1	1	7
0	1	0	0	0	8
0	1	0	0	1	9
0	1	0	1	0	10
0	1	0	1	1	11
0	1	1	0	0	12
0	1	1	0	1	13
0	1	1	1	0	14
0	1	1	1	1	15
1	0	0	0	0	16

**Bit 3 SDP0L (Latched SCSI Parity)**

This bit reflects the SCSI parity signal (SDP0/), corresponding to the data latched in the SCSI Input Data Latch register (SIDL). It changes when a new byte is latched into the least significant byte of the SIDL register. This bit is active high, in other words, it is set when the parity signal is active.

**Bit 2 MSG (SCSI MSG/ Signal)**

**Bit 1 C/D (SCSI C\_D/ Signal)**

**Bit 0 I/O (SCSI I\_O/ Signal)**

These SCSI phase status bits are latched on the asserting edge of SREQ/ when operating in either initiator or target mode. These bits are set when the corresponding signal is active. They are useful when operating in low level mode.

**Register 0F (8F)  
SCSI Status Two (SSTAT2)  
(Read Only)**

ILF1	ORF1	OLF1	FF4	SPL1	RES	LDSC	SDP1
7	6	5	4	3	2	1	0
Default>>>							
0	0	0	0	X	X	1	X

**Bit 7 ILF1 (SIDL Most Significant Byte Full)**

This bit is set when the most significant byte in the SCSI Input Data Latch register (SIDL) contains data. Data is transferred from the SCSI bus to the SCSI Input Data Latch register before being sent to the DMA FIFO and then to the host bus. The SIDL register contains SCSI data received asynchronously. Synchronous data received does not flow through this register.

**Bit 6 ORF1 (SODR Most Significant Byte Full)**

This bit is set when the most significant byte in the SCSI Output Data Register (SODR, a hidden buffer register which is not accessible) contains data. The SODR register is used by the SCSI logic as a second storage register when sending data synchronously. It is not accessible to the user. This bit can be used to determine how many bytes reside in the chip when an error occurs.

**Bit 5 OLF1 (SODL Most Significant Byte Full)**

This bit is set when the most significant byte in the SCSI Output Data Latch (SODL) contains data. The SODL register is the interface between the DMA logic and the SCSI bus. In synchronous mode, data is transferred from the host bus to the SODL register, and then to the SCSI Output Data Register (SODR, a hidden buffer register which is not accessible) before being sent to the SCSI bus. In asynchronous mode, data is transferred from the host bus to the SODL register, and then to the SCSI bus. The SODR buffer register is not

used for asynchronous transfers. This bit can be used to determine how many bytes reside in the chip when an error occurs.

**Bit 4 FF4 (FIFO Flags bit 4)**

This is the most significant bit in the SCSI FIFO Flags field, with the rest of the bits in SSTAT1. For a complete description of this field, see the definition for SSTAT1 bits 7-4.

**Bit 3 SPL1(Latched SCSI parity for SD15-8)**

This active high bit reflects the SCSI odd parity signal corresponding to the data latched into the most significant byte in the SIDL register.

**Bit 2 DIFFSENSE SENSE**

If this bit is reset, the correct cable type has been connected for the differential operation. If this bit is set, a single-ended cable has been connected to the device's DIFFSENSE pin.

**Bit 1 LDSC (Last Disconnect)**

Used in conjunction with the Connected (CON) bit in SCNTL1, this status bit allows the user to detect the case in which a target device disconnects, and then some SCSI device selects or reselects, the SYM53C875. If the Connected bit is asserted and the LDSC bit is asserted, a disconnect has occurred. This bit is set when the Connected bit in SCNTL1 is off. This bit is cleared when a Block Move instruction is executed while the Connected bit in SCNTL1 is on.

**Bit 0 SDP1 (SCSI SDP1 Signal)**

This bit represents the active-high current state of the SCSI SDP1 parity signal. It is unlatched and may be changing as it is read.

---

**Registers 10-13 (90-93)  
Data Structure Address (DSA)  
Read/Write**

This 32-bit register contains the base address used for all table indirect calculations. The DSA register is usually loaded prior to starting an I/O, but it is possible for a SCRIPTS Memory Move to load the DSA during the I/O.

During any Memory-to-Memory Move operation, the contents of this register are preserved. The power-up value of this register is indeterminate.

**Register 14 (94)**  
**Interrupt Status (ISTAT)**  
 (Read/Write)

ABRT	SRST	SIGP	SEM	CON	INTF	SIP	DIP
7	6	5	4	3	2	1	0
Default>>>							
0	0	0	0	0	0	0	0

This is the only register that can be accessed by the host CPU while the SYM53C875 is executing SCRIPTS (without interfering in the operation of the SYM53C875). It may be used to poll for interrupts if hardware interrupts are disabled. There may be stacked interrupts pending; read this register after servicing an interrupt to check for stacked interrupts. For more information on interrupt handling refer to Chapter 2, "Functional Description."

**Bit 7 ABRT (Abort Operation)**

Setting this bit aborts the current operation being executed by the SYM53C875. If this bit is set and an interrupt is received, reset this bit before reading the DSTAT register to prevent further aborted interrupts from being generated. The sequence to abort any operation is:

1. Set this bit.
2. Wait for an interrupt.
3. Read the ISTAT register.
4. If the SCSI Interrupt Pending bit is set, then read the SIST0 or SIST1 register to determine the cause of the SCSI Interrupt and go back to Step 2.
5. If the SCSI Interrupt Pending bit is clear, and the DMA Interrupt Pending bit is set, then write 00h value to this register.
6. Read the DSTAT register to verify the aborted interrupt and to see if any other interrupting conditions have occurred.

**Bit 6 SRST (Software Reset)**

Setting this bit resets the SYM53C875. All operating registers are cleared to their respective default values and all SCSI signals are deasserted. Setting this bit does not cause the SCSI RST/ signal to be asserted. This reset will not clear the 53C700 Compatibility bit or any of the PCI configuration registers. This bit is not self-clearing; it must be cleared to clear the reset condition (a hardware reset will also clear this bit).

**Bit 5 SIGP (Signal Process)**

SIGP is a R/W bit that can be written at any time, and polled and reset via CTEST2. The SIGP bit can be used in various ways to pass a flag to or from a running SCRIPTS instruction.

The only SCRIPTS instruction directly affected by the SIGP bit is Wait For Selection/ Reselection. Setting this bit causes that instruction to jump to the alternate address immediately. The instructions at the alternate jump address should check the status of SIGP to determine the cause of the jump. The SIGP bit may be used at any time and is not restricted to the wait for selection/ reselection condition.

**Bit 4 SEM (Semaphore)**

This bit can be set by the SCRIPTS processor using a SCRIPTS register write instruction. The bit may also be set by an external processor while the SYM53C875 is executing a SCRIPTS operation. This bit enables the SYM53C875 to notify an external processor of a predefined condition while SCRIPTS are running. The external processor may also notify the SYM53C875 of a predefined condition and the SCRIPTS processor may take action while SCRIPTS are executing.

**Bit 3 CON (Connected)**

This bit is automatically set any time the SYM53C875 is connected to the SCSI bus as an initiator or as a target. It will be set after successfully completing selection or when the SYM53C875 has responded to a bus-initiated selection or reselection. It will also be set after the SYM53C875 wins arbitration when operating in low level mode. When this bit is clear, the SYM53C875 is not connected to the SCSI bus.

**Bit 2 INTF (Interrupt on the Fly)**

This bit is asserted by an INTFLY instruction during SCRIPTS execution. SCRIPTS programs will not halt when the interrupt occurs. This bit can be used to notify a service routine, running on the main processor while the SCRIPTS processor is still executing a SCRIPTS program. If this bit is set, when the ISTAT register is read it will not automatically be cleared. To clear this bit, it must be written to a one. The reset operation is self-clearing.

**Note:** if the INTF bit is set but SIP or DIP is not set, do not attempt to read the other chip status registers. An interrupt-on-the-fly interrupt must be cleared before servicing any other interrupts indicated by SIP or DIP.

**Note:** this bit must be written to one in order to clear it after it has been set.

**Bit 1 SIP (SCSI Interrupt Pending)**

This status bit is set when an interrupt condition is detected in the SCSI portion of the SYM53C875. The following conditions will cause a SCSI interrupt to occur:

- A phase mismatch (initiator mode) or SATN/ becomes active (target mode)
- An arbitration sequence completes
- A selection or reselection time-out occurs
- The SYM53C875 was selected
- The SYM53C875 was reselected
- A SCSI gross error occurs
- An unexpected disconnect occurs
- A SCSI reset occurs
- A parity error is detected
- The handshake-to-handshake timer is expired
- The general purpose timer is expired.

To determine exactly which condition(s) caused the interrupt, read the SIST0 and SIST1 registers.

**Bit 0 DIP (DMA Interrupt Pending)**

This status bit is set when an interrupt condition is detected in the DMA portion of the SYM53C875. The following conditions will cause a DMA interrupt to occur:

- A PCI parity error is detected
- A bus fault is detected
- An abort condition is detected
- A SCRIPTS instruction is executed in single-step mode
- A SCRIPTS interrupt instruction is executed
- An illegal instruction is detected.

To determine exactly which condition(s) caused the interrupt, read the DSTAT register.

---

**Register 18 (98)**  
**Chip Test Zero (CTEST0)**  
 Read/Write

This was a general purpose read/write register in previous SYM53C8XX family chips. Although it is still a read/write register, Symbios reserves the right to use these bits for future 53C8XX family enhancements.

---

**Register 19 (99)**  
**Chip Test One (CTEST1)**  
 Read Only

FMT3	FMT2	FMT1	FMT0	FFL3	FFL2	FFL1	FFL0
7	6	5	4	3	2	1	0
Default>>>							
1	1	1	1	0	0	0	0

**Bits 7-4 FMT3-0 (Byte Empty in DMA FIFO)**

These bits identify the bottom bytes in the DMA FIFO that are empty. Each bit corresponds to a byte lane in the DMA FIFO. For example, if byte lane three is empty, then FMT3 will be set. Since the FMT flags indicate the status of bytes at the bottom of the FIFO, if all FMT bits are set, the DMA FIFO is empty.

**Bits 3-0 FFL3-0 (Byte Full in DMA FIFO)**

These status bits identify the top bytes in the DMA FIFO that are full. Each bit corresponds to a byte lane in the DMA FIFO. For example, if byte lane three is full then FFL3 will be set. Since the FFL flags indicate the status of bytes at the top of the FIFO, if all FFL bits are set, the DMA FIFO is full.

**Register 1A (9A)**  
**Chip Test Two (CTEST2)**  
**Read/Write**

DDIR	SIGP	CIO	CM	SRTCH	TEOP	DREQ	DACK
7	6	5	4	3	2	1	0
Default>>>							
0	0	X	X	0	0	0	1

**Bit 7 DDIR (Data Transfer Direction)**

This status bit indicates which direction data is being transferred. When this bit is set, the data will be transferred from the SCSI bus to the host bus. When this bit is clear, the data will be transferred from the host bus to the SCSI bus.

**Bit 6 SIGP (Signal Process)**

This bit is a copy of the SIGP bit in the ISTAT register (bit 5). The SIGP bit is used to signal a running SCRIPTS instruction. When this register is read, the SIGP bit in the ISTAT register is cleared.

**Bit 5 CIO (Configured as I/O)**

This bit is defined as the Configuration I/O Enable Status bit. This read-only bit indicates if the chip is currently enabled as I/O space.

Note: both bits 4 and 5 may be set if the chip is dual-mapped.

**Bit 4 CM (Configured as Memory)**

This bit is defined as the configuration memory enable status bit. This read-only bit indicates if the chip is currently enabled as memory space.

Note: both bits 4 and 5 may be set if the chip is dual-mapped.

**Bit 3 SRTCH (SCRATCHA/B Operation)**

This bit controls the operation of the SCRATCHA and SCRATCHB registers. When it is set, SCRATCHB contains the RAM base address value from the PCI configuration RAM Base Address register. This is the base address for the 4 KB internal RAM. In addition, the SCRATCHA register displays the memory-mapped based address of the chip

operating registers. When this bit is clear, the SCRATCHA and SCRATCHB registers return to normal operation.

Note: bit 3 is the only writable bit in this register. All other bits are read only. When modifying this register, all other bits must be written to zero. Do not execute a Read-Modify-Write to this register.

**Bit 2 TEOP (SCSI True End of Process)**

This bit indicates the status of the SYM53C875's internal TEOP signal. The TEOP signal acknowledges the completion of a transfer through the SCSI portion of the SYM53C875. When this bit is set, TEOP is active. When this bit is clear, TEOP is inactive.

**Bit 1 DREQ (Data Request Status)**

This bit indicates the status of the SYM53C875's internal Data Request signal (DREQ). When this bit is set, DREQ is active. When this bit is clear, DREQ is inactive.

**Bit 0 DACK (Data Acknowledge Status)**

This bit indicates the status of the SYM53C875's internal Data Acknowledge signal (DACK/). When this bit is set, DACK/ is inactive. When this bit is clear, DACK/ is active.



**Register 1B (9B)**  
**Chip Test Three (CTEST3)**  
 Read/Write

V3	V2	V1	V0	FLF	CLF	FM	WRIE
7	6	5	4	3	2	1	0
Default>>>							
x	x	x	x	0	0	0	0

**Bits 7-4 V3-V0 (Chip revision level)**

These bits identify the chip revision level for software purposes. The value should be the same as the lower nibble of the PCI Revision ID Register, at address 08h in configuration space.

**Bit 3 FLF (Flush DMA FIFO)**

When this bit is set, data residing in the DMA FIFO is transferred to memory, starting at the address in the DNAD register. The internal DMAWR signal, controlled by the CTEST5 register, determines the direction of the transfer. This bit is not self clearing; once the SYM53C875 has successfully transferred the data, this bit should be reset.

Note: polling of FIFO flags is allowed during flush operations.

**Bit 2 CLF (Clear DMA FIFO)**

When this bit is set, all data pointers for the DMA FIFO are cleared. Any data in the FIFO is lost. This bit automatically resets after the SYM53C875 has successfully cleared the appropriate FIFO pointers and registers.

Note: this bit does not clear the data visible at the bottom of the FIFO.

**Bit 1 FM (Fetch Pin Mode)**

When set, this bit causes the FETCH/ pin to deassert during indirect and table indirect read operations. FETCH/ will only be active during the op code portion of an instruction fetch. This allows SCRIPTS to be stored in a PROM while data tables are stored in RAM.

If this bit is not set, FETCH/ will be asserted for all bus cycles during instruction fetches.

**Bit 0 WRIE (Write and Invalidate Enable)**

This bit, when set, causes Memory Write and Invalidate commands to be issued on the PCI bus after certain conditions have been met. These conditions are described in detail in Chapter 3.

**Registers 1C-1F (9C-9F)  
Temporary (TEMP)  
Read/Write**

This 32-bit register stores the Return instruction address pointer from the Call instruction. The address pointer stored in this register is loaded into the DSP register when a Return instruction is executed. This address points to the next instruction to be executed. Do not write to this register while the SYM53C875 is executing SCRIPTS.

During any Memory-to-Memory Move operation, the contents of this register are preserved. The power-up value of this register is indeterminate.

**Register 20 (A0)  
DMA FIFO (DFIFO)  
Read/Write**

B07	B06	B05	B04	B03	B02	B01	B00
7	6	5	4	3	2	1	0
Default>>>							
X	0	0	0	0	0	0	0

**Bits 7-0 BO7-BO0 (Byte offset counter)**

These bits, along with bits 1-0 in the CTEST5 register, indicate the amount of data transferred between the SCSI core and the DMA core. It may be used to determine the number of bytes in the DMA FIFO when an interrupt occurs. These bits are unstable while data is being transferred between the two cores; once the chip has stopped transferring data, these bits are stable.

Since the DFIFO register counts the number of bytes transferred between the DMA core and the SCSI core, and the DBC register counts the number of bytes transferred across the host bus, the difference between these two counters represents the number of bytes remaining in the DMA FIFO.

The following steps will determine how many bytes are left in the DMA FIFO when an error occurs, regardless of the direction of the transfer:

1. If the DMA FIFO size is set to 88 bytes, subtract the seven least significant bits of the DBC register from the 7-bit value of the DFIFO register.  
If the DMA FIFO size is set to 536 bytes (using bit 5 of the CTEST5 register), subtract the 10 least significant bits of the DBC register from the 10-bit value of the DMA FIFO Byte Offset Counter, which consists of bits 1-0 in the CTEST5 register and bits 7-0 of the DMA FIFO register.
2. If the DMA FIFO size is set to 88 bytes, AND the result with 7Fh for a byte count

between zero and 64. If the DMA FIFO size is set to 536 bytes, AND the result with 3FFh for a byte count between 0 and 536.

Note: to calculate the total number of bytes in both the DMA FIFO and SCSI logic, see the section on Data Paths in Chapter Two, “Functional Description.”

### Register 21 (A1) Chip Test Four (CTEST4) Read/Write

BDIS	ZMOD	ZSD	SRTM	MPEE	FBL2	FBL1	FBL0
7	6	5	4	3	2	1	0
Default>>>							
0	0	0	0	0	0	0	0

#### Bit 7 **BDIS (Burst Disable)**

When set, this bit will cause the SYM53C875 to perform back to back cycles for all transfers. When reset, the SYM53C875 will perform back to back transfers for op code fetches and burst transfers for data moves.

#### Bit 6 **ZMOD (High Impedance Mode)**

Setting this bit causes the SYM53C875 to place all output and bidirectional pins into a high-impedance state. In order to read data out of the SYM53C875, this bit must be cleared. This bit is intended for board-level testing only. Do not set this bit during normal system operation.

#### Bit 5 **ZSD (SCSI Data High Impedance)**

Setting this bit causes the SYM53C875 to place the SCSI data bus SD(15-0) and the parity lines SDP(1-0) in a high-impedance state. In order to transfer data on the SCSI bus, this bit must be cleared.

#### Bit 4 **SRTM (Shadow Register Test Mode)**

Setting this bit allows access to the shadow registers used by memory-to-memory Move operations. When this bit is set, register accesses to the TEMP and DSA registers are directed to the shadow copies STEMP (Shadow TEMP) and SDSA (Shadow DSA). The registers are shadowed to prevent them from being overwritten during a Memory-to-Memory Move operation. The DSA and TEMP registers contain the base address used for table indirect calculations, and the address pointer for a call or return instruction, respectively. This bit is intended for manufacturing diagnostics only and should not be set during normal operations.

**Bit 3 MPEE (Master Parity Error Enable)**

Setting this bit enables parity checking during master data phases. A parity error during a bus master read is detected by the SYM53C875. A parity error during a bus master write is detected by the target, and the SYM53C875 is informed of the error by the PERR/ pin being asserted by the target. When this bit is reset, the SYM53C875 will not interrupt if a master parity error occurs. This bit is reset at power up.

**Bits 2-0 FBL2-FBL0 (FIFO Byte Control)**

FBL2	FBL1	FBL0	DMA FIFO Byte lane	Pins
0	X	X	Disabled	n/a
1	0	0	0	D(7-0)
1	0	1	1	D(15-8)
1	1	0	2	D(23-16)
1	1	1	3	D(31-24)

These bits steer the contents of the CTEST6 register to the appropriate byte lane of the 32-bit DMA FIFO. If the FBL2 bit is set, then FBL1 and FBL0 determine which of four byte lanes can be read or written. When cleared, the byte lane read or written is determined by the current contents of the DNAD and DBC registers. Each of the four bytes that make up the 32-bit DMA FIFO can be accessed by writing these bits to the proper value. For normal operation, FBL2 must equal zero.

**Register 22 (A2)  
Chip Test Five (CTEST5)  
Read/Write**

ADCK	BBCK	DFS	MASR	DDIR	BL2	B09	B08
7	6	5	4	3	2	1	0
Default>>>							
0	0	0	0	0	X	X	X

**Bit 7 ADCK (Clock Address Incrementor)**

Setting this bit increments the address pointer contained in the DNAD register. The DNAD register is incremented based on the DNAD contents and the current DBC value. This bit automatically clears itself after incrementing the DNAD register.

**Bit 6 BBCK (Clock Byte Counter)**

Setting this bit decrements the byte count contained in the 24-bit DBC register. It is decremented based on the DBC contents and the current DNAD value. This bit automatically clears itself after decrementing the DBC register.

**Bit 5 DFS (DMA FIFO Size)**

This bit controls the size of the DMA FIFO. When clear, the DMA FIFO will appear to be only 88 bytes deep. When set, the DMA FIFO size will increase to 536 bytes. Using an 88-byte FIFO allows software written for other SYM53C8XX family chips to properly calculate the number of bytes residing in the chip after a target disconnect. The default value of this bit is zero.

**Bit 4 MASR (Master Control for Set or Reset Pulses)**

This bit controls the operation of bit 3. When this bit is set, bit 3 asserts the corresponding signals. When this bit is reset, bit 3 deasserts the corresponding signals. This bit and bit 3 should not be changed in the same write cycle.

**Bit 3 DDIR (DMA Direction)**

Setting this bit either asserts or deasserts the internal DMA Write (DMAWR) direction signal depending on the current status of the MASR bit in this register. Asserting the DMAWR signal indicates that data will be transferred from the SCSI bus to the host bus. Deasserting the DMAWR signal transfers data from the host bus to the SCSI bus.

**Bit 2 BL2 (Burst Length bit 2)**

This bit works with bits 6 and 7 in the DMODE register to determine the burst length. For complete definitions of this field, refer to the descriptions of DMODE bits 6 and 7. This bit is disabled if an 88-byte FIFO is selected by clearing the DMA FIFO Size bit.

**Bits 1-0 BO9-8**

These are the upper two bits of the DMA FIFO byte offset counter. The entire field is described under the DFIFO register, bits 7-0.

---

**Register 23 (A3)**  
**Chip Test Six (CTEST6)**  
 Read/Write

DF7	DF6	DF5	DF4	DF3	DF2	DF1	DF0
7	6	5	4	3	2	1	0
Default>>>							
0	0	0	0	0	0	0	0

**Bits 7-0 DF7-DF0 (DMA FIFO)**

Writing to this register writes data to the appropriate byte lane of the DMA FIFO as determined by the FBL bits in the CTEST4 register. Reading this register unloads data from the appropriate byte lane of the DMA FIFO as determined by the FBL bits in the CTEST4 register. Data written to the FIFO is loaded into the top of the FIFO. Data read out of the FIFO is taken from the bottom. To prevent DMA data from being corrupted, this register should not be accessed before starting or restarting SCRIPTS operation. This register should only be written when testing the DMA FIFO using the CTEST4 register. Writes to this register while the test mode is not enabled will have unexpected results.

---

**Registers 24-26 (A4-A6)**  
**DMA Byte Counter (DBC)**  
**Read/Write**

This 24-bit register determines the number of bytes to be transferred in a Block Move instruction. While sending data to the SCSI bus, the counter is decremented as data is moved into the DMA FIFO from memory. While receiving data from the SCSI bus, the counter is decremented as data is written to memory from the SYM53C875. The DBC counter is decremented each time that data is transferred on the PCI bus. It is decremented by an amount equal to the number of bytes that were transferred.

The maximum number of bytes that can be transferred in any one Block Move command is 16,777,215 bytes. The maximum value that can be loaded into the DBC register is FFFFFFFh. If the instruction is a Block Move and a value of 000000h is loaded into the DBC register, an illegal instruction interrupt will occur if the SYM53C875 is not in target mode, Command phase.

The DBC register is also used to hold the least significant 24 bits of the first dword of a SCRIPT fetch, and to hold the offset value during table indirect I/O SCRIPTS. For a complete description, see Chapter Six, "Instruction Set of the I/O Processor." The power-up value of this register is indeterminate.

---

**Register 27 (A7)**  
**DMA Command (DCMD)**  
**Read/Write**

This 8-bit register determines the instruction for the SYM53C875 to execute. This register has a different format for each instruction. For complete descriptions, see Chapter Six, "Instruction Set of the I/O Processor."

---

**Registers 28-2B (A8-AB)**  
**DMA Next Address (DNAD)**  
Read/Write

This 32-bit register contains the general purpose address pointer. At the start of some SCRIPTS operations, its value is copied from the DSPS register. Its value may not be valid except in certain abort conditions. The default value of this register is zero.

This register should not be used to determine data addresses during a Phase Mismatch interrupt, as its value is not always correct for this use. The DBC, DFIFO, and DSPS registers should be used to calculate residual byte counts and addresses as described in the Data Paths section of Chapter 2.

---

**Registers 2C-2F (AC-AF)**  
**DMA SCRIPTS Pointer (DSP)**  
Read/Write

To execute SCSI SCRIPTS, the address of the first SCRIPTS instruction must be written to this register. In normal SCRIPTS operation, once the starting address of the SCRIPT is written to this register, SCRIPTS are automatically fetched and executed until an interrupt condition occurs.

In single-step mode, there is a single step interrupt after each instruction is executed. The DSP register does not need to be written with the next address, but the Start DMA bit (bit 2, DCNTL register) must be set each time the step interrupt occurs to fetch and execute the next SCRIPTS command. When writing this register eight bits at a time, writing the upper eight bits begins execution of SCSI SCRIPTS. The default value of this register is zero.

---

**Registers 30-33 (B0-B3)**

**DMA SCRIPTS Pointer Save (DSPS)**

**Read/Write**

This register contains the second dword of a SCRIPTS instruction. It is overwritten each time a SCRIPTS instruction is fetched. When a SCRIPTS interrupt instruction is executed, this register holds the interrupt vector. The power-up value of this register is indeterminate.

---

**Registers 34-37 (B4-B7)**

**Scratch Register A (SCRATCH A)**

**Read/Write**

This is a general purpose, user-definable scratch pad register. Apart from CPU access, only Register Read/Write and Memory Moves into the SCRATCH register will alter its contents. The SYM53C875 cannot fetch SCRIPTS instructions from this location. When bit 3 in the CTEST2 register is set, this register contains the memory-mapped base address of the operating registers. Setting CTEST2 bit 3 only causes the base address to appear in this register; any information that was previously in the register will remain intact. Any writes to this register while CTEST2 bit 3 is set will pass through to the actual SCRATCHA register. The power-up value of this register is indeterminate.



**Register 38 (B8)**  
**DMA Mode (DMODE)**  
 Read/Write

BL1	BL0	SIOM	DIOM	ER	ERMP	BOF	MAN
7	6	5	4	3	2	1	0
Default>>>							
0	0	0	0	0	0	0	0

**Bit 7-6 BL1-BL0 (Burst Length)**

These bits control the maximum number of transfers performed per bus ownership, regardless of whether the transfers are back-to-back, burst, or a combination of both. The SYM53C875 asserts the Bus Request (REQ/) output when the DMA FIFO can accommodate a transfer of at least one burst size of data. Bus Request (REQ/) is also asserted during start-of-transfer and end-of-transfer cleanup and alignment, even though less than a full burst of transfers may be performed. The SYM53C875 inserts a “fairness delay” of four CLKs between burst-length transfers (as set in BL1-0) during normal operation. The fairness delay is not inserted during PCI retry cycles. This gives the CPU and other bus master devices the opportunity to access the PCI bus between bursts.

BL2 (CTEST5 bit 2)	BL1	BL0	Burst Length
0	0	0	2- transfer burst
0	0	1	4- transfer burst
0	1	0	8-transfer burst
0	1	1	16-transfer burst
1	0	0	32-transfer burst*
1	0	1	64-transfer burst*
1	1	0	128-transfer burst*
1	1	1	Reserved

\* Only valid if the FIFO size is set to 536 bytes

**Bit 5 SIOM (Source I/O-Memory Enable)**

This bit is defined as an I/O Memory Enable bit for the source address of a Memory Move or Block Move Command. If this bit is set, then the source address is in I/O space; and if reset, then the source address is in memory space.

This function is useful for register-to-memory operations using the Memory Move instruction when the SYM53C875 is I/O mapped. Bits 4 and 5 of the CTEST2 register can be used to determine the configuration status of the SYM53C875.

**Bit 4 DIOM (Destination I/O-Memory Enable)**

This bit is defined as an I/O Memory Enable bit for the destination address of a Memory Move or Block Move Command. If this bit is set, then the destination address is in I/O space; and if reset, then the destination address is in memory space.

This function is useful for memory-to-register operations using the Memory Move instruction when the SYM53C875 is I/O mapped. Bits 4 and 5 of the CTEST2 register can be used to determine the configuration status of the SYM53C875.

**Bit 3 ERL (Enable Read Line)**

This bit enables a PCI Read Line command. If PCI cache mode is enabled by setting bits in the PCI Cache Line Size register, this chip issues a Read Line command on all read cycles if other conditions are met. For more information on these conditions, refer to Chapter 3.

**Bit 2 ERMP (Enable Read Multiple)**

This bit, when set, will cause Read Multiple commands to be issued on the PCI bus after certain conditions have been met. These conditions are described in Chapter 3.

**Bit 1 BOF (Burst Op Code Fetch Enable)**

Setting this bit causes the SYM53C875 to fetch instructions in burst mode. Specifically, the chip will burst in the first two dwords of all instructions using a single bus ownership. If the instruction is a memory-to-memory move type, the third dword will be accessed in a subsequent bus ownership. If the instruction is an indirect type, the additional dword will be accessed in a subsequent bus ownership. If the instruction is a table indirect block move type, the chip will access the remaining two dwords in a subsequent bus ownership, thereby fetching the four dwords required in two bursts of two dwords each. This bit has no effect if SCRIPTS instruction prefetching is enabled.

**Bit 0 MAN (Manual Start Mode)**

Setting this bit prevents the SYM53C875 from automatically fetching and executing SCSI SCRIPTS when the DSP register is written. When this bit is set, the Start DMA bit in the DCNTL register must be set to begin SCRIPTS execution. Clearing this bit causes the SYM53C875 to automatically begin fetching and executing SCSI SCRIPTS when the DSP register is written. This bit normally is not used for SCSI SCRIPTS operations.

**Register 39 (B9)  
DMA Interrupt Enable (DIEN)  
Read/Write**

RES	MDPE	BF	ABRT	SSI	SIR	RES	IID
7	6	5	4	3	2	1	0
Default>>>							
X	0	0	0	0	0	X	0

This register contains the interrupt mask bits corresponding to the interrupting conditions described in the DSTAT register. An interrupt is masked by clearing the appropriate mask bit. Masking an interrupt prevents IRQ/ from being asserted for the corresponding interrupt, but the status bit will still be set in the DSTAT register. Masking an interrupt will not prevent the ISTAT DIP from being set. All DMA interrupts are considered fatal, therefore SCRIPTS will stop running when this condition occurs, whether or not the interrupt is masked. Setting a mask bit enables the assertion of IRQ/ for the corresponding interrupt. (A masked non-fatal interrupt will not prevent un-masked or fatal interrupts from getting through; interrupt stacking begins when either the ISTAT SIP or DIP bit is set.)

The SYM53C875 IRQ/ output is latched; once asserted, it will remain asserted until the interrupt is cleared by reading the appropriate status register. Masking an interrupt after the IRQ/ output is asserted will not cause IRQ/ to be deasserted.

For more information on interrupts, see Chapter Two, "Functional Description."

<b>Bit 7</b>	<b>Reserved</b>	<b>Register 3A (BA)</b>
<b>Bit 6</b>	<b>MDPE (Master Data Parity Error)</b>	<b>Scratch Byte Register (SBR)</b>
<b>Bit 5</b>	<b>BF (Bus Fault)</b>	<b>Read/Write</b>
<b>Bit 4</b>	<b>ABRT (Aborted)</b>	This is a general purpose register. Apart from CPU access, only Register Read/Write and Memory Moves into this register will alter its contents. The default value of this register is zero. This register was called the DMA Watchdog Timer on previous SYM53C8XX family products.
<b>Bit 3</b>	<b>SSI (Single -step Interrupt)</b>	
<b>Bit 2</b>	<b>SIR (SCRIPTS Interrupt Instruction Received)</b>	
<b>Bit 1</b>	<b>EBPE (Extended Byte Parity Enable) (SYM53C875N only)</b>	
<b>Bit 0</b>	<b>IID (Illegal Instruction Detected)</b>	

**Register 3B (BB)**  
**DMA Control (DCNTL)**  
 Read/Write

CLSE	PFF	PFEN	SSM	IRQM	STD	IRQD	COM
7	6	5	4	3	2	1	0
Default>>>							
0	0	0	0	0	0	0	0

**Bit 7 CLSE (Cache Line Size Enable)**

Setting this bit enables the SYM53C875 to sense and react to cache line boundaries set up by the DMODE or PCI Cache Line Size register, whichever contains the smaller value. Clearing this bit disables the cache line size logic and the SYM53C875 monitors the cache line size via the DMODE register.

**Bit 6 PFF (Pre-fetch Flush)**

Setting this bit will cause the pre-fetch unit to flush its contents. The bit will reset after the flush is complete.

**Bit 5 PFEN (Pre-fetch Enable)**

Setting this bit enables the pre-fetch unit if the burst size is equal to or greater than four. For more information on SCRIPTS instruction prefetching, see Chapter 2.

**Bit 4 SSM (Single-step Mode)**

Setting this bit causes the SYM53C875 to stop after executing each SCRIPTS instruction, and generate a single step interrupt. When this bit is clear the SYM53C875 will not stop after each instruction; instead it continues fetching and executing instructions until an interrupt condition occurs. For normal SCSI SCRIPTS operation, this bit should be clear. To restart the SYM53C875 after it generates a SCRIPTS Step interrupt, the ISTAT and DSTAT registers should be read to recognize and clear the interrupt and then the START DMA bit in this register should be set.

**Bit 3 IRQM (IRQ Mode)**

When set, this bit will enable a totem pole driver for the IRQ pin. When reset, this bit will enable an open drain driver for the IRQ pin

with a internal weak pull-up. This bit is reset at power up. The bit should remain clear to retain full PCI compliance.

**Bit 2 STD (Start DMA Operation)**

The SYM53C875 fetches a SCSI SCRIPTS instruction from the address contained in the DSP register when this bit is set. This bit is required if the SYM53C875 is in one of the following modes:

1. Manual start mode – Bit 0 in the DMODE register is set
2. Single-step mode – Bit 4 in the DCNTL register is set

When the SYM53C875 is executing SCRIPTS in manual start mode, the Start DMA bit needs to be set to start instruction fetches, but does not need to be set again until an interrupt occurs. When the SYM53C875 is in single-step mode, the Start DMA bit needs to be set to restart execution of SCRIPTS after a single-step interrupt.

**Bit 1 IRQD (IRQ Disable)**

Setting this bit disables the IRQ pin; clearing the bit enables normal operation. As with any other register other than ISTAT, this register can not be accessed except by a SCRIPTS instruction during SCRIPTS execution. For more information on the use of this bit in interrupt handling, see Chapter 2.

**Bit 0 COM (53C700 Compatibility)**

When this bit is clear, the SYM53C875 will behave in a manner compatible with the SYM53C700; selection/reselection IDs will be stored in both the SSID and SFBR registers.

When this bit is set, the ID will be stored only in the SSID register, protecting the SFBR from being overwritten if a selection/reselection occurs during a DMA register-to-register operation. This bit is not affected by a software reset.

**Register 3C-3F (BC-BF)**  
**Adder Sum Output (ADDER)**  
 Read Only

This register contains the output of the internal adder, and is used primarily for test purposes. The power-up value for this register is indeterminate.

**Register 40 (C0)**  
**SCSI Interrupt Enable Zero (SIEN0)**  
 Read/Write

M/A	CMP	SEL	RSL	SGE	UDC	RST	PAR
7	6	5	4	3	2	1	0
Default>>>							
0	0	0	0	0	0	0	0

This register contains the interrupt mask bits corresponding to the interrupting conditions described in the SIST0 register. An interrupt is masked by clearing the appropriate mask bit. For more information on interrupts, see Chapter 2.

**Bit 7 M/A (SCSI Phase Mismatch - Initiator Mode; SCSI ATN Condition - Target Mode)**

In initiator mode, this bit is set when the SCSI phase asserted by the target and sampled during SREQ/ does not match the expected phase in the SOCL register. This expected phase is automatically written by SCSI SCRIPTS. In target mode, this bit is set when the initiator has asserted SATN/. See the Disable Halt on Parity Error or SATN/ Condition bit in the SCNTL1 register for more information on when this status is actually raised.

**Bit 6 CMP (Function Complete)**

Full arbitration and selection sequence has completed.

**Bit 5 SEL (Selected)**

The SYM53C875 has been selected by a SCSI target device. The Enable Response to Selection bit in the SCID register must be set for this to occur.

**Bit 4 RSL (Reselected)**

The SYM53C875 has been reselected by a SCSI initiator device. The Enable Response to Reselection bit in the SCID register must be set for this to occur.

**Bit 3 SGE (SCSI Gross Error)**

The following conditions are considered SCSI Gross Errors:

1. Data underflow - the SCSI FIFO was read when no data was present.
2. Data overflow - the SCSI FIFO was written to while full.
3. Offset underflow - in target mode, a SACK/ pulse was received before the corresponding SREQ/ was sent.
4. Offset overflow - in initiator mode, an SREQ/ pulse was received which caused the maximum offset (Defined by the MO3-0 bits in the SXFER register) to be exceeded.
5. In initiator mode, a phase change occurred with an outstanding SREQ/SACK offset.
6. Residual data in SCSI FIFO - a transfer other than synchronous data receive was started with data left in the SCSI synchronous receive FIFO.

**Bit 2 UDC (Unexpected Disconnect)**

This condition only occurs in initiator mode. It happens when the target to which the SYM53C875 is connected disconnects from the SCSI bus unexpectedly. See the SCSI Disconnect Unexpected bit in the SCNTL2 register for more information on expected versus unexpected disconnects. Any disconnect in low level mode causes this condition.

**Bit 1 RST (SCSI Reset Condition)**

The SRST/ signal has been asserted by the SYM53C875 or any other SCSI device. This condition is edge-triggered, so multiple interrupts cannot occur because of a single SRST/ pulse.

**Bit 0 PAR (SCSI Parity Error)**

The SYM53C875 detected a parity error while receiving or sending SCSI data. See the Disable Halt on Parity Error or SATN/ Condition bits in the SCNTL1 register for more information on when this condition will actually be raised.

**Register 41 (C1)**

**SCSI Interrupt Enable One (SIEN1)  
Read/Write**

RES	RES	RES	RES	RES	STO	GEN	HTH
7	6	5	4	3	2	1	0
Default>>>							
X	X	X	X	X	0	0	0

This register contains the interrupt mask bits corresponding to the interrupting conditions described in the SIST1 register. An interrupt is masked by clearing the appropriate mask bit. For more information on interrupts, refer to Chapter 2, "Functional Description."

**Bits 7-3 Reserved**

**Bit 2 STO (Selection or Reselection Time-out)**

The SCSI device which the SYM53C875 was attempting to select or reselect did not respond within the programmed time-out period. See the description of the STIME0 register bits 3-0 for more information on the time-out timer.

**Bit 1 GEN (General Purpose Timer Expired)**

The general purpose timer has expired. The time measured is the time between enabling and disabling of the timer. See the description of the STIME1 register, bits 3-0, for more information on the general purpose timer.

**Bit 0 HTH ( Handshake-to-Handshake Timer Expired)**

The handshake-to-handshake timer has expired. The time measured is the SCSI Request-to-Request (target) or Acknowledge-to-Acknowledge (initiator) period. See the description of the STIME0 register, bits 7-4, for more information on the handshake-to-handshake timer.

**Register 42 (C2)**  
**SCSI Interrupt Status Zero (SIST0)**  
**Read Only**

M/A	CMP	SEL	RSL	SGE	UDC	RST	PAR
7	6	5	4	3	2	1	0
Default>>>							
0	0	0	0	0	0	0	0

Reading the SIST0 register returns the status of the various interrupt conditions, whether they are enabled in the SIEN0 register or not. Each bit set indicates that the corresponding condition has occurred. Reading the SIST0 will clear the interrupt status.

Reading this register will clear any bits that are set at the time the register is read, but will not necessarily clear the register because additional interrupts may be pending (the SYM53C875 stacks interrupts). SCSI interrupt conditions may be individually masked through the SIEN0 register.

When performing consecutive 8-bit reads of the DSTAT, SIST0, and SIST1 registers (in any order), insert a delay equivalent to 12 CLK periods between the reads to ensure the interrupts clear properly. Also, if reading the registers when both the ISTAT SIP and DIP bits may not be set, the SIST0 and SIST1 registers should be read before the DSTAT register to avoid missing a SCSI interrupt. For more information on interrupts, refer to Chapter 2, "Functional Description."

**Bit 7 M/A (Initiator Mode: Phase Mismatch; Target Mode: SATN/ Active)**

In initiator mode, this bit is set if the SCSI phase asserted by the target does not match the instruction. The phase is sampled when SREQ/ is asserted by the target. In target mode, this bit is set when the SATN/ signal is asserted by the initiator.

**Bit 6 CMP (Function Complete)**

This bit is set when an arbitration only or full arbitration sequence has completed.

**Bit 5 SEL (Selected)**

This bit is set when the SYM53C875 is selected by another SCSI device. The Enable Response to Selection bit must have been set in the SCID register (and the RESPID register must hold the chip's ID) for the SYM53C875 to respond to selection attempts.

**Bit 4 RSL (Reselected)**

This bit is set when the SYM53C875 is reselected by another SCSI device. The Enable Response to Reselection bit must have been set in the SCID register (and the RESPID register must hold the chip's ID) for the SYM53C875 to respond to reselection attempts.

**Bit 3 SGE (SCSI Gross Error)**

This bit is set when the SYM53C875 encounters a SCSI Gross Error Condition. The following conditions can result in a SCSI Gross Error Condition:

1. Data Underflow - the SCSI FIFO register was read when no data was present.
2. Data Overflow - too many bytes were written to the SCSI FIFO or the synchronous offset caused the SCSI FIFO to be overwritten.
3. Offset Underflow - the SYM53C875 is operating in target mode and a SACK/ pulse is received when the outstanding offset is zero.
4. Offset Overflow - the other SCSI device sent a SREQ/ or SACK/ pulse with data which exceeded the maximum synchronous offset defined by the SXFER register.
5. A phase change occurred with an outstanding synchronous offset when the SYM53C875 was operating as an initiator.
6. Residual data in the Synchronous data FIFO - a transfer other than synchronous data receive was started with data left in the synchronous data FIFO.

**Bit 2 UDC (Unexpected Disconnect)**

This bit is set when the SYM53C875 is operating in initiator mode and the target device unexpectedly disconnects from the SCSI bus. This bit is only valid when the SYM53C875 operates in the initiator mode. When the SYM53C875 operates in low level mode, any disconnect will cause an interrupt, even a valid SCSI disconnect. This bit will also be set if a selection time-out occurs (it may occur before, at the same time, or stacked after the STO interrupt, since this is not considered an expected disconnect).

**Bit 1 RST (SCSI RST/ Received)**

This bit is set when the SYM53C875 detects an active SRST/ signal, whether the reset was generated external to the chip or caused by the Assert SRST/ bit in the SCNTL1 register. This SYM53C875 SCSI reset detection logic is edge-sensitive, so that multiple interrupts will not be generated for a single assertion of the SRST/ signal.

**Bit 0 PAR (Parity Error)**

This bit is set when the SYM53C875 detects a parity error while receiving SCSI data. The Enable Parity Checking bit (bit 3 in the SCNTL0 register) must be set for this bit to become active. The SYM53C875 always generates parity when sending SCSI data.

**Register 43 (C3)**

**SCSI Interrupt Status One (SIST1)**

Read Only

RES 7	RES 6	RES 5	RES 4	RES 3	STO 2	GEN 1	HTH 0
Default>>>							
X	X	X	X	X	0	0	0

Reading the SIST1 register returns the status of the various interrupt conditions, whether they are enabled in the SIEN1 register or not. Each bit that is set indicates the corresponding condition has occurred.

Reading the SIST1 will clear the interrupt condition.

**Bits 7-3 Reserved**

**Bit 2 STO (Selection or Reselection Time-out)**

The SCSI device which the SYM53C875 was attempting to select or reselect did not respond within the programmed time-out period. See the description of the STIME0 register, bits 3-0, for more information on the time-out timer.

**Bit 1 GEN (General Purpose Timer Expired)**

This bit is set when the general purpose timer has expired. The time measured is the time between enabling and disabling of the timer. See the description of the STIME1 register, bits 3-0, for more information on the general purpose timer.

**Bit 0 HTH (Handshake-to-Handshake Timer Expired)**

This bit is set when the handshake-to-handshake timer has expired. The time measured is the SCSI Request to Request (target) or Acknowledge to Acknowledge (initiator) period. See the description of the STIME0 register, bits 7-4, for more information on the handshake-to-handshake timer.



## Register 44 (C4) SCSI Longitudinal Parity (SLPAR) Read/Write

The SLPAR register consists of two multiplexed bytes; other register bit settings determine what is displayed at this memory location at any given time. When bit 5 in the SCNTL2 (SLPMD) register is cleared, the chip XORs the high and low bytes of the SLPAR register together to give a single-byte value which is displayed in the SLPAR register. If the SLPMD bit is set, then the SLPAR register shows either the high byte or the low byte of the SLPAR word. The SLPAR High Byte Enable bit, SCNTL2 bit 4, determines which byte of the SLPAR register is visible on the SLPAR register at any given time. If this bit is cleared, the SLPAR register contains the low byte of the SLPAR word; if it is set, the SLPAR register contains the high byte of the SLPAR word.

This register performs a bitwise longitudinal parity check on all SCSI data received or sent through the SCSI core. If one of the bytes received or sent (usually the last) is the set of correct even parity bits, SLPAR should go to zero (assuming it started at zero). As an example, suppose that the following

three data bytes and one check byte are received from the SCSI bus (all signals are shown active high):

Data Bytes	Running SLPAR
---	00000000
1. 11001100	11001100 (XOR of word 1)
2. 01010101	10011001 (XOR of word 1 and 2)
3. 00001111	10010110 (XOR of word 1, 2 and 3) Even Parity >>>10010110
4. 10010110	00000000

A one in any bit position of the final SLPAR value would indicate a transmission error.

The SLPAR register can also be used to generate the check bytes for SCSI send operations. If the SLPAR register contains all zeros prior to sending a block move, it will contain the appropriate check byte at the end of the block move. This byte must then be sent across the SCSI bus.

Note: writing any value to this register resets it to zero.

The longitudinal parity checks are meant to provide an added measure of SCSI data integrity and are entirely optional. This register does not latch SCSI selection/reselection IDs under any circumstances. The default value of this register is zero.

**Register 45 (C5)**  
**SCSI Wide Residue (SWIDE)**  
 Read/Write

After a wide SCSI data receive operation, this register will contain a residual data byte if the last byte received was never sent across the DMA bus. It represents either the first data byte of a subsequent data transfer, or it is a residue byte which should be cleared when an Ignore Wide Residue message is received. It may also be an overrun data byte. The power-up value of this register is indeterminate.

**Register 46 (C6)**  
**Memory Access Control (MACNTL)**  
 Read/Write

TYP3	TYP2	TYP1	TYP0	DWR	DRD	PSCPT	SCPTS
7	6	5	4	3	2	1	0
Default>>>							
0	1	1	1	0	0	0	0

**Bits 7-4 TYP3-0 (Chip Type)**

These bits identify the chip type for software purposes. This data manual applies to devices that have these bits set to 07h.

Bits 3 through 0 of this register are used to determine if an external bus master access is to local or far memory. When bits 3 through 0 are set, the corresponding access is considered local and the MAC/\_TESTOUT pin is driven high. When these bits are clear, the corresponding access is to far memory and the MAC/\_TESTOUT pin is driven low. This function is enabled after a Transfer Control SCRIPTS instruction is executed.

**Bit 3 DWR (DataWR)**

This bit is used to define if a data write is considered local memory access.

**Bit 2 DRD (DataRD)**

This bit is used to define if a data read is considered local memory access.

**Bit 1 PSCPT (Pointer SCRIPTS)**

This bit is used to define if a pointer to a SCRIPTS indirect or table indirect fetch is considered local memory access.

**Bit 0 SCPTS (SCRIPTS)**

This bit is used to define if a SCRIPTS fetch is considered local memory access.

**Register 47 (C7)**  
**General Purpose Pin Control (GPCNTL)**  
 Read/Write

ME	FE	RES	GPIO4	GPIO3	GPIO2	GPIO1	GPIO0
7	6	5	4	3	2	1	0
Default>>>							
0	0	X	0	1	1	1	1

This register is used to determine if the pins controlled by the General Purpose register (GPREG) are inputs or outputs. Bits 4-0 in GPCNTL correspond to bits 4-0 in the GPREG register. When the bits are enabled as inputs, an internal pull-up is also enabled.

**Bit 7 Master Enable**

The internal bus master signal will be presented on GPIO1 if this bit is set, regardless of the state of Bit 1 (GPIO1\_EN).

**Bit 6 Fetch Enable**

The internal op code fetch signal will be presented on GPIO0 if this bit is set, regardless of the state of Bit 0 (GPIO0\_EN).

**Bit 5 Reserved**

**Bits 4-2 GPIO4\_EN-GPIO2\_EN (GPIO Enable)**

General purpose control, corresponding to bits 4-2 in the GPREG register and pins 60, 59, and 57. GPIO4 powers up as a general purpose output, and GPIO3-2 power up as general purpose inputs.

**Bits 1-0 GPIO1\_EN- GPIO0\_EN (GPIO Enable)**

These bits power up set, causing the GPIO1 and GPIO0 pins to become inputs. Resetting these bits causes GPIO1-0 to become outputs.

**Register 48 (C8)**  
**SCSI Timer Zero (STIME0)**  
 Read /Write

HTH	HTH	HTH	HRH	SEL	SEL	SEL	SEL
7	6	5	4	3	2	1	0
Default>>>							
0	0	0	0	0	0	0	0

**Bits 7-4 HTH (Handshake-to-Handshake Timer Period)**

These bits select the handshake-to-handshake time-out period, the maximum time between SCSI handshakes (SREQ/ to SREQ/ in target mode, or SACK/ to SACK/ in initiator mode). When this timing is exceeded, an interrupt is generated and the HTH bit in the SIST1 register is set. The following table contains time-out periods for the Handshake-to-Handshake Timer, the Selection/Reselection Timer (bits 3-0), and the General Purpose Timer (STIME1 bits 3-0). For a more detailed explanation of interrupts, refer to Chapter 2, "Functional Description."

HTH 7-4, SEL 3-0, GEN 3-0	Minimum Time-out)	
	40 MHz/80 MHz	50 MHz
0000	Disabled	Disabled
0001	125 µs	100 µs
0010	250 µs	200 µs
0011	500 µs	400 µs
0100	1 ms	800 µs
0101	2 ms	1.6 ms
0110	4 ms	3.2 ms
0111	8 ms	6.4 ms
1000	16 ms	12.8 ms
1001	32 ms	25.6 ms

*These values will be correct if the CCF bits in the SCNTL3 register are set according to the valid combinations in the bit description.*

HTH 7-4, SEL 3-0, GEN 3-0	Minimum Time-out)	
	40 MHz/80 MHz	50 MHz
1010	64 ms	51.2 ms
1011	128 ms	102.4 ms
1100	256 ms	204.8 ms
1101	512 ms	409.6 ms
1110	1.024 sec	819.2 ms
1111	2.048 sec	1.6384 sec

*These values will be correct if the CCF bits in the SCNTL3 register are set according to the valid combinations in the bit description.*

**Bits 3-0 SEL (Selection Time-Out)**

These bits select the SCSI selection/reselection time-out period. When this timing (plus the 200 µs selection abort time) is exceeded, the STO bit in the SIST1 register is set. For a more detailed explanation of interrupts, refer to Chapter 2, “Functional Description.”

**Register 49 (C9)  
SCSI Timer One (STIME1)  
Read/Write**

RES	HTHBA	GENSF	HTHSF	GEN3	GEN2	GEN1	GEN0
7	6	5	4	3	2	1	0
Default>>>							
X	0	0	0	0	0	0	0

**Bit 7 Reserved**

**Bit 6 HTHBA (Handshake-to-Handshake  
Timer Bus Activity Enable)**

Setting this bit causes this timer to begin testing for SCSI REQ/ACK activity as soon as SBSY/ is asserted, regardless of the agents participating in the transfer.

**Bit 5 GENSF (General Purpose Timer  
Scale Factor)**

Setting this bit causes this timer to shift by a factor of 16.

HTH 7-4, SEL 3-0, GEN 3-0	Minimum Time-out (50 MHz Clock)	
	GENSF= 0	GENSF=1
0000	Disabled	Disabled
0001	100 µs	1.6 ms
0010	200 µs	3.2 ms
0011	400 µs	6.4 ms
0100	800 µs	12.8 ms
0101	1.6 ms	25.6 ms
0110	3.2 ms	51.2 ms
0111	6.4 ms	102.4 ms
1000	12.8 ms	204.8 ms
1001	25.6 ms	409.6 ms
1010	51.2 ms	819.2 ms
1011	102.4 ms	1.6 sec
1100	204.8 ms	3.2 sec
1101	409.6 ms	6.4 sec
1110	819.2 ms	12.8 sec
1111	1.6 sec	25.6 sec

*These values will be correct if the CCF bits in the SCNTL3 register are set according to the valid combinations in the bit description.*

HTH 7-4, SEL 3-0, GEN 3-0	Minimum Time-out (80 MHz Clock)	
	GENSF= 0	GENSF=1
0000	Disabled	Disabled
0001	125 $\mu$ s	2 ms
0010	250 $\mu$ s	4 ms
0011	500 $\mu$ s	8 ms
0100	1 $\mu$ s	16 ms
0101	2 ms	32 ms
0110	4 ms	64 ms
0111	8 ms	128 ms
1000	16 ms	256 ms
1001	32 ms	512 ms
1010	64 ms	1 sec
1011	128 ms	2 sec
1100	256 ms	4.1 sec
1101	512 ms	8.2 sec
1110	1.024 sec	16.4 sec
1111	2.048 sec	32.8 sec

*These values will be correct if the CCF bits in the SCNTL3 register are set according to the valid combinations in the bit description.*

the SIST1 register is set. Refer to the table under STIME0, bits 3-0, for the available time-out periods.

Note: to reset a timer before it has expired and obtain repeatable delays, the time value must be written to zero first, and then written back to the desired value. This is also required when changing from one time value to another. See Chapter 2, "Functional Description," for an explanation of how interrupts will be generated when the timers expire.

#### Bit 4 HTHSF (Handshake to Handshake Timer Scale Factor)

Setting this bit causes this timer to shift by a factor of 16.

#### Bits 3-0 GEN3-0 (General Purpose Timer Period)

These bits select the period of the general purpose timer. The time measured is the time between enabling and disabling of the timer. When this timing is exceeded, the GEN bit in

**Register 4A (CA)**  
**Response ID Zero (RESPID0)**  
 Read/Write

**Register 4B (CB)**  
**Response ID One (RESPID1)**  
 Read/Write

RESPID0 and RESPID1 contain the selection or reselection IDs. In other words, these two 8-bit registers contain the ID that the chip responds to on the SCSI bus. Each bit represents one possible ID with the most significant bit of RESPID1 representing ID 15 and the least significant bit of RESPID0 representing ID 0. The SCID register still contains the chip ID used during arbitration. The chip can respond to more than one ID because more than one bit can be set in the RESPID1 and RESPID0 registers. However, the chip can arbitrate with only one ID value in the SCID register.

**Register 4C (CC)**  
**SCSI Test Zero (STEST0)**  
 Read Only

SSAID3	SSAID2	SSAID1	SSAID0	SLT	ART	SOZ	SOM
7	6	5	4	3	2	1	0
Default >>>							
0	0	0	0	0	X	1	1

**Bits 7-4 SSAID (SCSI Selected As ID)**

These bits contain the encoded value of the SCSI ID that the SYM53C875 was selected or reselected as during a SCSI selection or reselection phase. These bits are read only and contain the encoded value of 0-15 possible IDs that could be used to select the SYM53C875. During a SCSI selection or reselection phase when a valid ID has been put on the bus, and the 53C875 responds to that ID, the “selected as” ID is written into these bits. These bits are used with the RESPID registers to allow response to multiple IDs on the bus.

**Bit 3 SLT (Selection Response Logic Test)**

This bit is set when the SYM53C875 is ready to be selected or reselected. This does not take into account the bus settle delay of 400 ns. This bit is used for functional test and fault purposes.

**Bit 2 ART (Arbitration Priority Encoder Test)**

This bit will always be set when the SYM53C875 exhibits the highest priority ID asserted on the SCSI bus during arbitration. It is primarily used for chip level testing, but it may be used during low level mode operation to determine if the SYM53C875 has won arbitration.

**Bit 1 SOZ (SCSI Synchronous Offset Zero)**

This bit indicates that the current synchronous SREQ/SACK offset is zero. This bit is not latched and may change at any time. It is used in low level synchronous SCSI operations. When this bit is set, the SYM53C875, as an initiator, is waiting for the target to request data transfers. If the SYM53C875 is a target, then the initiator has sent the offset number of acknowledges.

**Bit 0 SOM (SCSI Synchronous Offset Maximum)**

This bit indicates that the current synchronous SREQ/SACK offset is the maximum specified by bits 3-0 in the SCSI Transfer register. This bit is not latched and may change at any time. It is used in low level synchronous SCSI operations. When this bit is set, the SYM53C875, as a target, is waiting for the initiator to acknowledge the data transfers. If the SYM53C875 is an initiator, then the target has sent the offset number of requests.

**Register 4D (CD)  
SCSI Test One (STEST1)  
Read/Write**

SCLK	SISO	RES	RES	RES	RES	RES	RES
7	6	5	4	3	2	1	0
Default>>>							
0	0	X	X	X	X	X	X

**Bit 7 SCLK**

This bit, when set, will disable the external SCLK (SCSI Clock) pin, and the chip will use the PCI clock as the internal SCSI clock. If a transfer rate of 10 MB/s (or 20 MB/s on a wide SCSI bus) is to be achieved on the SCSI bus, this bit must be reset and at least a 40 MHz external SCLK must be provided.

**Bit 6 SISO (SCSI Isolation Mode)**

This bit allows the SYM53C875 to put the SCSI bi-directional and input pins into a low power mode when the SCSI bus is not in use. When this bit is set, the SCSI bus inputs are logically isolated from the SCSI bus.

**Bits 5-4 Reserved**

**Bit 3 SCLK Doubler Enable (DBLEN)**

Set this bit to bring the SCSI clock doubler out of the powered-down state. The default value of this bit is clear (SCSI clock doubler powered down). Set bit 2 after setting this bit, to double the SCLK frequency.

**Bit 2 SCLK Doubler Select (DBLSEL)**

Set this bit after powering up the SCSI clock doubler to double The SCLK frequency. This bit has no effect unless bit 3 is set.

**Bits1-0 Reserved**

### Doubling the SCSI CLK Frequency

The SYM53C875 SCSI clock doubler doubles a 40 MHz SCSI clock, increasing the frequency to 80 MHz. Follow these steps to use the clock doubler:

1. Set the SCLK Doubler Enable bit (STEST1, bit 3)
2. Wait 20  $\mu$ s
3. Halt the SCSI clock by setting the Halt SCSI Clock bit (STEST3 bit 5)
4. Set the clock conversion factor using the SCF and CCF fields in the SCNTL3 register
5. Set the SCLK Doubler Select bit (STEST1, bit2)
6. Clear the Halt SCSI Clock bit

### Register 4E (CE) SCSI Test Two (STEST2) Read/Write

SCE	ROF	DIF	SLB	SZM	AWS	EXT	LOW
7	6	5	4	3	2	1	0
Default>>>							
0	0	0	0	0	0	0	0

#### Bit 7 SCE (SCSI Control Enable)

This bit, when set, allows all SCSI control and data lines to be asserted through the SOCL and SODL registers regardless of whether the SYM53C875 is configured as a target or initiator.

Note: this bit should not be set during normal operation, since it could cause contention on the SCSI bus. It is included for diagnostic purposes only.

#### Bit 6 ROF (Reset SCSI Offset)

Setting this bit clears any outstanding synchronous SREQ/SACK offset. This bit should be set if a SCSI gross error condition occurs, to clear the offset when a synchronous transfer does not complete successfully. The bit automatically clears itself after resetting the synchronous offset.

#### Bit 5 DIF (SCSI Differential Mode)

Setting this bit allows the SYM53C875 to interface properly to external differential transceivers. Its only real effect is to tri-state the SBSY/, SSEL/, and SRST/ pads so that they can be used as pure inputs. Clearing this bit enables single-ended mode operation. This bit should be set in the initialization routine if the differential pair interface is used.

#### Bit 4 SLB (SCSI Loopback Mode)

Setting this bit allows the SYM53C875 to perform SCSI loopback diagnostics. That is, it enables the SCSI core to simultaneously perform as both initiator and target.



**Bit 3 SZM (SCSI High-Impedance Mode)**

Setting this bit places all the open-drain 48 mA SCSI drivers into a high-impedance state. This is to allow internal loopback mode operation without affecting the SCSI bus.

**Bit 2 AWS (Always Wide SCSI)**

When this bit is set, all SCSI information transfers will be done in 16-bit wide mode. This includes data, message, command, status and reserved phases. This bit should normally be deasserted since 16-bit wide message, command, and status phases are not supported by the SCSI specifications.

**Bit 1 EXT (Extend SREQ/SACK Filtering)**

Symbios TolerANT SCSI receiver technology includes a special digital filter on the SREQ/ and SACK/ pins which will cause glitches on deasserting edges to be disregarded. Setting this bit will increase the filtering period from 30ns to 60ns on the deasserting edge of the SREQ/ and SACK/ signals.

**Note:** this bit must never be set during fast SCSI (greater than 5M transfers per second) operations, because a valid assertion could be treated as a glitch.

**Note:** this bit does not affect the filtering period when the Fast-20 Enable bit in the SCNTL3 register is set. When the SYM53C875 is executing UltraSCSI transfers, the filtering period is automatically set at 15 ns.

**Bit 0 LOW (SCSI Low level Mode)**

Setting this bit places the SYM53C875 in low level mode. In this mode, no DMA operations occur, and no SCRIPTS execute. Arbitration and selection may be performed by setting the start sequence bit as described in the SCNTL0 register. SCSI bus transfers are performed by manually asserting and polling SCSI signals. Clearing this bit allows instructions to be executed in SCSI SCRIPTS mode.

**Note:** it is not necessary to set this bit for access to the SCSI bit-level registers (SODL, SBCL, and input registers).

**Register 4F (CF)  
SCSI Test Three (STEST3)  
Read/Write**

TE	STR	HSC	DSI	S16	TTM	CSF	STW
7	6	5	4	3	2	1	0
Default>>>							
0	0	0	0	0	0	0	0

**Bit 7 TE (TolerANT Enable)**

Setting this bit enables the active negation portion of Symbios TolerANT technology. Active negation causes the SCSI Request, Acknowledge, Data, and Parity signals to be actively deasserted, instead of relying on external pull-ups, when the SYM53C875 is driving these signals. Active deassertion of these signals will occur only when the SYM53C875 is in an information transfer phase. When operating in a differential environment or at fast SCSI timings, TolerANT Active negation should be enabled to improve setup and deassertion times. Active negation is disabled after reset or when this bit is cleared. For more information on Symbios TolerANT technology, refer to Chapter 1.

**Note:** this bit must be set if the Enable Fast 20 bit in SCNTL3 is set.

**Bit 6 STR (SCSI FIFO Test Read)**

Setting this bit places the SCSI core into a test mode in which the SCSI FIFO can be easily read. Reading the least significant byte of the SODL register will cause the FIFO to unload. The functions are summarized in the table below.

Register Name	Register Operation	FIFO Bits	FIFO Function
SODL	Read	15-0	Unload
SODL0	Read	7-0	Unload
SODL1	Read	15-8	None

**Bit 5 HSC (Halt SCSI Clock)**

Asserting this bit causes the internal divided SCSI clock to come to a stop in a glitchless manner. This bit may be used for test purposes or to lower I<sub>DD</sub> during a power down mode.

This bit is used when the SCSI clock doubler is operating. For additional information on the clock doubler, please see Chapter 2.

the SCSI FIFO pointers, the SIDL, SODL, and SODR full bits in the SSTAT0 and SSTAT2 are cleared.

**Bit 4 DSI (Disable Single Initiator Response)**

If this bit is set, the SYM53C875 will ignore all bus-initiated selection attempts that employ the single-initiator option from SCSI-1. In order to select the SYM53C875 while this bit is set, the SYM53C875's SCSI ID and the initiator's SCSI ID must both be asserted. This bit should be asserted in SCSI-2 systems so that a single bit error on the SCSI bus will not be interpreted as a single initiator response.

**Bit 0 STW (SCSI FIFO Test Write)**

Setting this bit places the SCSI core into a test mode in which the FIFO can easily be read or written. While this bit is set, writes to the least significant byte of the SODL register will cause the entire word contained in this register to be loaded into the FIFO. Writing the least significant byte of the SODL register will cause the FIFO to load. These functions are summarized in the table below:

**Bit 3 S16 (16-bit System)**

If this bit is set, all devices in the SCSI system implementation are assumed to be 16 bits. This causes the SYM to always check the parity bit for SCSI IDs 15-8 during bus-initiated selection or reselection, assuming parity checking has been enabled. If an 8-bit SCSI device attempts to select the SYM53C875 while this bit is set, the SYM53C875 will ignore the selection attempt, because the parity bit for IDs 15-8 will be undriven. See the description of the Enable Parity Checking bit in the SCNTL0 register for more information.

Register Name	Register Operation	FIFO Bits	FIFO Function
SODL	Write	15-0	Load
SODL0	Write	7-0	Load
SODL1	Write	15-8	None

**Bit 2 TTM (Timer Test Mode)**

Asserting this bit facilitates testing of the selection time-out, general purpose, and handshake-to-handshake timers by greatly reducing all three time-out periods. Setting this bit starts all three timers and if the respective bits in the SIEN1 register are asserted, the SYM53C875 will generate interrupts at time-out. This bit is intended for internal manufacturing diagnosis and should not be used.

**Bit 1 CSF (Clear SCSI FIFO)**

Setting this bit will cause the "full flags" for the SCSI FIFO to be cleared. This empties the FIFO. This bit is self-resetting. In addition to

---

**Register 50-51 (D0-D1)**  
**SCSI Input Data Latch (SIDL)**  
**Read Only**

This register is used primarily for diagnostic testing, programmed I/O operation, or error recovery. Data received from the SCSI bus can be read from this register. Data can be written to the SODL register and then read back into the SYM53C875 by reading this register to allow loopback testing. When receiving SCSI data, the data will flow into this register and out to the host FIFO. This register differs from the SBDL register; SIDL contains latched data and the SBDL always contains exactly what is currently on the SCSI data bus. Reading this register causes the SCSI parity bit to be checked, and will cause a parity error interrupt if the data is not valid. The power-up values are indeterminate.

---

**Registers 54-55 (D4-D5)**  
**SCSI Output Data Latch (SODL)**  
**Read/Write**

This register is used primarily for diagnostic testing or programmed I/O operation. Data written to this register is asserted onto the SCSI data bus by setting the Assert Data Bus bit in the SCNTL1 register. This register is used to send data via programmed I/O. Data flows through this register when sending data in any mode. It is also used to write to the synchronous data FIFO when testing the chip. The power-up value of this register is indeterminate.

---

**Registers 58-59 (D8-D9)**  
**SCSI Bus Data Lines (SBDL)**  
**Read Only**

This register contains the SCSI data bus status. Even though the SCSI data bus is active low, these bits are active high. The signal status is not latched and is a true representation of exactly what is on the data bus at the time the register is read. This register is used when receiving data via programmed I/O. This register can also be used for diagnostic testing or in low level mode. The power-up value of this register is indeterminate.

---

**Registers 5C-5F (DC-DF)**  
**Scratch Register B (SCRATCHB)**  
**(Read/Write)**

This is a general purpose user definable scratch pad register. Apart from CPU access, only Register Read/Write and Memory Moves directed at the SCRATCH register will alter its contents. The SYM53C875 cannot fetch SCRIPTS instructions from this location. When bit 3 in the CTEST2 register is set, this register contains the base address for the 4 KB internal RAM. Setting CTEST2 bit 3 only causes the base address to appear in the SCRATCHB register; any information that was previously in the register will remain intact. Any writes to this register while the bit is set will pass through to the actual SCRATCHB register. The power-up values are indeterminate.

---

**Registers 60h-7Fh (E0h-FFh)  
Scratch Registers C-J  
(SCRATCHC-SCRATCHJ)  
Read/Write**

These registers are general-purpose scratch registers for user-defined functions. The SYM53C875 cannot fetch SCRIPTS instructions from this location. The power up value of these registers is indeterminate.

(

(

## Chapter 6

# Instruction Set of the I/O Processor

After power up and initialization of the SYM53C875, the chip may be operated in the low level register interface mode, or using SCSI SCRIPTS.

With the low level register interface, the user has access to the DMA control logic and the SCSI bus control logic. An external processor has access to the SCSI bus signals and the low level DMA signals, which allows creation of complicated board level test algorithms. The low level interface is useful for backward compatibility with SCSI devices that require certain unique timings or bus sequences to operate properly. Another feature allowed at the low level is loopback testing. In loopback mode, the SCSI core can be directed to talk to the DMA core to test internal data paths all the way out to the chip's pins.

## SCSI SCRIPTS

To operate in the SCSI SCRIPTS mode, the SYM53C875 requires only a SCRIPTS start address. The start address must be at a longword (four byte) boundary. This will align all the following SCRIPTS at a longword boundary since all SCRIPTS are 8 or 12 bytes long.

Instructions are fetched until an interrupt instruction is encountered, or until an unexpected event (such as a hardware error) causes an interrupt to the external processor.

Once an interrupt is generated, the SYM53C875 halts all operations until the interrupt is serviced. Then, the start address of the next SCRIPTS instruction may be written to the DMA SCRIPTS Pointer register to restart the automatic fetching and execution of instructions.

The SCSI SCRIPTS mode of execution allows the SYM53C875 to make decisions based on the status of the SCSI bus, which off-loads the microprocessor from servicing the numerous interrupts inherent in I/O operations.

Given the rich set of SCSI-oriented features included in the instruction set, and the ability to re-enter the SCSI algorithm at any point, this high level interface is all that is required for both normal and exception conditions. Switching to low level mode for error recovery should never be required.

The following types of SCRIPTS instructions are implemented in the SYM53C875:

- Block Move—used to move data between the SCSI bus and memory
- I/O or Read/Write—causes the SYM53C875 to trigger common SCSI hardware sequences, or to move registers
- Transfer Control—allows SCRIPTS instructions to make decisions based on real time SCSI bus conditions
- Memory Move—causes the SYM53C875 to execute block moves between different parts of main memory
- Load and Store—provides a more efficient way to move data to/from memory from/to an internal register in the chip without using the Memory Move instruction

Each instruction consists of two or three 32-bit words. The first 32-bit word is always loaded into the DCMD and DBC registers, the second into the DSPS register. The third word, used only by Memory Move instructions, is loaded into the TEMP shadow register. In an indirect I/O or Move instruction, the first two 32-bit op code fetches will be followed by one or two more 32-bit fetch cycles.

---

## Sample Operation

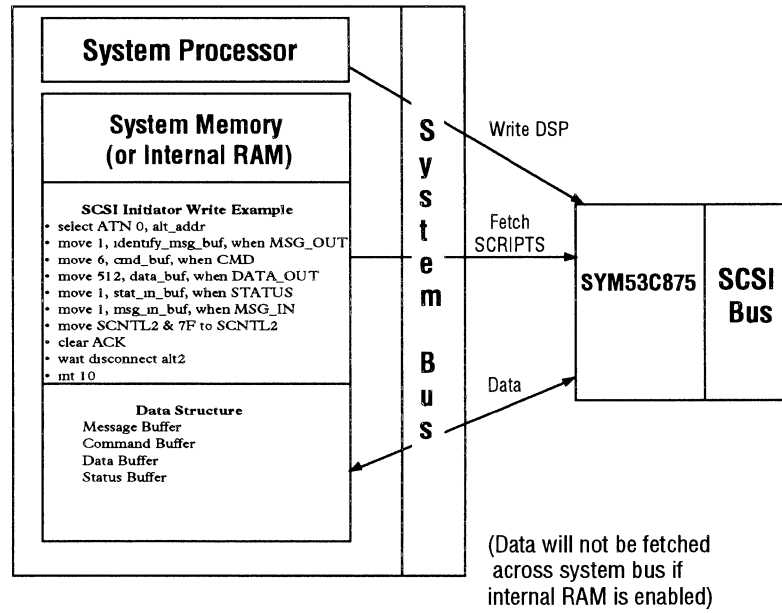
The following example describes execution of a SCRIPTS instruction. This sample operation is for a Block Move instruction.

1. The host CPU, through programmed I/O, gives the DMA SCRIPTS Pointer (DSP) register (in the Operating Register file) the starting address in main memory that points to a SCSI SCRIPTS program for execution.
2. Loading the DSP register causes the SYM53C875 to fetch its first instruction at the address just loaded. This will be from main memory or the internal RAM, depending on the address.
3. The SYM53C875 typically fetches two longwords (64 bits) and decodes the high order byte of the first longword as a SCRIPTS instruction. If the instruction is a Block Move, the lower three bytes of the first longword are stored and interpreted as the number of bytes to be moved. The second longword is stored and interpreted as the 32-bit beginning address in main memory to which the move is directed.
4. For a SCSI send operation, the SYM53C875 waits until there is enough space in the DMA FIFO to transfer a programmable size block of data. For a SCSI receive operation, it waits until enough data is collected in the DMA FIFO for transfer to memory. At this point, the SYM53C875 requests use of the PCI bus again to transfer the data.
5. When the SYM53C875 is granted the PCI bus, it will execute (as a bus master) a burst transfer (programmable size) of data, decrement the internally stored remaining byte count, increment the address pointer, and then release the PCI bus. The SYM53C875 stays off the PCI bus until the FIFO can again hold (for a write) or has collected (for a read) enough data to repeat the process.

The process repeats until the internally stored byte count has reached zero. The SYM53C875 releases the PCI bus and then performs another SCRIPTS

instruction fetch cycle, using the incremented stored address maintained in the DMA SCRIPTS Pointer register. Execution of SCRIPTS instructions continues until an error condition occurs or an interrupt SCRIPTS instruction is received. At this point, the SYM53C875 interrupts the host CPU and waits for further servicing by the host system. It can execute independent Block Move instructions specifying new byte counts and starting locations in main memory. In this manner, the SYM53C875 performs scatter/gather operations on data without requiring help from the host program, generating a host interrupt, or requiring an external DMA controller to be programmed.





**Figure 6-1: SCRIPTS Overview**

## Block Move Instructions

For Block Move instructions, bits 5 and 4 (SIOM and DIOM) in the DMODE register determine whether the source/destination address resides in memory or I/O space. When data is being moved onto the SCSI bus, SIOM controls whether that data comes from I/O or memory space. When data is being moved off of the SCSI bus, DIOM controls whether that data goes to I/O or memory space.

### First Dword

#### Bits 31-30 Instruction Type-Block Move

##### Bit 29 Indirect Addressing

When this bit is cleared, user data is moved to or from the 32-bit data start address for the Block Move instruction. The value is loaded into the chip's address register and incremented as data is transferred. The address of the data to be moved is in the second dword of this instruction.

When set, the 32-bit user data start address for the Block Move is the address of a pointer to the actual data buffer address. The value at the 32-bit start address is loaded into the chip's DNAD register via a third longword fetch (4-byte transfer across the host computer bus).

##### Direct

The byte count and absolute address are as follows.

Command	Byte Count
Address of Data	

##### Indirect

Use the fetched byte count, but fetch the data address from the address in the instruction.

Command	Byte Count
Address of Pointer to Data	

Once the data pointer address is loaded, it is executed as when the chip operates in the direct mode. This indirect feature allows a table of data buffer addresses to be specified. Using the NCR SCSI SCRIPTS compiler, the table offset is placed in the script at compile time. Then at the actual data transfer time, the offsets are added to the base address of the data address table by the external processor. The logical I/O driver builds a structure of addresses for an I/O rather than treating each address individually. This feature makes it possible to locate SCSI SCRIPTS in a PROM.

Note: indirect and table indirect addressing cannot be used simultaneously; only one addressing method may be used at a time.

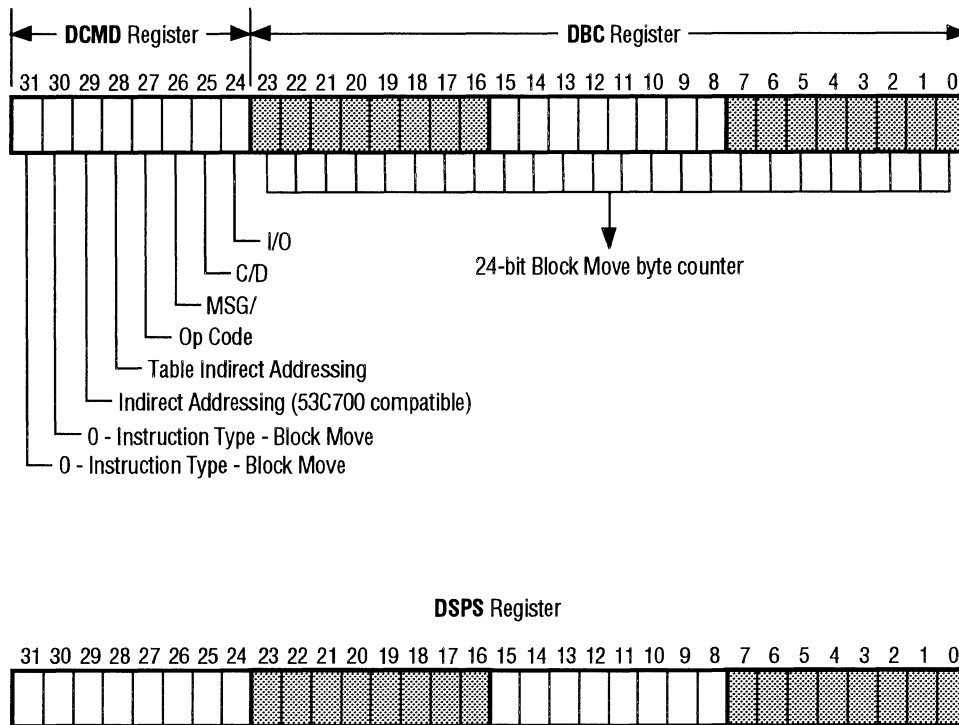
##### Bit 28 Table Indirect

When this bit is set, the 24-bit signed value in the start address of the move is treated as a relative displacement from the value in the DSA register. Both the transfer count and the source/destination address are fetched from this address.

Use the signed integer offset in bits 23-0 of the second four bytes of the instruction, added to the value in the DSA register, to fetch first the byte count and then the data address. The signed value is combined with the data structure base address to generate the physical address used to fetch values from the data structure. Sign-extended values of all ones for negative values are allowed, but bits 31-24 are ignored.

Command	Not Used
Don't Care	Table Offset

Note: indirect and table indirect addressing cannot be used simultaneously; only one addressing method may be used at a time.



**Figure 6-2: Block Move Instruction Register**

Prior to the start of an I/O, the Data Structure Base Address register (DSA) should be loaded with the base address of the I/O data structure. The address may be any address on a long word boundary.

After a Table Indirect op code is fetched, the DSA is added to the 24-bit signed offset value from the op code to generate the address of the required data; both positive and negative offsets are allowed. A subsequent fetch from that address brings the data values into the chip.

For a MOVE instruction, the 24-bit byte count is fetched from system memory. Then the 32-bit physical address is brought into the SYM53C875. Execution of the move begins at this point.

SCRIPTS can directly execute operating system I/O data structures, saving time at the beginning of an I/O operation. The I/O data structure can begin on any longword boundary and may cross system segment boundaries.

There are two restrictions on the placement of pointer data in system memory: the eight bytes of data in the MOVE instruction must be contiguous, as shown below; and indirect data fetches are not available during execution of a Memory-to-Memory DMA operation.

00	Byte Count
Physical Data Address	

**Bit 27 Op Code**

This 1-bit field defines the instruction to be executed as a block move (MOVE).

**Target Mode**

OPC	Instruction Defined
0	MOVE
1	CHMOV

1. The SYM53C875 verifies that it is connected to the SCSI bus as a target before executing this instruction.
2. The SYM53C875 asserts the SCSI phase signals (MSG/, SC\_D/, and SI\_O/) as defined by the Phase Field bits in the instruction.
3. If the instruction is for the command phase, the SYM53C875 receives the first command byte and decodes its SCSI Group Code.
  - a) If the SCSI Group Code is either Group 0, Group 1, Group 2, or Group 5, and if the Vendor Unique Enhancement 1 (VUE1) bit (SCNTL2 bit 1) is clear, then the SYM53C875 overwrites the DBC register with the length of the Command Descriptor Block: 6, 10, or 12 bytes.
  - b) If the Vendor Unique Enhancement 1 (VUE1) bit (SCNTL2 bit 1) is set, the SYM53C875 receives the number of bytes in the byte count regardless of the group code.
  - c) If the Vendor Unique Enhancement 1 bit is clear and group code is vendor unique, the SYM53C875 receives the number of bytes in the count regardless of the group code.
  - d) If any other Group Code is received, the DBC register is not modified and the SYM53C875 will request the number of bytes specified in the DBC register. If the DBC register contains 000000h, an illegal instruction interrupt is generated.

4. The SYM53C875 transfers the number of bytes specified in the DBC register starting at the address specified in the DNAD register. If the Op Code bit is set and a data transfer ends on an odd byte boundary, the SYM53C875 will store the last byte in the SCSI Wide Residue Data Register during a receive operation. This byte will be combined with the first byte from the subsequent transfer so that a wide transfer can be completed.
5. If the SATN/ signal is asserted by the initiator or a parity error occurred during the transfer, the transfer can optionally be halted and an interrupt generated. The Disable Halt on Parity Error or ATN bit in the SCNTL1 register controls whether the SYM53C875 will halt on these conditions immediately, or wait until completion of the current Move.

**Initiator Mode**

OPC	Instruction Defined
0	CHMOV
1	MOVE

1. The SYM53C875 verifies that it is connected to the SCSI bus as an initiator before executing this instruction.
2. The SYM53C875 waits for an unserviced phase to occur. An unserviced phase is defined as any phase (with SREQ/ asserted) for which the SYM53C875 has not yet transferred data by responding with a SACK/.
3. The SYM53C875 compares the SCSI phase bits in the DCMD register with the latched SCSI phase lines stored in the SSTAT1 register. These phase lines are latched when SREQ/ is asserted.
4. If the SCSI phase bits match the value stored in the SSTAT1 register, the SYM53C875 will transfer the number of bytes specified in the DBC register starting at the address pointed to by the DNAD register. If the op code bit is

cleared and a data transfer ends on an odd byte boundary, the SYM53C875 will store the last byte in the SCSI Wide Residue Data Register during a receive operation, or in the SCSI Output Data Latch Register during a send operation. This byte will be combined with the first byte from the subsequent transfer so that a wide transfer can be completed.

5. If the SCSI phase bits do not match the value stored in the SSTAT1 register, the SYM53C875 generates a phase mismatch interrupt and the instruction is not executed.
6. During a Message Out phase, after the SYM53C875 has performed a select with Attention (or SATN/ has been manually asserted with a Set ATN instruction), the SYM53C875 will deassert SATN/ during the final SREQ/SACK handshake.
7. When the SYM53C875 is performing a block move for Message In phase, it will not deassert the SACK/ signal for the last SREQ/SACK handshake. The SACK signal must be cleared using the Clear SACK I/O instruction.

**Bits 26-24 SCSI Phase**

This 3-bit field defines the desired SCSI information transfer phase. When the SYM53C875 operates in initiator mode, these bits are compared with the latched SCSI phase bits in the SSTAT1 register. When the SYM53C875 operates in target mode, the SYM53C875 asserts the phase defined in this field. The following table describes the possible combinations and the corresponding SCSI phase.

MSG	C/D	I/O	SCSI Phase
0	0	0	Data out
0	0	1	Data in
0	1	0	Command
0	1	1	Status
1	0	0	Reserved out
1	0	1	Reserved in
1	1	0	Message out
1	1	1	Message in

**Bits 23-0 Transfer Counter**

This 24-bit field specifies the number of data bytes to be moved between the SYM53C875 and system memory. The field is stored in the DBC register. When the SYM53C875 transfers data to/from memory, the DBC register is decremented by the number of bytes transferred. In addition, the DNAD register is incremented by the number of bytes transferred. This process is repeated until the DBC register has been decremented to zero. At that time, the SYM53C875 fetches the next instruction.

If bit 28 is set, indicating table indirect addressing, this field is not used. The byte count is instead fetched from a table pointed to by the DSA register.

**Second Dword**

**Bits 31-0 Start Address**

This 32-bit field specifies the starting address of the data to be moved to/from memory. This field is copied to the DNAD register. When the SYM53C875 transfers data to or from memory, the DNAD register is incremented by the number of bytes transferred.

When bit 29 is set, indicating indirect addressing, this address is a pointer to an address in memory that points to the data location. When bit 28 is set, indicating table indirect addressing, the value in this field is an offset into a table pointed to by the DSA. The table entry contains byte count and address information.

# I/O Instructions

## First Dword

### Bits 31-30 Instruction Type - I/O Instruction

#### Bits 29-27 Op Code

The following Op Code bits have different meanings, depending on whether the SYM53C875 is operating in initiator or target mode. Note: Op Code selections 101-111 are considered Read/Write instructions, and are described in that section.

### Target Mode

OPC2	OPC1	OPC0	Instruction Defined
0	0	0	Reselect
0	0	1	Disconnect
0	1	0	Wait Select
0	1	1	Set
1	0	0	Clear

#### Reselect Instruction

1. The SYM53C875 arbitrates for the SCSI bus by asserting the SCSI ID stored in the SCID register. If the SYM53C875 loses arbitration, then it tries again during the next available arbitration cycle without reporting any lost arbitration status.
2. If the SYM53C875 wins arbitration, it attempts to reselect the SCSI device whose ID is defined in the destination ID field of the instruction. Once the SYM53C875 has won arbitration, it fetches the next instruction from the address pointed to by the DSP register. Therefore, the SCRIPTS can move on to the next instructions before the reselection has completed. It will continue executing SCRIPTS until a SCRIPT that requires a response from the initiator is encountered.
3. If the SYM53C875 is selected or reselected before winning arbitration, it fetches the next

instruction from the address pointed to by the 32-bit jump address field stored in the DNAD register. The SYM53C875 should manually be set to initiator mode if it is reselected, or to target mode if it is selected.

#### Disconnect Instruction

The SYM53C875 disconnects from the SCSI bus by deasserting all SCSI signal outputs.

#### Wait Select Instruction

1. If the SYM53C875 is selected, it fetches the next instruction from the address pointed to by the DSP register.
2. If reselected, the SYM53C875 fetches the next instruction from the address pointed to by the 32-bit jump address field stored in the DNAD register. The SYM53C875 should manually be set to initiator mode when reselected.
3. If the CPU sets the SIGP bit in the ISTAT register, the SYM53C875 will abort the Wait Select instruction and fetch the next instruction from the address pointed to by the 32-bit jump address field stored in the DNAD register.

#### Set Instruction

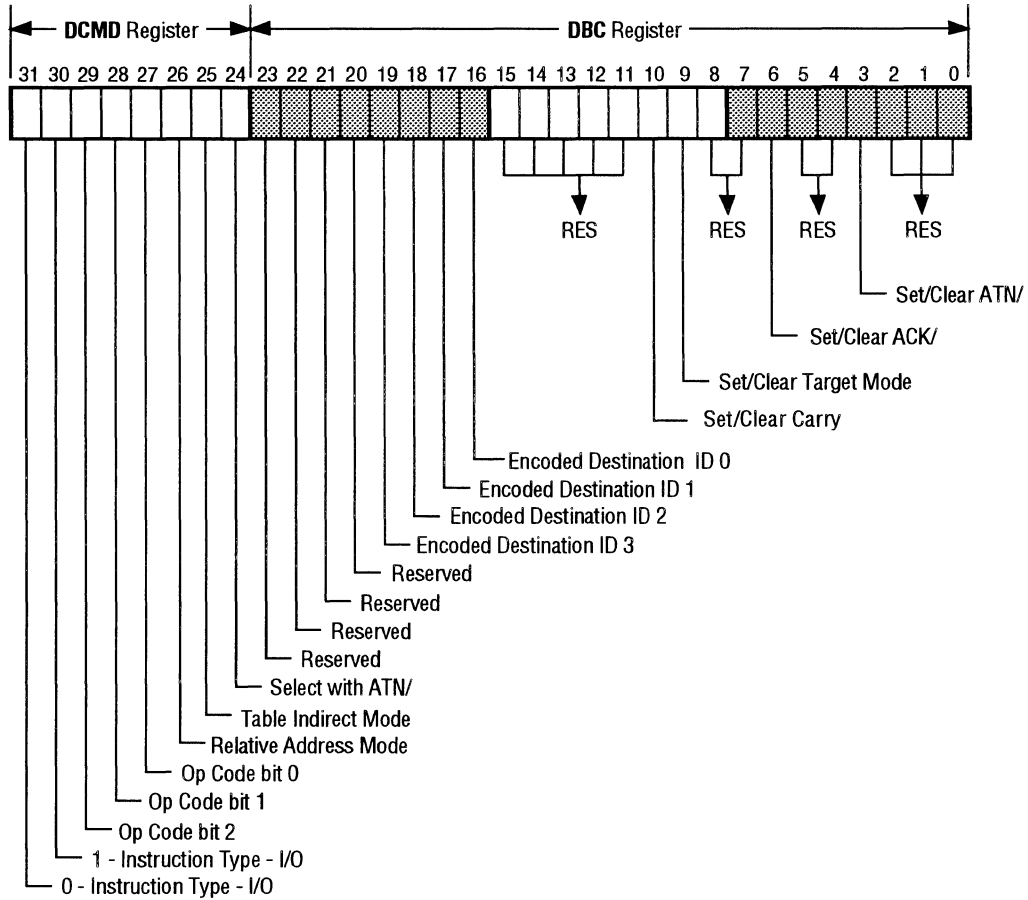
When the SACK/ or SATN/ bits are set, the corresponding bits in the SOCL register are set. SACK/ or SATN/ should not be set except for testing purposes. When the target bit is set, the corresponding bit in the SCNTL0 register is also set. When the carry bit is set, the corresponding bit in the Arithmetic Logic Unit (ALU) is set.

Note: None of the signals are set on the SCSI bus in target mode.

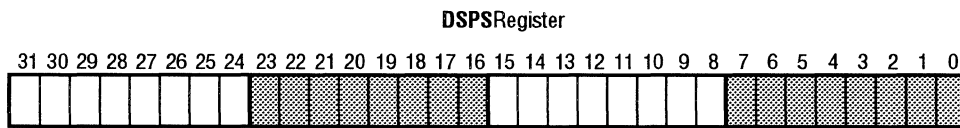
#### Clear Instruction

When the SACK/ or SATN/ bits are set, the corresponding bits are cleared in the SOCL register. SACK/ or SATN/ should not be set except for testing purposes. When the target bit is set, the corresponding bit in the SCNTL0 register is cleared. When the carry bit is set, the corresponding bit in the ALU is cleared.

Note: None of the signals are reset on the SCSI bus in target mode.



Second 32-bit word of the I/O instruction



32-bit Jump Address

**Figure 6-3: I/O Instruction Register**

### Initiator Mode

OPC2	OPC1	OPC0	Instruction Defined
0	0	0	Select
0	0	1	Wait Disconnect
0	1	0	Wait Reselect
0	1	1	Set
1	0	0	Clear

#### Select Instruction

1. The SYM53C875 arbitrates for the SCSI bus by asserting the SCSI ID stored in the SCID register. If the SYM53C875 loses arbitration, it tries again during the next available arbitration cycle without reporting any lost arbitration status.
2. If the SYM53C875 wins arbitration, it attempts to select the SCSI device whose ID is defined in the destination ID field of the instruction. Once the SYM53C875 has won arbitration, it fetches the next instruction from the address pointed to by the DSP register. Therefore, the scripts can move to the next instruction before the selection has completed. It will continue executing SCRIPTS until a SCRIPT that requires a response from the target is encountered.
3. If the SYM53C875 is selected or reselected before winning arbitration, it fetches the next instruction from the address pointed to by the 32-bit jump address field stored in the DNAD register. The SYM53C875 should manually be set to initiator mode if it is reselected, or to target mode if it is selected.
4. If the Select with SATN/ field is set, the SATN/ signal is asserted during the selection phase.

#### Wait Disconnect Instruction

The SYM53C875 waits for the target to perform a “legal” disconnect from the SCSI bus. A “legal” disconnect occurs when SBSY/ and SSEL/ are

inactive for a minimum of one Bus Free delay (400 ns), after the SYM53C875 has received a Disconnect Message or a Command Complete Message.

#### Wait Reselect Instruction

1. If the SYM53C875 is selected before being reselected, it fetches the next instruction from the address pointed to by the 32-bit jump address field stored in the DNAD register. The SYM53C875 should be manually set to target mode when selected.
2. If the SYM53C875 is reselected, it fetches the next instruction from the address pointed to by the DSP register.
3. If the CPU sets the SIGP bit in the ISTAT register, the SYM53C875 will abort the Wait Reselect instruction and fetch the next instruction from the address pointed to by the 32-bit jump address field stored in the DNAD register.

#### Set Instruction

When the SACK/ or SATN/ bits are set, the corresponding bits in the SOCL register are set. When the target bit is set, the corresponding bit in the SCNTL0 register is also set. When the carry bit is set, the corresponding bit in the ALU is set.

#### Clear Instruction

When the SACK/or SATN/ bits are set, the corresponding bits are cleared in the SOCL register. When the target bit is set, the corresponding bit in the SCNTL0 register is cleared. When the carry bit is set, the corresponding bit in the ALU is cleared.

### Bit 26 Relative Addressing Mode

When this bit is set, the 24-bit signed value in the DNAD register is used as a relative displacement from the current DSP address. This bit should only be used in conjunction with the Select, Reselect, Wait Select, and Wait Reselect instructions. The Select and Reselect instructions can contain an absolute alternate jump address or a relative transfer address.



**Bit 25 Table Indirect Mode**

When this bit is set, the 24-bit signed value in the DBC register is added to the value in the DSA register, used as an offset relative to the value in the Data Structure Base Address (DSA) register. The SCNTL3 value, SCSI ID, synchronous offset and synchronous period are loaded from this address. Prior to the start of an I/O, the DSA should be loaded with the base address of the I/O data structure. The address may be any address on a longword boundary. After a Table Indirect op code is fetched, the DSA is added to the 24-bit signed offset value from the op code to generate the address of the required data; both positive and negative offsets are allowed. A subsequent fetch from that address brings the data values into the chip.

SCRIPTS can directly execute operating system I/O data structures, saving time at the beginning of an I/O operation. The I/O data structure can begin on any longword boundary and may cross system segment boundaries. There are two restrictions on the placement of data in system memory:

1. The I/O data structure must lie within the 8 MB above or below the base address.
2. An I/O command structure must have all four bytes contiguous in system memory, as shown below. The offset/period bits are ordered as in the SXFER register. The configuration bits are ordered as in the SCNTL3 register.

Config	ID	Offset/ period	(00)
--------	----	-------------------	------

This bit should only be used in conjunction with the Select, Reselect, Wait Select, and Wait Reselect instructions. Bits 25 and 26 may be set individually or in combination:

	Bit 25	Bit 26
Direct	0	0
Table Indirect	0	1
Relative	1	0
Table Relative	1	1

**Direct**

Uses the device ID and physical address in the instruction.

Command	ID	Not Used	Not Used
Absolute Alternate Address			

**Table Indirect**

Uses the physical jump address, but fetches data using the table indirect method.

Command	Table Offset
Absolute Alternate Address	

**Relative**

Uses the device ID in the instruction, but treats the alternate address as a relative jump

Command	ID	Not Used	Not Used
Alternate Jump Offset			

**Table Relative**

Treats the alternate jump address as a relative jump and fetches the device ID, synchronous offset, and synchronous period indirectly. Adds the value in bits 23-0 of the first four bytes of the SCRIPTS instruction to the data structure base address to form the fetch address.

Command	Table Offset
Alternate Jump Offset	

**Bit 24 Select with ATN/**

This bit specifies whether SATN/ will be asserted during the selection phase when the SYM53C875 is executing a Select instruction. When operating in initiator mode, set this bit for the Select instruction. If this bit is set on any other I/O instruction, an illegal instruction interrupt is generated.

**Bit 23-20 Reserved**

**Bits 19-16 Encoded SCSI Destination ID**

This 4-bit field specifies the destination SCSI ID for an I/O instruction.

**Bits 15-11 Reserved**

**Bit 10 Set/Clear Carry**

This bit is used in conjunction with a Set or Clear instruction to set or clear the Carry bit. Setting this bit with a Set instruction asserts the Carry bit in the ALU. Setting this bit with a Clear instruction deasserts the Carry bit in the ALU.

**Bits 8-7 Reserved**

**Bit 9 Set/Clear Target Mode**

This bit is used in conjunction with a Set or Clear instruction to set or clear target mode. Setting this bit with a Set instruction configures the SYM53C875 as a target device (this sets bit 0 of the SCNTL0 register). Setting this bit with a Clear instruction configures the SYM53C875 as an initiator device (this clears bit 0 of the SCNTL0 register).

**Bit 6 Set/Clear SACK/**

**Bit 3 Set/Clear SATN/**

These two bits are used in conjunction with a Set or Clear instruction to assert or deassert the corresponding SCSI control signal. Bit 6 controls the SCSI SACK/ signal; bit 3 controls the SCSI SATN/ signal.

Setting either of these bits will set or reset the corresponding bit in the SOCL register, depending on the instruction used. The Set instruction is used to assert SACK/ and/or SATN/ on the SCSI bus. The Clear instruction is used to deassert SACK/ and/or SATN/ on the SCSI bus.

Since SACK/ and SATN/ are initiator signals, they will not be asserted on the SCSI bus unless the SYM53C875 is operating as an initiator or the SCSI Loopback Enable bit is set in the STEST2 register.

The Set/Clear SCSI ACK/ATN instruction would be used after message phase Block Move operations to give the initiator the opportunity to assert attention before acknowledging the last message byte. For example, if the initiator wishes to reject a message, an Assert SCSI ATN instruction would be issued before a Clear SCSI ACK instruction.

**Bits 2-0 Reserved**

---

**Second Dword**

**Bits 31-0 Start Address**

This 32-bit field contains the memory address to fetch the next instruction if the selection or reselection fails.

If relative or table relative addressing is used, this value is a 24-bit signed offset relative to the current DSP register value.

## Read/Write Instructions

The Read/Write instruction supports addition, subtraction, and comparison of two separate values within the chip. It performs the desired operation on the specified register and the SFBR register, then stores the result back to the specified register or the SFBR.

### First Dword

#### Bits 31-30 Instruction Type - Read/Write Instruction

The Read/Write instruction uses operator bits 26 through 24 in conjunction with the op code bits to determine which instruction is currently selected.

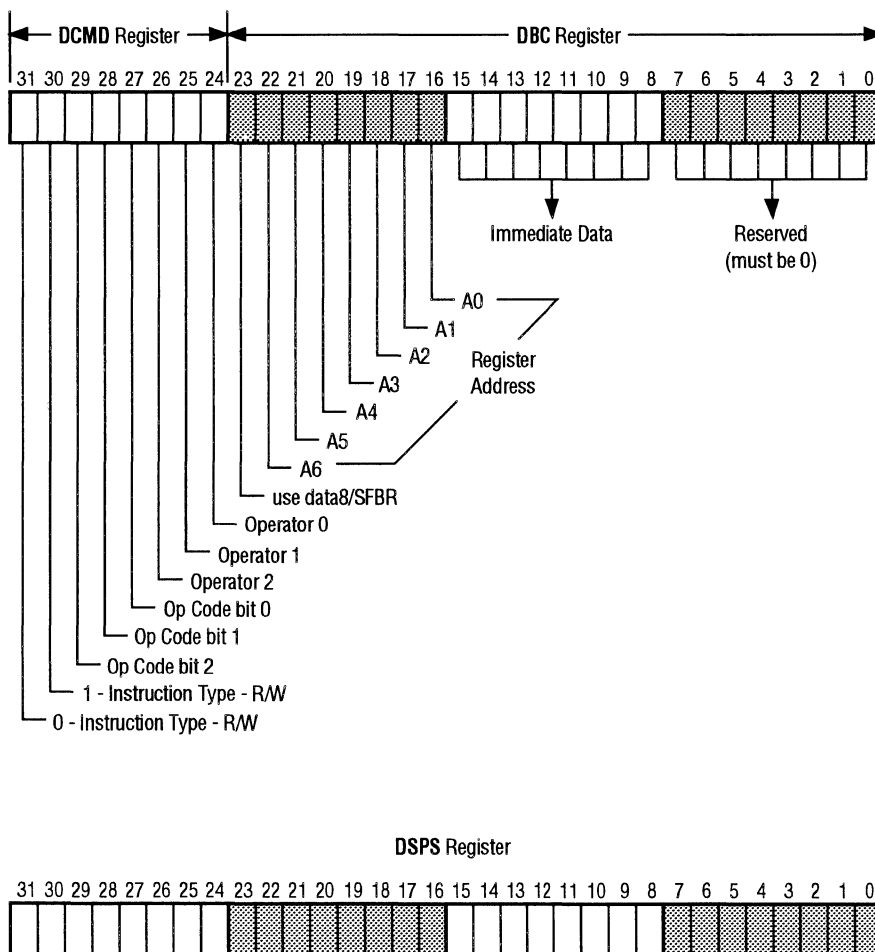


Figure 6-4: Read/Write Instruction Register

### Bits 29-27 Op Code

The combinations of these bits determine if the instruction is a Read/Write or an I/O instruction. Op codes 000 through 100 are considered I/O instructions.

### Bits 26-24 Operator

These bits are used in conjunction with the op code bits to determine which instruction is currently selected. Refer to table 6-1 for field definitions.

### Bit 23 Use data8/SFBR

When this bit is set, SFBR will be used instead of the data8 value during a Read-Modify-Write instruction (see Table 6-1). This allows the user to add two register values.

### Bits 22-16 Register Address - A(6-0)

Register values may be changed from SCRIPTS in read-modify-write cycles or move to/from SFBR cycles. A(6-0) select an 8-bit source/destination register within the SYM53C875.

### Bits 15-8 Immediate Data

This 8-bit value is used as a second operand in logical and arithmetic functions.

### Bits 7-0 Reserved

---

## Second Dword

### Bits 31-0 Destination Address

This field contains the 32-bit destination address where the data is to be moved.

---

## Read-Modify-Write Cycles

During these cycles the register is read, the selected operation is performed, and the result is written back to the source register.

The Add operation can be used to increment or decrement register values (or memory values if used in conjunction with a Memory-to-Register Move operation) for use as loop counters.

Subtraction is not available when SFBR is used instead of data8 in the instruction syntax. To subtract one value from another when using SFBR, first XOR the value to subtract (subtrahend) with 0xFF, and add 1 to the resulting value. This creates the 2's complement of the subtrahend. The two values can then be added to obtain the difference.

---

## Move to/from SFBR Cycles

All operations are read-modify-writes. However, two registers are involved, one of which is always the SFBR. The possible functions of this instruction are:

- Write one byte (value contained within the SCRIPTS instruction) into any chip register.
- Move to/from the SFBR from/to any other register.
- Alter the value of a register with AND/OR/ADD/XOR/SHIFT LEFT/SHIFT RIGHT operators.
- After moving values to the SFBR, the compare and jump, call, or similar instructions may be used to check the value.
- A Move-to-SFBR followed by a Move-from-SFBR can be used to perform a register to register move.

Table 6-1: Read/Write Instructions

Operator	Op Code 111 Read Modify Write	Op Code 110 Move to SFBR	Op Code 101 Move from SFBR
000	Move data into register. Syntax: "Move data8 to RegA"	Move data into SFBR register. Syntax: "Move data8 to SFBR"	Move data into register. Syntax: "Move data8 to RegA"
001*	Shift register one bit to the left and place the result in the same register. Syntax: "Move RegA SHL RegA"	Shift register one bit to the left and place the result in the SFBR register. Syntax: "Move RegA SHL SFBR"	Shift the SFBR register one bit to the left and place the result in the register. Syntax: "Move SFBR SHL RegA"
010	OR data with register and place the result in the same register. Syntax: "Move RegA   data8 to RegA"	OR data with register and place the result in the SFBR register. Syntax: "Move RegA   data8 to SFBR"	OR data with SFBR and place the result in the register. Syntax: "Move SFBR   data8 to RegA"
011	XOR data with register and place the result in the same register. Syntax: "Move RegA XOR data8 to RegA"	XOR data with register and place the result in the SFBR register. Syntax: "Move RegA XOR data8 to SFBR"	XOR data with SFBR and place the result in the register. Syntax: "Move SFBR XOR data8 to RegA"
100	AND data with register and place the result in the same register. Syntax: "Move RegA & data8 to RegA"	AND data with register and place the result in the SFBR register. Syntax: "Move RegA & data8 to SFBR"	AND data with SFBR and place the result in the register. Syntax: "Move SFBR & data8 to RegA"
101*	Shift register one bit to the right and place the result in the same register. Syntax: "Move RegA SHR RegA"	Shift register one bit to the right and place the result in the SFBR register. Syntax: "Move RegA SHR SFBR"	Shift the SFBR register one bit to the right and place the result in the register. Syntax: "Move SFBR SHR RegA"
110	Add data to register without carry and place the result in the same register. Syntax: "Move RegA + data8 to RegA"	Add data to register without carry and place the result in the SFBR register. Syntax: "Move RegA + data8 to SFBR"	Add data to SFBR without carry and place the result in the register. Syntax: "Move SFBR + data8 to RegA"
111	Add data to register with carry and place the result in the same register. Syntax: "Move RegA + data8 to RegA with carry"	Add data to register with carry and place the result in the SFBR register. Syntax: "Move RegA + data8 to SFBR with carry"	Add data to SFBR with carry and place the result in the register. Syntax: "Move SFBR + data8 to RegA with carry"

Notes:

1. Substitute the desired register name or address for "RegA" in the syntax examples

2. data8 indicates eight bits of data

3. Use SFBR instead of data8 to add two register values.

\* Data is shifted through the Carry bit and the Carry bit is shifted into the data byte

# Transfer Control Instructions

## First Dword

### Bits 31-30 Instruction Type - Transfer Control Instruction

#### Bits 29-27 Op Code

This 3-bit field specifies the type of transfer control instruction to be executed. All transfer control instructions can be conditional. They can be dependent on a true/false comparison of the ALU Carry bit or a comparison of the SCSI information transfer phase with the Phase field, and/or a comparison of the First Byte Received with the Data Compare field. Each instruction can operate in initiator or target mode.

OPC2	OPC1	OPC0	Instruction Defined
0	0	0	Jump
0	0	1	Call
0	1	0	Return
0	1	1	Interrupt
1	X	X	Reserved

## Jump Instruction

1. The SYM53C875 can do a true/false comparison of the ALU carry bit, or compare the phase and/or data as defined by the Phase Compare, Data Compare and True/False bit fields. If the comparisons are true, the SYM53C875 loads the DSP register with the contents of the DSPTS register. The DSP register now contains the address of the next instruction.

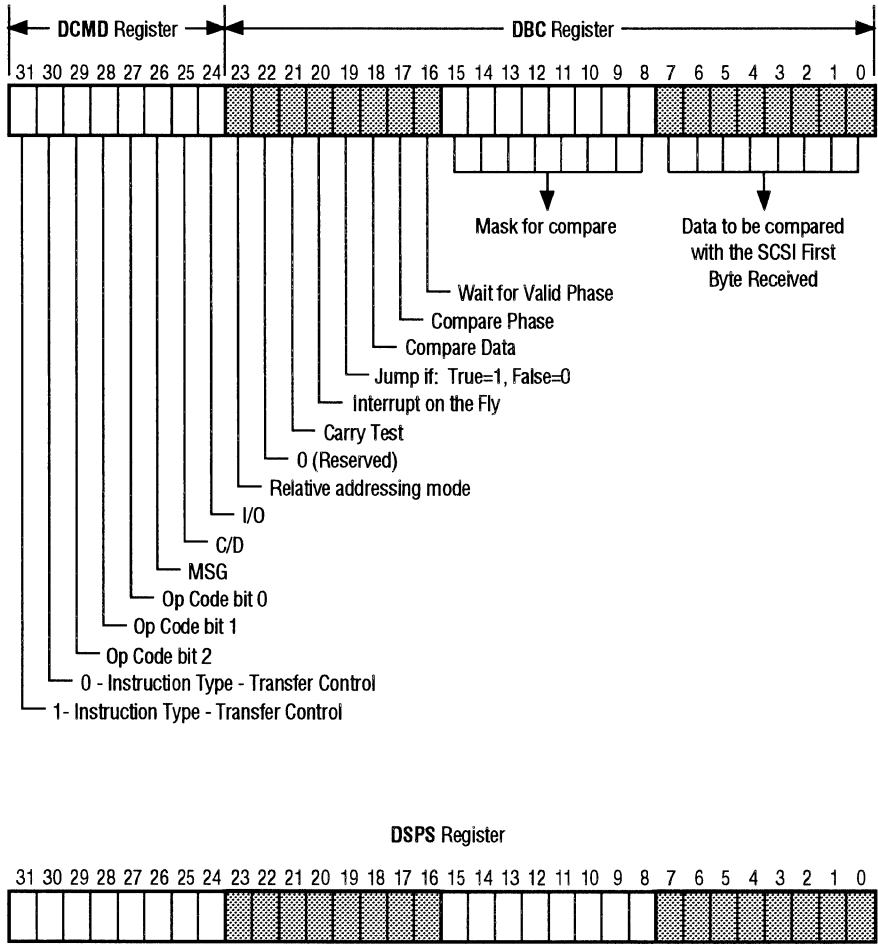
2. If the comparisons are false, the SYM53C875 fetches the next instruction from the address pointed to by the DSP register, leaving the instruction pointer unchanged.

## Call Instruction

1. The SYM53C875 can do a true/false comparison of the ALU carry bit, or compare the phase and/or data as defined by the Phase Compare, Data Compare, and True/False bit fields. If the comparisons are true, the SYM53C875 loads the DSP register with the contents of the DSPTS register and that address value becomes the address of the next instruction.

When the SYM53C875 executes a Call instruction, the instruction pointer contained in the DSP register is stored in the TEMP register. Since the TEMP register is not a stack and can only hold one longword, nested call instructions are not allowed.

2. If the comparisons are false, the SYM53C875 fetches the next instruction from the address pointed to by the DSP register and the instruction pointer is not modified.



**Figure 6-5: Transfer Control Instruction**

**Return Instruction**

1. The SYM53C875 can do a true/false comparison of the ALU carry bit, or compare the phase and/or data as defined by the Phase Compare, Data Compare, and True/False bit fields. If the comparisons are true, then the SYM53C875 loads the DSP register with the contents of the DSPS register. That address value becomes the address of the next instruction.

When a Return instruction is executed, the value stored in the TEMP register is returned to the DSP register. The SYM53C875 does not check to see whether the Call instruction has already been executed. It will not generate an interrupt if a Return instruction is executed without previously executing a Call instruction.

2. If the comparisons are false, then the SYM53C875 fetches the next instruction from the address pointed to by the DSP register and the instruction pointer will not be modified.

**Interrupt Instructions**

*Interrupt*

- a) The SYM53C875 can do a true/false comparison of the ALU carry bit, or compare the phase and/or data as defined by the Phase Compare, Data Compare, and True/False bit fields. If the comparisons are true, then the SYM53C875 generates an interrupt by asserting the IRQ/ signal.
- b) The 32-bit address field stored in the DSPS register (not DNAD as in 53C700) can contain a unique interrupt service vector. When servicing the interrupt, this unique status code allows the ISR to quickly identify the point at which the interrupt occurred.
- c) The SYM53C875 halts and the DSP register must be written to start any further operation.

*Interrupt on-the-Fly*

- a) The SYM53C875 can do a true/false comparison of the ALU carry bit or compare the phase and/or data as defined by the Phase Compare, Data Compare, and True/False bit fields. If the comparisons are true, and the Interrupt on the Fly bit is set (bit 20), the SYM53C875 will assert the Interrupt on the Fly bit (ISTAT bit 2).

**Bits 26-24 SCSI Phase**

This 3-bit field corresponds to the three SCSI bus phase signals which are compared with the phase lines latched when SREQ/ is asserted. Comparisons can be performed to determine the SCSI phase actually being driven on the SCSI bus. The following table describes the possible combinations and their corresponding SCSI phase. These bits are only valid when the SYM53C875 is operating in initiator mode; when the SYM53C875 is operating in the target mode, these bits should be cleared.

MSG	C/D	I/O	SCSI Phase
0	0	0	Data out
0	0	1	Data in
0	1	0	Command
0	1	1	Status
1	0	0	Reserved out
1	0	1	Reserved in
1	1	0	Message out
1	1	1	Message in

**Bit 23 Relative Addressing Mode**

When this bit is set, the 24-bit signed value in the DSPS register is used as a relative offset from the current DSP address (which is pointing to the next instruction, not the one currently executing). Relative mode does not apply to Return and Interrupt SCRIPTS.



### Jump/Call an Absolute Address

Start execution at the new absolute address.

Command	Condition Codes
Absolute Alternate Address	

### Jump/Call a Relative Address

Start execution at the current address plus (or minus) the relative offset.

Command	Condition Codes
Don't Care	Alternate Jump Offset

The SCRIPTS program counter is a 32-bit value pointing to the SCRIPT currently being executed by the SYM53C875. The next address is formed by adding the 32-bit program counter to the 24-bit signed value of the last 24 bits of the Jump or Call instruction. Because it is signed (twos complement), the jump can be forward or backward.

A relative transfer can be to any address within a 16-MB segment. The program counter is combined with the 24-bit signed offset (using addition or subtraction) to form the new execution address.

SCRIPTS programs may contain a mixture of direct jumps and relative jumps to provide maximum versatility when writing SCRIPTS. For example, major sections of code can be accessed with far calls using the 32-bit physical address, then local labels can be called using relative transfers. If a SCRIPT is written using only relative transfers it would not require any run time alteration of physical addresses, and could be stored in and executed from a PROM.

### Bit 21 Carry Test

When this bit is set, decisions based on the ALU carry bit can be made. True/False comparisons are legal, but Data Compare and Phase Compare are illegal.

### Bit 20 Interrupt on the Fly

When this bit is set, the interrupt instruction will not halt the SCRIPTS processor. Once the interrupt occurs, the Interrupt on the Fly bit (ISTAT bit 2) will be asserted.

### Bit 19 Jump If True/False

This bit determines whether the SYM53C875 should branch when a comparison is true or when a comparison is false. This bit applies to phase compares, data compares, and carry tests. If both the Phase Compare and Data Compare bits are set, then both compares must be true to branch on a true condition. Both compares must be false to branch on a false condition.

Bit 19	Result of Compare	Action
0	False	Jump Taken
0	True	No Jump
1	False	No Jump
1	True	Jump Taken

### Bit 18 Compare Data

When this bit is set, the first byte received from the SCSI data bus (contained in SFBR register) is compared with the Data to be Compared Field in the Transfer Control instruction. The Wait for Valid Phase bit controls when this compare will occur. The Jump if True/False bit determines the condition (true or false) to branch on.

### Bit 17 Compare Phase

When the SYM53C875 is in initiator mode, this bit controls phase compare operations. When this bit is set, the SCSI phase signals (latched by SREQ/) are compared to the Phase Field in the Transfer Control instruction; if they match, the comparison is true. The Wait for Valid Phase bit controls when the compare will occur. When the SYM53C875 is operating in target mode this bit, when set, tests for an active SCSI SATN/ signal.

### Bit 16 Wait For Valid Phase

If the Wait for Valid Phase bit is set, the SYM53C875 waits for a previously unserved phase before comparing the SCSI phase and data.

If the Wait for Valid Phase bit is clear, the SYM53C875 compares the SCSI phase and data immediately.

### Bits 15-8 Data Compare Mask

The Data Compare Mask allows a SCRIPT to test certain bits within a data byte. During the data compare, any mask bits that are set cause the corresponding bit in the SFBR data byte to be ignored. For instance, a mask of 01111111b and data compare value of 1XXXXXXXXb allows the SCRIPTS processor to determine whether or not the high order bit is set while ignoring the remaining bits.

### Bits 7-0 Data Compare Value

This 8-bit field is the data to be compared against the SCSI First Byte Received (SFBR) register. These bits are used in conjunction with the Data Compare Mask Field to test for a particular data value.

---

## Second Dword

### Bits 31-0 Jump Address

This 32-bit field contains the address of the next instruction to fetch when a jump is taken. Once the SYM53C875 has fetched the instruction from the address pointed to by these 32 bits, this address is incremented by 4, loaded into the DSP register and becomes the current instruction pointer.

---

## Memory Move Instructions

For Memory Move instructions, bits 5 and 4 (SIOM and DIOM) in the DMODE register determine whether the source or destination addresses reside in memory or I/O space. By setting these bits appropriately, data may be moved within memory space, within I/O space, or between the two address spaces.

The Memory Move instruction is used to copy the specified number of bytes from the source address to the destination address.

Allowing the SYM53C875 to perform memory moves frees the system processor for other tasks and moves data at higher speeds than available from current DMA controllers. Up to 16 MB may be transferred with one instruction. There are two restrictions:

1. Both the source and destination addresses must start with the same address alignment (A(1-0) must be the same). If source and destination are not aligned, then an illegal instruction interrupt will occur. For the PCI Cache Line Size register setting to take effect, the source and destination must be the same distance from a cache line boundary.
2. Indirect addresses are not allowed. A burst of data is fetched from the source address, put into the DMA FIFO and then written out to the destination address. The move continues until the byte count decrements to zero, then another SCRIPT is fetched from system memory.

The DSPS and DSA registers are additional holding registers used during the Memory Move; however, the contents of the DSA register are preserved.

### Bits 31-39 Instruction Type—Memory Move

### Bits 28-25 Reserved

These bits are reserved and must be zero. If any of these bits is set, an illegal instruction interrupt will occur.

#### **Bit 24 No Flush**

Note: this bit has no effect unless the Pre-fetch Enable bit in the DCNTL register is set. For information on SCRIPTS instruction prefetching, see Chapter 2.

When this bit is set, the SYM53C875 performs a Memory Move without flushing the prefetch unit. When this bit is clear, the Memory Move instruction automatically flushes the prefetch unit. The No Flush option should be used if the source and destination are not within four instructions of the current Memory Move instruction.

#### **Bits 23-0 Transfer Count**

The number of bytes to be transferred is stored in the lower 24 bits of the first instruction word.

---

### **Read/Write System Memory from a Script**

By using the Memory Move instruction, single or multiple register values may be transferred to or from system memory.

Because the SYM53C875 will respond to addresses as defined in the Base I/O or Base Memory registers, it could be accessed during a Memory Move operation if the source or destination address decodes to within the chip's register space. If this occurs, the register indicated by the lower seven bits of the address is taken to be the data source or destination. In this way, register values can be saved to system memory and later restored, and SCRIPTS can make decisions based on data values in system memory.

The SFBR is not writable via the CPU, and therefore not by a Memory Move. However, it can be loaded via SCRIPTS Read/Write operations. To load the SFBR with a byte stored in system memory, the byte must first be moved to an intermediate SYM53C875 register (for example, a SCRATCH register), and then to the SFBR.

The same address alignment restrictions apply to register access operations as to normal memory-to-memory transfers.

---

### **Second Dword**

#### **Bits 31-0, DSPS Register**

These bits contain the source address of the Memory Move.

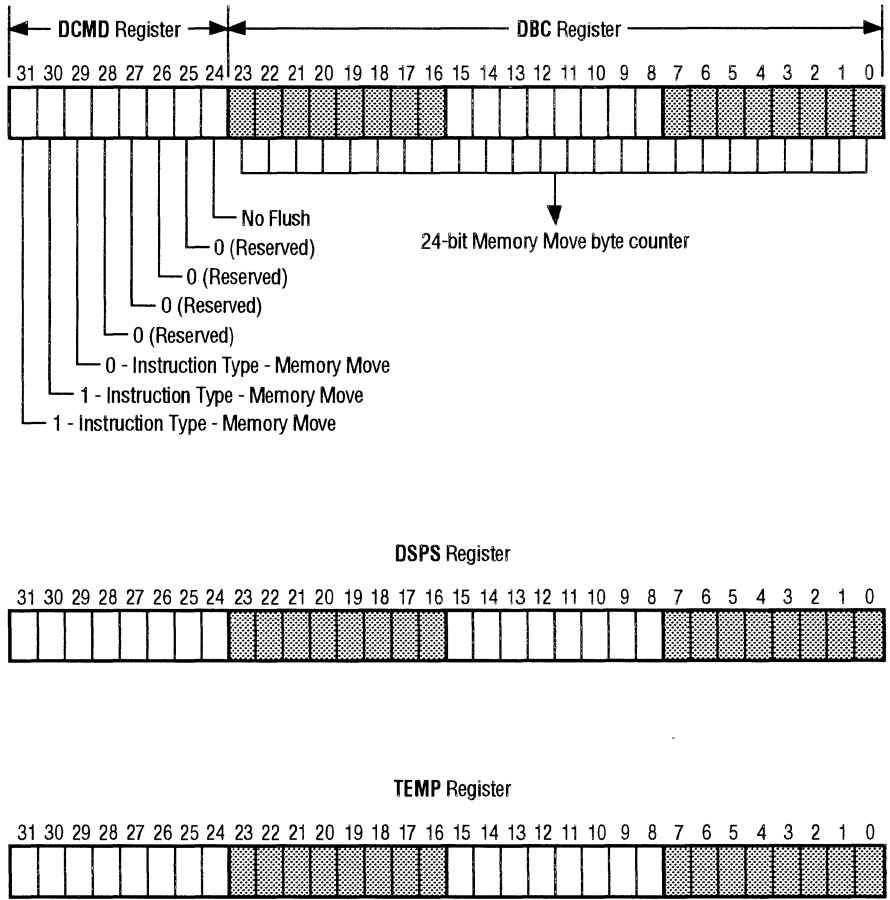
---

### **Third Dword**

#### **Bits 31-0, TEMP Register**

These bits contain the destination address for the Memory Move.

Instruction Set of the I/O Processor  
**Memory Move Instructions**



**Figure 6-6: Memory Move Instructions**

## Load and Store Instructions

The Load and Store instruction provides a more efficient way to move data from/to memory to/from an internal register in the chip without using the normal memory move instruction.

The load and store instructions are represented by two-dword op codes. The first dword contains the DCMD and DBC register values. The second dword contains the DSPS value. This is either the actual memory location of where to load or store, or the offset from the DSA, depending on the value of Bit 28 (DSA Relative).

A maximum of 4 bytes may be moved with these instructions. The register address and memory address must have the same byte alignment, and the count set such that it does not cross dword boundaries. The destination memory address in the Store instruction and the source memory address of the Load instruction may not map back to the operating register set of the chip. This excludes the SCRIPTS RAM and ROM memory spaces. If it does, a PCI read/write cycle will occur (the data does not actually transfer to/from the chip), and the chip will issue an interrupt (Illegal Instruction Detected) immediately following.

Bits A1, A0	Number of bytes allowed to load/store
00	One, two, three or four
01	One, two, or three
10	One or two
11	One

The SIOM and DIOM bits in the DMODE register determine whether the destination or source address of the instruction is in Memory space or I/O space, as illustrated in the following table. The Load/Store utilizes the PCI commands for I/O READ and I/O WRITE to access the I/O space.

Bit	Source	Destination
SIOM (Load)	memory	register
DIOM (Store)	register	memory

### First Dword

#### Bit 31-29, Instruction Type

These bits should be 111, indicating the Load and Store instruction.

#### Bit 28, DSA Relative

When this bit is clear, the value in the DSPS is the actual 32-bit memory address to perform the load/store to/from. When this bit is set, the chip determines the memory address to perform the load/store to/from by adding the 24-bit signed offset value in the DSPS to the DSA.

#### Bits 27-26, Reserved

#### Bit 25, No Flush (Store instruction only)

Note: this bit has no effect unless the Pre-fetch Enable bit in the DCNTL register is set. For information on SCRIPTS instruction prefetching, see Chapter 2.

When this bit is set, the SYM53C875 performs a Store without flushing the prefetch unit. When this bit is clear, the Store instruction automatically flushes the prefetch unit. No Flush should be used if the source and destination are not within four instructions of the current Store instruction. This bit has no effect on the Load instruction.

#### Bit 24, Load/Store

When this bit is set, the instruction is a Load. When cleared, it is a Store.

#### Bit 23, Reserved

#### Bits 22-16, Register Address

A6-A0 select the register to load/store to/from within the SYM53C875.

Note: it is not possible to Load the SFBR register, although it is possible to store the SFBR contents to another location.

#### Bits 15-3, Reserved

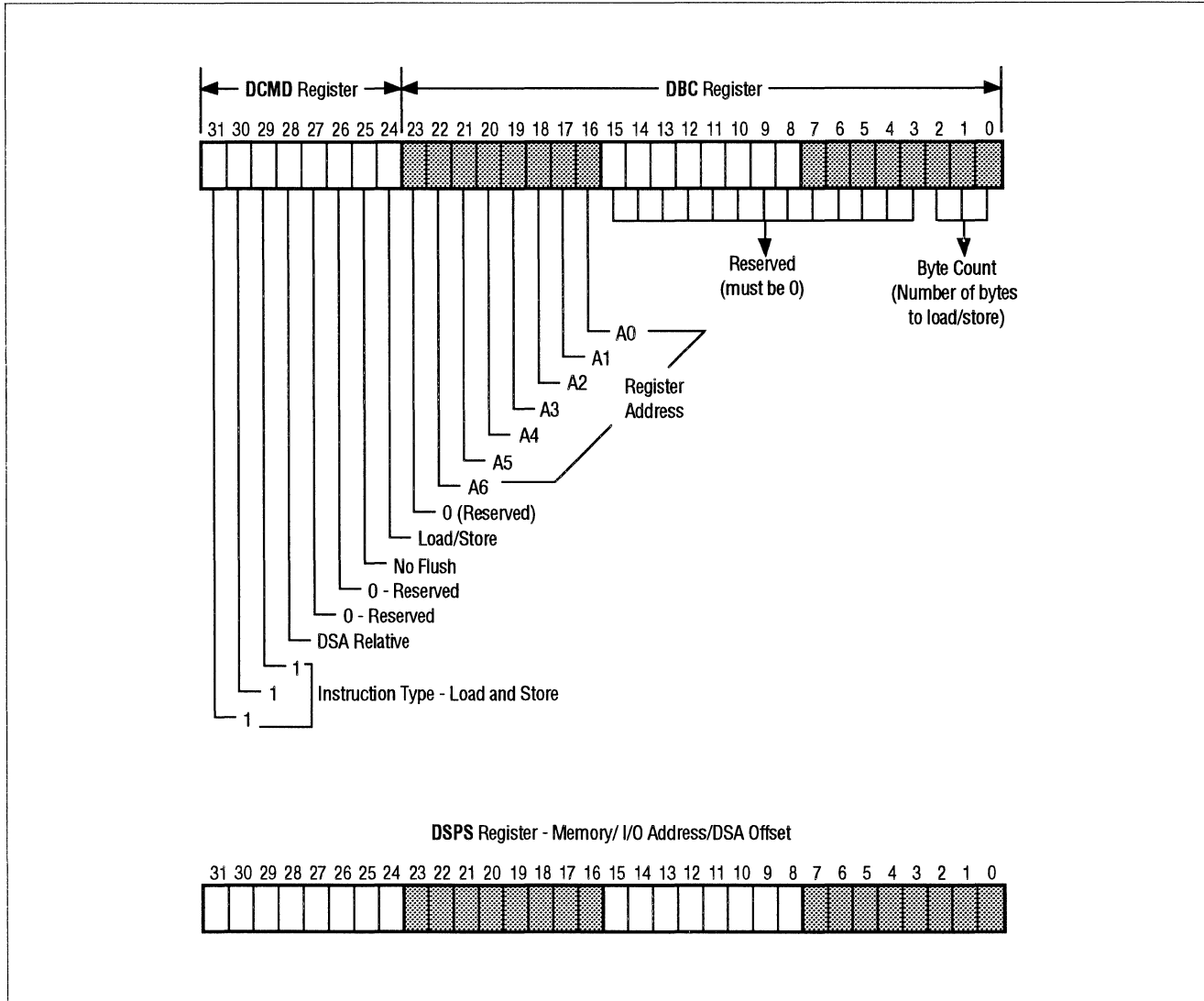
#### Bits 2-0, Byte Count

This value is the number of bytes to load/store.

**Second Dword**

**Bits 31-0, Memory/IO Address / DSA Offset**

This is the actual memory location of where to load or store, or the offset from the DSA register value.



**Figure 6-7: Load and Store Instruction Format**

## Chapter 7 Electrical Characteristics

### DC Characteristics

**Table 7-1: Absolute Maximum Stress Ratings**

Symbol	Parameter	Min	Max	Unit	Test Conditions
$T_{STG}$	Storage temperature	-55	150	°C	-
$V_{DD}$	Supply voltage	-0.5	7.0	V	-
$V_{IN}$	Input Voltage	$V_{SS} - 0.5$	$V_{DD} + 0.5$	V	-
$I_{LP}^*$	Latch-up current	$\pm 150$	-	mA	-
ESD**	Electrostatic discharge	-	2K	V	MIL-STD 883C, Method 3015.7

*Stresses beyond those listed above may cause permanent damage to the device. These are stress ratings only; functional operation of the device at these or any other conditions beyond those indicated in the Operating Conditions section of the manual is not implied.*

\*  $-2V < V_{PIN} < 8V$

\*\* SCSI pins only

**Table 7-2: Operating Conditions**

Symbol	Parameter	Min	Max	Unit	Test Conditions
$V_{DD}$	Supply voltage	4.75	5.25	V	-
$I_{DD}$	Supply current (dynamic)	-	130	mA	-
	Supply current (static)	-	1	mA	-
$T_A$	Operating free air	0	70	°C	-
$\theta_{JA}$	Thermal resistance (junction to ambient air)	-	67	°C/W	-

*Conditions that exceed the operating limits may cause the device to function incorrectly*

**Table 7-3: SCSI Signals - SD(15-0)/, SDP(1-0)/, SREQ/ SACK/**

Symbol	Parameter	Min	Max	Unit	Test Conditions
V <sub>IH</sub>	Input high voltage	2.0	V <sub>DD</sub> + 0.5	V	-
V <sub>IL</sub>	Input low voltage	V <sub>SS</sub> - 0.5	0.8	V	-
V <sub>OH</sub> *	Output high voltage	2.5	3.5	V	2.5 mA
V <sub>OL</sub>	Output low voltage	V <sub>SS</sub>	0.5	V	48 mA
I <sub>OZ</sub>	Tristate leakage	-10	10	μA	-

\*TolerANT active negation enabled

**Table 7-4: SCSI Signals - SMSG, SI\_O/, SC\_D/, SATN/, SBSY/, SSEL/, SRST/**

Symbol	Parameter	Min	Max	Unit	Test Conditions
V <sub>IH</sub>	Input high voltage	2.0	V <sub>DD</sub> + 0.5	V	-
V <sub>IL</sub>	Input low voltage	V <sub>SS</sub> - 0.5	0.8	V	-
V <sub>OL</sub>	Output low voltage	V <sub>SS</sub>	0.5	V	48 mA
I <sub>OZ</sub>	Tristate leakage (SRST/ only)	-10 -500	10 -50	μA	-

**Table 7-5: Input Signals - CLK, SCLK, GNT/, IDSEL, RST/, TESTIN, DIFFSENS, BIG\_LIT/**

Symbol	Parameter	Min	Max	Unit	Test Conditions
V <sub>IH</sub>	Input high voltage	2.0	V <sub>DD</sub> + 0.5	V	-
V <sub>IL</sub>	Input low voltage	V <sub>SS</sub> - 0.5	0.8	V	-
I <sub>IN</sub>	Input leakage	-10	10	μA	-

Note: CLK, SCLK, and BIG\_LIT/ have 100 μA pull-ups, and GNT/ and IDSEL have 25 μA pull-ups, that are enabled when TESTIN is low. TESTIN has a 100 μA pull-up that is always enabled.



Table 7-6: Capacitance

Symbol	Parameter	Min	Max	Unit	Test Conditions
C <sub>I</sub>	Input capacitance of input pads	-	7	pF	-
C <sub>IO</sub>	Input capacitance of I/O pads	-	10	pF	-

Table 7-7: Output Signal - MAC/\_TESTOUT, REQ/

Symbol	Parameter	Min	Max	Unit	Test Conditions
V <sub>OH</sub>	Output high voltage	2.4	V <sub>DD</sub>	V	-16 mA
V <sub>OL</sub>	Output low voltage	V <sub>SS</sub>	0.4	V	16 mA
I <sub>OZ</sub>	Tristate leakage	-10	10	μA	-

Note: REQ/ has a 100 μA pull-up that is enabled when TESTIN is low

Table 7-8: Output Signals - IRQ/, SDIR(15-0), SDIRP0, SDIRP1, BSYDIR, SELDIR, RSTDIR, TGS, IGS, MAS/(1-0), MCE/, MOE/, MWE/

Symbol	Parameter	Min	Max	Unit	Test Conditions
V <sub>OH</sub>	Output high voltage	2.4	V <sub>DD</sub>	V	-4 mA*
V <sub>OL</sub>	Output low voltage	V <sub>SS</sub>	0.4	V	4 mA*
I <sub>OZ</sub>	Tristate leakage	-10	10	μA	-

Note: IRQ/, MAS/(1-0), MCE/, MOE/, and MWE/ have a 100 μA pull-up that is enabled when TESTIN is low. IRQ/ can be enabled with a register bit as an open drain output with an internal 100 μA pull-up.

\*for IRQ/, Test Conditions are ± 8mA

Table 7-9: Output Signal - SERR/

Symbol	Parameter	Min	Max	Unit	Test Conditions
V <sub>OL</sub>	Output low voltage	V <sub>SS</sub>	0.4	V	16 mA
I <sub>OZ</sub>	Tristate leakage	-10	10	μA	-

**Table 7-10: Bidirectional Signals - AD(31-0), C\_BE/(3-0), FRAME/, IRDY/, TRDY/, DEVSEL/, STOP/, PERR/, PAR, BYTEPAR3-0 (53C875N only)**

Symbol	Parameter	Min	Max	Unit	Test Conditions
V <sub>IH</sub>	Input high voltage	2.0	V <sub>DD</sub> + 0.5	V	-
V <sub>IL</sub>	Input low voltage	V <sub>SS</sub> - 0.5	0.8	V	-
V <sub>OH</sub>	Output high voltage	2.4	V <sub>DD</sub>	V	16 mA
V <sub>OL</sub>	Output low voltage	V <sub>SS</sub>	0.4	V	16 mA
I <sub>OZ</sub>	Tristate leakage	-10	10	μA	-

*Note: All the signals in this table have 25 μA pull-ups that are enabled when TESTIN is low*

**Table 7-11: Bidirectional Signals - GPIO0\_FETCH/, GPIO1\_MASTER/, GPIO2\_MAS2/, GPIO3, GPIO4**

Symbol	Parameter	Min	Max	Unit	Test Conditions
V <sub>IH</sub>	Input high voltage	2.0	V <sub>DD</sub> + 0.5	V	-
V <sub>IL</sub>	Input low voltage	V <sub>SS</sub> - 0.5	0.8	V	-
V <sub>OH</sub>	Output high voltage	2.4	V <sub>DD</sub>	V	-16 mA
V <sub>OL</sub>	Output low voltage	V <sub>SS</sub>	0.4	V	16 mA
I <sub>OZ</sub>	Tristate leakage	-10	10	μA	-

*Note: All the signals in this table have 100 μA pull-ups that are enabled when TESTIN is low*

**Table 7-12: Bidirectional Signals - MAD(7-0)**

Symbol	Parameter	Min	Max	Unit	Test Conditions
V <sub>IH</sub>	Input high voltage	2.0	V <sub>DD</sub> + 0.5	V	-
V <sub>IH</sub>	Input high voltage - external memory pull-downs	3.85	V <sub>DD</sub> + 0.5	V	-
V <sub>IL</sub>	Input low voltage	V <sub>SS</sub> - 0.5	0.8	V	-
V <sub>IL</sub>	Input low voltage - external memory pull-downs	V <sub>SS</sub> - 0.5	1.35	V	-
V <sub>OH</sub>	Output high voltage	2.4	V <sub>DD</sub>	V	-4 mA
V <sub>OL</sub>	Output low voltage	V <sub>SS</sub>	0.4	V	4 mA
I <sub>OZ</sub>	Tristate leakage	-10	10	μA	-

*Note: All the signals in this table have 100 μA pull-ups that are enabled when TESTIN is low*

**Table 7-13: Input Signals —TDI, TMS, TCK (53C875J, 53C875JB, 53C875N only)**

Symbol	Parameter	Min	Max	Unit	Test Conditions
V <sub>IH</sub>	Input high voltage	3.85	V <sub>DD</sub> + 0.5	V	-
V <sub>IL</sub>	Input low voltage	V <sub>SS</sub> - 0.5	1.35	V	-
I <sub>IN</sub>	Input leakage	-800	-200	μA	-

**Table 7-14: Output Signal —TDO (53C875, 53C875JB, 53C875N only)**

Symbol	Parameters	Min	Max	Units	Test Conditions
V <sub>OH</sub>	Output high voltage	V <sub>DD</sub> - 0.5	V <sub>DD</sub>	V	-4 mA
V <sub>OL</sub>	Output low voltage	V <sub>SS</sub>	0.5	V	4 mA
I <sub>OZ</sub>	Tristate leakage	-10	10	μA	-

## TolerANT Technology Electrical Characteristics

Table 7-15: TolerANT Technology Electrical Characteristics

Symbol	Parameter	Min	Max	Units	Test Conditions
$V_{OH}^1$	Output high voltage	2.5	3.5	V	$I_{OH} = 2.5 \text{ mA}$
$V_{OL}$	Output low voltage	0.1	0.5	V	$I_{OL} = 48 \text{ mA}$
$V_{IH}$	Input high voltage	2.0	7.0	V	-
$V_{IL}$	Input low voltage	-0.5	0.8	V	Referenced to $V_{SS}$
$V_{IK}$	Input clamp voltage	-0.66	-0.77	V	$V_{DD} = 4.75$ ; $I_I = -20 \text{ mA}$
$V_{TH}$	Threshold, high to low	1.1	1.3	V	-
$V_{TL}$	Threshold, low to high	1.5	1.7	V	-
$V_{TH} - V_{TL}$	Hysteresis	200	400	mV	-
$I_{OH}^1$	Output high current	2.5	24	mA	$V_{OH} = 2.5 \text{ V}$
$I_{OL}$	Output low current	100	200	mA	$V_{OL} = 0.5 \text{ V}$
$I_{OSH}^1$	Short-circuit output high current	-	625	mA	Output driving low, pin shorted to $V_{DD}$ supply <sup>2</sup>
$I_{OSL}$	Short-circuit output low current	-	95	mA	Output driving high, pin shorted to $V_{SS}$ supply
$I_{LH}$	Input high leakage	-	10	$\mu\text{A}$	$-0.5 < V_{DD} < 5.25$ $V_{PIN} = 2.7 \text{ V}$
$I_{LL}$	Input low leakage	-	-10	$\mu\text{A}$	$-0.5 < V_{DD} < 5.25$ $V_{PIN} = 0.5 \text{ V}$
$R_I$	Input resistance	20	-	$\text{M}\Omega$	SCSI pins <sup>3</sup>
$C_P$	Capacitance per pin	-	10	pF	PQFP
$t_R^1$	Rise time, 10% to 90%	9.7	18.5	ns	Figure 7-1
$t_F$	Fall time, 90% to 10%	5.2	14.7	ns	Figure 7-1

Note: These values are guaranteed by periodic characterization; they are not 100% tested on every device.

<sup>1</sup> Active negation outputs only: Data, Parity, SREQ, SACK

<sup>2</sup> Single pin only; irreversible damage may occur if sustained for one second

<sup>3</sup> SCSI RESET pin has 10 k $\Omega$  pull-up resistor

**Table 7-15: TolerANT Technology Electrical Characteristics (Continued)**

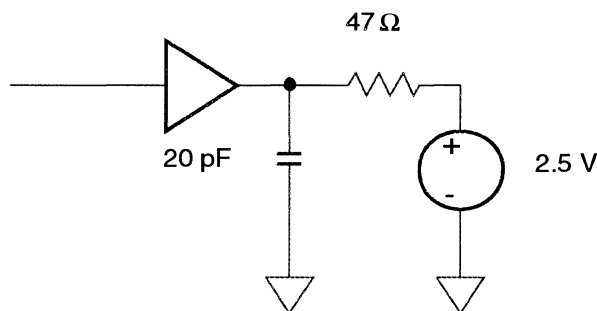
Symbol	Parameter	Min	Max	Units	Test Conditions
$dV_H/dt$	Slew rate, low to high	0.15	0.49	V/ns	Figure 7-1
$dV_L/dt$	Slew rate, high to low	0.19	0.67	V/ns	Figure 7-1
ESD	Electrostatic discharge	2	-	KV	MIL-STD-883C; 3015-7
	Latch-up	100	-	mA	-
	Filter delay	20	30	ns	Figure 7-2
	Extended filter delay	40	60	ns	Figure 7-2

Note: These values are guaranteed by periodic characterization; they are not 100% tested on every device.

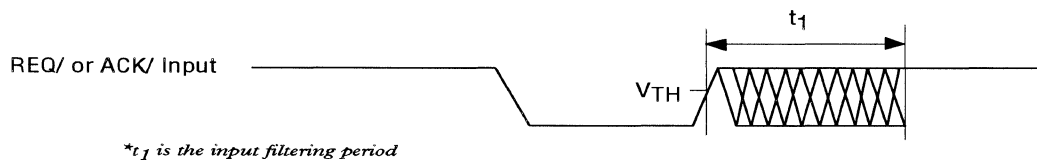
<sup>1</sup> Active negation outputs only: Data, Parity, SREQ/, SACK/

<sup>2</sup> Single pin only; irreversible damage may occur if sustained for one second

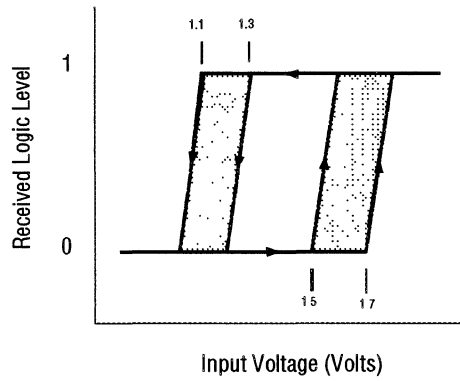
<sup>3</sup> SCSI RESET pin has 10 kΩ pull-up resistor



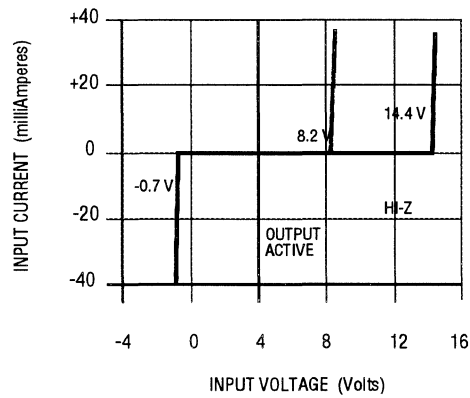
**Figure 7-1: Rise and Fall Time Test Conditions**



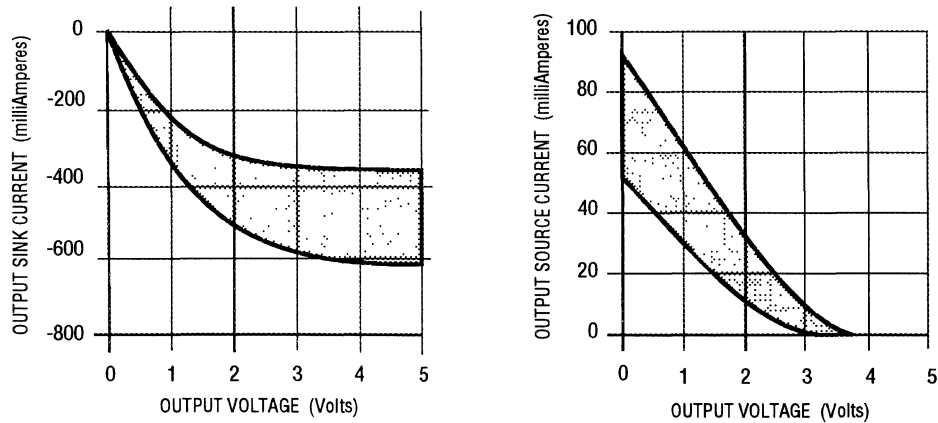
**Figure 7-2: SCSI Input Filtering**



**Figure 7-3: Hysteresis of SCSI Receiver**



**Figure 7-4: Input Current as a Function of Input Voltage**



**Figure 7-5: Output Current as a Function of Output Voltage**

## AC Characteristics

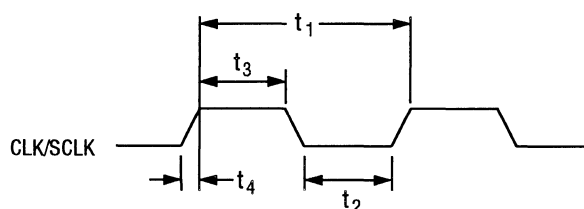
The AC characteristics described in this section apply over the entire range of operating conditions (refer to the DC Characteristics section). Chip timings are based on simulation at worst case voltage, temperature, and processing. Timings were developed with a load capacitance of 50 pF.

Figure 7-6: Clock Timing

Symbol	Parameter	Min	Max	Units
$t_1$	Bus clock cycle time	30	DC	ns
	SCSI clock cycle time (SCLK)*	12.5	60	ns
$t_2$	CLK low time**	12	-	ns
	SCLK low time**	5	-	ns
$t_3$	CLK high time**	12	-	ns
	SCLK high time**	5	-	ns
$t_4$	CLK slew rate	1	-	V/ns
	SCLK slew rate	1	-	V/ns

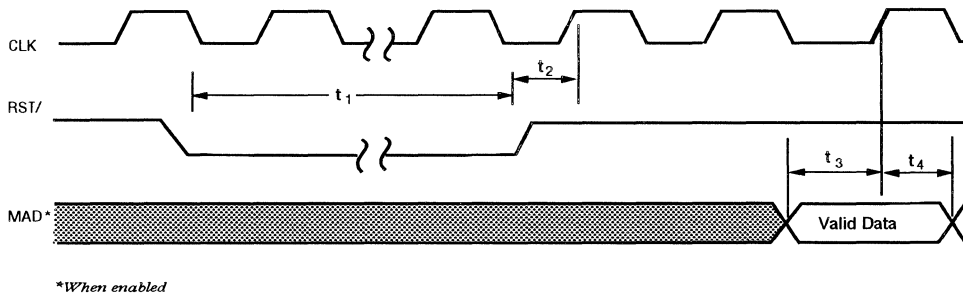
\*This parameter must be met to insure SCSI timings are within specification

\*\*Duty cycle not to exceed 60/40



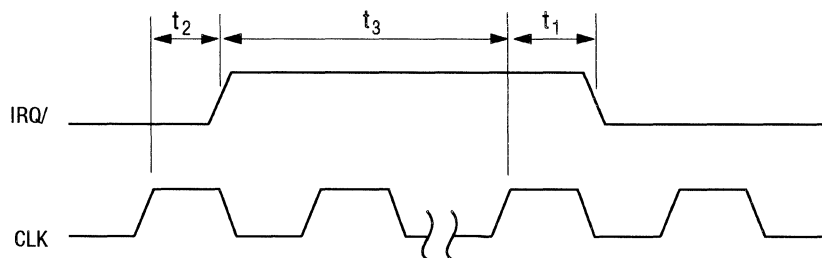
**Figure 7-7: Reset Input**

Symbol	Parameter	Min	Max	Units
$t_1$	Reset pulse width	10	-	$t_{CLK}$
$t_2$	Reset deasserted setup to CLK high	0	-	ns
$t_3$	MAD setup time to CLK high (for configuring the MAD bus only)	20	-	ns
$t_4$	MAD hold time from CLK high (for configuring the MAD bus only)	20	-	ns



**Figure 7-8: Interrupt Output**

Symbol	Parameter	Min	Max	Units
$t_1$	CLK high to IRQ/ low	20	-	ns
$t_2$	CLK high to IRQ/ high	40	-	ns
$t_3$	IRQ/ deassertion time	3	-	CLKs





## PCI and External Memory Interface Timing Diagrams

Figure 7-9 through Figure 7-30 represent signal activity when the SYM53C875 accesses the PCI bus. The timings for the PCI and external memory buses are listed on page 7-34. This section includes timing diagrams for access to three groups of external memory configurations. The first group applies to systems with memory size of 128 KB and above; one byte read or write cycle, and fast or normal ROMs. The second group applies to systems with memory size of 128 KB and above, one-byte read or write cycles, and slow ROMs. The third group applies to systems with memory size of 64 KB or less, one-byte read or write cycles, and normal or fast ROM.

Note: multiple byte accesses to the external memory bus increase the read or write cycle by 11 clocks for each additional byte.

### Timing diagrams included in this section

#### Target Cycles

- PCI configuration register read
- PCI configuration register write
- Operating Register/SCRIPTS RAM read
- Operating Register/SCRIPTS RAM write
- External Memory Read
- External Memory Write

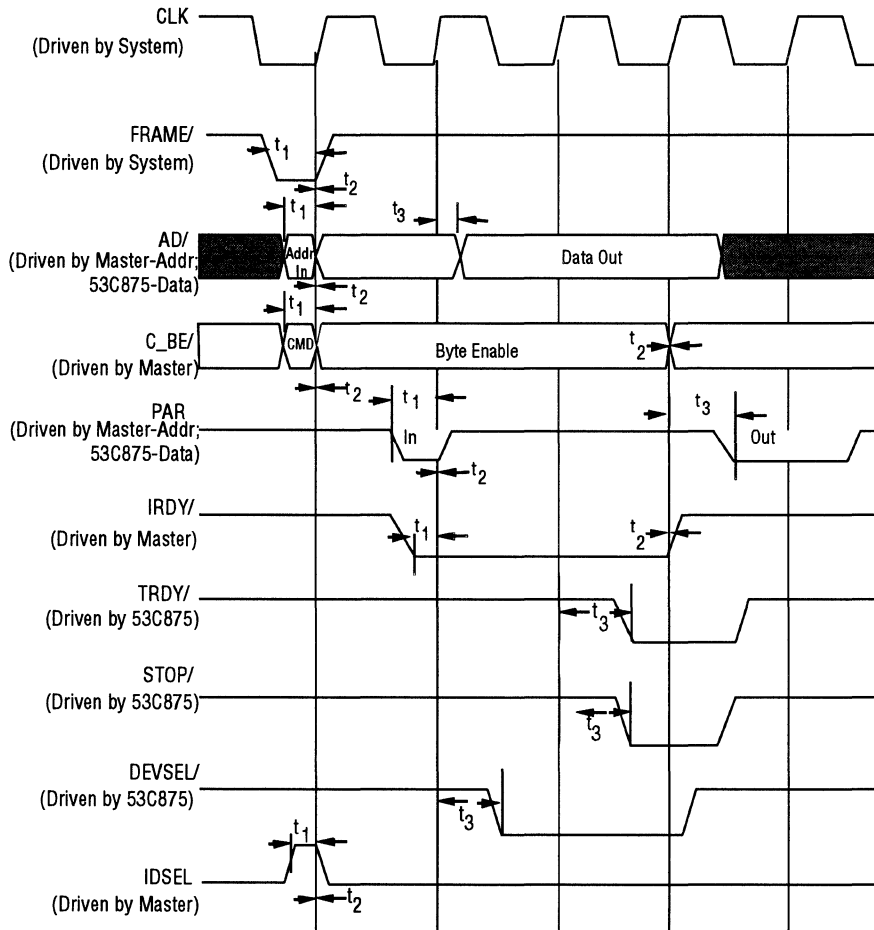
#### Initiator Cycles

- Op code fetch, non-burst
- Burst op code fetch
- Back-to-back read
- Back-to-back write
- Burst read
- Burst write

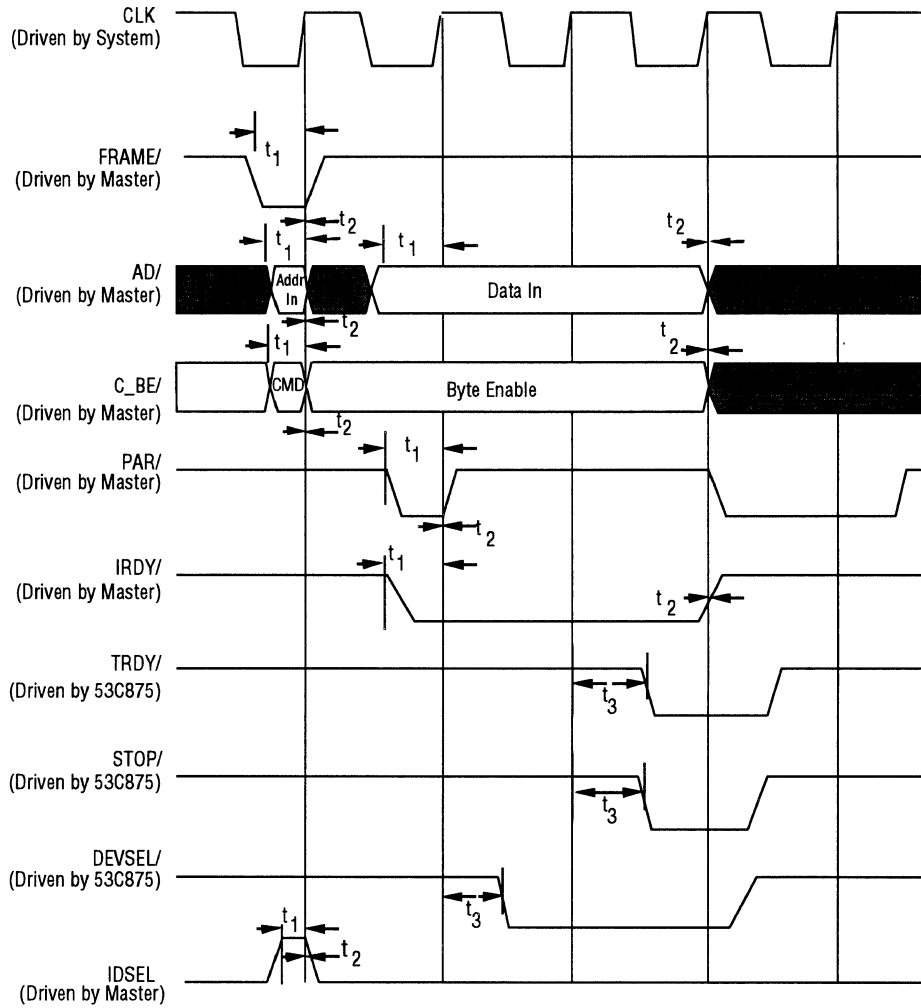
#### External Memory Cycles

- Read cycle, normal/fast memory ( $\geq 64\text{K}$ ), single-byte access
- Write cycle, normal/fast memory ( $\geq 64\text{K}$ ), single-byte access
- Read cycle, normal/fast memory ( $\geq 64\text{K}$ ), multiple-byte access
- Write cycle, normal/fast memory ( $\geq 64\text{K}$ ), multiple-byte access
- Read cycle, slow memory ( $> 64\text{K}$ )
- Write cycle, slow memory ( $> 64\text{K}$ )
- Read cycle, Normal/Fast Memory ( $\geq 64\text{K}$ )
- Write cycle, Normal/Fast Memory ( $\geq 64\text{K}$ )
- Read Cycle, Slow Memory ( $\leq 64\text{K}$ )
- Write Cycle, Slow Memory ( $\leq 64\text{K}$ )

Electrical Characteristics  
**PCI and External Memory Interface Timing Diagrams**

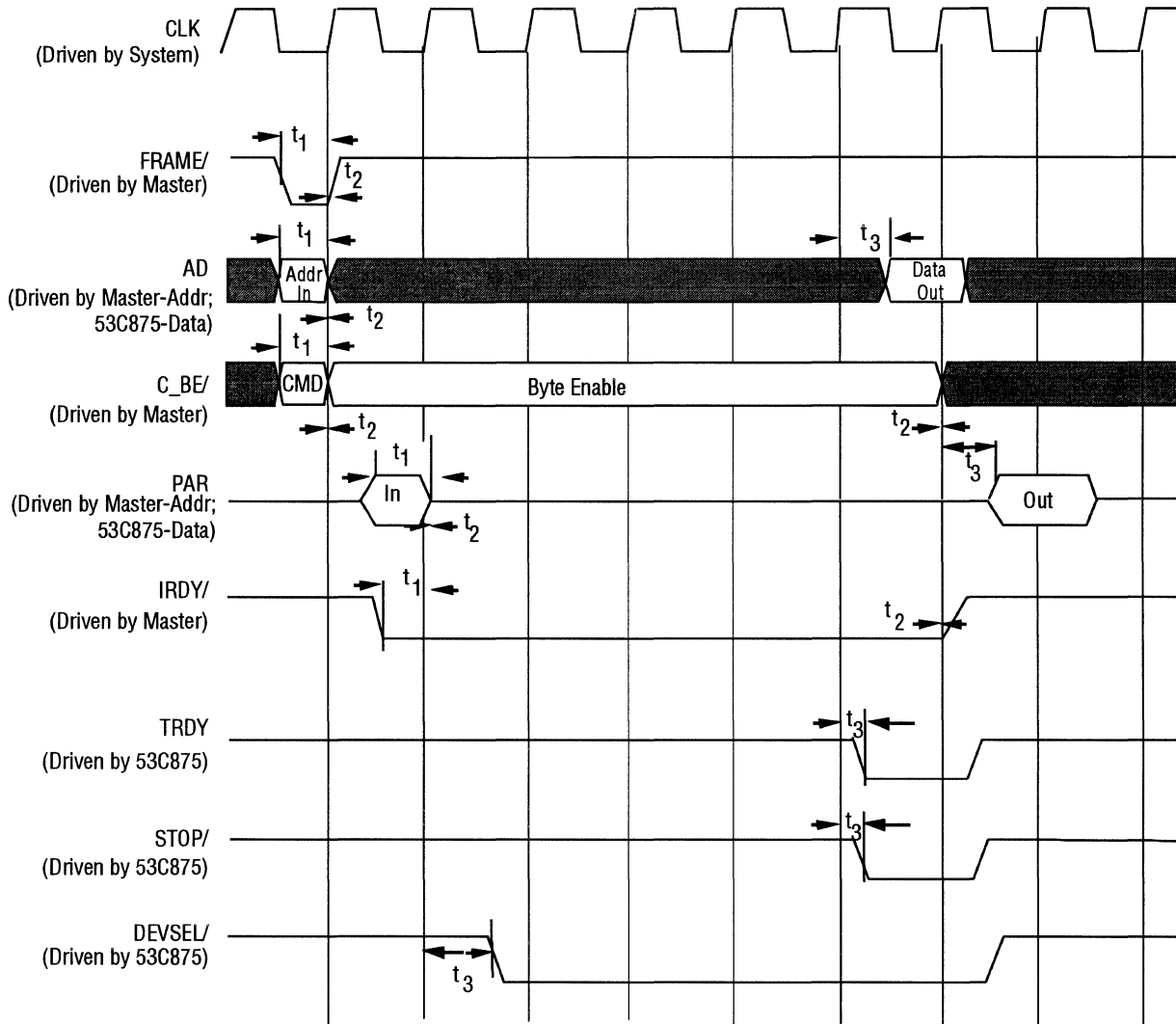


**Figure 7-9: PCI Configuration Register Read**



**Figure 7-10: PCI Configuration Register Write**

Electrical Characteristics  
**PCI and External Memory Interface Timing Diagrams**



**Figure 7-11: Operating Register/SCRIPTS RAM read**

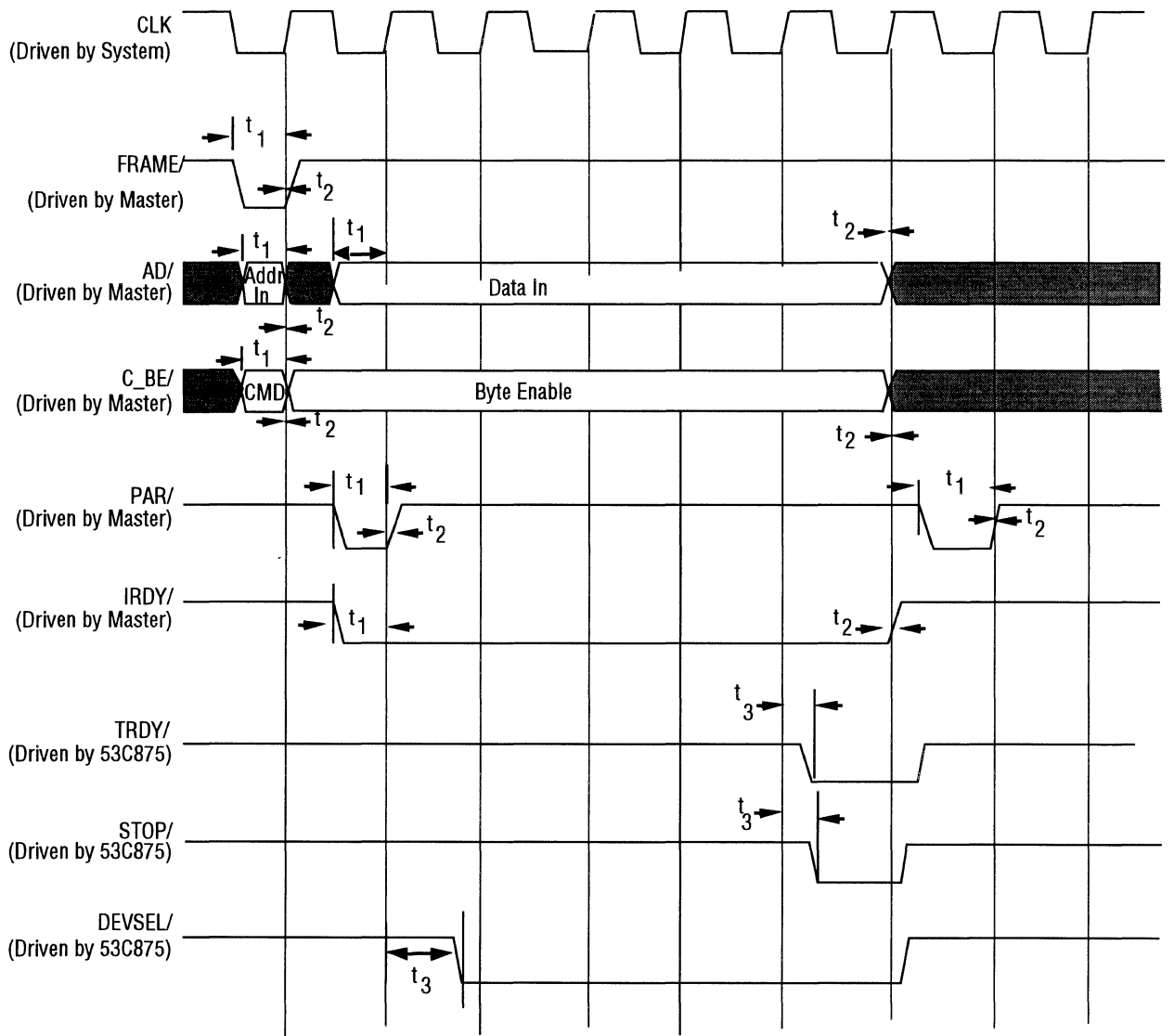


Figure 7-12: Operating Register/SCRIPTS RAM write

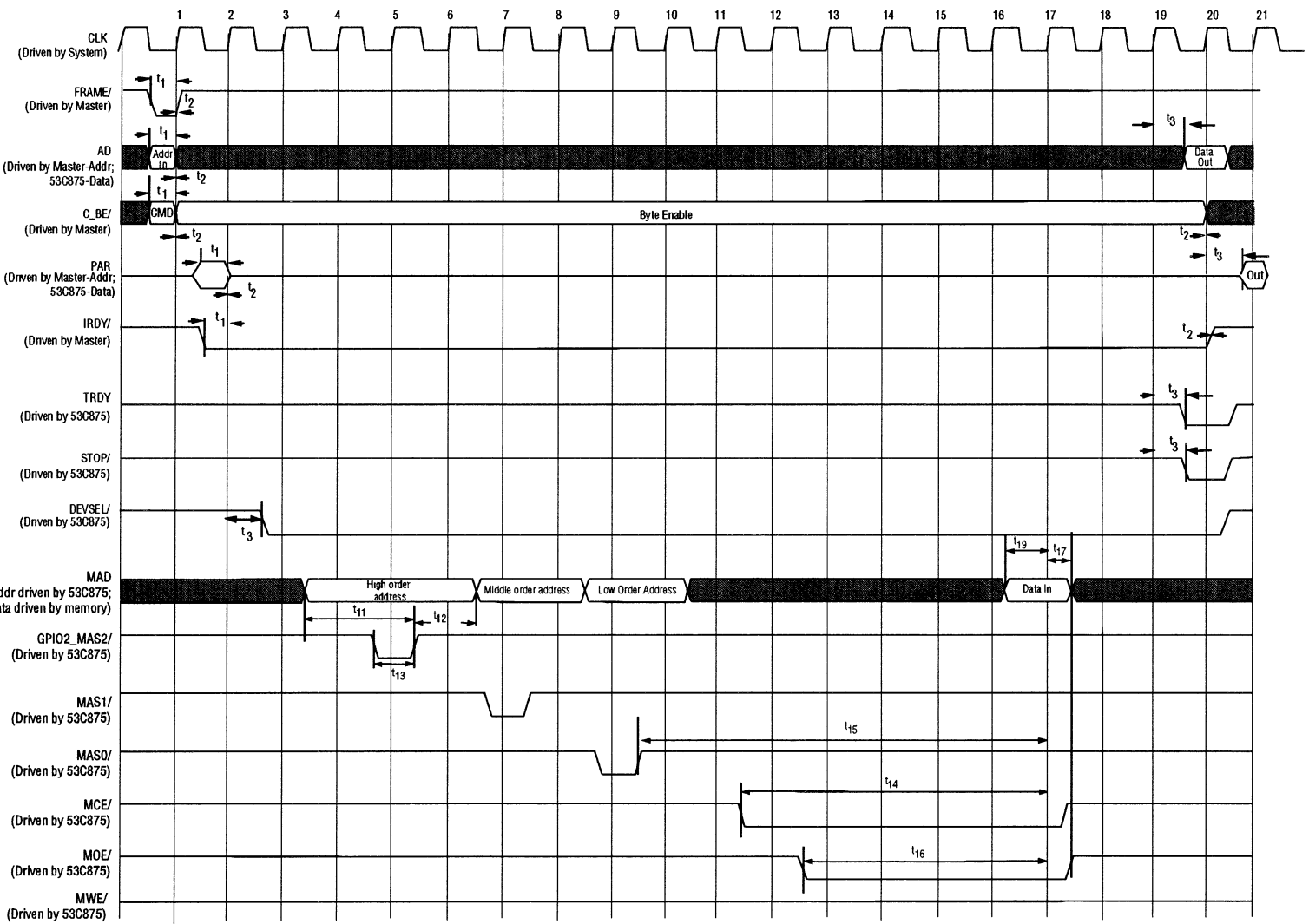


Figure 7-13: External Memory Read

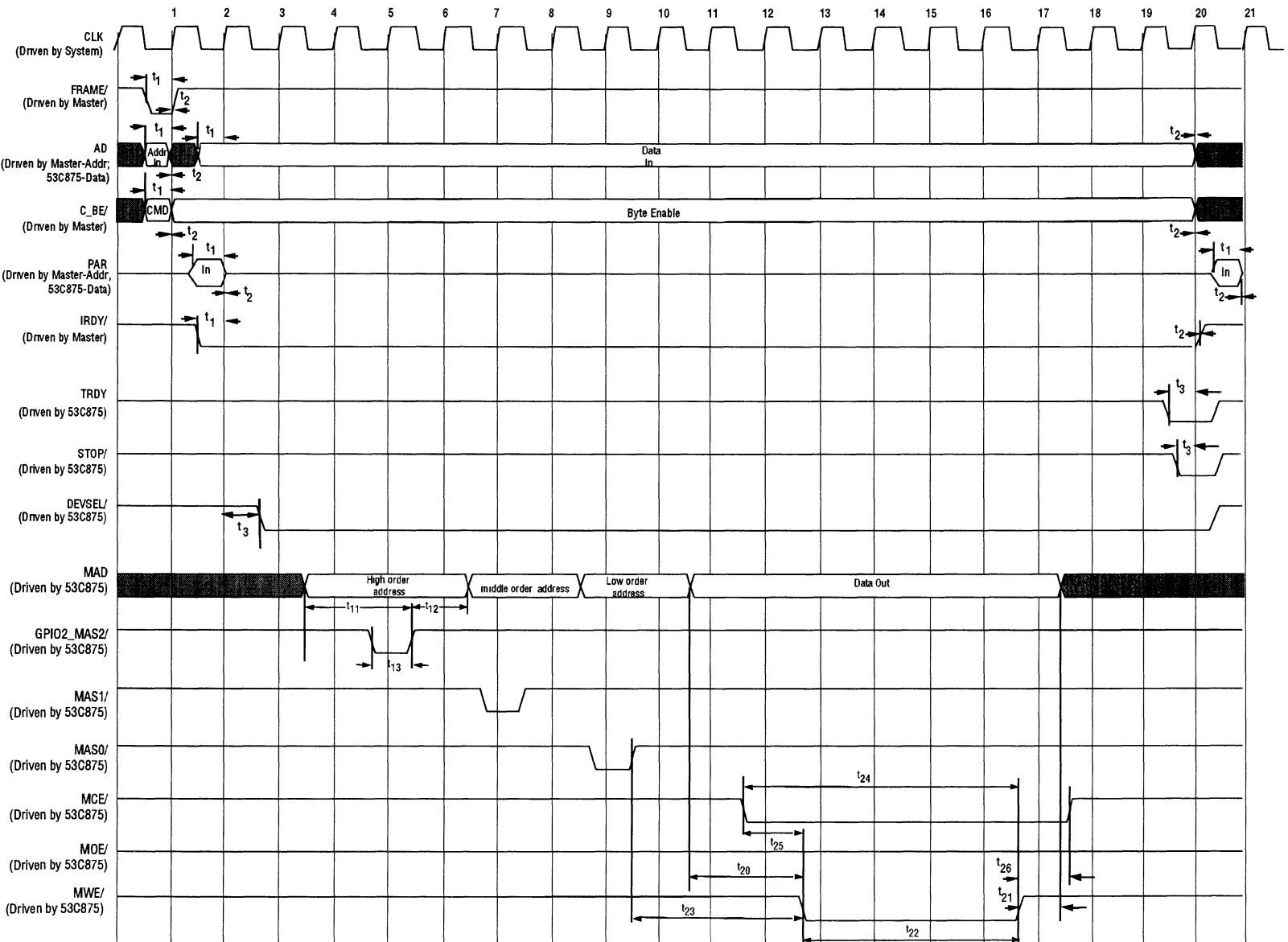
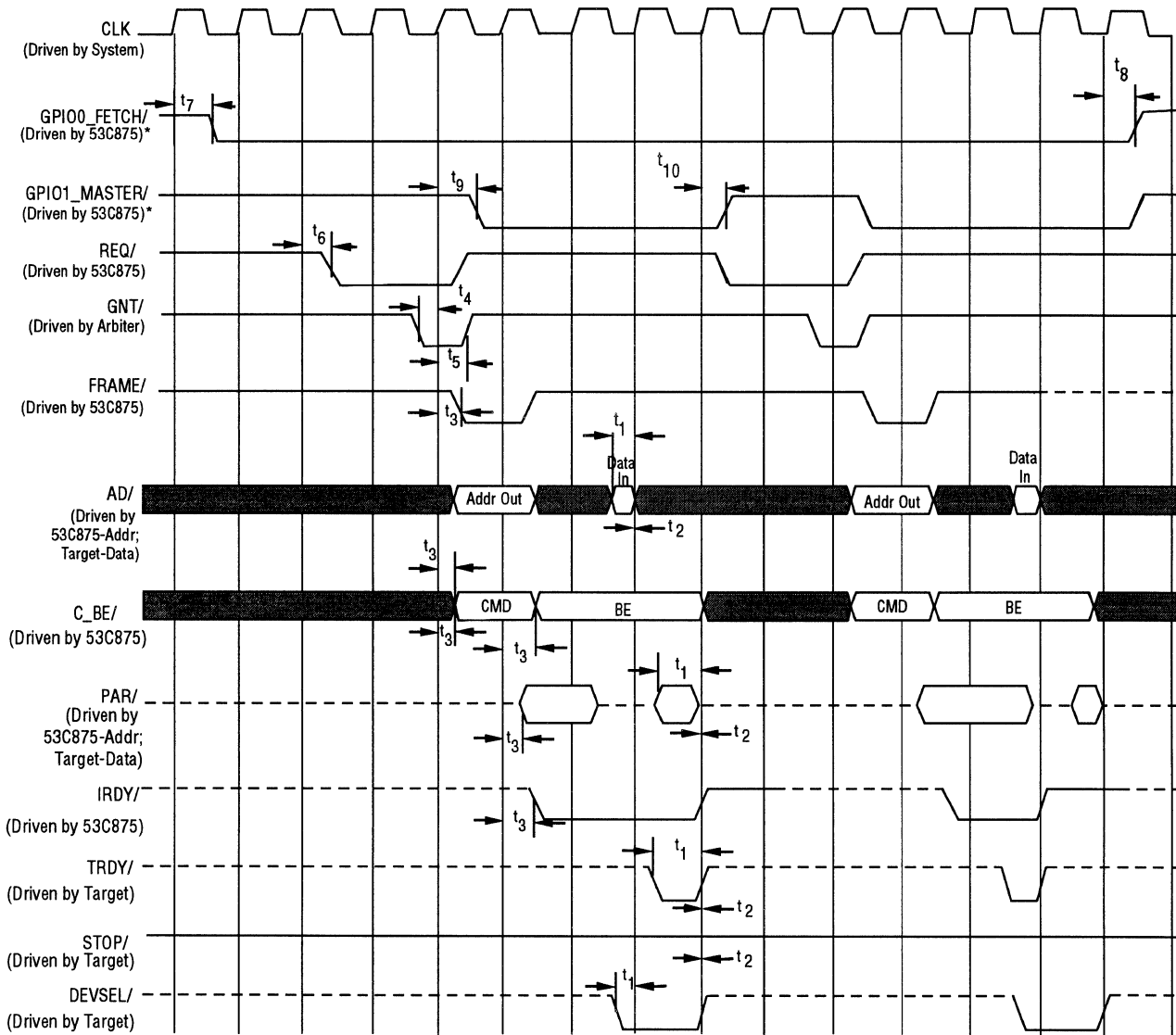


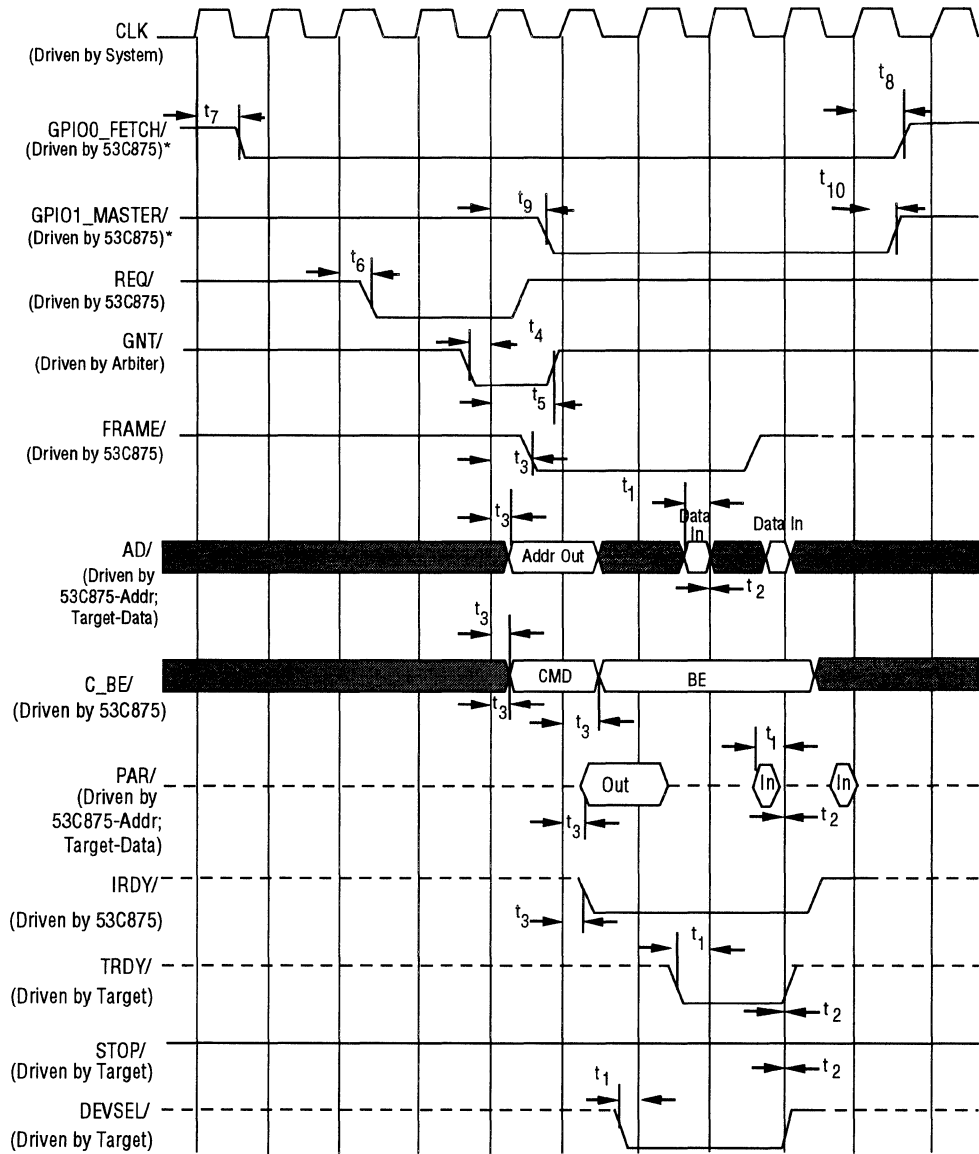
Figure 7-14: External Memory Write

Electrical Characteristics  
**PCI and External Memory Interface Timing Diagrams**



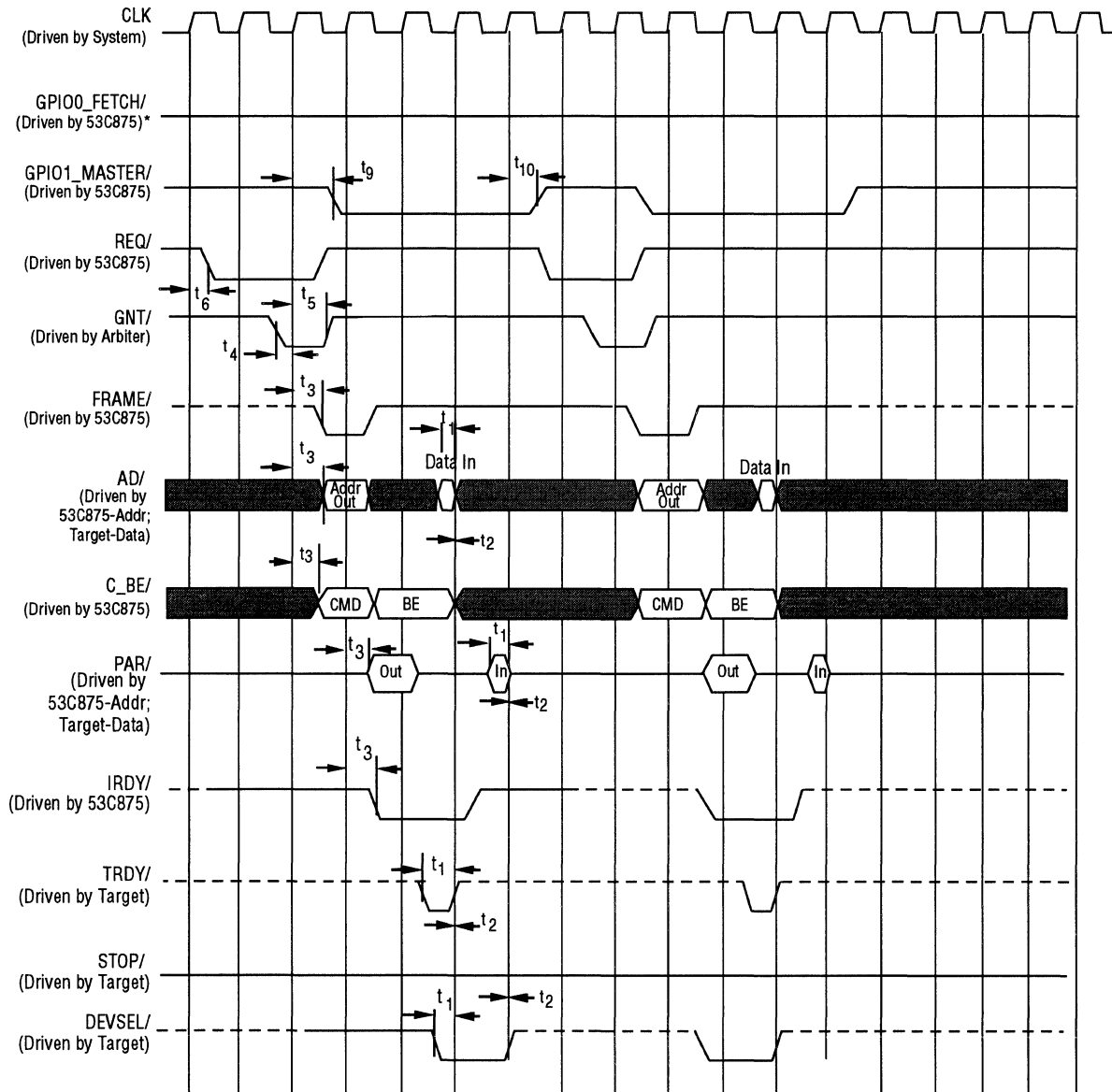
**Figure 7-15: Op Code Fetch, non-burst**



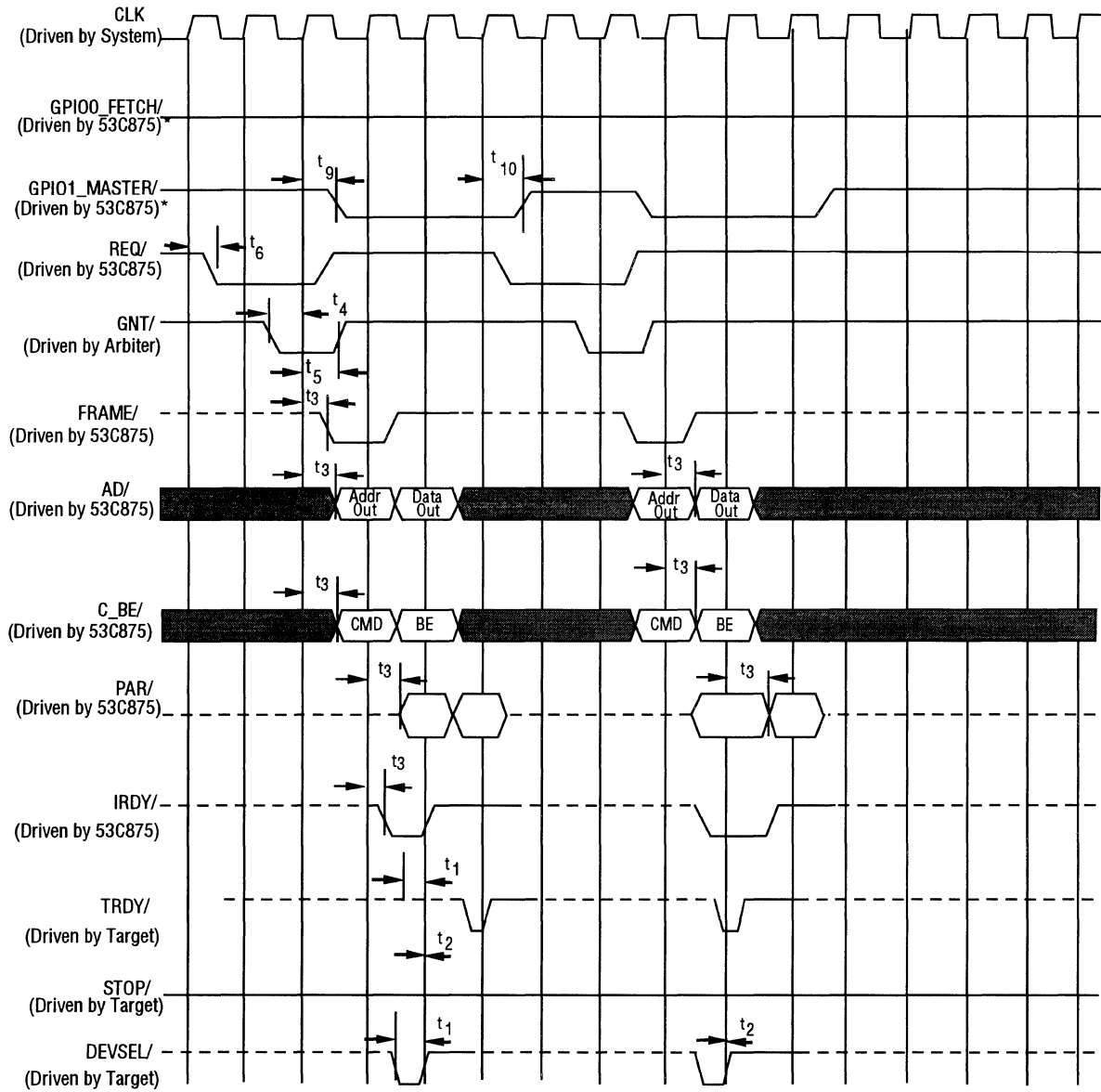


**Figure 7-16: Burst Op Code Fetch**

Electrical Characteristics  
**PCI and External Memory Interface Timing Diagrams**



**Figure 7-17: Back to Back Read**



**Figure 7-18: Back to Back Write**

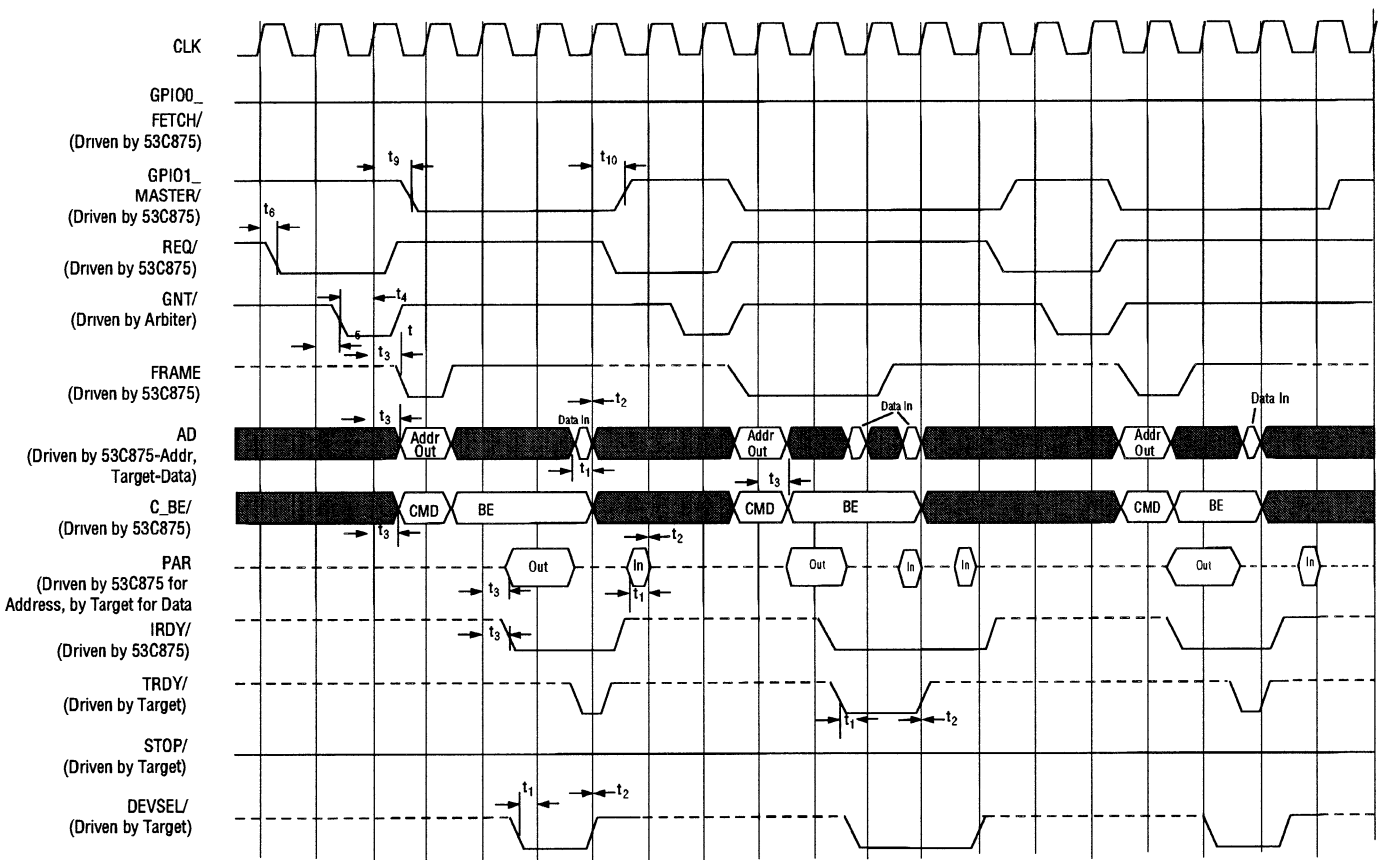


Figure 7-19: Burst Read

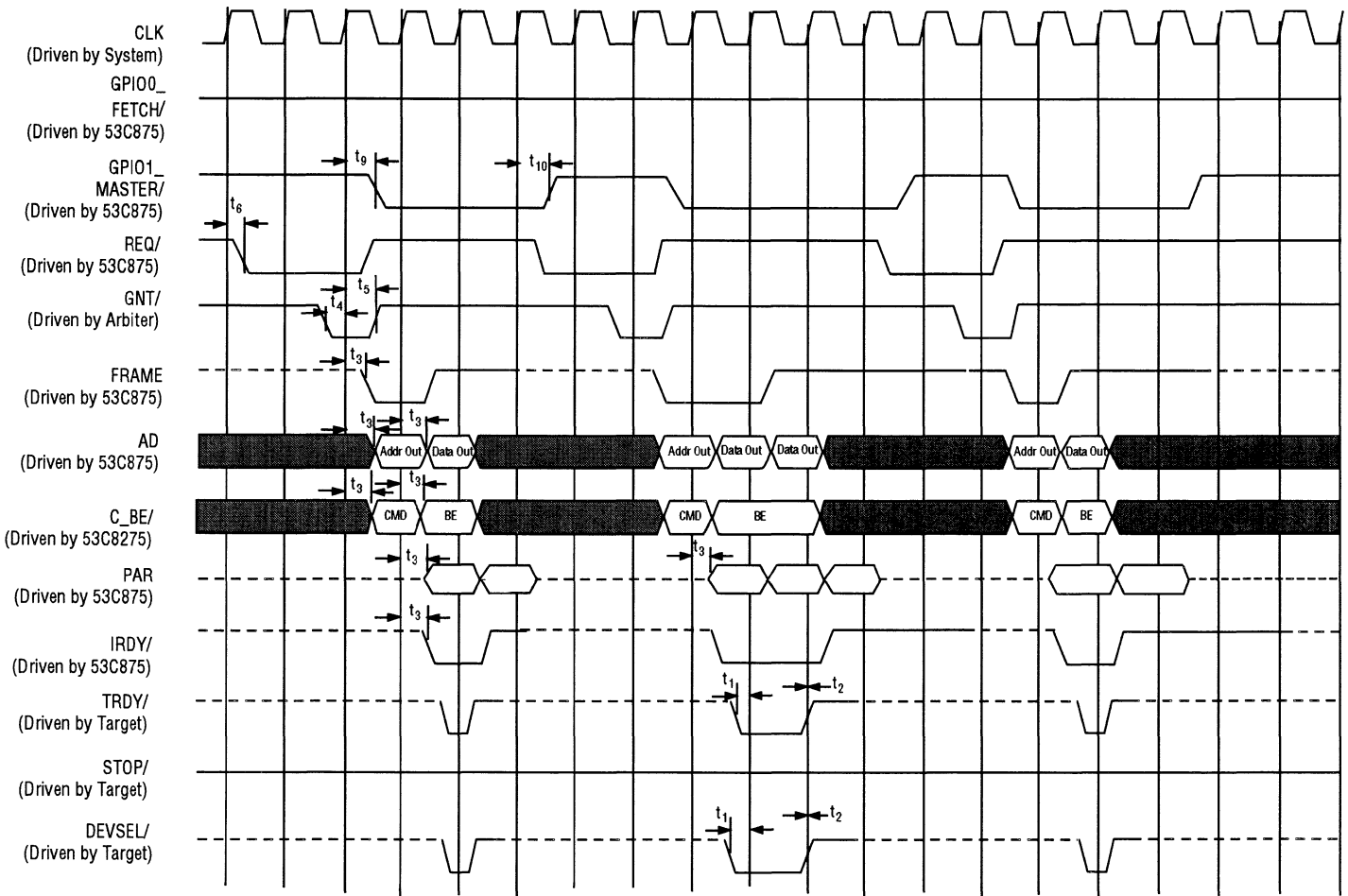


Figure 7-20: Burst Write

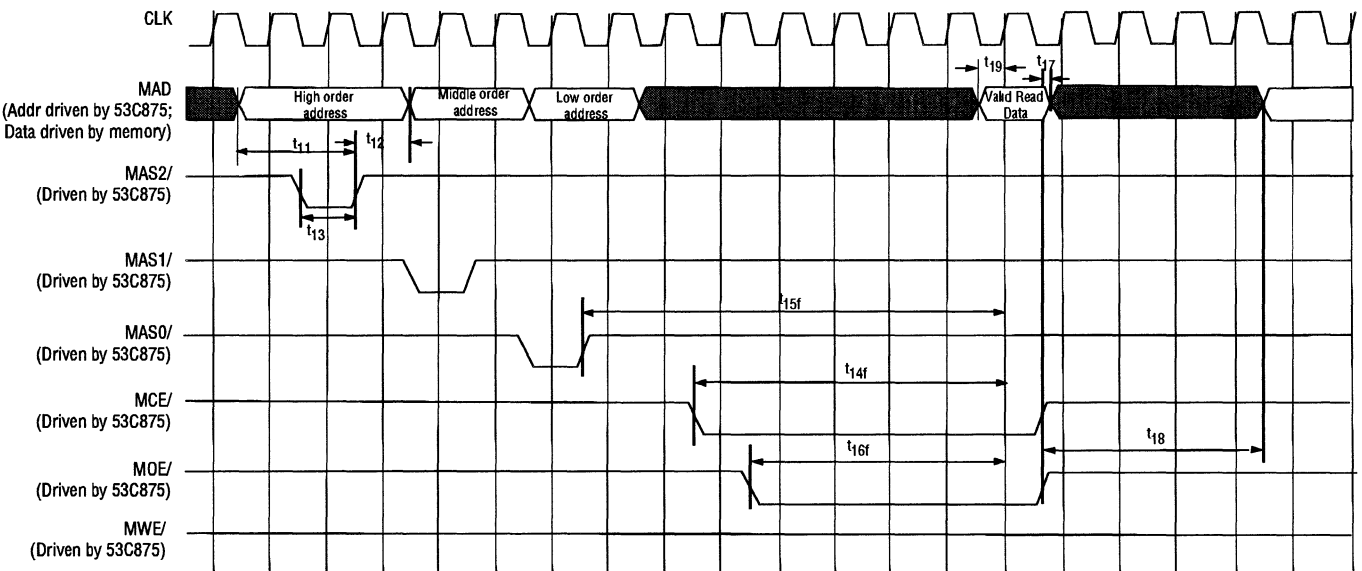


Figure 7-21: Read Cycle, Normal/Fast Memory (>64 KB), single-byte access

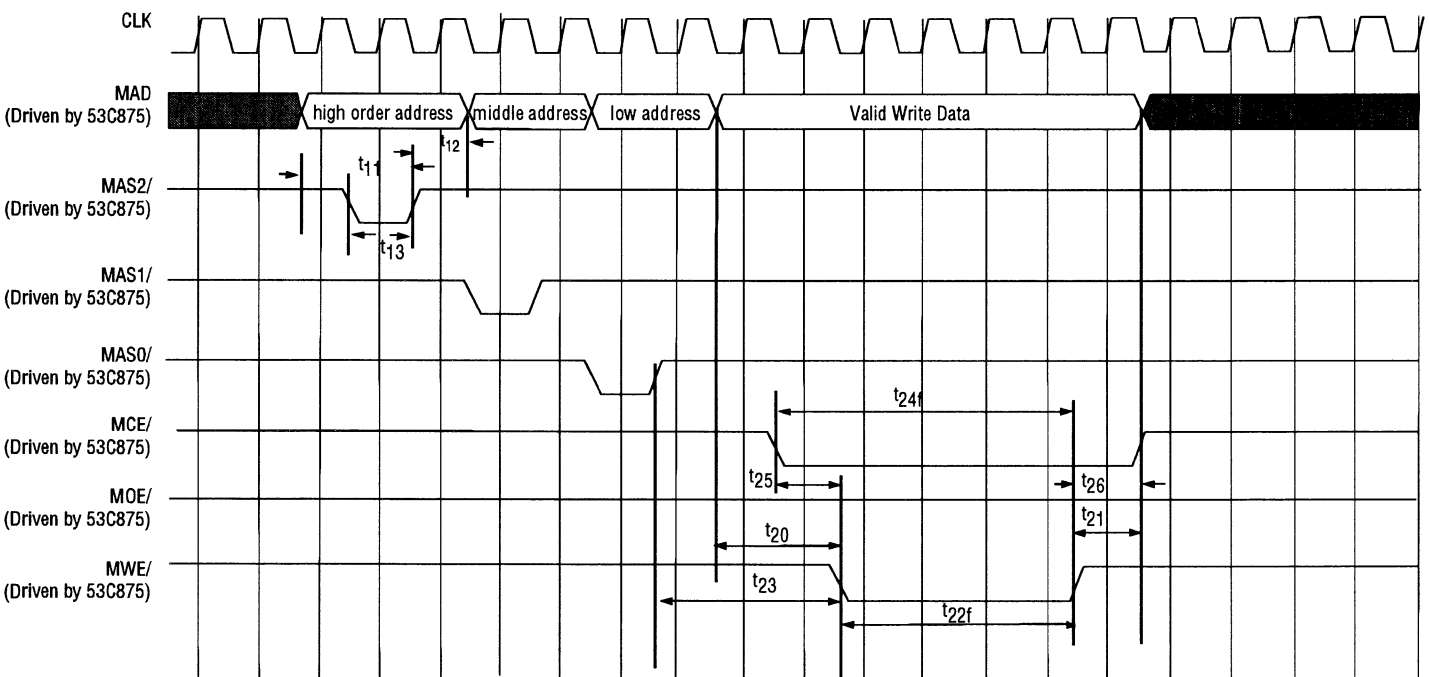


Figure 7-22: Write Cycle, Normal/Fast Memory (>64 KB), single-byte access

Electrical Characteristics  
 PCI and External Memory Interface Timing Diagrams

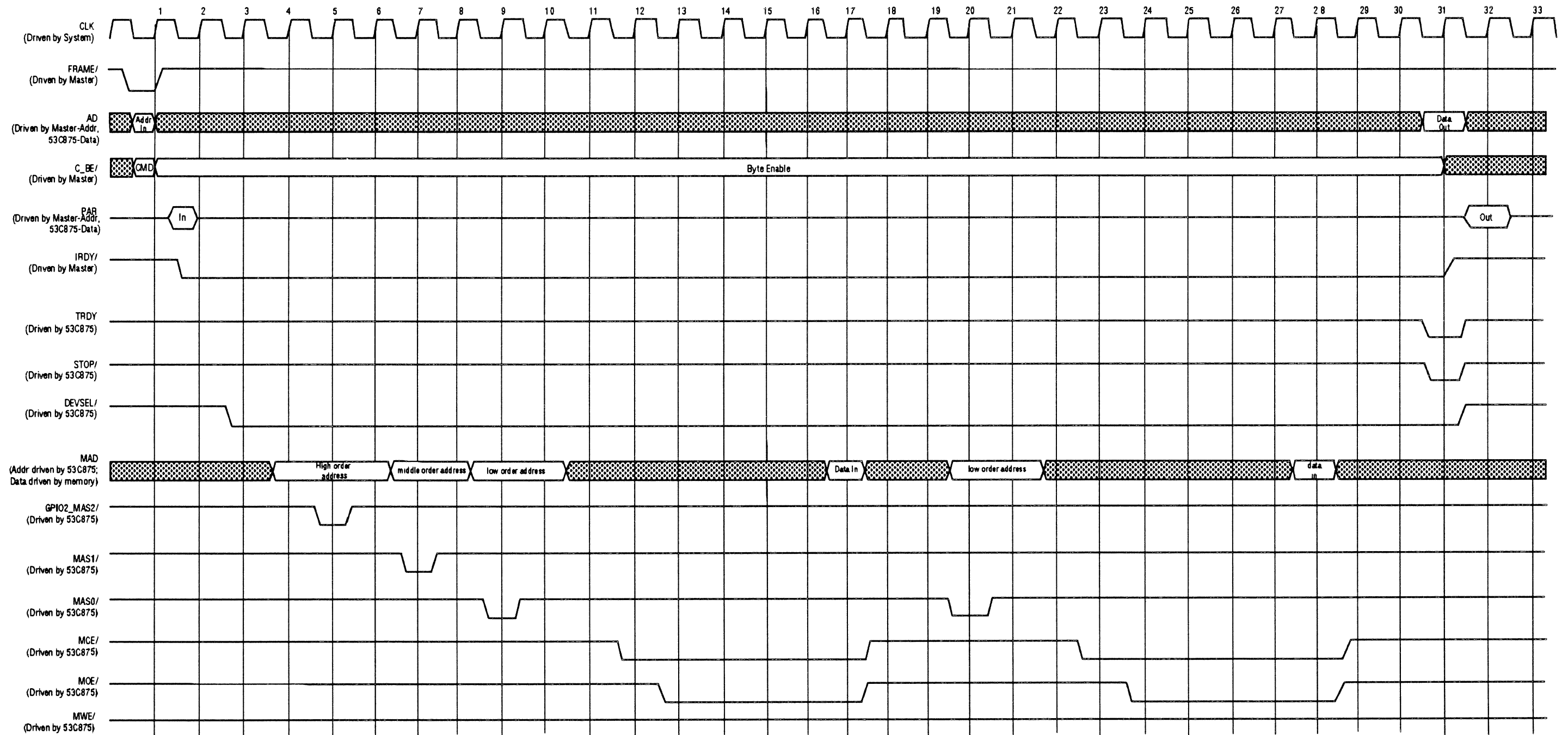


Figure 7-23: Read Cycle, Normal/Fast Memory (≥ 64 KB ), multiple-byte access



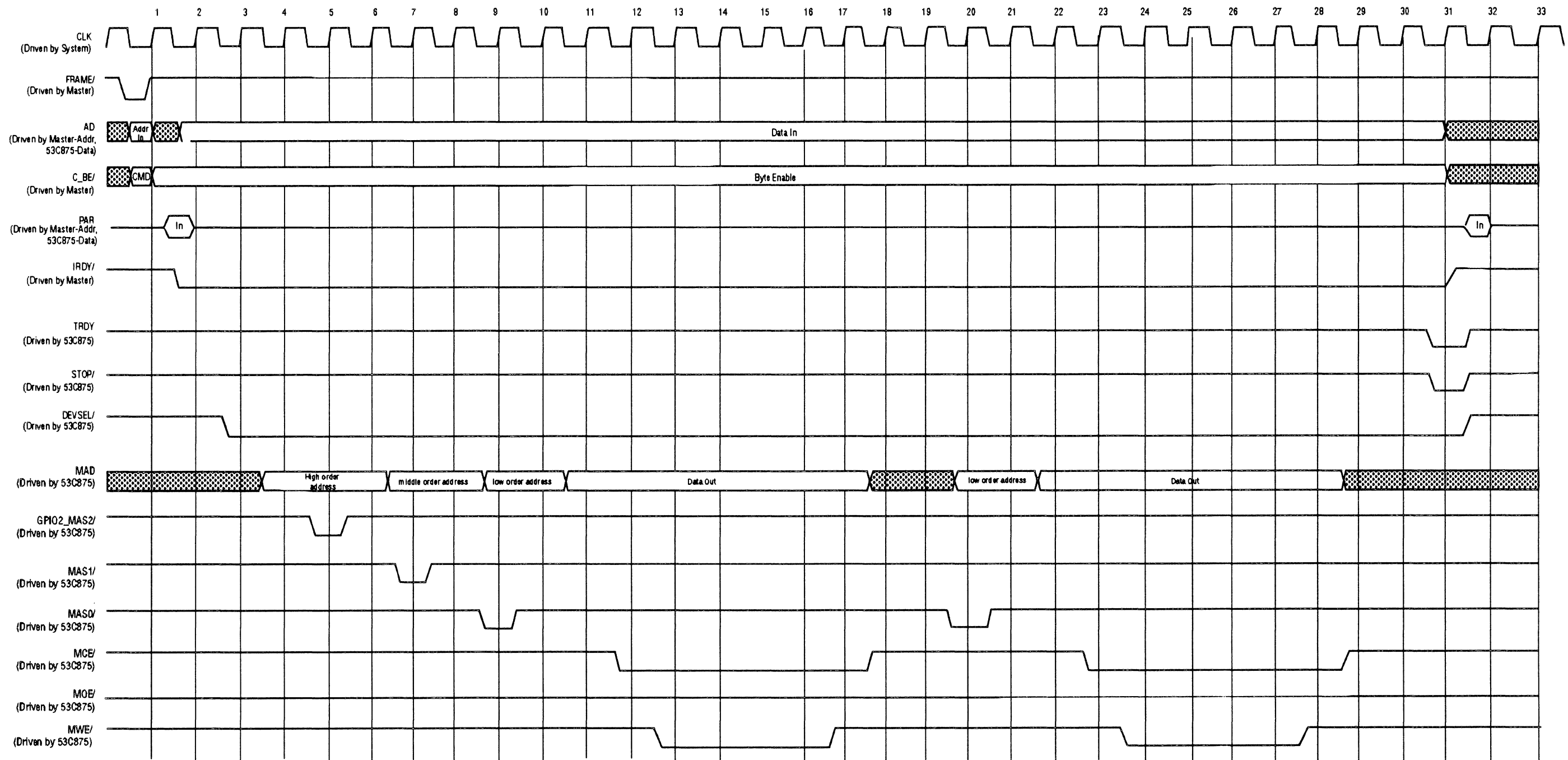


Figure 7-24: Write Cycle, Normal/Fast Memory ( $\geq 64$  KB), multiple-byte access

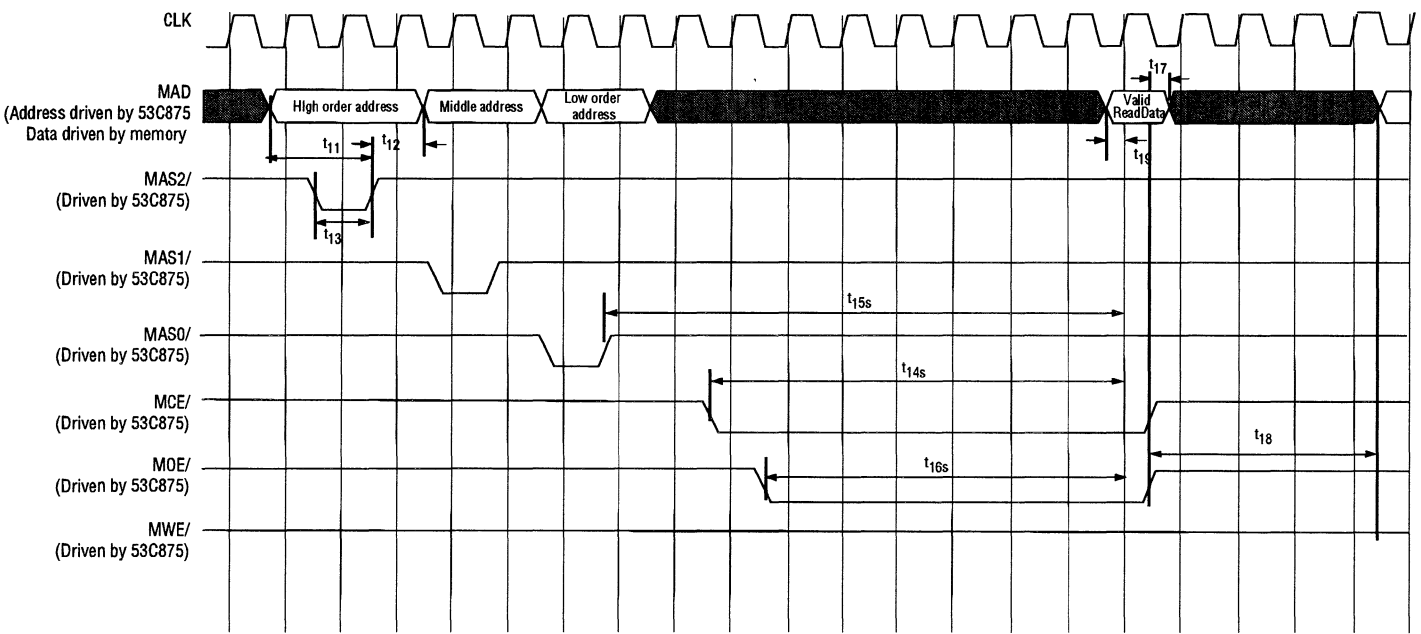


Figure 7-25: Read Cycle, Slow Memory (>64 KB)

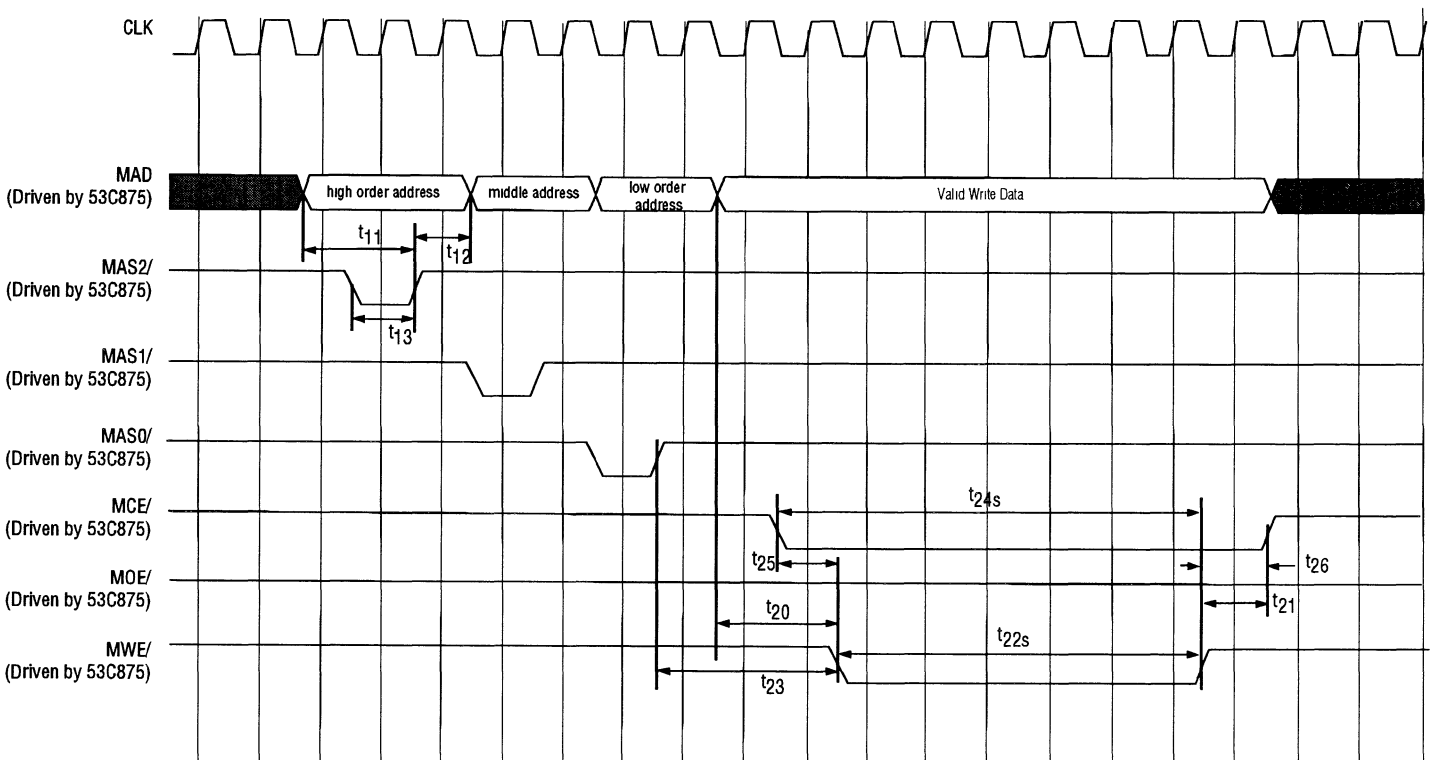


Figure 7-26: Write Cycle, Slow Memory (>64 KB)

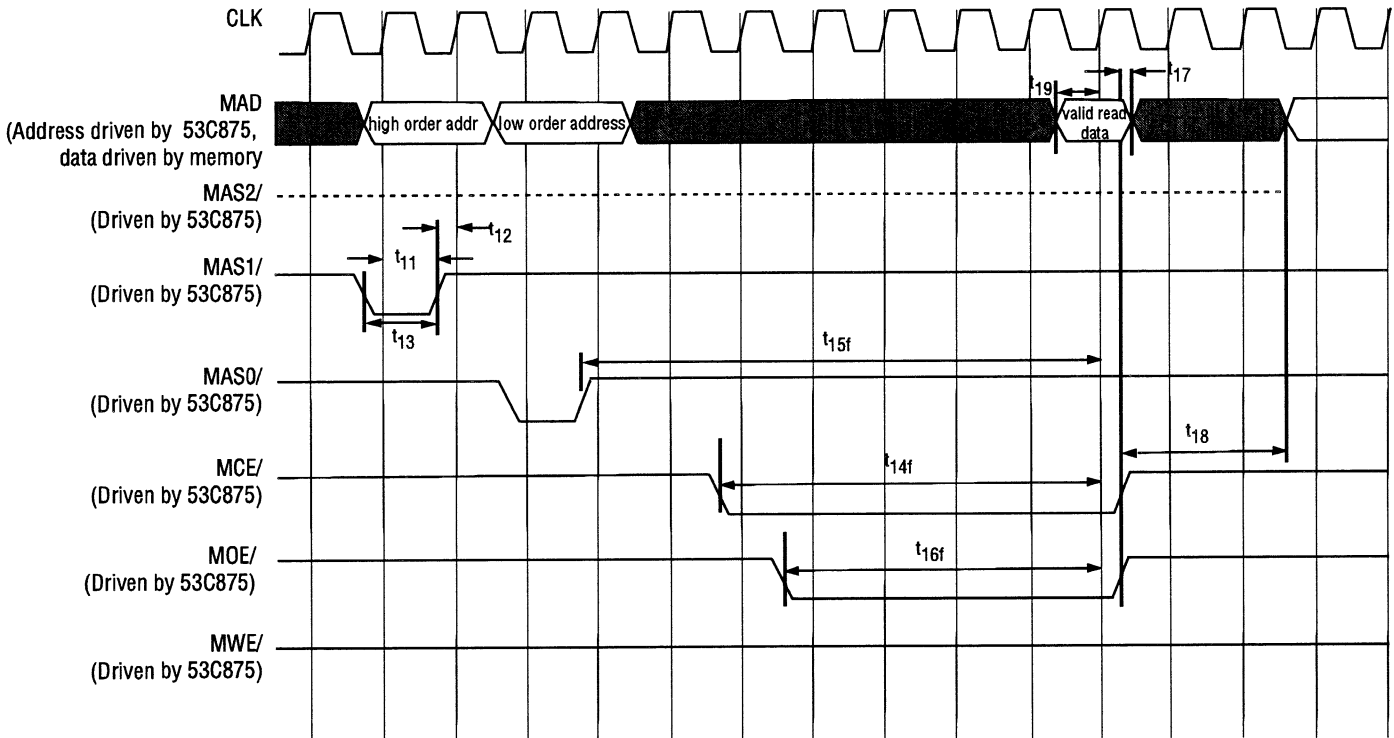


Figure 7-27: Read Cycle, Normal/Fast Memory ( $\leq 64K$ )

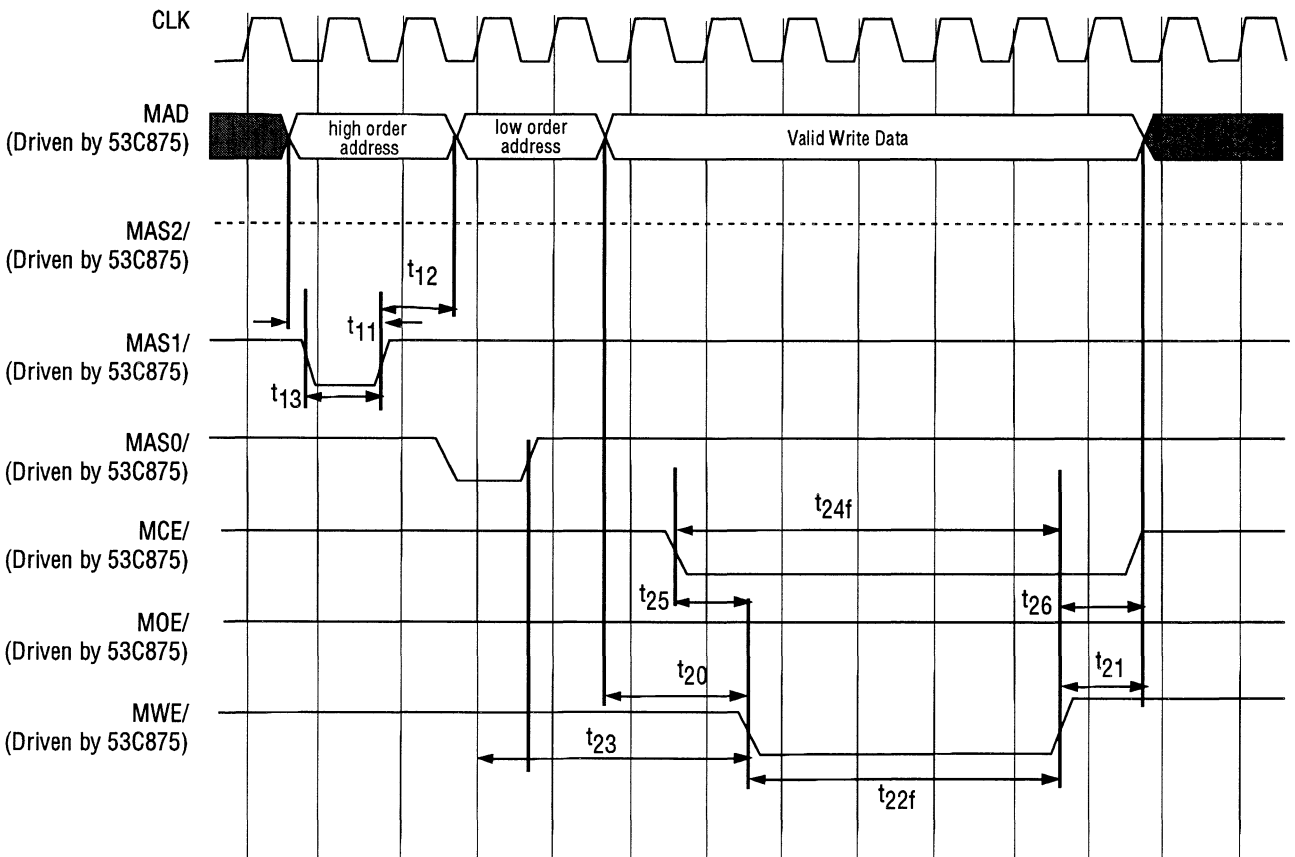


Figure 7-28: Write Cycle, Normal/Fast Memory ( $\leq 64$  K)

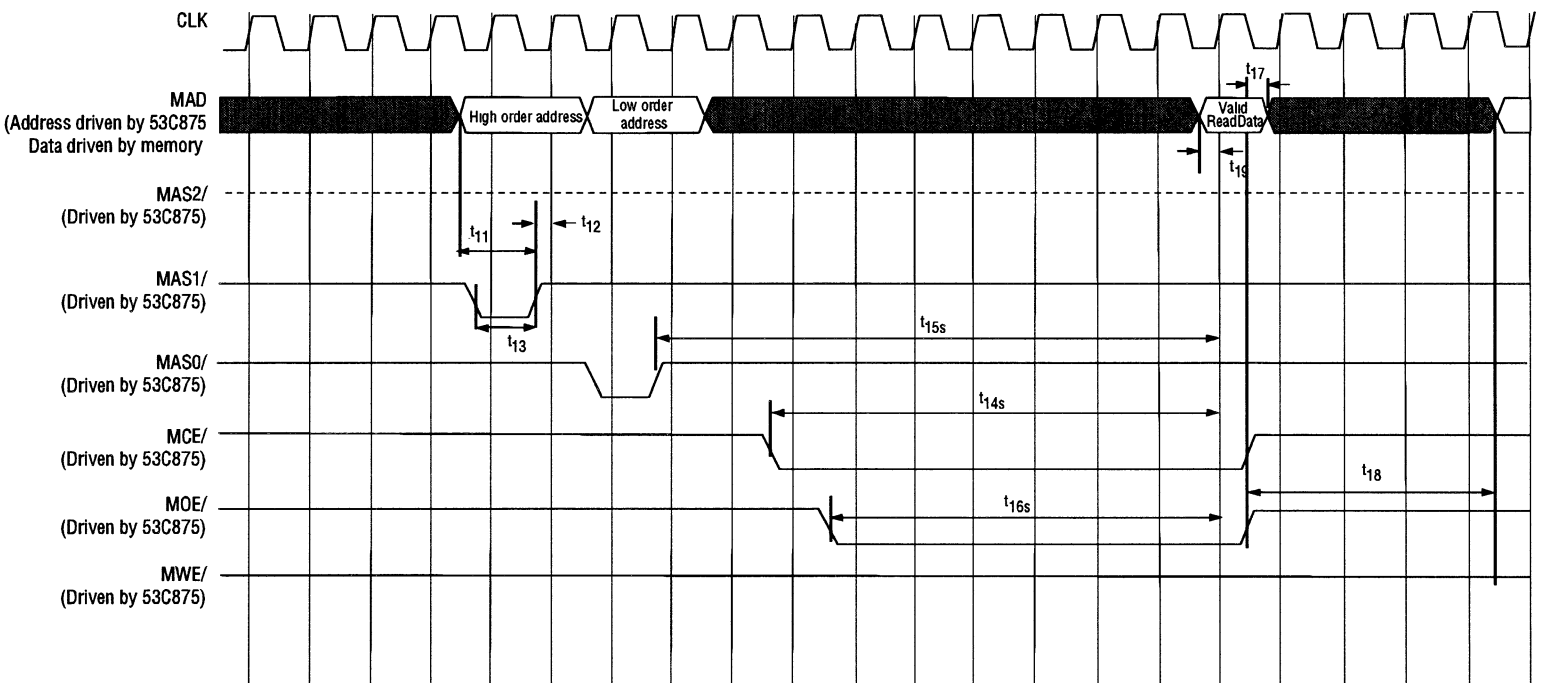


Figure 7-29: Read Cycle, Slow Memory ( $\leq 64K$ )

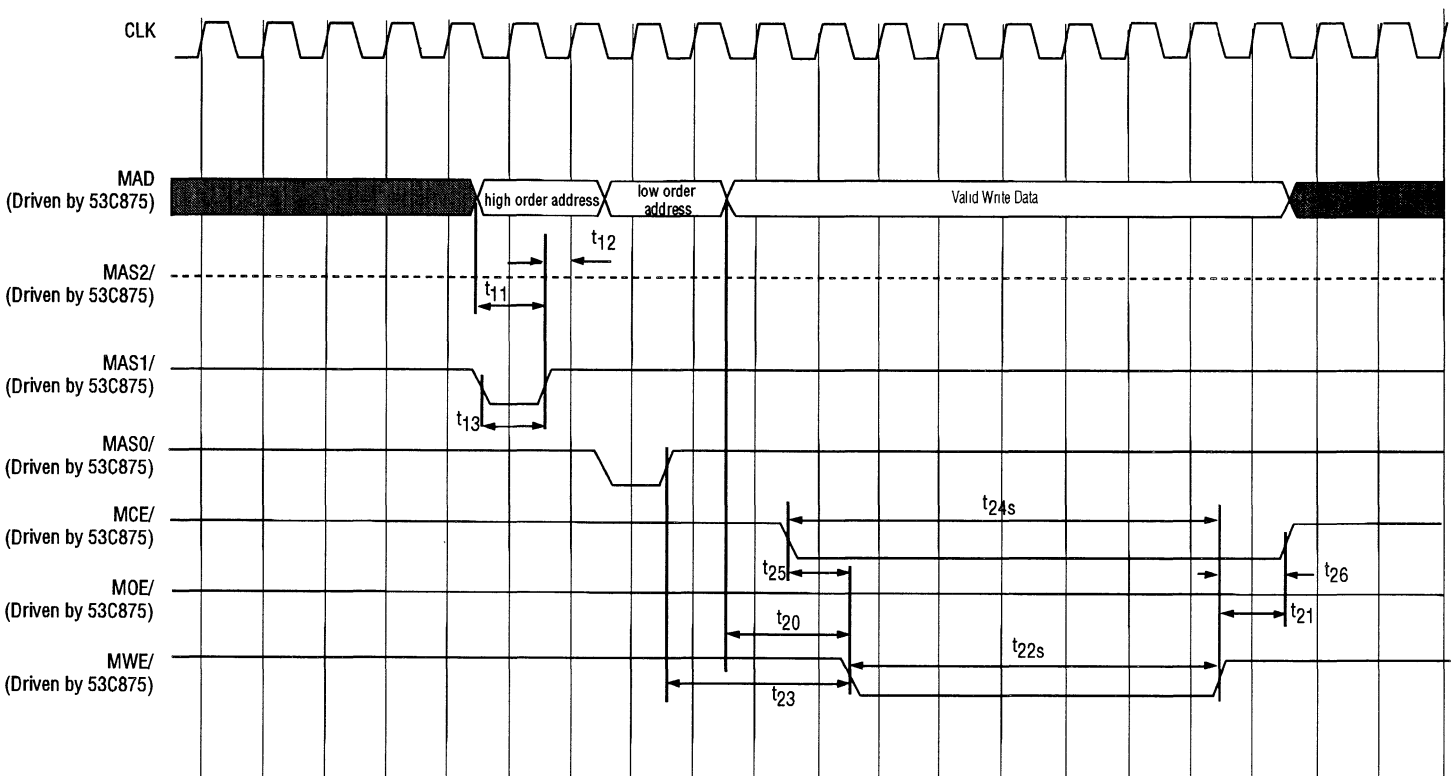


Figure 7-30: Write Cycle, Slow Memory (<math>\leq 64K</math>)

## PCI and External Memory Interface Timings

**Table 7-16: SYM53C875 PCI and External Memory Interface Timings**

Symbol	Parameter	Min	Max	Unit
t <sub>1</sub>	Shared signal input setup time	7	-	ns
t <sub>2</sub>	Shared signal input hold time	0	-	ns
t <sub>3</sub>	CLK to shared signal output valid	-	11	ns
t <sub>4</sub>	Side signal input setup time	10	-	ns
t <sub>5</sub>	Side signal input hold time	0	-	ns
t <sub>6</sub>	CLK to side signal output valid	-	12	ns
t <sub>7</sub>	CLK high to FETCH/ low	-	20	ns
t <sub>8</sub>	CLK high to FETCH/ high	-	20	ns
t <sub>9</sub>	CLK high to MASTER/ low	-	20	ns
t <sub>10</sub>	CLK high to MASTER/ high	-	20	ns
t <sub>11</sub>	Address setup to MAS/ high	25	-	ns
t <sub>12</sub>	Address hold from MAS/ high	15	-	ns
t <sub>13</sub>	MAS/ pulse width	25	-	ns
t <sub>14f</sub>	MCE/ low to data clocked in (fast memory)	160	-	ns
t <sub>14s</sub>	MCE/ low to data clocked in (slow memory)	220	-	ns
t <sub>15f</sub>	Address valid to data clocked in (fast memory)	205	-	ns
t <sub>15s</sub>	Address valid to data clocked in (slow memory)	265	-	ns
t <sub>16f</sub>	MOE/ low to data clocked in (fast memory)	100	-	ns
t <sub>16s</sub>	MOE/ low to data clocked in (slow memory)	160	-	ns
t <sub>17</sub>	Data hold from address, MOE/, MCE/ change	0	-	ns
t <sub>18</sub>	Next address out from MOE/, MCE/ high	50	-	ns
t <sub>19</sub>	Data setup to CLK high	5	-	ns
t <sub>20</sub>	Data setup to MWE/ low	30	-	ns
t <sub>21</sub>	Data hold from MWE/ high	20	-	ns
t <sub>22f</sub>	MWE/ pulse width (fast memory)	100	-	ns
t <sub>22s</sub>	MWE/ pulse width (slow memory)	160	-	ns



**Table 7-16: SYM53C875 PCI and External Memory Interface Timings (Continued)**

Symbol	Parameter	Min	Max	Unit
$t_{23}$	Address setup to MWE/ low	75	-	ns
$t_{24f}$	MCE/ low to MWE/ high (fast memory)	120	-	ns
$t_{24s}$	MCE/ low to MWE/ high (slow memory)	180	-	ns
$t_{25}$	MCE/ low to MWE/ low	25	-	ns
$t_{26}$	MWE/ high to MCE/ high	25	-	ns

## SCSI Timings

Figure 7-31: Initiator Asynchronous Send

Symbol	Parameter	Min	Max	Units
$t_1$	SACK/ asserted from SREQ/ asserted	5	-	ns
$t_2$	SACK/ deasserted from SREQ/ deasserted	5	-	ns
$t_3$	Data setup to SACK/ asserted	55	-	ns
$t_4$	Data hold from SREQ/ deasserted	20	-	ns

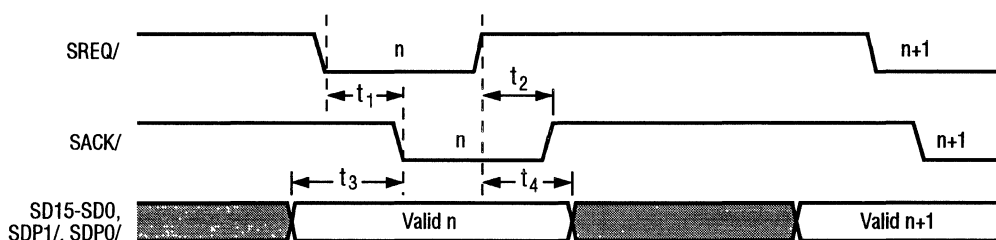


Figure 7-32: Initiator Asynchronous Receive

Symbol	Parameter	Min	Max	Units
$t_1$	SACK/ asserted from SREQ/ asserted	5	-	ns
$t_2$	SACK/ deasserted from SREQ/ deasserted	5	-	ns
$t_3$	Data setup to SREQ/ asserted	0	-	ns
$t_4$	Data hold from SACK/ asserted	0	-	ns

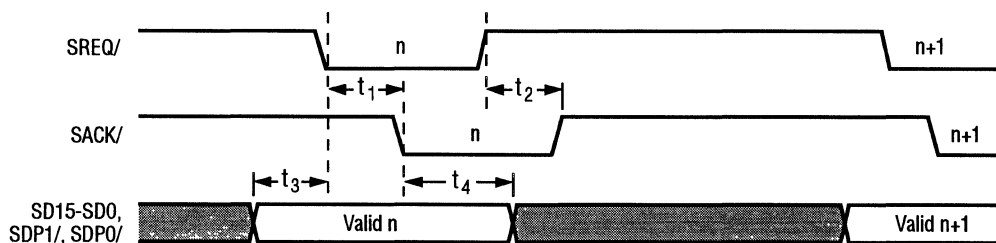


Figure 7-33: Target Asynchronous Send

Symbol	Parameter	Min	Max	Units
$t_1$	SREQ/ deasserted from SACK/ asserted	5	-	ns
$t_2$	SREQ/ asserted from SACK/ deasserted	5	-	ns
$t_3$	Data setup to SREQ/ asserted	55	-	ns
$t_4$	Data hold from SACK/ asserted	20	-	ns

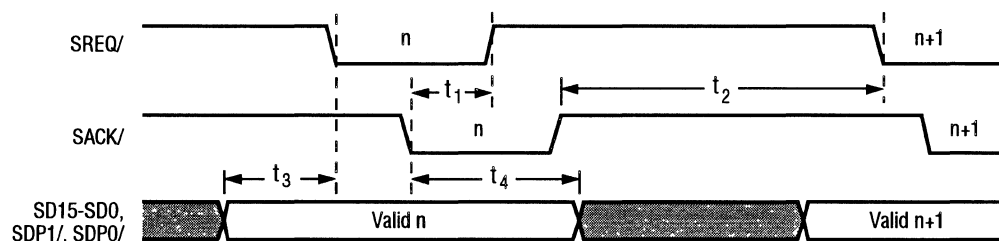
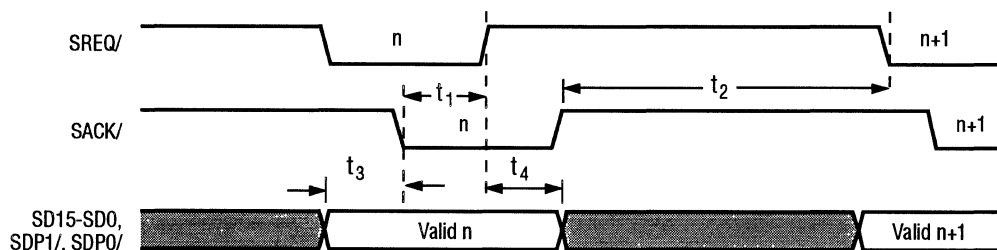


Figure 7-34: Target Asynchronous Receive

Symbol	Parameter	Min	Max	Units
$t_1$	SREQ/ deasserted from SACK/ asserted	5	-	ns
$t_2$	SREQ/ asserted from SACK/ deasserted	5	-	ns
$t_3$	Data setup to SACK/ asserted	0	-	ns
$t_4$	Data hold from SREQ/ deasserted	0	-	ns



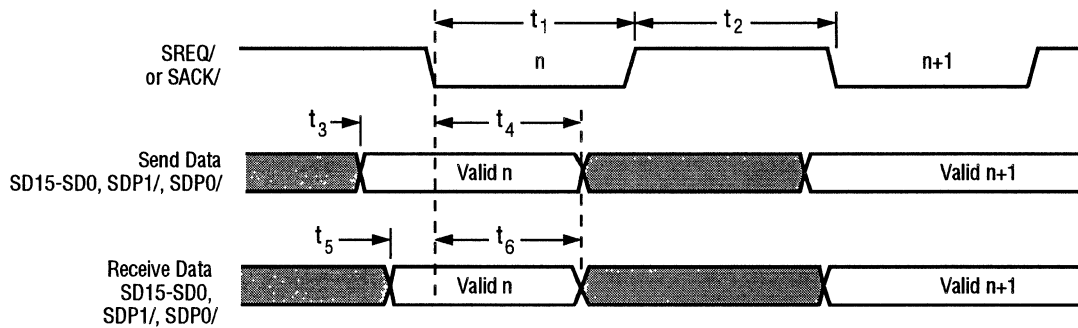


Figure 7-35: Initiator and Target Synchronous Transfers

Table 7-17: SCSI-1 transfers (Single-Ended, 5.0 MB/s)

Symbol	Parameter	Min	Max	Units
$t_1$	Send SREQ/ or SACK/ assertion pulse width	90	-	ns
$t_2$	Send SREQ/ or SACK/ deassertion pulse width	90	-	ns
$t_1$	Receive SREQ/ or SACK/ assertion pulse width	90	-	ns
$t_2$	Receive SREQ/ or SACK/ deassertion pulse width	90	-	ns
$t_3$	Send data setup to SREQ/ or SACK/ asserted	55	-	ns
$t_4$	Send data hold from SREQ/ or SACK/ asserted	100	-	ns
$t_5$	Receive data setup to SREQ/ or SACK/ asserted	0	-	ns
$t_6$	Receive data hold from SREQ/ or SACK/ asserted	45	-	ns

**Table 7-18: SCSI-1 Transfers (Differential, 4.17 MB/s)**

Symbol	Parameter	Min	Max	Units
t <sub>1</sub>	Send SREQ/ or SACK/ assertion pulse width	96	-	ns
t <sub>2</sub>	Send SREQ/ or SACK/ deassertion pulse width	96	-	ns
t <sub>1</sub>	Receive SREQ/ or SACK/ assertion pulse width	84	-	ns
t <sub>2</sub>	Receive SREQ/ or SACK/deassertion pulse width	84	-	ns
t <sub>3</sub>	Send data setup to SREQ/ or SACK/ asserted	65	-	ns
t <sub>4</sub>	Send data hold from SREQ/ or SACK/ asserted	110	-	ns
t <sub>5</sub>	Receive data setup to SREQ/ or SACK/ asserted	0	-	ns
t <sub>6</sub>	Receive data hold from SREQ/ or SACK/ asserted	45	-	ns

**Table 7-19: SCSI-2 Fast Transfers (10.0 MB/s (8-bit transfers) or 20.0 MB/s (16-bit transfers), 40 MHz clock)**

Symbol	Parameter	Min	Max	Units
t <sub>1</sub>	Send SREQ/ or SACK/ assertion pulse width	35	-	ns
t <sub>2</sub>	Send SREQ/ or SACK/ deassertion pulse width	35	-	ns
t <sub>1</sub>	Receive SREQ/ or SACK/ assertion pulse width	20	-	ns
t <sub>2</sub>	Receive SREQ/ or SACK/ deassertion pulse width	20	-	ns
t <sub>3</sub>	Send data setup to SREQ/ or SACK/ asserted	33	-	ns
t <sub>4</sub>	Send data hold from SREQ/ or SACK/ asserted	45	-	ns
t <sub>5</sub>	Receive data setup to SREQ/ or SACK/ asserted	0	-	ns
t <sub>6</sub>	Receive data hold from SREQ/ or SACK/ asserted	10	-	ns

**Table 7-20: SCSI-2 Fast Transfers (10.0 MB/s (8-bit transfers) or 20.0 MB/s (16-bit transfers), 50 MHz clock)**

Symbol	Parameter	Min	Max	Unit
t <sub>1</sub>	Send SREQ/ or SACK/ assertion pulse width	35	-	ns
t <sub>2</sub>	Send SREQ/ or SACK/ deassertion pulse width	35	-	ns
t <sub>1</sub>	Receive SREQ/ or SACK/ assertion pulse width	20	-	ns
t <sub>2</sub>	Receive SREQ/ or SACK/ deassertion pulse width	20	-	ns
t <sub>3</sub>	Send data setup to SREQ/ or SACK/ asserted	33	-	ns
t <sub>4</sub>	Send data hold from SREQ/ or SACK/ asserted	40**	-	ns
t <sub>5</sub>	Receive data setup to SREQ/ or SACK/ asserted	0	-	ns
t <sub>6</sub>	Receive data hold from SREQ/ or SACK/ asserted	10	-	ns

\*Transfer period bits (bits 6-4 in the SXFER register) are set to zero and the Extra Clock Cycle of Data Setup bit (bit 7 in SCNTL1) is set.

\*\*Analysis of system configuration is recommended due to reduced driver skew margin in differential systems

Note: for fast SCSI, set the TolerANT Enable bit (bit 7 in STEST3).

**Table 7-21: UltraSCSI Single-Ended Transfers (20.0 MB/s (8-bit transfers) or 40.0 MB/s (16-bit transfers), 80 MHz clock)**

Symbol	Parameter	Min	Max	Unit
t <sub>1</sub>	Send SREQ/ or SACK/ assertion pulse width	16	-	ns
t <sub>2</sub>	Send SREQ/ or SACK/ deassertion pulse width	16	-	ns
t <sub>1</sub>	Receive SREQ/ or SACK/ assertion pulse width	10	-	ns
t <sub>2</sub>	Receive SREQ/ or SACK/ deassertion pulse width	10	-	ns
t <sub>3</sub>	Send data setup to SREQ/ or SACK/ asserted	12	-	ns
t <sub>4</sub>	Send data hold from SREQ/ or SACK/ asserted	17	-	ns
t <sub>5</sub>	Receive data setup to SREQ/ or SACK/ asserted	0	-	ns
t <sub>6</sub>	Receive data hold from SREQ/ or SACK/ asserted	6	-	ns

\*Transfer period bits (bits 6-4 in the SXFER register) are set to zero and the Extra Clock Cycle of Data Setup bit (bit 7 in SCNTL1) is set.

Note: for fast SCSI, set the TolerANT Enable bit (bit 7 in STEST3). During UltraSCSI transfers, the value of the Extend REQ/ ACK Filtering bit (STEEST2, bit 1) has no effect.

**Table 7-22: UltraSCSI Differential Transfers (20.0 MB/s (8-bit transfers) or 40.0 MB/s (16-bit transfers), 80 MHz clock)**

Symbol	Parameter	Min	Max	Unit
t <sub>1</sub>	Send SREQ/ or SACK/ assertion pulse width	16	-	ns
t <sub>2</sub>	Send SREQ/ or SACK/ deassertion pulse width	16	-	ns
t <sub>1</sub>	Receive SREQ/ or SACK/ assertion pulse width	10	-	ns
t <sub>2</sub>	Receive SREQ/ or SACK/ deassertion pulse width	10	-	ns
t <sub>3</sub>	Send data setup to SREQ/ or SACK/ asserted	16	-	ns
t <sub>4</sub>	Send data hold from SREQ/ or SACK/ asserted	21	-	ns
t <sub>5</sub>	Receive data setup to SREQ/ or SACK/ asserted	0	-	ns
t <sub>6</sub>	Receive data hold from SREQ/ or SACK/ asserted	6	-	ns

*\*Transfer period bits (bits 6-4 in the SXFER register) are set to zero and the Extra Clock Cycle of Data Setup bit (bit 7 in SCNTL1) is set.*

*Note: for fast SCSI, set the TolerANT Enable bit (bit 7 in STEST3). During UltraSCSI transfers, the value of the Extend REQ/ ACK Filtering bit (STEST2, bit 1) has no effect.f*

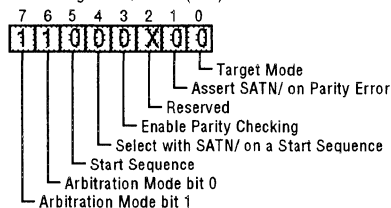
(

:

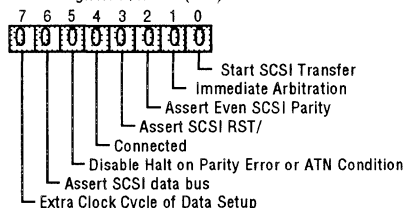


# Appendix A Register Summary

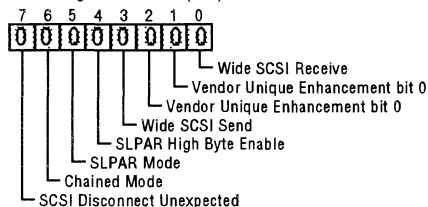
SCNTL0 Register R/W 00h (80h)



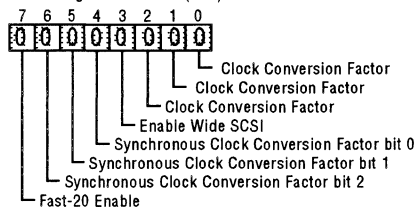
SCNTL1 Register R/W 01h (81h)



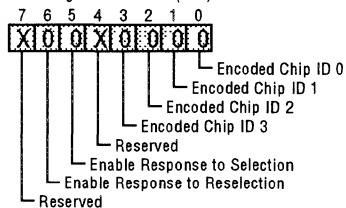
SCNTL2 Register R/W 02h (82h)



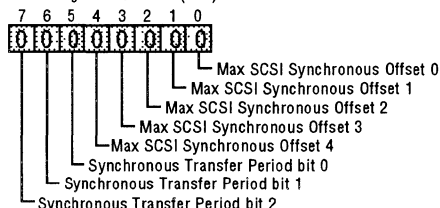
SCNTL3 Register R/W 03h (83h)



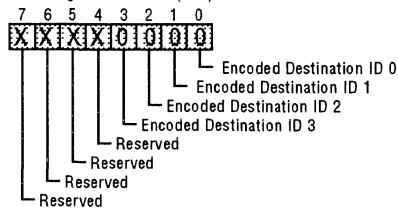
SCID Register R/W 04h (84h)



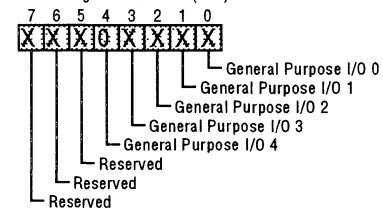
SXFER Register R/W 05h (85h)



SDID Register R/W 06h (86h)

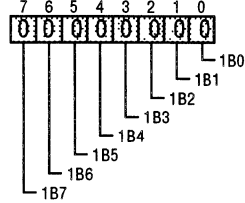


GPREG Register R/W 07h (87h)

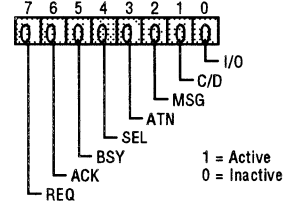


# Register Summary

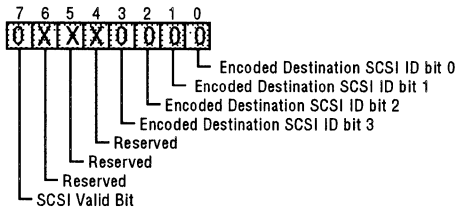
SFBR Register R/W 08h (88h)



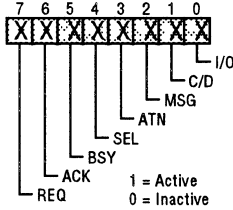
SOCL Register R/W 09h (89h)



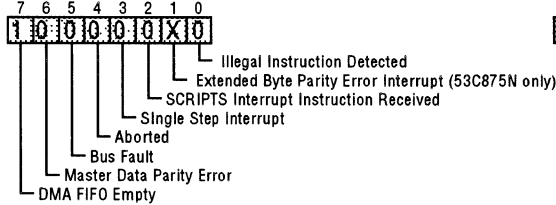
SSID Register R 0Ah (09h)



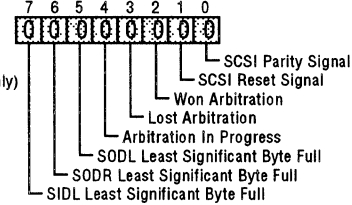
SBCL Register R 0Bh (8Bh)



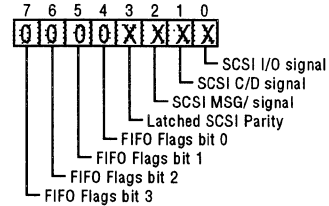
DSTAT Register R 0Ch (8Ch)



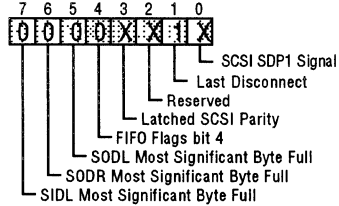
SSTAT0 Register R 0Dh (8Dh)



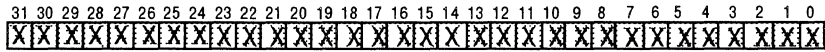
SSTAT1 Register R 0Eh (8Eh)



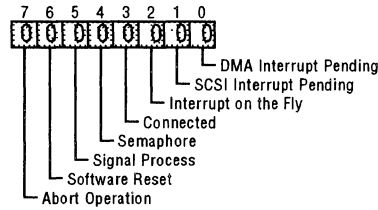
SSTAT2 Register R 0Fh (8Fh)



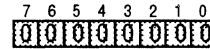
DSA Register R/W 10-13h (90-93h)



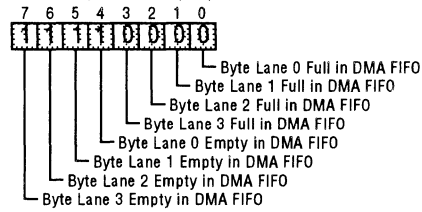
ISTAT Register R/W 14h (94h)



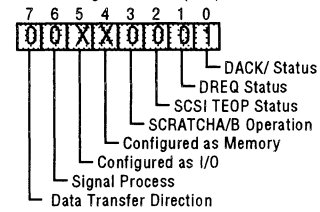
CTEST0 Register R/W 18h (98h)



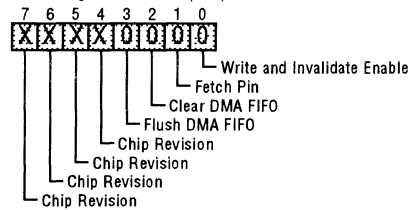
CTEST1 Register R 19h (91h)



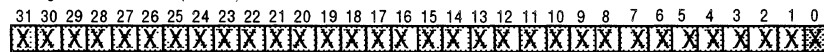
CTEST2 Register R 1Ah (9Ah)



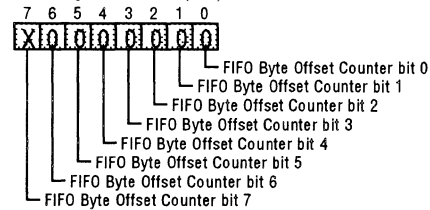
CTEST3 Register R/W 1Bh (9Bh)



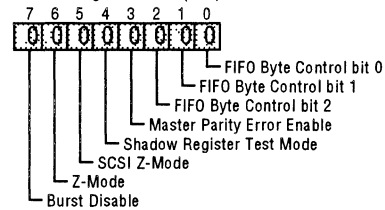
TEMP Register R/W 1C-1Fh (9C-9Fh)



DFIFO Register R/W 20h (A0h)

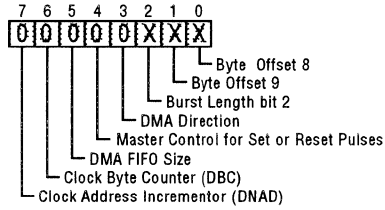


CTEST4 Register R 21h (A1h)

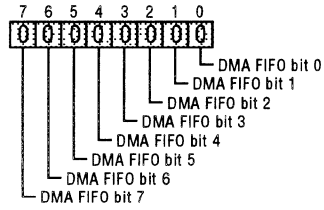


# Register Summary

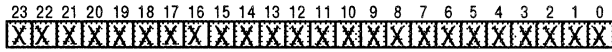
CTEST5 Register R/W 22h (A2h)



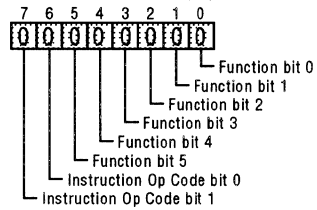
CTEST6 Register R/W 23h (A3h)



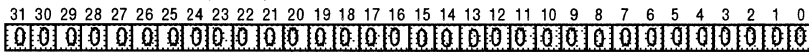
DBC Register R/W 24-26h (A4-A6)



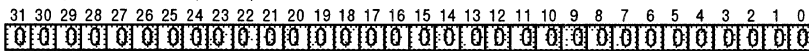
DCMD Register R/W 27h (A7)



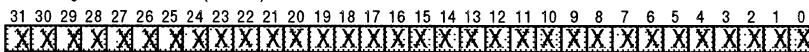
DNAD Register R/W 28-2Bh (A8-ABh)



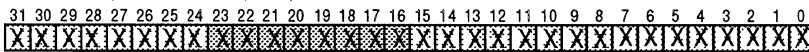
DSP Register R/W 2C-2Fh (AC-AFh)



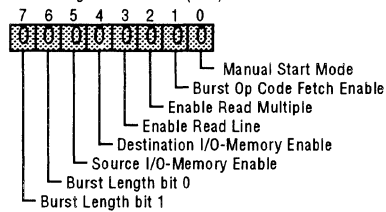
DSPS Register R/W 30-33h (B0-B3h)



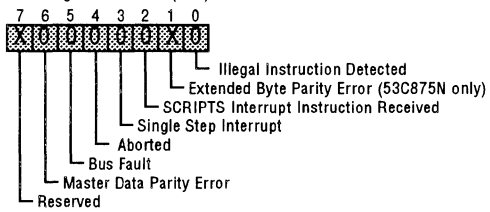
SCRATCHA Register R/W 34-37h (B4-B7)



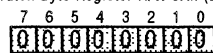
DMODE Register R/W 38h (B8h)



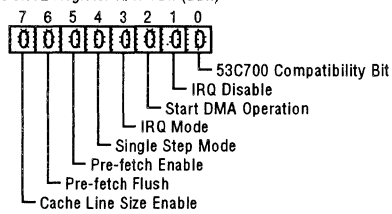
DIEN Register R/W 39h (B9h)



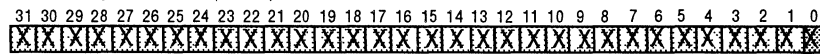
Scratch Byte Register R/W 3Ah (BAh)



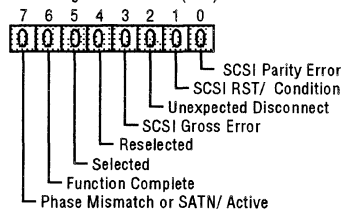
DCNTL Register R/W 3Bh (BBh)



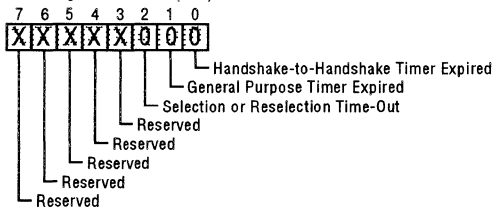
ADDER Register R 3C-3Fh (BC-BFh)



SIEN0 Register R/W 40h (C0h)

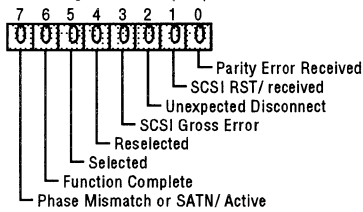


SIEN1 Register R/W 41h (C1h)

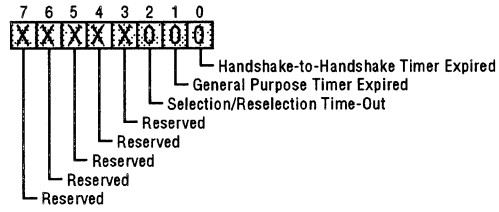


# Register Summary

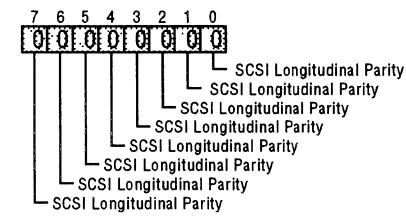
SIST0 Register R 42h (C2h)



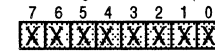
SIST1 Register R 43h (C3h)



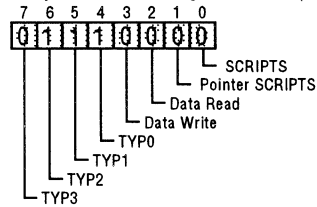
SLPAR Register R/W 44h (C4h)



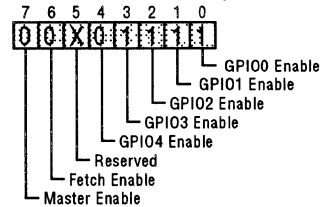
SWIDE Register R/W 45h (C5h)



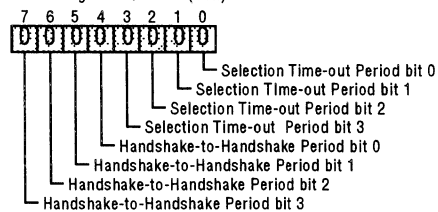
Memory Access Control Register R/W 46h (C6h)



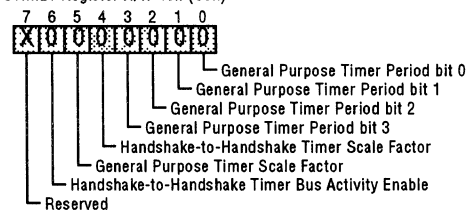
General Purpose Pin Control Register R/W 47h (C7h)



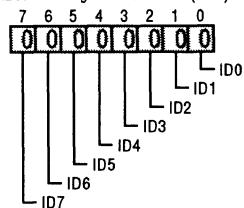
STIME0 Register R/W 48h (C8h)



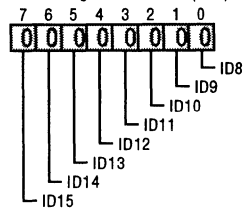
STIME1 Register R/W 49h (C9h)



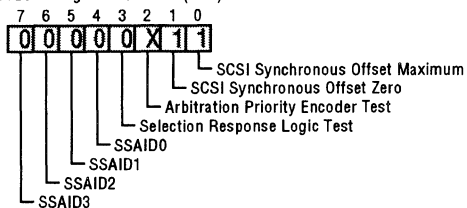
RESPID0 Register R/W 4Ah (CAh)



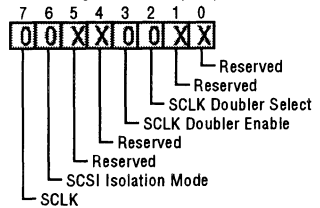
RESPID1 Register R/W 4Bh (CBh)



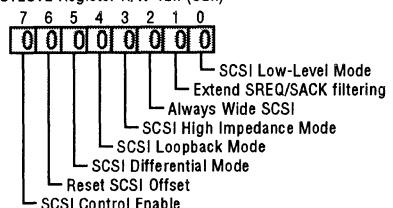
STEST0 Register R/W 4Ch (CCh)



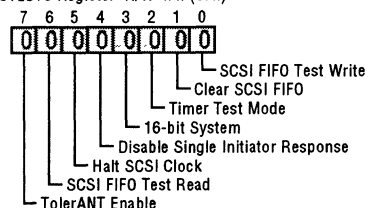
STEST1 Register R 4Dh (CDh)



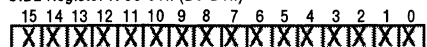
STEST2 Register R/W 4Eh (CEh)



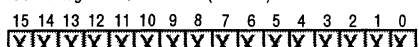
STEST3 Register R/W 4Fh (CFh)



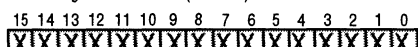
SIDL Register R 50-51h (D0-D1h)



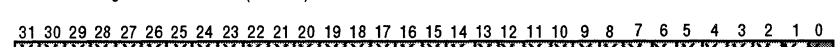
SODL Register R/W 54-55h (D4-D5h)



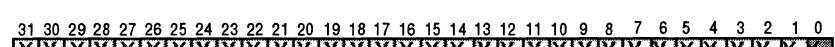
SBDL Register R 58-59h (D8-D9h)



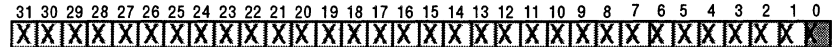
SCRATCHB Register R/W 5C-5Fh (DC-DFh)



SCRATCHC-J Registers R/W 60-70h (E0-F0h)



•  
•  
•



## Register Summary

(

(



# Appendix B Mechanical Drawings

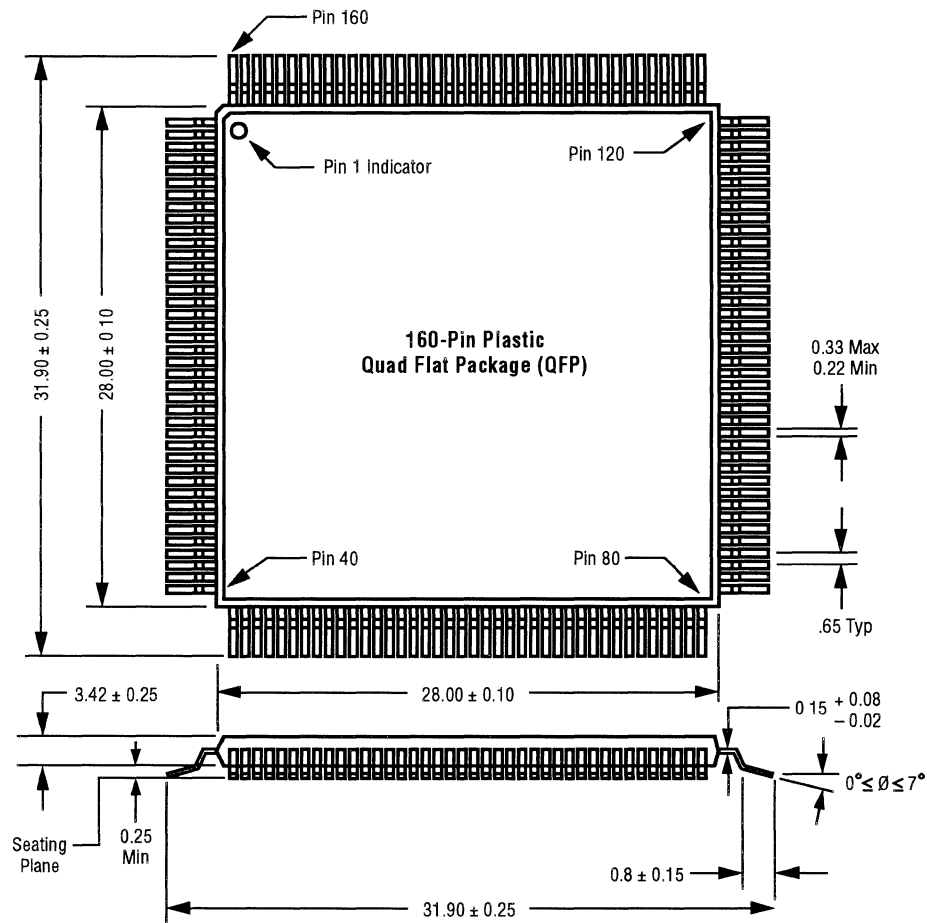


Figure B-1: SYM53C875, SYM 53C875J Mechanical Drawing

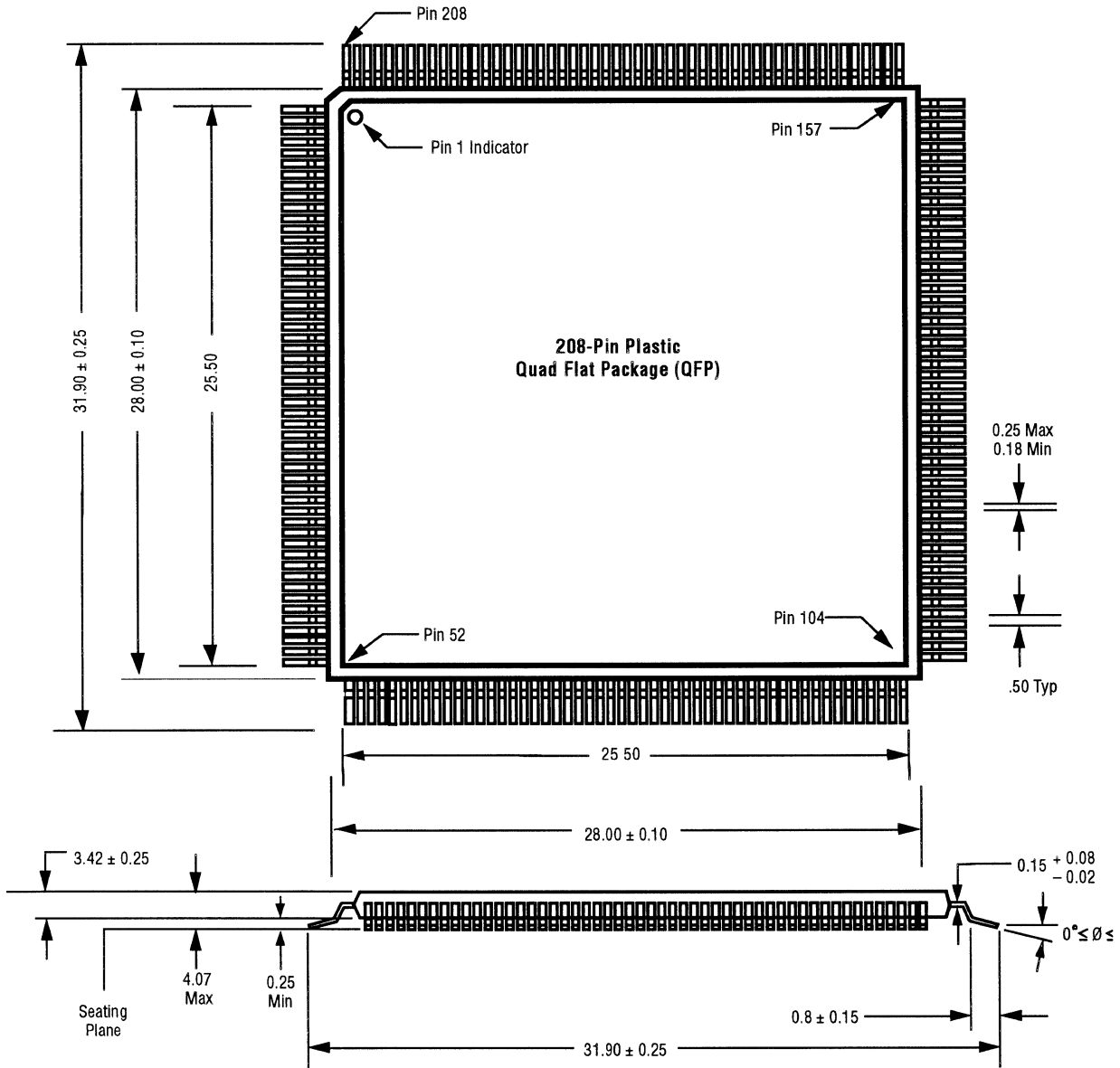
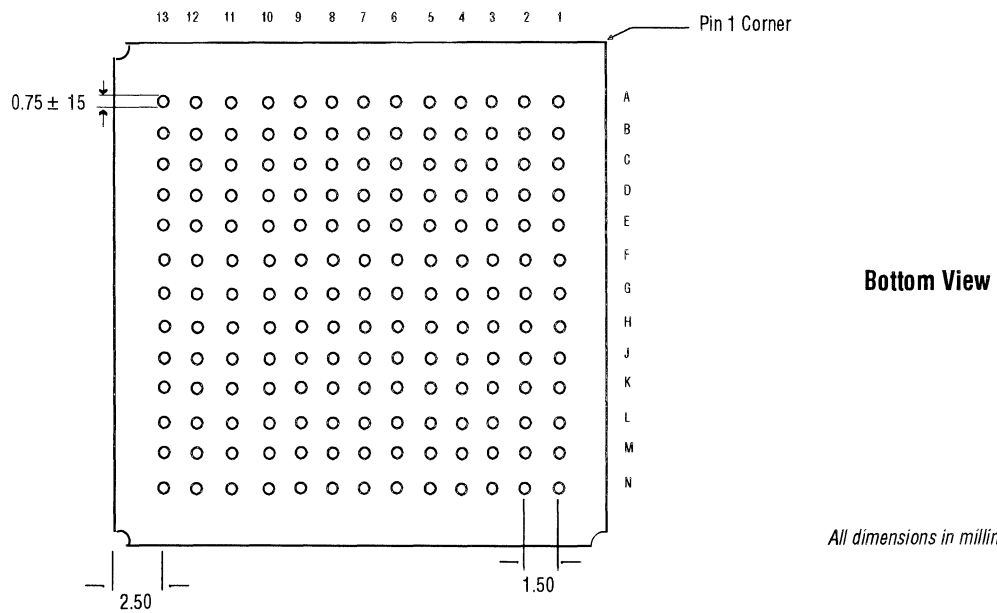
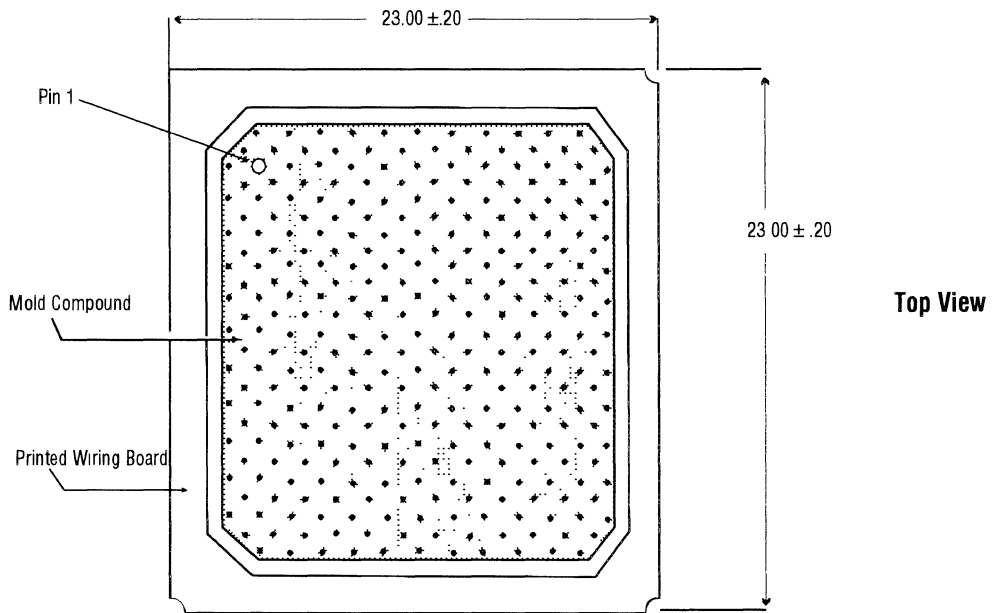
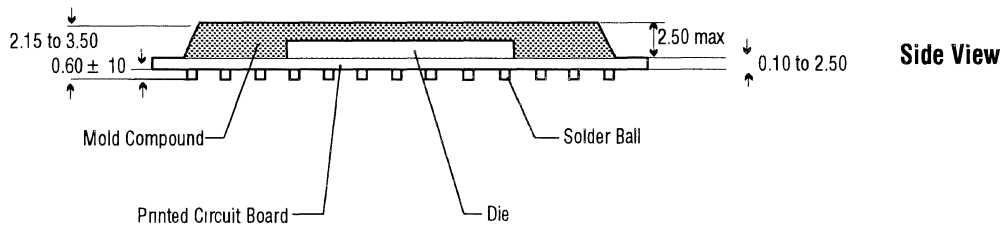


Figure B-2:SYM53C875N Mechanical Drawing



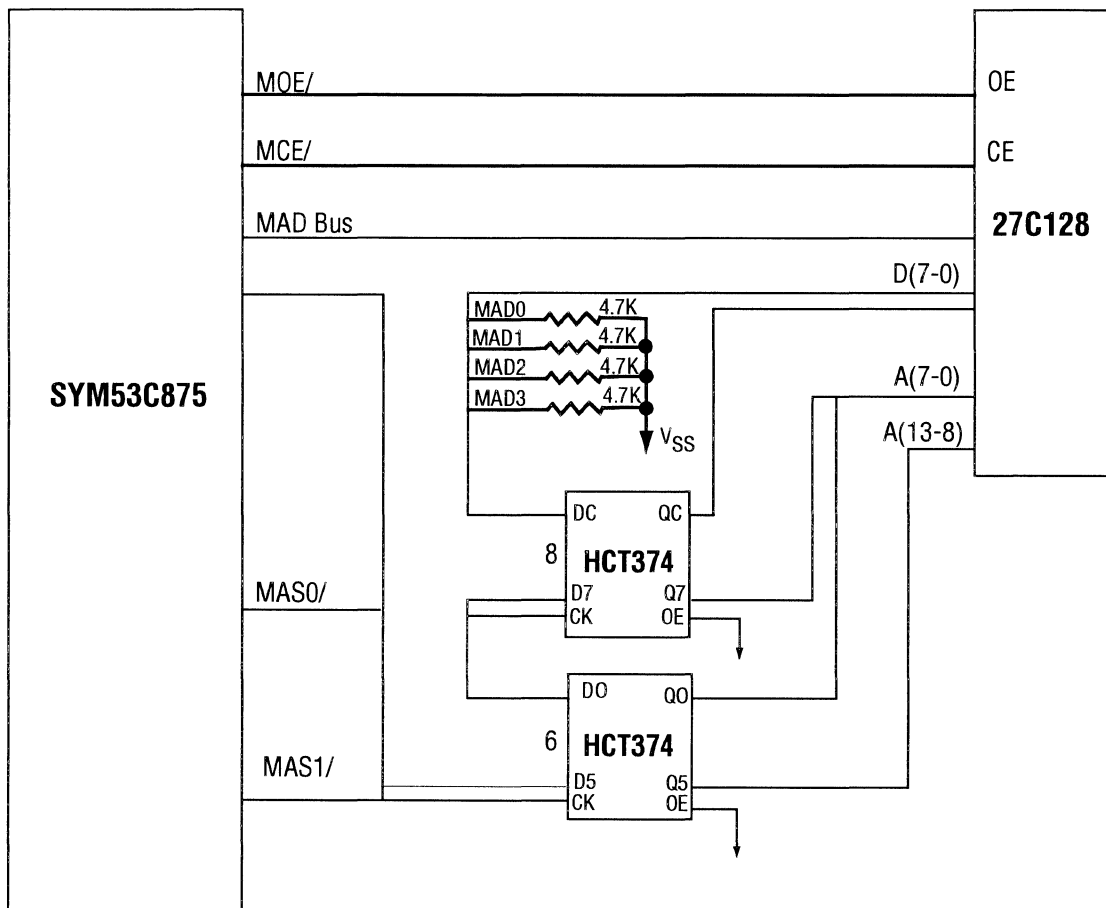
All dimensions in millimeters

**Figure B-3:SYM53C875JB (169-pin BGA) Mechanical Drawing**

(

**Appendix C**

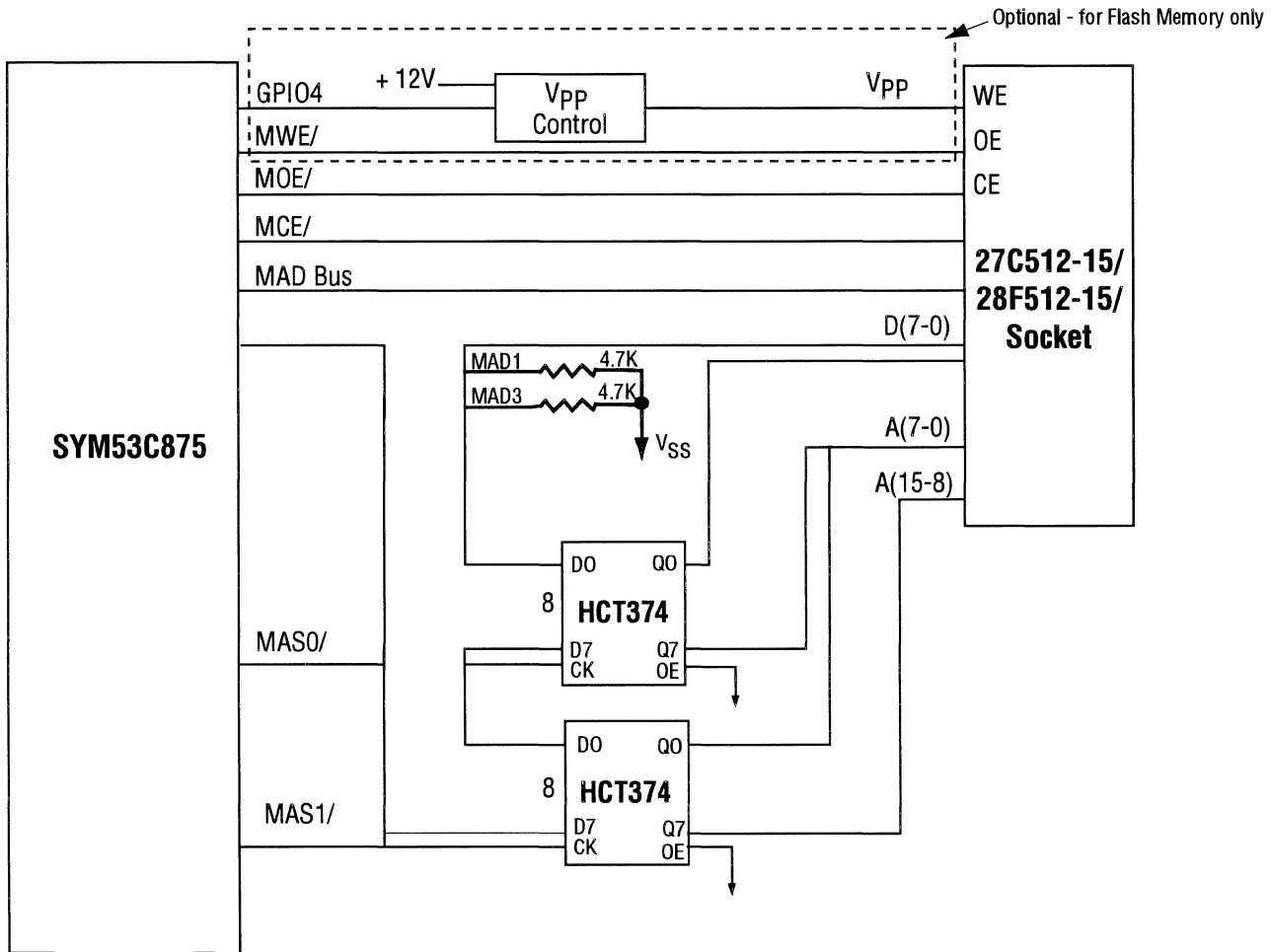
**External Memory Interface Diagram Examples**



MAD Bus Sense Logic Enabled for 16 KB of slow memory (200 ns Device @ 33MHz)

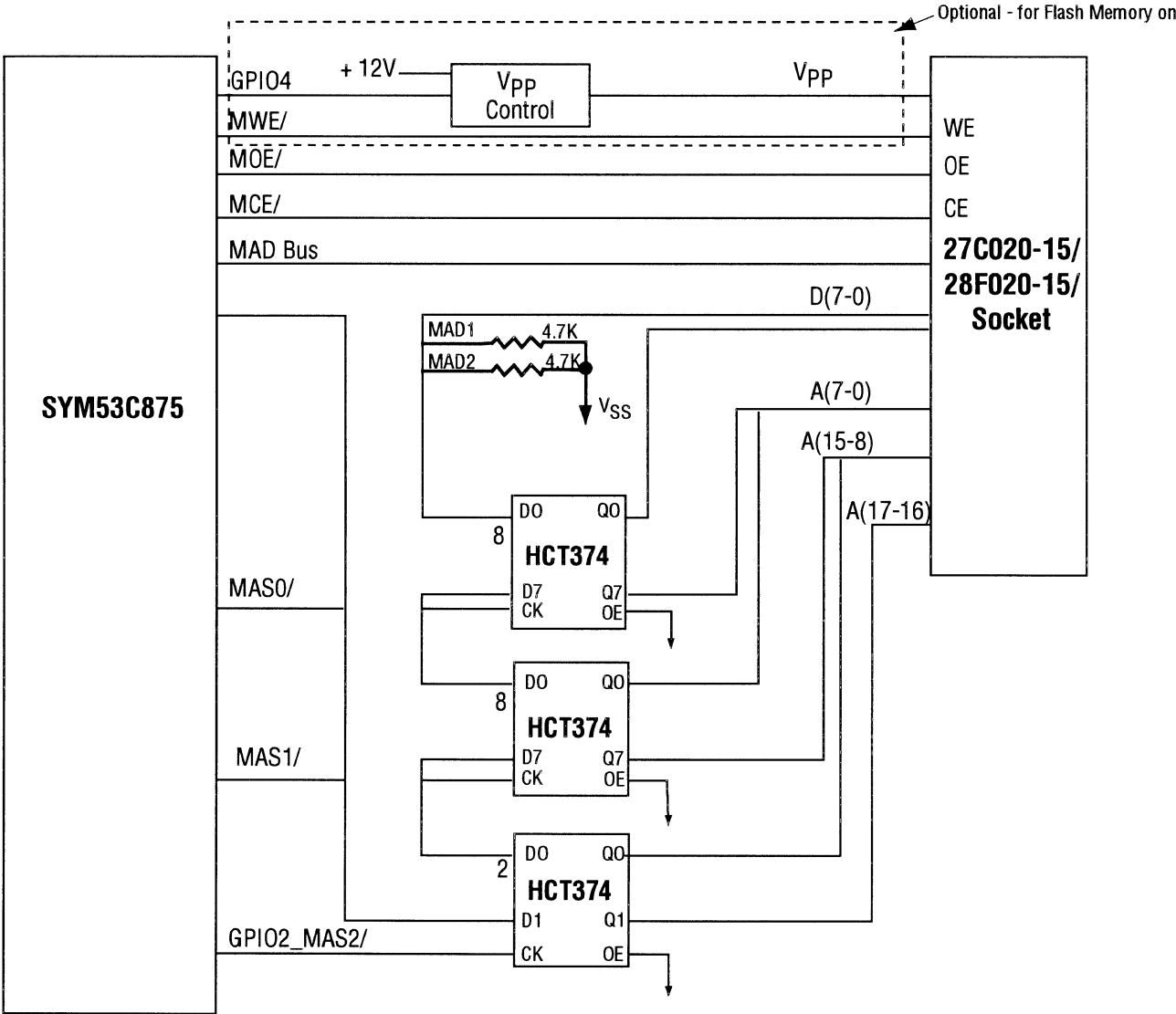
**Figure C-1:16 K Interface With 200 ns Memory**

External Memory Interface Diagram Examples



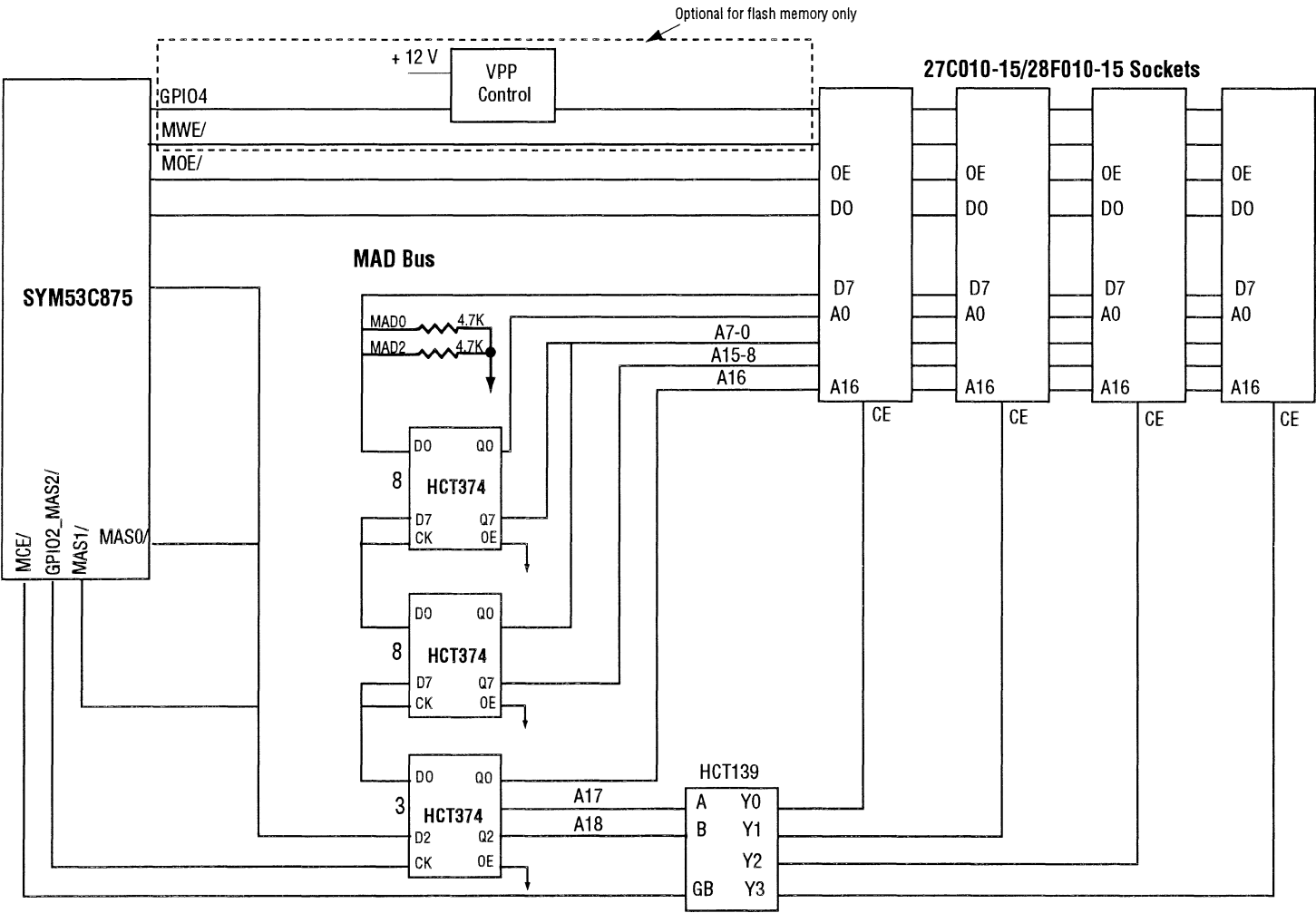
MAD Bus Sense Logic Enabled for 64 KB of fast memory (150 ns Device @ 33MHz)

Figure C-2:64 K Interface with 150 ns Memory



MAD Bus Sense Logic Enabled for 256K bytes of fast memory (150 ns Device @ 33MHz). Address latch for the two most significant bits can be a HCT74, HCT174, HCT175, etc.

Figure C-3:256 K Interface With 150 ns Memory



MAD bus sense logic enabled for 512 KB of slow memory (150 ns devices, additional decode time required for HCT 139 @ 33MHz)

Figure C-4:512 K Interface With 150 ns Memory



---

# Index

## Numerics

3.3/5 Volt PCI interface 2-5

53C700 Compatibility bit 5-38

## A

Abort Operation bit 5-23

Aborted bit 5-18, 5-37

Absolute Maximum Stress Ratings 7-1

active negation

    see TolerANT Technology

active termination 2-16

ADDER register 5-39

Adder Sum Output register 5-39

Additional Interface Pins 4-15

Address and Data Pins 4-9

Alternative Two Termination 2-17

Always Wide SCSI bit 5-51

Arbitration in Progress bit 5-20

arbitration

    Arbitration in Progress bit 5-20

    Arbitration Mode bits 5-5

    arbitration pins 4-12

    Immediate Arbitration bit 5-8

    Lost Arbitration bit 5-20

    Won Arbitration bit 5-20

Arbitration Mode bits 5-5

Arbitration Priority Encoder Test bit 5-48

Assert Even SCSI Parity bit 5-7

Assert SATN/ on Parity Error bit 5-6

Assert SCSI ACK/ Signal bit 5-16

Assert SCSI ATN/ Signal bit 5-16

Assert SCSI BSY/ Signal bit 5-16

Assert SCSI C\_D/ Signal bit 5-16

Assert SCSI Data Bus bit 5-7

Assert SCSI I\_O Signal bit 5-16

Assert SCSI MSG/ Signal bit 5-16

Assert SCSI REQ/ Signal bit 5-16

Assert SCSI RST/ Signal bit 5-7

Assert SCSI SEL/ Signal bit 5-16

## B

Big and Little Endian Support 2-6

Block Move Instructions 6-4

Burst Disable bit 5-29

Burst Length bits 5-31, 5-35

Burst Op Code Fetch Enable bit 5-36

Bus Fault bit 5-18, 5-37

Byte Empty in DMA FIFO bit 5-25

Byte Full in DMA FIFO bit 5-25

Byte Offset Counter bits 5-28, 5-31

## C

Cache Line Size Enable bit 5-38

cache mode, see PCI cache mode 3-3

chained block moves 2-24–2-26

    SODL register 2-25

    SWIDE register 2-25

    Wide SCSI Receive Bit 2-25

    Wide SCSI Send Bit 2-24

Chained Mode bit 5-9

Chip Revision Level bits 5-27

Chip Test Five register 5-30

Chip Test Four register 5-29

Chip Test One register 5-25

Chip Test Six register 5-31

- Chip Test Two register 5-26
- Chip Test Zero register 5-25
- Chip Type bits 5-44
- Clear DMA FIFO bit 5-27
- Clear SCSI FIFO bit 5-52
- Clock Address Incrementor bit 5-30
- Clock Byte Counter bit 5-30
- Clock Conversion Factor bits 5-11
- Clock Timing 7-9
- Configured as I/O bit 5-26
- Configured as Memory bit 5-26
- Connected bit 5-7, 5-24
- CTEST0 register 5-25
- CTEST1 register 5-25
- CTEST2 register 5-26
- CTEST4 register 5-29
- CTEST5 register 5-30
- CTEST6 register 5-31
- D**
- Data Acknowledge Status bit 5-26
- data path 2-11
- Data Request Status bit 5-26
- Data Structure Address register 5-22
- Data Transfer Direction bit 5-26
- DataRD bit 5-44
- DataWR bit 5-44
- DBC register 5-32
- DBLEN bit 3 5-49
- DBLSEL bit 2 5-49
- DCMD register 5-32
- DCNTL register 5-38
- Destination I/O-Memory Enable bit 5-35
- DFIFO register 5-28
- DIEN register 5-36
- DIF bit 5-50
- differential mode
  - DIFFSENS pin 4-16
  - DIFFSENSE Sense bit 5-22
  - direction control pins 4-14
  - operation 2-13
  - SCSI Differential Mode bit 5-50
  - wiring diagram 2-15
- differential wiring diagram 2-15
- DIFFSENSE Sense bit 5-22
- Disable Halt on Parity Error or ATN 5-7
- Disable Single Initiator Response bit 5-52
- DMA Byte Counter register 5-32
- DMA Command register 5-32
- DMA Control register 5-38
- DMA core 2-1
- DMA Direction bit 5-31
- DMA FIFO bits 5-31
- DMA FIFO Empty bit 5-18
- DMA FIFO register 5-28
- DMA FIFO Size bit 5-30
- DMA Interrupt Enable register 5-36
- DMA Interrupt Pending bit 5-24
- DMA Mode register 5-35
- DMA Next Address register 5-33
- DMA SCRIPTS Pointer register 5-33
- DMA SCRIPTS Pointer Save register 5-34
- DMA Status register 5-18
- DMODE register 5-35
- DNAD register 5-33
- DSA register 5-22
- DSP register 5-33
- DSPS register 5-34
- DSTAT register 5-18
- E**
- Enable Parity Checking bit 5-6
- Enable Read Line bit 5-35

Enable Read Multiple bit 5-35  
 Enable Response to Reselection bit 5-12  
 Enable Response to Selection bit 5-12  
 Enable Wide SCSI bit 5-11  
 Encoded Chip SCSI ID 5-12  
 Encoded Destination SCSI ID bit 5-17  
 Encoded Destination SCSI ID bits 5-14  
 Error Reporting Pins 4-12  
 Extend SREQ/SACK Filtering bit 5-51  
 external memory interface 2-4  
     configuration 2-5  
     diagram 1-6  
     flash ROM updates 2-4  
     GPIO4 bit 5-15  
     memory sizes supported 2-4  
     multiple byte accesses 7-11  
     pins 4-17  
     slow memory 2-5  
     system requirements 2-4  
 Extra Clock Cycle of Data Setup bit 5-7

## F

Fast-20 Enable bit 5-10  
 Fetch Enable 5-45  
 Fetch Pin Mode bit 5-27  
 FIFO Byte Control bits 5-30  
 FIFO Flags bits 5-20, 5-22  
 Flush DMA FIFO bit 5-27  
 Function Complete bit 5-39, 5-41  
 functional signal grouping 4-7

## G

general description 1-1  
 General Purpose Pin Control register 5-45  
 General Purpose register 5-15  
 General Purpose Timer Expired bit 5-40, 5-42  
 General Purpose Timer Period bits 5-47  
 General Purpose Timer Scale Factor bit 5-46  
 GPCNTL register 5-45

GPIO Enable bit 5-45  
 GPIO4-0 bits 5-15  
 GPREG register 5-15

## H

Halt SCSI Clock bit 5-51  
 Handshake to Handshake Timer Scale Factor bit 5-47  
 Handshake-to-Handshake Timer Bus Activity Enable bit 5-46  
 Handshake-to-Handshake Timer Expired bit 5-40, 5-42  
 Handshake-to-Handshake Timer Period bit 5-45  
 High Impedance Mode bit 5-29

## I

I/O Instructions 6-8  
 Illegal Instruction Detected bit 5-18  
 Immediate Arbitration bit 5-8  
 instruction prefetching 2-3  
     Pre-fetch Enable bit 5-38  
     Pre-fetch Flush bit 5-38  
     prefetch unit flushing 2-3  
 instructions  
     block move 6-4  
     I/O 6-8  
     load and store 6-23  
     memory move 6-20  
     read/write 6-13  
     transfer control 6-16  
 Interface Control Pins 4-11  
 internal RAM, see also SCRIPTS RAM 2-2  
 Interrupt on the Fly bit 5-24  
 Interrupt Output 7-10  
 Interrupt Status register 5-23  
 interrupts 2-21  
     fatal vs. non-fatal interrupts 2-22  
     halting 2-23

- IRQ Disable bit 2-22
    - masking 2-22
    - sample interrupt service routine 2-24
    - stacked interrupts 2-23
- IRQ Disable bit 5-38
- IRQ Mode bit 5-38
- ISTAT register 5-23
- J**
- JTAG support 2-6
- L**
- Last Disconnect bit 5-22
- Latched SCSI Parity bit 5-21
- Latched SCSI Parity for SD15-8 bit 5-22
- load and store instructions 6-23
  - no flush option 6-23
  - prefetch unit and Store instructions 2-3, 6-23
- Lost Arbitration bit 5-20
- M**
- MACNTL register 5-44
- Manual Start Mode bit 5-36
- Master Control for Set or Reset Pulses bit 5-30
- Master Data Parity Error bit 5-18, 5-37
- Master Enable bit 5-45
- Master Parity Error Enable bit 5-30
- Max SCSI Synchronous Offset bits 5-14
- Memory Access Control register 5-44
- Memory Move instructions 6-20
  - and SCRIPTS instruction prefetching 2-3
  - No Flush option 2-3
- Memory Read Line command 3-5
- Memory Read Multiple command 3-6
- Memory Write and Invalidate command 3-4
  - Write and Invalidate Mode bit 3-8
- N**
- No Flush Memory Move instruction 6-21

**O**

- op code fetch bursting 2-4
- Operating Conditions 7-1
- operating registers
  - Adder Sum Output 5-39
  - Chip Test Five 5-30
  - Chip Test Four 5-29
  - Chip Test One 5-25
  - Chip Test Six 5-31
  - Chip Test Three 5-27
  - Chip Test Two 5-26
  - Chip Test Zero 5-25
  - Data Structure Address 5-22
  - DMA Byte Counter 5-32
  - DMA Command 5-32
  - DMA Control 5-38
  - DMA FIFO 5-28
  - DMA Interrupt Enable 5-36
  - DMA Mode 5-35
  - DMA Next Address 5-33
  - DMA SCRIPTS Pointer 5-33
  - DMA SCRIPTS Pointer Save 5-34
  - DMA Status 5-18
  - general information 5-1
  - General Purpose 5-15
  - General Purpose Pin Control 5-45
  - Interrupt Status 5-23
  - Memory Access Control 5-44
  - register address map 5-4
  - Response ID One 5-48
  - Response ID Zero 5-48
  - Scratch Byte 5-37
  - Scratch Register A 5-34
  - Scratch Register B 5-54
  - SCSI Bus Control Lines 5-17
  - SCSI Bus Data Lines 5-54
  - SCSI Chip ID 5-12
  - SCSI Control One register 5-7

- SCSI Control Register Two 5-9
  - SCSI Control Three 5-10
  - SCSI Control Zero 5-5
  - SCSI Destination ID 5-14
  - SCSI First Byte Received 5-16
  - SCSI Input Data Latch 5-53
  - SCSI Interrupt Enable One 5-40
  - SCSI Interrupt Enable Zero 5-39
  - SCSI Interrupt Status One 5-42
  - SCSI Interrupt Status Zero 5-41
  - SCSI Longitudinal Parity 5-43
  - SCSI Output Control Latch 5-16
  - SCSI Output Data Latch 5-53
  - SCSI Selector ID 5-17
  - SCSI Status One 5-20
  - SCSI Status Two 5-21
  - SCSI Status Zero 5-19
  - SCSI Test One 5-49
  - SCSI Test Three 5-51
  - SCSI Test Two 5-50
  - SCSI Test Zero 5-48
  - SCSI Timer One 5-46
  - SCSI Timer Zero 5-45
  - SCSI Transfer 5-12
  - SCSI Wide Residue 5-44
  - Temporary Stack 5-28
- P**
- package and feature options 1-2
  - Parity 2-7-2-9
  - Parity Error bit 5-42
  - PCI cache mode 3-3
    - Cache Line Size Enable bit 5-38
    - Cache Line Size register 3-11
    - Enable Read Multiple bit 5-35
    - Memory Read Line command 3-5
    - Memory Read Multiple command 3-6
    - Memory Write and Invalidate command 3-4
      - Write and Invalidate Mode bit 3-8
    - Write and Invalidate Enable bit 5-27
  - PCI commands 3-1
  - PCI Configuration registers
    - Subsystem Data 3-12
  - PCI configuration registers 3-7-3-13
    - Base Address One (Memory) 3-12
    - Base Address Zero (I/O) 3-12
    - Cache Line Size 3-11
    - Class Code 3-11
    - Command 3-8
    - Device ID 3-8
    - Expansion ROM Base Address 3-12
    - Header Type 3-12
    - Interrupt Line 3-13
    - Interrupt Pin 3-13
    - Latency Timer 3-12
    - Max\_Lat 3-13
    - Min\_Gnt 3-13
    - Revision ID 3-11
    - Status 3-10
    - Vendor ID 3-8
  - PCI configuration space 3-1
  - PCI I/O space 3-1
  - PCI memory space 3-1
  - PCI Timings 7-34
  - Phase Mismatch bit 5-41
  - pins
    - additional interface pins 4-15
    - address and data pins 4-9
    - arbitration pins 4-12
    - error reporting pins 4-12
    - external memory interface pins 4-17
    - functional signal grouping 4-7
    - interface control pins 4-11
    - SCSI pins 4-13
    - system pins 4-8
  - Pointer SCRIPTS bit 5-44
  - Power and Ground Pins 4-5, 4-6
  - Pre-fetch Enable bit 5-38
  - Pre-fetch Flush bit 5-38

- R**
- RAM, see also SCRIPTS RAM 2-2
- Read/Write Instructions 6-13
- register addresses
- operating registers
    - 00h 5-5
    - 01h 5-7
    - 02h 5-9
    - 03h 5-10
    - 04h 5-12
    - 05h 5-12
    - 06h 5-14
    - 07h 5-15
    - 08h 5-16
    - 09h 5-16
    - 0Ah 5-17
    - 0Bh 5-17
    - 0Ch 5-18
    - 0Dh 5-19
    - 0Eh 5-20
    - 0Fh 5-21
    - 10-13h 5-22
    - 14h 5-23
    - 18h 5-25
    - 19h 5-25
    - 1Ah 5-26
    - 1Bh 5-27
    - 1C-1Fh 5-28
    - 20h 5-28
    - 21h 5-29
    - 22h 5-30
    - 23h 5-31
    - 24-26h 5-32
    - 27h 5-32
    - 28-2Bh 5-33
    - 2C-2Fh 5-33
    - 30-33h 5-34
    - 34-37h 5-34
    - 38h 5-35
    - 39h 5-36
    - 3Ah 5-37
    - 3Bh 5-38
    - 3C-3Fh 5-39
    - 40h 5-39
    - 41h 5-40
    - 42h 5-41
    - 43h 5-42
    - 44h 5-43
    - 45h 5-44
    - 46h 5-44
    - 47h 5-45
    - 48h 5-45
    - 49h 5-46
    - 4Ah 5-48
    - 4Bh 5-48
    - 4Ch 5-48
    - 4Dh 5-49
    - 4Eh 5-50
    - 4Fh 5-51
    - 50-51h 5-53
    - 54-55h 5-53
    - 58-59h 5-54
    - 5C-5Fh 5-54
    - 60-70h 5-55
  - PCI configuration registers
    - 00h 3-8
    - 02h 3-8
    - 04h 3-8
    - 06h 3-10
    - 08h 3-11
    - 09h 3-11
    - 0Ch 3-11
    - 0Dh 3-12
    - 0Eh 3-12
    - 10h 3-12
    - 14h 3-12
    - 18h 3-12
    - 30h 3-12
    - 3Ch 3-13
    - 3Dh 3-13
    - 3Eh 3-13
    - 3Fh 3-13
    - Subsystem Data 3-12
  - register bits
    - 53C700 Compatibility 5-38
    - Abort Operation 5-23
    - Aborted 5-18, 5-37

- Always Wide SCSI 5-51
- Arbitration in Progress 5-20
- Arbitration Mode 5-5
- Arbitration Priority Encoder Test 5-48
- Assert Even SCSI Parity (force bad parity)) 5-7
- Assert SATN/ on Parity Error 5-6
- Assert SCSI ACK/ Signal 5-16
- Assert SCSI ATN/ Signal 5-16
- Assert SCSI BSY/ Signal 5-16
- Assert SCSI C\_D/ Signal 5-16
- Assert SCSI Data Bus 5-7
- Assert SCSI I\_O/ Signal 5-16
- Assert SCSI MSG/ Signal 5-16
- Assert SCSI REQ/ Signal) 5-16
- Assert SCSI RST/ Signal 5-7
- Assert SCSI SEL/ Signal 5-16
- Burst Disable 5-29
- Burst Length 5-31, 5-35
- Burst Op Code Fetch Enable 5-36
- Bus Fault 5-18, 5-37
- Byte Empty in DMA FIFO 5-25
- Byte Full in DMA FIFO 5-25
- Byte Offset Counter 5-28, 5-31
- Cache Line Size Enable 5-38
- Chained Mode 5-9
- Chip Revision Level 5-27
- Chip Type 5-44
- Clear DMA FIFO 5-27
- Clear SCSI FIFO 5-52
- Clock Address Incrementor 5-30
- Clock Byte Counter 5-30
- Clock Conversion Factor 5-11
- Configured as I/O 5-26
- Configured as Memory 5-26
- Connected 5-7, 5-24
- Data Acknowledge Status 5-26
- Data Request Status 5-26
- Data Transfer Direction 5-26
- DataRD 5-44
- DataWR 5-44
- Destination I/O-Memory Enable 5-35
- DIFFSENSE Sense 5-22
- Disable Halt on Parity Error or ATN 5-7
- Disable Single Initiator Response 5-52
- DMA Direction 5-31
- DMA FIFO 5-31
- DMA FIFO Empty 5-18
- DMA FIFO Size 5-30
- DMA Interrupt Pending 5-24
- Enable Parity Checking 5-6
- Enable Read Line 5-35
- Enable Read Multiple 5-35
- Enable Response to Reselection 5-12
- Enable Response to Selection 5-12
- Enable Wide SCSI 5-11
- Encoded Chip SCSI ID, bits 3-0 5-12
- Encoded Destination SCSI ID 5-14, 5-17
- Extend SREQ/SACK Filtering 5-51
- Extra clock cycle of data setup 5-7
- Fast-20 Enable 5-10
- Fetch Enable 5-45
- Fetch Pin Mode 5-27
- FIFO Byte Control 5-30
- FIFO Flags 5-20, 5-22
- Flush DMA FIFO 5-27
- Function Complete 5-39, 5-41
- General Purpose Timer Expired 5-40, 5-42
- General Purpose Timer Period 5-47
- General Purpose Timer Scale Factor 5-46
- GPIO Enable 5-45
- GPIO4-0 5-15
- Halt SCSI Clock 5-51
- Handshake to Handshake Timer Scale Factor 5-47
- Handshake-to-Handshake Timer Bus Activity Enable 5-46
- Handshake-to-Handshake Timer Expired 5-40, 5-42

- Handshake-to-Handshake Timer Period 5-45
- High Impedance Mode 5-29
- Illegal Instruction Detected 5-18
- Illegal instruction detected 5-37
- Immediate Arbitration 5-8
- Interrupt on the Fly 5-24
- IRQ Disable 5-38
- IRQ Mode 5-38
- Last Disconnect 5-22
- Latched SCSI Parity 5-21
- Latched SCSI Parity for SD15-8 5-22
- Lost Arbitration 5-20
- Manual Start Mode 5-36
- Master Control for Set or Reset Pulses 5-30
- Master Data Parity Error 5-18, 5-37
- Master Enable 5-45
- Master Parity Error Enable 5-30
- Max SCSI Synchronous Offset 5-14
- Parity Error 5-42
- Phase Mismatch 5-41
- Pointer SCRIPTS 5-44
- Pre-fetch Enable 5-38
- Pre-fetch Flush 5-38
- Reselected 5-39, 5-41
- Reset SCSI Offset 5-50
- SACK/ Status 5-17
- SATN/ Status 5-17
- SBSY/ Status 5-17
- SC\_D/ Status 5-17
- SCLK 5-49
- SCLK Doubler Enable bit 5-49
- SCLK Doubler Select bit 5-49
- SCRATCHA/B Operation 5-26
- SCRIPTS 5-44
- SCRIPTS Interrupt Instruction Received 5-18
- SCRIPTS interrupt instruction received 5-37
- SCSI C\_D/ Signal 5-21
- SCSI Control Enable 5-50
- SCSI Data High Impedance 5-29
- SCSI Differential Mode 5-50
- SCSI Disconnect Unexpected 5-9
- SCSI FIFO Test Read 5-51
- SCSI FIFO Test Write 5-52
- SCSI Gross Error 5-40, 5-41
- SCSI High-Impedance Mode 5-51
- SCSI I\_O/ Signal 5-21
- SCSI Interrupt Pending 5-24
- SCSI Isolation Mode 5-49
- SCSI Loopback Mode 5-50
- SCSI Low level Mode 5-51
- SCSI MSG/ Signal 5-21
- SCSI Parity Error 5-40
- SCSI Phase Mismatch - Initiator Mode 5-39
- SCSI Reset Condition 5-40
- SCSI RST/ Received 5-42
- SCSI RST/ Signal 5-20
- SCSI SDP0/ Parity Signal 5-20
- SCSI SDP1 Signal 5-22
- SCSI Selected As ID 5-48
- SCSI Synchronous Offset Maximum 5-49
- SCSI Synchronous Offset Zero 5-49
- SCSI Synchronous Transfer Period 5-12
- SCSI True End of Process 5-26
- SCSI Valid 5-17
- Select with SATN/ on a Start Sequence 5-6
- Selected 5-39, 5-41
- Selection or Reselection Time-out 5-40, 5-42
- Selection Response Logic Test 5-48
- Selection Time-Out 5-46
- Semaphore 5-23
- Shadow Register Test Mode 5-29
- SI\_O/ Status 5-17
- SIDL Least Significant Byte Full 5-19
- SIDL Most Significant Byte Full 5-21
- Signal Process 5-23, 5-26
- Single Step Interrupt 5-18
- Single-step Interrupt 5-37
- Single-step Mode 5-38
- SLPAR High Byte Enable 5-9



- SLPAR Mode 5-9
- SMSG/ Status 5-17
- SODL Least Significant Byte Full 5-19
- SODL Most Significant Byte Full 5-21
- SODR Least Significant Byte Full 5-19
- SODR Most Significant Byte Full 5-21
- Software Reset 5-23
- Source I/O-Memory Enable 5-35
- SREQ/ Status 5-17
- SSEL/ Status 5-17
- Start DMA Operation 5-38
- Start SCSI Transfer 5-8
- Start Sequence 5-5
- Synchronous Clock Conversion Factor 5-10
- Target Mode 5-6
- Timer Test Mode 5-52
- TolerANT Enable 5-51
- Unexpected Disconnect 5-40, 5-42
- Vendor Unique Enhancements bit 0 5-9
- Vendor Unique Enhancements bit 1 5-10
- Wide SCSI Receive 5-10
- Wide SCSI Send 5-9
- Won Arbitration 5-20
- Write and Invalidate Enable 5-27
- Registers
  - see operating registers
- Reselected bit 5-39, 5-41
- Reset Input 7-10
- Reset SCSI Offset bit 5-50
- RESPID0 register 5-48
- RESPID1 register 5-48
- Response ID One register 5-48
- Response ID Zero register 5-48
- Revision Level bits 5-27
- S**
- SACK/ Status bit 5-17
- SATN/ Active 5-41
- SATN/ Active bit 5-41
- SATN/ Status bit 5-17
- SBCL register 5-17
- SBDL register 5-54
- SBR Register 5-37
- SBSY Status bit 5-17
- SC\_D/ Status bit 5-17
- SCID register 5-12
- SCLK bit 5-49
- SCNTL0 register 5-5
- SCNTL1 register 5-7
- SCNTL2 register 5-9
- SCNTL3 register 5-10
- Scratch Register 5-37
- SCRATCHA register 5-34
- SCRATCHA/B Operation bit 5-26
- SCRATCHB register 5-54
- SCRIPTS bit 5-44
- SCRIPTS Interrupt Instruction Received bit 5-18
- SCRIPTS processor 2-1
  - instruction prefetching 2-3
  - internal RAM for instruction storage 2-2
  - performance 2-1
- SCRIPTS RAM 2-2
  - SCRATCHA/B Operation bit 5-26
- SCSI
  - core 2-1
  - differential mode 2-13
  - pins 4-13
  - termination 2-16
  - timings 7-36
  - TolerANT technology 1-3
- SCSI ATN Condition - Target Mode 5-39
- SCSI ATN Condition bit 5-39
- SCSI Bus Control Lines register 5-17
- SCSI Bus Data Lines register 5-54

- SCSI bus interface 2-13–2-18
- SCSI C\_D/ Signal bit 5-21
- SCSI Chip ID register 5-12
- SCSI CLK frequency
  - doubling 5-50
- SCSI Clock Doubler
  - using 2-3
- SCSI Control Enable bit 5-50
- SCSI Control One register 5-7
- SCSI Control Three register 5-10
- SCSI Control Two register 5-9
- SCSI Control Zero register 5-5
- SCSI core 2-1
- SCSI Data High impedance bit 5-29
- SCSI Destination ID register 5-14
- SCSI Device Management System (SDMS) 2-2
- SCSI Differential Mode bit 5-50
- SCSI Disconnect Unexpected bit 5-9
- SCSI FIFO Test Read bit 5-51
- SCSI FIFO Test Write bit 5-52
- SCSI First Byte Received register 5-16
- SCSI Gross Error bit 5-40, 5-41
- SCSI High-Impedance Mode bit 5-51
- SCSI I\_O/ Signal bit 5-21
- SCSI Input Data Latch register 5-53
- SCSI instructions
  - block move 6-4
  - I/O 6-8
  - load/store 6-23
  - memory move 6-20
  - read/write 6-13
  - transfer control 6-16
- SCSI Interrupt Enable One register 5-40
- SCSI Interrupt Enable Zero register 5-39
- SCSI Interrupt Pending bit 5-24
- SCSI Interrupt Status One register 5-42
- SCSI Interrupt Status Zero register 5-41
- SCSI Isolation Mode bits 5-49
- SCSI Longitudinal Parity register 5-43
- SCSI Loopback Mode bit 5-50
- SCSI Low Level Mode bit 5-51
- SCSI MSG/ Signal bit 5-21
- SCSI Output Control Latch register 5-16
- SCSI Output Data Latch register 5-53
- SCSI Parity Error bit 5-40
- SCSI Phase Mismatch bit 5-39
- SCSI Reset Condition bit 5-40
- SCSI RST/ Received bit 5-42
- SCSI RST/ Signal bit 5-20
- SCSI SCRIPTS operation 6-1
  - sample instruction 6-2
- SCSI SDP0/ Parity Signal bit 5-20
- SCSI SDP1 Signal bit 5-22
- SCSI Selected As ID bits 5-48
- SCSI Selector ID register 5-17
- SCSI Status One register 5-20
- SCSI Status Two register 5-21
- SCSI Status Zero register 5-19
- SCSI Synchronous Offset Maximum 5-49
- SCSI Synchronous Offset Zero bit 5-49
- SCSI Test One register 5-49
- SCSI Test Three register 5-51
- SCSI Test Two register 5-50
- SCSI Test Zero register 5-48
- SCSI Timer One register 5-46
- SCSI Timer Zero register 5-45
- SCSI timings 7-36–7-40, ??–7-40, ??–7-41
- SCSI Transfer register 5-12
- SCSI True End of Process bit 5-26
- SCSI Valid bit 5-17
- SCSI Wide Residue register 5-44
- SDID register 5-14

- Select with SATN/ on a Start Sequence bit 5-6
- Selected bit 5-39, 5-41
- Selection or Reselection Time-out bit 5-40, 5-42
- Selection Response Logic Test bits 5-48
- Semaphore bit 5-23
- SFBR register 5-16
- Shadow Register Test Mode bit 5-29
- SI\_O/ Status bit 5-17
- SIDL Least Significant Byte Full bit 5-19
- SIDL Most Significant Byte Full bit 5-21
- SIDL register 5-53
- SIEN0 register 5-39
- SIEN1 register 5-40
- SIGP bit 5-23, 5-26
- Single Step Interrupt bit 5-18
- single-ended operation 2-13
- Single-step Interrupt bit 5-37
- Single-step Mode bit 5-38
- SIST0 register 5-41
- SIST1 register 5-42
- SLPAR High Byte Enable 5-9
- SLPAR Mode bit 5-9
- SLPAR register 5-43
- SMSG/ Status bit 5-17
- SOCL Least Significant Byte Full bit 5-19
- SOCL register 5-16
- SODL Most Significant Byte Full bit 5-21
- SODL register 5-53
- SODR Least Significant Byte Full bit 5-19
- SODR Most Significant Byte Full bit 5-21
- Software Reset bit 5-23
- Source I/O-Memory Enable bit 5-35
- SREQ/ Status bit 5-17
- SSEL/ Status bit 5-17
- SSID register 5-17
- SSTAT0 register 5-19
- SSTAT1 register 5-20
- SSTAT2 register 5-21
- Stacked Interrupts 2-23
- Start DMA Operation bit 5-38
- Start SCSI Transfer 5-8
- Start Sequence bit 5-5
- STEST0 register 5-48
- STEST1 register 5-49
- STEST2 register 5-50
- STEST3 register 5-51
- STIME0 register 5-45
- STIME1 register 5-46
- SWIDE register 5-44
- SXFER register 5-12
- Synchronous Clock Conversion Factor bits 5-10
- Synchronous data transfer rates 2-18
- Synchronous Transfer Period bits 5-12
- System Pins 4-8
- T**
- Target Mode bit 5-6
- TEMP register 5-28
- Temporary register 5-28
- termination 2-16
- Timer Test Mode bit 5-52
- timings
  - clock 7-9
  - interrupt output 7-10
  - PCI 7-34
  - reset input 7-10
  - SCSI 7-36
  - Ultra SCSI 7-40, 7-41
- TolerANT Enable bit 5-51
- TolerANT Technology 1-3
  - benefits 1-3

electrical characteristics 7-6

Extend SREQ/SACK Filtering bit 5-51

TolerANT Enable bit 5-51

Transfer Control Instructions 6-16

and SCRIPTS instruction prefetching 2-4

transfer rate 1-3

synchronous 2-18

## U

Ultra SCSI

benefits 1-2

Clock Conversion Factor bits 5-11

designing an Ultra SCSI system 2-2

doubling the SCSI CLK frequency 5-50

Fast-20 Enable bit 5-10

SCLK Doubler Enable bit 5-49

SCLK Doubler Select bit 5-49

Synchronous Clock Conversion Factor bits 5-10

Synchronous Transfer Period bits 5-12  
timings 7-40, 7-41

using the SCSI clock doubler 2-3

Unexpected Disconnect bit 5-40, 5-42

## V

Vendor Unique Enhancements bit 0 5-9

Vendor Unique Enhancements bit 1 5-10

## W

WATN/ bit 5-6

what is covered in this manual 1-1

wide SCSI

Always Wide SCSI bit 5-51

chained block moves 2-24

Chained Mode bit 5-9

Enable Wide SCSI bit 5-11

SWIDE register 5-44

Wide SCSI Receive bit 5-10

Wide SCSI Send bit 5-9

Wide SCSI Receive bit 5-10

Wide SCSI Send bit 5-9

Won Arbitration bit 5-20

Write and Invalidate Enable bit 5-27

---

# Symbios Logic Sales Locations

For literature on any Symbios Logic product or service,  
call our hotline toll-free  
1-800-856-3093

## Plant Locations

### Symbios Logic

2001 Danfield Court  
Fort Collins, CO 80525  
(970) 226-9500

### Symbios Logic

1635 Aeroplaza Drive  
Colorado Springs, CO 80916  
(719) 573-3200

### Symbios Logic

3718 N. Rock Road  
Wichita, KS 67226-1397  
U.S. RAID Hotline: 1-800-440-5606  
International RAID Hotline: 316-636-8652

## North American Sales Locations

### Western Sales Area

1731 Technology Drive, Suite 610  
San Jose, CA 95110  
(408) 441-1080

3300 Irvine Avenue, Suite 255  
Newport Beach, CA 92660  
(714) 474-7095

## Eastern Sales Area

8000 Townline Avenue, Suite 209  
Bloomington, MN 55438-1000  
(612) 941-7075

17304 Preston Road, Suite 635  
Dallas, TX 75252  
(214) 733-3594

92 Montvale Avenue, Suite 3500  
Stoneham, MA 02180-3623  
(617) 438-0043

30 Mansell Court, Suite 220  
Roswell, GA 30076  
(404) 641-8001

## International Sales Locations

### European Sales Headquarters

Westendstrasse 193\II  
80686 Muenchen  
Germany  
011-49-89-547470-0

### Asia/Pacific Sales Headquarters

37th Floor, Room 3702, Lippo Tower  
Lippo Centre, 89 Queensway  
Hong Kong  
011-852-253-00727

T16962I  
0396-15MD

Symbios Logic is the name and  
mark of  
Symbios Logic Inc.  
©1995,1996 Symbios Logic Inc.  
Printed in U.S.A.