User Manual **CC93**
Hard Disk/Floppy Disk
Module

VERSION 2.2 **December 1991**

Documentation history

date	MAN Rev.	PCB Rev.	FIRM Rev.	change / description
06/01/86	1.0	1.0	0.0	first release
14/11/86	2.0	1.0	0.1	CCS implemented
21/11/86	2.0	1.1	0.2	schem./softw. change
10/12/87	2.1	1.1	0.3	added parity jumper
05/12/91	2.2	1.2	2.3	software and hardware changes

Copyright

Disclaimer

The information in this document has been carefully checked and is believed to be entirely reliable. However, no responsibility is assumed for inaccuracies. Compcontrol Int B.V. makes no representations or warranties with respect to the contents hereof and specifically disclaims any implied warranties of merchantability or fitness for any particular purpose. Furthermore, Compcontrol Int B.V. reserves the right to make changes to any product herein to improve reliability, function or design, without obligation of Compcontrol Int B.V. to notify any person of such revision or changes. Compcontrol Int B.V. does not assume any liability arising out of applications or use of any product or circuit described herein; neither does it convey any license under its patent rights nor the rights of others.

CC93 MODULE

TABLE OF CONTENTS

TABLE OF CONTENTS (cont'd)

TABLE OF CONTENTS (cont'd)

CHAPTER 1

General Information

1.1 Introduction

This manual provides information for use, installation instructions and theory of operation the CC93 Mass Storage Subsystem.

1.2 Features of the Module

The CC93 module integrates floppy, hard, tape, or optical disk drives into compact mass-storage subsystems. These subsystems have the standard VMEbus double Eurocard form factor, and take only 3 slots in a VMEbus rack. The modules do not occupy any address in the VMEbus address space. Only power is obtained from the VMEbus backplane. All subsystems have interfaces compatible with the Small Computer System Interface (SCSI) standard, and use the Common Command Set (CCS) as the standard software interface. Connection to the SCSI bus is accomplished through the P2 connector.

1.3 General Description

The new 3.5" form factor devices (floppy disk, hard disk, tape unit) supplied with an SCSI interface make it possible to design VMEbus modules which contain two of these devices (floppy/hard, tape/hard). However, floppy disk drives are not yet available which feature the SCSI bus interface. Therefore, the CC93 module is developed to overcome this problem by integrating the SCSI/floppy interface on the CC93 board, instead of on the floppy drive.
The CC93 Floppy/Hard Disk Drive module offers a hard disk drive and a floppy disk drive, both with a full implementation of the Small Computer System Interface (SCSI) on a single VMEbus compatible module. Both drives have their own address on the SCSI bus and therefor operate independently of each other.

1.4 Specifications

1.4.1 VMEbus

Power requirements:
+ 5Vdc: 1.6 Amp + power for drive(s)
+12Vdc: power for drive(s)

Physical configuration:
Double Height, Triple Width Board
Floppy Disk Drive(s) accessible from front
SCSI on P2 connector

Environmental conditions:
operational temperature: 10 - 50 degrees C
maximum operating humidity: 20 - 80 %

1.4.2 Host Interface

Floppy disk drive:

Type	SCSI
Termination	socketed 220/330 resistor packs
Addressing	jumperable, 0 to 7

Hard disk drive:

Type	SCSI
Termination	\ see Hard Disk Drive manual
Addressing	/ (supplied separately)

Tape drive:

Type	SCSI
Termination	\ see Tape Drive manual
Addressing	/ (supplied separately)

1.4.3 Floppy Drive Interface

Drive type	3.5-inch (5.25 and 8.00 inch compatible)
Encoding method	FM or MFM
Cylinders per drive	programmable
Bytes per sector	128, 256, 512, 1024, 2048, 4096 or 8192
Sectors per track	programmable
Head select	2
Drive select	4
Stepping rate	programmable (1-16 ms)
Head load timing	programmable (2-254 ms)
Head unload timing	programmable (16-240 ms)
Motor ON/OFF timing	programmable (0-65535 ms)
Sectoring	soft

CHAPTER 2

Contents of this Manual

The CC93 disk drive module can be considered to consist of two parts.

The floppy disk drive:
the drive is interfaced to the SCSI bus via the electronics and software supplied by the CC93. This manual will discuss the hardware and software supplied by the CC93 to interface a floppy disk drive to the SCSI bus.

The hard disk or tape drive:
the drive has a SCSI interface integral with the drive itself. It connects directly to the SCSI bus. The description of the software interface of the drive and the setting of the address jumpers is dependent of the type of drive used. A manual for the hard disk or tape drive is therefor supplied separately.

Chapter 3: gives information about jumpers and connectors to install the CC93 module in a VMEbus card cage.
Chapter 4: contains the physical definitions of the SCSI and their implementation on the CC93. The signals, terminators, and bus timing are covered.
Chapter 5: contains the logical definitions of the SCSI and their implementation on the CC93. Bus phases, asynchronous bus conditions, phase sequences, and the message system are covered.
Chapter 6: defines the SCSI command structure and gives a detailed description of the commands implemented by the CC93 module.
Chapter 7: contains several examples of how to initialize the CC93 for the different formats commonly used.
Chapter 8, 9, 10, and 11: provide information about the hardware and the internal workings of the CC93. These chapters may be skipped by users who are not planning to program the CC93.

For users familiar with SCSI devices it is not necessary to read chapter 4 and 5, although they do contain specific information about the CC93.

This Page Intentionally Left Blank

CHAPTER 3

Installation Instructions VMEbus

3.1 Introduction

This chapter provides the preparation and installation instructions for the CC93 disk drive board.

3.2 Use of P1 and P2

The connectors P1 and P2 to the VMEbus serve several functions:
1) they give mechanical strength to the board
2) they give access to the power supply pins of the VMEbus backplane
3) connection to the SCSI bus is accomplished through the P2 connector

For modules which draw more current from the +12Vdc or +5Vdc than is allowed via the P1 pins, there are extra pins defined on the P2 connector. These pins may be externally connected to the main power supply voltages used on the VME bus.

Table 3-1
VMEbus Connectors P1 and P2

P1 a	P1 b	P1 c		P2 a	P2 b	P2 c
			1	-DB0	+5V	GND
			2	-DB1	GND	GND
			3	-DB2		GND
			4	-DB3		GND
			5	-DB4		GND
			6	-DB5		GND
			7	-DB6		GND
			8	-DB7		GND
GND		GND	9	-DBP		GND
			10	GND		GND
GND			11	GND		GND
			12	GND	GND	GND
			13	TPWR	+5V	
			14	GND		GND
GND			15	GND		GND
			16	-ATN		GND
GND			17	GND		GND
			18	-BSY		GND
GND			19	-ACK		GND
	GND		20	-RST		GND
			21	-MSG		GND
			22	-SEL	GND	GND
	GND		23	-C/D		GND
			24	-REQ		GND
			25	-I/O		GND
			26			
			27	+5V		GND
			28	+5V		GND
			29	+5V		GND
			30	+12V		GND
		+12V	31	+12V	GND	GND
+5V	+5V	+5V	32	+12V	+5V	GND

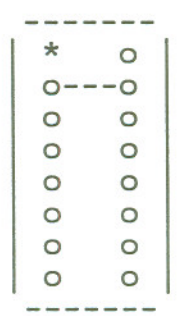
3.3 Jumper Settings

Jumper settings discussed in the next section are illustrated as seen from the component side with the VMEbus connectors downwards. For location see appendix C. Pin number one is identified as '*', rest of pins are identified as 'o'.

JB 8 SCSI interface ID number Floppy

Before connecting the board to a SCSI bus an ID number must be selected for the Floppy controller. For the floppy drive JB8 must be used. Only one jumper may be installed.

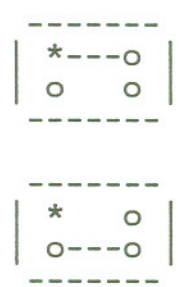
JB8



JB 11 SCSI parity enable

While the use of parity on the SCSI bus is a system option (i.e., a system is configured so that all SCSI devices on a bus generate parity and have parity detection enabled, or all SCSI devices have parity detection disabled or not implemented) this jumper is provided to disable the parity check of this module. The CC93 will however always generate odd parity on all SCSI bus transfers regardless the setting of this jumper. The setting of the jumper is sampled only once when the board is reset. Only one jumper may be installed.

JB11



JB 12 Termination Power (TRMPWR)

When this jumper is removed, the termination power (TRMPWR) for the resistor networks must be supplied via the SCSI bus. When the jumper is installed the resistor networks use the +5V of the board and TRMPWR will be fed to the SCSI bus (using a diode).

JB 13 SCSI interface ID number Hard Disk

Before connecting the board to a SCSI bus an ID number must be selected. For the Hard disk drive JB13 must be used.

JB13

This jumper block is not always usable, depending on the type of Hard Disk is mounted. For the function of the jumper block please consult the Hard Disk manual (supplied seperately).

CHAPTER 4

Interface Characteristics

4.1 Electrical Description

NOTE: For these measurements, SCSI bus termination is assumed to be external to the SCSI device. An SCSI device may have the provision for allowing optional internal termination.

4.1.1 Single-ended Alternative All assigned signals shall be terminated with 220 ohms to +5V and 330 ohms to ground at each end of the cable. (See figure 4-1). All signals shall use open-collector or three- state drivers.

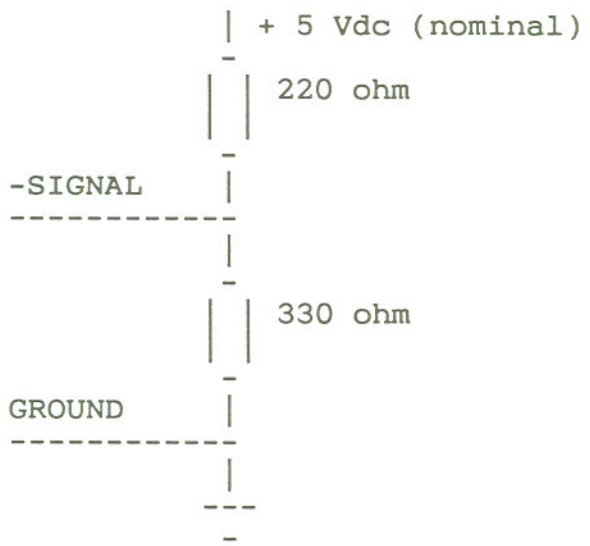


figure 4-1
termination for Single-Ended Devices

4.1.1.1 Output Characteristics Each signal driven by an SCSI device shall have the following output characteristics when measured at the SCSI device's connector:

signal assertion	0.0 Vdc to 0.4 Vdc
minimum driver output capability	48 mA sinking at 0.5 Vdc
signal negation	2.5 Vdc to 5.25 Vdc

4.1.1.2 Input Characteristics Each signal received by an SCSI device shall have the following input characteristics when measured at the SCSI device's connector:

signal true	0.0 Vdc to 0.8 Vdc
maximum total input load	-0.4 mA at 0.4 Vdc
signal false	2.0 Vdc to 5.25 Vdc
minimum input hysteresis	0.2 Vdc

4.1.2 Termination Power (Optional) Single-ended SCSI devices providing termination power (TERMPWR) shall have the following characteristics:

Vterm = 4.0 Vdc to 5.25 Vdc
800 mA minimum source drive capability
1.0 mA maximum sink capability (except for the purpose of providing power to an internal terminator) with 1.0 A recommended current limiting(e.g. a fuse).

4.2 SCSI Bus

Communication on the SCSI bus is allowed between only two SCSI devices at any given time. There is a maximum of eight SCSI devices. Each SCSI device has an SCSI ID bit assigned as shown in Figure 4-2.

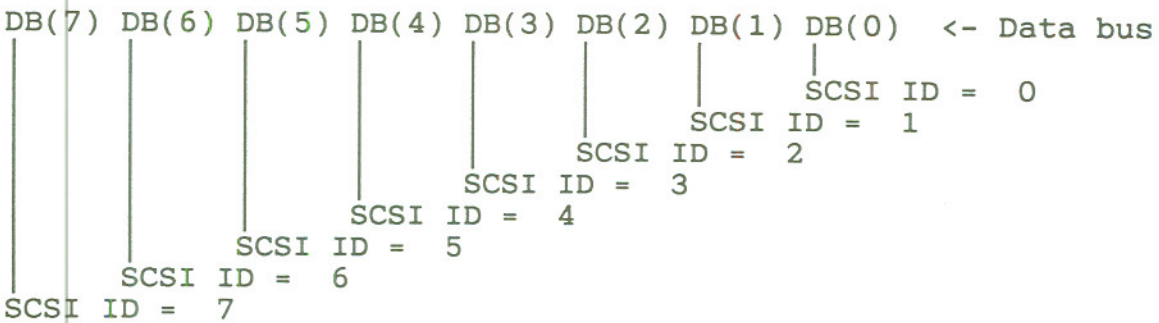
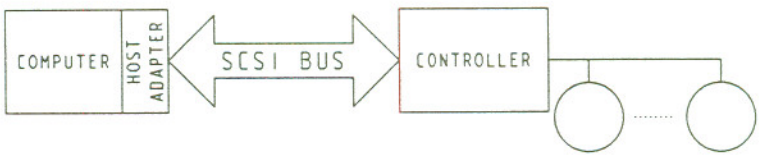


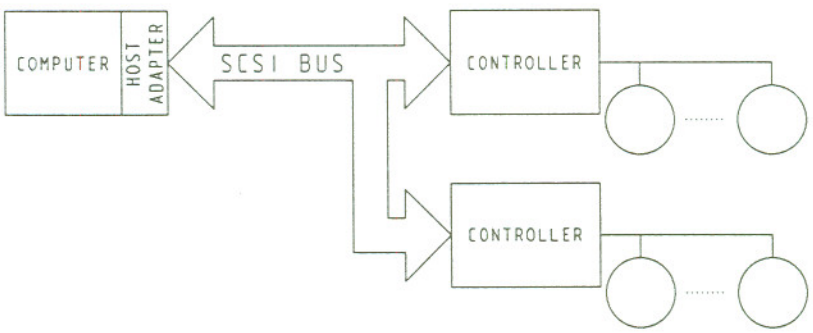
Figure 4-2 SCSI ID bits

When two SCSI devices communicate on the SCSI bus, one acts as an initiator and the other acts as a target. The initiator originates an operation and the target performs the operation. An SCSI device usually has a fixed role as an initiator or target, but some devices may be able to assume either role.

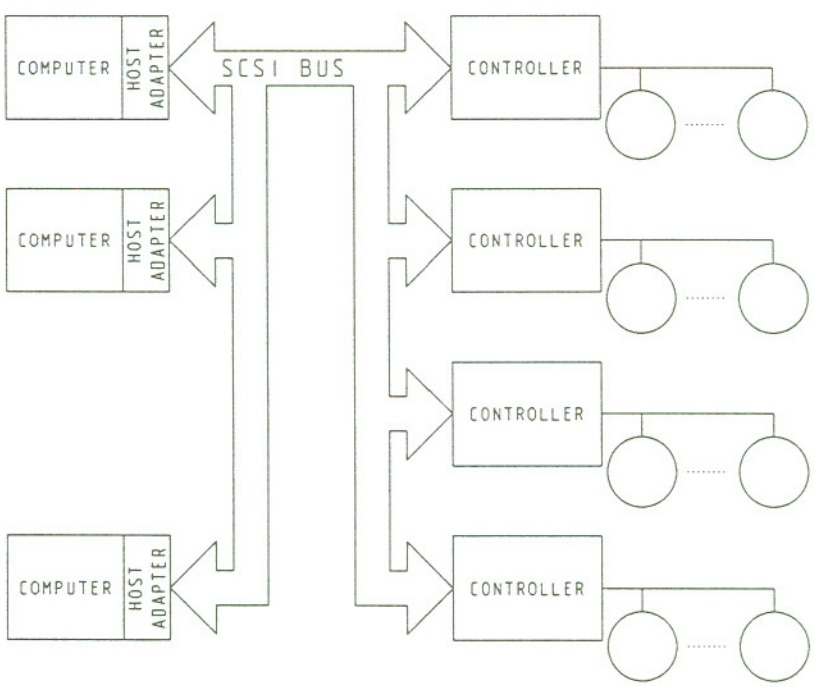
An initiator may address up to eight peripheral devices (LUN's) that are connected to a target. An option allows the addressing of up to 2,048 peripheral devices per target using extended messages. Three sample system configurations are shown in Figure 4-3.



SINGLE INITIATOR, SINGLE TARGET



SINGLE INITIATOR, MULTI TARGET



MULTI INITIATOR, MULTI TARGET

figure 4-3

Up to eight SCSI devices can be supported on the SCSI bus. They can be any combination of initiators and targets.

Certain SCSI bus functions are assigned to the initiator and certain SCSI bus functions are assigned to the target. The initiator may arbitrate for the SCSI bus and select a particular target. The target may request the transfer of COMMAND, DATA, STATUS, or other information on the DATA bus, and in some cases it may arbitrate for the SCSI bus and reselect an initiator for the purpose of continuing an operation.

Information transfers on the DATA bus are asynchronous and follow a defined REQ/ACK handshake protocol. One byte of information may be transferred with each handshake. An option is defined for synchronous data transfer.

4.3 SCSI Bus Signals

There are a total of eighteen signals. Nine are used for control and nine are used for data. (Data signals include the parity signal option). These signals are described as follows:

BSY (BUSY). An "OR-tied" signal that indicates that the bus is being used.

SEL (SELECT). A signal used by an initiator to select a target or by a target to reselect an initiator.

C/D (CONTROL/DATA). A signal driven by a target that indicates whether CONTROL or DATA information is on the DATA bus. True indicates CONTROL.

I/O (INPUT/OUTPUT). A signal driven by a target that controls the direction of data movement on the DATA bus with respect to an initiator. True indicates input to the initiator. This signal is also used to distinguish between SELECTION and RESELECTION phases.

MSG (MESSAGE). A signal driven by a target during the MESSAGE phase.

REQ (REQUEST). A signal driven by the target to indicate a request for a REQ/ACK date transfer handshake.

ACK (ACKNOWLEDGE). A signal driven by an initiator to indicate an acknowledgment for a REQ/ACK data transfer handshake.

ATN (ATTENTION). A signal driven by an initiator to indicate the ATTENTION condition.

RST (RESET). An "OR-tied" signal that indicates the RESET condition.

DB(7-0,P) (DATA BUS). Eight data-bit signals, plus a parity-bit signal that form a DATA bus. DB(7) is the most significant

bit and has the highest priority during the ARBITRATION phase. Bit number, significance, and priority decrease downward to DB(0). A data bit is defined as one when the signal value is true and is defined zero when the signal value is false.

Data parity DB(P) is odd. The use of parity is a system option (i.e., a system is configured so that all SCSI devices on a bus generate parity and have parity detection enabled, or all SCSI devices have parity detection disabled or not implemented). Parity is not valid during the ARBITRATION phase.

4.3.1 Signal Values Signals may assume true or false values. There are two methods of driving these signals. In both cases, the signal shall be actively driven true, or asserted. In the case of OR-tied drivers, the driver does not drive the signal to the false state, rather the bias circuitry of the bus terminators pulls the signal false whenever it is released by the drivers at every SCSI device. If any driver is asserted, then the signal is true. In the case of non-OR-tied drivers, the signal may be actively driven false, or negated. In this standard, wherever the term negated is used it means that the signal may be actively driven false, or may be simply released (in which case the bias circuitry pulls it false), at the option of the implementor. The advantage to actively drive signals false is that the transition from true to false occurs more quickly, and noise margins may be somewhat improved; this may permit somewhat faster date transfer.

4.3.2 OR-tied Signals The BSY and RST signals shall be OR-tied only. In the ordinary operation of the bus, these signals are simultaneously driven true by several drivers. No signals other than BSY, RST, and DB(P) are simultaneously driven by two or more drivers, and any signal other than BSY and RST may employ OR-tied or non-OR-tied drivers. DB(P) shall not be driven false during the ARBITRATION phase. There is no operational problem in mixing OR-tied and non-OR-tied drivers on signals other than BSY and RST.

4.3.3 Signal Sources Table 4-1 indicate which type of SCSI device is allowed to source each signal. No attempt is made to show if the source is driving asserted, driving negated, or is passive. All SCSI device drivers that are not active sources shall be in the passive state. Note that the RST signal may be sourced by any SCSI device at any time.

Table 4-1
Signal Sources

Bus Phase	Signals: BSY	SEL	C/D, I/O, MSG, REQ	ACK/ATN	DB(7-0,P)
BUS FREE	None	None	None	None	None
ARBITRATION	All	Winner	None	None	SCSI ID
SELECTION	I&T	Initiator	None	Initiator	Initiator
RESELECTION	I&T	Target	Target	Initiator	Target
COMMAND	Target	None	Target	Initiator	Initiator
DATA IN	Target	None	Target	Initiator	Target
DATA OUT	Target	None	Target	Initiator	Initiator
STATUS	Target	None	Target	Initiator	Target
MESSAGE IN	Target	None	Target	Initiator	Target
MESSAGE OUT	Target	None	Target	Initiator	Initiator

All: The signal shall be driven by all SCSI devices that are actively arbitrating.

SCSI ID: A unique data bit (the SCSI ID) shall be driven by each SCSI device that is actively arbitrating; the other seven data bits shall be released (i.e., not driven) by this SCSI device. The parity bit (DB(P)) may be undriven or driven to the true state, but shall never be driven to the false state during this phase.

I&T: The signal shall be driven by the Initiator, Target, or both, as specified in the SELECTION phase and RESELECTION phase.

Initiator: If this signal is driven, it shall be driven only by the active Initiator.

None: The signal shall be released; that is, not driven by any SCSI device. The bias circuitry of the bus terminators pulls the signal to the false state.

Winner: The signal shall be driven by the one SCSI device that wins arbitration.

Target: If this signal is driven, it shall be driven only by the active target.

4.4 SCSI Bus Timing

Unless otherwise indicated, the delay-time measurements for each SCSI device, shown in 4.4.1 through 4.4.14, shall be calculated from signal conditions existing at that SCSI device's own SCSI bus connection. Thus, these measurements (except cable skew delay) can be made without considering delays in the cable.

4.4.1 Arbitration Delay (2.2 microseconds) The minimum time an SCSI device shall wait from asserting BSY for arbitration until the DATA bus can be examined to see if arbitration has been won. There is no maximum time.

4.4.2 Assertion Period (90 nanoseconds) The minimum time that a target shall assert REQ while using synchronous data transfers. Also, the minimum time that an initiator shall assert ACK while using synchronous data transfers.

4.4.3 Bus Clear Delay (800 nanoseconds) The maximum time for an SCSI device to stop driving all bus signals after:

(1) The BUS FREE phase is detected (BSY and SEL both false for a bus settle delay)
(2) SEL is received from another SCSI device during the ARBITRATION phase
(3) The transition of RST to true.

NOTE: For the first condition above, the maximum time for an SCSI device to clear the bus is 1200 nanoseconds from BSY and SEL first becoming both false. If an SCSI device requires more than a bus settle delay to detect BUS FREE phase, it shall clear the bus within a bus clear delay minus the excess time.

4.4.4 Bus Free Delay (800 nanoseconds) The minimum time that an SCSI device shall wait from its detection of the BUS FREE phase (BSY and SEL both false for a bus settle delay) until its assertion of BSY when going to the ARBITRATION phase.

4.4.5 Bus Set Delay (1.8 microseconds) The maximum time for an SCSI device to assert BSY and its SCSI ID bit on the DATA bus after it detects BUS FREE phase (BSY and SEL both false for a bus settle delay) for the purpose of entering the ARBITRATION phase.

4.4.6 Bus Settle Delay (400 nanoseconds) The time to wait for the bus to settle after changing certain control signals as called out in the protocol definitions.

4.4.7 Cable Skew Delay (10 nanoseconds) The maximum difference in propagation time allowed between any two SCSI bus signals when measured between any two SCSI devices.

4.4.8 Data Release Delay (400 nanoseconds) The maximum time for an initiator to release the DATA bus signals following the transition of the I/O signal from false to true.

4.4.9 Deskew Delay (45 nanoseconds) The minimum time required for deskew of certain signals.

4.4.10 Hold Time (45 nanoseconds) The minimum time added between the assertion of REQ or ACK and the changing of the data lines to provide hold time in the initiator or target, respectively, while using synchronous data transfers.

4.4.11 Negation Period (90 nanoseconds) The minimum time that a target shall negate REQ while using synchronous data transfers. Also, the minimum time that a initiator shall negate ACK while using synchronous data transfers.

4.4.12 Reset Hold Time (25 microseconds) The minimum time for which RST is asserted. There is no maximum time.

4.4.13 Selection Abort Time (200 microseconds) The maximum time that a target (or initiator) shall take from its most recent detection of being selected (or reselected) until asserting a BSY response. This timeout is required to ensure that a target (or initiator) does not assert BSY after a SELECTION (or RESELECTION) phase has been aborted. This is not the selection time out period; see Sections 5.1.3.5 and 5.1.4.2 for a complete description.

4.4.14 Selection Timeout Delay (250 milliseconds, recommended) The minimum time that an initiator (or target) should wait for a BSY response during the SELECTION (or RESELECTION) phase before starting the timeout procedure. Note that this is only a recommended time period.

4.4.15 Transfer Period (set during a MESSAGE phase) The Transfer Period specifies the minimum time allowed between the leading edges of successive REQ pulses and of successive ACK pulses while using synchronous data transfers.

CHAPTER 5

Logical Characteristics

5.1 SCSI Bus Phases

The SCSI includes eight distinct phases:

- BUS FREE phase
- ARBITRATION phase
- SELECTION phase
- RESELECTION phase
- COMMAND phase
- DATA phase
- STATUS phase
- MESSAGE phase

These phases are collectively termed the Information Transfer phases.

The SCSI bus can never be in more than one phase at any given time. Unless otherwise noted in the following description, signals that are not mentioned shall not be asserted.

5.1.1 Bus Free Phase The BUS FREE phase is used to indicate that no SCSI device is actively using the SCSI bus and that it is available for subsequent users.

SCSI devices shall detect the BUS FREE phase after SEL and BSY are both false for at least a bus settle delay.

SCSI devices shall release all SCSI bus signals within a bus clear delay after BSY and SEL become continuously false for a bus settle delay. If an SCSI device requires more than a bus settle delay to detect the BUS FREE phase then it shall release all SCSI bus signals within a bus clear delay minus the excess time to detect the BUS FREE phase. The total time to clear the SCSI bus shall not exceed a bus settle delay plus a bus clear delay.

5.1.2 Arbitration Phase The ARBITRATION phase allows one SCSI device to gain control of the SCSI bus so that it can assume the role of an initiator or target.

NOTE: Implementation of the ARBITRATION phase is a system option. Systems that do not implement this option can have only one initiator. The ARBITRATION phase is required for systems that use the RESELECTION phase.

The procedure for an SCSI device to obtain control of the SCSI bus is as follows:

(1) The SCSI device shall first wait for the BUS FREE phase to occur. The BUS FREE phase is detected whenever both BSY and SEL are simultaneously and continuously false for a bus settle delay. (Implementors Note: This bus settle delay is

necessary because a transmission line phenomenon known as a "wire-OR glitch" may cause BSY to briefly appear false, even though it is being driven true.)

(2) The SCSI device shall wait a minimum of a bus free delay after detection of the BUS FREE phase(i.e. after BSY and SEL are both false for a bus settle delay) before driving any signal.

(3) Following the bus free delay in Step (2), the SCSI device may arbitrate for the SCSI bus by asserting both BSY and its own SCSI ID, however the SCSI device shall not arbitrate (i.e. assert BSY and its own SCSI ID) if more than a bus set delay has passed since the BUS FREE phase was last observed. (Implementers Note: There is no maximum delay before asserting BSY and the SCSI ID following the bus free delay in Step (2) as long as the bus remains in the BUS FREE phase. However, SCSI devices that delay longer than a bus settle delay plus a bus set delay from the time when BSY and SEL first become false may fail to participate in arbitration when competing with faster SCSI devices.)

(4) After waiting at least an arbitration delay (measured from its assertion of BSY) the SCSI device shall examine the DATA BUS. If a higher priority SCSI ID is true on the DATA BUS (DB(7) is the highest), then the SCSI device has lost the arbitration and the SCSI device may release its signals and return to Step (1). If no higher priority SCSI ID bit is true on the DATA BUS, then the SCSI device has won the arbitration and it shall assert SEL. Any other SCSI device that is participating in the ARBITRATION phase has lost the arbitration and shall release BSY and its SCSI ID bit within a bus clear delay after SEL becomes true. An SCSI device that loses arbitration may return to Step (1).

(5) The SCSI device that wins arbitration shall wait at least a bus clear delay plus a bus settle delay after asserting SEL before changing any signals.

NOTE: The SCSI ID bit is a single bit on the DATA BUS that corresponds to the SCSI device's unique SCSI address. All other seven DATA BUS bits shall be released by the SCSI device. Parity is not valid during the ARBITRATION phase. During the ARBITRATION phase, DB(P) may be undriven or driven to the true state, but shall not be driven to the false state.

5.1.3 Selection Phase The SELECTION phase allows an initiator to select a target for the purpose of initiating some target function (e.g. READ or WRITE command).

NOTE: During the SELECTION phase the I/O signal shall be negated so that this phase can be distinguished from the RESELECTION phase.

5.1.3.1 Nonarbitrating Systems In systems with the ARBITRATION phase not implemented, the initiator shall first detect the BUS FREE phase and then wait a minimum of a bus clear delay. Then, except in certain single initiator environments with initiators employing the single initiator option (see 5.1.3.4), the initiator shall assert the desired target's SCSI ID and its own initiator SCSI ID (and the parity bit in systems with parity implemented) on the DATA bus. After two deskew delays the initiator shall assert SEL.

5.1.3.2 Arbitrating Systems In systems with ARBITRATION phase implemented, the SCSI device that won the arbitration has both BSY and SEL asserted and has delayed at least a bus clear delay plus a bus settle delay before ending the ARBITRATION phase. The SCSI device that won the arbitration becomes an initiator by releasing I/O. Except in certain single initiator environments with initiators employing the single initiator option (see 5.1.3.4), the initiator shall assert the desired target's SCSI ID and its own initiator SCSI ID (and the parity bit in systems with parity implemented) on the DATA BUS. The initiator shall then wait at least two deskew delays and release BSY. The initiator shall then wait at least a bus settle delay before looking for a response from the target.

5.1.3.3 All Systems In all systems, the target shall determine that it is selected when SEL and its SCSI ID bit are true and BSY and I/O are false for at least a bus settle delay. The selected target may examine the DATA bus in order to determine the SCSI ID of the selecting initiator unless the initiator employed the single initiator option (see 5.1.3.4). The selected target shall then assert BSY within a selection abort time of its most recent detection of being selected; this is required for correct operation of the timeout procedure. In systems with parity implemented, the target shall not respond to a selection if bad parity is detected. Also, if more than two SCSI ID bits are on the DATA bus, the target shall not respond to selection.

At least two deskew delays after the initiator detects BSY is true, it shall release SEL and may change the DATA bus.

5.1.3.4 Single Initiator Option Initiators that do not implement the RESELECTION phase and do not operate in the multiple initiator environment are allowed to set only the target's SCSI ID during the SELECTION phase. This makes it impossible for the target to determine the initiator's SCSI ID.

5.1.3.5 Selection Timeout Procedure Two optional selection timeout procedures are specified for clearing the SCSI bus if the initiator waits a minimum of a selection timeout delay and there has been no BSY response from the target:

(1) Optionally, the initiator shall assert the RST signal (see 5.2.2).
(2) Optionally, the initiator shall continue asserting SEL and shall release the DATA bus. If the initiator has not detected

BSY to be true after at least a selection abort time plus two deskew delays, the initiator shall release SEL allowing the SCSI bus to go to the BUS FREE phase. SCSI devices shall ensure that when responding to selection that the selection was still valid within a selection abort time of their assertion of BSY. Failure to comply with this requirement could result in an improper selection (two targets connected to the same initiator, wrong target connected to an initiator, or a target connected to no initiator).

5.1.4 Reselection Phase (optional) RESELECTION is an optional phase that allows a target to reconnect to an initiator for the purpose of continuing some operation that was previously started by the initiator but was suspended by the target (i.e., the target disconnected by allowing a BUS FREE phase to occur before the operation was complete).

5.1.4.1 Reselection RESELECTION can only be used in systems that have the ARBITRATION phase implemented.

Upon completing the ARBITRATION phase, the winning SCSI device has both BSY and SEL asserted and has delayed at least a bus clear delay plus a bus settle delay. The winning SCSI device becomes a target by asserting the I/O signal. The winning SCSI device shall also set the DATA BUS to a value that is the OR of its SCSI ID bit and the initiator's SCSI ID bit. The target shall wait at least two deskew delays and release BSY. The target shall then wait at least a bus settle delay before looking for a response from the initiator.

The initiator shall determine that it is reselected when SEL, I/O, and its SCSI ID bit are true and BSY is false for at least a bus settle delay. The reselected initiator may examine the DATA BUS in order to determine the SCSI ID of the reselecting target. The reselected initiator shall then assert BSY within a selection abort time of its most recent detection of being reselected; this is required for correct operation of the timeout procedure. In systems with parity implemented, the initiator shall not respond to a RESELECTION if bad parity is detected. Also, the initiator shall not respond to a RESELECTION if more than two SCSI ID bits are on the DATA BUS.

After the target detects BSY, it shall also assert BSY and wait at least two deskew delays and then release SEL. The target may then change the I/O signal and the DATA BUS. After the reselected initiator detects SEL false, it shall release BSY. The target shall continue asserting BSY until the target is ready to relinquish the SCSI bus.

NOTE: When the target is asserting BSY, a transmission line phenomenon known as a "wire-OR glitch" may cause BSY to appear false for up to a round-trip propagation delay following the release of BSY by the initiator. This is the reason why the BUS FREE phase is recognized only after both BSY and SEL are continuously false for a minimum of a bus settle delay. Cables longer than 25 meters should not be used even if the chosen

driver, receiver and cable provide adequate noise margins because they increase the duration of the glitch and could cause SCSI devices to inadvertently detect the BUS FREE phase.

5.1.4.2 Reselection Timeout Procedure Two optional RESELECTION timeout procedures are specified for clearing the SCSI bus during a RESELECTION phase if the target waits for a minimum of a selection timeout period and there has been no BSY response from the initiator:

(1) Optionally, the target shall assert the RST signal (see 5.2.2).

NOTE: This module will never assert the RST signal.

(2) Optionally, the target shall continue asserting SEL and I/O and shall release all DATA BUS signals. If the target has not detected BSY to be true after at least a selection abort time plus two deskew delays, the target shall release SEL and I/O allowing the SCSI bus to go to the BUS FREE phase. SCSI devices that respond to RESELECTION shall ensure that the RESELECTION was still valid within a selection abort time of their assertion of BSY. Failure to comply with this requirement could result in an improper reselection (two initiators connected to the same target or the wrong initiator connected to a target).

5.1.5 Information Transfer Phases

note: The COMMAND, DATA, STATUS, and MESSAGE phases are all grouped together as the information transfer phases because they are all used to transfer data or control information via the DATA bus. The actual contents of the information is beyond the scope of this section.

The C/D, I/O, and MSG signals are used to distinguish between the different information transfer phases. (see Table 5-1.) The target drives these three signals and, therefore, controls all changes from one phase to another. The initiator can request a MESSAGE OUT phase by asserting ATN, while the target can cause the BUS FREE phase by releasing MSG, C/D, I/O, and BSY.

Table 5-1
Information Transfer Phases

signal					
MSG	C/D	I/O	Phase Name	Direction of Transfer	Comment
0	0	0	Data Out	Initiator to Target	\ Data
0	0	1	Data In	Initiator from Target	/ Phase
0	1	0	Command	Initiator to Target	
0	1	1	Status	Initiator from Target	
1	0	0	*		
1	0	1	*		
1	1	0	Message Out	Initiator to Target	\ Message
1	1	1	Message In	Initiator from Target	/ Phase

key: 0 = false, 1 = true,
* = reserved for future standardization.

The information transfer phases use one or more REQ/ACK handshakes to control the information transfer. Each REQ/ACK handshake allows the transfer of one byte of information. During the information transfer phases BSY shall remain true and SEL shall remain false. Additionally, during the information transfer phases, the target shall continuously envelope the REQ/ACK handshake(s) with C/D, I/O, and MSG in such a manner that these control signals are valid for a bus settle delay before the assertion of REQ of the first handshake and remain valid until the negation of ACK at the end of the last handshake.

5.1.5.1 Asynchronous Information Transfer The target shall control the direction of information transfer by means of the I/O signal. When I/O is true, information shall be transferred from the target to the initiator. When I/O is false, information shall be transferred from the initiator to the target.

If I/O is true (transfer to the initiator), the target shall first drive DB(7-0,P) to their desired values, delay at least one deskew delay plus a cable skew delay, then assert REQ. DB(7-0,P) shall remain valid until ACK is true at the target. The initiator shall read DB(7-0,P) after REQ is true, then signal its acceptance of the data by asserting ACK. When ACK becomes true at the target, the target may change or release DB(7-0,P) and shall negate REQ. After REQ is false the initiator shall then negate ACK. After ACK is false the target may continue the transfer by driving DB(7-0,P) and asserting REQ, as described above.

If I/O is false (transfer to the target), the target shall request information by asserting REQ. The initiator shall drive DB(7- 0,P) to their desired values, delay at least one deskew delay plus a cable skew delay and assert ACK. The initiator shall continue to drive DB(7-0,P) until REQ is false. When ACK becomes true at the target, the target shall read DB(7-0,P), then negate REQ. When REQ becomes false at the initiator, the initiator may change or release DB(7-0,P) and shall negate ACK. The target may

continue the transfer by asserting REQ, as described above.

5.1.5.2 Synchronous Data Transfer (optional) Synchronous data transfer is optional, and may be used only in the data phase if previously agreed to by the initiator and target through the message system (see SYNCHRONOUS DATA TRANSFER REQUEST message). The messages determine the use of synchronous mode by both SCSI devices and establish a REQ/ACK offset and a transfer period.

The REQ/ACK offset specifies the maximum number of REQ pulses that can be sent by the target in advance of the number of ACK pulses received from the initiator, establishing a pacing mechanism. If the number of REQ pulses exceeds the number of ACK pulses by the REQ/ACK offset, the target shall not assert REQ until the next ACK pulse is received. A requirement for successful completion of the data phase is that the number of ACK and REQ pulses be equal.

The target shall assert the REQ signal for a minimum of an assertion period. The target shall wait at least the greater of a transfer period from the last transition of REQ to true or a minimum of a negation period from the last transition of REQ to false before asserting the REQ signal.

The initiator shall send one pulse on the ACK signal for each REQ pulse received. The initiator shall assert the ACK signal for a minimum of an assertion period. the initiator shall wait at least the greater of a transfer period from the last transition of ACK to true or for a minimum of a negation period from the last transition of ACK to false before asserting the ACK signal.

If I/O is true (transfer to the initiator), the target shall first drive DB(7-0,P) to their desired values, wait at least one deskew delay plus a cable skew delay, then assert REQ. DB(7-0,P) shall be held valid for a minimum of one deskew delay plus one cable skew delay plus one hold time after the assertion of REQ. The target shall assert REQ for a minimum of an assertion period. The target may then negate REQ and change or release DB(7-0,P). The initiator shall read the value on DB(7-0,P) within one hold time of the transition of REQ to true. The initiator shall then respond with an ACK pulse.

If I/O is false (transfer to the target), the initiator shall transfer one byte for each REQ pulse received. After receiving a REQ pulse, the initiator shall first drive DB(7-0,P) to their desired values, delay at least one deskew delay plus one cable skew delay, then assert ACK. The initiator shall hold DB(7-0,P) valid for at least one deskew delay plus one cable skew delay plus one hold time after the assertion of ACK. The initiator shall assert ACK for a minimum of an assertion period. The initiator may then negate ACK and may change or release DB(7-0,P). The target shall read the value of DB(7-0,P) within one hold time of the transition of ACK to true.

5.1.6 Command Phase The COMMAND phase allows the target to request command information from the initiator. The target shall assert the C/D signal and negate the I/O and MSG signals during the REQ/ACK handshake(s) of this phase.

5.1.7 Data Phase The data phase is a term that encompasses both the DATA IN phase and the DATA OUT phase.

5.1.7.1 Data In Phase The DATA IN phase allows the target to request that data be sent to the initiator from the target. The target shall assert the I/O signal and negate the C/D and MSG signals during the REQ/ACK handshake(s) of this phase.

5.1.7.2 Data Out Phase The DATA OUT phase allows the target to request that data be sent from the initiator to the target. The target shall negate the C/D, I/O, and MSG signals during the REQ/ACK handshake(s) of this phase.

5.1.8 Status Phase The STATUS phase allows the target to request that status information be sent from the target to the initiator. The target shall assert C/D and I/O and negate the MSG signal during the REQ/ACK handshake of this phase.

5.1.9 Message Phase The message phase is a term that references either a MESSAGE IN, or a MESSAGE OUT phase. The first byte transferred in either of these phases shall be either a single-byte message or the first of a multiple-byte message. Multiple-byte messages shall be wholly contained within a single message phase.

5.1.9.1 Message In Phase The MESSAGE IN phase allows the target to request that messages be sent to the initiator from the target. The target shall assert C/D, I/O, and MSG during the REQ/ACK handshake(s) of this phase.

5.1.9.2 Message Out Phase The MESSAGE OUT phase allows the target to request that a message be sent from the initiator to the target. The target may invoke this phase at its convenience in response to the ATTENTION condition (see 5.2.1) created by the initiator. The target shall assert C/D and MSG and negate I/O during the REQ/ACK handshake(s) of this phase. The target shall handshake byte(s) in this phase until ATN goes false.

If the target detects one or more parity error(s) on the message byte(s) received, it may indicate its desire to retry the message by asserting REQ after detecting ATN has gone false and prior to changing to any other phase. The initiator, upon detecting this condition, shall resend all of the previous message byte(s) sent during this phase. When resending more than one message byte, the initiator shall assert ATN prior to asserting ACK on the first byte and shall maintain ATN asserted until the last byte is sent as described in 5.2.1.

If the target receives all of the message byte(s) successfully (i.e. no parity errors), it shall indicate that it does not wish

to retry by changing to any information transfer phase other than the MESSAGE OUT phase and transfer at least one byte. The target may also indicate that it has successfully received the message byte(s) by changing to the BUS FREE phase (e.g. ABORT or BUS DEVICE RESET messages).

5.1.10 Signal Restrictions Between Phases When the SCSI bus is between two information transfer phases, the following restrictions shall apply to the SCSI bus signals:

(1) The BSY, SEL, REQ, and ACK signals shall not change.

(2) The C/D, I/O, MSG, and DATA BUS signals may change. When switching the DATA BUS direction from out (initiator driving) to in (target driving), the target shall delay driving the DATA BUS by at least a data release delay plus a bus settle delay after asserting the I/O signal and the initiator shall release the DATA BUS no later than a data release delay after the transition of the I/O signal to true. When switching the DATA BUS direction from in (target driving) to out (initiator driving), the target shall release the DATA BUS no later than a deskew delay after negating the I/O signal.

(3) The ATN and RST signals may change as defined under the descriptions for the ATTENTION condition (5.2.1) and RESET condition (5.2.2).

5.2 SCSI Bus Conditions

The SCSI bus has two asynchronous conditions: the ATTENTION condition and the RESET condition. These conditions cause the SCSI device to perform certain actions and can alter the phase sequence.

5.2.1 Attention Condition The ATTENTION condition allows an initiator to inform a target that the initiator has a message ready. The target may get this message at its convenience by performing a MESSAGE OUT phase.

The initiator creates the ATTENTION condition by asserting ATN at any time except during the ARBITRATION or BUS FREE phases.

The target may respond with the MESSAGE OUT phase.

The initiator shall keep ATN asserted if more than one byte is to be transferred. The initiator may negate the ATN at any time except it shall not negate the ATN signal while the ACK signal is asserted during a MESSAGE OUT phase. Normally, the initiator negates ATN while REQ is true and ACK is false during the last REQ/ACK handshake of the MESSAGE OUT phase.

5.2.2 Reset Condition The RESET condition is used to immediately clear all SCSI devices from the bus. This condition shall take precedence over all other phases and conditions. Any SCSI device may create the RESET condition by asserting RST for a minimum of a reset hold time.

NOTE: This module will never assert RST.

During the RESET condition, the state of all SCSI bus signals other than RST is not defined.

All SCSI devices shall release all SCSI bus signals (except RST) within a bus clear delay of the transition of RST to true. The BUS FREE phase always follows the RESET condition.

This module implements the "hard" RESET option. Upon detecting of the RST signal, or the BUS DEVICE RESET message, this module will perform a power-on reset. Asserting the RST signal, or sending the BUS DEVICE RESET message, will cause the same effect as if this module were powered off then back on.

5.3 SCSI Bus Phase Sequences

The order in which phases are used on the SCSI bus follows a predescribed sequence.

In all systems, the RESET condition can abort any phase and is always followed by the BUS FREE phase. Also, any other phase can be followed by the BUS FREE phase.

5.3.1 Nonarbitrating Systems In systems where ARBITRATION phase is not implemented, the allowable sequences shall be as shown in Figure 5-1. The normal progression is from the BUS FREE phase to SELECTION, and from SELECTION to one or more of the information transfer phases (COMMAND, DATA, STATUS, or MESSAGE).

5.3.2 Arbitrating Systems In systems where ARBITRATION phase is implemented, the allowable sequences shall be as shown in Figure 5-2. The normal progression is from the BUS FREE phase to ARBITRATION, from ARBITRATION to SELECTION or RESELECTION, and from SELECTION or RESELECTION to one or more of the information transfer phases (COMMAND, DATA, STATUS, or MESSAGE).

5.3.3 All Systems There are no restrictions on the sequences between information transfer phases. A phase type may even be followed by the same phase type (e.g. a data phase may be followed by another data phase).

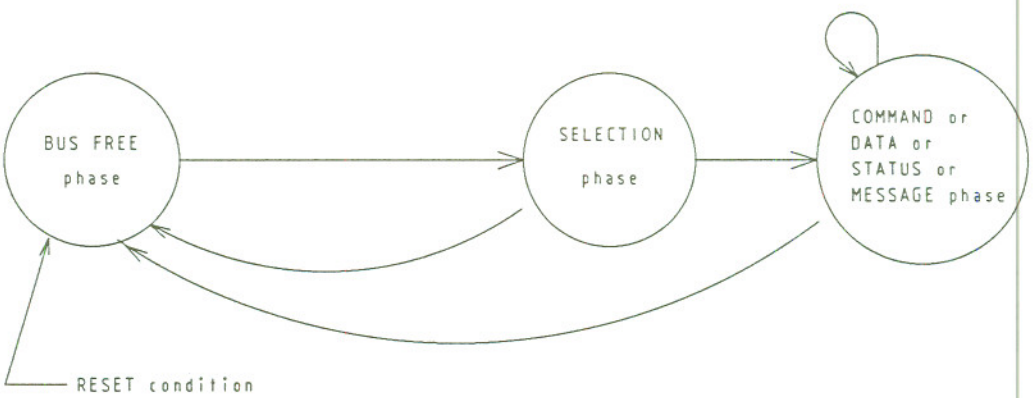


figure 5-1. Phase Sequence without Arbitration

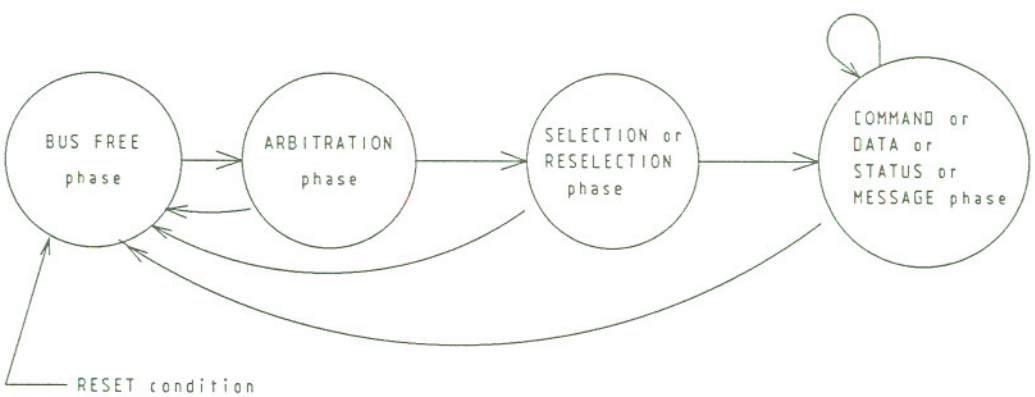


figure 5-2. Phase Sequence with Arbitration

5.4 SCSI Pointers

Consider the system shown in figure 5-3 in which an initiator and target communicate on the SCSI bus in order to execute a command.

Figure 5-3. Simplified SCSI System

The SCSI architecture provides for two sets of three pointers within each initiator. The pointers reside in the initiator path control. The first set of pointers are known as the current (or active) pointers. These pointers are used to represent the state of the interface and point to the next command, data, or status byte to be transferred between the initiator's memory and the target. There is only one set of current pointers in each initiator. The current pointers are used by the target currently connected to the initiator.

The second set of pointers are known as the saved pointers. There is one set of saved pointers for each command that is currently active (whether or not it is currently connected). The saved command pointer always points to the start of the command descriptor block for the current command. The saved status pointer always points to the start of the status area for the current command. At the beginning of each command, the saved data pointer points to the start of the data area. It remains at this value until the target sends a SAVE DATA POINTER message (see 5.5.2) to the initiator. In response to this message, the initiator stores the value of the current data pointer into the saved data pointer. The target may restore the current pointers to their saved values by sending a RESTORE POINTERS message (see 5.5.2) to the initiator. The initiator moves the saved value of each pointer into the corresponding current pointer. Whenever an SCSI device disconnects from the bus, only the saved pointer values are retained. The current pointer values are restored from the saved values upon the next reconnection.

5.5 Message System Specification

The messages system allows communication between an initiator and target for the purpose of physical data path management.

5.5.1 Message Protocol This module requires that any SCSI device it communicates with be able to receive and understand the COMMAND COMPLETE message. This is the only message an initiator communicating with this module is required to implement. To maximize flexibility, however, this module supports all single byte (nonextended) messages defined in the ANSI SCSI Specification (excepting the LINKED COMMAND COMPLETE messages).

This module will not send any message other than COMMAND COMPLETE until it is informed by the initiator that it can support other messages. The initiator does this in the SELECTION phase by asserting ATN prior to the SCSI bus condition of SEL and BSY false. This informs this module that the initiator can at least generate a MESSAGE REJECT message if it does not understand a message from this module.

If this module detects the ATN signal asserted during SELECTION phase, it will enter the MESSAGE OUT phase and take a message byte from the initiator. This module expects an IDENTIFY message at this time. The NO OPERATION, BUS DEVICE RESET, and ABORT messages will also function properly here. All other messages will be rejected with a MESSAGE REJECT message.

The IDENTIFY message the initiator sends to this module should specify LUN = 0, 1, 2, or 3. If the initiator supports disconnect/reconnect, it should set bit 6 in the IDENTIFY message (see section 5.5.2). Unless bit 6 is set, this module will not attempt to disconnect at any point during a command. Bit 6 should not be set unless the initiator can support receiving the SAVE DATA POINTER, RESTORE POINTERS, and DISCONNECT messages. Upon receipt of a legitimate IDENTIFY message, this module will enter the COMMAND phase and begin requesting command bytes from the initiator.

This module supports the following messages:

Table 5-2
Message Codes

Code	Description	Direction	
00	COMMAND COMPLETE	In	
02	SAVE DATA POINTER	In	
03	RESTORE POINTERS	In	
04	DISCONNECT	In	
05	INITIATOR DETECTED ERROR		Out
06	ABORT		Out
07	MESSAGE REJECT	In	Out
08	NO OPERATION		Out
09	MESSAGE PARITY ERROR		Out
0C	BUS DEVICE RESET		Out
80-FF	IDENTIFY	In	Out

In = this module to initiator
Out = initiator to this module

5.5.2 Messages The single byte messages supported by this module (Table 5-2) are listed along with their code values and their definitions.

COMMAND COMPLETE 00: This message is sent from this module to an initiator to indicate that the execution of a command has

terminated and that valid status has been sent to the initiator. After successfully sending this message, this module shall go to the BUS FREE phase by releasing BSY. This message must be supported by any initiator selecting this module.

NOTE: The command may have been executed successfully or unsuccessfully as indicated in the status.

SAVE DATA POINTER 02: This message is sent from this module to direct the initiator to save a copy of the present active data pointer for the currently attached logical unit. (See 5.4 for a definition of pointers.) This message is usually sent just prior to the DISCONNECT message when disconnecting in the DATA phase.

RESTORE POINTERS 03: This message is sent by this module to direct the initiator to restore the most recently saved pointers (for the currently attached logical unit) to the active state. Pointers to the command, data, and status locations for the logical unit shall be restored to the active pointers. Command and status pointers shall be restored to the beginning of the present command and status areas. the data pointer shall be restored to the value at the beginning of the data area in the absence of a SAVE DATA POINTER message or to the value at the point at which the last SAVE DATA POINTER message occurred for that logical unit.

DISCONNECT 04: this message is sent from this module to inform an initiator that the present physical data path is going to be broken (this module plans to disconnect by releasing BSY), but that a later reconnect will be required in order to complete the current operation. If the initiator detects a BUS FREE phase (other than as a result of an ABORT or BUS DEVICE RESET message or a RESET condition) without first receiving a DISCONNECT or COMMAND COMPLETE message, the initiator shall consider this as a catastrophic error condition. This message shall not cause the initiator to save the data pointer.
NOTE: If DISCONNECT messages are used to break a long data transfer into two or more shorter transfers, then a SAVE DATA POINTER message is issued before the DISCONNECT message.

INITIATOR DETECTED ERROR 05: This message is sent from an initiator to inform this module that an error (e.g, parity error on the SCSI bus) has occurred that does not preclude a retry of the operation. Although present pointer integrity is not assured, a RESTORE POINTERS message or a disconnect followed by a reconnect, shall cause the pointers to be restored to their defined prior saved state. Depending on which phase is being processed, this module will either attempt to retransmit the data (STATUS or MESSAGE IN phase) or it will terminate the command (DATA IN phase). If the data is to be retransmitted, this module will first send a RESTORE POINTERS message and then continue. If the command is to be terminated a CHECK CONDITION status will result. The sense key shall be set to ABORTED COMMAND with a sense code of Initiator Detected Error.

ABORT 06: This message is sent from the initiator to this module

to clear the present operation. All pending data and status for the issuing initiator shall be cleared, and this module shall go to the BUS FREE phase. No status or ending message shall be sent for the operation.

MESSAGE REJECT 07: This message will be sent by this module to an initiator when the last message received from the initiator was inappropriate or has not been implemented. This message will be interpreted in the same way when received from the initiator.

In order to indicate its intentions of sending this message, the initiator shall assert the ATN signal prior to its release of ACK for the REQ/ACK handshake of the message that is to be rejected. When a target sends this message, it shall change to MESSAGE IN phase and send this message prior to requesting additional message bytes from the initiator. This provides an interlock so that the initiator can determine which message is rejected. The initiator must support this message if it asserts ATN during the SELECTION phase of the command.

NO OPERATION 08: This message is sent from the initiator in response to this module's request for a message when the initiator does not currently have any other valid message to send. No action will be taken by this module upon receipt of this message.

MESSAGE PARITY ERROR 09: This message is sent from the initiator to this module to indicate that one or more bytes in the last message it received has a parity error.

In order to indicate its intention of sending this message, the initiator shall assert the ATN signal prior to its release of ACK for the REQ/ACK handshake of the message that has the parity error. This provides an interlock so that the target can determine which message has the parity error.

BUS DEVICE RESET 0C: This message is sent from the initiator to direct this module to clear all current commands. This message forces this module to an initial state with no operations pending for any initiator. Upon recognizing this message, this module will go to the BUS FREE phase. This message has the same effect as a power-on reset or "hard" reset condition for this module.

IDENTIFY MESSAGE 80-FF: These messages may be either sent from or received by this module to establish the physical path connection between an initiator and this module for a particular logical unit.

bit 7: This bit is always set to one to distinguish an IDENTIFY message from the other messages.

bit 6: This bit is only set to one by the initiator. When set to one, it indicates that the initiator has the ability to accommodate disconnection and reconnection. It also indicates that the SAVE DATA POINTER, RESTORE POINTERS, and DISCONNECT messages are supported by the initiator.

bit 5-3: Reserved; must be set to zero.

bit 2-0: These bits specify a logical unit number (LUN) in a target. The only valid LUN's on this module are zero to three.

When sent from a target to an initiator during reconnection, an implied RESTORE POINTERS message shall be performed by the initiator prior to completion of this message.

5.5.3 Message Error Recovery Strategy

5.5.3.1 Parity Errors During Message Phase Parity is an option in SCSI. If the parity option is not enabled the discussion in this section is not relevant. If the initiator detects a parity violation as this module transfers a message byte to it, the initiator should ignore that message and immediately assert ATN (before releasing ACK on the offending byte). The initiator should continue receiving bytes and ignoring them until this module requests a MESSAGE OUT byte. At that time, the initiator should send a MESSAGE PARITY ERROR message. This module will then cause the previous message to be retransmitted. If the retransmitted message is again garbled, the initiator should resend the MESSAGE PARITY ERROR message. After a limited amount of retries, this module will give up, abort the command and go to the BUS FREE phase. The sense key will be set to ABORTED COMMAND and the sense code set to Parity Error On The SCSI Bus.

If this module detects a parity error during a MESSAGE OUT phase, it will continue to request MESSAGE OUT bytes until the initiator deasserts ATN. This procedure should guarantee that the initiator transfers the entire message containing the parity error. This module then requests the initiator to resend the message by not changing the phase and requesting more MESSAGE OUT bytes. After a limited amount of retries, this module will give up, abort the command and go to the BUS FREE phase. The sense key will be set to ABORTED COMMAND and the sense code to Parity Error On The SCSI Bus.

5.5.3.2 Rejected Messages This module will send MESSAGE REJECT to the initiator any time it receives a message it does not support (e.g. extended messages) or a message that does not make sense at that time (e.g. the initiator sending COMMAND COMPLETE).

When this module receives MESSAGE REJECT from the initiator, it takes action based on which message was rejected.

COMMAND COMPLETE: this module goes to the BUS FREE phase. This is not considered an error.

DISCONNECT: This module will stop trying to disconnect at this time. It may attempt to disconnect at a later time.

IDENTIFY: This module only sends this when attempting to reconnect. Rejecting this message during reconnect will terminate

the present command. The sense key is set to ABORTED COMMAND and the sense code to Message Reject Error. This module then goes immediately to the BUS FREE phase.

MESSAGE REJECT: this module will immediately terminate the current command. The sense key will be set to ABORTED COMMAND and the sense code to Message Reject Error. This module then goes to the BUS FREE phase.

RESTORE POINTERS: This message is only sent by this module when attempting to retry a command or part of a command. This module will abort the retry process.

SAVE DATA POINTERS: Same procedure as for DISCONNECT message.

This Page Intentionally Left Blank