

INTERNATIONAL
STANDARD

ISO
9316

First edition
1989-07-15

**Information processing systems — Small
Computer System Interface (SCSI)**

*Systèmes de traitement de l'information — Interface système pour petits
ordinateurs (SCSI)*



Contents

	Page
1 Scope	1
2 Normative references.....	3
3 Definitions, abbreviations and conventions	5
3.1 Definitions	5
3.2 Abbreviations	5
3.3 Editorial conventions	6
4 Physical Characteristics	7
4.1 Physical Description	7
4.2 Cable Requirements	7
4.3 Connector Requirements	7
4.4 Electrical Description	12
4.5 SCSI Bus	15
4.6 SCSI Bus Signals.....	17
4.7 SCSI Bus Timing	18
5 Logical characteristics	21
5.1 SCSI Bus Phases	21
5.2 SCSI Bus Conditions	26
5.3 SCSI Bus Phase Sequences	28
5.4 SCSI Pointers	29
6 SCSI commands	36
6.1 Command Implementation Requirements.....	36
6.2 Command Descriptor Block (CDB).....	37

6.3 Command Examples	41
7 Command descriptions for all device types.....	43
7.1 Group 0 Commands for All Device Types.....	43
7.2 Group 1 Commands for All Device Types.....	59
7.3 Group 2 Commands for All Device Types.....	62
7.4 Group 3 Commands for All Device Types.....	62
7.5 Group 4 Commands for All Device Types.....	62
7.6 Group 5 Commands for All Device Types.....	62
7.7 Group 6 Commands for All Device Types.....	62
7.8 Group 7 Commands for All Device Types.....	62
8 Command descriptions for direct-access devices.....	63
8.1 Group 0 Commands for Direct-Access Devices.....	63
8.2 Group 1 Commands for Direct-Access Devices.....	86
9 Group 0 command descriptions for sequential-access devices	99
9.1 REWIND Command	100
9.2 READ BLOCK LIMITS Command	101
9.3 READ Command	102
9.4 WRITE Command.....	103
9.5 TRACK SELECT Command.....	104
9.6 READ REVERSE Command	105
9.7 WRITE FILEMARKS Command.....	106
9.8 SPACE Command	107
9.9 VERIFY Command.....	108
9.10 RECOVER BUFFERED DATA Command.....	109
9.11 MODE SELECT Command	110
9.12 RESERVE UNIT and RELEASE UNIT Commands.....	113
9.13 ERASE Command.....	114
9.14 MODE SENSE Command	115
9.15 LOAD/UNLOAD Command	117
9.16 PREVENT/ALLOW MEDIUM REMOVAL Command	118

10 Group 0 command descriptions for printer devices.....	119
10.1 FORMAT Command.....	120
10.2 PRINT Command	121
10.3 SLEW AND PRINT Command.....	121
10.4 FLUSH BUFFER Command.....	122
10.5 RECOVER BUFFERED DATA Command.....	123
10.6 MODE SELECT Command	124
10.7 RESERVE UNIT and RELEASE UNIT Commands.....	125
10.8 MODE SENSE Command	126
10.9 STOP PRINT Command.....	128
11 Group 0 command descriptions for processor devices.....	129
11.1 RECEIVE Command.....	130
11.2 SEND Command.....	130
12 Command descriptions for write-once read-multiple devices	132
12.1 Group 0 Commands for Write-Once Read-Multiple Devices.....	132
13 Command descriptions for read-only direct-access devices	147
13.1 Group 0 Commands for Read-Only Direct-Access Devices.	147
13.2 Group 1 Commands for Read-Only Direct-Access Devices.	148
14 Status	149
Annexes	
A SCSI Signal Sequence Example.....	151
B Typical Bus Phase Sequence.....	155
C SCSI System Operation.....	157
D Recommended Shielded Connectors.....	161
E Conformance	168
F Additional Medium Type and Density Code Standards.....	169

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

Draft International Standards adopted by the technical committees are circulated to the member bodies for approval before their acceptance as International Standards by the ISO Council. They are approved in accordance with ISO procedures requiring at least 75 % approval by the member bodies voting.

International Standard ISO 9316 was prepared by Technical Committee ISO/TC 97, *Information processing systems*.

Annexes A, B, C, D, E and F are for information only.

© ISO 1989

All rights reserved. No part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from the publisher.

International Organization for Standardization

Case postale 56 • CH-1211 Genève 20 • Switzerland

Printed in Switzerland

Information processing systems — Small Computer System Interface (SCSI)

1 Scope

This International Standard provides the mechanical, electrical, and functional requirements for a small computer input/output bus and command sets for peripheral device types commonly used with small computers.

The small computer system interface, described in this International Standard, is a local I/O bus that can be operated at data rates up to 4 Mbytes/s depending upon circuit implementation choices. The primary objective of the interface is to provide host computers with device independence within a class of devices. Thus, different disk drives, tape drives, printers, and even communication devices can be added to the host computer(s) without requiring modifications to generic system hardware or software. Provision is made for the addition of nongeneric features and functions through vendor unique fields and codes.

The interface uses logical rather than physical addressing for all data blocks. For direct access devices, each logical unit may be interrogated to determine how many blocks it contains. A logical unit may coincide with all or part of a peripheral device.

Provision is made for cable lengths up to 25 m using differential drivers and receivers. A single-ended driver and receiver configuration is defined for cable lengths of up to 6 m and is primarily intended for applications within a cabinet.

The interface protocol includes provision for the connection of multiple initiators (SCSI devices capable of initiating an operation) and multiple targets (SCSI devices capable of responding to a request to perform an operation). Optional distributed arbitration (i.e., bus-contention logic) is built into the architecture of SCSI. A priority system awards interface control to the highest priority SCSI device that is contending for use of the bus. The time to complete arbitration is independent of the number of devices that are contending and can be completed in less than 10 μ s.

The physical characteristics are described in clause 4. There are two electrical alternatives: single-ended and differential. Single-ended and differential devices are electrically different and shall not be mixed on the same bus. In addition, there are several options: shielded or unshielded connectors may be used and parity may or may not be implemented.

Clause 5 describes the logical characteristics of the interface. An arbitration option is defined to permit multiple initiators and to permit concurrent I/O operations. All SCSI devices are required to be capable of operating with the defined asynchronous transfer protocol. In addition, an optional synchronous transfer protocol is defined. Clause 5 also specifies a message protocol for control of the interface. In most cases, messages are not directly apparent to the host computer software. Only one message, COMMAND COMPLETE, is mandatory; all others are optional and are not necessarily implemented. Note that some options (e.g., synchronous transfer) require the implementation of certain messages.

The SCSI command structure is specified in clause 6. Commands are classified as mandatory (M), extended (E), optional (O), or vendor unique (V). SCSI devices shall implement all mandatory commands defined for the appropriate device type and may implement other commands as well. Extended SCSI devices shall implement all extended plus all mandatory commands and may implement other commands as well. Extended SCSI devices contain commands that facilitate the writing of self-configuring software drivers that can "discover" all necessary attributes without prior knowledge of specific peripheral characteristics (such as storage capacity). Extended commands for direct access devices also implement a very large logical block address space (2^{32} blocks), although mandatory commands for direct access devices implement a somewhat smaller logical block address space (2^{21} blocks).

Clause 7 specifies those commands that have a consistent meaning for all device types.

Clauses 8 through 13 contain commands for direct-access (*e.g.*, magnetic disk), sequential-access (*e.g.*, magnetic tape), printer, processor, write-once-read-multiple (*e.g.*, optical disk), and read-only direct-access devices, respectively. The commands in each of these clauses are unique to the device type, or they have interpretations, fields, or features that are specific for the device type. Thus, for example, although the WRITE command is used for several device types, it has a somewhat different form for each type, with different parameters and meanings. Therefore, it is specified separately for each device type.

Clause 14 describes the status byte for all device types. Status is returned by targets at the end of each command.

2 Normative references

The following standards contain provisions, which through reference in this text, constitute provisions of this International Standard. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this International Standard are encouraged to investigate the possibility of applying the most recent editions of the standards listed below. Members of IEC and ISO maintain registers of currently valid International Standards.

ISO 8482:1987, *Information processing systems — Data communication — Twisted pair multipoint interconnections*.

This International Standard defines codes which controllers may use to identify that storage media conform to the following storage media standards.

ISO 1863:1976, *Information processing — 9-track, 12,7 mm (0.5 in) wide magnetic tape for information interchange recorded at 32 rpm (800 rpi)*.

ISO 3788:1976, *Information processing — 9-track, 12,7 mm (0.5 in) wide magnetic tape for information interchange recorded at 63 rpm (1 600 rpi), phase encoded*.

ISO 4057:1986, *Information processing — Data interchange on 6,30 mm (0.25 in) magnetic tape cartridge 63 bpm (1 600 bpi) phase-encoded*.

ISO 5652:1984, *Information processing — 9-track, 12,7 mm (0.5 in) wide magnetic tape for information interchange -- Format and recording, using group coding at 246 cpm (6 250 cpi)*.

ISO 5654/1:1984, *Information processing — Data interchange on 200 mm (8 in) flexible disk cartridges using two-frequency recording at 13 262 f/track, 1,9 tpm (48 tpi), on one side — Part 1: Dimensional, physical and magnetic characteristics*.

ISO 5654/2:1985, *Information processing — Data interchange on 200 mm (8 in) flexible disk cartridges using two-frequency recording at 13 262 f/track, 1,9 tpm (48 tpi), on one side — Part 2: Track format*.

ISO 6596/1:1985, *Information processing — Data interchange on 130 mm (5.25 in) flexible disk cartridges using two-frequency recording at 7 958 f/track, 1,9 tpm (48 tpi), on one side — Part 1: Dimensional, physical and magnetic characteristics*.

ISO 6596/2:1985, *Information processing — Data interchange on 130 mm (5.25 in) flexible disk cartridges using two-frequency recording at 7 958 f/track, 1,9 tpm (48 tpi) on one side — Part 2: Track format*.

ISO 7065/1:1985, *Information processing — Data interchange on 200 mm (8 in) flexible disk cartridges using modified frequency modulation recording at 13 262 f/track, 1,9 tpm (48 tpi), on both sides — Part 1: Dimensional, physical and magnetic characteristics*.

ISO 7065/2:1985, *Information processing — Data interchange on 200 mm (8 in) flexible disk cartridges using modified frequency modulation recording at 13 262 f/track, 1,9 tpm (48 tpi), on both sides — Part 2: Track format*.

ISO 7487/1:1985, *Information processing — Data interchange on 130 mm (5.25 in) flexible disk cartridges using modified frequency modulation recording at 7 958 f/track, 1,9 tpm (48 tpi), on both sides — Part 1: Dimensional, physical and magnetic characteristics*.

ISO 7487/2:1985, *Information processing — Data interchange on 130 mm (5.25 in) flexible disk cartridges using modified frequency modulation recording at 7 958 f/track, 1,9 tpm (48 tpi), on both sides — Part 2: Track format*.

ISO 7487/3:1986, *Information processing — Data interchange on 130 mm (5.25 in) flexible disk cartridges using modified frequency modulation recording at 7 958 f/track, 1,9 tpm (48 tpi), on both sides — Part 3: Track format B*.

ISO 8063/1:1986, *Information processing — Data interchange on 6,30 mm (0.25 in) wide magnetic tape cartridge using IMFM recording at 252 f/track (6 400 fpi) — Part 1: Mechanical, physical and magnetic properties*.

ISO 8063/2:1986, *Information processing — Data interchange on 6,30 mm (0.25 in) wide magnetic tape cartridge using IMFM recording at 252 fipmm (6 400 fpi) — Part 2: Track format and method of recording for data interchange in start/stop mode.*

ISO 8378/1:1986, *Information processing — Data interchange on 130 mm (5.25 in) flexible disk cartridges using modified frequency modulation recording at 7 958 fiprad, 3,8 tpmm (96 tpi), on both sides — Part 1: Dimensional, physical and magnetic characteristics.*

ISO 8378/2:1986, *Information processing — Data interchange on 130 mm (5.25 in) flexible disk cartridges using modified frequency modulation recording at 7 958 fiprad, 3,8 tpmm (96 tpi), on both sides — Part 2: Track format A.*

ISO 8378/3:1986, *Information processing — Data interchange on 130 mm (5.25 in) flexible disk cartridges using modified frequency modulation recording at 7 958 fiprad, 3,8 tpmm (96 tpi), on both sides — Part 3: Track format B.*

ISO 8462/1:1986, *Information processing — Data Interchange on 6,30 mm (0.25 in) magnetic tape cartridge using GCR recording at 394 fipmm (10 000 fpi), 39 cpmm (1 000 cpi) — Part 1: Mechanical, physical and magnetic properties.*

ISO 8462/2:1986, *Information processing — Data Interchange on 6,30 mm (0.25 in) magnetic tape cartridges using GCR recording at 394 fipmm (10 000 fpi), 39 cpmm (1 000 cpi) — Part 2: Streaming mode.*

ISO 8630/1:1987, *Information processing -- Data interchange on 130 mm (5.25 in) flexible disk cartridges using modified frequency modulation recording at 13 262 fiprad, on 80 tracks on each side — Part 1: Dimensional, physical and magnetic characteristics.*

ISO 8630/2:1987, *Information processing -- Data interchange on 130 mm (5.25 in) flexible disk cartridges using modified frequency modulation recording at 13 262 fiprad, on 80 tracks on each side — Part 2: Track format A for 77 tracks.*

ISO 8630/3:1987, *Information processing -- Data interchange on 130 mm (5.25 in) flexible disk cartridges using modified frequency modulation recording at 13 262 fiprad, on 80 tracks on each side — Part 3: Track format B for 80 tracks.*

ISO 8860/1:1987, *Information processing - Data interchange on 90 mm (3.5 in) flexible disk cartridges using modified frequency modulation recording at 7 958 fiprad on 80 tracks on each side — Part 1: Dimensional, physical and magnetic characteristics.*

ISO 8860/2:1987, *Information processing -- Data interchange on 90 mm (3.5 in) flexible disk cartridges using modified frequency modulation recording at 7 958 fiprad on 80 tracks on each side — Part 2: Track format.*

3 Definitions, abbreviations and conventions

3.1 Definitions

For the purposes of this International Standard the following definitions apply.

3.1.1 byte: In this standard, this term indicates an 8-bit (octet) byte.

3.1.2 command descriptor block (CDB): The structure used to communicate requests from an initiator to a target.

3.1.3 connect: The function that occurs when an initiator selects a target to start an operation.

3.1.4 disconnect: The function that occurs when a target releases control of the SCSI bus, allowing it to go to the BUS FREE phase.

3.1.5 initiator: An SCSI device (usually a host system) that requests an operation to be performed by another SCSI device.

3.1.6 INTERMEDIATE status: A status code sent from a target to an initiator upon completion of each command in a set of linked commands except the last command in the set.

3.1.7 logical unit: A physical or virtual device addressable through a target.

3.1.8 logical unit number: An encoded three-bit identifier for the logical unit.

3.1.9 one: A true signal value.

3.1.10 peripheral device: A peripheral that can be attached to an SCSI device (e.g., magnetic-disk, printer, optical-disk, or magnetic-tape).

3.1.11 reconnect: The function that occurs when a target selects an initiator to continue an operation after a disconnect.

3.1.12 reserved: The term used for bits, bytes, fields, and code values that are set aside for future standardization.

3.1.13 SCSI address: The octal representation of the unique address (0-7) assigned to an SCSI device. This address would normally be assigned and set in the SCSI device during system installation.

3.1.14 SCSI ID: The bit-significant representation of the SCSI address referring to one of the signal lines DB(7-0).

3.1.15 SCSI device: A host computer adapter or a peripheral controller or an intelligent peripheral that can be attached to the SCSI bus.

3.1.16 signal assertion: The act of driving a signal to the true state.

3.1.17 signal negation: The act of driving a signal to the false state or allowing the cable terminators to bias the signal to the false state (by placing the driver in the high impedance condition).

3.1.18 signal release: The act of allowing the cable terminators to bias the signal to the false state (by placing the driver in the high impedance condition).

3.1.19 status: One byte of information sent from a target to an initiator upon completion of each command.

3.1.20 target: An SCSI device that performs an operation requested by an initiator.

3.1.21 vendor unique: In this standard, this term indicates bits, fields, or code values that are vendor specific and are not defined by this standard.

3.1.22 zero: A false signal value.

3.2 Abbreviations

3.2.1 LSB. Least significant byte.

3.2.2 LUN. Logical unit number.

3.2.3 MSB. Most significant byte.

3.3 Editorial conventions

Certain words, terms, and phrases used in this standard have a specific meaning beyond the normal English meaning. These words, terms, and phrases are defined either in the definitions (see 3.1) or in the text where they first appear (e.g., Arbitration Delay, see 4.7.1). Names of signals, phases, conditions, messages, commands, statuses, and sense keys are in all uppercase (e.g., REQUEST SENSE). Lower case is used for words having the normal English meaning.

Hexidecimal numbers are indicated by the digits 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E and F followed by the lower case "h" (e. g. 3F6Ah). All other numbers are decimal values.

4 Physical Characteristics

This section contains the physical definition of the SCSI. The connectors, cables, signals, terminators, and bus timing needed to implement SCSI are covered.

4.1 Physical Description

SCSI devices are daisy-chained together using a common cable. Both ends of the cable are terminated. All signals are common between all SCSI devices. Two driver/receiver alternatives are available:

- a) Single-ended drivers and receivers, which allow a maximum cable length of 6 m (primarily for connection within a cabinet).
- b) Differential drivers and receivers, which allow a maximum cable length of 25 m (primarily for connection outside of a cabinet).

4.2 Cable Requirements

An ideal impedance match with cable terminators implies a cable characteristic impedance of $132\ \Omega$ (single-ended option) or $122\ \Omega$ (differential option). In general, cables with this high of a characteristic impedance are not available; however, impedances that are somewhat lower are satisfactory. A characteristic impedance of $100\ \Omega \pm 10\%$ is recommended for unshielded flat or twisted pair ribbon cable. A characteristic impedance greater than $90\ \Omega$ is preferred for shielded cables; however, most available cables have a somewhat lower characteristic impedance. To minimize discontinuities and signal reflections, cables of different impedances should not be used in the same bus. Implementations may require trade-offs in shielding effectiveness, cable length, the number of loads, transfer rates, and cost to achieve satisfactory system operation.

A minimum conductor size of nominal cross section $0,080\ 42\ \text{mm}^2$ shall be employed to minimize noise effects and ensure proper distribution of optional terminator power.

4.2.1 Single-Ended Cable

A 50-conductor flat cable or 25-signal twisted-pair cable shall be used. The maximum cable length shall be 6 m.

A stub length of no more than 0,1 m is allowed off the mainline interconnection within any connected equipment.

SCSI bus termination may be internal to the SCSI devices that are at the ends of the cable.

4.2.2 Differential Cable

A 50-conductor cable or 25-signal twisted-pair cable shall be used. The maximum cable length shall be 25 m.

A stub length of no more than 0,2 m is allowed off the mainline interconnection within any connected equipment.

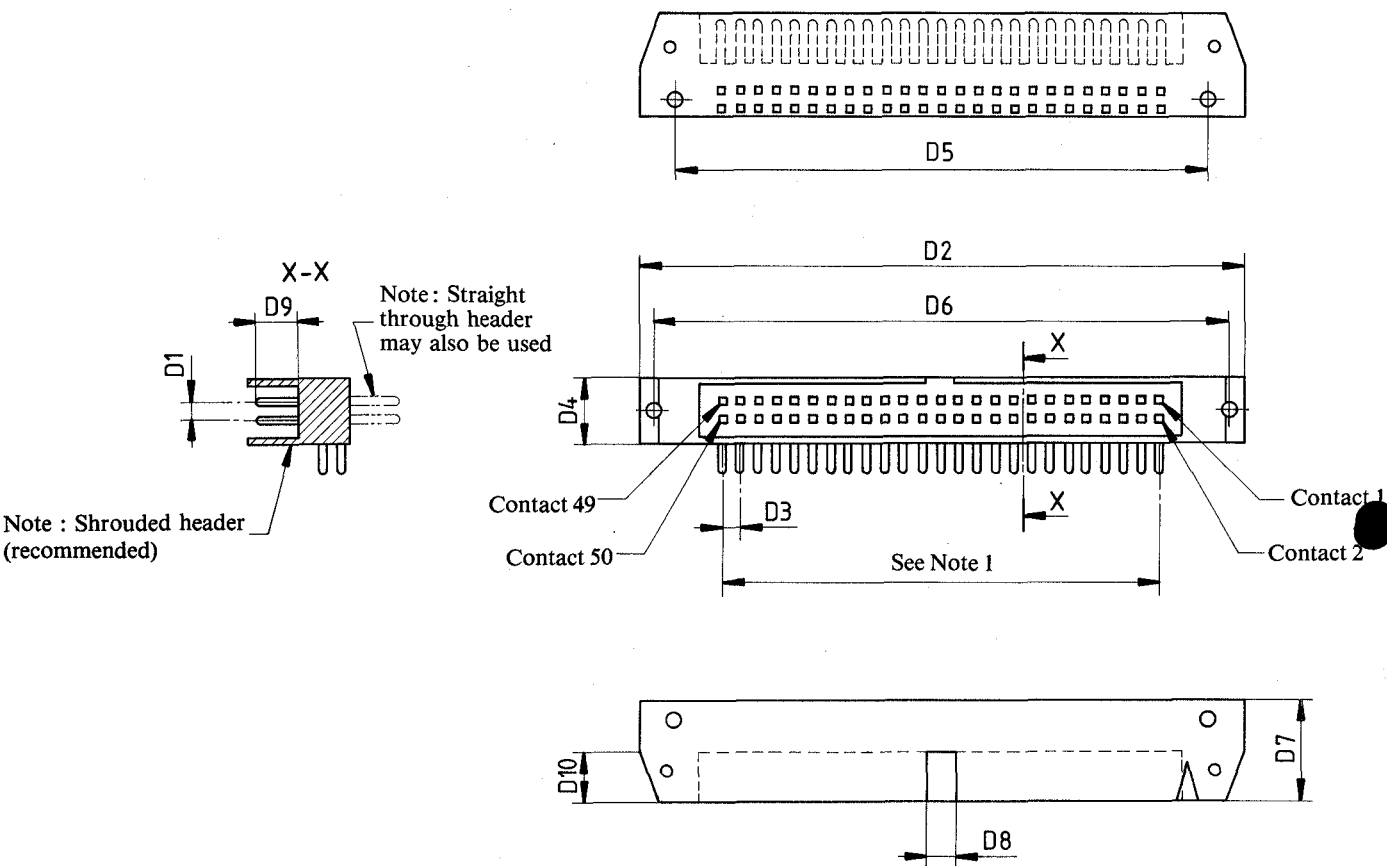
SCSI bus termination may be internal to the SCSI devices that are at the ends of the cable.

4.3 Connector Requirements

Nonshielded connectors are specified. The nonshielded connectors are typically used for in-cabinet applications. Annex D defines recommended shielded connectors and their pin assignments. These connectors are typically used for external applications where electromagnetic compatibility (EMC) and electrostatic discharge (ESD) protection may be required. Either type of connector may be used with the single-ended or differential drivers.

The nonshielded SCSI device connector (figure 1) shall be a 50-conductor connector consisting of two rows of 25 male contacts with adjacent contacts 2,54 mm (0,1 in) apart. A shroud and header body should be used. The nonmating portion of the connector is shown for reference only.

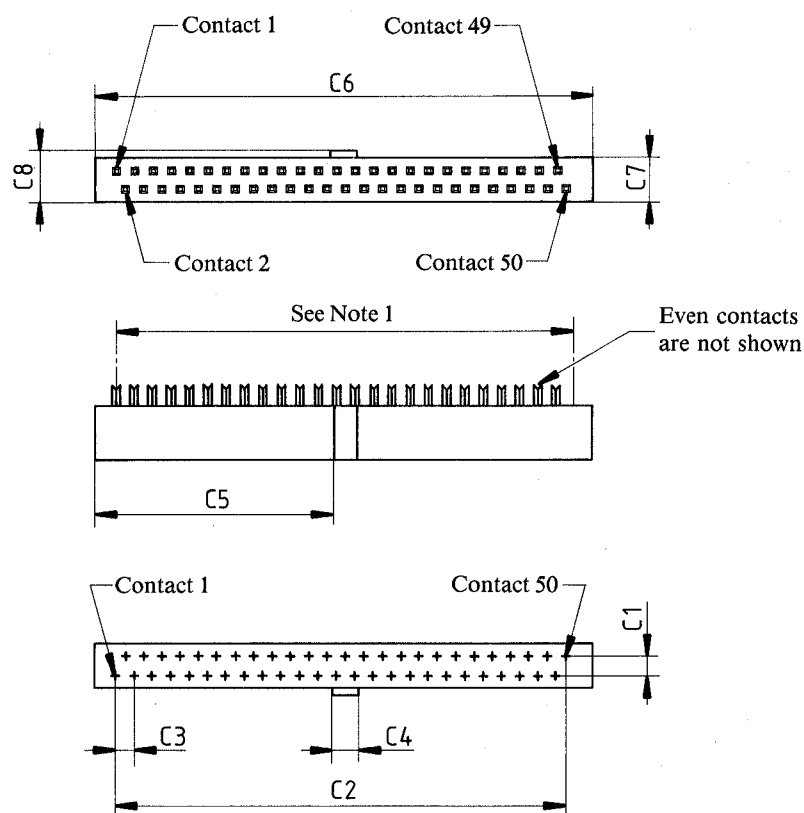
The nonshielded cable connector (figure 2) shall be a 50-conductor connector consisting of two rows of 25 female contacts with adjacent contacts 2,54 mm (0,1 in) apart. It is recommended that keyed connectors be used.



Dimensions	mm	in	Comments
D1	2,54	0,100	Reference only
D2	82,50	3,260	
D3	2,54	0,100	
D4	8,89	0,350	Reference only
D5	72,64	2,860	Reference only
D6	78,74	3,100	Reference only
D7	13,94	0,549	Reference only
D8	4,19 ± 0,25	0,165 ± 0,01	
D9	6,10	0,240	
D10	6,60	0,260	Reference only

- NOTES :
- 1) Two rows of twenty five contacts on 2,54 mm (0,100 in) spacing = 60,96 mm (2,400 in).
 - 2) Tolerances ± 0,127 mm (0,005 in) non-cumulative, unless specified otherwise.

Figure 1 — Nonshielded SCSI Device Connector



Dimensions	mm	in	Comments
C1	2,54	0,100	
C2	60,96	2,400	
C3	2,54	0,100	
C4	3,30	0,130	
C5	32,39	1,275	
C6	68,07	2,680	
C7	6,10	0,240	
C8	7,62	0,300	max.

NOTES :

- 1) Fifty contacts on 1,27 mm (0,05 in) staggered spacing = 62,23 mm (2,450 in) [reference only].
- 2) Tolerances $\pm 0,127$ mm (0,005 in) non-cumulative, unless specified otherwise.
- 3) Connector cover and strain relief are optional.

Figure 2 — Nonshielded SCSI Cable Connector

The unshielded connector contact assignments shall be as shown in table 1 for single-ended drivers.

Table 1 – Single-Ended Connector Contact Assignments

Signal	Contact Number	Signal	Contact Number
GROUND	1	-DB(0)	2
GROUND	3	-DB(1)	4
GROUND	5	-DB(2)	6
GROUND	7	-DB(3)	8
GROUND	9	-DB(4)	10
GROUND	11	-DB(5)	12
GROUND	13	-DB(6)	14
GROUND	15	-DB(7)	16
GROUND	17	-DB(P)	18
GROUND	19	GROUND	20
GROUND	21	GROUND	22
GROUND	23	GROUND	24
OPEN	25	TERMPWR	26
GROUND	27	GROUND	28
GROUND	29	GROUND	30
GROUND	31	-ATN	32
GROUND	33	GROUND	34
GROUND	35	-BSY	36
GROUND	37	-ACK	38
GROUND	39	-RST	40
GROUND	41	-MSG	42
GROUND	43	-SEL	44
GROUND	45	-C/D	46
GROUND	47	-REQ	48
GROUND	49	-I/O	50

NOTE: The minus sign next to the signals indicates active low.

The unshielded connector contact assignments shall be as shown in table 2 for differential drivers.

Table 2 – Differential Connector Contact Assignments

Signal	Contact Number	Signal	Contact Number
SHIELD GROUND	1	GROUND	2
+DB(0)	3	-DB(0)	4
+DB(1)	5	-DB(1)	6
+DB(2)	7	-DB(2)	8
+DB(3)	9	-DB(3)	10
+DB(4)	11	-DB(4)	12
+DB(5)	13	-DB(5)	14
+DB(6)	15	-DB(6)	16
+DB(7)	17	-DB(7)	18
+DB(P)	19	-DB(P)	20
DIFFSENS	21	GROUND	22
GROUND	23	GROUND	24
TERMPWR	25	TERMPWR	26
GROUND	27	GROUND	28
+ATN	29	-ATN	30
GROUND	31	GROUND	32
+BSY	33	-BSY	34
+ACK	35	-ACK	36
+RST	37	-RST	38
+MSG	39	-MSG	40
+SEL	41	-SEL	42
+C/D	43	-C/D	44
+REQ	45	-REQ	46
+I/O	47	-I/O	48
GROUND	49	GROUND	50

NOTE: SHIELD GROUND is optional on some cables. Some shielded flat ribbon cables use contact 1 as a connection to the shield.

4.4 Electrical Description

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.

4.4.1 Single-Ended Alternative

All assigned signals shall be terminated with $220\ \Omega$ to +5 V (nominal) and $330\ \Omega$ to ground at each end of the cable. (See figure 3.) All signals shall use open-collector or three-state drivers.

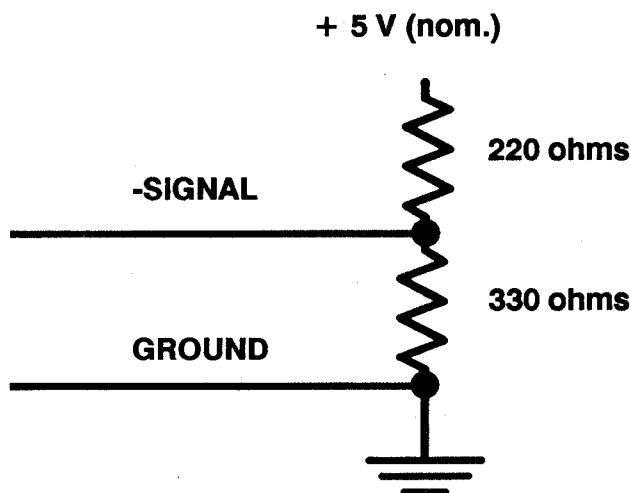


Figure 3 — Termination for Single-Ended Devices

4.4.1.1 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 V dc to 0,4 V dc
- Minimum driver output capability = 48 mA (sinking) at 0,5 V dc
- Signal negation = 2,5 V dc to 5,25 V dc

4.4.1.2 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 V dc to 0,8 V dc
- Maximum total input load = -0,4 mA at 0,4 V dc
- Signal false = 2,0 V dc to 5,25 V dc
- Minimum input hysteresis = 0,2 V dc

4.4.2 Differential Alternative.

All signals consist of two lines denoted +SIGNAL and -SIGNAL. A signal is true when +SIGNAL is more positive than -SIGNAL, and a signal is false when -SIGNAL is more positive than +SIGNAL. All assigned signals shall be terminated at each end of the cable as shown in figure 4.

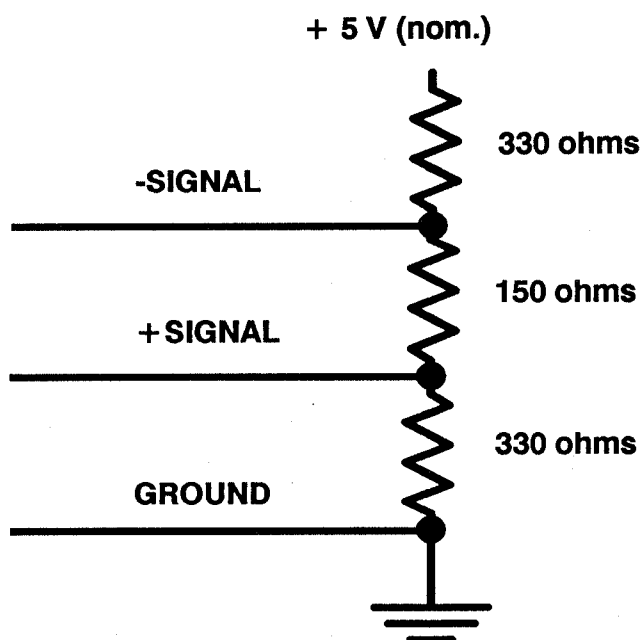


Figure 4 – Termination for Differential Devices

As an option, the DIFFSENS signal of the connector is reserved for an active high enable for the differential drivers. If a single-ended device or terminator is inadvertently connected, this signal is grounded, disabling the drivers. (See figure 5). This will prevent damage should single-ended and differential drivers be connected to the same cable.

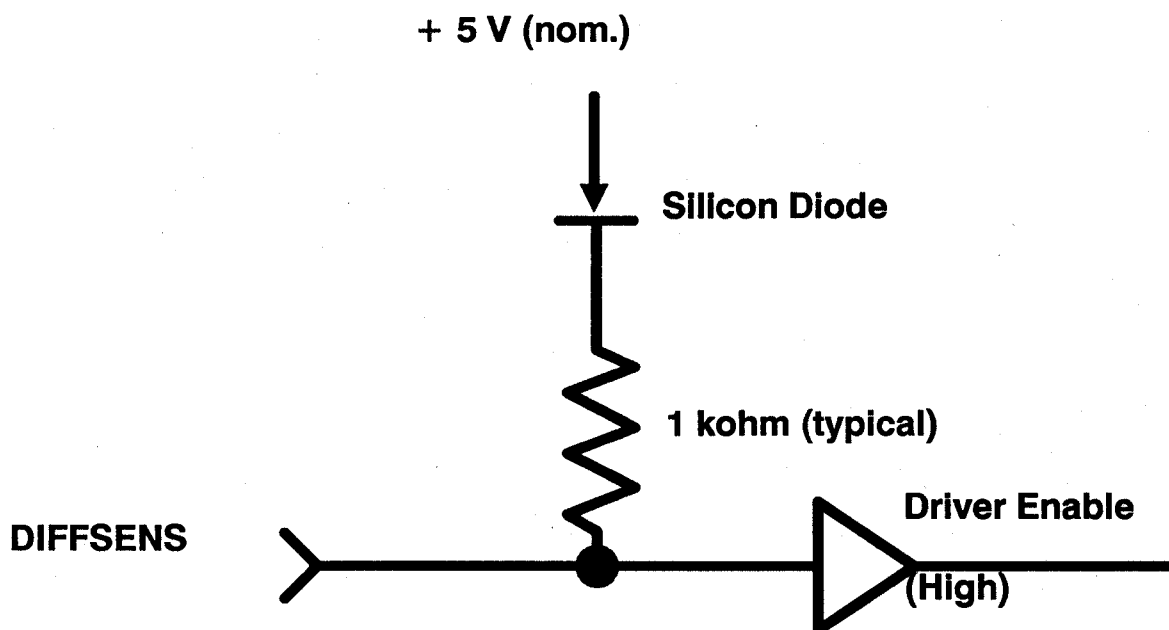


Figure 5 – Differential Driver Protection Circuit (Opt.)

4.4.2.1 Output Characteristics

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

V_{OL} (Low-level output voltage) = 2,0 V max. at I_{OL} (Low-level output current) = 55 mA.
 V_{OH} (High-level output voltage) = 3,0 V min. at I_{OH} (High-level output current) = -55 mA.
 V_{OD} (Differential voltage) = 1,0 V min. with common-mode voltage ranges from -7 V dc to +12 V dc.

V_{OL} and V_{OH} shall be as measured between the output terminal and the SCSI device's logic ground reference.

The output characteristics shall additionally conform to ISO 8482.

4.4.2.2 Input Characteristics

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

I_I (Input current on either input) = $\pm$ 2,0 mA max.

NOTE: These characteristics include both receivers and passive drivers.

This requirement shall be met with the input voltage varying between -7 V dc and +12 V dc, with power on or off, and with the hysteresis equaling 35 mV, min.

The input characteristics shall additionally conform to ISO 8482.

4.4.3 Terminator Power (Optional)

Single-ended SCSI devices providing terminator power (TERMPWR) shall have the following characteristics:

V_{Term} = 4,0 V dc to 5,25 V dc
800 mA min. source drive capability
1,0 mA max. sink capability (except for the purposes of providing power to an internal terminator)
with 1,0 A recommended current limiting (e.g., a fuse).

Differential SCSI devices providing terminator power (TERMPWR) shall have the following characteristics:

V_{Term} = 4,0 V dc to 5,25 V dc
600 mA min. source drive capability
1,0 mA max. sink capability (except for the purposes of providing power to an internal terminator) with 1,0 A recommended current limiting (e.g., a fuse).

The use of keyed connectors is recommended in SCSI devices that provide terminator power to prevent accidental grounding or misconnection of terminator power.

SCSI devices that supply terminator power shall do so through a diode or similar semiconductor that prevents the back-flow of power to the SCSI device.

4.5 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 6.

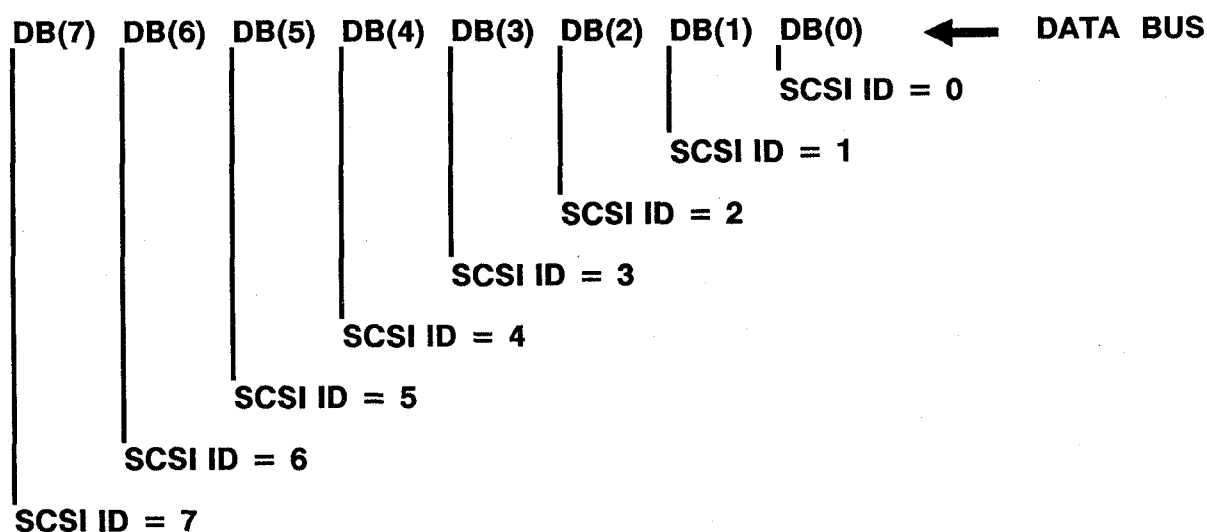
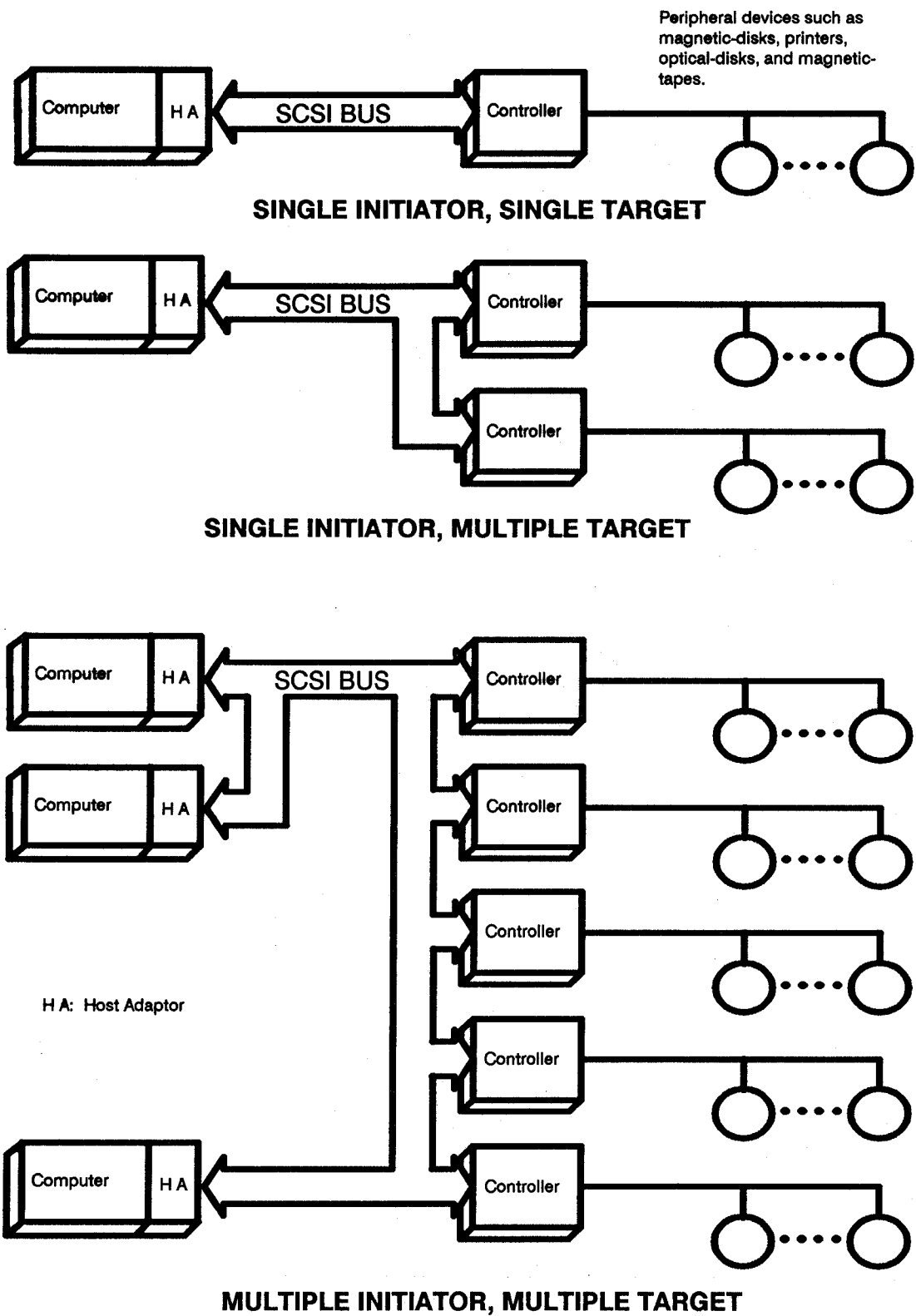


Figure 6 – SCSI ID Bits

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 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 7.



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.

4.6 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 a target to indicate a request for a REQ/ACK data 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 as 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.

4.6.1 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 International 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 data transfer.

4.6.2 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.

4.6.3 Signal Sources

Table 3 indicates 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. The RST signal may be sourced by any SCSI device at any time.

Table 3 – Signal Sources

Signals					
C/D, I/O,					
Bus Phase	BSY	SEL	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 be 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 the signal is driven, it shall be driven only by the active target.

4.7 SCSI Bus Timing

Unless otherwise indicated, the delay-time measurements for each SCSI device, shown in 4.7.1 through 4.7.14, 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.

4.7.1 Arbitration Delay (2,2 μ s)

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.

4.7.2 Assertion Period (90 ns)

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.

4.7.3 Bus Clear Delay (800 ns)

The maximum time for an SCSI device to stop driving all bus signals after:

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

For condition (a) above, the maximum time for an SCSI device to clear the bus is 1200 ns 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.

4.7.4 Bus Free Delay (800 ns)

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.

4.7.5 Bus Set Delay (1,8 μ s)

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.

4.7.6 Bus Settle Delay (400 ns)

The time to wait for the bus to settle after changing certain control signals as called out in the protocol definitions.

4.7.7 Cable Skew Delay (10 ns)

The maximum difference in propagation time allowed between any two SCSI bus signals when measured between any two SCSI devices.

4.7.8 Data Release Delay (400 ns)

The maximum time for an initiator to release the DATA BUS signals following the transition of the I/O signal from false to true.

4.7.9 Deskew Delay (45 ns)

The minimum time required for deskew of certain signals.

4.7.10 Hold Time (45 ns)

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.

4.7.11 Negation Period (90 ns)

The minimum time that a target shall negate REQ while using synchronous data transfers. Also, the minimum time that an initiator shall negate ACK while using synchronous data transfers.

4.7.12 Reset Hold Time (25 μ s)

The minimum time for which RST is asserted. There is no maximum time.

4.7.13 Selection Abort Time (200 μ s)

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 timeout period; see 5.1.3.5 and 5.1.4.2 for a complete description.

4.7.14 Selection Timeout Delay (250 ms, 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. The specifications for the peripheral devices shall be consulted for the actual timing requirements.

4.7.15 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. (See 5.1.5.2 and 5.5.5.)

5 Logical characteristics

5.1 SCSI Bus Phases

The SCSI architecture 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 descriptions, 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. 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:

- a) 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 minimum of 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.)
- b) 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.
- c) Following the bus free delay in step b), 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 SCSI ID) if more than a bus set delay has passed since the BUS FREE phase was last observed. (Note: There is no maximum delay before asserting BSY and the SCSI ID following the bus free delay in step b) 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.)
- d) 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 bit 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 a). 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 a).

- e) 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.

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).

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 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 not asserting the I/O signal. Except in certain single initiator environments with initiators employing the single initiator option (see 5.1.3.4), the initiator shall set the DATA BUS to a value which is the OR of its SCSI ID bit and the target's SCSI ID bit. 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 bit 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:

- a) Optionally, the initiator shall assert the RST signal (see 5.2.2).
- b) 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 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 m 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 a minimum of a selection timeout period and there has been no BSY response from the initiator:

- a) Optionally, the target shall assert the RST signal (see 5.2.2).
- b) 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

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 4.) 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.

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 con-

tinuously 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.

Table 4 – Information Transfer Phases

Signal			Phase Name	Direction Of Transfer	Comment
MSG	C/D	I/O			
0	0	0	DATA OUT	Initiator to target	Data Phase
0	0	1	DATA IN	Initiator from target	
0	1	0	COMMAND	Initiator to target	
0	1	1	STATUS	Initiator from target	
1	0	0	*		
1	0	1	*		Message Phase
1	1	0	MESSAGE OUT	Initiator to target	
1	1	1	MESSAGE IN	Initiator from target	

Key: 0 = False, 1 = True, * = Reserved for future standardization.

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, 5.5.5). 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 one 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. Multiple messages may be sent during either phase. The first byte transferred in either of these phases shall be either a single-byte message or the first byte 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 message(s) 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 message(s) 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, unless an error occurs (see MESSAGE REJECT, 5.5.2).

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(s) 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:

- a) The BSY, SEL, REQ, and ACK signals shall not change.
- b) 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.
- c) 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 signal 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. 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.

The effect of the RESET condition on uncompleted commands, SCSI device reservations, and SCSI device operating modes is determined by whether the SCSI device has implemented the "hard" RESET option or the "soft" RESET option (one of which shall be implemented) as defined in 5.2.2.1 and 5.2.2.2.

5.2.2.1 "Hard" RESET Option.

SCSI devices that implement the "hard" RESET option, upon detection of the RESET condition, shall

- a) clear all uncompleted commands;
- b) release all SCSI device reservations;
- c) return any SCSI device operating modes (MODE SELECT, PREVENT/ALLOW MEDIUM REMOVAL commands, etc) to their default conditions.

5.2.2.2 "Soft" RESET Option.

SCSI devices that implement the "soft" RESET option, upon detection of the RESET condition, shall

- a) attempt to complete any uncompleted commands that were fully identified;
- b) preserve all SCSI device reservations;
- c) pPreserve any SCSI device operating modes (MODE SELECT, PREVENT/ALLOW MEDIUM REMOVAL commands, etc).

The "soft" RESET option allows a single initiator to reset the SCSI bus without disturbing the operation of other initiators in a multiple initiator system. To ensure proper operation the following conditions shall be met:

- a) An initiator shall not consider a command to be fully identified until the IDENTIFY message is sent to the target and the target responds by changing to any other information transfer phase and requests that at least one byte be transferred.
- b) A target shall consider a command to be fully identified when it successfully receives the IDENTIFY message.
- c) If an initiator selects a logical unit for which there already is an active command for the same initiator, the target shall clear the original command and perform the new command.
- d) If a target reselects an initiator to continue a command for which the initiator has no record, the initiator shall abort that command by sending the ABORT message.
- e) An initiator shall consider a command to be completed when it negates ACK for a successfully received COMMAND COMPLETE message.
- f) A target shall consider a command to be completed when it detects the false transition of ACK for the COMMAND COMPLETE message with the ATN signal false.
- g) An initiator shall not negate ACK for the SAVE DATA POINTER message until it has actually saved the data pointer for the operation.
- h) A target shall consider the data pointer to be saved when it detects the false transition of ACK for the SAVE DATA POINTER message with the ATN signal false.
- i) If the RESET condition occurs between the time that the target asserts REQ for the SAVE DATA POINTER message and it detects the false transition of ACK, the target shall terminate the command with a CHECK CONDITION status. If extended sense is implemented, the target shall set the sense key to ABORTED COMMAND. This is necessary because the target cannot determine whether the data pointer has actually been saved.

If the ATN signal is true in conditions f) or h), the target would normally switch to MESSAGE OUT phase and attempt to transfer a message byte. If the RESET condition occurs before it is able to successfully receive the message byte, the target shall assume that the initiator may not have successfully received the COMMAND COMPLETE message or the SAVE DATA POINTER message. In the case of COMMAND COMPLETE message, the target shall reselect the initiator and attempt to send the COMMAND COMPLETE message again. In the case of the SAVE DATA POINTER message, the target shall reselect the initiator and terminate the command as described in condition i).

5.3 SCSI Bus Phase Sequences

The order in which phases are used on the SCSI bus follows a prescribed 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 the ARBITRATION phase is not implemented, the allowable sequences shall be as shown in figure 8. 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).

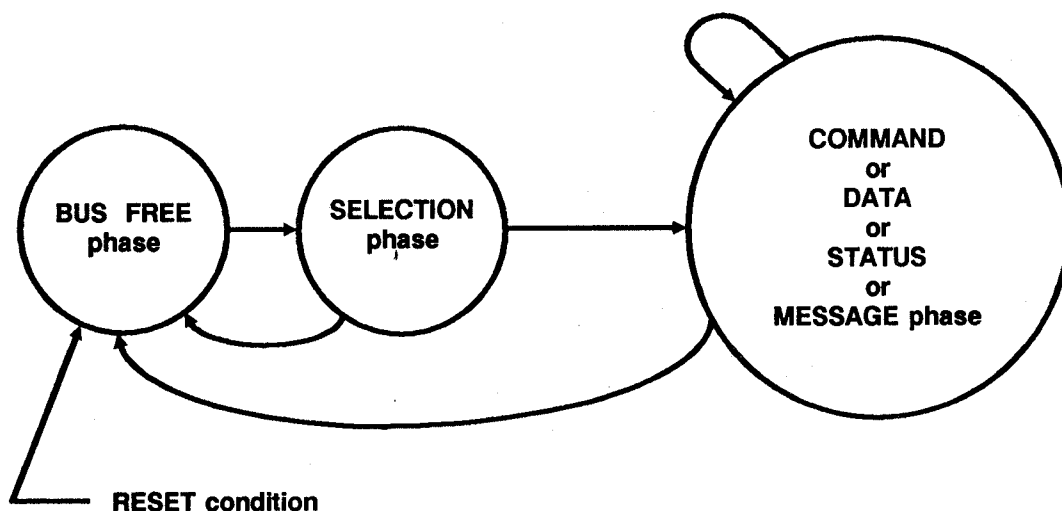


Figure 8 — Phase Sequence without Arbitration

5.3.2 Arbitrating Systems

In systems where the ARBITRATION phase is implemented, the allowable sequences shall be as shown in figure 9. 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).

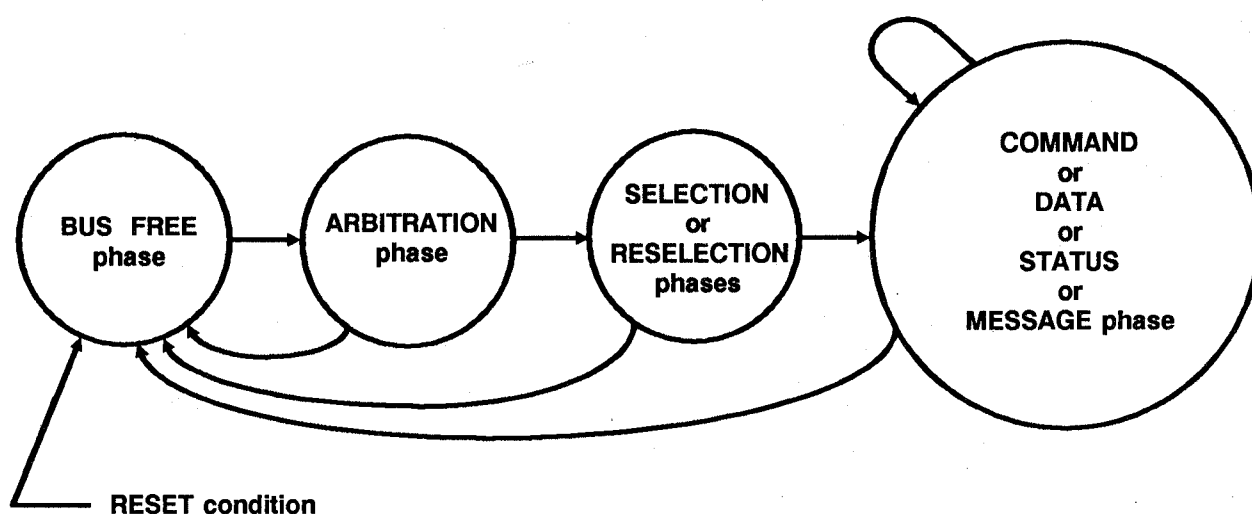


Figure 9 — Phase Sequence with Arbitration

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).

5.4 SCSI Pointers

Consider the system shown in figure 10 in which an initiator and target communicate on the SCSI bus in order to execute a command.

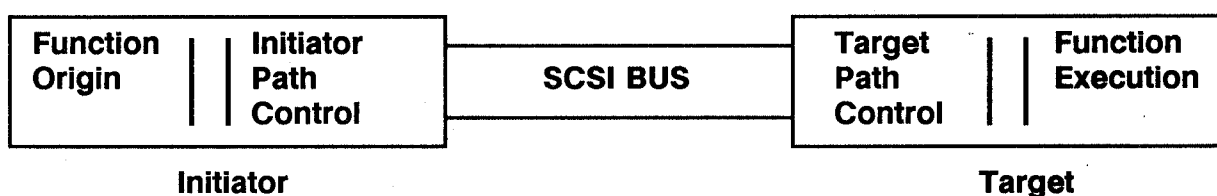


Figure 10 — Simplified SCSI System

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 (see 6.2) 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 message system allows communication between an initiator and target for the purpose of physical path management.

5.5.1 Message Protocol

All SCSI devices shall implement the COMMAND COMPLETE message. A functional SCSI device can be constructed without using any of the other messages if the logical unit number is specified in the command descriptor block. The remainder of this section deals with the additional requirements on SCSI devices that support messages other than COMMAND COMPLETE.

SCSI devices indicate their ability to accommodate more than the COMMAND COMPLETE message by asserting or responding to the ATN signal. The initiator indicates this in the SELECTION phase by asserting ATN prior to the SCSI bus condition of SEL true, and BSY false. The target indicates its ability to accommodate more messages by responding to the ATTENTION condition with the MESSAGE OUT phase after going through the SELECTION phase.

For SCSI devices that support messages other than COMMAND COMPLETE, the first message sent by the initiator after the SELECTION phase shall be the IDENTIFY message. This allows the establishment of the physical path for a particular logical unit specified by the initiator. After the RESELECTION phase, the target's first message shall be IDENTIFY. This allows the physical path to be reestablished for the target's specified logical unit number. Under some exceptional conditions, an initiator may send the ABORT message or the BUS DEVICE RESET message instead of the IDENTIFY message, as the first message. Only one logical unit number shall be identified for any one selection sequence; a second IDENTIFY message with a new logical unit number shall not be issued before the SCSI bus has been released (BUS FREE phase).

Whenever a physical path is established in an initiator that can accommodate disconnection and reconnection, the initiator shall ensure that the active pointers of the physical path are equal to the saved pointers for that particular logical unit number. (An implied restore pointers operation occurs as a result of connect or reconnect.)

SCSI devices that implement any message other than the COMMAND COMPLETE message shall also implement the MESSAGE REJECT message.

5.5.2 Messages

The single byte messages (table 5) are listed along with their code values and their definitions.

Table 5 – Message Codes

Code	Type	Description	Direction
00h	M	COMMAND COMPLETE	In
01h	O	EXTENDED MESSAGE	In Out
02h	O	SAVE DATA POINTER	In
03h	O	RESTORE POINTERS	In
04h	O	DISCONNECT	In
05h	O	INITIATOR DETECTED ERROR	Out
06h	O	ABORT	Out
07h	O	MESSAGE REJECT	In Out
08h	O	NO OPERATION	Out
09h	O	MESSAGE PARITY ERROR	Out
0Ah	O	LINKED COMMAND COMPLETE	In
0Bh	O	LINKED COMMAND COMPLETE (WITH FLAG)	In
0Ch	O	BUS DEVICE RESET	Out
0Dh – 7Fh	R	Reserved Codes	
80h – FFh	O	IDENTIFY	In Out

Key: In = Target to initiator, Out = Initiator to target.

COMMAND COMPLETE 00h (Mandatory). This message is sent from a target to an initiator to indicate that the execution of a command (or series of linked commands) has terminated and that valid status has been sent to the initiator. After successfully sending this message, the target shall go to the BUS FREE phase by releasing BSY. The command may have been executed successfully or unsuccessfully as indicated in the status.

EXTENDED MESSAGE 01h (Optional). This message is sent from either the initiator or the target as the first byte of a multiple-byte message. (See 5.5.3 for descriptions of extended messages.)

SAVE DATA POINTER 02h (Optional). This message is sent from a target 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.)

RESTORE POINTERS 03h (Optional). This message is sent from a target 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 04h (Optional). This message is sent from a target to inform an initiator that the present physical path is going to be broken (the target 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 the BUS FREE phase (other than as a result of a RESET condition) without first receiving a DISCONNECT or COMMAND COMPLETE message, the initiator shall consider this as a catastrophic error condition. If the target intentionally creates this condition, the target shall clear the current command. This message shall not cause the initiator to save the data pointer. If DISCONNECT messages are used to break a long data transfer into two or more shorter transfers, then a SAVE DATA POINTER should be issued before each DISCONNECT message.

INITIATOR DETECTED ERROR 05h (Optional). This message is sent from an initiator to inform a target that an error (e.g., parity error) has occurred that does not preclude the target from retrying 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 state.

ABORT 06h (Optional). This message is sent from the initiator to the target to clear the present operation. If a logical unit has been identified, all pending data and status for the issuing initiator from the affected logical unit shall be cleared, and the target shall go to the BUS FREE phase. Pending data and status for other initiators shall not be cleared. If a logical unit has not been identified, the target shall go to the BUS FREE phase. No status or ending message shall be sent for the operation. It is not an error to issue this message to an logical unit that is not currently performing an operation for the initiator.

MESSAGE REJECT 07h (Optional). This message is sent from either the initiator or target to indicate that the last message it received was inappropriate or has not been implemented.

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.

This message shall be implemented if any other optional messages are implemented.

NO OPERATION 08h (Optional). This message is sent from an initiator in response to a target's request for a message when the initiator does not currently have any other valid message to send.

MESSAGE PARITY ERROR 09h (Optional). This message is sent from the initiator to the target to indicate that one or more bytes in the last message it received had a parity error.

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 has the parity error. This provides an interlock so that the target can determine which message has the parity error.

LINKED COMMAND COMPLETE 0Ah (Optional). This message is sent from a target to an initiator to indicate that the execution of a linked command has completed and that status has been sent. The initiator shall then set the pointers to the initial state for the next linked command.

LINKED COMMAND COMPLETE (WITH FLAG) 0Bh (Optional). This message is sent from a target to an initiator to indicate that the execution of a linked command (with the flag bit set to one) has completed and that status has been sent. The initiator shall then set the pointers to the initial state of the next linked command. Typically this message would be used to cause an interrupt in the initiator between two linked commands.

BUS DEVICE RESET 0Ch (Optional). This message is sent from an initiator to direct a target to clear all current commands on that SCSI device. This message forces the SCSI device to an initial state with no operations pending for any initiator. Upon recognizing this message, the target shall go to the BUS FREE phase.

Reserved 0Dh to 7Fh. These message codes are reserved for future standardization.

IDENTIFY 80h to FFh (Optional). These messages are sent by either the initiator or the target to establish the physical path connection between an initiator and target for a particular logical unit.

Bit 7. This bit is always set to one to distinguish these messages 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.

Bits 5-3. Reserved.

Bits 2-0. These bits specify a logical unit number in a target.

Only one logical unit number shall be identified for any one selection sequence; a second IDENTIFY message with a new logical unit number shall not be issued before the bus has been released (BUS FREE phase).

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 Extended Messages (Optional)

A value of one in the first byte of a message indicates the beginning of a multiple-byte extended message. The minimum number of bytes sent for an extended message is three. The extended message format and the extended message codes are shown in tables 6 and 7, respectively.

Table 6 – Extended Message Format

Byte	Value	Description
0	01h	Extended message
1	n	Extended message length
2	y	Extended message code
3 to n+1	x	Extended message arguments

The extended message length specifies the length in bytes of the extended message code plus the extended message arguments to follow. Therefore, the total length of the message is equal to the extended message length plus two. A value of zero for the extended message length indicates 256 bytes follow.

The extended message codes are listed in table 7. The extended message arguments are specified for the defined extended messages in 5.5.4 through 5.5.6.

Table 7 – Extended Message Codes

Code (y)	Description
00h	MODIFY DATA POINTER (Optional)
01h	SYNCHRONOUS DATA TRANSFER REQUEST (Optional)
02h	EXTENDED IDENTIFY (Optional)
03h – 7Fh	Reserved
80h – FFh	Vendor Unique

5.5.4 MODIFY DATA POINTER Message (Optional)

The MODIFY DATA POINTER message (table 8) is sent from the target to the initiator and requests that the signed argument be added (two's complement) to the value of the current data pointer.

Table 8 – MODIFY DATA POINTER

Byte	Value	Description
0	01h	Extended message
1	05h	Extended message length
2	00h	MODIFY DATA POINTER code
3	x	Argument
4	x	Argument
5	x	Argument
6	x	Argument (LSB)

5.5.5 SYNCHRONOUS DATA TRANSFER REQUEST Message (Optional)

A pair of SYNCHRONOUS DATA TRANSFER REQUEST messages (table 9) are exchanged between an initiator and a target whenever an SCSI device that can support synchronous data transfer recognizes that it has not communicated with the other SCSI device since receiving the last "hard" RESET condition or a BUS DEVICE RESET message. The SCSI devices may also exchange messages to establish synchronous data transfer when requested to do so. The message exchange establishes the transfer period and the REQ/ACK offset. The transfer period is the minimum time between leading edges of successive REQ pulses and of successive ACK pulses.

Table 9 – SYNCHRONOUS DATA TRANSFER REQUEST

Byte	Value	Description
0	01h	Extended message
1	03h	Extended message length
2	01h	SYNCHRONOUS DATA TRANSFER REQUEST code
3	m	Transfer period (m times 4 nanoseconds)
4	x	REQ/ACK offset

The REQ/ACK offset is the maximum number of REQ pulses that may be outstanding before its corresponding ACK pulse is received at the target. A REQ/ACK offset value of zero shall indicate asynchronous mode; a value of FFh shall indicate unlimited offset.

If the initiator recognizes that negotiation is required, it asserts ATN and, if the target implements message transfers, sends a SYNCHRONOUS DATA TRANSFER REQUEST message indicating an REQ/ACK offset and minimum transfer period. The REQ/ACK offset is chosen to prevent initiator buffer overflows, while the minimum transfer period is chosen to meet the data handling requirements of the initiator. The target responds in any of the following ways:

Target Response	Implied Agreement
a) REQ/ACK offset less than or equal to the requested value. Minimum transfer period equal to or greater than requested period.	REQ/ACK offset equal to target value. Minimum transfer period equal to the target value.
b) REQ/ACK offset equal to zero.	Asynchronous transfer.
c) MESSAGE REJECT.	Asynchronous transfer.

If the target recognizes that negotiation is required, it sends a SYNCHRONOUS DATA TRANSFER REQUEST message to the initiator. The REQ/ACK offset is selected to prevent buffer and offset counter overflows, while the minimum transfer period is chosen to meet the data handling requirements of the target. The initiator responds in any of the following ways if the target chooses an REQ/ACK offset equal to FFh:

Initiator Response	Implied Agreement
a) REQ/ACK offset equal to FFh. Minimum transfer period equal to or greater than requested period.	REQ/ACK offset unlimited. Minimum transfer period equal to the initiator value.
b) REQ/ACK offset equal to zero.	Asynchronous transfer. The target may renegotiate for an REQ/ACK offset less than FFh and greater than 00h.
c) MESSAGE REJECT.	Asynchronous transfer.

The initiator responds in any of the following ways if the target selects an REQ/ACK offset less than FFh.

Initiator Response	Implied Agreement
a) REQ/ACK offset less than or equal to the requested value. Minimum transfer period equal to or greater than requested period.	REQ/ACK offset equals initiator value. Minimum transfer period equal to the initiator value.
b) REQ/ACK offset equal to zero.	Asynchronous transfer.
c) MESSAGE REJECT.	Asynchronous transfer.

The implied agreement shall remain in effect until a BUS DEVICE RESET message is received, until a "hard" RESET condition occurs, or until one of the two SCSI devices elects to modify the agreement. Renegotiation at every selection is not recommended, since a significant performance impact is likely. The default mode of data transfer is asynchronous mode. The default mode is entered at power on, after a BUS DEVICE RESET message, or after a "hard" RESET condition. The SYNCHRONOUS DATA TRANSFER REQUEST message exchange can only take place following a SELECTION phase that includes the SCSI IDs for both the initiator and the target. Violation of this rule may make data transfer impossible owing to disagreements among SCSI devices about the data transfer mode.

5.5.6 EXTENDED IDENTIFY Message (Optional)

The EXTENDED IDENTIFY message (table 10) is optional and may be sent by a target or an initiator. It may be used in conjunction with the normal IDENTIFY message in order to expand the logical unit number addressing in a target. The sub-logical unit number specifies the encoded eight-bit sub- logical unit number used to identify one of 256 sub-logical units within the logical unit. This allows up to 2048 units to be addressed on a single target.

Table 10 – EXTENDED IDENTIFY

Byte	Value	Description
0	01h	Extended message
1	02h	Extended message length
2	02h	EXTENDED IDENTIFY code
3	x	Sub-logical unit number

6 SCSI commands

This section defines the SCSI command structure and gives several examples.

The command definitions assume a data structure providing the appearance at the interface of a contiguous set of logical blocks of a fixed or explicitly defined data length. The SCSI device maps the physical characteristics of the attached peripheral devices to one of several logical structures defined by the device type code.

A single command may transfer one or more logical blocks of data. Multiple commands may be linked if they are sent to the same logical unit. A target may disconnect from the SCSI bus to allow activity by other SCSI devices while a logical unit is being prepared to transfer data.

Upon command completion (successful or unsuccessful), the target returns a status byte to the initiator. Since most error and exception conditions cannot be adequately described with a single status byte, one status code, CHECK CONDITION, indicates that additional information is available. The initiator may issue a REQUEST SENSE command to retrieve this additional information.

By keeping to a minimum the functions essential to communicate via this protocol, a wide range of peripheral devices of varying capability can operate in the same environment.

Because subsets of the full architecture may be implemented, optional functions are noted.

6.1 Command Implementation Requirements.

The first byte of any SCSI command shall contain an operation code as defined in this International Standard. Three bits (bits 7 to 5) of the second byte of each SCSI command specify the logical unit if it is not specified using the IDENTIFY message (see 5.5.2). The last byte of all SCSI commands shall contain a control byte as defined in 6.2.6.

6.1.1 Reserved.

Reserved bits, fields, bytes, and code values are set aside for future standardization. Their use and interpretation will be specified by future extensions to this International Standard. A reserved bit, field, or byte shall be set to zero, or in accordance with a future extension to this standard. A target that receives a reserved bit, field, or byte that is not zero or receives a reserved code value shall terminate the command with a CHECK CONDITION status and, if extended sense is implemented, the sense key shall be set to ILLEGAL REQUEST. It shall also be acceptable for a target to interpret the bit, field, byte, or code value in accordance with a future extension to this International Standard.

6.1.2 Operation Code Types

The Operation code types are defined in table 11.

Table 11 – Operation Codes

Operation Code Type	Description
M	Mandatory - Commands so designated shall be implemented in order to meet the minimum requirement of this International Standard.
E	Extended - Commands so designated shall be implemented in addition to mandatory commands to meet the extended requirement of this International Standard.
O	Optional - Commands so designated, if implemented, shall be implemented as defined in this International Standard.
V	Vendor unique - Operation codes so designated are available for vendor defined commands. See the vendor specifications where compatibility is desired.
R	Reserved - Operation codes so designated shall not be used. They are reserved for future extensions to this International Standard.

6.1.3 Unit Attention Condition

A unit attention condition for a logical unit shall begin for each initiator whenever the removable medium may have been changed or the target has been reset (by a BUS DEVICE RESET message or a "hard" RESET condition). The unit attention condition shall persist for each initiator until that initiator issues a command to the logical unit other than REQUEST SENSE or INQUIRY for which the target shall return CHECK CONDITION status. If the next command from that initiator to the logical unit (following the CHECK CONDITION status) is REQUEST SENSE, and if the target supports extended sense, then the UNIT ATTENTION sense key shall be returned. (If any command other than REQUEST SENSE is received, the unit attention condition is lost.)

If an INQUIRY command is received from an initiator with a pending unit attention condition (before the target reports CHECK CONDITION status), the target shall perform the INQUIRY command and shall not clear the unit attention condition.

If a REQUEST SENSE command is received from an initiator with a pending unit attention condition (before the target reports CHECK CONDITION status), then the target may either:

- a) report any pending sense data and preserve the unit attention condition;
- b) discard any pending sense data, report UNIT ATTENTION sense key, and clear the unit attention condition for that initiator.

If an initiator issues a command other than INQUIRY or REQUEST SENSE while a unit attention condition exists for that initiator, the target shall not perform the command and shall report CHECK CONDITION status.

6.2 Command Descriptor Block (CDB)

A request to a peripheral device is performed by sending a command descriptor block to the target. For several commands, the request is accompanied by a list of parameters sent during the DATA OUT phase. See the specific commands for detailed information.

The command descriptor block always has an operation code as the first byte of the command. This is followed by a logical unit number, command parameters (if any), and a control byte.

For all commands, if there is an invalid parameter in the command descriptor block, then the target shall terminate the command without altering the medium.

6.2.1 Operation Code.

The operation code (table 12) of the command descriptor block has a group code field and a command code field. The three-bit group code field provides for eight groups of command codes. The five-bit command code field provides for thirty-two command codes in each group. Thus, a total of 256 possible operation codes exist. Operation codes are defined in clauses 7 to 13.

Table 12 – Operation Code

Byte \ Bit	7	6	5	4	3	2	1	0
0	Group Code			Command Code				

The group code specifies one of the following groups:

- Group 0 - six-byte commands (see table 13)
- Group 1 - ten-byte commands (see table 14)
- Group 2 - reserved
- Group 3 - reserved
- Group 4 - reserved
- Group 5 - twelve-byte commands (see table 15)

Group 6 - vendor unique
Group 7 - vendor unique

Table 13 – Typical CDB for Six-Byte Commands

Byte \ Bit	7	6	5	4	3	2	1	0
0	Operation Code							
1	Logical Unit Number			Logical Block Address (if required) (MSB)				
2	Logical Block Address (if required)							
3	Logical Block Address (if required) (LSB)							
4	Transfer Length (if required)							
5	Control Byte							

Table 14 – Typical CDB for Ten-Byte Commands

Byte \ Bit	7	6	5	4	3	2	1	0
0	Operation Code							
1	Logical Unit Number			Reserved				RelAdr
2	Logical Block Address (if required) (MSB)							
3	Logical Block Address (if required)							
4	Logical Block Address (if required)							
5	Logical Block Address (if required) (LSB)							
6	Reserved							
7	Transfer Length (if required) (MSB)							
8	Transfer Length (if required) (LSB)							
9	Control Byte							

Table 15 – Typical CDB for Twelve-Byte Commands

Byte \ Bit	7	6	5	4	3	2	1	0
0	Operation Code							
1	Logical Unit Number			Reserved				RelAdr
2	Logical Block Address (if required) (MSB)							
3	Logical Block Address (if required)							
4	Logical Block Address (if required)							
5	Logical Block Address (if required) (LSB)							
6	Reserved							
7	Reserved							
8	Reserved							
9	Transfer Length (if required) (MSB)							
10	Transfer Length (if required) (LSB)							
11	Control Byte							

6.2.2 Logical Unit Number

The logical unit number addresses one of up to eight physical or virtual devices attached to a target. This method of addressing is provided for systems that do not implement the IDENTIFY message. A target that accepts an IDENTIFY message shall use the logical unit number specified within the message. In this case, the target shall ignore the logical unit number specified within the command descriptor block. (Implementors note: It is a good practice for initiators that implement the IDENTIFY message to specify the same logical unit number in the command descriptor block.)

6.2.3 Logical Block Address.

The logical block address on logical units shall begin with block zero and be contiguous up to the last logical block on that logical unit.

Group 0 command descriptor blocks contain 21-bit logical block addresses. Groups 1 and 5 command descriptor blocks contain 32-bit logical block addresses.

The logical block concept implies that the initiator and target shall have previously established the number of data bytes per logical block. This may be established through the use of the READ CAPACITY command or the MODE SENSE command or by prior arrangement.

6.2.4 Relative Address Bit

The relative address (RelAdr) bit of the group 1 and group 5 commands is set to one to indicate that the logical block address portion of the command descriptor block is a two's complement displacement. This negative or positive displacement is to be added to the logical block address last accessed on the logical unit to form the logical block address for this command. This feature is only available when linking commands. The feature requires that a previous command in the linked group have accessed a block of data on the logical unit. (For an example of the operation of this function, see 6.3.3.)

6.2.5 Transfer Length

The transfer length specifies the amount of data to be transferred, usually the number of blocks. For several commands the transfer length indicates the requested number of bytes to be sent as defined in the command description. For these commands the transfer length field may be identified by a different name. See the following descriptions and the individual command descriptions for further information.

Commands that use one byte for transfer length allow up to 256 blocks of data to be transferred by one command. A transfer length value of 1 to 255 indicates the number of blocks that shall transferred. A value of zero indicates 256 blocks.

Commands that use two bytes for transfer length allow up to 65,535 blocks of data to be transferred by one command. In this case, a transfer length of zero indicates that no data transfer shall take place. A value of 1 to 65,535 indicates the number of blocks that shall be transferred.

For several commands more than two bytes are allocated for transfer length. Refer to the specific command description for further information.

The transfer length of the commands that are used to send a list of parameters to a target is called the parameter list length. The parameter list length specifies the number of bytes sent during the DATA OUT phase.

The transfer length of the commands that are used to return sense data (e.g., REQUEST SENSE, INQUIRY, MODE SENSE, etc) to an initiator is called the allocation length. The allocation length specifies the number of bytes that the initiator has allocated for returned data. The target shall terminate the DATA IN phase when allocation length bytes have been transferred or when all available sense data have been transferred to the initiator, whichever is less.

6.2.6 Control Byte

The control byte is the last byte of every command descriptor block. A typical control byte is described in table 16.

Table 16 – Control Byte

Byte \ Bit	7	6	5	4	3	2	1	0
Last	Vendor Unique		Reserved				Flag	Link
Bit	Description							
7 - 6	Vendor unique							
5 - 2	Reserved							
1	Flag bit - If the link bit is zero, then the flag bit shall be set to zero. If the link bit is one, and if the command terminates successfully, the target shall send LINKED COMMAND COMPLETE message if the flag bit is zero and shall send LINKED COMMAND COMPLETE (WITH FLAG) message if the flag bit is one. Typically, this bit is used to cause an interrupt in the initiator between linked commands.							
0	Link bit - This bit is set to one to indicate that the initiator desires an automatic link to the next command upon successful completion of the current command. Implementation of linked commands is optional. If the link bit is one, targets that implement linked commands, upon successful termination of the command, shall return INTERMEDIATE status and shall then send one of the two messages defined by the flag bit (above).							
Targets that do not implement linked commands shall return a CHECK CONDITION status and, if extended sense is implemented, shall set the sense key to ILLEGAL REQUEST if either of the link and flag bits are set to one.								

6.3 Command Examples

6.3.1 Single Command Example

A typical operation on the SCSI bus is likely to include a single READ command to a peripheral device. This operation is described in detail starting with a request from the initiator. This example assumes that no linked commands and no malfunctions or errors occur.

The initiator has active pointers and a set of stored pointers representing active disconnected SCSI devices (an initiator without disconnect capability does not require stored pointers). The initiator sets up the active pointers for the operation requested, arbitrates for the SCSI bus, and selects the target. Once this process is completed, the target assumes control of the operation.

The target obtains the command from the initiator (in this case, a READ command). The target interprets the command and executes it. In this case, the target gets the data from the peripheral device and sends it to the initiator. At the completion of the READ command, the target sends a status byte to the initiator. To end the operation, the target sends a COMMAND COMPLETE message to the initiator.

6.3.2 Disconnect Example

In the above single command example, the length of time necessary to obtain the data may require a time-consuming physical seek. In order to improve system throughput, the target may disconnect from the initiator, freeing the SCSI bus to allow other requests to be sent to other logical units. To do this, the initiator needs to be reselectable and capable of restoring the pointers upon reconnection. The target needs to be capable of arbitrating for the SCSI bus and reselecting the initiator.

After the target has received the READ command (and has determined that there will be a delay), it disconnects by sending a DISCONNECT message and releasing BSY.

When the data are ready to be transferred, the target reconnects to the initiator. As a result of this reconnection, the initiator restores the pointers to their most recent saved values (which, in this case, are the initial values) and the target continues (as in the single command example) to finish the operation. The initiator recognizes that the operation is complete when COMMAND COMPLETE message is received.

If target wishes to disconnect after transferring part of the data (e.g., while crossing a cylinder boundary), it may do so by sending a SAVE DATA POINTER message and a DISCONNECT message to the initiator and then disconnecting. When reconnection is completed, the current data pointer value is restored to its value immediately prior to the SAVE DATA POINTER message.

On those occasions when an error or exception condition occurs and the target elects to repeat the information transfer, the target may repeat the transfer by either issuing a RESTORE POINTERS message or by disconnecting without issuing a SAVE DATA POINTER message. When reconnection is completed, the most recent saved pointer values are restored.

6.3.3 Linked Command Example

The link function defines a relationship between commands that when combined with the relative address bit, allows previous operations to modify subsequent operations. Link makes high-performance functions possible by providing a relative addressing capability and allowing multiple command execution without invoking the functional component of the initiator.

If the desired data address (in the previously described READ command example) is unknown, but a search key defined as some particular bytes of a field is known, then by linking the READ command to a SEARCH DATA EQUAL command, the data can be quickly and effectively transferred to the initiator.

A LINKED COMMAND COMPLETE message is sent from the target to the initiator to indicate linked command completion. The initiator then updates the stored pointers so that subsequent requests from the target will reference the next command of the chain. Command processing of linked and single commands is similar except that relative addressing is permitted in linked commands. Linked commands shall be addressed to a single logical unit.

For example, the successful completion of a **SEARCH DATA EQUAL** command causes the target to fetch the linked **READ** command from the initiator. If the relative address bit in the **READ** command has been set to one, and the address field of the **READ** command is set to zero, the target transfers the successfully searched block to the initiator.

7 Command descriptions for all device types

7.1 Group 0 Commands for All Device Types

These commands shall be as listed in table 17.

Table 17 – Group 0 Commands for all Device Types

Operation Code	Type	Command Name	Subclause	Page
00h	O	TEST UNIT READY	7.1.1	44
01h	*			
02h	V	REQUEST SENSE	7.1.2	44
03h	M			
04h	*			
05h	*			
06h	V			
07h	*			
08h	*			
09h	V			
0Ah	*			
0Bh	*			
0Ch	V			
0Dh	V			
0Eh	V			
0Fh	*			
10h	*	INQUIRY	7.1.3	48
11h	*			
12h	E			
13h	*			
14h	*			
15h	*			
16h	*			
17h	*	COPY	7.1.4	50
18h	O			
19h	*			
1Ah	*			
1Bh	*	RECEIVE DIAGNOSTIC RESULTS SEND DIAGNOSTIC	7.1.5	57
1Ch	O		7.1.6	58
1Dh	O			
1Eh	*			
1Fh	R			

Key:

- M = Command implementation is mandatory.
- E = Command implementation is required for SCSI devices that support device-independent self-configuring software.
- O = Command implementation is optional.
- R = Operation code is reserved for future standardization.
- V = Operation code is available for vendor unique commands.
- * = These operation codes may have different meanings for specific types of peripheral devices. (See the appropriate section for further information.)

7.1.1 TEST UNIT READY Command
Peripheral Device Type: All
Operation Code Type: Optional
Operation Code: 00h

Table 18 – TEST UNIT READY Command

Byte \ Bit	7	6	5	4	3	2	1	0
0	Operation Code							
1	Logical Unit Number			Reserved				
2	Reserved							
3	Reserved							
4	Reserved							
5	Vendor Unique		Reserved				Flag	Link

The TEST UNIT READY command (table 18) provides a means to check if the logical unit is ready. This is not a request for a self test. If the logical unit would accept an appropriate medium-access command without returning CHECK CONDITION status, this command shall return a GOOD status.

7.1.2 REQUEST SENSE Command
Peripheral Device Type: All
Operation Code Type: Mandatory
Operation Code: 03h

Table 19 – REQUEST SENSE Command

Byte \ Bit	7	6	5	4	3	2	1	0
0	Operation Code							
1	Logical Unit Number			Reserved				
2	Reserved							
3	Reserved							
4	Allocation Length							
5	Vendor Unique		Reserved				Flag	Link

The REQUEST SENSE command (table 19) requests that the target transfer sense data to the initiator.