

American National Standard

*for Information Systems –
Small Computer System
Interface-2*



American National Standards Institute

*11 West 42nd Street
New York, New York
10036*

**American National Standard
for Information Systems –
Small Computer System Interface-2**

**Secretariat
Computer and Business Equipment Manufacturers Association**

**Approved January 31, 1994
American National Standards Institute, Inc.**

Abstract

The SCSI protocol is designed to provide an efficient peer-to-peer I/O bus with up to 16 devices, including one or more hosts. Data may be transferred asynchronously at rates that only depend on device implementation and cable length. Synchronous data transfers are supported at rates up to 10 mega-transfers per second. With the 32-bit wide data transfer option, data rates of up to 40 megabytes per second are possible.

SCSI-2 includes command sets for magnetic and optical disks, tapes, printers, processors, CD-ROMs, scanners, medium changers, and communications devices.

American National Standard

Approval of an American National Standard requires verification by ANSI that the requirements for due process, consensus, and other criteria for approval have been met by the standards developer.

Consensus is established when, in the judgment of the ANSI Board of Standards Review, substantial agreement has been reached by directly and materially affected interests. Substantial agreement means much more than a simple majority, but not necessarily unanimity. Consensus requires that all views and objections be considered, and that a concerted effort be made toward their resolution.

The use of American National Standards is completely voluntary; their existence does not in any respect preclude anyone, whether he has approved the standards or not, from manufacturing, marketing, purchasing, or using products, processes, or procedures not conforming to the standards.

The American National Standards Institute does not develop standards and will in no circumstances give an interpretation of any American National Standard. Moreover, no person shall have the right or authority to issue an interpretation of an American National Standard in the name of the American National Standards Institute. Requests for interpretations should be addressed to the secretariat or sponsor whose name appears on the title page of this standard.

CAUTION NOTICE: This American National Standard may be revised or withdrawn at any time. The procedures of the American National Standards Institute require that action be taken periodically to reaffirm, revise, or withdraw this standard. Purchasers of American National Standards may receive current information on all standards by calling or writing the American National Standards Institute.

CAUTION: The developers of this standard have requested that holders of patents that may be required for the implementation of the standard disclose such patents to the publisher. However, neither the developers nor the publisher have undertaken a patent search in order to identify which, if any, patents may apply to this standard.

As of the date of publication of this standard and following calls for the identification of patents that may be required for the implementation of the standard, no such claims have been made. No further patent search is conducted by the developer or publisher in respect to any standard it processes. No representation is made or implied that licenses are not required to avoid infringement in the use of this standard.

Published by

**American National Standards Institute
11 West 42nd Street, New York, New York 10036**

Copyright ©1994 by American National Standards Institute
All rights reserved.

No part of this publication may be reproduced in any form, in an electronic retrieval system or otherwise, without prior written permission of the publisher.

Printed in the United States of America

APS1M894/125

Contents

	Page
Foreword	xix
Introduction	xxii
1 Scope	1
2 Normative references	2
3 Definitions, symbols and abbreviations	3
3.1 Definitions	3
3.2 Symbols and abbreviations	5
4 General	6
4.1 Overview	6
4.2 Conventions	7
5 Physical characteristics	8
5.1 Physical description	8
5.2 Cable requirements	8
5.2.1 Single-ended cable	8
5.2.2 Differential cable	9
5.2.3 Cable requirements for fast synchronous data transfer	9
5.3 Connector requirements	9
5.3.1 Non-shielded connector requirements	9
5.3.1.1 Non-shielded connector alternative 1- A cable	9
5.3.1.2 Non-shielded connector alternative 2- A cable	10
5.3.1.3 Non-shielded connector - B cable	10
5.3.2 Shielded connector requirements	15
5.3.2.1 Shielded connector alternative 1- A cable	15
5.3.2.2 Shielded connector alternative 2- A cable	15
5.3.2.3 Shielded connector - B cable	15
5.3.3 Connector contact assignments	20
5.4 Electrical description	25
5.4.1 Single-ended alternative	25
5.4.1.1 Output characteristics	25
5.4.1.2 Input characteristics	25
5.4.2 Differential alternative	26
5.4.2.1 Output characteristics	26
5.4.2.2 Input characteristics	26
5.4.3 Terminator power	26
5.4.4 RESERVED lines	29
5.5 SCSI bus	30
5.6 SCSI bus signals	32
5.6.1 Signal values	33
5.6.2 OR-tied signals	33
5.6.3 Signal sources	33
5.7 SCSI bus timing	34
5.7.1 Arbitration delay	35
5.7.2 Assertion period	35
5.7.3 Bus clear delay	35
5.7.4 Bus free delay	35
5.7.5 Bus set delay	36

	Page
5.7.6	Bus settle delay 36
5.7.7	Cable skew delay 36
5.7.8	Data release delay 36
5.7.9	Deskew delay 36
5.7.10	Disconnection delay 36
5.7.11	Hold time 36
5.7.12	Negation period 36
5.7.13	Power-on to selection time 36
5.7.14	Reset to selection time 36
5.7.15	Reset hold time 37
5.7.16	Selection abort time 37
5.7.17	Selection time-out delay 37
5.7.18	Transfer period 37
5.8	Fast synchronous transfer option 37
5.8.1	Fast assertion period 37
5.8.2	Fast cable skew delay 37
5.8.3	Fast deskew delay 37
5.8.4	Fast hold time 37
5.8.5	Fast negation period 38
6	Logical characteristics 39
6.1	SCSI bus phases 39
6.1.1	BUS FREE phase 39
6.1.2	ARBITRATION phase 40
6.1.3	SELECTION phase 40
6.1.3.1	SELECTION time-out procedure 41
6.1.4	RESELECTION phase 41
6.1.4.1	RESELECTION 41
6.1.4.2	RESELECTION time-out procedure 42
6.1.5	Information transfer phases 42
6.1.5.1	Asynchronous information transfer 43
6.1.5.2	Synchronous data transfer 43
6.1.5.3	Wide data transfer 44
6.1.6	COMMAND phase 46
6.1.7	Data phase 46
6.1.7.1	DATA IN phase 46
6.1.7.2	DATA OUT phase 46
6.1.8	STATUS phase 46
6.1.9	Message phase 46
6.1.9.1	MESSAGE IN phase 46
6.1.9.2	MESSAGE OUT phase 46
6.1.10	Signal restrictions between phases 47
6.2	SCSI bus conditions 47
6.2.1	Attention condition 47
6.2.2	Reset condition 48
6.2.2.1	Hard reset alternative 49
6.2.2.2	Soft reset alternative 49
6.3	SCSI bus phase sequences 50
6.4	SCSI pointers 51
6.5	Message system description 52
6.6	Messages 55
6.6.1	ABORT 55
6.6.2	ABORT TAG 55
6.6.3	BUS DEVICE RESET 56

	Page
6.6.4	CLEAR QUEUE 56
6.6.5	COMMAND COMPLETE 56
6.6.6	DISCONNECT 56
6.6.7	IDENTIFY 57
6.6.8	IGNORE WIDE RESIDUE 58
6.6.9	INITIATE RECOVERY 58
6.6.10	INITIATOR DETECTED ERROR 59
6.6.11	LINKED COMMAND COMPLETE 59
6.6.12	LINKED COMMAND COMPLETE (WITH FLAG) 59
6.6.13	MESSAGE PARITY ERROR 59
6.6.14	MESSAGE REJECT 59
6.6.15	MODIFY DATA POINTER Message 60
6.6.16	NO OPERATION 60
6.6.17	Queue tag messages 60
6.6.17.1	HEAD OF QUEUE TAG 61
6.6.17.2	ORDERED QUEUE TAG 61
6.6.17.3	SIMPLE QUEUE TAG 61
6.6.18	RELEASE RECOVERY 61
6.6.19	RESTORE POINTERS 61
6.6.20	SAVE DATA POINTER 62
6.6.21	SYNCHRONOUS DATA TRANSFER REQUEST 62
6.6.22	TERMINATE I/O PROCESS 64
6.6.23	WIDE DATA TRANSFER REQUEST 64
7	SCSI commands and status 67
7.1	Command implementation requirements 67
7.1.1	Reserved 67
7.1.2	Operation code types 67
7.2	Command descriptor block 68
7.2.1	Operation code 69
7.2.2	Logical unit number 70
7.2.3	Logical block address 70
7.2.4	Transfer length 70
7.2.5	Parameter list length 70
7.2.6	Allocation length 70
7.2.7	Control field 71
7.3	Status 71
7.4	Command examples 73
7.4.1	Single command example 73
7.4.2	Disconnect example 73
7.4.3	Linked command example 74
7.5	Command processing considerations and exception conditions 74
7.5.1	Programmable operating definition 74
7.5.2	Incorrect initiator connection 75
7.5.3	Selection of an invalid logical unit 75
7.5.4	Parameter rounding 76
7.5.5	Asynchronous event notification 76
7.5.6	Unexpected reselection 77
7.6	Contingent allegiance condition 78
7.7	Extended contingent allegiance condition 78
7.8	Queued I/O processes 79
7.8.1	Untagged queuing 79
7.8.2	Tagged queuing 79
7.8.3	Example of queued I/O process 80

	Page
7.8.3.1	Typical sequences for tagged queuing 81
7.8.3.2	Example of tagged queuing 81
7.9	Unit attention condition 83
8	All device types 84
8.1	Model for all device types 84
8.1.1	SCSI addresses 84
8.1.1.1	SCSI device address 84
8.1.1.2	Logical units 84
8.1.1.3	Target routines 84
8.1.2	Commands implemented by all SCSI devices 84
8.1.2.1	Using the INQUIRY command 85
8.1.2.2	Using the REQUEST SENSE command 85
8.1.2.3	Using the SEND DIAGNOSTIC command 85
8.1.2.4	Using the TEST UNIT READY command 85
8.2	Commands for all device types 85
8.2.1	CHANGE DEFINITION command 86
8.2.2	COMPARE command 88
8.2.3	COPY command 89
8.2.3.1	Errors detected by the managing SCSI device 90
8.2.3.2	Errors detected by a target 91
8.2.3.3	COPY function code 00h and 01h 91
8.2.3.4	COPY function code 02h 92
8.2.3.5	COPY function code 03h 93
8.2.3.6	COPY function code 04h 94
8.2.3.7	Copies with unequal block lengths 94
8.2.4	COPY AND VERIFY command 95
8.2.5	INQUIRY command 96
8.2.5.1	Standard INQUIRY data 97
8.2.5.2	Vital product data 100
8.2.6	LOG SELECT command 101
8.2.7	LOG SENSE command 103
8.2.8	MODE SELECT(6) command 104
8.2.9	MODE SELECT(10) command 106
8.2.10	MODE SENSE(6) command 106
8.2.10.1	Current values 107
8.2.10.2	Changeable values 108
8.2.10.3	Default values 108
8.2.10.4	Saved values 108
8.2.10.5	Initial responses 108
8.2.11	MODE SENSE(10) command 109
8.2.12	READ BUFFER command 109
8.2.12.1	Combined header and data mode (000b) 110
8.2.12.2	Vendor-specific mode (001b) 110
8.2.12.3	Data mode (010b) 110
8.2.12.4	Descriptor mode (011b) 111
8.2.13	RECEIVE DIAGNOSTIC RESULTS command 112
8.2.14	REQUEST SENSE command 112
8.2.14.1	Sense-key specific 116
8.2.14.2	Deferred errors 117
8.2.14.3	Sense key and sense code definitions 119
8.2.15	SEND DIAGNOSTIC command 125
8.2.16	TEST UNIT READY command 126
8.2.17	WRITE BUFFER command 127

	Page
8.2.17.1	Combined header and data mode (000b) 128
8.2.17.2	Vendor-specific mode (001b) 128
8.2.17.3	Data mode (010b) 128
8.2.17.4	Download microcode mode (100b) 129
8.2.17.5	Download microcode and save mode (101b) 129
8.3	Parameters for all device types 129
8.3.1	Diagnostic parameters 129
8.3.1.1	Supported diagnostic pages 130
8.3.2	Log parameters 131
8.3.2.1	Buffer over-run/under-run page 134
8.3.2.2	Error counter pages 135
8.3.2.3	Last <i>n</i> error events page 136
8.3.2.4	Non-medium error page 136
8.3.2.5	Supported log pages 136
8.3.3	Mode parameters 137
8.3.3.1	Control mode page 140
8.3.3.2	Disconnect-reconnect page 142
8.3.3.3	Peripheral device page 144
8.3.4	Vital product data parameters 144
8.3.4.1	ASCII implemented operating definition page 145
8.3.4.2	ASCII information page 145
8.3.4.3	Implemented operating definition page 146
8.3.4.4	Supported vital product data pages 148
8.3.4.5	Unit serial number page 149
9	Direct-access devices 150
9.1	Direct-access device model 150
9.1.1	Removable medium 150
9.1.2	Logical blocks 150
9.1.3	Ready state 151
9.1.4	Initialization 151
9.1.5	Medium defects 151
9.1.6	Data cache 152
9.1.7	Reservations 153
9.1.8	Seek and rezero 154
9.1.9	Notched drives 154
9.1.10	Rotational position locking 154
9.1.11	Relative addressing 154
9.1.12	Error reporting 154
9.1.13	Examples 155
9.1.13.1	Rotating media 155
9.1.13.2	Sequential media 156
9.1.13.3	Memory media 156
9.2	Commands for direct-access devices. 157
9.2.1	FORMAT UNIT command 158
9.2.1.1	Defect list formats 162
9.2.1.2	Initialization pattern option 163
9.2.2	LOCK UNLOCK CACHE command 165
9.2.3	PRE-FETCH command 166
9.2.4	PREVENT ALLOW MEDIUM REMOVAL command 167
9.2.5	READ(6) command 168
9.2.6	READ(10) command 168
9.2.7	READ CAPACITY command 169
9.2.8	READ DEFECT DATA command 171

	Page
9.2.9	READ LONG command 173
9.2.10	REASSIGN BLOCKS command 174
9.2.11	RELEASE command 175
9.2.11.1	Logical unit release 176
9.2.11.2	Extent release 176
9.2.11.3	Third-party release 176
9.2.12	RESERVE command 177
9.2.12.1	Logical unit reservation 177
9.2.12.2	Extent reservation 177
9.2.12.3	Third-party reservation 179
9.2.12.4	Superseding reservations 180
9.2.13	REZERO UNIT command 180
9.2.14	SEARCH DATA commands 181
9.2.14.1	SEARCH DATA EQUAL command 183
9.2.14.2	SEARCH DATA HIGH command 183
9.2.14.3	SEARCH DATA LOW command 183
9.2.15	SEEK(6) and SEEK(10) commands 183
9.2.16	SET LIMITS command 184
9.2.17	START STOP UNIT command 185
9.2.18	SYNCHRONIZE CACHE command 186
9.2.19	VERIFY command 187
9.2.20	WRITE(6) command 188
9.2.21	WRITE(10) command 188
9.2.22	WRITE AND VERIFY command 189
9.2.23	WRITE LONG command 190
9.2.24	WRITE SAME command 191
9.3	Parameters for direct-access devices 192
9.3.1	Diagnostic parameters 192
9.3.1.1	Translate address page - SEND DIAGNOSTIC 192
9.3.1.2	Translate address page - RECEIVE DIAGNOSTIC 193
9.3.2	Log parameters 194
9.3.3	Mode parameters 194
9.3.3.1	Caching page 196
9.3.3.2	Flexible disk page 199
9.3.3.3	Format device page 202
9.3.3.4	Medium types supported page 205
9.3.3.5	Notch and partition page 206
9.3.3.6	Read-write error recovery page 207
9.3.3.7	Rigid disk drive geometry page 214
9.3.3.8	Verify error recovery page 215
9.4	Definitions specific to direct-access devices 217
10	Sequential-access devices 218
10.1	Sequential-access device model 218
10.1.1	Physical elements 218
10.1.2	Data storage characteristics 219
10.1.3	Partitions within a volume 221
10.1.4	Logical elements within a partition 222
10.1.5	Data buffering 223
10.1.6	Recorded object descriptors (block identifiers) 224
10.1.7	Direction and position definitions 224
10.1.8	Error reporting 225
10.2	Command descriptions for sequential-access devices 226
10.2.1	ERASE command 227

9 Direct-access devices

9.1 Direct-access device model

Direct-access devices store blocks of data for later retrieval. Each block of data is stored at a unique logical block address. An initiator issues WRITE commands to store the blocks of data (write operations) and READ commands to retrieve the blocks of data (read operations). Other commands issued by the initiator may also cause write and read operations to occur. A write operation causes a block of data to be written on the medium. A read operation causes a block of data to be read from the medium. A verify operation confirms that a block of data can be read without error from the medium.

Blocks of data are stored by a process that causes localized changes or transitions within the medium. The changes made to the medium to store the blocks of data may be volatile (i.e. not retained through a power cycle) or non-volatile (retained through a power cycle). The medium may be divided in parts that are used for data blocks, parts that are reserved for defect management, and parts that are reserved for use by the controller for the management of the device.

9.1.1 Removable medium

The medium may be removable (e.g. used in a floppy disk drive) or non-removable (e.g. used in a hard disk drive). Removable medium is contained within a cartridge (or jacket) to prevent damage to the recording surfaces. The combination of medium and cartridge is often called a volume.

A volume has an attribute of being mounted or de-mounted on a suitable transport mechanism. A volume is mounted when the direct-access device is capable of performing write or read operations to the medium. A mounted volume may not be accessible by an initiator if it is reserved by another initiator. A volume is de-mounted at any other time (e.g. during loading, unloading, or storage).

An initiator may check whether a volume is mounted by issuing a TEST UNIT READY command. A volume that is loaded may need a START STOP UNIT command issued to become accessible for write or read operations.

The PREVENT ALLOW MEDIUM REMOVAL command allows an initiator to restrict the demounting of the volume. This is useful in maintaining system integrity. If the direct-access device implements cache memory, it must ensure that all logical blocks of the medium contain the most recent data prior to permitting demounting of the volume. If the initiator issues a START STOP UNIT command to eject the cartridge, and the direct-access device is prevented from demounting by the PREVENT ALLOW MEDIUM REMOVAL command, the START STOP unit command is rejected by the direct-access device.

9.1.2 Logical blocks

Blocks of data are stored on the medium along with additional information that the controller uses to manage the storage and retrieval. The format of the additional information is unique and is hidden from the initiator during normal read or write operations. This additional information is often used to identify the physical location of the blocks of data and the address of the logical block, and to provide protection against the loss of the user data.

The address of the first logical block is zero. The address of the last logical block is $[n-1]$, where $[n]$ is the number of logical blocks available on the medium. A READ CAPACITY command may be issued to determine the value of $[n-1]$. If a command is issued that requests access to a logical block not within the capacity of the medium, the command is terminated with CHECK CONDITION.

The number of bytes of data contained in a logical block is known as the block length. Each logical block has a block length associated with it. The block length may be different for each logical block on the medium. However, in a typical device only one block length is used at a time. The block descriptor in the MODE SENSE data describes the block lengths that are used on the medium. A MODE SELECT command can be used to set up extents. An

extent is a specified number of logical blocks that have the specified block length. The FORMAT UNIT command is typically required to change the block length of devices that support variable block lengths and make the extents that were setup active.

The location of a logical block on the medium does not have a relationship to the location of any other logical block. However, in a typical device the logical blocks are located in an ascending order. The time to access the logical block at address $[x]$ and then the logical block at address $[x+1]$ need not be less than time to access $[x]$ and then $[x+100]$. The READ CAPACITY with a PMI bit of one is useful in determining where longer access times occur.

9.1.3 Ready state

A direct-access device is ready when medium access commands can be executed. A device using removable media is usually not ready until a volume is mounted. Such a device, with a volume not mounted, normally returns CHECK CONDITION status and sets the sense key to NOT READY.

Some direct-access devices may be switched from being ready to being not ready by using the START STOP UNIT command. An initiator may need to issue a START UNIT command to bring a device ready.

9.1.4 Initialization

Many direct-access devices require initialization prior to write or read operations. This initialization is usually performed by a FORMAT UNIT command. Parameters related to the geometry and performance characteristics can be set with the MODE SELECT command prior to the format operation. Some devices are initialized by means not specified in this standard. The time at which this occurs is specific to the implementation of the direct-access device.

Devices using a non-volatile medium typically save the parameters and only need to be initialized once. However, some mode parameters may need to be initialized after each power-on and reset. A catastrophic failure of the direct-access device may require the FORMAT UNIT command to be reissued.

Devices that use a volatile medium may need to be initialized at each power-on prior to the execution of read or write operations. Mode parameters may also need initialization.

9.1.5 Medium defects

Any medium has the potential for defects that can cause user data to be lost. Therefore, each logical block may contain information that allows the detection of changes to the user data caused by defects in the medium or other phenomena, and may also allow the data to be reconstructed following the detection of such a change. Some devices provide the initiator control through use of the mode parameters. Some devices allow the initiator to examine and modify the additional information by using the READ LONG and WRITE LONG commands. Some media having a very low probability of defects may not require these structures.

Defects may also be detected and managed during execution of the FORMAT UNIT command. The FORMAT UNIT command defines four sources of defect information. These defects may be reassigned or avoided during the initialization process so that they do not appear in a logical block.

Defects may also be avoided after initialization. The initiator issues a REASSIGN BLOCKS command to request that the specified logical block address be reassigned to a different part of the medium. This operation can be repeated if a new defect appears at a later time. The total number of defects that may be handled in this manner can be specified in the mode parameters.

Defect management on direct-access devices is usually vendor-specific. Devices not using a removable medium typically optimize the defect management for capacity or performance or both. Devices that use a removable medium typically do not support defect management (e.g. some floppy disk drives) or use defect management that is based on the ability to interchange the medium.

9.1.6 Data cache

Some direct-access devices implement cache memory. A cache memory is usually an area of temporary storage in the direct-access device with a fast access time that is used to enhance performance. It exists separately from the blocks of data stored and is normally not directly accessible by the initiator. Use of cache memory for write or read operations typically reduces the access time to a logical block and can increase the overall data throughput.

During read operations, the direct-access device uses the cache memory to store blocks of data that the initiator may request at some future time. The algorithm used to manage the cache memory is not part of this standard. However, parameters are provided to advise the direct-access device about future requests, or to restrict the use of cache memory for a particular request.

During write operations, the direct-access device uses the cache memory to store data that is written to the medium at a later time. This is called a write-back caching algorithm. Thus the command may complete prior to blocks of data being written to the medium. As a result of using a write-back caching algorithm there is a period of time when the data may be lost if a power or a hardware failure occurs. There is also the possibility of an error occurring during the write operation. If an error occurred during the write, it may be reported as a deferred error on a later command. However, the initiator can request write-through caching to prevent these circumstances from arising.

When the cache memory fills up with blocks of data that are being kept for possible future access, new blocks of data that are to be kept must replace those currently in cache memory. The disable page out (DPO) bit is used to control replacement of logical blocks in the cache. For write operations, setting this bit to one advises the direct-access device to not replace existing blocks in the cache memory with the write data. For read operations, setting this bit to one causes blocks of data that are being read to not replace existing ones in the cache memory.

Sometimes the initiator may wish to have the blocks of data read from the medium instead of from the cache memory. The force unit access (FUA) bit is used to indicate that the direct-access device shall access the physical medium. For a write operation, setting FUA to one causes the direct-access device to complete the data write to the physical medium before completing the command. For a read operation, setting FUA to one causes the logical blocks to be retrieved from the physical medium.

When the DPO and FUA bits are both set to one, write and read operations bypass the cache memory.

When a VERIFY command is executed, a forced unit access is implied, since the blocks of data stored on the medium are being verified. Furthermore, a SYNCHRONIZE CACHE operation (see below) is also implied to write unwritten blocks of data still in the cache memory. These blocks of data must be stored on the medium before the verify operation can begin. The DPO bit is provided since the VERIFY command may cause the replacement of blocks in the cache. The above also applies to the WRITE AND VERIFY command.

Commands may be implemented by the direct-access device that allow the initiator to control other behaviour of the cache memory:

- LOCK UNLOCK CACHE controls whether certain logical blocks will be held in the data cache for future use. Locking a logical block prevents its replacement by a future access. Unlocking a logical block exposes it to possible replacement by a future access (see 9.2.2).
- PRE-FETCH causes a set of logical blocks requested by the initiator to be read into the data cache for possible future access. The blocks fetched are subject to later replacement unless they are locked (see 9.2.3).
- SYNCHRONIZE CACHE forces any pending write data in the requested set of logical blocks to be stored in the physical medium. This command can be used to ensure that the data was written and any errors reported (see 9.2.18).
- The MODE SELECT command defines a page for the control of cache behavior and handles certain basic elements of cache replacement algorithms (see 9.3.3.1).

9.1.7 Reservations

The access enabled or access disabled condition determines when an initiator may store or retrieve user data on all or part of the medium. Access may be restricted for read operations, write operations, or both. This attribute may be controlled by an external mechanism or by the RESERVE and RELEASE commands (see 9.2.12 and 9.2.11).

The RESERVE and RELEASE commands define how different types of restricted access may be achieved, and to whom the access is restricted. This subclause describes the interaction of the initiator that requested the reservation, and the other initiators.

An initiator uses reservations to gain a level of exclusivity in access to all or part of the medium for itself or another initiator. It is expected that the reservation will be retained until released. The direct-access device must ensure that the initiator with the reservation is able to access the reserved media within the operating parameters established by that initiator.

The following paragraphs explain, on a command by command basis, the appropriate target response when a reservation exists. Unless otherwise noted, the appropriate response to an initiator that issues a command to a direct-access device that is reserved to another initiator is RESERVATION CONFLICT status.

The CHANGE DEFINITION command is dealt with as follows. If any initiator has an extent reservation on a direct-access device, no other initiator may affect the operating definition of that initiator by use of this command. If the direct-access device allows different operating definitions for each initiator, then there is no conflict; otherwise, a reservation conflict occurs.

The COMPARE, COPY, and COPY AND VERIFY commands are evaluated for reservation conflict as if they were normal write and read operations even when a direct-access device is requested to copy to or from itself. For example, if a COPY is issued to logical unit 0 that requests the direct-access device to copy from logical unit 0 to logical unit 1, access to logical unit 1 must also be evaluated for conflict.

The FORMAT UNIT, PREVENT ALLOW MEDIUM REMOVAL (with a prevent bit of one), REZERO UNIT, and START STOP UNIT commands return a RESERVATION CONFLICT status if any other initiator has an extent reservation on a direct-access device.

The INQUIRY and REQUEST SENSE commands are not affected by any kind of reservation.

The LOG SELECT, LOG SENSE, MODE SENSE, TEST UNIT READY, READ CAPACITY (PMI set to zero), READ BUFFER, WRITE BUFFER, and READ DEFECT DATA commands are not affected by extent reservations.

The SEEK, LOCK UNLOCK CACHE, PRE-FETCH, and SYNCHRONIZE CACHE commands are evaluated for reservation conflict as if they were normal write or read operations.

The MODE SELECT command is dealt with as follows. If an initiator has an extent reservation on a direct-access device, and another initiator attempts one of these commands, a reservation conflict occurs if the command affects the manner in which access of the extent by the first initiator is performed. If the command does not affect access to the extent, or parameters are saved for each initiator, then a conflict does not occur.

The SEND DIAGNOSTIC, RECEIVE DIAGNOSTIC RESULTS commands conflict with an extent reservation only if they affect access to the extent (as with MODE SELECT).

The REASSIGN BLOCKS command may not reassign a block that is in an extent reserved to another initiator.

The SET LIMITS command generates a reservation conflict if the logical blocks specified are within an extent reserved to another initiator.

ALL other commands are that request read or write operations are evaluated for reservation conflict as described in the RESERVE command.

When a system is integrated with more than one initiator, there must be agreement between the initiators as to how media is reserved and released during operations, otherwise, an initiator may be locked out of access to a target in the middle of an operation. For example, initiator 'A' has a write operation in progress to a direct-access device which has data stored in cache memory. Then, initiator 'B' issues a RESERVE command to the direct-access device. As a result, initiator 'A' is locked out of issuing a SYNCHRONIZE CACHE command to ensure the integrity of the data. To prevent this from happening, initiator 'A' should issue a RESERVE prior to the write command.

9.1.8 Seek and rezero

The SEEK command provides a way for the initiator to position the device in preparation for access to a particular logical block at some later time. Since this positioning action is implicit in other commands, the SEEK command may not be useful with some direct-access devices.

The REZERO UNIT command is provided to bring the direct-access device to a known condition. This standard does not specify the condition. The REZERO UNIT command is used in some devices to position the actuator at cylinder zero. Some devices return GOOD status without attempting any action.

9.1.9 Notched drives

A notched (also called partitioned or zoned) drive has areas of the medium in which the drive geometry changes. In the simplest case, the entire medium consists of a single notch. Multiple notches are often used to increase capacity of the drive. The notch page is used to indicate the notch for assignment of values to the parameters in the format device page. By sequencing the notch page through each notch, the format device parameters of each notch are set. This is usually done prior to initialization by the FORMAT UNIT command.

9.1.10 Rotational position locking

Rotational position locking is an optional feature implemented in some direct-access devices to allow the synchronization of spindles between a number of devices. The rotational position offset feature allows devices to synchronize spindles at offsets from index. This may be useful in improving performance in systems that implement arrays of devices.

9.1.11 Relative addressing

Relative addressing is a technique useful in accessing structured data in a uniform manner. Relative addressing is only allowed when commands are linked. An example of relative addressing and linking for SEARCH DATA commands appropriate to direct-access devices is given in 7.4.3.

The SET LIMITS command is provided to define the limits of a linked chain of relative addressing commands. This gives an additional protection against exceeding a particular set of blocks. The SET LIMITS command has no effect on any other initiator.

9.1.12 Error reporting

If any of the following conditions occur during the execution of a command, the target shall return CHECK CONDITION status. The appropriate sense key and additional sense code should be set. The following list illustrates some error conditions and the applicable sense keys. The list does not provide an exhaustive enumeration of all conditions that may cause the CHECK CONDITION status.

<u>Condition</u>	<u>Sense key</u>
Invalid logical block address	ILLEGAL REQUEST
Unsupported option requested	ILLEGAL REQUEST
Target reset or medium change since last command from this initiator	UNIT ATTENTION

Self diagnostic failed	HARDWARE ERROR
Unrecovered read error	MEDIUM ERROR or HARDWARE ERROR
Recovered read error	RECOVERED ERROR
Overrun or other error that might be resolved by repeating the command	ABORTED COMMAND
Attempt to write on write protected medium	DATA PROTECT

In the case of an invalid logical block address, the sense data information field shall be set to the logical block address of the first invalid address.

In the case of an attempt to read a blank or previously unwritten block, the information field shall be set to the logical block address of the first blank block encountered. The data read up to that block shall be transferred (optical memory and write-once devices only).

In the case of an attempt to write a previously written block when blank checking is enabled, the information field shall be set to the logical block address of the first non-blank block encountered (optical memory and write-once devices only).

9.1.13 Examples

The following examples show some typical variations of the direct-access device. Other variations are possible.

9.1.13.1 Rotating media

The typical application of a direct-access device is a disk drive. The medium is a disk coated with a material in which flux changes may be induced. The disk drive allows direct and random access to the medium. This is done with an actuator that positions the read-write head, and a rotating disk. Data is stored and retrieved through the interaction of the read-write head and the disk.

The disk is typically divided into cylinders. Each cylinder is typically divided into tracks. Each track is typically divided into sectors. A cylinder is a set of tracks that can be accessed without movement of the actuator. A track is a recording path over which the read-write head travels during one rotation of the disk. A sector is a part of a track that contains the stored data blocks.

A logical block is stored in one or more sectors, or a sector may store more than one logical block. A sector is typically made up of a header, data and a trailer. The header contains a preamble used to synchronize read circuits to the data, an address field to identify the sector, flags to use for defect management, and a checksum that validates the header. The data contains the block of data. The trailer contains checksum or error correction information. The checksum or the error correction information allows the correction of data for medium defects.

A disk drive is ready when the disks are rotating at the correct speed and the read-write circuitry is powered and ready to access the disks. Some disks, particularly removable disks, require the user to issue load or start commands to bring the disk drive to the ready state.

A disk drive will typically have to be formatted prior to the initial access. Exceptions to this are drives that are formatted at the factory and some optical drives with pre-formatted media (see 13.1). A disk drive format will typically create the headers for each sector and initialize the data field. The MODE SELECT command is often used at format time to establish the geometry (number of heads and tracks, sectors per track, etc.) and defect management scheme. Disk drives are usually non-volatile.

The defect management scheme of a disk drive is often shielded from the user, though some aspects can be evaluated and controlled by the initiator. The direct-access device will usually reserve some sectors and tracks for recording defect lists and for reassigning defective blocks. The READ LONG and WRITE LONG commands will typically access the user data and checksum portions of the data field so that defects may be induced by the initiator to test the defect detection logic of the direct-access device.

Notches find their most typical use in a rotating disk drive. On a disk, the inner tracks are physically shorter than the outer tracks. As a result, if each track is made to store the same number of data bits, the data is packed more densely on the inner tracks than the outer tracks. By using notches, the outer tracks may be made to contain a different number of sectors than the inner tracks, while balancing the data density. This results in increased capacity.

9.1.13.2 Sequential media

Some tape devices are implemented as a direct access device so that they can be used in disk oriented operating system environments. These devices are sometimes referred to as random access tape or floppy tape. These devices might be thought of as a disk drive with one or more long tracks. Access time to a logical block is usually longer than for a disk drive, since the tape must be fast forwarded or rewound to the block. As a result, the SEEK command will often be more useful for a tape than for a disk. The only way an initiator may determine if a direct-access device is a tape is by using the medium type code returned by the MODE SENSE command.

9.1.13.3 Memory media

Memory media includes devices that are traditionally used for primary storage within computer systems, such as solid state static or dynamic random access memories (SRAM or DRAM), or magnetic core or bubble memory. These devices are typically non-mechanical, and therefore the entire physical medium may be accessed in virtually the same access time. The data is typically accessed as a bit or byte and this also speeds access time. Memory devices typically store less data than disks or tapes, and are usually volatile when not protected by battery backup.

9.2 Commands for direct-access devices.

The commands for direct-access devices shall be as shown in table 108.

Table 108 - Commands for direct-access devices

Command name	Operation code	Type	Subclause
CHANGE DEFINITION	40h	O	8.2.1
COMPARE	39h	O	8.2.2
COPY	18h	O	8.2.3
COPY AND VERIFY	3Ah	O	8.2.4
FORMAT UNIT	04h	M	9.2.1
INQUIRY	12h	M	8.2.5
LOCK UNLOCK CACHE	36h	O	9.2.2
LOG SELECT	4Ch	O	8.2.6
LOG SENSE	4Dh	O	8.2.7
MODE SELECT(6)	15h	O	8.2.8
MODE SELECT(10)	55h	O	8.2.9
MODE SENSE(6)	1Ah	O	8.2.10
MODE SENSE(10)	5Ah	O	8.2.11
PRE-FETCH	34h	O	9.2.3
PREVENT ALLOW MEDIUM REMOVAL	1Eh	O	9.2.4
READ(6)	08h	M	9.2.5
READ(10)	28h	M	9.2.6
READ BUFFER	3Ch	O	8.2.12
READ CAPACITY	25h	M	9.2.7
READ DEFECT DATA	37h	O	9.2.8
READ LONG	3Eh	O	9.2.9
REASSIGN BLOCKS	07h	O	9.2.10
RECEIVE DIAGNOSTIC RESULTS	1Ch	O	8.2.13
RELEASE	17h	M	9.2.11
REQUEST SENSE	03h	M	8.2.14
RESERVE	16h	M	9.2.12
REZERO UNIT	01h	O	9.2.13
SEARCH DATA EQUAL	31h	O	9.2.14.1
SEARCH DATA HIGH	30h	O	9.2.14.2
SEARCH DATA LOW	32h	O	9.2.14.3
SEEK(6)	0Bh	O	9.2.15
SEEK(10)	2Bh	O	9.2.15
SEND DIAGNOSTIC	1Dh	M	8.2.15
SET LIMITS	33h	O	9.2.16
START STOP UNIT	1Bh	O	9.2.17
SYNCHRONIZE CACHE	35h	O	9.2.18
TEST UNIT READY	00h	M	8.2.16
VERIFY	2Fh	O	9.2.19
WRITE(6)	0Ah	O	9.2.20
WRITE(10)	2Ah	O	9.2.21
WRITE AND VERIFY	2Eh	O	9.2.22
WRITE BUFFER	3Bh	O	8.2.17
WRITE LONG	3Fh	O	9.2.23
WRITE SAME	41h	O	9.2.24

Key: M = Command implementation is mandatory.
O = Command implementation is optional.

The following operation codes are vendor-specific: 02h, 05h, 06h, 09h, 0Ch, 0Dh, 0Eh, 0Fh, 10h, 11h, 13h, 14h, 19h, 20h, 21h, 22h, 23h, 24h, 26h, 27h, 29h, 2Ch, 2Dh and C0h through FFh. All remaining operation codes for direct-access devices are reserved for future standardization.

9.2.1 FORMAT UNIT command

The FORMAT UNIT command (see table 109) formats the medium into initiator addressable logical blocks per the initiator defined options. In addition, the medium may be certified and control structures may be created for the management of the medium and defects. There is no guarantee that the medium has or has not been altered.

Table 109 - FORMAT UNIT command

Bit Byte	7	6	5	4	3	2	1	0
0	Operation code (04h)							
1	Logical unit number			FmtData	CmpLst	Defect list format		
2	Vendor-specific							
3	(MSB) Interleave (LSB)							
4								
5	Control							

The simplest mandatory form of the FORMAT UNIT command (with no format data) accomplishes medium formatting with little initiator control over defect management. The target implementation determines the degree of defect management that is to be performed. Two additional mandatory forms of this command increase the initiator's control over defect management. Several optional forms of this command further increase the initiator's control over defect management, by allowing the initiator to specify which defect list(s) are to be used, to specify defect locations (in several formats), to enable target certification, and to specify what to do in the event that defect lists are not accessible.

The FORMAT UNIT command shall be rejected with RESERVATION CONFLICT status if the logical unit is reserved, or any extent reservation, from any initiator, is active in the specified logical unit.

During the format operation, the target shall respond to commands as follows:

- a) In response to all commands except REQUEST SENSE and INQUIRY, the target shall return CHECK CONDITION status unless a reservation conflict exists, in which case RESERVATION CONFLICT status shall be returned.
- b) In response to the INQUIRY command, the target shall respond as commanded.
- c) In response to the REQUEST SENSE command, unless an error has occurred, the target shall return a sense key of NOT READY and an additional sense code of LOGICAL UNIT NOT READY FORMAT IN PROGRESS, with the sense-key specific bytes set for progress indication (as described in 8.2.14.1). Refer to 8.2.14.2 for a description of deferred error handling that may occur during the format operation.

NOTE 103 It is recommended that MODE SELECT parameters (if any) be set prior to issuing the FORMAT UNIT command.

During the execution of the FORMAT UNIT command, the target may perform a medium defect management algorithm (which can be controlled by the initiator, using optional forms of this command). Four sources of defect location information (hereafter called defects) are defined as follows:

- a) **Primary defect list (Plist).** This is the list of defects, usually supplied by the original manufacturer of the device or medium, that are considered permanent defects. The Plist is located outside of the initiator-accessible logical block space. The Plist is accessible by the target (to reference while formatting), but it is not normally accessible by the initiator except through the READ DEFECT DATA command. Once created, the original Plist shall not be subject to change.

- b) **Target certification list (Clist).** This list includes defects detected by the target during an optional certification process executed during the FORMAT UNIT command. This list shall be added to the Glist.
- c) **Data defect list (Dlist).** This list of defect descriptors may be supplied to the target by the initiator in the DATA OUT phase of the FORMAT UNIT command. This list shall be added to the Glist. The defect list length in the defect list header may be zero, in which case there is no Dlist.
- d) **Grown defect list (Glist).** The Glist includes all defects sent by the initiator or detected by the target. The Glist does not include the Plist. If the CmpLst bit is zero, the Glist shall include Dlists provided to the target during the previous and the current FORMAT UNIT commands. The Glist shall also include:
- defects detected by the format operation during medium certification;
 - defects previously identified with a REASSIGN BLOCKS command;
 - defects previously detected by the target and automatically reallocated.

A format data (FmtData) bit of one indicates that the FORMAT UNIT parameter list (see table 110) shall be transferred during the DATA OUT phase. The DATA OUT phase consists of a defect list header (see table 111), followed by an initialization pattern descriptor, followed by zero or more defect descriptors. Each defect descriptor identifies a location on the medium that the target shall map out of the user-accessible area.

Table 110 - FORMAT UNIT parameter list

Bit Byte	7	6	5	4	3	2	1	0
	Defect list header							
	Initialization pattern descriptor (if any)							
	Defect descriptor(s) (if any)							
0	Defect descriptor 0							
n	(See specific table for length.)							
	.							
0	Defect descriptor x							
n	(See specific table for length.)							

The defect list header (see table 111) provides several optional format control bits. Targets that implement these bits give the initiator additional control over the use of the four defect sources, and the formatting operation. If the initiator attempts to select any function not implemented by the target, the target shall terminate the command with CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and the additional sense code shall be set to INVALID FIELD IN PARAMETER LIST.

Table 111 - Defect list header

Bit Byte	7	6	5	4	3	2	1	0
0	Reserved							
1	FOV	DPRY	DCRT	STPF	IP	DSP	Immed	VS
2	(MSB)							
3	Defect list length							
	(LSB)							

A FmtData bit of zero indicates that a DATA OUT phase shall not occur. The source of defect information is not specified.

A complete list (CmpLst) bit of one indicates that the defect list sent by the initiator is a complete list of defects. Any existing defect list except the Plst shall be ignored by the target. As a result, a new Glist is constructed that contains the Dlist (if it is sent by the initiator), and the Clist (if certification is enabled). The target may add any defects it detects during the format operation to this Dlist.

A CmpLst bit of zero indicates that the defect list sent by the initiator is an addition to the existing list of defects. As a result a new Glist is constructed that contains the existing Glist, the Dlist (if it is sent by the initiator), and the Clist (if certification is enabled). The target may add any defects it detects during the format operation to this Dlist.

Table 112 defines the implementation requirements for the FORMAT UNIT command.

Table 112 - FORMAT UNIT defect descriptor format and requirements

FmtData	CmpLst	Defect list format	Defect list length	Type	Comments
0	0	000b	N/A	M	Vendor-specific
BLOCK FORMAT					
1	0	000b	Zero	M	See notes (1) and (3)
1	1	000b	Zero	M	See notes (1) and (4)
1	0	000b	>0	0	See notes (2) and (3)
1	1	000b	>0	0	See notes (2) and (4)
BYTES FROM INDEX FORMAT					
1	0	100b	Zero	0	See notes (1) and (3)
1	1	100b	Zero	0	See notes (1) and (4)
1	0	100b	>0	0	See notes (2) and (3)
1	1	100b	>0	0	See notes (2) and (4)
PHYSICAL SECTOR FORMAT					
1	0	101b	Zero	0	See notes (1) and (3)
1	1	101b	Zero	0	See notes (1) and (4)
1	0	101b	>0	0	See notes (2) and (3)
1	1	101b	>0	0	See notes (2) and (4)
VENDOR-SPECIFIC FORMAT					
1	0	110b			
1	1	110b			
All remaining codes are reserved.					
Key: M = Command implementation is mandatory. O = Command implementation is optional.					
NOTES 1 No Dlist is transferred to the target during the DATA OUT phase. 2 A Dlist is transferred to the target during the DATA OUT phase. Add the Dlist defects to the new Glist. 3 Use the existing Glist as a defect source. Add existing Glist defects to the new Glist. 4 Discard the existing Glist. Do not add existing Glist defects to the new Glist. 5 All the options described in this table cause a new Glist to be created during execution of the FORMAT UNIT command as described in the text.					

The defect list format field specifies which defect descriptor is used if the FmtData bit is one (see table 112).

The interleave field specifies the interleave that is used when performing the format operation. This allows the logical blocks to be related in a way that facilitates matching the transfer rate between the initiator and the peripheral. An interleave of zero specifies that the target use its default interleave. An interleave of one specifies that consecutive logical blocks be placed in contiguous ascending order. All other values are vendor-specific.

A format options valid (FOV) bit of zero indicates that the target shall use its default settings for the DPRY, DCRT, STPF, IP and DSP bits (see below). The initiator shall set these bits to zero. If any of these bits are not zero, the target shall terminate the command with CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and the additional sense code shall be set to INVALID FIELD IN PARAMETER LIST.

A FOV bit of one indicates that the target shall examine the setting of the DPRY, DCRT, STPF, IP and DSP bits. When the FOV bit is one, the DPRY, DCRT, STPF, IP and DSP bits are defined as follows.

A disable primary (DPRY) bit of zero indicates that the target shall not use portions of the medium identified as defective in the primary defect Plist for initiator addressable logical blocks. If the target cannot locate the Plist or it cannot determine whether a Plist exists, it shall perform the action specified by the STPF bit. A DPRY bit of one indicates that the target shall not use the Plist to identify defective areas of the medium. The Plist is not deleted.

A disable certification (DCRT) bit of zero indicates that the target shall perform a vendor-specific medium certification operation to generate a Clist. A DCRT bit of one indicates that the target shall not perform any vendor-specific medium certification process or format verification operation while executing the FORMAT UNIT command.

The stop format (STPF) bit controls the behaviour of the target when one of the following events occurs:

- a) The target has been requested to use the primary defect list (DPRY is set to zero), or the grown defect list (CmpLst is set to zero) and the target cannot locate the list nor determine whether the list exists.
- b) The target has been requested to use the primary defect list (DPRY is set to zero) or the grown defect list (CmpLst is set to zero), and the target encounters an error while accessing the defect list.

A STPF bit of zero indicates that, if one or both of the above conditions occurs, the target shall continue to execute the FORMAT UNIT command. The target shall return CHECK CONDITION status at the completion of the FORMAT UNIT command. The sense key shall be set to RECOVERED ERROR and the additional sense code shall be set to either DEFECT LIST NOT FOUND if the first condition occurred, or DEFECT LIST ERROR if the second condition occurred.

A STPF bit of one indicates that, if one or both of the above conditions occurs, the target shall terminate the FORMAT UNIT command with CHECK CONDITION status. The sense key shall be set to MEDIUM ERROR and the additional sense code shall be set to either DEFECT LIST NOT FOUND if the first condition occurred, or DEFECT LIST ERROR if the second condition occurred.

NOTE 104 The use of the FmtData bit, the CmpLst bit, and the defect header allow the initiator to control the source of the defect lists used by the FORMAT UNIT command. Setting the defect list length to zero allows the initiator to control the use of Plist and Clist without having to specify a Dlist.

An initialization pattern (IP) bit of one indicates that an initialization pattern descriptor (see 9.2.1.2) is included in the FORMAT UNIT parameter list immediately following the defect list header. An IP bit of zero indicates that an initialization pattern descriptor is not included and that the target shall use its default initialization pattern.

A disable saving parameters (DSP) bit of one specifies that the target shall not save the MODE SELECT savable parameters to non-volatile memory during the format operation. A DSP bit of zero specifies that the target shall save all the MODE SELECT savable parameters for all initiators to non-volatile memory during the format operation. Pages that are not reported as savable are not affected by the DSP bit (i.e. if pages 03h & 04h are not returned with the PS bit set they may be saved even if DSP is cleared).

An immediate (Immed) bit of zero indicates that status shall be returned after the format operation has completed. An Immed bit value of one indicates that the target shall return status as soon as the command descriptor block has been validated, and the entire defect list has been transferred.

The bit designated VS is vendor-specific.

The defect list length field in the defect list header specifies the total length in bytes of the defect descriptors that follow and does not include the initialization pattern descriptor or initialization pattern, if any. The length of the defect descriptors varies with the format of the defect list. The three formats for the defect descriptor(s) field in the defect lists are shown in 9.2.1.1.

9.2.1.1 Defect list formats

This clause describes the defect list formats used in the FORMAT UNIT, READ DEFECT DATA and translate page of the SEND DIAGNOSTIC and RECEIVE DIAGNOSTIC RESULTS commands.

NOTE 105 The selected reporting format accounts for variables that impact the information in the returned data. For example, the specific location of a defect, while constant in angular and radial location on the device, may change in reported location a format operation with different geometry parameters is performed. It is the responsibility of the initiator to use a defect list format appropriate for the intended operation with the current or future geometry parameters. If the target is able to detect that the selected defect list format would provide inconsistent results, the target may return CHECK CONDITION status.

Each block format defect descriptor (see table 113) specifies a four-byte defective block address that contains the defect.

Table 113 - Defect descriptor - Block format

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)							
3	Defective block address (LSB)							

The defect list length is equal to four times the number of defect descriptors.

The defect descriptors should be in ascending order. More than one physical or logical block may be affected by each defect descriptor. A target may return CHECK CONDITION if the defect descriptors are not in ascending order.

Each byte from index defect descriptor (see table 114) specifies the location of a defect that is no more than eight bytes long.

Table 114 - Defect descriptor - Bytes from index format

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)							
2	Cylinder number of defect (LSB)							
3	Head number of defect							
4	(MSB)							
7	Defect bytes from index (LSB)							

The defect list length is equal to eight times the number of defect descriptors.

Each descriptor is comprised of the cylinder number of defect, the head number of defect, and the defect bytes from index to the defect. The defect descriptors shall be in ascending order. The cylinder number of defect is the most significant part of the address and the defect bytes from index is the least significant part of the address. More than one physical or logical block may be affected by each defect. If the defect bytes from index has a value of FFFFFFFFh, this indicates that the entire track shall be considered defective.

Each physical sector defect descriptor (see table 115) specifies the location of a defect that is the length of a sector.

Table 115 - Defect descriptor - Physical sector format

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)							
2	Cylinder number of defect							(LSB)
3	Head number of defect							
4	(MSB)							
7	Defective sector number							(LSB)

The defect list length is equal to eight times the number of defect descriptors.

Each descriptor is comprised of a cylinder number of defect, the head number of defect, and the defective sector number. The defect descriptors shall be in ascending order. The cylinder number of defect is the most significant part of the address and the defective sector number is the least significant part of the address. More than one block may be affected by each defect descriptor. A defective sector number of FFFFFFFFh indicates that the entire track shall be considered defective.

9.2.1.2 Initialization pattern option

The initialization pattern option specifies that the logical blocks contain the specified initialization pattern. The initialization pattern descriptor (see table 116) is sent to the target as part of the FORMAT UNIT parameter list.

Table 116 - Initialization pattern descriptor

Bit Byte	7	6	5	4	3	2	1	0
0	IP modifier		Reserved					
1	Pattern type							
2	(MSB)							
3	Initialization pattern length							(LSB)
4	Initialization pattern							
n								

NOTE 106 The initialization pattern option is not intended for media analysis or certification. This option may only initialize the initiator accessible area of the media to the specified pattern and may not write to any initiator inaccessible areas of the disk.

The IP modifier field specifies the type and location of a header that modifies the initialization pattern (see table 117).

Table 117 - Initialization pattern modifier

IP modifier	Description
00b	No header. The target shall not modify the initialization pattern.
01b	The target shall overwrite the initialization pattern to write the logical block address in the first four bytes of the logical block. The logical block address shall be written with the most significant byte first.
10b	The target shall overwrite the initialization pattern to write the logical block address in the first four bytes of each physical block contained within the logical block. The lowest numbered logical block or part thereof that occurs within the physical block is used. The logical block address shall be written with the most significant byte first.
11b	Reserved.

The initialization pattern type field (see table 118) indicates the type of pattern the target shall use to initialize each logical block within the initiator accessible portion of the medium. All bytes within a logical block shall be written with the initialization pattern. The initialization pattern is modified by the IP modifier field as described in table 117.

Table 118 - Initialization pattern type

Initialization pattern type	Description
00h	Use default pattern. (note 1)
01h	Repeat the initialization pattern as required to fill the logical block. (note 2)
02h - 7Fh	Reserved
80h - FFh	Vendor-specific
Notes 1) If the initialization pattern length is not zero the target shall terminate the command with CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and the additional sense code to INVALID FIELD IN PARAMETER LIST. 2) If the initialization pattern length is zero the target shall terminate the command with CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and the additional sense code to INVALID FIELD IN PARAMETER LIST.	

The initialization pattern length field indicates the number of bytes contained in the initialization pattern. If the length exceeds the current logical block size the target shall terminate the command with CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and the additional sense code shall be set to INVALID FIELD IN PARAMETER LIST. The pattern is modified by the IP modifier field.

9.2.2 LOCK UNLOCK CACHE command

The LOCK UNLOCK CACHE command (see table 119) requests that the target disallow or allow logical blocks within the specified range to be removed from the cache memory by the target's cache replacement algorithm. Locked logical blocks may be written to the medium when modified, but a copy of the modified logical block shall remain in the cache memory.

Table 119 - LOCK UNLOCK CACHE command

Bit Byte	7	6	5	4	3	2	1	0
0	Operation code (36h)							
1	Logical unit number			Reserved			Lock	RelAdr
2	(MSB)							
3	Logical block address							
4								
5								
6	(LSB)							
7	Reserved							
8	(MSB)							
9	Number of blocks							
	(LSB)							
9	Control							

A lock bit of one indicates that any logical block in the specified range that is currently present in the cache memory shall be locked into cache memory. Only logical blocks that are already present in the cache memory are actually locked. A lock bit of zero indicates that all logical blocks in the specified range that are currently locked into the cache memory shall be unlocked, but not necessarily removed.

A relative address (RelAdr) bit of one indicates that the logical block address field is a two's complement displacement. This negative or positive displacement shall 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.

A RelAdr bit of zero indicates that the logical block address field specifies the first logical block of the range of logical blocks to be operated on by this command.

The number of blocks specifies the total number of contiguous logical blocks within the range. A number of blocks field of zero indicates that all remaining logical blocks on the logical unit shall be within the range.

Multiple locks may be in effect from more than one initiator. Locks from different initiators may overlap. An unlock of an overlapped area does not release the lock of another initiator.

9.2.3 PRE-FETCH command

The PRE-FETCH command (see table 120) requests that the target transfer the specified logical blocks to the cache memory. No data shall be transferred to the initiator.

Table 120 - PRE-FETCH command

Bit Byte	7	6	5	4	3	2	1	0
0	Operation code (34h)							
1	Logical unit number			Reserved			Immed	RelAdr
2	(MSB)							
3	Logical block address							
4								
5								
6	(LSB)							
6	Reserved							
7	(MSB)							
8	Transfer length							
9	(LSB)							
9	Control							

An immediate (Immed) bit of one indicates that status shall be returned as soon as the command descriptor block has been validated. An Immed bit of zero indicates that status shall be returned after the operation is complete.

See 9.2.2 for a definition of the RelAdr bit and the logical block address field.

The transfer length field specifies the number of contiguous logical blocks of data that shall be transferred to the target's cache memory. A transfer length of zero indicates that the contiguous logical blocks up to and including the last logical block of the logical unit shall be transferred to the target's cache memory. Any other value indicates the number of logical blocks that shall be transferred. The target may elect to not transfer logical blocks that already are contained in the cache memory.

If the Immed bit is zero and the specified logical blocks were successfully transferred to the cache memory, the target shall return CONDITION MET status. If the link bit (see 7.2.7) is one, the target shall return INTERMEDIATE-CONDITION MET status.

If Immed is one, and the unlocked cache memory has sufficient capacity to accept all of the specified logical blocks, the target shall return CONDITION MET status. If the link bit (see 7.2.7) is one, the target shall return INTERMEDIATE-CONDITION MET status.

If Immed is one, and the unlocked cache memory does not have sufficient capacity to accept all of the specified logical blocks, the target shall return GOOD status. The target shall transfer to cache memory as many logical blocks as will fit. If the link bit (see 7.2.7) is one, the target shall return INTERMEDIATE status.

9.2.4 PREVENT ALLOW MEDIUM REMOVAL command

The PREVENT ALLOW MEDIUM REMOVAL command (see table 121) requests that the target enable or disable the removal of the medium in the logical unit. This mechanism is independent of device reservations and the target shall not allow medium removal if any initiator currently has medium removal prevented.

Table 121 - PREVENT ALLOW MEDIUM REMOVAL command

Bit Byte	7	6	5	4	3	2	1	0
0	Operation code (1Eh)							
1	Logical unit number			Reserved				
2	Reserved							
3	Reserved							
4	Reserved							Prevent
5	Control							

The prevention of medium removal shall begin when any initiator issues a PREVENT ALLOW MEDIUM REMOVAL command with a prevent bit of one (medium removal prevented). The prevention of medium removal for the logical unit shall terminate:

- after all initiators that have medium removal prevented issue PREVENT ALLOW MEDIUM REMOVAL commands with a prevent bit of zero, and the target has successfully performed a synchronize cache operation;
- upon the receipt of a BUS DEVICE RESET message from any initiator; or
- upon a hard RESET condition.

While a prevention of medium removal condition is in effect the target shall inhibit mechanisms that normally allow removal of the medium by an operator.

9.2.5 READ(6) command

The READ(6) command (see table 122) requests that the target transfer data to the initiator. The most recent data value written in the addressed logical block shall be returned.

Table 122 - READ(6) command

Bit Byte	7	6	5	4	3	2	1	0
0	Operation code (08h)							
1	Logical unit number			(MSB)				
2	Logical block address							
3	(LSB)							
4	Transfer length							
5	Control							

The cache control bits (see 9.2.6) are not provided for this command. Targets with cache memory may have values for the cache control bits that affect the READ(6) command; however, no default value is defined by this standard. If explicit control is required, the READ(10) command should be used.

The logical block address field specifies the logical block at which the read operation shall begin.

The transfer length field specifies the number of contiguous logical blocks of data to be transferred. A transfer length of zero indicates that 256 logical blocks shall be transferred. Any other value indicates the number of logical blocks that shall be transferred.

9.2.6 READ(10) command

The READ(10) command (see table 123) requests that the target transfer data to the initiator. The most recent data value written in the addressed logical block shall be returned.

Table 123 - READ(10) command

Bit Byte	7	6	5	4	3	2	1	0
0	Operation code (28h)							
1	Logical unit number			DPO	FUA	Reserved		RelAdr
2	(MSB)							
3	Logical block address							
4								
5								
6	Reserved							
7	(MSB)							
8	Transfer length							
9	(LSB)							
	Control							

A disable page out (DPO) bit of one indicates that the target shall assign the logical blocks accessed by this command the lowest priority for being fetched into or retained by the cache. A DPO bit of one overrides any retention priority specified in the cache page (see 9.3.3.1). A DPO bit of zero indicates the priority shall be determined by the retention priority fields in the cache page. All other aspects of the algorithm implementing the cache memory replacement strategy are not defined by this standard.

NOTE 107 The DPO bit is used to control replacement of logical blocks in the cache memory when the host has information on the future usage of the logical blocks. If the DPO bit is set to one, the host knows the logical blocks accessed by the command are not likely to be accessed again in the near future and should not be put in the cache memory nor retained by the cache memory. If the DPO bit is zero, the host expects that logical blocks accessed by this command are likely to be accessed again in the near future.

A force unit access (FUA) bit of one indicates that the target shall access the media in performing the command prior to returning GOOD status. Read commands shall access the specified logical blocks from the media (i.e. the data is not directly retrieved from the cache). In the case where the cache contains a more recent version of a logical block than the media, the logical block shall first be written to the media. Write commands shall not return GOOD status until the logical blocks have actually been written on the media (i.e. the data is not write cached).

An FUA bit of zero indicates that the target may satisfy the command by accessing the cache memory. For read operations, any logical blocks that are contained in the cache memory may be transferred to the initiator directly from the cache memory. For write operations, logical blocks may be transferred directly to the cache memory. GOOD status may be returned to the initiator prior to writing the logical blocks to the medium. Any error that occurs after the GOOD status is returned is a deferred error, and information regarding the error is not reported until a subsequent command.

See 9.2.2 for a definition of the RelAdr bit and the logical block address field.

The transfer length field specifies the number of contiguous logical blocks of data that shall be transferred. A transfer length of zero indicates that no logical blocks shall be transferred. This condition shall not be considered an error. Any other value indicates the number of logical blocks that shall be transferred.

9.2.7 READ CAPACITY command

The READ CAPACITY command (see table 124) provides a means for the initiator to request information regarding the capacity of the logical unit.

Table 124 - READ CAPACITY command

Bit Byte	7	6	5	4	3	2	1	0
0	Operation code (25h)							
1	Logical unit number			Reserved				RelAdr
2	(MSB)							
3	Logical block address							
4								
5								
6	(LSB)							
6	Reserved							
7	Reserved							
8	Reserved							PMI
9	Control							

See 9.2.2 for a definition of the RelAdr bit and the logical block address field.

The logical block address shall be zero if the PMI bit is zero. If the PMI bit is zero and the logical block address is not zero, the target shall return a CHECK CONDITION status, the sense key shall be set to ILLEGAL REQUEST and the additional sense code set to ILLEGAL FIELD IN CDB.

A partial medium indicator (PMI) bit of zero indicates that the returned logical block address and the block length in bytes are those of the last logical block on the logical unit.

A PMI bit of one indicates that the returned logical block address and block length in bytes are those of the logical block address after which a substantial delay in data transfer will be encountered. This returned logical block address shall be greater than or equal to the logical block address specified by the RelAdr and logical block address fields in the command descriptor block.

NOTE 108 This function is intended to assist storage management software in determining whether there is sufficient space on the current track, cylinder, etc., to contain a frequently accessed data structure, such as a file directory or file index, without incurring an access delay.

The READ CAPACITY data (see table 125) shall be sent during the DATA IN phase of the command.

Table 125 - READ CAPACITY data

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB) _____ Returned logical block address _____ (LSB)							
3								
4	(MSB) _____ Block length In bytes _____ (LSB)							
7								

9.2.8 READ DEFECT DATA command

The READ DEFECT DATA command (see table 126) requests that the target transfer the medium defect data to the initiator.

Table 126 - READ DEFECT DATA command

Bit Byte	7	6	5	4	3	2	1	0
0	Operation code (37h)							
1	Logical unit number			Reserved				
2	Reserved			Plist	Glist	Defect list format		
3	Reserved							
4	Reserved							
5	Reserved							
6	Reserved							
7	(MSB)	Allocation length						(LSB)
8								
9	Control							

If the target is unable to access any medium defect data, it shall terminate the command with CHECK CONDITION status. The sense key shall be set to either MEDIUM ERROR, if a medium error occurred, or NO SENSE, if the list does not exist; and the additional sense code shall be set to DEFECT LIST NOT FOUND.

NOTE 109 Some targets may not be able to return medium defect data until after a FORMAT UNIT command has been completed successfully.

A primary defect list (Plist) bit of one request that the target return the primary list of defects. A Plist bit of zero requests that the target not return the primary list of defects.

A grown defect list (Glist) bit of one request that the target return the grown defect list. A Glist bit of zero requests that the target not return the grown defect list.

A Plist bit of one and a Glist bit of one requests that the target return the primary and the grown defect lists. The order in which the lists are returned is vendor-specific. Whether the lists are merged or not is vendor-specific.

A Plist bit of zero and a Glist bit of zero requests that the target return only the defect list header.

The defect list format field is used by the initiator to indicate the preferred format for the defect list. This field is intended for those targets capable of returning more than one format, as defined in the FORMAT UNIT command (see 9.2.1.2, defect list format). A target unable to return the requested format shall return the defect list in its default format (see the defect list format field in the defect list header below).

If the requested defect list format and the returned defect list format are not the same, the target shall transfer the defect data and then terminate the command with CHECK CONDITION status. The sense key shall be set to RECOVERED ERROR and the additional sense code shall be set to DEFECT LIST NOT FOUND.

The READ DEFECT DATA defect list (see table 127) contains a four-byte header, followed by zero or more defect descriptors.

Table 127 - READ DEFECT DATA defect list

Bit Byte	7	6	5	4	3	2	1	0
0	Reserved							
1	Reserved			Plist	Glist	Defect list format		
2	(MSB) Defect list length							
3	(LSB)							
	Defect descriptor(s) (if any)							
0	Defect descriptor 0							
n	(See specific table for length.)							
	.							
0	Defect descriptor x							
n	(See specific table for length.)							

A Plist bit of one indicates that the data returned contains the primary defect list. A Plist bit of zero indicates that the data returned does not contain the primary defect list.

A Glist bit of one indicates that the data returned contains the grown defect list. A Glist bit of zero indicates that the data returned does not contain the grown defect list.

The defect list format field indicates the format of the defect descriptors returned by the target. This field is defined in the FORMAT UNIT command (see 9.2.1.2).

NOTE 110 The use of the block format is not recommended. There is no universal model that sensibly defines the meaning of the logical block address of a defect. In the usual case, a defect that has been reassigned no longer has a logical block address.

Defect descriptors returned in the block format are vendor-specific. Defect descriptors returned in the physical sector format may or may not include defects in areas not accessible to the initiator. Defect descriptors returned in bytes-from-index format shall comprise a complete list of defects. A complete list of defects may include defects in areas not within the capacity returned in the READ CAPACITY command.

The defect list length field specifies the length in bytes of the defect descriptors that follow. The defect list length is equal to four or eight times the number of defect descriptors, depending on the format of the returned descriptors (see 9.2.1.1).

If the allocation length is insufficient to transfer all of the defect descriptors, the defect list length shall not be adjusted to reflect the truncation. The target shall not create CHECK CONDITION status. The initiator is responsible for comparing the defect list length and the allocation length to ensure that a partial list was not received.

NOTE 111 The initiator may determine the length of the defect list by sending the READ DEFECT DATA command with an allocation length of four. The target will return the defect list header that contains the length of the defect list.

The defect descriptors may or may not be sent in ascending order. The initiator may determine the exact number of defects by dividing the defect list length by the length of a single defect descriptor for the returned format.

9.2.9 READ LONG command

The READ LONG command (see table 128) requests that the target transfer data to the initiator. The data passed during the READ LONG command is vendor-specific, but shall include the data bytes and the ECC bytes recorded on the medium. The most recent data written in the addressed logical block shall be returned.

Table 128 - READ LONG command

Bit Byte	7	6	5	4	3	2	1	0
0	Operation code (3Eh)							
1	Logical unit number			Reserved			CORRCT	RelAdr
2	(MSB)							
3	Logical block address							
4								
5								
6								
7	(MSB)							
8	Byte transfer length							(LSB)
9	Control							

NOTE 112 Any other bytes that can be corrected by ECC should be included (e.g. data synchronization mark within the area covered by ECC). It is not important for the ECC bytes to be at the end of the data bytes; however, they should be in the same order as they are on the media.

A corrected (CORRCT) bit of zero causes a logical block to be read without any correction made by the target. A CORRCT bit of one causes the data to be corrected by ECC before being transferred to the initiator.

See 9.2.2 for a definition of the RelAdr bit and the logical block address field.

The byte transfer length field should specify exactly the number of bytes of data that are available for transfer. If a non-zero byte transfer length does not exactly match the available data length, the target shall terminate the command with CHECK CONDITION status, the sense key shall be set to ILLEGAL REQUEST and an additional sense code set to INVALID FIELD IN CDB. The valid and ILI bits shall be set to one and the information field shall be set to the difference (residue) of the requested length minus the actual length in bytes. Negative values shall be indicated by two's complement notation.

A byte transfer length of zero indicates that no bytes shall be transferred and shall not be considered an error.

9.2.10 REASSIGN BLOCKS command

The REASSIGN BLOCKS command (see table 129) requests the target to reassign the defective logical blocks to another area on the medium set aside for this purpose. The target should also record the location of the defective logical blocks to the grown defect list if such a list is supported. More than one physical or logical block may be relocated by each defect descriptor sent by the initiator. This command does not alter the contents or location of the Plist (see 9.2.1, FORMAT UNIT command).

Table 129 - REASSIGN BLOCKS command

Bit Byte	7	6	5	4	3	2	1	0
0	Operation code (07h)							
1	Logical unit number			Reserved				
2	Reserved							
3	Reserved							
4	Reserved							
5	Control							

The initiator transfers a defect list that contains the logical block addresses to be reassigned. The target shall reassign the physical medium used for each logical block address in the list. The data contained in the logical blocks specified in the defect list may be altered, but the data in all other logical blocks on the medium shall be preserved.

NOTE 113 The effect of specifying a logical block to be reassigned that previously has been reassigned is to reassign the block again. Over the life of the medium, a logical block can be assigned to multiple physical addresses until no more spare locations remain on the medium.

The REASSIGN BLOCKS defect list (see table 130) contains a four-byte header followed by one or more defect descriptors. The length of each defect descriptor is four bytes.

Table 130 - REASSIGN BLOCKS defect list

Bit Byte	7	6	5	4	3	2	1	0
0	Reserved							
1	Reserved							
2	(MSB)	Defect list length						_____
3								(LSB)
	Defect descriptor(s)							
0	(MSB)	Defect logical block address						_____
3								(LSB)
	:							
0	(MSB)	Defect logical block address						_____
3								(LSB)

The defect list length field specifies the total length in bytes of the defect descriptors that follow. The defect list length is equal to four times the number of defect descriptors and does not include the defect list header length.

The defect descriptor specifies a four-byte defect logical block address that contains the defect. The defect descriptors shall be in ascending order.

If the logical unit has insufficient capacity to reassign all of the logical blocks specified in the defect descriptors, the command shall terminate with CHECK CONDITION status, the sense key shall be set to HARDWARE ERROR and the additional sense code set to NO DEFECT SPARE LOCATION AVAILABLE.

If the logical unit is unable to successfully complete a REASSIGN BLOCKS command, the command shall terminate with CHECK CONDITION status with the appropriate sense information. The logical block address of the first defect descriptor not reassigned shall be returned in the command-specific information field of the sense data. If information about the first defect descriptor not reassigned is not available, or if all the defects have been reassigned, this field shall be set to FFFFFFFFh.

If the REASSIGN BLOCKS command failed due to an unexpected unrecoverable read error that would cause the loss of data in a block not specified in the defect list, the logical block address of the unrecoverable block shall be returned in the information field of the sense data and the valid bit shall be set to one.

NOTE 114 If the REASSIGN BLOCKS command returns CHECK CONDITION status and the sense data command-specific information field contains a valid logical block address, the initiator should remove all defect descriptors from the defect list prior to the one returned in the command-specific information field. If the sense key is MEDIUM ERROR and the valid bit is one (the information field contains the valid block address) the initiator should insert that new defective logical block address into the defect list and reissue the REASSIGN BLOCKS command with the new defect list. Otherwise, the initiator should perform any corrective action indicated by the sense data and then reissue the REASSIGN BLOCKS command with the new defect list.

9.2.11 RELEASE command

The RELEASE command (see table 131) is used to release a previously reserved logical unit, or, if the extent release option is implemented, to release previously reserved extents within a logical unit.

Table 131 - RELEASE command

Bit Byte	7	6	5	4	3	2	1	0
0	Operation code (17h)							
1	Logical unit number			3rdPty	Third party device ID			Extent
2	Reservation identification							
3	Reserved							
4	Reserved							
5	Control							

The RESERVE and RELEASE commands provide the basic mechanism for contention resolution in multiple-initiator systems. A reservation may only be released by the initiator that made it. It is not an error for an initiator to attempt to release a reservation that is not currently valid, or is held by another initiator. In this case, the target shall return GOOD status without altering any other reservation.

NOTE 115 The reservation queuing option in X3.131-1986 has been removed from SCSI-2.

9.2.11.1 Logical unit release

Implementation of logical unit release is mandatory. If the extent bit is zero, this command shall cause the target to terminate all non-third-party logical unit and extent reservations that are active from the initiator to the specified logical unit. The reservation ID field in the command descriptor block is ignored by the target.

9.2.11.2 Extent release

Implementation of extent release is optional. If the extent bit is one and the extent release option is not implemented, then the RELEASE command shall be terminated with CHECK CONDITION status and the sense key shall be set to ILLEGAL REQUEST. This option shall be implemented if the extent reservation option (see 9.2.12.2) is implemented.

If the extent bit is one and the extent release option is implemented, this command shall cause any reservation from the requesting initiator with a matching reservation identification to be terminated. Other reservations from the requesting initiator shall remain in effect.

9.2.11.3 Third-party release

Implementation of third-party release is mandatory. Third-party release allows an initiator to release a logical unit or extents within a logical unit that were previously reserved using third-party reservation (see 9.2.12.3). Third-party release shall be implemented and is intended for use in multiple-initiator systems that use the COPY command.

If the third-party (3rdPty) bit is zero, then a third-party release is not requested. If the 3rdPty bit is one then the target shall release the specified logical unit or extents, but only if the reservation was made using a third-party reservation by the initiator that is requesting the release for the same SCSI device as specified in the third-party device ID field.

If the 3rdPty bit is one the target shall not modify the mode parameters for commands received from the third-party device even if the target implements the transfer of mode parameters with a third-party RESERVE command.

NOTE 116 If a target implements independent storage of mode parameters for each initiator, a third-party RESERVE command copies the current mode parameters for the initiator that sent the RESERVE command to the current mode parameters for the initiator specified as the third-party device (usually a copy master device). A unit attention condition notifies the third-party of the changed mode parameters due to the reservation. A successful third-party RELEASE command does not return the third-party devices' current mode parameters back to their previous values. The third-party device can issue MODE SENSE and MODE SELECT commands to query and modify the mode parameters.

9.2.12 RESERVE command

The RESERVE command (see table 132) is used to reserve a logical unit or, if the extent reservation option is implemented, extents within a logical unit.

Table 132 - RESERVE command

Bit Byte	7	6	5	4	3	2	1	0
0	Operation code (16h)							
1	Logical unit number			3rdPty	Third party device ID			Extent
2	Reservation identification							
3	(MSB)							
4	Extent list length							(LSB)
5	Control							

The RESERVE and RELEASE commands provide the basic mechanism for contention resolution in multiple-initiator systems. The third-party reservation allows logical units or extents to be reserved for another specified SCSI device.

NOTE 117 The reservation queuing option in X3.131-1986 has been removed from SCSI-2.

9.2.12.1 Logical unit reservation

Implementation of logical unit reservation is mandatory. If the extent bit is zero, this command shall request that the entire logical unit be reserved for the exclusive use of the initiator until the reservation is superseded by another valid RESERVE command from the initiator that made the reservation or until released by a RELEASE command from the same initiator that made the reservation, by a BUS DEVICE RESET message from any initiator, by a hard RESET condition, or by a power on cycle. A logical unit reservation shall not be granted if the logical unit or any extent is reserved by another initiator. It shall be permissible for an initiator to reserve a logical unit that is currently reserved by that initiator. If the extent bit is zero, the reservation identification and the extent list length shall be ignored.

If the logical unit, or any extent within the logical unit is reserved for another initiator, the target shall return RESERVATION CONFLICT status.

If, after honouring the reservation, any other initiator attempts to perform any command on the reserved logical unit other than an INQUIRY, REQUEST SENSE, PREVENT ALLOW MEDIUM REMOVAL (with a prevent bit of zero), or a RELEASE command the command shall be rejected with RESERVATION CONFLICT status.

9.2.12.2 Extent reservation

Implementation of extent reservation is optional. The reservation identification field provides a means for an initiator to identify each extent reservation. This allows an initiator in a multiple tasking environment, to have multiple reservations outstanding. The reservation identification is used in the RELEASE command to specify which reservation is to be released. It is also used in superseding RESERVE commands to specify which reservation is to be superseded.

If the extent reservation option is implemented, then the extent release option (see 9.2.11.2) shall also be implemented. These options permit multiple extents within the logical unit to be reserved, each with a separate reservation type.

If the extent bit is one, and the extent reservation option is implemented, then the target shall process the reservation request as follows:

- a) The extent list shall be checked for the number of extents in the reservation request. If the extent list length is zero, no current reservations shall be changed, no new reservations shall be created, and this condition shall not be treated as an error. If the extent list contains more extents than are supported on the logical unit, the command shall be terminated with CHECK CONDITION status and the sense key shall be set to ILLEGAL REQUEST. If the extent list contains more extents than are currently available on the logical unit, then the target shall return a RESERVATION CONFLICT status.
- b) The extent list shall be checked for valid extent logical block addresses. If any logical block address is invalid for this logical unit, the command shall be terminated with CHECK CONDITION status and the sense key shall be set to ILLEGAL REQUEST. The extent list shall be checked for invalid extent overlaps (as defined by reservation type) with other extent descriptors in the extent list and if invalid overlaps are found, the command shall be terminated with CHECK CONDITION status and the sense key shall be set to ILLEGAL REQUEST.
- c) If the requested reservation does not conflict with an existing reservation, the extents specified shall be reserved until superseded by another valid RESERVE command from the initiator that made the reservation or until released by a RELEASE command from the same initiator, by a BUS DEVICE RESET message from any initiator, or by a hard RESET condition. If either of the last two conditions occur, the next command from each initiator shall be terminated with CHECK CONDITION status and the sense key shall be set to UNIT ATTENTION.
- d) If the reservation request conflicts with an existing reservation, then the target shall return a RESERVATION CONFLICT status.

If the extent bit is one, and the extent reservation option is not implemented, then the RESERVE command shall be rejected with CHECK CONDITION status and the sense key shall be set to ILLEGAL REQUEST.

The size of the extent list shall be defined by the extent list length field. The extent list shall consist of zero or more descriptors as shown in table 133. Each extent descriptor defines an extent beginning at the specified logical block address for the specified number of blocks. If the number of blocks is zero, the extent shall begin at the specified logical block address and continue through the last logical block address on the logical unit.

Table 133 - Data format of extent descriptors

Bit Byte	7	6	5	4	3	2	1	0
0	Reserved					RelAdr	Reservation type	
1	(MSB)							
3	Number of blocks							(LSB)
4	(MSB)							
7	Logical block address							(LSB)

The reservation type field shall define the type of reservation in effect for the extent. The types of reservation are defined in table 134.

Table 134 - Reservation types

Reservation type	Description
00b	Read shared
01b	Write exclusive
10b	Read exclusive
11b	Exclusive access

- a) **Read exclusive.** While this reservation is active, no other initiator shall be permitted read operations to the indicated extent. This reservation shall not inhibit write operations from any initiator or conflict with a write

exclusive reservation; however, read exclusive, exclusive access, and read shared reservations that overlap this extent shall conflict with this reservation.

- b) **Write exclusive.** While this reservation is active, no other initiator shall be permitted write operations to the indicated extent. This reservation shall not inhibit read operations from any initiator or conflict with a read exclusive reservation from any initiator. This reservation shall conflict with write exclusive, exclusive access, and read shared reservations that overlap this extent.
- c) **Exclusive access.** While this reservation is active, no other initiator shall be permitted any access to the indicated extent. All reservation types that overlap this extent shall conflict with this reservation.
- d) **Read shared.** While this reservation is active, no write operations shall be permitted by any initiator to the indicated extent. This reservation shall not inhibit read operations from any initiator or conflict with a read shared reservation. Read exclusive, write exclusive, and exclusive access reservations that overlap with this extent shall conflict with this reservation.

If the relative address bit is one, the logical block address in the extent descriptor shall be treated as a two's complement displacement. This displacement shall be added to the logical block address last accessed on the logical unit to form the logical block address for this extent. This feature is only available when linking commands and requires that a previous command in the linked group has accessed a logical block on the logical unit; if not, the RESERVE command shall be terminated with CHECK CONDITION status and the sense key shall be set to ILLEGAL REQUEST.

If an initiator attempts a command to a logical block that has been reserved and that access is prohibited by the reservation, the command shall not be performed and the command shall be terminated with a RESERVATION CONFLICT status. If a reservation conflict precludes any part of the command, none of the command shall be performed. COPY commands shall be terminated with CHECK CONDITION status and the sense key shall be set to DATA PROTECT if any part of the copy operation is prohibited by an active reservation. If any extent in a logical unit is reserved in any way, by any initiator, a FORMAT UNIT command shall be rejected with a RESERVATION CONFLICT status.

9.2.12.3 Third-party reservation

Implementation of third-party reservation is mandatory. The third-party reservation for the RESERVE command allows an initiator to reserve a logical unit or extents within a logical unit for another SCSI device. This is intended for use in multiple-initiator systems that use the COPY command. Third-party reservation is required.

If the third-party (3rdPty) bit is zero, then a third-party reservation is not requested. If the 3rdPty bit is one then the target shall reserve the specified logical unit or extents for the SCSI device specified in the third-party device ID field. The target shall preserve the reservation until it is superseded by another valid RESERVE command from the initiator that made the reservation or until it is released by the same initiator, by a BUS DEVICE reset message from any initiator, or a hard reset condition. The target shall ignore any attempt to release the reservation made by any other initiator.

If independent sets of parameters are implemented, a third party reservation shall cause the target to transfer the set of parameters in effect for the initiator of the RESERVE command to the parameters used for commands from the third party device. Any subsequent command issued by the third-party device is executed according to the mode parameters in effect for the initiator that sent the RESERVE command.

NOTE 118 This transfer of the mode parameters is applicable to target devices which store mode information independently for different initiators. This mechanism allows an initiator to set the mode parameters of a target for the use of a copy master (i.e. the third-party device). The third-party copy master may subsequently issue a MODE SELECT command to modify the mode parameters.

9.2.12.4 Superseding reservations

Implementation of superseding reservations is mandatory. An initiator that holds a current reservation (unit or extent) may modify that reservation by issuing another RESERVE command (unit or extent) to the same logical unit. The superseding RESERVE command shall release the previous reservation state (unit or extent) when the new reservation request is granted. If the superseding reservation is for an extent reservation and the current reservation is also an extent reservation, the current extent reservation identification value is used for the superseding reservation. The current reservation shall not be modified if the superseding reservation request cannot be granted. If the superseding reservation cannot be granted because of conflicts with a previous reservation (other than the reservation being superseded), then the target shall return RESERVATION CONFLICT status.

NOTE 119 Superseding reservations allow the SCSI device ID to be changed on a reservation using the third-party reservation option. This capability is necessary for certain situations when using COMPARE, COPY, and COPY AND VERIFY commands.

9.2.13 REZERO UNIT command

The REZERO UNIT command (see table 135) requests that the target set the logical unit to a specific state. See vendor specifications for details.

Table 135 - REZERO UNIT command

Bit Byte	7	6	5	4	3	2	1	0
0	Operation code (01h)							
1	Logical unit number			Reserved				
2	Reserved							
3	Reserved							
4	Reserved							
5	Control							

9.2.14 SEARCH DATA commands

The SEARCH DATA commands (see table 136) search one or more logical blocks for equality or inequality to a data pattern. The concept of records within a logical block is used to allow multiple records within a logical block to be searched.

Table 136 - SEARCH DATA commands

Bit Byte	7	6	5	4	3	2	1	0
0	Operation code (31h 30h 32h)							
1	Logical unit number			Invert	Reserved		SpnDat	RelAdr
2	(MSB)							
3	Logical block address							
4								
5								
6								
7	(LSB)							
8	Reserved							
9	(MSB)							
10	Number of blocks to search							
11	(LSB)							
12	Control							

The invert bit determines whether the search condition is to be inverted. See 9.2.14.1 through 9.2.14.3 for a description of the search conditions for the individual SEARCH DATA commands.

A spanned data (SpnDat) bit of zero indicates that each record shall be wholly contained within a single block. Any space at the end of a block that is smaller than the record length is ignored by the SEARCH DATA commands. A SpnDat bit of one indicates that records span block boundaries (i.e. record may start in one block and end in the next or a subsequent block).

The number of blocks to search field specifies the maximum number of contiguous logical blocks to be searched. A value of zero indicates that no logical blocks shall be searched. This condition shall not be considered an error. Any other value indicates the maximum number of logical blocks that shall be searched.

A link bit (see 7.2.7) of zero indicates a non-linked command and if the search is satisfied, the command shall be terminated with a CONDITION MET status. A REQUEST SENSE command can then be issued to determine the logical block address and record offset of the matching record. If the search is not satisfied and no error occurs, the command shall be terminated with GOOD status.

A link bit (see 7.2.7) of one indicates a command is linked to the SEARCH DATA command and if the search is satisfied, INTERMEDIATE-CONDITION MET status is returned and the next command is executed. If the RelAdr bit in the next command is one, the logical block address of the next command is used as a displacement from the logical block address at which the search was satisfied. If a linked search is not satisfied, the command is terminated with CHECK CONDITION status. A REQUEST SENSE command may then be issued.

A REQUEST SENSE command following a satisfied SEARCH DATA command shall:

- return a sense key of EQUAL if the search was satisfied by an exact match. If the search was satisfied by an inequality then a sense key of NO SENSE shall be returned.
- return the valid bit set to one.
- return the logical block address of the logical block containing the first matching record in the information field.
- return the record offset of the matching record in the command-specific information field.

A REQUEST SENSE command following a SEARCH DATA command that is not satisfied shall:

- a) return a sense key of NO SENSE, if no errors occurred during the command execution.
- b) return the valid bit set to zero.

The SEARCH DATA parameter list (see table 137) contains a fourteen-byte header, followed by one or more search argument descriptors.

Table 137 - SEARCH DATA parameter list

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB) Logical record length (LSB)							
3								
4	(MSB) First record offset (LSB)							
7								
8	(MSB) Number of records (LSB)							
11								
12	(MSB) Search argument length (LSB)							
13								
	Search argument descriptor(s)							
0	(MSB) Displacement (LSB)							
3								
4	(MSB) Pattern length (LSB)							
5								
6	Pattern							
n								

The logical record length field specifies the record length in bytes.

The first record offset field specifies the number of bytes that shall be ignored in the first logical block before the search begins. If the value of the first record offset field shall be larger than the logical block length the target shall terminate the command with a CHECK CONDITION status, set the sense key to ILLEGAL REQUEST and set the additional sense code to INVALID FIELD IN PARAMETERS LIST. Subsequent logical blocks shall be searched beginning with the first byte in the logical block. This permits one or more records to be skipped initially.

The number of records field specifies the maximum number of records that shall be searched by this command. An search shall terminate when the search pattern is found or when the number of records is exhausted or when the number of blocks to search is exhausted.

The search argument length field specifies the length in bytes of all the search argument descriptors that follow.

NOTE 120 Since the pattern length can vary, there is no fixed multiple of the search argument descriptor to determine the search argument length.

The search argument descriptors specify one or more search conditions to execute within a single record in order to satisfy the search. Each search argument descriptor is made up of a displacement field, a pattern length field, and a pattern field.

The displacement field specifies the displacement in bytes of the first byte of the data to be compared from the start of the logical record.

The pattern length field specifies the length in bytes of the pattern that follows.

The pattern field specifies the data to compare to the logical record.

9.2.14.1 SEARCH DATA EQUAL command

The SEARCH DATA EQUAL command (see table 136, operation code 31h) shall be satisfied by the first logical record searched that contains data that satisfies all of the search argument descriptor(s). If the invert bit in the command descriptor block is zero, the search argument descriptor(s) shall be satisfied by data in the logical record being equal to the data in the pattern. If the invert bit is one, the search argument descriptor(s) shall be satisfied by data in the logical record being not equal to the data in the pattern. (See 9.2.14.)

9.2.14.2 SEARCH DATA HIGH command

The SEARCH DATA HIGH command (see table 136, operation code 30h) shall be satisfied by the first logical record searched that contains data that satisfies all of the search argument descriptor(s). If the invert bit in the command descriptor block is zero, the search argument descriptor(s) shall be satisfied by data in the logical record being greater than the data in the pattern. If the invert bit is one, the search argument descriptor(s) shall be satisfied by data in the logical record being less than or equal to the data in the pattern. (See 9.2.14.)

9.2.14.3 SEARCH DATA LOW command

The SEARCH DATA LOW command (see table 136, operation code 32h) shall be satisfied by the first logical record searched that contains data that satisfies all of the search argument descriptor(s). If the invert bit in the command descriptor block is zero, the search argument descriptor(s) shall be satisfied by data in the logical record being less than the data in the pattern. If the invert bit is one, the search argument descriptor(s) shall be satisfied by data in the logical record being greater than or equal to the data in the pattern. (See 9.2.14.)

9.2.15 SEEK(6) and SEEK(10) commands

The SEEK(6) (see table 138) and SEEK(10) (see table 139) commands request that the logical unit seek to the specified logical block address.

Table 138 - SEEK(6) command

Bit Byte	7	6	5	4	3	2	1	0
0	Operation code (0Bh)							
1	Logical unit number			(MSB)				
2	Logical block address							
3	(LSB)							
4	Reserved							
5	Control							

Table 139 - SEEK(10) command

Bit Byte	7	6	5	4	3	2	1	0
0	Operation code (2Bh)							
1	Logical unit number			Reserved				
2	(MSB)							
3	Logical block address							
4								
5								
6								
7	(LSB)							
8	Reserved							
9	Reserved							
10	Reserved							
11	Control							

9.2.16 SET LIMITS command

The SET LIMITS command (see table 140) defines the range within which subsequent linked commands may operate. A second SET LIMITS command may not be linked to a chain of commands in which a SET LIMITS command has already been issued.

Table 140 - SET LIMITS command

Bit Byte	7	6	5	4	3	2	1	0
0	Operation code (33h)							
1	Logical unit number			Reserved			RdInh	WrInh
2	(MSB)							
3	Logical block address							
4								
5								
6								
7	(LSB)							
6	Reserved							
7	(MSB)							
8	Number of blocks							(LSB)
9	Control							

The read inhibit (RdInh) bit of one indicates that read operations within the range shall be inhibited.

A write inhibit (WrInh) bit of one indicates that write operations within the range shall be inhibited.

The logical block address field specifies the starting address for the range.

The number of blocks field specifies the number of logical blocks within the range. A number of blocks of zero indicates that the range shall extend to the last logical block on the logical unit.

Any attempt to access outside of the restricted range or any attempt to perform an inhibited operation within the restricted range shall not be performed. The command shall be terminated with CHECK CONDITION status and the sense key shall be set to DATA PROTECT. A second SET LIMITS command within a linked list of commands shall be rejected with CHECK CONDITION status and the sense key shall be set to DATA PROTECT.

9.2.17 START STOP UNIT command

The START STOP UNIT command (see table 141) requests that the target enable or disable the logical unit for media access operations.

Table 141 - START STOP UNIT command

Bit Byte	7	6	5	4	3	2	1	0
0	Operation code (1Bh)							
1	Logical unit number			Reserved				Immed
2	Reserved							
3	Reserved							
4	Reserved						LoEj	Start
5	Control							

An immediate (Immed) bit of one indicates that status shall be returned as soon as the command descriptor block has been validated. An Immed bit of zero indicates that status shall be returned after the operation is completed.

A load eject (LoEj) bit of zero requests that no action be taken regarding loading or ejecting the medium. A LoEj bit of one requests that the medium shall be unloaded if the start bit is zero. A LoEj bit of one requests that the medium is to be loaded if the start bit is one.

A start bit of one requests the logical unit be made ready for use. A start bit of zero requests that the logical unit be stopped (media cannot be accessed by the initiator).

Targets that contain cache memory shall implicitly perform a SYNCHRONIZE CACHE command for the entire medium prior to executing the STOP UNIT command.

9.2.18 SYNCHRONIZE CACHE command

The SYNCHRONIZE CACHE command (see table 142) ensures that logical blocks in the cache memory, within the specified range, have their most recent data value recorded on the physical medium. If a more recent data value for a logical block within the specified range exists in the cache memory than on the physical medium, then the logical block from the cache memory shall be written to the physical medium. Logical blocks are not necessarily removed from the cache memory as a result of the synchronize cache operation.

Table 142 - SYNCHRONIZE CACHE command

Bit Byte	7	6	5	4	3	2	1	0
0	Operation code (35h)							
1	Logical unit number			Reserved			Immed	RelAdr
2	(MSB)							
3	Logical block address							
4								
5								
6	(LSB)							
6	Reserved							
7	(MSB)							
8	Number of blocks							(LSB)
9	Control							

An immediate (Immed) bit of one indicates that the target shall return status as soon as the command descriptor block has been validated. An Immed bit of zero indicates that the status shall not be returned until the operation has been completed. If the Immed bit is one and the target does not support it then the command shall terminate with CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and the additional sense code shall be set to INVALID FIELD IN CDB.

See 9.2.2 for a definition of the RelAdr bit and the logical block address field.

The number of blocks field specifies the total number of contiguous logical blocks within the range. A number of blocks of zero indicates that all remaining logical blocks on the logical unit shall be within the range.

A logical block within the specified range that is not in cache memory is not considered an error.

9.2.19 VERIFY command

The VERIFY command (see table 143) requests that the target verify the data written on the medium.

Table 143 - VERIFY command

Bit Byte	7	6	5	4	3	2	1	0
0	Operation code (2Fh)							
1	Logical unit number			DPO	Reserved	Reserved	BytChk	RelAdr
2	(MSB)							
3	Logical block address							
4								
5								
6								
7	Reserved							
8	(MSB)							
9	Verification length							
	(LSB)							
	Control							

If the MODE SELECT command is implemented, and the verify error recovery parameters page is also implemented, then the current settings in that page specifies the verification error criteria. If the verify error recovery parameters page is not implemented, then the verification criteria is vendor-specific.

A byte check (BytChk) bit of zero causes a medium verification to be performed with no data comparison. A BytChk bit of one causes a byte-by-byte compare of data written on the medium and the data transferred from the initiator. If the compare is unsuccessful for any reason, the target shall return CHECK CONDITION status with the sense key set to MISCOMPARE.

See 9.2.2 for a definition of the RelAdr bit and the logical block address field.

The verification length field specifies the number of contiguous logical blocks of data that shall be verified. A transfer length of zero indicates that no logical blocks shall be verified. This condition shall not be considered as an error. Any other value indicates the number of logical blocks that shall be verified.

See 9.2.6 for a description of the cache control bit (DPO).

9.2.20 WRITE(6) command

The WRITE(6) command (see table 144) requests that the target write the data transferred by the initiator to the medium.

Table 144 - WRITE(6) command

Bit Byte	7	6	5	4	3	2	1	0
0	Operation code (0Ah)							
1	Logical unit number			(MSB)				
2	Logical block address							
3	(LSB)							
4	Transfer length							
5	Control							

The cache control bits (see 9.2.6) are not provided for this command. Targets with cache memory may have values for the cache control bits which may affect the WRITE(6) command, however no default value is defined by this standard. If explicit control is required, the WRITE(10) command should be used.

The logical block address field specifies the logical block at which the write operation shall begin.

The transfer length field specifies the number of contiguous logical blocks of data to transferred. A transfer length of zero indicates that 256 logical blocks shall be transferred. Any other value indicates the number of logical blocks that shall be transferred.

9.2.21 WRITE(10) command

The WRITE(10) command (see table 145) requests that the target write the data transferred by the initiator to the medium.

Table 145 - WRITE(10) command

Bit Byte	7	6	5	4	3	2	1	0
0	Operation code (2Ah)							
1	Logical unit number			DPO	FUA	Reserved	Reserved	RelAdr
2	(MSB)							
3	Logical block address							
4								
5	(LSB)							
6	Reserved							
7	(MSB)							
8	Transfer length							
9	(LSB)							
9	Control							

See READ(10) command (9.2.6) for a definition of the cache control bits (DPO and FUA).

See 9.2.2 for a definition of the RelAdr bit and the logical block address field.

The transfer length field specifies the number of contiguous logical blocks of data that shall be transferred. A transfer length of zero indicates that no logical blocks shall be transferred. This condition shall not be considered an error and no data shall be written. Any other value indicates the number of logical blocks that shall be transferred.

9.2.22 WRITE AND VERIFY command

The WRITE AND VERIFY command (see table 146) requests that the target write the data transferred from the initiator to the medium and then verify that the data is correctly written. The data is only transferred once from the initiator to the target.

Table 146 - WRITE AND VERIFY command

Bit Byte	7	6	5	4	3	2	1	0
0	Operation code (2Eh)							
1	Logical unit number			DPO	Reserved	Reserved	BytChk	RelAdr
2	(MSB)							
3	Logical block address							
4								
5								
6	(LSB)							
7	Reserved							
8	(MSB)							
9	Transfer length							
	(LSB)							
9	Control							

If the MODE SELECT command is implemented, and the verify error recovery page is also implemented (see 9.3.3.8), then the current settings in that page along with the AWRE bit from the read-write error recovery page specify the verification error criteria. If these pages are not implemented, then the verification criteria is vendor-specific.

A byte check (BytChk) bit of zero requests a medium verification to be performed with no data comparison. A BytChk bit of one requests a byte-by-byte compare of data written on the medium and the data transferred from the initiator. If the compare is unsuccessful for any reason, the target shall return CHECK CONDITION status with the sense key set to MISCOMPARE.

See 9.2.2 for a definition of the RelAdr bit and the logical block address field.

See WRITE(10) command (see 9.2.21) for a definition of the transfer length field.

See 9.2.6 for a description of the cache control bit (DPO).

NOTE 121 The WRITE AND VERIFY command specifically states that the data are not to be transferred twice (i.e. once for the write pass, and once for the verify pass) when performing a byte compare. If there is a need for two transfers to occur (e.g. to ensure the integrity of the path to the media), then the initiator should issue a WRITE command with a LINK bit of one followed by a VERIFY command with a BytCmp bit of one, transferring the same data on each command.

9.2.23 WRITE LONG command

The WRITE LONG command (see table 147) requests that the target write the data transferred by the initiator to the medium. The data passed during the WRITE LONG command is implementation specific, but shall include the data bytes and the ECC bytes.

Table 147 - WRITE LONG command

Bit Byte	7	6	5	4	3	2	1	0
0	Operation code (3Fh)							
1	Logical unit number			Reserved				RelAdr
2	(MSB)							
3	Logical block address							
4								
5								
6								
7	(LSB)							
8	Reserved							
9	(MSB)							
10	Byte transfer length							
11	(LSB)							
12	Control							

NOTE 122 Any other bytes that can be corrected by ECC should be included (e.g. a data synchronization mark within the area covered by ECC). The READ LONG command is usually issued before issuing a WRITE LONG command. The WRITE LONG data should be the same length and in the same order as the data returned by the READ LONG command.

See 9.2.2 for a definition of the RelAdr bit and the logical block address field.

The byte transfer length field should specify the number of bytes of data that the target would return for the READ LONG command. If a non-zero byte transfer length does not exactly match the data length the target would return for the READ LONG command, then the target shall terminate the command with CHECK CONDITION status and a sense key of ILLEGAL REQUEST and an additional sense code of INVALID FIELD IN CDB. The ILI and valid bits shall be set to one and the information field shall be set to the difference (residue) of the requested length minus the actual length in bytes. Negative values shall be indicated by two's complement notation. A transfer length of zero indicates that no bytes shall be transferred and shall not be considered an error.

9.2.24 WRITE SAME command

The WRITE SAME command (see table 148) requests that the target write the single block of data transferred by the initiator to the medium multiple times.

Table 148 - WRITE SAME command

Bit Byte	7	6	5	4	3	2	1	0
0	Operation code (41h)							
1	Logical unit number			Reserved		PBdata	LBdata	RelAdr
2	(MSB)							
3	Logical block address							
4								
5								
6	(LSB)							
7	Reserved							
8	(MSB)							
9	Number of blocks							(LSB)
9	Control							

NOTE 123 This command is useful if large areas of the medium need to be written, prepared for certification, or otherwise initialized without the initiator having to transfer all the data.

A logical block data (LBdata) bit of one requests that the target replace the first four bytes of the data to be written to the current logical block with the logical block address of the block currently being written.

A physical block data (PBdata) bit of one requests that the target replace the first eight bytes of the data to be written to the current physical sector with the physical address of the sector currently being written using the physical sector format (see 9.2.1.1).

If PBdata and LBdata are one the command shall be terminated with CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST.

See 9.2.2 for a definition of the RelAdr bit and the logical block address field.

The number of blocks field specifies the number of contiguous logical blocks to be written. A number of blocks field of zero requests that all the remaining logical blocks on the medium be written.

9.3 Parameters for direct-access devices

9.3.1 Diagnostic parameters

This subclause defines the descriptors and pages for diagnostic parameters used with direct-access devices.

The diagnostic page codes for direct-access devices are defined in table 149.

Table 149 - Diagnostic page codes

Page code	Description	Subclause
00h	Supported diagnostic pages	8.3.1.1
40h	Translate address page	9.3.1.1
01h - 3Fh	Reserved (for all device type pages)	
41h - 7Fh	Reserved	
80h - FFh	Vendor-specific pages	

9.3.1.1 Translate address page - SEND DIAGNOSTIC

The translate address page allows the initiator to translate a logical block address, physical sector address or physical bytes from index address into any one of the other formats. The address to be translated is passed to the target with the SEND DIAGNOSTIC command and the results are returned to the initiator by the RECEIVE DIAGNOSTIC RESULTS command. The format of the translate address page - SEND DIAGNOSTIC is shown in table 150. The translated address is returned in the translate address page - RECEIVE DIAGNOSTIC RESULTS (see 9.3.1.2).

Table 150 - Translate address page - SEND DIAGNOSTIC

Bit Byte	7	6	5	4	3	2	1	0
0	Page code (40h)							
1	Reserved							
2	(MSB)							
3	Page length (000Ah)							
	(LSB)							
4	Reserved					Supplied format		
5	Reserved					Translate format		
6	Address to translate							
13								

The supplied format field specifies the format of address to translate field. Valid values for this field are defined in the FORMAT UNIT command (see 9.2.1). If the target does not support the requested format it shall terminate the SEND DIAGNOSTIC command with CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and an additional sense code shall be set to INVALID FIELD IN PARAMETER LIST.

The translate format field specifies which format the initiator would like the address to be translated to. Valid values for this field are defined in the FORMAT UNIT command (see 9.2.1). If the target does not support the requested format it shall terminate the command with CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and an additional sense code shall be set to INVALID FIELD IN PARAMETER LIST.

The address to translate field contains a single address the initiator is requesting the target to translate. The format of this field depends on the value in the supplied format field. The formats are described in 9.2.1.1. If the logical block format is specified the block address shall be in the first four bytes of the field with the remaining bytes set to zero.

9.3.1.2 Translate address page - RECEIVE DIAGNOSTIC

The translate address page allows the initiator to translate a logical block address, physical sector address, or physical bytes from index address into any one of the other formats. The address to be translated is passed to the target with the SEND DIAGNOSTIC command and the results are returned to the initiator by the RECEIVE DIAGNOSTIC RESULTS command. The translated address is returned in the translate address page - RECEIVE DIAGNOSTIC (see table 151).

Table 151 - Translate address page - RECEIVE DIAGNOSTIC

Bit Byte	7	6	5	4	3	2	1	0
0	Page code (40h)							
1	Reserved							
2	(MSB) Page length							
3	(LSB)							
4	Reserved					Supplied format		
5	RAREA	ALTSEC	ALTTRK	Reserved		Translated format		
Translated addresses								
6	Translated address 1							
13								
	:							
n	Translated address x (if required)							
n+7								

The translate address page contains a four byte page header which specifies the page code and length followed by two bytes which describe the translated address followed by zero or more translated address(s).

The page length field contains the number of parameter bytes which follow.

The supplied format field contains the value from the SEND DIAGNOSTIC command supplied format field (see 9.3.1.1).

A reserved area (RAREA) bit of one indicates that all or part of the translated address falls within a reserved area of the medium (e.g. speed tolerance gap, alternate sector, vendor reserved area, etc.). If the entire translated address falls within a reserved area the target may not return a translated address. An RAREA bit of zero indicates that no part of the translated address falls within a reserved area of the medium.

An alternate sector (ALTSEC) bit of one indicates that the translated address is physically located in an alternate sector of the medium. If the target cannot determine if all or part of the translated address is located in an alternate sector it shall set this bit to zero. An ALTSEC bit of zero indicates that no part of the translated address is located in an alternate sector of the medium or that the target is unable to determine this information.

An alternate track (ALTTRK) bit of one indicates that part or all of the translated address is located on an alternate track of the medium or the target cannot determine if all or part of the translated address is located on an alternate track. An ALTTRK bit of zero indicates that no part of the translated address is located on an alternate track of the medium.

The translated format field contains the value from the SEND DIAGNOSTIC command translate format field (see 9.3.1.1).

The translated address field contains the address(s) the target translated from the address supplied by the initiator in the SEND DIAGNOSTIC command. This field shall be in the format specified in the translate format field. The different formats are described in 9.2.1.1. If the logical block format is specified the block address shall be in the first four bytes of the field and the remaining bytes shall be set to zero.

If the returned data is in the logical block or physical sector format and the address to be translated covers more than one address after it has been translated (e.g. accounting for speed tolerance or multiple physical sectors within a single logical block or multiple logical blocks within a single physical sector) the target shall return all possible addresses which are contained in the area specified by the address to be translated.

If the returned data is in bytes from index format the target shall return a pair of translated values for each of the possible addresses which are contained in the area specified by the address to translate field. Of the pair of translated values returned, the first indicates the starting location and the second the ending location of the area.

9.3.2 Log parameters

This subclause defines the descriptors and pages for log parameters used with direct-access devices.

The log page codes for direct-access devices are defined in table 152.

Table 152 - Log page codes

Page code	Description	Subclause
01h	Buffer over-run/under-run page	8.3.2.1
03h	Error counter page (read) page	8.3.2.2
05h	Error counter page (verify) page	8.3.2.2
02h	Error counter page (write) page	8.3.2.2
07h	Last n error events page	8.3.2.3
06h	Non-medium error page	8.3.2.4
00h	Supported log pages	8.3.2.5
04h	Reserved	
08h - 2Fh	Reserved	
3Fh	Reserved	
30h - 3Eh	Vendor-specific pages	

9.3.3 Mode parameters

This subclause defines the descriptors and pages for mode parameters used with direct-access devices.

The mode parameter list, including the mode parameter header and mode block descriptor are described in 8.3.3.

The medium-type code field is contained in the mode parameter header (see 8.3.3). Table 153 defines this field for direct-access devices.

Table 153 - Direct-access medium-type codes

Code value	Medium type				
00h	Default medium type (currently mounted medium type)				
01h	Flexible disk, single-sided; unspecified medium				
02h	Flexible disk, double-sided; unspecified medium				
	Flexible disks				
	Diameter mm (in)	Bit density Bits/radian	Track density /mm (/in)	Number of sides	ANSI reference standard
05h	200 (8,0)	6 631	1,9 (48)	1	ANSI X3.73-1980
06h	200 (8,0)	6 631	1,9 (48)	2	(Note 1)
09h	200 (8,0)	13 262	1,9 (48)	1	None
0Ah	200 (8,0)	13 262	1,9 (48)	2	ANSI X3.121-1984
0Dh	130 (5,25)	3 979	1,9 (48)	1	ANSI X3.82-1980
12h	130 (5,25)	7 958	1,9 (48)	2	ANSI X3.125-1985
16h	130 (5,25)	7 958	3,8 (96)	2	ANSI X3.126-1986
1Ah	130 (5,25)	13 262	3,8 (96)	2	ISO DIS8630-1985
1Eh	90 (3,5)	7 958	5,3 (135)	2	ANSI X3.137-1988
	Direct-access magnetic tapes				
	Width mm (in)	Tracks	Density ftpm (ftpi)	ANSI reference standard	
40h	6,3 (0,25)	12	394 (10 000)	Note 1	
44h	6,3 (0,25)	24	394 (10 000)	Note 1	
Code values 80h - FFh are vendor-specific. All remaining code values are reserved.					
NOTE					
1 See annex c for additional standards information.					

The device specific parameter field (see table 154) is contained in the mode parameter header (see 8.3.3).

Table 154 - Device specific parameter

Bit	7	6	5	4	3	2	1	0
	WP	Reserved		DPOFUA	Reserved			

When used with the MODE SELECT command the write protect (WP) bit is not defined.

When used with the MODE SENSE command a WP bit of zero indicates that the medium is write enabled. A WP bit of one indicates that the medium is write protected.

When used with the MODE SELECT command, the DPOFUA bit is not used and the field is reserved.

When used with the MODE SENSE command, a DPOFUA bit of one indicates that the target supports the DPO and FUA bits (see 9.2.6). When used with the MODE SENSE command, a DPOFUA bit of zero indicates that the target does not support the DPO and FUA bits.

The density code field is contained in the mode parameter block descriptor (see 7.3.3). This field is reserved for direct-access devices.

The mode page codes for direct-access devices are shown in table 155.

Table 155 - Mode page codes

Page code	Description	Subclause
08h	Caching page	9.3.3.1
0Ah	Control mode page	8.3.3.1
02h	Disconnect-reconnect page	8.3.3.2
05h	Flexible disk page	9.3.3.2
03h	Format device page	9.3.3.3
0Bh	Medium types supported page	9.3.3.4
0Ch	Notch and partition page	9.3.3.5
09h	Peripheral device page	8.3.3.3
01h	Read-write error recovery page	9.3.3.6
04h	Rigid disk geometry page	9.3.3.7
07h	Verify error recovery page	9.3.3.8
06h	Reserved	
0Dh - 1Fh	Reserved	
00h	Vendor-specific (does not require page format)	
20h - 3Eh	Vendor-specific (page format required)	
3Fh	Return all pages (valid only for the MODE SENSE command)	

For direct-access devices, if the notch page is not supported, or if the active notch field in the notch page is zero then each page descriptor specifies mode parameters for the target to use for subsequent operations on the specified logical unit. If the notch page is supported, and the active notch is not zero, then each page descriptor specifies parameters for the target to use for subsequent operations on the disk notch specified by the current value of the active notch field on the specified logical unit.

9.3.3.1 Caching page

The caching parameters page (see table 156) defines the parameters that affect the use of the cache.

Table 156 - Caching page

Bit Byte	7	6	5	4	3	2	1	0
0	PS	Reserved	Page code (08h)					
1	Page length (0Ah)							
2	Reserved					WCE	MF	RCD
3	Demand read retention priority				Write retention priority			
4	(MSB)	Disable pre-fetch transfer length						(LSB)
5								
6	(MSB)	Minimum pre-fetch						(LSB)
7								
8	(MSB)	Maximum pre-fetch						(LSB)
9								
10	(MSB)	Maximum pre-fetch ceiling						(LSB)
11								

The parameters savable (PS) bit is only used with the MODE SENSE command. This bit is reserved with the MODE SELECT command. A PS bit of one indicates that the target is capable of saving the page in a non-volatile vendor-

specific location. If the PS is set to one in MODE SENSE data then the page shall be savable by issuing a MODE SELECT command with SP set to one.

A write cache enable (WCE) bit of zero specifies that the target shall return GOOD status for a WRITE command after successfully writing all of the data to the medium. A WCE bit of one specifies that the target may return GOOD status for a WRITE command after successfully receiving the data and prior to having successfully written it to the medium.

A multiplication factor (MF) bit of zero specifies that the target shall interpret the minimum and maximum pre-fetch fields in terms of the number of logical blocks for each of the respective types of pre-fetch. An MF bit of one specifies that the target shall interpret the minimum and maximum pre-fetch fields to be specified in terms of a scaler number which, when multiplied by the number of logical blocks to be transferred for the current command, yields the number of logical blocks for each of the respective types of pre-fetch.

A read cache disable (RCD) bit of zero specifies that the target may return data requested by a READ command by accessing either the cache or media. A RCD bit of one specifies that the target shall transfer all of the data requested by a READ command from the medium (i.e. data cannot be transferred from the cache).

The demand read retention priority field (see table 157) advises the target on the retention priority to assign data read into the cache that has also been transferred from the target to the initiator.

Table 157 - Demand read and write retention priority

Value	Description
0h	Indicates the target should not distinguish between retaining the indicated data and data placed into the cache memory by other means (e.g. pre-fetch).
1h	Demand read retention priority: Data put into the cache via a READ command should be replaced sooner (has lower priority) than data placed into the cache by other means (e.g. pre-fetch). Write retention priority: Data put into the cache during a WRITE or WRITE AND VERIFY command should be replaced sooner (has lower priority) than data placed into the cache by other means (e.g. pre-fetch).
Fh	Demand read retention priority: Data put into the cache via a READ command should not be replaced if there is other data in the cache that was placed into the cache by other means (e.g. pre-fetch) and it may be replaced (i.e. it is not locked). Write retention priority: Data put into the cache during a WRITE or WRITE AND VERIFY command should not be replaced if there is other data in the cache that was placed into the cache by other means (e.g. pre-fetch) and it may be replaced (i.e. it is not locked).
2h - Eh	Reserved

The write retention priority field (see table 157) advises the target on the retention priority to assign data written into the cache that has also been transferred from the cache memory to the medium.

An anticipatory pre-fetch occurs when data is placed in the cache that has not been requested. This usually happens in conjunction with the reading of data that has been requested. All the following parameters give an indication to the target how it should manage the cache based on the last READ command. An anticipatory pre-fetch may occur based on other information. All the remaining caching parameters are only recommendations to the target and should not cause a CHECK CONDITION to occur if the target cannot satisfy the request.

The disable pre-fetch transfer length field specifies the selective disabling of anticipatory pre-fetch on long transfer lengths. The value in this field is compared to the number of blocks requested by the current READ command. If the number of blocks is greater than the disable pre-fetch transfer length, then an anticipatory pre-fetch is not done for the command. Otherwise the target should attempt an anticipatory pre-fetch. If the pre-fetch disable transfer length is set to zero, then all anticipatory pre-fetching is disabled for any request for data, including those for zero logical blocks.

The minimum pre-fetch field indicates either a number of blocks or a scaler multiplier of the transfer length, depending upon the setting of the MF bit. In either case, the resulting number of blocks is the number to pre-fetch regardless of the delays it might cause in executing subsequent commands.

The pre-fetching operation begins at the logical block immediately after the last logical block of the previous READ command. Pre-fetching shall always halt before the end of the media. Errors that occur during the pre-fetching operation shall not be reported to the initiator unless that target cannot, as a result of the error, execute subsequent commands correctly. In this case the error may be reported either immediately as an error for the current READ command, or as a deferred error, at the discretion of the target and according to the rules for reporting deferred errors.

If pre-fetch has read more than the amount of data indicated by the minimum pre-fetch then pre-fetching should be terminated whenever another command is ready to execute. This consideration is ignored when the minimum pre-fetch is equal to the maximum pre-fetch.

The maximum pre-fetch field indicates either a number of blocks or a scaler multiplier of the transfer length, depending upon the setting of the MF bit. In either case, the resulting number of blocks is the number to pre-fetch if there are no delays in executing subsequent commands.

The maximum pre-fetch field contains the maximum amount of data to pre-fetch into the cache as a result of one READ command. It is used in conjunction with the disable pre-fetch transfer length and maximum pre-fetch ceiling parameters to trade off pre-fetching new data with displacing old data already stored in the cache.

The maximum pre-fetch ceiling field specifies an upper limit on the number of logical blocks computed as the maximum pre-fetch. If this number of blocks is greater than the maximum pre-fetch, then the number of logical blocks to pre-fetch shall be truncated to the value stored in the maximum pre-fetch ceiling field.

NOTE 124 If the MF bit is one the maximum pre-fetch ceiling field is useful in limiting the amount of data to be pre-fetched.

9.3.3.2 Flexible disk page

The flexible disk page (see table 158) contains parameters for control and reporting of flexible disk drive parameters.

Table 158 - Flexible disk page

Bit Byte	7	6	5	4	3	2	1	0	
0	PS	Reserved	Page code (05h)						
1	Page length in bytes (1Eh)								
2	(MSB)	Transfer rate						(LSB)	
3									
4	Number of heads								
5	Sectors per track								
6	(MSB)	Data bytes per sector						(LSB)	
7									
8	(MSB)	Number of cylinders						(LSB)	
9									
10	(MSB)	Starting cylinder-write precompensation						(LSB)	
11									
12	(MSB)	Starting cylinder-reduced write current						(LSB)	
13									
14	(MSB)	Drive step rate						(LSB)	
15									
16	Drive step pulse width								
17	(MSB)	Head settle delay						(LSB)	
18									
19	Motor on delay								
20	Motor off delay								
21	TRDY	SSN	MO	Reserved					
22	Reserved				SPC				
23	Write compensation								
24	Head load delay								
25	Head unload delay								
26	Pin 34				Pin 2				
27	Pin 4				Pin 1				
28	Medium rotation rate								(LSB)
29									
30	Reserved								
31	Reserved								

The parameters savable (PS) bit is only used with the MODE SENSE command. This bit is reserved with the MODE SELECT command. A PS bit of one indicates that the target is capable of saving the page in a non-volatile vendor-specific location.

NOTE 125 This page is mainly intended for defining parameters of flexible disk drives, but may be used for other devices, if applicable.

The transfer rate indicates the data rate of the peripheral device. See table 159 for examples of common transfer rates.

Table 159 - Examples of transfer rates

Value	Transfer rate
00FAh	250 kbit/s transfer rate
012Ch	300 kbit/s transfer rate
01F4h	500 kbit/s transfer rate
03E8h	1 mbit/s transfer rate
07D0h	2 mbit/s transfer rate
1388h	5 mbit/s transfer rate

The number of heads field specifies the number of heads used for reading and writing data on the medium. Heads used exclusively for servo information are excluded.

The sectors per track field specifies the number of sectors per revolution per head.

The data bytes per sector field specifies the number of bytes of data per sector that an initiator can read or write.

The number of cylinders field specifies the number of cylinders used for data storage.

The starting cylinder for write precompensation field specifies the cylinder at which write precompensation is to begin. Cylinders are numbered starting with zero. If the starting cylinder for write precompensation is equal to the value in the number of cylinders field, write precompensation shall be disabled by the target.

The starting cylinder for reduced write current field specifies cylinder at which write current is reduced. Cylinders are numbered starting with zero. If the starting cylinder for reduced write current is equal to the value in the number of cylinders field, reduced write current shall be disabled by the target.

The drive step rate field specifies the step rate in units of 100 μ s. This value may be rounded as defined in 7.5.4. A value of zero requests the target to set its default value.

The drive step pulse width field specifies the width of the step pulse in microseconds. This value may be rounded as defined in 7.5.4. A value of zero requests the target to set its default value.

The head settle delay field specifies the head settle time in units of 100 μ s. This value may be rounded as defined in 7.5.4. A value of zero requests the target to set its default value.

If a true ready signal is not available, the motor on delay field specifies in tenths of a second the time that the target shall wait before attempting to access the medium after the motor on signal is asserted. If a true ready signal is available, the motor on delay field specifies in tenths of a second the time that the target shall wait for drive ready status before aborting an attempt to access the medium. This value may be rounded as defined in 7.5.4.

The motor off delay field specifies in tenths of a second the time that the target shall wait before releasing the motor on signal after an idle condition exists. A value of FFh indicates that the motor on signal shall not be released. The START STOP UNIT command is not affected by this parameter. This value may be rounded as defined in 7.5.4.

A true ready (TRDY) bit of one specifies that a signal is provided that indicates the medium is ready to be accessed.

A start sector number (SSN) bit of one specifies that sectors are numbered starting with one. An SSN bit of zero specifies that sectors are numbered starting with zero.

A motor on (MO) bit of one specifies that pin 16 (motor on) shall remain released. This bit shall be set to one when using high capacity (192 tracks per inch) drives and their pre-formatted diskettes. An MO bit of zero indicates that pin 16 (motor on) shall be asserted.

The step pulse per cylinder (SPC) field is used to specify the number of additional step pulses required per cylinder. Non-zero values allow a drive to read a diskette formatted on a drive with a lower number of tracks per inch. For example, a value of one allows a 96 track-per-inch drive to access tracks on a diskette that was formatted for 48 tracks per inch.

The write compensation field is used to specify the amount of write compensation to be used starting at the cylinder specified in the starting cylinder for write precompensation field. The correlation of any values used in this field to actual write precompensation time values is vendor-specific. If a zero is specified in this field the target shall use its default write precompensation value. This value may be rounded as defined in 7.5.4.

The head load delay field specifies the head loading time in milliseconds. This value may be rounded as defined in 7.5.4. A value of zero requests the target to set its default value.

The head unload delay field specifies the head unloading time in milliseconds. This value may be rounded as defined in 7.5.4. A value of zero requests the target to set its default value.

The Pin 34 field defines the usage of pin 34 of the flexible disk drive interface. This use of this pin varies among vendors and drives. The settings allow the initiator to select how pin 34 shall be used by the interface. See table 160.

Table 160 - Pin 34 field

Bit 7 6 5 4	Description of pin 34 use
P 0 0 0	Open
P 0 0 1	Ready
P 0 1 0	Disk changed
NOTES 1 P is a polarity bit, where 0 is active low and 1 is active high. 2 All undefined values are reserved.	

The pin 4 field defines the usage of pin 4 of the flexible disk drive interface. This use of this pin varies among drive vendors and drives. The settings allow the initiator to specify how pin 4 shall be used by the interface. See table 161.

Table 161 - Pin 4 field

Bit 7 6 5 4	Description of pin 4 use
P 0 0 0	Open
P 0 0 1	In use
P 0 1 0	Eject
P 0 0 0	Head load
NOTES 1 P is a polarity bit, where 0 is active low and 1 is active high. 2 All undefined values are reserved.	

The pin 1 field defines the usage of pin 1 of the flexible disk drive interface. This use of this pin varies among vendors and drives. The settings allow the initiator to specify how pin 1 shall be used by the interface. See table 162.

Table 162 - Pin 34 field

Bit 7 6 5 4	Description of pin 34 use
P 0 0 0	Open
P 0 0 1	Disk change reset
NOTES	
1 P is a polarity bit, where 0 is active low and 1 is active high.	
2 All undefined values are reserved.	

The medium rotation rate field specifies the speed at which the medium rotates. The unit of measure is rotations per minute (e.g. 2 400 rpm). This field cannot be changed by a MODE SELECT command.

9.3.3.3 Format device page

The format device page (see table 163) contains parameters which specify the medium format.

Table 163 - Format device page

Bit Byte	7	6	5	4	3	2	1	0
0	PS	Reserved	Page code (03h)					
1	Page length (16h)							
2	(MSB)	Tracks per zone						_____
3								(LSB)
4	(MSB)	Alternate sectors per zone						_____
5								(LSB)
6	(MSB)	Alternate tracks per zone						_____
7								(LSB)
8	(MSB)	Alternate tracks per logical unit						_____
9								(LSB)
10	(MSB)	Sectors per track						_____
11								(LSB)
12	(MSB)	Data bytes per physical sector						_____
13								(LSB)
14	(MSB)	Interleave						_____
15								(LSB)
16	(MSB)	Track skew factor						_____
17								(LSB)
18	(MSB)	Cylinder skew factor						_____
19								(LSB)
20	SSEC	HSEC	RMB	SURF	Reserved			
21	Reserved							
23								

The parameters savable (PS) bit is only used with the MODE SENSE command. This bit is reserved with the MODE SELECT command. A PS bit of one indicates that the target is capable of saving the page in a non-volatile vendor-specific location.

NOTE 126 If the initiator changes any of the current physical parameters defined below, the target may not be able to access the media until a FORMAT UNIT command has been successfully completed.

If the defect handling format parameters (tracks per zone, alternate sectors per zone, alternate tracks per zone and alternate tracks per logical unit) requested by the initiator are not supported by the target the target may round these fields to acceptable values as described in 7.5.4.

The tracks per zone field specifies the number of tracks per zone to use in dividing the capacity of the device for the purpose of allocating alternate sectors. A value of zero means that one zone is defined for the entire device. The last zone on the device might not contain the same number of tracks as the previous zone(s).

The alternate sectors per zone field specifies the number of sectors per zone the target shall reserve for defect handling. The target shall de-allocate these sectors from the initiator addressable blocks during the FORMAT UNIT command. If the notch page is implemented and the ND bit of the notch page is one and the active notch field of the notch page is zero, then a value of zero indicates that no alternate sectors shall be reserved. Otherwise, a value of zero indicates that the number of alternate sectors is target specific.

The alternate tracks per zone field specifies the number of tracks per zone the target shall reserve for defect handling. The target shall de-allocate these tracks from the initiator addressable blocks during the FORMAT UNIT command. If the notch page is implemented and the ND bit of the notch page is one and the active notch field of the notch page is zero, then a value of zero indicates that no alternate tracks shall be reserved. Otherwise, a value of zero indicates that the number of alternate tracks is target specific.

The alternate tracks per logical unit field specifies the number of tracks per logical unit the target shall reserve for defect handling. The target shall de-allocate these tracks from the initiator addressable blocks during the FORMAT UNIT command. If the notch page is implemented and the ND bit of the notch page is one and the active notch field of the notch page is zero, then a value of zero indicates that no alternate tracks shall be reserved. Otherwise, a value of zero indicates that the number of alternate tracks is target specific.

The sectors per track field specifies the number of physical sectors included within each track. This number includes any alternate sectors the target may allocate. A value of zero during MODE SELECT indicates that the target shall define the number of sectors per track. For devices with a variable number of sectors per track, the value in MODE SELECT shall be zero and the value reported in MODE SENSE for the number of sectors per track is vendor-specific.

The data bytes per physical sector field specifies the number of data bytes per physical sector that the target shall use. This value may be different than the logical block size reported in the MODE SELECT data. The target shall return CHECK CONDITION status if it determines that the combination of this field and the sectors per track field exceed the capability of the medium. A value of zero indicates that the data bytes per physical sector is defined by the target.

For MODE SENSE the interleave field returns the same parameter passed in the FORMAT UNIT command. The target shall report this field as target defined in the corresponding MODE SENSE command. For MODE SELECT this field is ignored.

NOTE 127 It is recommended that this field be marked non-changeable and that initiators send the value returned in MODE SENSE. This allows migration to specifying interleave as a mode parameter instead of in the FORMAT UNIT command.

The track skew factor field specifies the number of physical sectors between the last logical block of one track and the first logical block on the next sequential track of the same cylinder.

The cylinder skew factor field specifies the number of physical sectors between the last logical block of one cylinder and the first logical block on the next sequential cylinder.

The SSEC bit set to one indicates that the target shall use soft sector formatting.

The HSEC bit set to one indicates that the target shall use hard sector formatting. The HSEC bit and the SSEC bit are mutually exclusive in MODE SELECT commands.

The combinations sector formatting supported that are reported in response to a request for default values are defined in table 164.

Table 164 - Reporting of default sector formatting support

SSEC	HSEC	Description
0	0	Target shall not return this combination
1	0	Target supports soft sector formatting only
0	1	Target supports hard sector formatting only
1	1	Target supports both soft and hard sector formatting

The combinations sector formatting supported that are reported in response to a request for changeable values are defined in table 165.

Table 165 - Reporting of changeable sector formatting support

SSEC	HSEC	Description
0	0	Sector formatting not changeable
1	0	Target shall not return this combination
0	1	Target shall not return this combination
1	1	Target supports both soft and hard sector formatting

The removable medium (RMB) bit set to one indicates that the logical unit supports removable media. A RMB bit set to zero indicates that the logical unit does not support removable media. The status of this bit shall be reflected in the INQUIRY command RMB bit.

The surface (SURF) bit set to zero indicates that the target shall allocate progressive addresses to all logical blocks within a cylinder prior to allocating addresses on the next cylinder. A SURF bit set to one indicates that the target shall allocate progressive addresses to all logical blocks on a surface prior to allocating sector addresses on the next surface.

NOTE 128 If the target supports savable parameters, all savable parameters for this initiator, including those in page codes 3, 4, and 5, are saved to non-volatile memory when the save parameters bit (SP) in the command descriptor block is set to one. The savable parameters may also be saved to non-volatile memory during a FORMAT UNIT command (see 9.2.1).

9.3.3.4 Medium types supported page

The medium types supported page (see table 166) contains a list of the medium types implemented by the target for logical units.

Table 166 - Medium types supported page

Bit Byte	7	6	5	4	3	2	1	0
0	PS	Reserved	Page code (0Bh)					
1	Page length (06h)							
2	Reserved							
3	Reserved							
4	Medium type one supported							
5	Medium type two supported							
6	Medium type three supported							
7	Medium type four supported							

The parameters savable (PS) bit is only used with the MODE SENSE command. This bit is reserved with the MODE SELECT command. A PS bit of one indicates that the target is capable of saving the page in a non-volatile vendor-specific location.

The code values for each medium type supported by the target (up to four maximum), are reported in ascending order. If only the default medium type is supported zero is reported. If less than four medium types are supported the unused entries shall be returned as zero.

9.3.3.5 Notch and partition page

The notch page (see table 167) contains parameters for direct-access devices which implement a variable number of blocks per cylinder and support this page. Each clause of the logical unit with a different number of blocks per cylinder is referred to as a notch.

Table 167 - Notch page

Bit Byte	7	6	5	4	3	2	1	0
0	PS	Reserved	Page code (0Ch)					
1	Page length (16h)							
2	ND	LPN	Reserved					
3	Reserved							
4	(MSB)	Maximum number of notches						_____
5								(LSB)
6	(MSB)	Active notch						_____
7								(LSB)
8	(MSB)	Starting boundary						_____
11								(LSB)
12	(MSB)	Ending boundary						_____
15								(LSB)
16	(MSB)	Pages notched						_____
23								(LSB)

The parameters savable (PS) bit is only used with the MODE SENSE command. This bit is reserved with the MODE SELECT command. A PS bit of one indicates that the target is capable of saving the page in a non-volatile vendor-specific location.

A notched drive (ND) bit of zero indicates that the device is not notched and that all other parameters in this page shall be returned as zero by the target. A ND bit of one indicates that the device is notched. For each supported active notch value this page defines the starting and ending boundaries of the notch.

A logical or physical notch (LPN) bit of zero indicates that the boundaries are based on the physical parameters of the logical unit. The cylinder is considered most significant, the head least significant. A LPN bit of one indicates that the notch boundaries are based on logical blocks of the logical unit.

The maximum number of notches field indicates the maximum number of notches supported by the logical unit. This field shall be reported as unchangeable.

The active notch field indicates the notch that this and subsequent MODE SELECT and MODE SENSE commands shall refer to, until the active notch is changed by a later MODE SELECT command. The value of the active notch shall be greater than or equal to 0 and less than or equal to the maximum number of notches. An active notch value of zero indicates that this and subsequent MODE SELECT and MODE SENSE commands refer to the parameters that apply across all notches.

The starting boundary field indicates the beginning of the active notch or, if the active notch is zero, the beginning boundary of the logical unit. If the LPN bit is one, then the four bytes represent a logical block address. If the LPN bit is zero, then the three most significant bytes shall represent the cylinder number and the least significant byte shall represent the head number. This field shall be reported as unchangeable. When used with the MODE SELECT command this field is ignored.

The ending boundary field indicates the ending of the active notch or, if the active notch is zero, the ending of the logical unit. If the LPN bit is one, then the four bytes represent logical block address. If the LPN bit is zero, then the three most significant bytes shall represent the cylinder number and the least significant byte shall represent the head number. This field shall be reported as unchangeable. When used with the MODE SELECT command this field is ignored.

Each notch shall span a set of consecutive logical blocks on the logical unit, the notches shall not overlap, and no logical block shall be excluded from a notch.

The pages notched field is a bit map of the mode page codes that indicates which pages contain parameters that may be different for different notches. The most significant bit of this field corresponds to page code 3Fh and the least significant bit corresponds to page code 00h. If a bit is one, then the corresponding mode page contains parameters that may be different for different notches. If a bit is zero, then the corresponding mode page contains parameters that are constant for all notches. This field shall be reported as unchangeable.

9.3.3.6 Read-write error recovery page

The read-write error recovery page (see table 168) specifies the error recovery parameters the target shall use during any command that performs a read or write operation to the medium (e.g. READ(6), READ(10), WRITE(6), WRITE(10), COPY, COMPARE, WRITE & VERIFY, etc.).

Table 168 - Read-write error recovery page

Bit Byte	7	6	5	4	3	2	1	0
0	PS	Reserved	Page code (01h)					
1	Page length (0Ah)							
2	AWRE	ARRE	TB	RC	EER	PER	DTE	DCR
3	Read retry count							
4	Correction span							
5	Head offset count							
6	Data strobe offset count							
7	Reserved							
8	Write retry count							
9	Reserved							
10	(MSB)	Recovery time limit						(LSB)
11								

The parameters savable (PS) bit is only used with the MODE SENSE command. This bit is reserved with the MODE SELECT command. A PS bit of one indicates that the target is capable of saving the page in a non-volatile vendor-specific location.

An automatic write reallocation enabled (AWRE) bit of one indicates that the target shall enable automatic reallocation to be performed during write operations. The automatic reallocation shall be performed only if the target has the valid data (e.g. original data in the buffer or recovered from the medium). The valid data shall be placed in the reallocated block. Error reporting as required by the error recovery bits (EER, PER, DTE, and DCR) shall be performed only after completion of the reallocation. The reallocation operation shall report any failures that occur. See the REASSIGN BLOCKS command (9.2.10) for error procedures.

An AWRE bit of zero indicates that the target shall not perform automatic reallocation of defective data blocks during write operations.

An automatic read reallocation enabled (ARRE) bit of one indicates that the target shall enable automatic reallocation of defective data blocks during read operations. All error recovery actions required by the error recovery bits (TB, EER, PER, DTE, and DCR) shall be executed. The automatic reallocation shall then be performed only if the target successfully recovers the data. The recovered data shall be placed in the reallocated block. Error reporting as required by the error recovery bits shall be performed only after completion of the reallocation. The reallocation process shall present any failures that occur. See the REASSIGN BLOCKS command (9.2.10) for error procedures.

An ARRE bit of zero indicates that the target shall not perform automatic reallocation of defective data blocks during read operations.

A transfer block (TB) bit of one indicates that a data block that is not recovered within the recovery limits specified shall be transferred to the initiator before CHECK CONDITION status is returned. A TB bit of zero indicates that such a data block shall not be transferred to the initiator. The TB bit does not affect the action taken for recovered data.

A read continuous (RC) bit of one indicates the target shall transfer the entire requested length of data without adding delays to perform error recovery procedures. This implies that the target may send data that is erroneous or fabricated in order to maintain a continuous flow of data. The target shall assign priority to this bit over conflicting error control bits (EER, DCR, DTE, and PER) within this byte.

NOTE 129 Fabricated data may be data already in the buffer or any other target-specific data. This bit is typically used in image processing, audio, or video applications.

An RC bit of zero indicates that error recovery operations that cause delays are acceptable during the data transfer. Data shall not be fabricated.

The individual bit definitions for EER, PER, DTE and DCR are contained in table 169. The combinations of these bits are explained in table 170.

Table 169 - Error recovery bit definitions

EER	PER	DTE	DCR	Description
1	-	-	-	An enable early recovery (EER) bit of one indicates that the target shall use of the most expedient form of error recovery first. This bit only applies to data error recovery and it does not affect positioning retries and the message system error recovery procedures.
0	-	-	-	An EER bit of zero indicates that the target shall use an error recovery procedure that minimizes the risk of mis-detection or mis-correction.
-	1	-	-	A post error (PER) bit of one indicates that the target shall report recovered errors.
-	0	-	-	A PER bit of zero indicates that the target shall not report recovered errors. Error recovery procedures shall be performed within the limits established by the error recovery parameters.
-	-	1	-	A disable transfer on error (DTE) bit of one indicates that the target shall terminate the data phase upon detection of a recovered error.
-	-	0	-	A DTE bit of zero indicates that the target shall not terminate the data phase upon detection of a recovered error.
-	-	-	1	A disable correction (DCR) bit of one indicates that error correction codes shall not be used for data error recovery.
-	-	-	0	A DCR bit of zero allows the use of error correction codes for data error recovery.

NOTE 130 An EER bit of one may imply an increase in the probability of mis-detection or mis-correction. An EER bit of zero typically means that the specified retry limit is exhausted prior to using error correction codes.

Table 170 - Combined error recovery parameter descriptions

EER	PER	DTE	DCR	Description
0	0	0	0	The full number of retries (specified in the read, write or verify retry count field) and error correction are attempted to recover the data (EER and DCR set to 0). A CHECK CONDITION is not reported at the completion of the command for recovered errors (PER set to 0). The command terminates with CHECK CONDITION status before the transfer count is exhausted only if an unrecoverable error is detected. If an unrecoverable data error occurred, the data in the block with the unrecoverable error may or may not be transferred to the initiator depending on the setting of the transfer block (TB) bit (read operation only).
0	0	0	1	Error correction is disabled (DCR set to one) so only the full number of retries (specified in the read, write or verify retry count field) are attempted to recover the data (EER set to 0). A CHECK CONDITION is not reported at the completion of the command for recoverable errors (PER set to 0). The command terminates with CHECK CONDITION status before the transfer count is exhausted only if an unrecoverable error is detected. If an unrecoverable data error occurred, the data in the block with the unrecoverable error may or may not be transferred to the initiator depending on the setting of the transfer block (TB) bit (read operation only).
0	0	1	0	Invalid mode (PER must be set to one if DTE is set to one). 1)
0	0	1	1	Invalid mode (PER must be set to one if DTE is set to one). 1)
0	1	0	0	The full number of retries (specified in the read, write or verify retry count field) and error correction are attempted to recover the data (EER and DCR set to 0). The command terminates with CHECK CONDITION status before the transfer count is exhausted only if an unrecoverable error is detected. If an unrecoverable data error occurred, the data in the block with the unrecoverable error may or may not be transferred to the initiator depending on the setting of the transfer block (TB) bit (read operation only). A CHECK CONDITION with a sense key of RECOVERED ERROR is reported at the completion of the command for any recoverable error that occurs (PER set to 1). The information field in the sense data shall contain the logical block address of the last recovered error which occurred during the transfer.

Table 170 - (continued)

EER	PER	DTE	DCR	Description
0	1	0	1	Error correction is disabled (DCR set to one) so only the full number of retries (specified in the read, write or verify retry count field) are attempted to recover the data (EER set to 0). The command terminates with CHECK CONDITION status before the transfer count is exhausted only if an unrecoverable error is detected. If an unrecoverable data error occurred, the data in the block with the unrecoverable error may or may not be transferred to the initiator depending on the setting of the transfer block (TB) bit (read operation only). A CHECK CONDITION with a sense key of RECOVERED ERROR is reported at the completion of the command for any recoverable error that occurs (PER set to 1). The information field in the sense data shall contain the logical block address of the last recovered error which occurred during the transfer.
0	1	1	0	The full number of retries (specified in the read, write or verify retry count field) and error correction are attempted to recover the data (EER and DCR set to 0). The command terminates with CHECK CONDITION status before the transfer count is exhausted if any error (recoverable or unrecoverable) is detected (DTE set to 1). The information field in the sense data shall contain the logical block address of the block in error. If an unrecoverable data error occurs the data in the block with the error may or may not be transferred to the initiator depending on the setting of the transfer block (TB) bit (read operation only).
0	1	1	1	Error correction is disabled (DCR set to one) so only the full number of retries (specified in the read, write or verify retry count field) are attempted to recover the data (EER set to 0). The command terminates with CHECK CONDITION status before the transfer count is exhausted if any error (recoverable or unrecoverable) is detected (DTE set to 1). The information field in the sense data shall contain the logical block address of the block in error. If an unrecoverable data error occurs the data in the block with the error may or may not be transferred to the initiator depending on the setting of the transfer block (TB) bit (read operation only).
1	0	0	0	The fewest possible retries and error correction are attempted to recover the data (EER set to one and DCR set to 0). A CHECK CONDITION is not reported at the completion of the command for recoverable errors (PER set to 0). The command terminates with CHECK CONDITION status before the transfer count is exhausted only if an unrecoverable error is detected. If an unrecoverable data error occurred, the data in the block with the unrecoverable error may or may not be transferred to the initiator depending on the setting of the transfer block (TB) bit (read operation only).

Table 170 - (concluded)

EER	PER	DTE	DCR	Description
1	0	0	1	Invalid mode (DCR must be set to zero if EER is set to one). 1)
1	0	1	0	Invalid mode (PER must be set to one if DTE is set to one). 1)
1	0	1	1	Invalid mode (PER must be set to one if DTE is set to one). 1)
1	1	0	0	The fewest possible retries and error correction are attempted to recover the data (EER set to one and DCR set to 0). The command terminates with CHECK CONDITION status before the transfer count is exhausted only if an unrecoverable error is detected. If an unrecoverable data error occurred, the data in the block with the unrecoverable error may or may not be transferred to the initiator depending on the setting of the transfer block (TB) bit (read operation only). A CHECK CONDITION with a sense key of RECOVERED ERROR is reported at the completion of the command for any recoverable error that occurs (PER set to 1). The information field in the sense data shall contain the logical block address of the last recovered error which occurred during the transfer.
1	1	0	1	Invalid mode (DCR must be set to zero if EER is set to one). 1)
1	1	1	0	The fewest possible retries and error correction are attempted to recover the data (EER set to one and DCR set to 0). The command terminates with CHECK CONDITION status before the transfer count is exhausted if any error (recoverable or unrecoverable) is detected (DTE set to 1). The information field in the sense data shall contain the logical block address of the block in error. If an unrecoverable data error occurs the data in the block with the error may or may not be transferred to the initiator depending on the setting of the transfer block (TB) bit (read operation only).
1	1	1	1	Invalid mode (DCR must be set to zero if EER is set to one). 1)
1)				If an invalid mode for the error recovery combination is sent by the initiator the target shall return CHECK CONDITION status with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN PARAMETER LIST.

The read and write retry count fields specify the number of times that the target shall attempt its recovery algorithm during read and write operations, respectively. If the retry count field and the recovery time limit field are both specified in a MODE SELECT command, the field that requires the least time for data error recovery actions shall have priority.

The correction span field specifies the size, in bits, of the largest data error burst for which data error correction may be attempted. A correction span of zero specifies that the target shall use its default value or that this field is not supported.

The head offset count field specifies in two's-complement notation an incremental offset position from the track center to which the heads shall be moved. The effect of this field on write operations is unspecified. A head offset count of zero indicates that no offset is specified. A positive value indicates moving in the direction of increasing logical block addresses. A negative value indicates moving in the direction of decreasing logical block addresses. Any value specified in this field does not preclude the target from using positive or negative head

offset during error recovery. However, after any error recovery is completed the target shall return the head offset to the value specified in this field.

NOTE 131 The degree of offset for each incremental value and the number of valid values are device specific. It is recommended that the number of valid values be equal for the positive and negative head offset counts.

The target shall return CHECK CONDITION status and shall set the sense key to ILLEGAL REQUEST if an unsupported head offset value is specified. The valid bit shall be set to one and the information field shall be set to the positive value of the maximum head offset count that is supported. The target shall set the valid bit to zero if it cannot determine the maximum head offset count supported.

NOTE 132 If the target does not support this field, it returns a zero value in the MODE SENSE command.)

The data strobe offset count field specifies in two's-complement notation an incremental position to which the recovered data strobe shall be adjusted from its nominal setting. The effect of this field on write operations is unspecified. A value of zero indicates that no data strobe offset is specified. A positive value indicates movement in a positive direction as defined by the target. A negative value indicates movement in the negative direction as defined by the target. Any value specified in this field does not preclude the target from using positive or negative data strobe offset during error recovery. However, after any error recovery is completed the target shall return the data strobe offset to the value specified in this field.

NOTE 133 The degree of offset for each incremental value and the number of valid values are device specific. It is recommended that the number of valid values be equal for the positive and negative data strobe offset counts.

The target shall return CHECK CONDITION status and shall set the sense key to ILLEGAL REQUEST if an unsupported data strobe offset count value is specified. The valid bit shall be set to one and the information field shall be set to the positive value of the maximum data strobe offset count that is supported. The target shall set the valid bit to zero if it cannot determine the maximum data strobe offset supported.

NOTE 134 If the target does not support the data strobe offset count field, it returns a zero value in the MODE SENSE command.

The recovery time limit field specifies in increments of one ms the maximum time duration that the target shall use for data error recovery procedures. The target may round this value as described in 7.5.4. The limits in this field specifies the maximum error recovery time allowed for any individual logical block. A recovery time limit of zero specifies that the target shall use its default value.

If both retry count and recovery time limit are specified, the field that specifies the recovery action of least duration shall have priority.

9.3.3.7 Rigid disk drive geometry page

The rigid disk drive geometry page (see table 171) specifies parameters for direct-access devices employing a rigid disk drive.

Table 171 - Rigid disk drive geometry page

Bit Byte	7	6	5	4	3	2	1	0
0	PS	Reserved	Page code (04h)					
1	Page length (16h)							
2	(MSB)	Number of cylinders						(LSB)
4								
5	Number of heads							
6	(MSB)	Starting cylinder-write precompensation						(LSB)
8								
9	(MSB)	Starting cylinder-reduced write current						(LSB)
11								
12	(MSB)	Drive step rate						(LSB)
13								
14	(MSB)	Landing zone cylinder						(LSB)
16								
17	Reserved					RPL		
18	Rotational offset							
19	Reserved							
20	(MSB)	Medium rotation rate						(LSB)
21								
22	Reserved							
23	Reserved							

The parameters savable (PS) bit is only used with the MODE SENSE command. This bit is reserved with the MODE SELECT command. A PS bit of one indicates that the target is capable of saving the page in a non-volatile vendor-specific location.

NOTE 135 This page is intended to define drive geometry parameters of rigid disk drives. It may be used for other devices if applicable.

The number of cylinders field defines the number of physical cylinders used for data storage.

The number of heads field defines the physical number of heads used for data storage. Heads used exclusively for servo information are excluded.

The starting cylinder for write precompensation field is the physical cylinder at which write precompensation is to begin. The first cylinder is number zero. If the starting cylinder for write precompensation is equal to the value in the number of cylinders field, write precompensation shall be disabled by the target.

The starting cylinder for reduced write current field is the physical cylinder at which write current is reduced. The first cylinder is number zero. If the starting cylinder for reduced write current is equal to the value in the number of cylinders field, reduced write current shall be disabled by the target.

The drive step rate field indicates the step rate in 100 ns increments. The target shall use the drive step rate, greater than or equal to the drive step rate specified. If the target rounds this field it shall terminate the command as described in 7.5.4. A value of zero requests the target to set its default value.

The landing zone cylinder field indicates two's complement location where the target shall position the disk heads. A negative value indicates that the heads are positioned below cylinder zero by that number of cylinders. A positive value greater than the number of cylinders indicates that the heads are positioned beyond the cylinders used for data storage at the cylinder location specified. A zero indicates that the default should be used.

The rotational position locking (RPL) field is used for spindle synchronization as defined in table 172. See K of the ANSI X3.170A-1991 for further information on synchronized spindles.

Table 172 - Rotational position locking

RPL	Description
00b	Indicates that spindle synchronization is disabled or not supported
01b	The target operates as a synchronized-spindle slave
10b	The target operates as a synchronized-spindle master
11b	The target operates as a synchronized-spindle master control

NOTE 136 The signals and connectors used for rotational position locking are external to the SCSI bus and are not part of this standard.

If a target fails to achieve synchronization it shall create a unit attention condition to all initiators. The sense key shall be set to UNIT ATTENTION and the additional sense code set to RPL STATUS CHANGE.

If subsequent to achieving synchronization the target detects a change of synchronization:

- a) and, if the logical unit is not presently executing an I/O process for the initiator, the target shall create a unit attention condition. The sense key shall be set to UNIT ATTENTION and the additional sense code set to RPL STATUS CHANGE.
- b) and, if the logical unit is presently executing an I/O process and no other error occurs, then the target shall return CHECK CONDITION status. The sense key shall be set to RECOVERED ERROR if the target is able to complete the I/O process or HARDWARE ERROR if the target is unable to complete the I/O process. The additional sense code is set to RPL STATUS CHANGE.

The rotational offset indicates the amount of rotational skew that the target shall use when synchronized. The rotational skew is applied in the retarded direction (lagging the synchronized spindle master control). The value in the field is the numerator of a fractional multiplier that has 256 as its denominator (e.g. a value of 128 indicates a one-half revolution skew). A value of zero indicates that rotational offset shall not be used. This value may be rounded as defined in 7.5.4. The rotational offset is not used when a target is configured as synchronized-spindle master.

The medium rotation rate indicates the speed at which the medium rotates. The unit of measure is rotations per minute (e.g. 3 600 rpm).

9.3.3.8 Verify error recovery page

The verify error recovery page (see table 173) specifies the error recovery parameters the target shall use during the VERIFY command, the verify operation of the WRITE AND VERIFY command and the verify operation of the COPY AND VERIFY command.

Table 173 - Verify error recovery page

Bit Byte	7	6	5	4	3	2	1	0
0	PS	Reserved	Page code (07h)					
1	Parameter length (0Ah)							
2	Reserved				EER	PER	DTE	DCR
3	Verify retry count							
4	Verify correction span							
5	Reserved							
6	Reserved							
7	Reserved							
8	Reserved							
9	Reserved							
10	(MSB)	Verify recovery time limit						
11								(LSB)

The parameters savable (PS) bit is only used with the MODE SENSE command. This bit is reserved with the MODE SELECT command. A PS bit of one indicates that the target is capable of saving the page in a non-volatile vendor-specific location.

The AWRE bit as defined in the read-write error recovery page (see 9.3.3.6) applies to the WRITE AND VERIFY command. The VERIFY and COMPARE commands shall not perform automatic reallocation. The COPY and COPY AND VERIFY commands might or might not perform automatic reallocation depending on the settings of the mode parameters.

The EER, PER, DTE and DCR bits are defined in 9.3.3.6. The combinations of these bits is defined in 9.3.3.6.

The verify retry count field specifies the number of times that the target shall attempt its recovery algorithm during a verify operation. If the verify retry count and the verify recovery time limit are both specified, the one that requires the least time for data error recovery actions shall have priority.

The verify correction span field specifies the size, in bits, of the largest burst data error for which data error correction may be attempted. If the target does not implement this field, a value of zero is returned in MODE SENSE data.

The verify recovery time limit field specifies in increments of one millisecond the maximum time duration that the target shall use error recovery procedures to recover data for an individual logical block. The target may round this value as described in 7.5.4. If the verify retry count and the verify recovery time limit are both specified, the one that requires the least time for data error recovery actions shall have priority.

NOTE 137 To disable all types of correction and retries the initiator should set the EER bit to zero, the PER, DTE and DCR bits to one and the number of retries and recovery time limit to zero.

9.4 Definitions specific to direct-access devices

9.4.1 cache memory: A temporary (and often volatile) data storage area outside the user-accessible area. A cache memory is usually faster to access than the medium and thus has the effect of increasing data throughput by reducing the number of accesses to the medium.

9.4.2 non-volatile medium: Physical storage medium that retains data written to it for a subsequent read operation through a power-on condition. An example of this is a disk device that stores data as magnetic fields that do not require device power to exist.

9.4.3 notch: A notch refers to all or part of the medium having a consistent set of geometry parameters. Notches are used to increase storage capacity by optimizing the number of sectors per track between the inner and outer tracks.

9.4.4 user-accessible: The area of the medium that can be read from or written to by READ and WRITE commands.

9.4.5 volatile medium: Medium that may not retain data written to it for a subsequent read operation through a power-on condition. An example of this is a silicon memory device that loses data written to it if device power is lost.