IBM System Storage Tape Drive 3592

SCSI Reference

Note:
Before using this information and the product it supports, be sure to read the general information under Notices.

Ninth Edition (22 April 2019)

This edition applies to the IBM TS1160 Tape Drive, IBM TS1155 Tape Drive, IBM TS1150 Tape Drive, IBM System Storage TS1140 Tape Drive, IBM TotalStorage Enterprise TS1130 Tape Drive, IBM System Storage TS1120 Tape Drive, IBM TotalStorage 3592 Tape System, and to all subsequent releases and modifications until otherwise indicated in new editions.

Order publications through your IBM representative or the IBM branch office serving your locality.

If you have comments or suggestions to improve this book see “Do You Have Comments or Suggestions?” on page d

When you send information to IBM, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

0. Summary of Changes

The first edition of this book is the update to:

a) the *IBM System Storage TS1120 Tape Drive 3592 SCSI Reference; and*
b) the *TotalStorage Enterprise Tape System 3592 SCSI Reference.*

The title changed from *TotalStorage Enterprise Tape System 3592 SCSI Reference* to *IBM System Storage TS1120 Tape Drive 3592 SCSI Reference* for rebranding purposes and has now changed to *IBM System Storage Tape Drive 3592 SCSI Reference* since this document is not intended to be a marketing publication.

0.1 Revision History

0.1.1 The first edition (TotalStorage Enterprise Tape System 3592 SCSI Reference)

The first edition of the *TotalStorage Enterprise Tape System 3592 SCSI Reference*, released for the initial GA, provided basic SCSI command interface communication information for the IBM TotalStorage Enterprise Tape Drive 3592.

0.1.2 The second edition (TotalStorage Enterprise Tape System 3592 SCSI Reference)

The second edition of the *TotalStorage Enterprise Tape System 3592 SCSI Reference* was updated to include support for WORM and ECONOMY cartridges.

0.1.3 The third edition (IBM System Storage TS1120 Tape Drive 3592 SCSI Reference)

The third edition of the *TotalStorage Enterprise Tape System 3592 SCSI Reference* was updated to include support for the *IBM TotalStorage Enterprise Tape Drive 3592 Model E05* and was renamed to *IBM System Storage TS1120 Tape Drive 3592 SCSI Reference* and was called the first edition.

0.1.4 The fourth edition (IBM System Storage Tape Drive 3592 SCSI Reference)

The fourth modification is called *IBM System Storage Tape Drive 3592 SCSI Reference* and includes support for the *IBM TotalStorage Enterprise Tape Drive 3592 Model E06* also known as the *IBM TotalStorage Enterprise TS1130 Tape Drive.*

This is called the first edition of the IBM System Storage Tape Drive 3592 SCSI Reference.

0.1.5 The fifth edition (IBM System Storage Tape Drive 3592 SCSI Reference)

The fifth modification is called the second edition of the IBM System Storage Tape Drive 3592 SCSI Reference.

This edition contains the following Functional Change Requests (FCR):

FCR 3163r3 - IP Address Information Configuration;
FCR 3187 - SPIN & SPOUT (OOBE-KMIP-SSC-4);
FCR 3189 - Extended Write-Read Diagnostics;
FCR 3190r2 - SDTF MAM parameters (LTFS);
FCR 3193 - End of partition behavior control;
FCR 3197 - Update standard inquiry version field;
FCR 3205 - Drive Type in Inquiry C0h;
FCR 3208 - Logical block protection;
FCR 3209r2 - TS1140 Partitioning;
FCR 3213r1 - TS1140 SCSI Identifiers;

FCR 3215 - TS1140 Report Supported OpCodes;

FCR 3217 - TS1140 Programmable Early Warning;

FCR 3218 - TS1140 Volume statistics log page (17h);

FCR 3223 - TS1140 4 TB operating point;

This revision contains additions and corrections related to the following defects (fnnnn defects are Functional Change Requests (FCR)):

f3176r2	Encryption Configuration mode page 30h[20h]
f3178r3	TS1140 SkipSync Enhancements
f3178r4	SkipSync Policy limitation
f3179r3	Append-only mode (data-safe)
f3205r1	Drive type in inquiry C0h
f3208	FCR-3208: Logical block protection (E2E CRC)
f3208r1	Logical block protection - break out VERIFY
f3209r3	TS1140 Partitioning refinements
f3215r1	TS1140 Report Supported Operation Code
f3226	Add support for Verify command to 3592 products (TS1120 - TS1140)
f3227	Make OIR bit saveable
f3228	Log Page 0Ch updated from standard
f3229	Deferred Check Condition (DCC) affinity
f3235	Manufacture assigned serial number VPD page (B1h)
f3236	Sequential-access Device Capabilities VPD page (B0h)
f3237	Read Block Limits MLOI
f3240	LP17 Remaining Native Capacity
31638	Minimum code level for LBP support needs listed
31562	3592 SCSI Ref: ASC/ASCQ EE31 missing from Annex B (Err Sns Info)
22192	BENCH: Documentation Missing Error Sns asc/ascq 05/4900

0.1.6 The sixth edition (IBM System Storage Tape Drive 3592 SCSI Reference)

The sixth modification, coincident with the IBM TS1150 Tape Drive, is called the third edition of the *IBM System Storage Tape Drive 3592 SCSI Reference*.

The list of defects and changes applied to the second edition of the *IBM System Storage Tape Drive 3592 SCSI Reference* to create this edition follow:

Make various typo and cross-reference corrections and general cleanup

c)
d)
e)

34874 - Change TS1150 SCSI Ref form number to GA32-0968-02

34880 - Add SET TIMESTAMP / REPORT TIMESTAMP to SCSI Ref

f3182r2 - Dynamic Runtime Information

f3203r1 - Recommended Access Order
f3203r2 - Recommended Access Order - Clean up
f3203r3 - Update max number UDS supported
f3203r4 - GRAO PARAMETER LIST LENGTH error incorrect
f3203r5 - RAO synch with the standards

f3206 - Units of measure for data storage

f3232 - TS1140 Max UKAD/AKAD lengths

f3237r1 - Maximum Logical Object Identifier and EW

f3247 - Partitioning type modifications

f3250 - LOCATE to EOD

f3251 - Encryption Sense Key changes

f3255 - Correct LOAD UNLOAD command
f3255r1 - Correct PAMR instructions

f3259 - XCopy

f3260 - Cache Attributes for READ ATTRIBUTES

f3261 - TS1150 SCSI Identifiers
f3261r1 - TS1150 SCSI Identifiers
f3261r2 - TS1150 SCSI Identifiers

f3262 - Partition WORM

f3262r1 - Partition WORM clarifications

f3263 - 3592 Power Cycle Considerations

f3264 - Terminate Immediate

f3265 - Clarify Command Timeouts in RSOC

f3271 - Download ucode additions
f3271r1 - Download ucode additions

f3272 - Add LP11h to 3592 SCSI Reference

f3275 - LBP add support for CRC32C

f3275r1- Correct typo in sample code to generate CRC32C

f3276 - TS1150 increase counter sizes

f3277 - 3592 Read Buffer mode and Buffer ID Cleanup

f3278 - TS1150 Add JL Media Type

f3279 - TS1150 Command Timeouts

f3281 - Add Primary Port (i.e., FC) Wrap test to 3592

0.1.7 The seventh edition (IBM System Storage Tape Drive 3592 SCSI Reference)

The seventh modification, is called the fourth edition of the *IBM System Storage Tape Drive 3592 SCSI Reference*.

The list of defects and changes applied to the third edition of the *IBM System Storage Tape Drive 3592 SCSI Reference* to create this edition follow:

Make various typo and cross-reference corrections and general cleanup

f3309 - Disable BOP Caching (see 4.3.2)

3302 - MP 1Ch Info Except Ctrl mode page

f3283r1 - Environmental Conditions Log Page

f3311 - Inquiry Allocation Length

f3314 - LTFS MAM parms 0820h & 0821h

f3310 - Non-standard mode page behavior

f3323 - TS1155 Command Timeout Values

f3333 - Update DRA to support standards

f3321 - LBP Support Inquiry B5h

f3324 - ADC Ethernet Config

f3326 - TS1155 SCSI Identifiers

0.1.8 The eighth edition (IBM System Storage Tape Drive 3592 SCSI Reference)

The eighth modification, is called the fifth edition of the *IBM System Storage Tape Drive 3592 SCSI Reference*.

The list of defects and changes applied to the fourth edition of the *IBM System Storage Tape Drive 3592 SCSI Reference* to create this edition follow:

Make various typo and cross-reference corrections and general cleanup
36498 - Update copyright

f3337 & f3337r1 - Archive Mode setting MP30h[43h]

0.1.9 The ninth edition (IBM System Storage Tape Drive 3592 SCSI Reference)

The ninth modification, is called the ninth edition of the *IBM System Storage Tape Drive 3592 SCSI Reference*.

The list of defects and changes applied to the sixth edition of the *IBM System Storage Tape Drive 3592 SCSI Reference* to create this edition follow:

Make various typo and cross-reference corrections and general cleanup

36696 - Begin TS1160 Work
34864 - DOC: uKAD support in tables is incorrect
36039 - READ REVERSE recommended timeout value for E08 is incorrect
35236 - TA 31h clear on LOAD/UNLOAD instead of removal
36700 - Final prep for TS1160 Publication
36713 - Missed port status updates
36733 - VPD page 87h incorrectly listed as supported and other errors

f3342, f3342r1, f3342r1_typo - TS1160 Base Changes
f3345 - TS1160 Command timeouts
f3346 - Report Optical Xcvr Info
f3347 - SCSI Reference Log Page corrections
f3349 - 3592 add MP10h SEW bit
f3350 - Autoload mode in Control Mode Page
f3351, f3351r1 - IP C1h Drive Serial Numbers (Brick SN)
f3188, f3188r1 - Engineering and Speed log pages updates

0.2 Do You Have Comments or Suggestions?

Your feedback is important in helping to provide the most accurate and high-quality information. If you have comments or suggestions for improving this publication, you can send us comments electronically by using these addresses:

Internet: starpubs@us.ibm.com

IBMLink$^{(TM)}$ from U.S.A.: STARPUBS at SJEVM5

IBMLink from Canada: STARPUBS at TORIBM

IBM Mail Exchange: USIB3VVD at IBMMAIL

You can also mail your comments by using the Reader Comment Form in the back of this manual or direct your mail to:
International Business Machines Corporation
Information Development
Department GZW
9000 South Rita Road
Tucson, Arizona 85744-0001 U.S.A.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Contents

Tables

Figures

1. Preface

1.1 Organization

The information in this book is presented as follows:

0. Summary of Changes, lists a summary of changes to this document including references to its predecessors which carried a different name.
1. Preface, describes the organization of this document and provides references to related documents.
2. Definitions, symbols, abbreviations, and conventions, describes words, terms, and conventions used in this book.
3. Introduction, describes the tape system.
4. Implementation Considerations, describes the SCSI implementation considerations.
5. SCSI Commands, describes the SCSI commands supported for the 3592 tape drive.
6. Parameters for SCSI Commands, describes the parameters transferred with SCSI commands

Annex A. Protocol Implementation Notes, describes the SCSI/Fibre Channel protocol implementation choices.
Annex B. Error Sense Information, provides all error sense information reported by the TotalStorage devices.
Annex C. Product Comparisons, summarizes product differences between various IBM tape drive offerings.
Annex E. Protection Information CRC's, provides sample code for the CRC's available for use in logical block protection.
Index Of Counters (log parameters not in counter format are not included) provides an index of log parameters that are in counter format.

1.2 Related Information

1.2.1 IBM 3592 Publications

For additional information about the 3592 subsystem or additional information about the IBM operating systems, go to http://www-947.ibm.com/support/entry/portal/support and search for the product.

1.2.2 IBM 3590 Publications

For additional information about the 3590 subsystem, refer to:

a) *IBM TotalStorage Tape System 3590 Hardware Reference*, GA32-0331
b) *IBM TotalStorage Tape System 3590 Operator Guide*, GA32-0330
c) *IBM TotalStorage Tape System 3590 Introduction and Planning Guide,* GA32-0329
d) *IBM TotalStorage Tape System 3590 Silo Compatible Frame Introduction, Planning, and User Guide*, GA32-0366
e) *IBM General Information Installation Manual-Physical Planning*, GC22-7072

1.2.3 IBM 3494 Tape Library Dataserver Publications

For additional information about the 3494 Tape Library Dataserver, refer to:

a) *IBM TotalStorage Enterprise Automated Tape Library (3494) Introduction and Planning Guide*, GA32-0448
b) *IBM TotalStorage Enterprise Automated Tape Library (3494) Operator's Guide*, GA32-0449
c) *IBM Magstar 3494 Tape Library Physical Planning Template*, GX35-5049
d) *IBM 3494 User's Guide: Media Library Device Driver for AS/400*, GC35-0153

1.2.4 Standards Publications

The following publications may be obtained from one or the other of the following contacts:

a) the ANSI Customer Service Department at:
Phone +1 212-642-4980
Fax: +1 212-302-1286
Web: http://www.ansi.org
E-mail: ansionline@ansi.org

or

b) the InterNational Committee for Information Technology Standards (INCITS) at:
Phone +1 202-626-5738
Web: http://www.incits.org
E-mail: incits@itic.org

a) ISO/IEC 14776-414, SCSI Architecture Model - 4 (SAM-4) [ANSI INCITS 447-2008]
b) ISO/IEC 14776-334, SCSI Stream Commands - 4 (SSC-4) [ANSI INCITS 516-2014]
c) ISO/IEC 14776-452, SCSI Primary Commands - 2 (SPC-2) [ANSI INCITS 351-2001]
d) ANSI INCITS 481-2012, Fibre Channel Protocol for SCSI - 4 (FCP-4)
e) ANSI INCITS 484-2012, SCSI Media Changer Commands - 3, (SMC-3)
f) ISO/IEC 14776-353, SCSI Media Changer Commands - 3 standard (SMC-3) [T10/1730-D]
g) ISO/IEC 14776-358, SCSI Automation/Drive Interface Commands - 3 standard (ADC-3) [T10/1895-D]

The following publications are in development at the time this document was published. They may be obtained from http://www.t10.org/drafts.htm

a) ISO/IEC 14776-415, SCSI Architecture Model - 5 standard (SAM-5) [T10/2104-D]
b) ISO/IEC 14776-454, SCSI Primary Commands - 4 standard (SPC-4) [T10/1731-D]
c) ISO/IEC 14776-335, SCSI Stream Commands - 5 (SSC-5) [T10/BSR INCITS 503]

1.3 Related Software Information

For information regarding software related to the IBM(R) TotalStorage Enterprise Tape Subsystems, refer to:

a) the z/OS Internet Library at http://www-03.ibm.com/systems/z/os/zos/library/bkserv/index.html
b) IBM Tape Device Drivers Installation and User’s Guide , GC27-2130, available only online at http://www-01.ibm.com/support/docview.wss?rs=577&uid=ssg1S7002972.
c) IBM Device Driver Programming Reference , GC27-0566, available only online at http://www-01.ibm.com/support/docview.wss?rs=577&uid=ssg1S7003032.

2. Definitions, symbols, abbreviations, and conventions

2.1 Definitions

This clause defines the special terms, abbreviations, and acronyms that are used in this publication. If the term being looked for is not found, refer to http://www-01.ibm.com/software/globalization/terminology/index.jsp.

2.1.1 ABEND Abnormal end of task.

2.1.2 access method A technique for moving data between processor storage and input/output devices.

2.1.3 adapter card A circuit card that adds function to a computer.

2.1.4 AES Advanced Encryption Standard.

2.1.5 AK Authentication Key

2.1.6 ALPA Arbitrated Loop Physical Address.

2.1.7 ANSI American National Standards Institute.

2.1.8 Arbitrated Loop A Fibre Channel Loop topology protocol, also known as L-port.

2.1.9 archiving The storage of backup files and associated journals, usually for a given period of time.

2.1.10 archiving application The retention of records, in machine-readable form, for historical purposes.

2.1.11 argument Any value of an independent variable.

2.1.12 ASC Additional Sense Code.

2.1.13 ASCQ Additional Sense Code Qualifier.

2.1.14 ASN.1 Abstract Syntax Notation One - OSI's encoding (see X.208 standard)

2.1.15 beginning of tape (BOT) The location on a magnetic tape that indicates the beginning of the permissible recording area. Synonymous with BOP 0.

2.1.16 BER Basic Encoding Rules - used with ASN.1 (see X.209 standard)

2.1.17 bezel The frame that fits over the front of the tape drive. This includes a button and a message display.

2.1.18 bit The smallest unit of data in a computer. A bit (short for binary digit) has a single binary value or either 0b or 1b.

2.1.19 block A collection of contiguous records recorded as a unit. Blocks are separated by interblock gaps, and each block may contain one or more records.

2.1.20 BOP Beginning of Partition - logical beginning of a data area (logical block 0)

2.1.21 BOT Beginning of tape.

2.1.22 bpi Bits per inch.

2.1.23 BPI Bytes per inch.

2.1.24 buffer A routine or storage used to compensate for a difference in rate of flow of data, or time of occurrence of events, when transferring data from one device to another.

2.1.25 buffered mode The buffered mode allows a number of logical blocks to accumulate in the control unit buffer before the data is transferred to the device or channel. This mode is suppressed automatically, if the record exceeds the maximum buffered capacity.

2.1.26 byte A storage unit that contains a certain number of bits (usually 8) which are treated as a unit an represent a character. A byte is a fundamental unit of data.

2.1.27 CA Contingent allegiance.

2.1.28 capacity See media capacity .

2.1.29 cartridge See tape cartridge .

2.1.30 cartridge loader A standard function for the tape drive that allows the manual loading of single tape cartridges.

2.1.31 cartridge memory An embedded non-contact electronics and interface module that can store and retrieve information relevant to the cartridge.

2.1.32 CC Check Condition.

2.1.33 CDB Command description block.

2.1.34 command A control signal that initiates an action or the beginning of a sequence of actions.

2.1.35 command timeout A host controlled period of time, following the issuance of a command, after which it is determined that a bad connection or severe failing condition exists between the host and drive.

2.1.36 compaction See data compression .

2.1.37 compression See data compression .

2.1.38 contingent allegiance (1) A condition in which a drive owes a response to a specific channel path because of a unit check. (2) A condition generated by a check condition status during which a target preserves sense data.

2.1.39 control unit A device attached between the tape drive and one or more host systems, often to provide special protocol or attachment features or other advanced function.

2.1.40 conversion The process of changing from one method of data processing to another or from one data-processing system to another.

2.1.41 copy operation The operation spawned by an XCOPY (i.e., EXTENDED COPY (LID4)) command to process the segment descriptors (see 4.28.4.2.1).

2.1.42 CU Control Unit - a tape controller subsystem, such as J70 (ESCON/FICON bridge)

2.1.43 DASD Direct-access storage device.

2.1.44 data Any representations such as characters or analog quantities to which meaning is, or might be, assigned.

2.1.45 data base A set of data, consisting of at least one file, that is sufficient for a given purpose or for a given data-processing system.

2.1.46 data compression An algorithmic data-reduction technique that encodes data from the host and stores it in less space than unencoded data. The original data is recovered by an inverse process called decompression.

2.1.47 data compression ratio The number of host data bytes divided by the number of encoded bytes. It is variable depending on the characteristics of the data being processed. The more random the data stream, the lower the opportunity to achieve compression.

2.1.48 data transfer rate The amount of data that can be stored on a tape cartridge with respect to time.

2.1.49 dataset The major unit of data storage and retrieval, consisting of a collection of data in one of several prescribed arrangements and described by control information to which the system has access.

2.1.50 DCC Deferred Check Condition, also known as deferred unit check .

2.1.51 DDR Device Data Recovery - a write perm recovery operation performed by a control unit

2.1.52 deferred unit check A condition in which a drive returns a unit check indication for an event that occurred asynchronously with the channel commands. The deferred unit check may not refer to the command that receives the indication.

2.1.53 DER Distinguished Encoding Rules - a subset of BER

2.1.54 device Any hardware component or peripheral that can receive and transmit data, such as a tape drive or tape library.

2.1.55 device driver An executable file or program installed on a host system used to control or access a device.

2.1.56 diagnostic A test or procedure designed to detect, recognize, locate, isolate or explain faults in equipment or errors in programs.

2.1.57 diagnostic cartridge A tape cartridge used to perform a diagnostic.

2.1.58 digest a cryptographically strong hash (i.e., SHA-x, MD-x)

2.1.59 DK Data Key - key used for encryption/decryption

2.1.60 DKI Data Key Index - a field in the EEDK(s)/SEDK and part of the DKi/IV recorded on media which associates the encryption of the record to EEDK(s) and ultimately a DK

2.1.61 DKi/IV Combined DKi and IV prepended to each record in the meta format

2.1.62 DKi=0 Data Key Index Zero - a special DKi (value of 0) which indicates that the data is encrypted with a well known method (key of all zeroes). This is used to write “non-encrypted” data in a format consistent with encryption for intermix cases

2.1.63 DKv Data Key Validator - a hash or signature which verifies a DK matches that used for data without requiring data decryption and validation via MAC. Could be an arbitrary clear text constant encrypted with the DK

2.1.64 drive A device used to store data to media and subsequently restore data from media.

2.1.65 drive head The component of a tape drive which converts and records an electrical signal to a magnetic signal on tape, and subsequently detects and converts such signals.

2.1.66 drive loaded A condition of a tape drive in which a tape cartridge has been inserted in the drive, and the tape has been threaded to the beginning-of-tape position.

2.1.67 dump To write the contents of storage, or of a part of storage, usually from an internal storage to a external medium, for a specific purpose such as to allow other use of storage, as a safeguard against faults or errors, or in connection with debugging.

2.1.68 EB See exabyte .

2.1.69 ECD Extended Copy Descriptor (see 6.9.2.1).

2.1.70 effective data rate The average number of bits, bytes, characters, or blocks per unit time transferred from a data source to a data sink and accepted as valid. The rate is expressed in bits, bytes, characters, or blocks per second, minute, or hour.

2.1.71 effective recording density The number of user bytes per unit of length of the recording medium.

2.1.72 EiB See exbibyte.

2.1.73 eject To remove or force from within. Generally refers to the last part of the unload process to allow removal of a tape cartridge from the drive.

2.1.74 EKM External Key Manager

2.1.75 enable To provide the means or opportunity. The modification of system, control unit, or device action through the change of a software module or a hardware switch (circuit jumper) position.

2.1.76 end of tape (EOT) The extreme position along the medium in the direction away from the take-up reel that is accessible by the device. This position may be different than an end-of-partition position.

2.1.77 EOD End Of Data - a dataset denoting the end of user data

2.1.78 EOP End of partition. The position at the end of the permissible recording region of a partition.

2.1.79 EOT End of tape.

2.1.80 ERA Error-recovery action performed by the host.

2.1.81 ERP See error-recovery procedures (ERP)

2.1.82 error-recovery procedures (ERP) (1) Procedures designed to help isolate and, where possible, to recover from errors in equipment. The procedures are often used in conjunction with programs that record the statistics of machine malfunctions. (2) Error-recovery procedures performed by the subsystem.

2.1.83 exabit 1 000 000 000 000 000 bits (i.e. 10^{18}). This is a decimal order of magnitude.

2.1.84 exabyte 1 000 000 000 000 000 bytes (i.e. 10^{18}). This is a decimal order of magnitude.

2.1.85 exbibyte 1 152 921 504 606 846 976 bytes (2^{60}). This is a binary order of magnitude.

2.1.86 explicitly activated A process in which the attributes of an identifier are specified. Contrast with implicitly activated .

2.1.87 extended contingent allegiance (1) A condition caused by a permanent buffered-write error in which the drive responds only to the channel path group from which the write command was received. The extended contingent allegiance continues until a controlling computer in the channel path group retrieves the unwritten data from the buffer or issues a tape motion command. (2) A condition generated by an initiate recovery message to assist in extended error recovery procedures in multi-initiator systems.

2.1.88 F-port Fabric port.

2.1.89 FC Fibre Channel .

2.1.90 FCP Fibre Channel Protocol - the SCSI mapping to fibre channel

2.1.91 fiber A physical communications cable or connection used to attach two or more Fibre Channel devices.

2.1.92 Fibre Channel A standard interconnection interface used to attach host systems and/or peripheral devices.

2.1.93 FID Format Identification Dataset.

2.1.94 field replaceable unit (FRU) An assembly that is replaced in its entirety when any one of its components fails.

2.1.95 file A set of related records, treated as a unit; for example, in stock control, a file could consist of a set of invoices.

2.1.96 file protected Pertaining to a tape volume from which data can be read only. Data cannot be written on or erased from the tape.

2.1.97 filemark A logical demarcation, recorded on media, often used to separate files or provide other organizational structure to recorded data. Usage and convention of filemarks is controlled by the attached host system(s).

2.1.98 FIPS Federal Information Processing Standards

2.1.99 firmware Proprietary code that is usually delivered as part of an operating system or device. Firmware is more efficient than software loaded from an alterable medium, and is more adaptable to change than hardwired embedded logic.

2.1.100 FL-port Fabric loop port.

2.1.101 FMR Field microcode replacement.

2.1.102 format The arrangement or layout of data on a data medium.

2.1.103 FRU Field replaceable unit (see 2.1.94).

2.1.104 GB See gigabyte .

2.1.105 Gb See gigabit .

2.1.106 GCM Galois Counter Mode

2.1.107 GiB See gibibyte.

2.1.108 gibibyte 1 073 741 824 bytes (2^{30}). This is a binary order of magnitude.

2.1.109 gigabit 1 000 000 000 bits (i.e. 10^9). This is a decimal order of magnitude.

2.1.110 gigabyte 1 000 000 000 bytes (i.e. 10^9). This is a decimal order of magnitude.

2.1.111 hard addressing A method of specifying a fixed address for a device in a Fibre Channel loop configuration.

2.1.112 hardware The physical equipment or components that form a device or system.

2.1.113 HBA host bus adapter .

2.1.114 head See drive head

2.1.115 host bus adapter A specific type of adapter card which provides the connection to a physical device interconnect such as Fibre Channel.

2.1.116 host system A data-processing system that is used to prepare programs and the operating environments for use on another computer or controller.

2.1.117 IBM Proprietary Protocol IBM vendor-specific method of configuring and controlling encryption

2.1.118 implicitly activated A process in which the attributes of an identifier are determined by default. Contrast with explicitly activated .

2.1.119 initiator A SCSI device that requests an I/O process to be performed by another SCSI device (a target). In many cases, an initiator can also be a target.

2.1.120 install To set up for use or service. The act of adding a product, feature, or function to a system or device either by a singular change or by the addition of multiple components or devices.

2.1.121 interchange application The preparation of tapes for use on other systems or devices, either local or remote, or the use of tape data prepared by another system.

2.1.122 invoke To petition for help or support. The request for a feature or function to be utilized in future processing activities through the use of software or hardware commands.

2.1.123 IPP IBM Proprietary Protocol

2.1.124 IV Initialization Vector - a value also called a nonce, used with a key for AES block ciphers

2.1.125 journaling Recording transactions against a dataset so that the dataset can be reconstructed by applying transactions in the journal against a previous version of the dataset.

2.1.126 KB See kilobyte .

2.1.127 Kb See kilobit .

2.1.128 KiB See kibibyte.

2.1.129 kibibyte 1 024 bytes (2^{10}). This is a binary order of magnitude

2.1.130 kilobit 1000 bits (10^3). This is a decimal order of magnitude.

2.1.131 kilobyte 1000 bytes (10^3). This is a decimal order of magnitude.

2.1.132 L-port Arbitrated Loop Fibre Channel host connection. May attach to a fabric (switch) FL-port.

2.1.133 LDI Library Drive Interface - a specific interface protocol for tape device to automation interface (over RS422)

2.1.134 LEOT logical end of tape

2.1.135 LN-port Fibre Channel host attachment which attempts to negotiate first to Arbitrated Loop, then Point-to-Point. May attach to a fabric (switch) FNL-port.

2.1.136 load Following the insertion of a tape cartridge into the device, the act of positioning the tape (performed by the drive) for subsequent reading or writing.

2.1.137 load point The beginning of the recording area on magnetic tape.

2.1.138 loader See cartridge loader

2.1.139 logical end of tape A point on the tape where written data normally ends.

2.1.140 LPOS Longitudinal Position.

2.1.141 LSB Least significant byte.

2.1.142 lsb Least significant bit.

2.1.143 LTO Linear Tape Open.

2.1.144 LUN Logical unit number.

2.1.145 MAC Message Authentication Code - a digest which validates encrypted data. Appended to the record in the meta format for encryption integrity validation

2.1.146 magnetic recording A technique of storing data by selectively magnetizing portions of a magnetizable material.

2.1.147 magnetic tape A tape with a magnetizable surface layer on which data can be stored by magnetic recording.

2.1.148 magnetic tape drive A mechanism for moving magnetic tape and controlling its movement.

2.1.149 manual mode A mode of operation that can be selected on a cartridge loader or library. This mode allows a single tape cartridge feed, performed by the operator.

2.1.150 MB See megabyte .

2.1.151 Mb See megabit .

2.1.152 mebibyte 1 048 576 bytes (2^{20}). This is a binary order of magnitude.

2.1.153 media Plural of medium.

2.1.154 media capacity The amount of data that can be contained on storage media and expressed in bytes of data.

2.1.155 medium A physical material in or on which information may be represented, such as magnetic tape.

2.1.156 megabit 1 000 000 bits (10^6). This is a decimal order of magnitude.

2.1.157 megabyte 1 000 000 bytes (10^6). This is a decimal order of magnitude.

2.1.158 MiB See mebibyte.

2.1.159 microcode Embedded device programming which controls the behavior and functioning of the device.

2.1.160 microprocessor An integrated circuit that accepts coded instructions for execution; the instructions may be entered, integrated, or stored internally.

2.1.161 migration See conversion .

2.1.162 MIM Medium Information Message.

2.1.163 msb Most significant bit.

2.1.164 MSB Most significant byte.

2.1.165 N-port Point-to-Point Fibre Channel host connection. May attach to a fabric (switch) FL-port.

2.1.166 N/A Not Applicable.

2.1.167 native data transfer rate The amount of data that can be stored without compression on a tape cartridge with respect to time.

2.1.168 native storage capacity The amount of data that can be stored without compression on a tape cartridge.

2.1.169 NL-port Fibre Channel host attachment which attempts to negotiate first to Point-to-Point, then Arbitrated Loop. May attach to a fabric (switch) FL-port.

2.1.170 node Fibre channel term for the logical connection to a device.

2.1.171 nonce number used once - a value used in conjunction with the key for AES block ciphers (also IV)

2.1.172 OEM Original equipment manufacturer.

2.1.173 offline An operating condition where the host system cannot interact with the drive through the specified interface.

2.1.174 online An operating condition where the host system can interact normally with the drive through the specified interface.

2.1.175 OOB Out-Of-Band

2.1.176 open system Computer systems whose operating standards and methods are not proprietary.

2.1.177 operating system The master computer control program that translates the user commands and allows software application programs to interact with the computer hardware and attached devices.

2.1.178 overwrite A write operation that records a logical object in a logical position that is not an append point (see 4.13.3).

2.1.179 OSI Open Systems Interconnection - (see X.200 standard)

2.1.180 PB See petabyte .

2.1.181 pebibyte 1 125 899 906 842 624 bytes (2^{50}). This is a binary order of magnitude.

2.1.182 PEOT physical end of tape

2.1.183 petabit 1 000 000 000 000 bits (i.e. 10^{15}). This is a decimal order of magnitude.

2.1.184 petabyte 1 000 000 000 000 bytes (i.e. 10^{15}). This is a decimal order of magnitude.

2.1.185 physical end of tape A point on the tape beyond which the tape is not permitted to move.

2.1.186 PiB See pebibyte.

2.1.187 PKCS Public-Key Cryptography Standards

2.1.188 POR Power-on reset.

2.1.189 port Fibre channel term for the physical connection to a device.

2.1.190 primed Pertaining to a condition of a tape drive when the controlling computer addresses the drive but the drive is not in a ready state.

2.1.191 PRNG Pseudo Random Number Generator

2.1.192 processing application The execution of a systematic sequence of operations performed on data to accomplish a specific purpose.

2.1.193 Program Temporary Fix A temporary solution to a problem in the microcode.

2.1.194 PTF Program Temporary Fix

2.1.195 quiesce To bring a device or system to a halt by a rejection of new requests for work.

2.1.196 read To acquire or interpret data from a storage device, from a data medium, or from another source.

2.1.197 read-type commands Any commands that cause data to be read from tape.

2.1.198 record A collection of related data or words, treated as a unit; for example, in stock control, each invoice could constitute one record.

2.1.199 recording density The number of bits in a single linear track measured per unit of length of the recording medium.

2.1.200 reset To return a device, circuit, or value to a clear state.

2.1.201 retension (or refresh) The process or function of tightening the tape onto the cartridge, if it is sensed that the tape has a loose wrap on the cartridge.

2.1.202 RSA Method authored by Rivest, Shamir, Adleman

2.1.203 SAN Storage Area Network.

2.1.204 SCSI Small Computer System Interface.

2.1.205 SCSI address The octal representation of the unique address (0-F) assigned to a SCSI device. This address would normally be assigned and set in the SCSI device during system installation.

2.1.206 SCSI device A host adapter or a target controller that can be attached to the SCSI bus.

2.1.207 SCSI ID The identifier used to uniquely identify the address on the bus. When used on Fibre Channel devices this refers to the AL_PA.

2.1.208 SHA Secure Hash Algorithm (can be SHA-256, SHA-384, SHA-512, for bit size)

2.1.209 SIM Service Information Message.

2.1.210 soft addressing A method of specifying a standard arbitration method for assigning an address for a device in a Fibre Channel loop configuration.

2.1.211 special feature A specific design addition to an IBM product that is quoted in the IBM Sales Manual and ordered separately.

2.1.212 standard function The significant design elements of an IBM product that are included as part of the basic standard product.

2.1.213 synchronization The process of coordinating the activities of the controlling computer and the magnetic tape subsystem to obtain the condition in which the buffer is empty and the tape is in the correct position for the next operation.

2.1.214 T10 ANSI group responsible for SCSI model and command sets, see http://www.t10.org

2.1.215 T11 ANSI group responsible for FCP/fibre channel protocols, see http://www.t11.org

2.1.216 tape Commonly refers to magnetic tape or the tape cartridge.

2.1.217 tape cartridge A container holding magnetic tape that can be processed without separating it from the container.

2.1.218 tape drive A device that is used for moving magnetic tape and includes the mechanisms for writing and reading data to and from the tape.

2.1.219 tape unit A device that contains tape drives and their associated power supplies and electronics.

2.1.220 TapeAlert A patented technology and ANSI standard that defines conditions and problems that are experienced by tape drives.

2.1.221 TapeAlert flags Status and error messages that are generated by the TapeAlert utility and are reported to a host system.

2.1.222 target A SCSI device that performs an operation requested by the initiator.

2.1.223 target routine A target routine is an I/O process directed to a target, and not to a logical unit.

2.1.224 TB See terabyte.

2.1.225 tebibyte 1 099 511 627 776 bytes (2^{40}). This is a binary order of magnitude.

2.1.226 terabyte 1 000 000 000 000 bytes (10^{12}). This is a decimal order of magnitude.

2.1.227 TiB See tebibyte.

2.1.228 TRNG True Random Number Generator

2.1.229 TSM Tivoli Storage Manager

2.1.230 vital product data Non-volatile information including configuration, calibration, etc., used to control the behavior and operation of the device.

2.1.231 volume (1) A certain portion of data, together with its data carrier, that can be handled conveniently as a unit. (2) A data carrier that is mounted and demounted as a unit, for example, a reel of magnetic tape, a disk pack.

2.1.232 volume coherency set: A set of information contained in logical objects including a volume coherency count (see 4.24) for which coherency across an entire volume is desired.

2.1.233 VPD Vital Product Data - information stored in drive nonvolatile memory

2.1.234 WORM (Write Once, Read Many) A write or append methodology for allowing data to be written only once, disallowing overwriting.

2.1.235 write To store or encode data to a storage device, to data medium, or to another source.

2.1.236 Write Once, Read Many (WORM) A write or append methodology for allowing data to be written only once, disallowing overwriting.

2.1.237 write protected A state disallowing write operations to a device or medium.

2.1.238 write-type commands Any commands that cause data to be written on tape or affect buffered write data.

2.2 Conventions

2.2.1 Bit Numbering

Bit numbering follows ANSI standards as follows:

- Bit 7 is the most significant bit (msb) occupying the leftmost bit position in the diagrams
- Bits 6 through 1 continue from left to right in descending order
- Bit 0 is the least significant bit (lsb) occupying the rightmost bit position in the diagrams

2.2.2 Units of measure

Decimal units such as KB, MB, GB, and TB have commonly been used to express data storage values, though these values are more accurately expressed using binary units such as KiB, MiB, GiB, and TiB. At the kilobyte level, the difference between decimal and binary units of measurement is relatively small (2.4%). This difference grows as data storage values increase, and when values reach terabyte levels the difference between decimal and binary units approaches 10%.

This document represents values using both decimal units and binary units. Values are represented by the following formats:

a) for decimal units:
   1) numeric value;
   2) space;
   3) decimal prefix or decimal symbol (see table 3); and
   4) unit (e.g., byte or B);
   5) (;
   6) decimal power (e.g., 10^6);
   7))
b) for binary units:
   1) numeric value;
   2) space;
   3) binary prefix or binary symbol (see table 3); and
   4) unit (e.g., byte or B);
   5) (;
   6) binary power (e.g., 2^{20});
   7));
c) for an indication that all values in a row of a table are decimal units:

    1) (;
    2) decimal power (e.g., 10^6);
    3))
- d) for an indication that all values in a row of a table are binary units:
    1) (;
    2) binary power (e.g., 2^{20});
    3));

As an example:

- a) for decimal units:
  the value 3.5 terabytes is displayed as 3.5 TB (10^{12});
- b) for binary units:
  the value 400 mebibytes per second is displayed as 400 MiB/sec (2^{20})
- c) for an indication that all values in a row of a table are decimal units:

Table 1 — Density information

FIELD	PRIMARY DENSITY CODE			
	51h ^{1}	**52h ^{2}**	**53h ^{3}**	**54h ^{4}**
⋮				
CAPACITY IN MB (10^6)	0004_93E0h[a]	000A_AE60h[a]	000F_4240h[a]	0035_67E0h[a]
⋮				

[A] VALUE WHEN MEDIA = 0B

- d) for an indication that all values in a row of a table are binary units:

Table 2 — Differences between products

Item	Device model			
	J1A	**E05**	**E06**	**E07**
⋮				
Read/Write buffer size[a]	128	512	1024	1024
⋮				

[A] VALUES ARE IN MIB (2^{20}).

Table 3 compares the names, symbols, and values of the binary and decimal units. Table 4 shows the increasing percentage of difference between binary units and decimal units.

Table 3 — Comparison of binary and decimal units and values

Decimal			Binary		
Name	**Symbol**	**Value (base-10)**	**Name**	**Symbol**	**Value (base-2)**
kilo	K	10^3	kibi	Ki	2^{10}
mega	M	10^6	mebi	Mi	2^{20}
giga	G	10^9	gibi	Gi	2^{30}
tera	T	10^{12}	tebi	Ti	2^{40}
peta	P	10^{15}	pebi	Pi	2^{50}
exa	E	10^{18}	exbi	Ei	2^{60}
zetta	Z	10^{21}	zebi	Zi	2^{70}
yotta	Y	10^{24}	yobi	Yi	2^{80}

Table 4 — Percentage difference between binary and decimal units

Decimal Value	Binary Value	Percentage Difference
100 kilobytes (KB)	97.65 kibibytes (KiB)	2.35%
100 megabytes (MB)	95.36 mebibytes (MiB)	4.64%
100 gigabytes (GB)	93.13 gibibytes (GiB)	6.87%
100 terabytes (TB)	90.94 tebibytes (TiB)	9.06%
100 petabytes (PB)	88.81 pebibytes (PiB)	11.19%
100 exabytes (EB)	86.73 exbibytes (EiB)	13.27%
100 zettabytes (ZB)	84.70 zebibytes (ZiB)	15.30%
100 yottabytes (YB)	82.72 yobibytes (YiB)	17.28%

2.2.3 Subpages

When pages have subpages (e.g., Mode Pages, Log Pages) the convention used for Page XXh Subpage YYh is Page XXh[YYh].

When describing Security Protocol XXh with Security Protocol Specific YYYYh in the Security Protocol In command or the Security Protocol Out command XXh[YYYYh] is used.

2.2.4 Text Markers

The source files of this document contains conditional text to aid in maintaining the various versions that are generated from these files. During editing and review, these conditional texts have specific markers that delineate which condition they relate to. These markers include color, underlines, overlines, etc. Those markers contained in this version of the document are:

a) Text contained in all versions of the document (Unconditional)
b) Text removed by a Future feature
c) Text contained in the external version that is replaced by text in the encryption version of the document (Crypto-Removed)
d) Text in the external version that is replaced by text in the Internal version of the document (Internal-Removed)

2.2.5 Hyperlinks

This document contains many hyperlinks. Every place the text says “see clause number” should be a hyperlink. Hyperlinks have been given a special font to offset them from the rest of the text. That font is demonstrated in this following link (see 2.2.5)

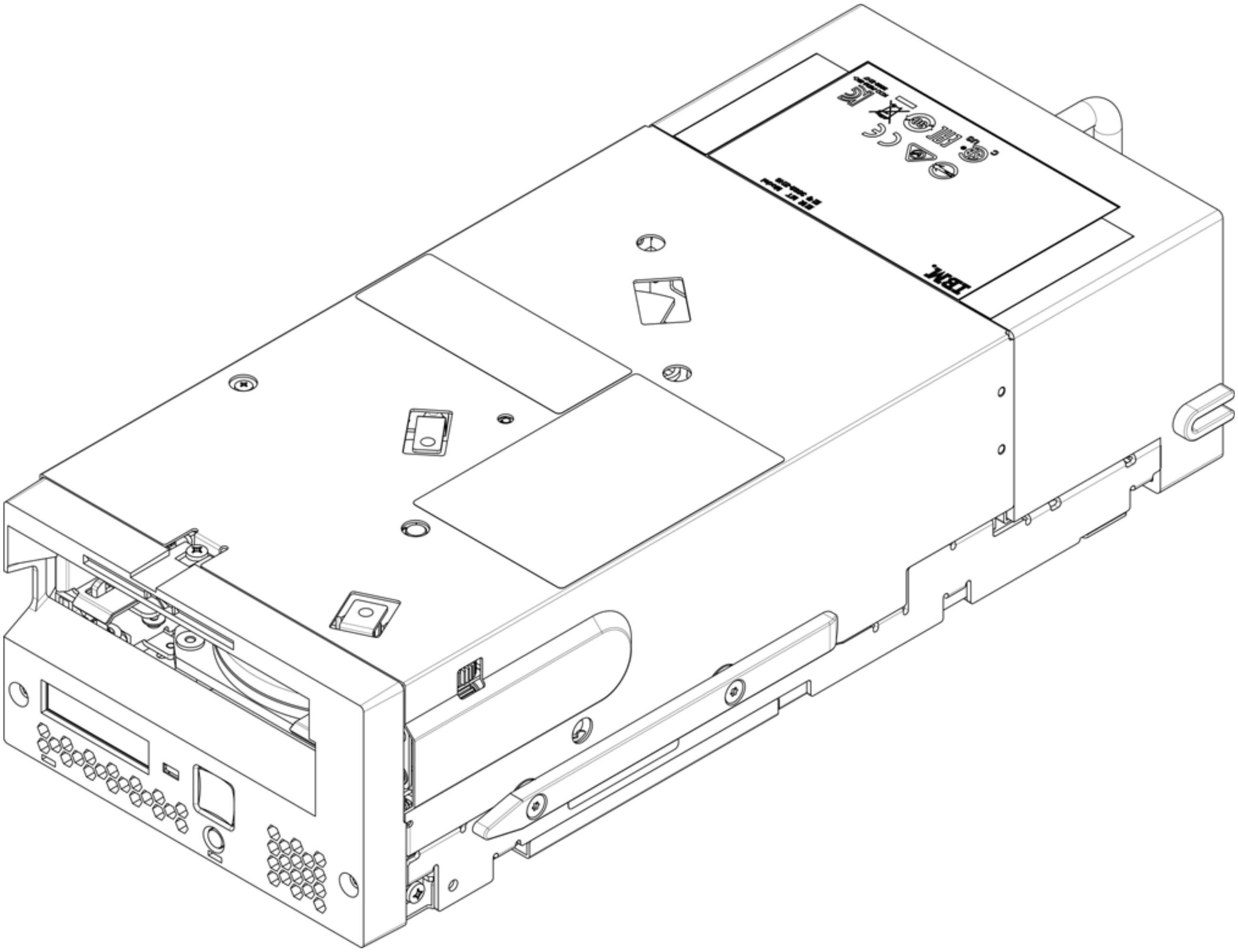
2.3 Tape Drive Model Names

From this section forward, through the remainder of this book, all Tape Drive models are referred to collectively as the IBM Enterprise Tape Drive 3592 Tape Drive or simply as the 3592. When specific 3592 model(s) are being discussed they may be referred to as 3592 xxx or simply xxx where xxx is the model designation. For example, the IBM TotalStorage Enterprise Tape System Model J1A is referred to as the 3592 J1A, the IBM System Storage TS1120 Tape Drive (3592 Model E05) is referred to as the 3592 E05 or simply E05, and so on.

3. Introduction

The IBM 3592 Enterprise Tape Drive provides new levels of function, performance, reliability, and cartridge capacity. Figure 1 shows a drawing of the IBM 3592 Tape Drive.

Figure 1 — IBM 3592 Tape Drive



3.1 Highlights

3.1.1 Comparison between products

The IBM 3592 Tape Drive is currently on its ninth generation. The various products are referred to throughout this document and other documentation in a number of ways which are shown in table 5

Table 5 — IBM System Storage Tape System 3592 products

Product	Shorthand	PRODUCT IDENTIFICATION
3592 J1A	J1A	03592J1A
TS1120	E05	03592E05
TS1130	E06	03592E06
TS1140	E07	03592E07
TS1150	E08	03592E08
TS1155	55F	0359255F
TS1160	60F	0359260F

Due to the media reuse supported by the 3592 products (see 3.1.6 on page 19), there is potential for confusion when referring to the 'Format' of the various capacities and densities supported. These formats are often referred to by the product (i.e., generation) in which they were introduced, though these formats (i.e., densities and capacities) are not limited to those products. In an attempt to reduce this potential for confusion, this document makes every effort to refer to these 'formats' by their technical designation (i.e., the DENSITY NAME from REPORT DENSITY SUPPORT - 44h (see 5.2.32 on page 179)). Table 6 provides a cross-reference to the colloquial term that is sometimes used.

Table 6 — Format versus DENSITY NAME

DENSITY NAME [a]	Colloquial term	Product in which introduced
3592A1	J1A Format	3592 J1A
3592A2	E05 Format	TS1120
3592A3	E06 Format	TS1130
3592A4	E07 Format	TS1140
3592A5	E08 Format	TS1150
3592B5	55F Format	TS1155
3592A6	60F Format	TS1160

[a] See REPORT DENSITY SUPPORT - 44h (see 5.2.32 on page 179)

Product Comparisons (see Annex C. on page 533) provides key differences between various products of the 3592 drive.

3.1.2 Exceptional Performance

The IBM Enterprise Tape Drive 3592 has leading-edge streaming and start/stop performance. This is important since many applications operate in start/stop mode. The 3592 streaming performance is more than seven times that of the 3590 and more than twenty times that of the 3490E, with a native data transfer rate of up to 400 MB/sec (1000 MB/sec with compression) depending on generation.

3.1.3 Improved Reliability and Integrity

The advanced IBM Enterprise Tape Drive 3592 uses a bidirectional serpentine recording technique that writes up to 32 data tracks at a time depending on generation. Improved Error Correction Code (ECC) and servo tracks with embedded longitudinal position written on tape help ensure data integrity and reliability. Resident diagnostics monitor operations to detect potential problems and aid in fast resolution.

3.1.4 Wide Platform Connectivity

The IBM Enterprise Tape Drive 3592 has two host interface ports allowing the drive to be shared in a multi-platform open systems environment with the following support varying by device:

a) Fibre Channel ports supporting combinations of speed and topologies including Arbitrated Loop (both private loops (attaching to an L-Port) and public loops (attaching to an FL-Port)), point-to-point (attaching to an N-Port), and Fabric (attaching to an F-Port):
    A) 1/2 GFC (L-Port; FL-Port; F-Port);
    B) 1/2/4 GFC (L-Port; FL-Port; F-Port);
    C) 2/4/8 GFC (L-Port; FL-Port; F-Port); and
    D) 4/8/16 GFC (N-Port; F-Port).

Data can be interchanged across a wide range of platforms. For a current listing of supported configurations, see the web at http://www-03.ibm.com/systems/support/storage/ssic/interoperability.wss.

3.1.5 Host Command Compatibility

The IBM Enterprise 3592 Tape Drive host command support is highly compatible with the IBM Enterprise Tape 3590 family of drives, simplifying device driver and overall application and software support.

3.1.6 High Capacity and Media Reuse

The IBM Enterprise Tape Cartridges used in the various products of 3592 have capacities as shown in table 7.

IBM Enterprise 3592 media is housed in a cartridge with the same physical size as 3480 and 3590 cartridges, enabling coexistence in an IBM Enterprise Automated Tape Library 3494 and IBM Enterprise 3584 Tape Library, together with current media.

The 3592 drive protects future media investments by supporting full forward read and write compatibility into the next generation 3592 drives. The 3592 drive also supports compatibility of the previous generation drives as shown in table 7. The 3592 drive provides an automatic reformatting function to allow current media to be reused and achieve higher capacities. Table 7 shows the media reuse support.

Table 7 — 3592 capacities by density, cartridges, and products

DENSITY NAME Colloquial Term	**Cartridge**	**Supported by** PRODUCT IDENTIFIER						
		3592 J1A 03592J1A	**TS1120** 03592E05	**TS1130** 03592E06	**TS1140** 03592E07	**TS1150** 03592E08	**TS1155** 0359255F	**TS1160** 0359260F
3592A1 J1A Format		WR	WR	RO	-	-	-	-
	JJ/JR	60 GB	100 GB	128 GB	-	-	-	-
	JA/JW	300 GB	500 GB	640 GB	-	-	-	-
3592A2 E05 Format		-	WR	WR	RO	-	-	-
	JB/JX	-	700 GB	700 GB	700 GB	-	-	-
3592A3 E06 Format		-	-	WR	WR	-	-	-
	JB/JX	-	-	1 000 GB	1 000 GB	-	-	-
3592A4 E07 Format		-	-	-	WR	WR	RO	RO
	JB/JX	-	-	-	1 600 GB	-	-	-
	JK	-	-	-	500 GB	500 GB	500 GB	500 GB
	JC/JY	-	-	-	4 000 GB	4 000 GB	4 000 GB	4 000 GB
3592A5 E08 Format		-	-	-	-	WR	WR	WR
	JK	-	-	-	-	900 GB	900 GB	900 GB
	JC/JY	-	-	-	-	7 000 GB	7 000 GB	7 000 GB
	JL	-	-	-	-	2 000 GB	2 000 GB	2 000 GB
	JD/JZ	-	-	-	-	10 000 GB	10 000 GB	10 000 GB
3592B5 55F Format		-	-	-	-	-	WR	WR
	JL	-	-	-	-	-	3 000 GB	3 000 GB
	JD/JZ	-	-	-	-	-	15 000 GB	15 000 GB
3592A6 60F Format		-	-	-	-	-	-	WR
	JM	-	-	-	-	-	-	5 000 GB
	JE/JV	-	-	-	-	-	-	20 000 GB

NOTES:

Values are in GB (10^9) native capacity (no compression)

Devices with updated firmware may support reformatting media written with unsupported formats.

JJ	Economy	JR	Economy WORM
JA	Standard	JW	Standard WORM
JB	Extended	JX	Extended WORM
JK	Advanced Type C Economy	JY	Advanced Type C WORM
JC	Advanced Type C	JZ	Advanced Type D WORM
JL	Advanced Type D Economy	JV	Advanced Type E WORM
JD	Advanced Type D	-	Not Supported
JM	Advanced Type E Economy	WR	Write and Read
JE	Advanced Type E	RO	Read Only

3.1.7 Optimal Data Compression

The IBM Enterprise 3592 uses an optimal dynamic compression method called byte level compression scheme swapping. This insures that the maximum data compression is always achieved, and unlike other tape drive compression methods, the data will never expand.

3.1.8 Hardware Encryption (Some products)

Beginning with some products of the IBM SystemStorage TS1120 Tape Drive this device contains encryption hardware capable of encrypting data at tape speeds. Encrypting data at tape speed helps to avoid the need for host-based encryption of data and the concurrent drain on host performance or the use of specialized encryption appliances. This capability supports high volume data encryption of tape data, helping protect information if tape cartridges are lost or stolen.

3.1.9 WORM Media Support

The IBM Enterprise Tape System 3592 supports critical data archive protection by supporting WORM (Write Once, Read Many) media. This special media and writing mode enables protection of permanent user data by disallowing data overwrite.

3.1.10 Cost Effectiveness

High capacity means that less equipment, fewer cartridges, and fewer tape mounts are required. High performance can reduce the number of drives required. This translates into less floor space for tape cartridge storage, tape drives, and tape libraries. Maintenance costs are also lower than those for high-performance helical and 3480/3490/3590 drives.

A reusable storage asset, the IBM Enterprise 3592 protects existing investments and can be used as the foundation for a broad array of storage solutions.

3.1.11 Ease of Use

IBM Enterprise Tape Drive 3592 features a message display showing device status, activities, error conditions, and messages. An optional operator/service display is available in some configurations which enables extended status and serviceability features.

3.1.12 Service

The IBM Enterprise Tape Drive 3592 does not require scheduled preventive maintenance. The IBM Enterprise 3592 is a single Field Replaceable Unit (FRU) and has no required field serviceable components. IBM customer engineers use an optional hot-pluggable service panel to perform service functions.

3.1.13 Products

The IBM Enterprise 3592 is available in several products and is designed to be incorporated into a variety of configurations including the following depending on the Product:

a) in the IBM Enterprise TS4500 Tape Library;
b) in the IBM Enterprise TS3500 Tape Library;
c) in the C06/C07 Controller;
d) in the TS7700;
e) in a Rack mount;
f) in the IBM Enterprise Automated Tape Library (3494);
g) in the J70 Controller;
h) in the TotalStorage Model C20 with one to twenty tape drives providing attachment to the StorageTek 4410 and 9310 ACS; and
i) in the IBM Enterprise TS3400 Tape Library.

3.1.14 Storage Management Software

Tivoli® Storage Manager (TSM) is a client/server storage management product that provides save/restore, archive, hierarchical storage, and disaster recovery functions for networked workstations and servers. TSM uses the full capacity of the IBM Enterprise Tape System 3592. TSM also supports the IBM Enterprise 3592 in an Enterprise Tape Library 3494, including libraries mixed with TotalStorage Enterprise 3590 for data migration.

Many other popular automated storage software products support the IBM Enterprise 3592 Tape Systems and 3494 Tape Libraries. For a complete list of these products see the ISV Compatibility Matrix on the web at

http://www-03.ibm.com/systems/storage/tape/pdf/compatibility/ts1120_isv_matrix.pdf.

4. Implementation Considerations

4.1 Power-on, and power cycle considerations

When power is applied to this device, some time is required for the electronics to reach a state where the microcode/firmware may be initialized. Once the firmware begins initializing, power-on diagnostics are run and the drive attempts to communicate with an attached library, if any. While this device has been designed to optimize the amount of time it takes to perform necessary diagnostics and establish communications with the library, this still requires a non-trivial amount of time.

Hosts communicating with this device must allow sufficient time for the device to become ready to accept commands and to report its configuration. The tape drive logical unit (i.e., LUN 0) requires a certain amount of time to accept commands and the library logical unit (i.e., LUN 1) through which communication with the library occurs requires a different amount of time to be ready to accept commands and report its configuration.

The amount of time required for the device to be ready to accept commands varies by device generation. Also, the time to ready on LUN 1 depends on, not only timing internal to this device, but also timing in the library. This is especially evident during a power-on of a large library where the library needs to configure many drives as each drive in that library is also powered on.

On some host systems when configuration of attached devices occurs, the configuration defines the complete set of devices to which communication is allowed to be attempted. This set of devices never changes without explicit action such as a power cycle of the server or a user explicitly running a configuration utility (e.g., cfgmgr on AIX®). Therefore, both LUN 0 and LUN 1 need to be fully configured and communicating before performing a host device configuration.

If this device is reset, the reset is either a hard reset or a soft reset. A hard reset causes the same behavior as when power is applied to a device that previously had no power applied. A soft reset performs many of the same functions that occur during a hard reset, but some initialization is not required. As such, a soft reset requires less time to complete than a hard reset.

If the microcode/firmware in this device is updated, the device completes the microcode update process by forcing a device hard reset.

If this device resets or powers on with a volume loaded, then a mid-tape recovery (MTR) may occur. An MTR causes some commands that are placed into the queue to wait until MTR completes prior to being processed and status returned.

It is recommended that a library containing this device and using ADI as the interface, saves the ENABLE bit set to one and the CACHE bit set to one in the SMC logical unit descriptor in the Logical Unit subpage of the ADC Device Server Configuration mode page (i.e., set the bits to one and set the SP bit to one in the MODE SELECT command used to set them). This greatly reduces issues with servers during power cycle events and thereafter.

The device behavior during a power-on or power reset must be examined on a per logical unit basis. Figure 2 describes this devices power-on sequencing assuming this device has previously been powered on in the current library configuration (i.e., the power-on sequencing during initial installation is not shown).

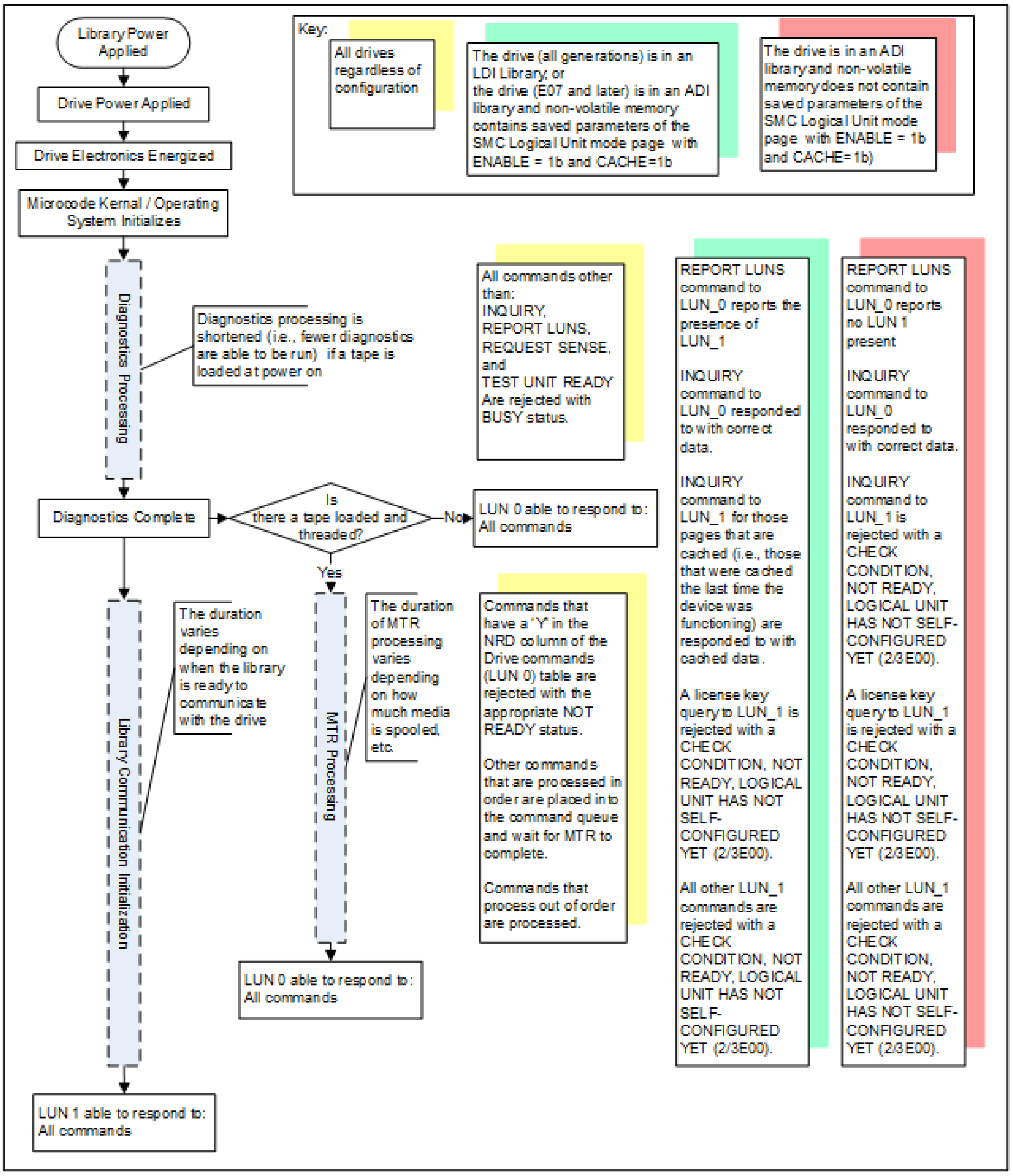


Figure 2 — Drive Power-on Sequencing

4.2 Addressing Assignments

Each Fibre Channel interface port for this device can be independently assigned a specific speed and topology, or may be set to auto-negotiate.

When the topology is set to or negotiates to L-port, a hard or soft ALPA ID may be assigned. The hard ALPA ID is in the range of 01h to EFh with only certain valid values (a total of 126 addresses). Validity is enforced by the entry process. This value should be unique to each device on the Fibre Channel loop. Fibre Channel loop protocol will detect an addressing conflict on the loop, and one of the conflicting drives will not be available for use.

The 60F product dropped support for L-port.

TS1155E and TS1160E are Ethernet devices supporting RoCEv2 with support for DHCP and library assigned addresses.

4.3 Object buffer

4.3.1 Object buffer introduction

This device contains an object buffer capable of holding logical objects being written to the medium or logical objects being transferred from the medium in read-ahead operations. The object buffer is used during write operations when the device is configured to use buffered mode (i.e., the BUFFER MODE field of the mode parameter header is set to a non-zero value per Mode Parameter Header for Mode Select (6/10) (see 6.6.1.1 on page 374))) and during read operations regardless of the buffer mode.

When the device is reading logical objects from the medium, it uses the object buffer in a read-ahead fashion to improve performance. Logical objects are read from the medium and placed into the object buffer such that they are available to an application that is reading without the application being required to wait for each block to be read from the medium prior to being transferred on the SCSI interface. Read-ahead operations often occur at the conclusion of space, locate, and load operations in order to prime the object buffer with logical objects in case a read operation follows.

4.3.2 BOP caching (model E08+)

Devices starting with model E08 use a small portion of the object buffer as a cache to retain data read at BOP while the remainder of the object buffer is used for read-ahead operations. The data around BOP, once read, is generally retained in the BOP cache until a demount or partition change. If D_BOPC of the MP 30h[43h]: Feature switches - Device attribute settings (see 6.6.22.5.3 on page 442) mode page is set to 0b and a command is received by the device that requests access to a logical object identifier (aka, logical block address) that is in the BOP cache, then that data is read from the BOP cache without requiring actual access to the medium (i.e., it uses the cached data without changing the physical location of the medium). This allows for a volume that is located away from BOP to read the data around BOP very quickly without disturbing the current physical position of the medium. If D_BOPC of the MP 30h[43h]: Feature switches - Device attribute settings mode page is set to 1b, then a request to perform positioning changes the physical location of the medium and performs a read-ahead operation as appropriate.

4.3.2.1 BOP caching side effects

It is important to understand the side effects that BOP caching may present:

a) Processing time to position to BOP may be transferred from the positioning command to a subsequent command (e.g., The time for a REWIND could be transferred to a subsequent UNLOAD);
b) When reading data in the BOP cache and the command requests a read through the cache boundary to data not in the cache, there may be processing time to position the medium to the proper position to read the data, as well as the time to read the data from the medium;
c) If a sequence of commands like:
    1) REWIND;
    2) READ one block;
    3) LOCATE to position prior to the REWIND;
    4) READ one block;
    5) Go to step 1,

    is performed, the tape will typically move in a sequential fashion like a READ without positioning to BOP in each iteration.

4.4 Multiple Port Behavior

The two primary interface ports provide alternate paths through which the logical unit(s) of the device may be reached. The ports are referred to as Port 0 and Port 1. Each port maintains its own unique settings and address.

Using the CE service panel, the operator may manually set each port in an online or offline state.

When an offline port is set online, all initiators on that port receive a Unit Attention condition.

Offline ports do not generate or maintain Unit Attention conditions for initiators while the port is in an offline state.

Usage of the device with both ports online is required for dual port failover to function correctly. Generally, all initiators, regardless of port, are treated the same as multiple initiators on the same port. The exception to this is the handling of mode pages and reservations when a hard port reset condition occurs (such as loss of light, etc). The following rules are described with respect to a local interface (the host port on which the hard reset condition occurred) and a remote interface (the other host port to which the device is attached).

a) If there are no reservations when a hard reset condition occurs, all mode pages are reset. All initiators on the local interface receive a Unit Attention condition for Power On, Reset, or Device Reset Occurred. All initiators on the remote interface receive a Unit Attention condition for Mode Parameters Changed.
b) If there are one or more reservations when a hard reset condition occurs and all reservations were granted to initiators on the local interface, all mode pages are reset and all SPC-2 reservations are reset. All persistent reservations remain in effect. All initiators on the local interface receive a Unit Attention condition for Power On, Reset, or Device Reset Occurred. All initiators on the remote interface receive a Unit Attention condition for Mode Parameters Changed.
c) If there are one or more reservations when a hard reset condition occurs and one or more of the reservations were granted to an initiator on the remote interface, only those mode pages and SPC-2 reservations unique to each initiator on the local interface are reset. Mode pages and reservations unique to each initiator on the remote interface are not reset. Mode pages which are defined as common to all initiators are not reset. All initiators on the local interface receive a Unit Attention condition for Power On, Reset, or Device Reset Occurred. All initiators on the remote interface see no effects of the hard reset condition on the other interface.

4.5 Mode Page Behaviors

4.5.1 Policy — non-standard

This device implements a non-standard behavior related to mode page policy. The mode page policies defined are:

Mode page policy	Number of mode page copies
<Shared>	One copy of the mode page that is shared by all I_T nexuses.
<Per target port>	A separate copy of the mode page for each target port with each copy shared by all initiator ports.
<Per I_T nexus>	A separate copy of the mode page for each I_T nexus

4.5.1.1 Mode parameter header and block descriptor policy

This device implements mode page policy in a manner different than specified in T10/SPC-4. The mode page policy for mode parameter header and block descriptor values depends on the specific parameter as shown here for the applicable parameters:

Field	Mode page policy
BUFFERED MODE	<Shared>
SPEED	<Shared>
BLOCK LENGTH	<Per I_T nexus>

4.5.1.2 Mode page policy

The mode page policy implemented by this device is shown in table 8

Table 8 — Mode page policy (part 1 of 2)

Mode Page	MLUS [a]	Mode page policy	Returned in MP 3Fh [b]
MP 01h: Read-Write Error Recovery (see 6.6.5 on page 383)	-	<Shared>	Y
MP 02h: Disconnect-Reconnect (see 6.6.6 on page 385)	Y	<Per I_T nexus>	Y
MP 0Ah: Control Mode (see 6.6.7 on page 387)	-	<Shared>	Y
MP 0Ah[01h]: Control Extension (see 6.6.8 on page 389)	Y	<Shared>	-
MP 0Ah[F0h]: Control Data Protection (see 6.6.9 on page 390)	-	<Per I_T nexus>	-
MP 0Fh: Data Compression (see 6.6.10 on page 392)	-	<Shared>	Y
MP 10h: Device Configuration (see 6.6.11 on page 394)	-	<Shared>	Y
MP 10h[01h]: Device Configuration Extension (see 6.6.12 on page 397)	-	<Shared>	-
MP 11h: Medium Partition Page (see 6.6.13 on page 399)	-	<Shared>	10
MP 18h: Fibre Channel Logical Unit Control (see 6.6.14 on page 405)	Y	<Per I_T nexus>	Y

Key:

-	No
Y	Yes
6	MODE SENSE 6
10	MODE SENSE 10

[a] The MLUS (multiple logical units share) indicates if this mode page—subpage combination may be shared by other logical units (e.g., The FCP port (19h) page controls port related functions)

[b] Whether on not the mode page is returned in mode page 3Fh or mode page 3Fh[FFh] depend on the command used (i.e., MODE SENSE 6 or MODE SENSE 10) and is indicated in this column. Some vendor-specific pages are not returned with an all pages request.

Table 8 — Mode page policy (part 2 of 2)

Mode Page	MLUS [a]	Mode page policy	Returned in MP 3Fh [b]
MP 19h: Fibre Channel Port Control (see 6.6.15 on page 406)	Y	<Per target port>	Y
MP 1Ch: Informational Exceptions Control (see 6.6.16 on page 408)	-	<Per I_T nexus>	Y
MP 21h: TOD Control (see 6.6.17 on page 411)	-	<Shared>	Y
MP 22h: Language (see 6.6.18 on page 413)	-	<Shared>	-
MP 23h: Medium Sense (see 6.6.19 on page 414)	-	<Shared>	Y
MP 24h: Initiator-Specific Extensions (see 6.6.20 on page 424)	-	<Shared>	Y
MP 25h: Read/Write Control (see 6.6.21 on page 427)	-	<Shared>	Y
MP 30h: Device Attribute Settings (see 6.6.22 on page 431)	Y	<Shared>	-
MP 3Eh: Engineering Support (see 6.6.24 on page 455)	-	<Shared>	-

Key:

-	No
Y	Yes
6	MODE SENSE 6
10	MODE SENSE 10

[a] The MLUS (multiple logical units share) indicates if this mode page—subpage combination may be shared by other logical units (e.g., The FCP port (19h) page controls port related functions)

[b] Whether on not the mode page is returned in mode page 3Fh or mode page 3Fh[FFh] depend on the command used (i.e., MODE SENSE 6 or MODE SENSE 10) and is indicated in this column. Some vendor-specific pages are not returned with an all pages request.

4.5.2 Classification of mode parameters

The page control (PC) field of the MODE SENSE command indicates four classifications of mode pages:

Value	Description
00b	Current values
01b	Changeable values
10b	Default values
11b	Saved values.

This device has the following behaviors for mode parameters:

Table 9 — Mode parameter change behavior

Term	Values reported for Mode Sense with page control of Changeable values (01b)	Action when a value of a field recieved is different than the Current values (00b) [a]
(changeable)	The bits of this field are set to one in the parameter data returned to a MODE SENSE command with the PC field set to 01b (i.e., Changeable values).	The current value is updated.
(changeable-special)		See the description of the parameter to determine the action (e.g., the parameter may be writeable and change the behavior to that indicated by the received value, but not update the Current values).
(changeable-saveable)		The current value is updated. If the SP bit in the MODE SELECT CDB is set to one, then the value for the Saved values (11b) for this page is updated and saved to non-volatile memory before SCSI status is returned.
(changeable-ignored)		The current value is unchanged. No action is taken.
(non-changeable)	The bits of this field are set to zero in the parameter data returned to a MODE SENSE with a PC field set to 01b (i.e., Changeable values).	The MODE SELECT command is rejected with a 5/2600h (i..e, ILLEGAL REQUEST, INVALID FIELD IN PARAMETER LIST).

[a] A value in the mode parameter data received with a MODE SELECT command is different than the value in the mode parameter data returned to a MODE SENSE command with the PC field set to 00b (i.e., Current values).

This device implements the following features differently than specified in SPC-4:

a) Save behavior — non-standard (see 4.5.2.1 on page 29); and
b) Parameter Saveable behavior — non-standard (see 4.5.2.2 on page 29).

4.5.2.1 Save behavior — non-standard

This device implements mode parameter saving in a manner different than specified in SPC-4. The SP bit of the MODE SELECT command (see 5.2.14) applies only to the parameters sent in parameter data to that MODE SELECT command. No other mode parameters' Current values are saved. This is contrary to SPC-4 which mandates that the Current values of all saveable mode pages be saved if the SP bit is set to one.

4.5.2.2 Parameter Saveable behavior — non-standard

The parameter saveable (PS) bit in the mode parameters is set to one in the parameter data returned to a MODE SENSE if at least one mode parameter in the page is saveable. Since only some parameters are saveable and others are not, it may be possible that some of the changeable parameters in the page are saveable and other changeable parameters in the page are not. There is no programmatic method for retrieving a list of which specific mode parameters are saveable.

The parameter saveable (PS) bit in the mode parameters is ignored during the processing of a MODE SELECT command.

4.5.3 Mode parameters and unit attentions

Some mode parameters, including mode parameters in the mode parameter header, in the block descriptor, and in some mode pages are affected by mounting a volume. When this occurs, there is no unit attention for MODE PARAMETERS CHANGED (i.e., 6/2A01h) established.

4.6 Programmable early warning

When writing, the application client may need an indication prior to early warning to allow for the application client to prepare to be ready for early warning (e.g., flush buffers in the application client).

Application clients that need this indication may request the device server to create a zone called the programmable-early-warning zone (PEWZ) by setting the PEWS field (see 6.6.12) of the Device configuration extension mode page to the requested size of the PEWZ. The EOP side of PEWZ is established at early-warning and extends towards BOP for a distance indicated by the PEWS field. See figure 3.

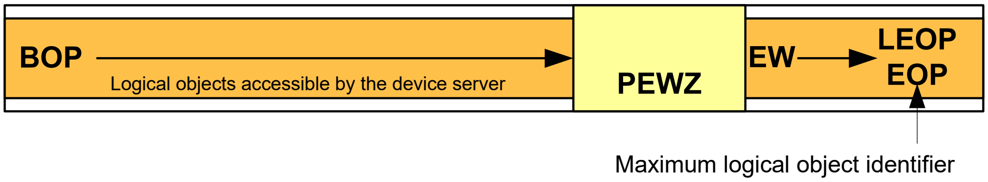


Figure 3 — Programmable early warning example

WARNING

If PEWZ is used, all applications that may access the drive when a PEWZ exists, should support PEWZ or there is a risk of the application that does not support PEWZ detecting an unknown error or a diminished capacity when the PROGRAMMABLE EARLY WARNING error is reported.

WARNING

The REW bit in the Device Configuration mode page (see 6.6.11) shall have no effect on the device server behavior in the PEWZ.

The device server shall return CHECK CONDITION status, with the sense key set to NO SENSE, the EOM bit set to one and the additional sense code set to PROGRAMMABLE EARLY WARNING DETECTED at the completion of a command that caused the medium to transition into the PEWZ if that command is:

a) WRITE(6);
b) WRITE(16);
c) WRITE FILEMARKS(6); or
d) WRITE FILEMARKS(16).

Encountering the PEWZ shall not cause the device server to perform a synchronize operation or terminate the command. If processing this command results in any other exception condition except early-warning, the CHECK CONDITION status associated with that exception condition shall be reported instead. If early-warning is crossed prior to the PROGRAMMABLE EARLY WARNING DETECTED additional sense being reported, the PROGRAMMABLE EARLY WARNING DETECTED additional sense shall be reported before the early-warning CHECK CONDITION.

If the PROGRAMMABLE EARLY WARNING DETECTED additional sense code was not reported, the next write in PEWZ or beyond early-warning that would otherwise complete with GOOD status, shall return the programmable-early-warning CHECK CONDITION instead.

If the PEWZ is entered and exited on the BOP side before the PROGRAMMABLE EARLY WARNING DETECTED additional sense code is returned, the device server shall not report CHECK CONDITION status with the additional sense code set to PROGRAMMABLE EARLY WARNING DETECTED.

4.7 Logical block protection

4.7.1 Logical block protection overview

The device contains hardware that is capable of checking and generating protection information (i.e., 4-byte CRC) that is transferred with logical blocks between the device server and an application client. This protection information transferred with logical blocks is saved to the medium with each logical block and read from the medium with each logical block. This protection information is validated at the destination prior to completing the task thereby ensuring that the logical block has not been corrupted. This level of detection is not achievable by methods where the application client inserts vendor-specific data protection information in its data. All 3592 devices support logical block protection using a vendor-specific means known by the term CRC Protection (see 4.7.1.1). Some devices support a standardized method of logical block protection (see 4.7.1.2). Setting of CRC Protection and Logical block protection are mutually exclusive. 3952 E08 devices and later also support the choice of which CRC algorithm may be used for transfers between the drive and the host. In all cases, the protection information (i.e., CRC) saved with the data on the medium is the same (see 4.7.2). The protection method (if any) used to write a given block does not need to be the same as the method (if any) used to read that same block. This includes where a drive (e.g., prior generation) which does not support the protection method used to write a given block may read those blocks using any (or no) protection method supported on the reading drive.

4.7.1.1 CRC Protection (vendor-specific)

The CRC Protection method is supported on all 3592 devices.This is the vendor-specific method of using protection information and is enabled by setting the CRC Target Enablement field of the MP 24h: Initiator-Specific Extensions (see 6.6.20 on page 424) mode page to a non-zero value. CRC Protection is disabled by setting the CRC Target Enablement field of the Initiator-Specific Extensions mode page to 00h. CRC Protection, when enabled checks and transfers the CRC on read data, write data, and recovered buffered data. There is no individual selection.

4.7.1.2 Logical block protection (standardized)

The Logical block protection method is supported on 3592 devices starting with the following code levels:

a) D3I1_F1F for E05 devices;
b) D3I2_7B0 for E06 devices;
c) D3I3_5CD for E07 devices; and
d) all code levels for E08+ devices.

The Logical block protection method is not supported on 3592 J1A devices.

Logical block protection method is the standardized method of using protection information and is configured using MP 0Ah[F0h]: Control Data Protection (see 6.6.9 on page 390). Logical block protection is enabled by setting the LOGICAL BLOCK PROTECTION METHOD field of the Control Data Protection mode page to a non-zero value. Logical block protection is disabled by setting the LOGICAL BLOCK PROTECTION METHOD field of the Control Data Protection mode page to zero.

A device that supports using this protection information sets:

a) the PROTECT bit in standard inquiry (see 5.2.9.1) to one;
b) the SPT field of the Extended INQUIRY Data VPD page (see 6.3.5) to 001b; and
c) the value returned in the MAXIMUM BLOCK LENGTH LIMIT field of the READ BLOCK LIMITS command to a value which when added to THE LARGEST VALUE SUPPORTED IN THE LOGICAL BLOCK PROTECTION INFORMATION LENGTH field of the Control Data Protection mode page is less than or equal to the maximum length able to be represented in commands that transfer logical blocks between the application client and the device server.

4.7.1.3 Interaction between CRC Protection and Logical block protection

CRC Protection method and Logical block protection method support the same CRC algorithm(s) and transfer the CRC in the same manner. The only differences are how the drive reports support for the method, how the method and CRC algorithm are enabled and disabled, and how errors in the CRC detected during the transfer of data are reported. Since CRC errors are reported differently between the two methods only one method is allowed to be enabled at any time. An attempt to enable a method when the other method is enabled causes the other method to be disabled and the mode pages settings to be modified to reflect that disablement.

4.7.2 Protection information on a volume

A recorded volume contains logical objects (see 4.2.7.1) and format specific symbols. Logical objects are application client accessible. Format specific symbols are used by the device server to provide methods for recording logical objects on the medium in a manner that allows them to be successfully read at a later date and may not be application client accessible. Format specific symbols contain information used to protect logical objects. The drive includes the protection information field as one of the format specific symbols. The format specific symbol that is the protection information field is written to the medium with each logical block. The protection information used as a format specific symbol by the drive is a 4-byte Reed-Solomon CRC (see E.1.2.). A representation of logical objects and format specific symbols is shown in figure 4.

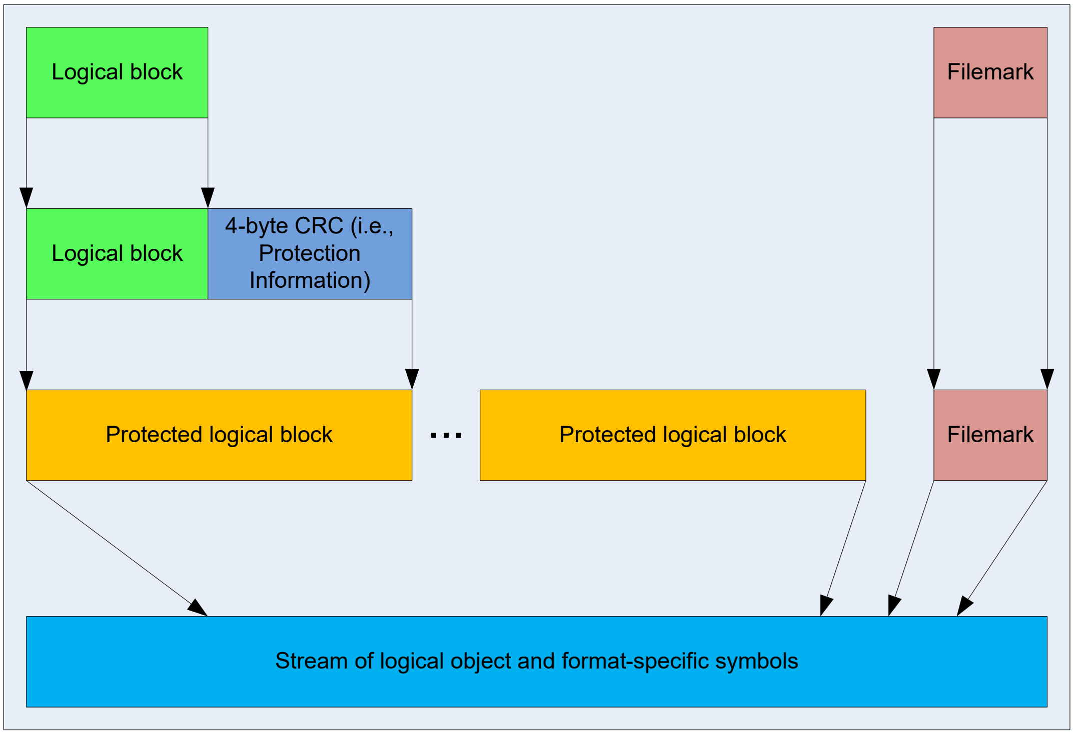


Figure 4 — Protection information shown in relation to logical objects and format specific symbols

The device generates the protection information and adds it to a logical block before recording the logical block to the medium if the command that transferred the logical block being recorded to medium was received on an I_T_L nexus for which the CRC TARGET ENABLEMENT field of the Initiator-Specific Extensions mode page is set to zero and the LOGICAL BLOCK PROTECTION METHOD field of the Control Data Protection mode page:

a) is set to zero; or
b) is set to a non-zero value and the LBP_W bit of the Control Data Protection mode page is set to zero.

The drive reads the protection information from the medium, validates it, and removes it from the logical block before transferring the logical block to the application client if the command that is requesting the transfer of a logical block being read was received on an I_T_L nexus for which the CRC TARGET ENABLEMENT field of the Initiator-Specific Extensions mode page is set to zero and the LOGICAL BLOCK PROTECTION METHOD field of the Control Data Protection mode page:

a) is set to zero; or
b) is set to a non-zero value and the LBP_R bit of the Control Data Protection mode page is set to zero.

Protection information may be:

a) compressed;
b) encrypted; or
c) included in byte counts in log parameters.

NOTE 1 - Device side counters reported in log pages generally include bytes from the protection information at all times. Host side counters reported in log pages when CRC Protection and Logical block protection are disabled generally do not include bytes from the protection information. Host side counters reported in log pages when CRC Protection is enabled or when Logical block protection is enabled generally include bytes from the protection information.

4.7.3 Logical blocks and protection information

A) If the LOGICAL BLOCK PROTECTION METHOD field of the Control Data Protection mode page is set to zero and the CRC TARGET ENABLEMENT field of the Initiator-Specific Extensions mode page is set to zero for a specific I_T_L nexus then a logical block transferred between the application client and the device server through that I_T_L nexus is defined by Table 10

Table 10 — Logical block with no protection information

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Data							
n-1								
n = the TRANSFER LENGTH field specified in CDB for variable length transfers; the BLOCK LENGTH field specified in the mode parameter header (see SPC-4) for fixed block transfers.								

If the LOGICAL BLOCK PROTECTION METHOD field of the Control Data Protection mode page is set to a non-zero value or if the CRC TARGET ENABLEMENT field of the Initiator-Specific Extensions mode page is set to a non-zero value for a specific I_T_L nexus then a logical block transferred between the application client and the device server through that I_T_L nexus is defined by Table 11

Table 11 — Logical block with protection information

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Data							
n-x-1								
n-x	Protection Information							
n-1								
n = the TRANSFER LENGTH field specified in the CDB for variable length transfers; the BLOCK LENGTH field specified in the mode parameter header (see SPC-4) for fixed block transfers. x = the LOGICAL BLOCK PROTECTION INFORMATION LENGTH specified in the Control Data Protection mode page.								

If the protection information to be transferred between the drive and the host is not the Reed-Solomon CRC, then the protection information is transformed between the Reed-Solomon CRC and the CRC algorithm selected (see 6.6.9 on page 390 and 6.6.20 on page 424).

4.7.4 Protecting logical blocks transferred during writes

If CRC Protection is enabled for a specific I_T_L nexus or if the LOGICAL BLOCK PROTECTION METHOD field and LBP_W bit of the Control Data Protection mode page (see 6.6.9) is set to a non-zero value for a specific I_T_L nexus, then each logical block transferred from the application client through that I_T_L nexus due to a WRITE(6) command is expected to contain protection information.

For the WRITE(6) command, the device server validates the protection information before the logical block is written to medium. If the FIXED bit in the CDB is set to one each logical block is validated before being written to the medium. If the validation of the protection information for a logical block fails, then the processing of the command terminates prior to writing the failed logical block to the medium. If the validation of the protection information fails, the device server;

a) if Logical block protection (standardized) (see 4.7.1.2 on page 31) is enabled, reports a CHECK CONDITION status with Sense Code of Current or Deferred, the sense key set to HARDWARE ERROR and the additional sense code set to LOGICAL BLOCK GUARD CHECK FAILED; or
b) if CRC Protection (vendor-specific) (see 4.7.1.1 on page 31) is enabled, reports a CHECK CONDITION status with Sense Code of Current or Deferred, the sense key set to HARDWARE ERROR and the additional sense code set to Write Internal CRC Error (47h/81h).

An application client shall add the protection information on each logical block before transferring that logical block and shall increase the TRANSFER LENGTH field by the length of the logical block protection information if it has:

a) set Logical Block Protection by:
   A) setting the LOGICAL BLOCK PROTECTION METHOD field of the Control Data Protection mode page to a non-zero value; and
   B) setting the LBP_W bit of the Control Data Protection mode page to one;

   or

b) set CRC Protection by setting the CRC TARGET ENABLEMENT field of the Initiator-Specific Extensions mode page to a non-zero value.

The application client should add the protection information to the logical block at the earliest point possible. If the data has had the protection information added to the logical block at some point in the application client prior to the hardware that transfers the logical block, then the protection information should be validated when it is transferred. If the validation fails, then the application client should abort the command and report a status to the user that validation failed.

NOTE 2 - The device server treats the LOGICAL BLOCK PROTECTION INFORMATION field as the protection information. If the protection information is not added to the logical block, then the validation fails when the bytes used do not validate (e.g., the last 4-bytes of the logical block are treated as the CRC and the last 4-bytes of the logical block do not calculate as the CRC of the previous data)

4.7.5 Protecting logical blocks transferred during reads

If CRC Protection is enabled for a specific I_T_L nexus or if the LOGICAL BLOCK PROTECTION METHOD field and the LBP_R bit of the Control Data Protection mode page (see 6.6.9) is set to a non-zero value, then the protection information is transferred with the logical block to the application client on that I_T_L nexus. The commands for which this applies are:

a) READ(6);
b) READ REVERSE(6); and
c) VERIFY(6) with the BYTCMP bit set to zero.

The protection information is validated by the device server before sending status to the command that caused the transfer of the logical block. If the FIXED bit in the CDB is set to one each logical block is validated before being transferred to the application client. If the validation of the protection information for a logical block fails, then the processing of the command is terminated prior to transferring any additional blocks to the application client. If the validation of the protection information fails, then the device server:

a) if Logical block protection (standardized) (see 4.7.1.2 on page 31) is enabled, reports a CHECK CONDITION status with Sense Code of Current Sense, the sense key set to HARDWARE ERROR and the additional sense code set to LOGICAL BLOCK GUARD CHECK FAILED; or
b) if CRC Protection (vendor-specific) (see 4.7.1.1 on page 31) is enabled, reports a CHECK CONDITION status with Sense Code of Current Sense, the sense key set to HARDWARE ERROR and the additional sense code set to Read Internal CRC Error (47h/80h).

An application client should validate the protection information on each logical block at the latest point possible before using the data if it has:

a) set Logical Block Protection by:
   A) setting the LOGICAL BLOCK PROTECTION METHOD field of the Control Data Protection mode page to a non-zero value; and
   B) setting the LBP_R bit of the Control Data Protection mode page to one;

   or

b) set CRC Protection by setting the CRC TARGET ENABLEMENT field of the Initiator-Specific Extensions mode page to a non-zero value.

4.7.6 Protecting logical blocks transferred from the object buffer in response to a RECOVER BUFFERED DATA command

If the LOGICAL BLOCK PROTECTION METHOD field and the RBDP bit of the Control Data Protection mode page (see 6.6.9) is set to a non-zero value or the RBD DATA CHECKED bit of the Initiator-Specific Extensions mode page is set to one for a specific I_T_L nexus, each logical block transferred between the device server and the application client on that I_T_L nexus during a RECOVER BUFFERED DATA command (see 5.2.30) shall include the protection information.

The protection information for each block is validated before sending status to the command. If the validation of the protection information fails for any logical block, the device server terminates the command without transferring any additional logical blocks that may exist in the object buffer and;

a) if Logical Block Protection is enabled, reports a CHECK CONDITION status with Sense Code of Current Sense, the sense key set to HARDWARE ERROR and the additional sense code set to LOGICAL BLOCK PROTECTION ERROR ON RECOVER BUFFERED DATA; or
b) if CRC Protection is enabled, reports CHECK CONDITION status with Sense Code of Current Sense, the sense key set to HARDWARE ERROR and the additional sense code set to Read Internal CRC Error (47h/80h).

An application client should validate the protection information on each logical block at the latest point possible before using the data if:

a) it has enabled Logical Block Protection by:
   A) setting the LOGICAL BLOCK PROTECTION METHOD field of the Control Data Protection mode page to a non-zero value; and
   B) the RBDP bit of the Control Data Protection mode page to one;

   or

b) it has enabled CRC Protection by setting the CRC TARGET ENABLEMENT field of the Initiator-Specific Extensions mode page to a non-zero value.

4.7.7 File verification of protection information

An application client may verify that protection information is present on each logical block on the medium between the current position and a specified number of filemarks from the current position and that the protection information validates correctly by setting to one the VBF bit and setting to zero the VLBPM bit of the VERIFY (6) command (see 5.2.47) and setting to one the LBP_R bit of the Control Data Protection mode page (see 6.6.9). The device reads the medium verifying that each logical block between the current position and the nth filemark is protected with the protection information and that the protection information validates as shown in figure 5.

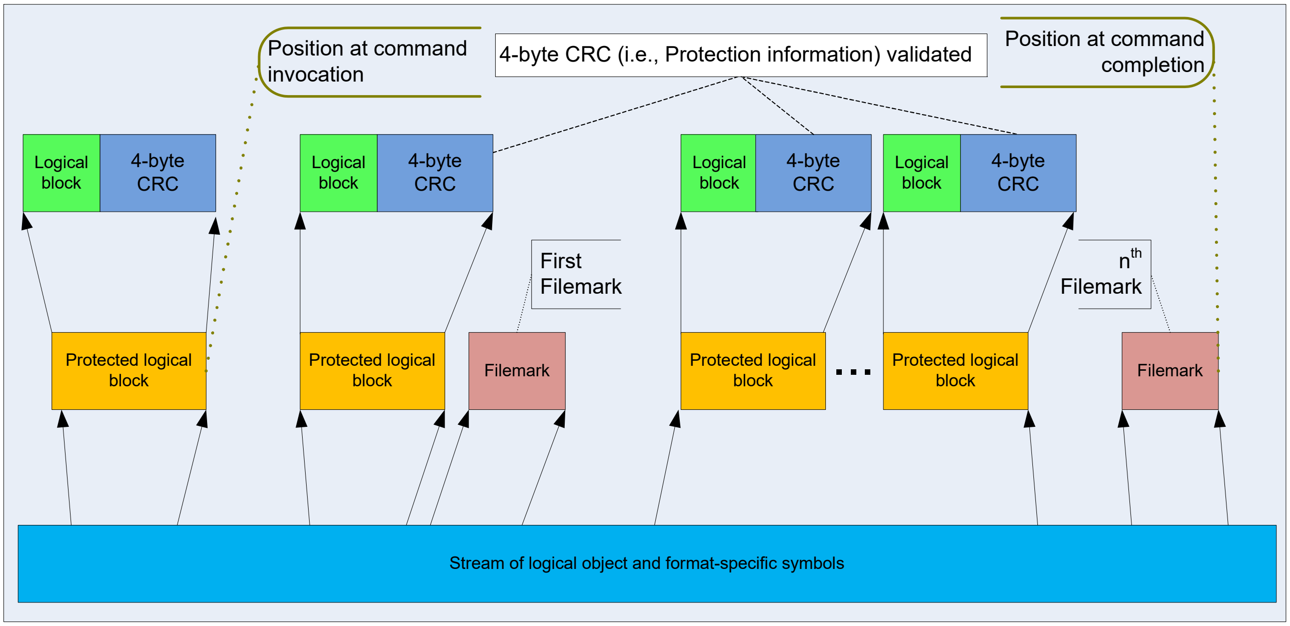


Figure 5 — Example file verification of protection information

4.7.8 Verification to EOD of protection information

An application client may verify that protection information is present on every logical block on the medium between the current position and EOD and that the protection information validates correctly setting to one the VBE bit and setting to zero the VLBPM bit of the VERIFY (6) command (see 5.2.47) and setting to one the LBP_R bit of the Control Data Protection mode page (see 6.6.9). The device reads each logical block between the current position and end-of-data verifying that each logical block is protected with the protection information and that the protection information validates as shown in figure 6.

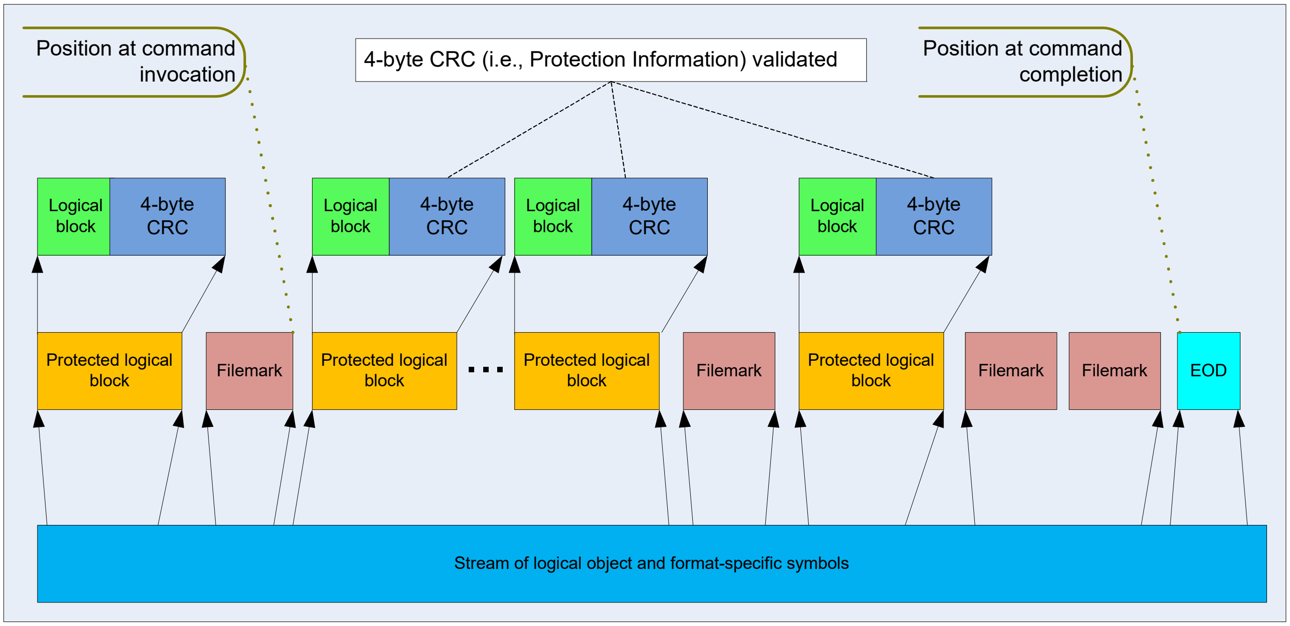


Figure 6 — Example verification to EOD of protection information

4.8 Data Transfer, Block Limits, and Fixed Block Option

This device is designed to buffer multiple records. Logical blocks may be prefetched to the buffer before they are requested by a READ command or held in the buffer after they are written by a WRITE command. For successive sequential-motion operations, the presence of the buffering in the device does not adversely affect the performance of the subsystem. Non-sequential motion does not result in errors, but may result in delays because of requirements to synchronize buffers or discard read ahead data. Buffer management in the device determines when to read additional data from the medium into the buffer, or when to write data from the buffer to the medium. A logical block is not written to tape until the block is entirely received into the buffer.

When the FIXED bit of the command is set to 1b, each command transfers zero or more logical blocks. The subsystem takes appropriate action to assemble or disassemble the logical blocks being transferred over the interface so that they remain independent blocks on the medium. There is no guarantee that the group of blocks transferred by the WRITE command is requested as a group by a subsequent READ command, therefore the device must be prepared to assemble and disassemble on a block boundary. This is managed by treating all blocks and filemarks as independent from one another, both for data compaction and for recording.

When the FIXED bit of the command is set to 0b, each command transfers zero or one logical blocks.

During the processing of a WRITE command, if EW is encountered or the medium is positioned between EW and LEOP, the command is completed with CHECK CONDITION status 0/0002 (END-OF-PARTITION/MEDIUM DETECTED, EARLY WARNING) and EOM set to 1b. In some devices (i.e., prior to some code levels), one logical block is transferred before terminating the command with the check condition. In other devices (i.e., more recent code levels), all blocks in the current command are transferred and the command is completed with the check condition.

The behavior when in the EW region related to the synchronization of the object buffer is determined by the setting of the SEW bit in MP 10h: Device Configuration (see 6.6.11 on page 394).

The device supports a minimum logical block length of 1 and a maximum logical block length is 2,097,152 bytes (200000h). Any block length between the limits is also supported. Refer also to READ BLOCK LIMITS - 05h (see 5.2.22) for further information on block sizes and limitations.

For write type commands, transfer lengths larger than the maximum device supported block size are terminated with CHECK CONDITION status of 5/2400 (ILLEGAL REQUEST, INVALID FIELD IN CDB). A transfer length of 000000h indicates that no bytes/blocks are transferred. This condition is not considered an error and the logical position is not changed.

For read type commands, including READ, READ REVERSE, and RECOVER BUFFERED DATA, transfer lengths larger than the maximum device supported block size are accepted and the underlength condition rules are applied for transfer requests bigger than the actual block size. A transfer Length of 000000h indicates that no bytes/blocks are transferred. This condition is not considered an error and the logical position is not changed.

4.9 Archive mode unthread (E08+)

E08 and 55F drives support choosing a trade-off between fast unthread times without optimizing the preparation of the medium for long term storage and slower unthread times that optimize the preparation of the medium for long term storage. This trade-off is selected through the Archive mode unthread feature. On 60F drives this feature is not configurable. This feature only applies to some media types.

Archive mode unthread is able to be invoked either by using the RETEN bit of the LOAD/UNLOAD command (see 5.2.10) or by configuring the drive to use the Archive mode unthread for every unload that occurs. The drive is configured to use the Archive mode unthread for every unload that occurs by setting the E_ARCHIVE bit to one in the MP 30h[43h]: Feature switches - Device attribute settings (see 6.6.22.5.3) mode page. On 60F drives this feature is not configurable. However, the E_ARCHIVE bit is allowed to be changed, but is ignored. This mode parameter has a mode parameter policy of (changeable-saveable) and may be saved by setting the SP (save pages) bit to one in the MODE SELECT command.

WARNING WARNING

Archive mode unthread should be used for volumes that may be stored for extended periods of time.

WARNING WARNING

4.10 Request Sense Information, ILI, and Command Interactions

The behavior and interactions between some of the commands and the Information and ILI fields in Request Sense are rather complicated. This section details the various commands which may set the Information or ILI fields, and summarizes the relationship between such commands, their parameters, the encountered conditions, the reported status, and the expected behavior of these fields and the resulting device position.

4.10.1 General Read-Type Handling

Commands which return block data from media or the buffer to the host have the same general behavior. These commands include Read, Read Reverse, and Recover Buffered Data. The major difference between these is centered around the logical direction of the read (order of the blocks read with respect to their write order), and its effect on positioning.

Forward commands include Read, and Recover Buffered Data (in FIFO mode). In these cases, the block at the current position is processed first, and subsequent blocks are processed in the order they were written (proceeding towards logical end of partition). The ending position is after the last block processed. For these commands, “after” will refer to the start of the next block towards the logical end of partition, and “before” will refer to the start of referenced block.

Reverse commands include Read Reverse, and Recover Buffered Data (in LIFO mode). In these cases, the block prior to the current position (towards the beginning of partition) is processed first, and subsequent blocks are processed against the order in which they were written (proceeding towards logical beginning of partition). The ending position is at the beginning of the last block processed. For these commands, “after” will refer to the start of the referenced block, and “before” will refer to the start of the next block towards the logical beginning of partition (0).

To illustrate this, from location 'N', a Read operation will return block 'N', and be positioned at 'N+1' (“after” N). From location 'M', a Read Reverse will return block 'M-1', and be positioned at 'M-1' (“after” M-1). This means that such a loop of Reads, then Read Reverses will continuously return the same block.

A successful command with a Fixed bit of 1b transfers the requested Transfer Length, times the current block length in bytes to the initiator. A successful command with a Fixed bit of 0b transfers the requested Transfer Length in bytes to the initiator. Upon completion, the logical position is “after” the last block transferred.

If SILI bit is 1b and the Fixed bit is 0b, the target performs one of the following actions:

a) Reports CHECK CONDITION status for an incorrect block length condition only if the overlength condition exists and the Block Length field in the mode parameter block descriptor is nonzero. The associated sense data is 0/0000 (Incorrect Length, No Sense Data).
b) Does not report CHECK CONDITION status if the only error is the underlength condition, or if the only error is the overlength condition and Block Length field of the mode parameters block descriptor is 0b. (See note 4 on page 41.)

If the SILI bit is 1b and the Fixed bit is 1b, the target terminates the command with CHECK CONDITION status with associated sense data of 5/2400 (Illegal Request, Invalid Field in CDB).

If the SILI bit is 0b and an incorrect length block is read, CHECK CONDITION status is returned and the ILI and valid bits are set to 1b in the sense data. Upon termination, the logical position is “after” the incorrect length block. If the Fixed bit is 1b, the information field is set to the requested Transfer Length, minus the actual number of blocks read (not including the incorrect length block).

If the Fixed bit is 0b, the information field is set to the requested Transfer Length, minus the actual block length in two's complement format.

If the logical unit encounters a filemark during a command, CHECK CONDITION status is returned and the filemark and valid bits are set to 1b in the sense data. The associated sense data is set to 0/0001 (No Sense, Filemark Detected). Upon termination, the logical position is “after” the filemark. If the Fixed bit is 1b, the information field is set to the requested Transfer Length, minus the actual number of blocks read (not including the filemark). If the Fixed bit is 0b the information field is set to the requested Transfer Length.

If the logical unit encounters end-of-partition during a command, CHECK CONDITION status is returned and the EOM and valid bits are set to 1b in the sense data. Associated sense data is set to 3/0002 (Medium Error, End of Partition/Medium).

If the logical unit encounters early warning and the REW bit is set to 1 in the device configuration page, CHECK CONDITION status is returned and the EOM and valid bits are set to 1b in the sense data. Associated sense data is set to D/0002 (Overflow, End-of-Partition/Medium Detected). If the Fixed bit is 1b, the information field is set to the requested Transfer Length, minus the actual number of blocks transferred. If the Fixed bit is 0b, the information field is set to the requested Transfer Length.

If the drive encounters End-of-Data (EOD) while executing this command, the command is terminated at the EOD position and CHECK CONDITION status is returned with associated sense data of 8/0005 (Blank Check, End-of-Data Detected). If the next motion command is another Read command (beyond EOD), and crossing EOD is allowed (See "MP 25h: Read/Write Control" on page 427.), the drive accepts the command and attempts to position beyond EOD in order to allow recovery of old data.

NOTE 3 - More than one Read command may be required to cross EOD. When EOD is successfully crossed, a CHECK condition status is returned with associated sense data set to 3/1404 (Medium Error, Block Sequence Error), which indicates that the position may have changed in a non-sequential fashion. It is recommended that Read Position be issued prior to reading data. If the drive encounters a permanent error as the result of a Read command whether or not EOD has been crossed, and crossing permanent errors is allowed (See "MP 25h: Read/Write Control" on page 427.) additional Read commands may be issued to attempt to traverse the data in error in a manner similar to reading past EOD.

If the logical unit encounters beginning-of-partition during a command, CHECK CONDITION status is returned and the EOM and valid bits are set to 1b in the sense data. Associated sense data is set to 0/0004 (No Sense, Beginning of Partition/Medium).

NOTE 4 - Because the residue information normally provided in the information field of the sense data may not be available when the SILI bit is set, use other methods to determine the actual block length. For example: include length information in the data block itself, or in the case of underlength transfers, the host adapter or device driver may return accurate transfer length information.

NOTE 5 - In the case of the Fixed bit of 1b with an overlength condition, only the position of the incorrect-length logical block can be determined from the sense data. The actual length of the incorrect block is not reported, and also cannot be derived from the transfer length (the device truncates the overlength block to match the current block length from the mode header). Other means may be used to determine the actual length (for example, backspace and read it again with Fixed bit set to 0b).

4.10.2 Interactions Summary

The following table summarizes various commands with the specified options, the encountered conditions, and the expected results.

Table 12 — Information and ILI Behavior Summary

Scenario	Fixed	SILI	Block Length	Sense Error [a]	Information [a], [b]	Flags IFE [a]	Position [a]
reportable UA	X	X	X	UA	not valid (0)	-	unchanged (no command)
reportable DCC	X	X	X	DCC	not valid (0)	-	unchanged (no command)
Read (any)	1	1	0	5/2400	transfer length	-	unchanged (no read)
	1	0	0	5/2400	transfer length	-	unchanged (no read)
Read transfer length 0	X	X	X	good	-	-	unchanged (no read)
Read (correct length(s))	X	X	X	good	-	-	after last block

Table 12 — Information and ILI Behavior Summary

Scenario	Fixed	SILI	Block Length	Sense Error[a]	Information[a],[b]	Flags IFE[a]	Position[a]
Read Underlength	0	0	X	0/0000	transfer length - block size (+)	I	after block
	0	1	X	good	-	-	after block
	1	0	non-0	0/0000	transfer length - blocks read not including incorrect block (+)	I	after incorrect block
Read Overlength	0	0	X	0/0000	transfer length - block size (-)	I	after block
	0	1	0	good	-	-	after block
	0	1	non-0	0/0000	transfer length - block size (-)	I	after block
	1	0	non-0	0/0000	transfer length - blocks read not including incorrect block (+)	I	after incorrect block
Read FM	0	X	X	0/0001	transfer length	F	after filemark
	1	0	non-0	0/0001	transfer length - blocks read not including filemark (+)	F	after filemark
Read EOD	0	X	X	8/0005	transfer length	E[g]	unchanged (at EOD)
	1	0	non-0	8/0005	transfer length - blocks read (+)	E[g]	after last block (at EOD)
Read Perm	0	X	X	perm	transfer length	-	unchanged (at perm)
	1	0	non-0	perm	transfer length - blocks read (+)	-	after last block (at perm)
Read after EOD/Perm	0	X	X	3/1404	transfer length	-	crossed EOD (position may change in non-predictable fashion, limited commands available)
	1	0	non-0	3/1404	transfer length - blocks read (+)	-	
Read (reverse) BOP	0	X	X	0/0004	transfer length	E	at BOP (0)
	1	0	non-0	0/0004	transfer length - blocks read (+)	E	at BOP (0)
Verify (any)	1	N/A	0	5/2400	transfer length	-	unchanged (no verify)
Verify transfer length 0 (VTE = 0b)	X	N/A	X	good	-	-	unchanged (no verify)
Verify (correct length(s))	X	N/A	X	good	-	-	after last block

Table 12 — Information and ILI Behavior Summary

Scenario	Fixed	SILI	Block Length	Sense Error[a]	Information[a],[b]	Flags IFE[a]	Position[a]
Verify Underlength	0	N/A	X	0/0000	transfer length - block size (+)	I	after block
	1	N/A	non-0	0/0000	transfer length - blocks verified not including incorrect block (+)	I	after incorrect block
Verify Overlength	0	N/A	X	0/0000	transfer length - block size (-)	I	after block
	1	N/A	non-0	0/0000	transfer length - blocks verified not including incorrect block (+)	I	after incorrect block
Verify FM (VBF=0b; VTE=0b)	0	N/A	X	0/0001	transfer length	F	after filemark
	1	N/A	non-0	0/0001	transfer length - blocks verified not including filemark (+)	F	after filemark
Verify EOD (VBF=0b; VTE=0b)	0	N/A	X	8/0005	transfer length	E[7]	unchanged (at EOD)
	1	N/A	non-0	8/0005	transfer length - blocks verified (+)	E[7]	after last block (at EOD)
Verify VBF=1b (correct length)	X	N/A	X	good	-	-	after last block
Verify VBF=1b (incorrect length)	0	N/A	X	good	-	-	after last block
Verify VBF=1b (incorrect length)	1	N/A	non-0	0/0000	transfer length - number of filemarks passed over (+)	I	after incorrect block
Verify VTE=1b (correct length)	X	N/A	X	good	-	E[7]	after last block (at EOD)
Verify VTE=1b (incorrect length)	0	N/A	X	good	-	E[7]	after last block (at EOD)
Verify VTE=1b (incorrect length)	1	N/A	non-0	0/0000	-	I	after incorrect block
Verify Perm	0	N/A	X	perm	transfer length	-	unchanged (at perm)
	1	N/A	non-0	perm	transfer length - blocks verified (+)	-	after last block (at perm)
Write (any)	1	-	0	5/2400	transfer length	-	unchanged (no write)
Write transfer length 0	X	-	X	-	-	-	unchanged (no write)

Table 12 — Information and ILI Behavior Summary

Scenario	Fixed	SILI	Block Length	Sense Error [a]	Information [a], [b]	Flags IFE [a]	Position [a]
Write in Early Warning	0	-	X	0/0000 0/0002	0	E	after block
	1	-	non-0	0/0000 0/0002	transfer length - blocks written (usually 1)	E	after blocks written
Write at EOM	X	-	X	D/0002	transfer length	E	unchanged (no write)
Write Perm	0	-	X	perm	transfer length or 0 (if data is in buffer)	-	after last block in buffer
	1	-	non-0	perm	transfer length - blocks transferred into buffer	-	after last block in buffer
Write after Perm	X	-	X	3/3100	transfer length	-	unchanged (no write)
Locate (target after EOD)	encountered EOD			8/0005	not valid (0) [e]	E [g]	at EOD [e]
Locate	encountered Perm			perm	not valid (0) [e]	-	indeterminate (unchanged or at perm) [e]
Space blocks	encountered FM			0/0001	Count - blocks traversed [c]	F	after FM
	encountered EOD			8/0005		E [f]	at EOD
	encountered BOP			0/0004		E	at BOP (0)
	encountered perm			perm		-	indeterminate (unchanged or at perm) [c]
Space filemarks	encountered EOD			8/0005	Count - FMs traversed [d]	E [f]	at EOD [d]
	encountered BOP			0/0004		E	at BOP (0)
	encountered perm			perm		-	indeterminate (unchanged or at perm) [d]

Table 12 — Information and ILI Behavior Summary

Scenario	Fixed	SILI	Block Length	Sense Error[a]	Information[a],[b]	Flags IFE[a]	Position[a]
Space sequential filemarks	encountered EOD			8/0005	Count - sequential FMs traversed immediately prior to ending position[e]	E[f]	at EOD[e]
	encountered BOP			0/0004		E	at BOP (0)
	encountered perm			perm		-	indeterminate (unchanged or at perm)[e]
Space EOD	encountered EOD			good	-[e]	-	at EOD[e]
	encountered perm			perm	not valid (0)[e]	-	indeterminate (unchanged or at perm)[e]

Legend:

Flags:

I ILI bit #/#### CC, sense of Sense Key/ASC ASCQ
E EOM bit perm CC, sense as per perm
F Filemark bit good No CC (no sense)

\- None set
\- Not applicable

Notes:

[a] These fields are outputs (results) from the scenario operation.
[b] Partial blocks are not considered read, written or traversed.
[c] Information field accurately reflects the ending position.
[d] Information field accurately reflects the ending position but it is not in units of logical blocks, so additional means of determining absolute location, such as Read Position, must be used.
[e] Information field does not accurately reflect the ending position, another means of determining absolute location, such as Read Position, must be used.
[f] The EOM bit is set only if the current position is in the early warning region or if the end of partition is encountered.
[g] The EOM bit is only set if end of partition is encountered (this condition should never occur), so EOM should not be set in this case. The standard specifies that EOM bit shall be set only if the current position is in the early warning region or if the end of partition is encountered.

4.11 Cleaning the Drive in a Library

In a library, the drive is automatically cleaned.

When the drive determines that either maintenance cleaning is required, or that the SARS thresholds have been reached, a message is sent to the library (via the drive-library interface) to request cleaning. This occurs when the Cleaning message is normally sent to the CE service panel. The library schedules the mounting of the cleaning cartridge. Thus, the host operating system and application are freed of any responsibility to facilitate the cleaning.

4.12 Drive Cleaning Indicators

For stand-alone drive models, automatic cleaning of the drive is not possible. For library models, automatic cleaning of the drives by the library may be disabled (although it is not recommended). For either case, cleaning of the drives must be managed by the host application or manually, by the operator.

NOTE 6 - Failure to clean a drive may result in data loss.

This section describes how cleaning indicators are presented from the drive. The cleaning indicators may be presented even with automatic cleaning enabled in a library environment. The cleaning indicators can be presented through the following:

a) Panel Cleaning Indication (see 4.12.1)
b) Host Interface - Dynamic Cleaning Indicators (see 4.12.2)
c) Host Interface - Static Cleaning Indicator (Sense Data Byte 70) (see 4.12.3)

4.12.1 Panel Cleaning Indication

A CLEAN message is displayed on the message display and the CE service panel when cleaning with a cleaning cartridge is required. For additional details, see the Operator Guide for this product.

4.12.2 Host Interface - Dynamic Cleaning Indicators

Dynamic cleaning indicators that are sent across the host interface include:

a) ASC/ASCQ codes related to cleaning in table 297. Cleaning Indicators reported with Sense Key 1 may only be reported in certain situations, see "MP 01h: Read-Write Error Recovery" on page 383.

Table 13 — ASC/ASCQ Codes Related to Cleaning

Code Description	Sense Key	ASC ASCQ
Drive Needs Cleaning	1	00 17
Drive Has Been Cleaned	1	83 83
Cleaning in Progress (cleaner cartridge)	2	30 03
Expired Cleaner	3	30 00
Failure to load (recognized cleaner cart)	3	30 02
Incompatible cleaner	3	30 00
Cleaning Failure	4	44 00
Cleaning failed	4	5300
Drive Has Been Cleaned (CU mode)	6	82 83

b) TapeAlert codes related to cleaning (see 6.4.11)
c) Service Information Message (SIM) bytes 20-21 of Log Page 31h (SIM availability is shown in sense data):

Value (ASCII)	**Description**
'55'	Drive Needs Cleaning. Load Cleaning Cartridge
'57'	Drive Has Been Cleaned

NOTE 7 - If the device driver shields the application from dynamic notifications, the information is usually available from the system error log.

4.12.3 Host Interface - Static Cleaning Indicator (Sense Data Byte 70)

The bit significance of sense data byte 70 follows:

Bit	Description
7	Set to 1b "Cleaning Required: Normal Maintenance” when cleaning is required because of the normal preventive maintenance guideline (determined by the device by read/write usage). Reset to 0b when the cleaning cartridge is loaded.
6	Set to 1b "Cleaning Required: Threshold Reached**”** when cleaning is required based on other internal threshold criteria. Reset to 0b when the cleaning cartridge is loaded.

4.13 Error Information

4.13.1 Sense Data

For a description of Sense data, see Sense Data Format.

4.13.2 Sense Data Management

Sense data returned by the device contains one of two types of errors. These errors are:

Type	Description
Current	The error condition associated with the command that is currently being processed (i.e., SCSI Status for the currently processing command is the status being returned); and
Deferred	The error condition resulting from a command that has been reported as GOOD, but has generated sense data after being reported. This may be a command with the Immediate bit set or may be a buffered write.

Sense data returned is described by the Sense Key (i.e., bits 3-0 of byte 2 of Sense data). Commands that terminate in an error generate Sense data and set the Sense Key depending on the the specifics of the error. Table 8. Supported Common SCSI commands, indicates which commands are allowed to be processed in the presence of specific error conditions and which return an error.

This device communicates on transports that use the autosense protocol. This means that any Sense data generated for return to a command is returned with the SCSI status. Once a particular set of sense data has been returned, that sense data is cleared and a REQUEST SENSE command is not required to be issued to collect the Sense data. Any other sense data that is still pending may still cause CHECK CONDITION status for subsequent commands. When a REQUEST SENSE command is received, typically the only Sense data available will be the default Sense data. While it is possible that a Deferred error may have generated Sense data or that a Unit Attention (see 4.13.4) has been established since the status to the last command, Sense data is not likely to exist.

4.13.3 Deferred Check Condition (DCC)

Deferred errors are generated by processing that occurs when that process is not attached to the currently processing command (see 4.13.2). Deferred errors are reported as sense data to a deferred check condition (DCC) eligible command (i.e., DCC column of table 37 is set to ‘Y’).

In the case of a deferred write error if buffered mode 1h is selected and a DCC eligible command is received, then the error is reported to the SCSI initiator device (i.e., I_T nexus) that has deferred error affinity.

If the drive receives a deferred error affinity command (i.e., DEA column of table 37 is set to ‘Y’), then the drive performs actions in the following order:

1) performs initial checking (e.g., Reservation Conflict, all pending Unit Attentions, and all pending errors to be reported to this I_T nexus) and reports these conditions, if any;
2) if none of the above conditions are reported, then all pending deferred errors are migrated to the I_T nexus through which this command was received;
3) the deferred error affinity is set to this I_T nexus;
4) if the command is DCC eligible, then pending deferred errors, if any, are reported; and
5) if no deferred errors were reported process the command.

4.13.4 Unit Attention Conditions

The drive generates a Unit Attention condition under the following circumstances:

a) Reset condition (for example, power-on, SCSI reset, bus device reset);
b) Tape Loaded condition (for example, media inserted, LOAD command from another initiator);
c) Mode parameters changed by another initiator; and
d) Drive firmware has been upgraded.

The drive only maintains one instance of each type of Unit Attention condition at any one time for any one initiator. If a subsequent Unit Attention condition of the same type is generated, it replaces the existing one. Unit Attentions are returned in priority order. The priorities are in the order listed above, with a reset being highest priority and a firmware upgrade being lowest priority.

4.13.5 Persistent Errors

When errors occur that prevent tape operation, they are reported persistently until the problem is cleared. For medium-related errors (usually reported with a Sense Key of 3), the error is reported until the cartridge is successfully unloaded. For hardware-related errors (usually reported with a Sense Key of 4), the error is reported until the drive successfully performs a power-on self test. These persistent errors are only reported on those commands that are eligible for deferred Check Condition reporting (see Table 4). The error may or may not be reported as Deferred.

4.14 Environmental Conditions Thresholding

This device reports current and historical limits of temperature and relative humidity in the Environmental Reporting log page (see 6.4.8). In addition, this device monitors temperature and humidity and takes protective actions such as:

a) warning the application/library through TapeAlerts;
b) fencing the drive from any medium access commands; and
c) ejecting the volume and fencing load commands.

These protective actions are centered around a set of thresholds with built in hysteresis. When a threshold is met, the drive performs the indicated action. When the environmental condition (i.e., temperature or humidity) changes such that the threshold is no longer met, there is hysteresis built in prior to the resetting of that threshold. This hysteresis is designed to provide a stable return to operation and may be substantially different than the value of the threshold. These thresholds and reset points are managed by the drive.

Figure 7 shows an example of how the thresholds are set. This is only an example to describe the concept and not intended to show the relative difference in values of the various thresholds. This example describes the progression through the various thresholds.

1) The Normal Operation Condition range of temperature is where the drive is designed to operate. The drive operates normally in this range;
2) When the temperature rises towards an unsafe value, the *TapeAlert Asserted* threshold is crossed and the Drive Temperature TapeAlert (i.e., 0024h) trigger is activated. The drive asserts the Drive Temperature TapeAlert;
3) As the temperature continues to rise it passes through the *Fence Commands* threshold. When this occurs, the *Fence Commands* trigger is activated and the drive disallows commands determined to be unsafe in this condition (e.g., disallows medium access commands). These commands are rejected with a CHECK CONDITION with the sense key set to ABORTED COMMAND and the additional sense code set to WARNING - SPECIFIED TEMPERATURE EXCEEDED (i.e., B/0B01h);
4) As the temperature continues to rise, it passes through the *Eject Volume* threshold. When this occurs, the *Eject Volume* trigger is activated and the drive ejects the volume. An attempt to load the volume is

rejected with a CHECK CONDITION with the sense key set to NOT READY and the additional sense code set to WARNING - SPECIFIED TEMPERATURE EXCEEDED (i.e., 2/0B01h);

5) If the temperature decreases, it eventually passes through the temperatures of each of the reset thresholds and each of the threshold triggers are reset.

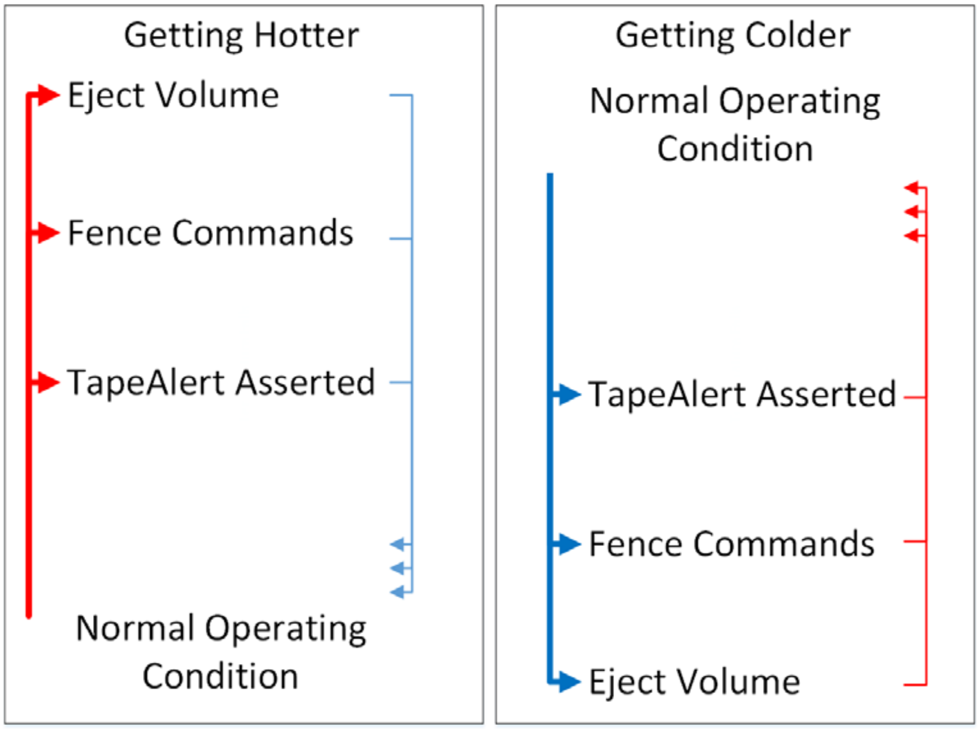


Figure 7 — Example of temperature thresholds

4.15 WORM Behavior

This device supports specially formatted WORM media which enables tamper-resistant WORM write and append methods. The drive microcode leverages this support by providing an interface and control mechanisms which allow an application or system to manage this as needed. The control and status mechanisms for this can be found primarily in mode page 23h (see 6.6.19) and mode page 24h (see 6.6.20). A new medium identifier is specified, as well as a unique Volume Label Cartridge Type identifier (character 2), reported in sense data. Generally, the WORM function may be used without application awareness, however if the write/append behavior is violated (an illegal overwrite is attempted) then a write perm will occur.

To further data performance and provide an audit trail, a unique cartridge identifier is reported in mode page 23h which may be tracked by a host system to insure that the intended cartridge is being processed. This identifier is guaranteed unique and unalterable, and is imbedded into the servo format of the media itself. This identifier may be used to differentiate whether an original tape or a copy is being used. The host should determine the appropriate tracking methodology for the system design. For the most tamper resistant and detection, the host might query this identifier from the drive and add it to header and trailer records within the host written data. This may then be compared during the read process. Provisions may need to be made to allow intentionally copied media to be processed in such a system.

4.15.1 WORM Write/Append General Behavior

The WORM feature was implemented in a manner which is designed for maximum application compatibility, while meeting the necessary data permanence requirements. To accomplish this, methods to identify common label and trailer constructs have been implemented, and a new ASC/ASCQ (7/5001, Data Protect, WRITE APPEND POSITION ERROR) has been added to indicate that writing is not allowed at the current location. There are several general rules surrounding WORM writing. Mode specific details can be found in following section(s).

a) Appending is always allowed at the EOD location for non-locked medium.
b) Writing is never allowed at any location prior to the start of an identified non-buffered label or trailer construct.
c) Writing may not be allowed after any construct which is followed by data which has been explicitly synchronized. This includes: Write Filemarks 0, Write Filemarks non-immediate, Unload. (depending on WORM mode).
d) Writing may not be allowed after any construct which is followed by data which has been implicitly synchronized by host or operator action. This includes: Space, Locate, Rewind, Load/Unload, etc. (depending on WORM mode).
e) Writing may not be allowed after any construct which is followed by data which has been implicitly synchronized by the drive. This includes: Time-based buffer flush, buffer full conditions, etc. (depending on WORM mode).
f) Writing may be allowed up to and including an overwrite of all or part of the last construct when a construct is either incomplete, or is not followed by additional data or filemarks (depending on WORM mode).
g) Writing may be allowed up to and including an overwrite of the last construct if a permanent write error occurs prior to or during the first host involved synchronize of the data following that construct (depending on WORM mode).
h) Buffered constructs (those where all filemarks are written immediate, and no other synchronizing events occur) may have a longer overwrite horizon in the permanent error cases. This may extend to a location earlier than constructs which were fully written to media. However, on any normal host invoked synchronizing event, the appropriate overwrite location will be locked (depending on WORM mode).
i) Writing may not be allowed at points other than the last complete logical block when a tape is loaded and it's ending disposition can not be definitively determined. This can occur in loss of power and other severe error conditions.

Many common label, header, trailer, and end of volume constructs are recognized by the device. In the following lists:

a) BOP indicates that this sequence must start at the logical beginning of a partition (logical block number 0),
b) <FM> indicates a filemark,
c) <?> indicates any character,
d) <?..?> indicates one or more records of the same type with any character in any order,
e) <#> indicates a sequential numeric character,
f) <n..#> indicates one or more records of the same type with sequential numerical character(s) starting with n,
g) Items listed in quotes indicate that a record must start with that string, in either ASCII or EBCDIC (note that all records in the same construct must be either ASCII or EBCDIC, intermix will terminate the construct). While most common label constructs consist of records of 80 bytes (or more), this is not enforced by the identification process. A minimum record size of 4 bytes is required for identification. There is no further limit for maximum record size for identification beyond the maximum supported device logical block length.
h) In the case of “VOL1” at least six more characters indicating the VOLSER in the same encoding are expected (VOL1 records smaller than 10 bytes are not recognized).
i) The identification process treats synchronizes (Write Filemarks 0 or implicit) as transparent.
j) Repositions are allowed and handled during construct recognition, however, constructs are only recognized within the current mount. If media is unloaded with a partial construct or construct not followed by user data, then the next series of writes must overwrite the existing construct with a new complete construct, otherwise the partial construct on media is treated as user data and subject to normal overwrite rules. In this case, the leading filemark in the general forms below is not required for construct indentification.

The constructs which the device recognizes follow the Labels shown in 4.15.1.1.

4.15.1.1 Labels

The general form (items listed in brackets [] are optional) is:

BOP<“VOL1<volser>”>[<“VOL<2..#>”>][<“UVL<1..#>”>][<“HDR<1..#>”>[<“UHL<?..?>”>]]<FM>[<FM>]

The following special form is also recognized:

BOP<“VOL1”<volser>>

Some examples of the general form include (but are not limited to):

BOP<“VOL1”<volser>><FM>
BOP<“VOL1”<volser>><FM><FM>
BOP<“VOL1”<volser>><“VOL2”><“VOL3”><“HDR1”><FM>
BOP<“VOL1”<volser>><“HDR1”><“HDR2”><FM><FM>
BOP<“VOL1”<volser>><“HDR1”><“HDR2”><“HDR3”><“UHL6”><“UHLA”><FM>

4.15.1.1.1 Headers

The general form (items listed in brackets [] are optional) is:

<FM><“HDR<1..#>”>[<“UHL<?..?>”>]<FM>[<FM>]

Some examples include (but are not limited to):

<FM><“HDR1”><FM>
<FM><“HDR1”><“HDR2”><FM><FM>
<FM><“HDR1”><“UHLC”><“UHL4”><FM><FM>

4.15.1.1.2 Trailers

The general from (items listed in brackets [] are optional) is:

<FM>[<“EOF<1..#>”>[<“UTL<?..?>][<“OIB<?..?>”>]<FM>][<FM>]

Some examples include (but are not limited to):

<FM>
<FM><FM>
<FM><“EOF1”><FM><FM>
<FM><“EOF1”><“EOF2”><FM>
<FM><“EOF1”><“EOF2”><“EOF3”><“UTL2”><“UTL1”><FM>
<FM><“EOF1”><“OIBD”><“OIB6”><“OIBZ”><FM>
<FM><“EOF1”><“EOF2”><“UTL_”><“UTL5”><“OIB-”><“OIB$”><FM><FM>

4.15.1.1.3 End Of Volume

The END OF VOLUME constructs would normally be written shortly after early warning indicators are raised and will never be overwritten.

The general form (items listed in brackets [] are optional) is:

<FM><“EOV<1..#>”>[<“UTL<?..?>][<“OIB<?..?>”>]<FM>[<FM>]

Some examples include (but are not limited to):

<FM><“EOV1”><FM>
<FM><“EOV1”><“EOV2”><FM><FM>
<FM><“EOV1”><“EOV2”><“EOV3”><“UTLX”><FM><FM>
<FM><“EOV1”><“OIBK”><“OIB3”><“OIB+”><FM><FM>
<FM><“EOV1”><“EOV2”><“UTLL”><“UTLL”><“OIBK”><“OIB3”><“OIB+”><FM><FM>

4.15.1.1.4 Example of a general form construct fully expanded

BOP<“VOL1”<volser of 123456>><“HDR<1..2>”><“UHL<1..3>”><FM><FM>

which can be expanded to:

BOP<“VOL1123456”><“HDR1”><“HDR2”><“UHL1”><“UHL2”><“UHL3”><FM><FM>

and appears on medium as a sequence where:

BOP indicates this sequence must start at LB 0
LB 0 is a record of 10 bytes or more starting with “VOL1123456”
LB 1 is a record of 4 bytes or more starting with “HDR1”
LB 2 is a record of 4 bytes or more starting with “HDR2”
LB 3 is a record of 4 bytes or more starting with “UHL1”
LB 4 is a record of 4 bytes or more starting with “UHL2”
LB 5 is a record of 4 bytes or more starting with “UHL3”
LB 6 is a filemark
LB 7 is a filemark

4.15.2 WORM mode 2h Write/Append Behavior

Refer to MP 23h: Medium Sense (see 6.6.19) to identify some of the possible operation possibilities and for overall behaviors. Specific details for this mode follow:

a) Writing is not allowed prior to any non-construct record which has been completely written to medium.
b) Overwriting of all or part of the most recent construct (label, header, or trailer) is allowed, provided that the construct is partially or fully recognized and is not followed by unrecognized records (i.e. user data).

4.15.3 WORM Device Driver Application Development/Debug Support

The WORM mode may be set for Data cartridges to enable WORM constraint reporting and other overwrite checks. This mode is active until changed or until the media is unloaded. The effects are not persistent (not stored on medium). Contact IBM for additional details.

4.16 Medium Reuse at Higher (or Lower) Densities

This device is designed to enable the reuse of different medium types and multiple densities across various device generations. Certain device models may only support a subset of densities (such as model J1A, which can only read and write at a single density), while others (such as the model E05 and E06) can read and write at multiple densities. Some models (such as the model E06) may be able to read certain formats or densities, but may be unable to write or append at those densities. Density information is available by using the Report Density Support command (see 5.2.32).

> NOTE 8 - It should be noted that an entire partition (or tape) must be recorded using the same density. This means that medium at a given density can only have data appended by a device capable of writing at that density. However, medium may be reformatted (destructively written from BOP) to any device and medium supported format.

> NOTE 9 - Mode Sense always reports the actual current format/density of the medium and does not reflect any potential (currently pending) change in density.

Media may be reused at a different density. The changing from one density to another is called reformatting and occurs during a write from logical block zero of partition 0 (i.e., BOP 0). This process is subject to the rules of Write modes (see 4.18) including append-only mode.

If a write operation is attempted at BOP 0 (i.e., logical object 0 of partition 0) of a volume, then:

a) if the volume is a single partition volume and if the density specified in the PENDING WRITE DENSITY AT BOP 0 field of mode page 25h (see 6.4.18) does not match the primary density code (see 5.2.24) of the volume, then the volume is formatted to the density specified in the PENDING WRITE DENSITY AT BOP 0 field;
b) if the volume is partitioned and the density specified in the PENDING WRITE DENSITY AT BOP 0 field of mode page 25h does not support partitioning, then the volume is unpartitioned and reformatted to the density specified in the PENDING WRITE DENSITY AT BOP 0 field and all data on the volume in all partitions is lost;
c) if the volume is partitioned and the density specified in the PENDING WRITE DENSITY AT BOP 0 field of mode page 25h matches the primary density code of the volume, then the volume is not unpartitioned and data on other partitions is not lost.

If a write operation is attempted at a location away from BOP of partition 0 (this includes BOP of a partition other than partition 0), then no reformatting occurs and if the device is capable of wrting at the current density then logical objects are written with the current density.

> NOTE 10 - If a volume is partitioned, then automatic reformatting to a higher density is not supported on that volume and explicit formatting using the FORMAT MEDIUM command must be used.

4.17 Selecting the Density/Format

The density to use for reformatting is controlled by the density code mechanism found in the Mode Select Block Descriptor (see 6.6.1.2), the Default Write Density at BOP 0 field (see 4.17.3), and the PENDING WRITE DENSITY AT BOP 0 field (see 4.17.4). All reformatting is implicit and only takes effect on commands which would otherwise invalidate or overwrite all data starting at BOP. Such commands are always issued when positioned at BOP and include: Write [non-zero transfer length], Write Filemarks [non-zero transfer length], Erase, Capacity Scaling [Mode Select], Format Medium (unsupported), etc. The density selection itself is a passive process which can be setup or changed without affecting the recorded data currently on the medium. This drives three separate mechanisms found in Mode Select (Block Descriptor and Page 25h), which are:

a) Density Code (Block Descriptor for Mode Select (6/10) (see 6.6.1.2))
b) Default Write Density at BOP 0 (MP 25h: Read/Write Control (see 6.6.21))
c) PENDING WRITE DENSITY AT BOP 0 (MP 25h: Read/Write Control (see 6.6.21))

Any of the previously listed commands which normally cause destructive writing at BOP may additionally reformat the tape to the density specified in the Pending Write Density field if that density differs from the currently recorded density. This operation does not substantively affect the execution time of the command causing the reformat. The Default Write Density field is used to set a default value into the Pending Write Density field at load time, unload time, or if the mode pages are reset to default values. Additionally, the Density Code field may be used to set the Pending Write Density. There are some special considerations which affect the precedence and interaction of the fields, as described below.

> NOTE 11 - On devices which support encryption, secondary encryption density codes may be selected but are interchangeable and identical to primary density codes. Use of any density codes has no effect on the encrypted state of medium and only reports encryption state information and cannot be used to control encryption.

4.17.1 Recommended Method for Specifying Density

The simplest and recommended method for explicitly specifying density comprises two parts. The first part is to always set the Density Code field in every mode select command to the special value 7Fh as per the SCSI standard (the same effect will be accomplished by leaving it unchanged from the value returned by Mode Sense, though this is non-standard). Secondly, the PENDING WRITE DENSITY AT BOP 0 field (byte 10) in mode page 25h may then be used for explicit density control. The specified density is used for any write operation from BOP.

4.17.2 Density Code

See also “Block Descriptor for Mode Select (6/10)” on page 375.

This is the lowest precedence method for setting density. Since this method has strong existing application usage, care has been taken to prevent accidental specification of density, or loss of desired density change by possible overwrite. Since applications are expected to issue a Mode Sense prior to a Mode Select and not all applications use the standard 7Fh value, only explicit changes in this field have any effect. This means that it is not possible to set the Pending Density to the current density of the medium by using an explicit density in this field. However, this may be accomplished by using the FFh density code. This may need to be done if automatic up formatting is not desired. Setting a value of 00h causes the density from the Default Density Code at BOP field to be used. If this field is not changed from the value returned in Mode Sense, or if the special 7Fh density code is specified, the Pending Write Density field will not be evaluated and updated according to this field.

4.17.3 Default Write Density at BOP 0

See also “MP 25h: Read/Write Control” on page 427.

This field is used to set the Pending Write density at unload or load time, or if the Density Code field is set to 00h. If this field is changed with Mode Select, the Pending Write Density field will be evaluated and updated according to this field.

4.17.4 Pending Write Density at BOP 0

See also “MP 25h: Read/Write Control” on page 427.

This is the actual field for selecting density and when read (Mode Sense) always reflects the actual density which will be used if a command which causes reformatting is issued. This value may be directly modified but the value might be altered to an absolute density code if either 00h, 7Fh, or FFh is specified.

4.18 Write modes

4.18.1 Write mode introduction

Write modes of the device entity specify the allowable behaviors for altering logical objects on a mounted volume. When the write mode rules allow altering of logical objects then the operation is processed following the write protection rules defined in SSC-4.

4.18.2 Overwrite-allowed mode

Overwrite-allowed mode is used to allow alteration of any logical object on the medium. Overwrite-allowed mode is enabled or disabled using the WRITE MODE field of the Device Configuration Extension mode page (see 6.6.12). This mode enables device server behaviors. The overwrite-allowed mode does not modify the volume. When the volume is removed from the device no indication of whether overwrite-allowed mode is enabled or disabled is carried with the volume.

When overwrite-allowed mode is enabled in the drive, then any command that would result in a write shall be processed normally. If the mounted volume is a WORM volume, then a write type command shall be processed following the WORM rules.

4.18.3 Append-only mode (beginning with E07)

Append-only mode is also referred to as Data-safe mode. Append-only mode is used to protect data from being accidently overwritten. Sometimes, due to errors in the configuration of the environment an application client attempts to rewind a drive that it is not transferring data to. Without append-only mode, an application client that is not writing to the volume may cause, by rewinding that volume, a scenario where the writing application client overwrites the data. This may occur if the drive has not been reserved or if a competing application has been installed on the same server (e.g., a monitoring application). With append-only mode enabled, the medium is protected against an overwrite because the drive rejects an attempt to write data anywhere except at an append point.

Append-only mode is enabled or disabled using the WRITE MODE field of the Device Configuration Extension mode page (see 6.6.12). Append-only mode is only allowed to be enabled if a volume is not mounted or if the volume is located at BOP 0. Append-only mode is only allowed to be disabled if there is no volume mounted in the drive. This mode is set in the device server to enable device server behaviors. The append-only mode does not modify the volume. When the volume is removed from the device it behaves as a normal volume.

When append-only mode is enabled in the drive, then any command that would result in a write to a location that is not an append point shall be rejected with a CHECK CONDITION status and the Sense Key set to DATA PROTECT with the additional sense code set to OPERATOR SELECTED WRITE PROTECT (7h / 5A02h) and TapeAlert 09h shall be set. An append point is:

a) the logical position zero if there are no logical objects beyond BOP;
b) the current logical position if:
   A) the current logical position is at BOP; and
   B) there are only filemarks between the current logical position and EOD;
c) the current logical position if:
   A) the current logical position is between BOP and EOD;
   B) there are only filemarks from the current logical position to EOD; and
   C) there is at least one filemark immediately before the current logical position;

   or

d) the current logical position if the current logical position is at EOD.

The device server maintains an allow_overwrite variable. The allow_overwrite variable defines what operation is currently allowed when in append-only mode. The allow_overwrite variable values are defined in table 14.

Table 14 — ALLOW_OVERWRITE variable definition

Name	Description
Disabled	A write type operation at a position that is not an append point is not allowed.
Current Position	A write type operation is allowed at the position specified by the allow_overwrite_position variable. Note that reformatting as defined in Medium Reuse at Higher (or Lower) Densities (see 4.10) is considered a write operation.
Format	An operation that modifies the format of the medium is allowed (e.g., FORMAT MEDIUM command, Capacity Scaling operation in Mode Page 23h, etc.)

The allow_overwrite_position variable specifies the position (i.e., partition and logical object identifier) at which a write to a position that is not an append point is allowed.

Append-only mode is a function of the device server and is not a function of the volume. Append-only mode may be used when accessing Data Volumes or WORM volumes. An application client may overwrite data by using a special command called the ALLOW OVERWRITE command (see 5.2.1). The ALLOW OVERWRITE command specifies the logical position where the overwrite is to occur. After succesfully processing an ALLOW OVERWRITE command, a write type command at the specified position is processed normally. If the position of the medium is changed or the volume is unmounted, then the device server shall set the allow_overwrite variable to Disabled (i.e., 0h) and the allow_overwrite_position variable to invalid. The ALLOW OVERWRITE command requires the partition number and the logical position to be passed in the CDB. If the position information passed in the ALLOW OVERWRITE command does not specify the current position of the medium, then the command is terminated with CHECK CONDITION status with the sense key set to ILLEGAL REQUEST and the additional sense code set to SEQUENTIAL POSITIONING ERROR. If there is no volume loaded and the device server processes an ALLOW OVERWRITE command, then the command is terminated with CHECK CONDITION status with sense key set to NOT READY.

An ALLOW OVERWRITE command that returns GOOD status:

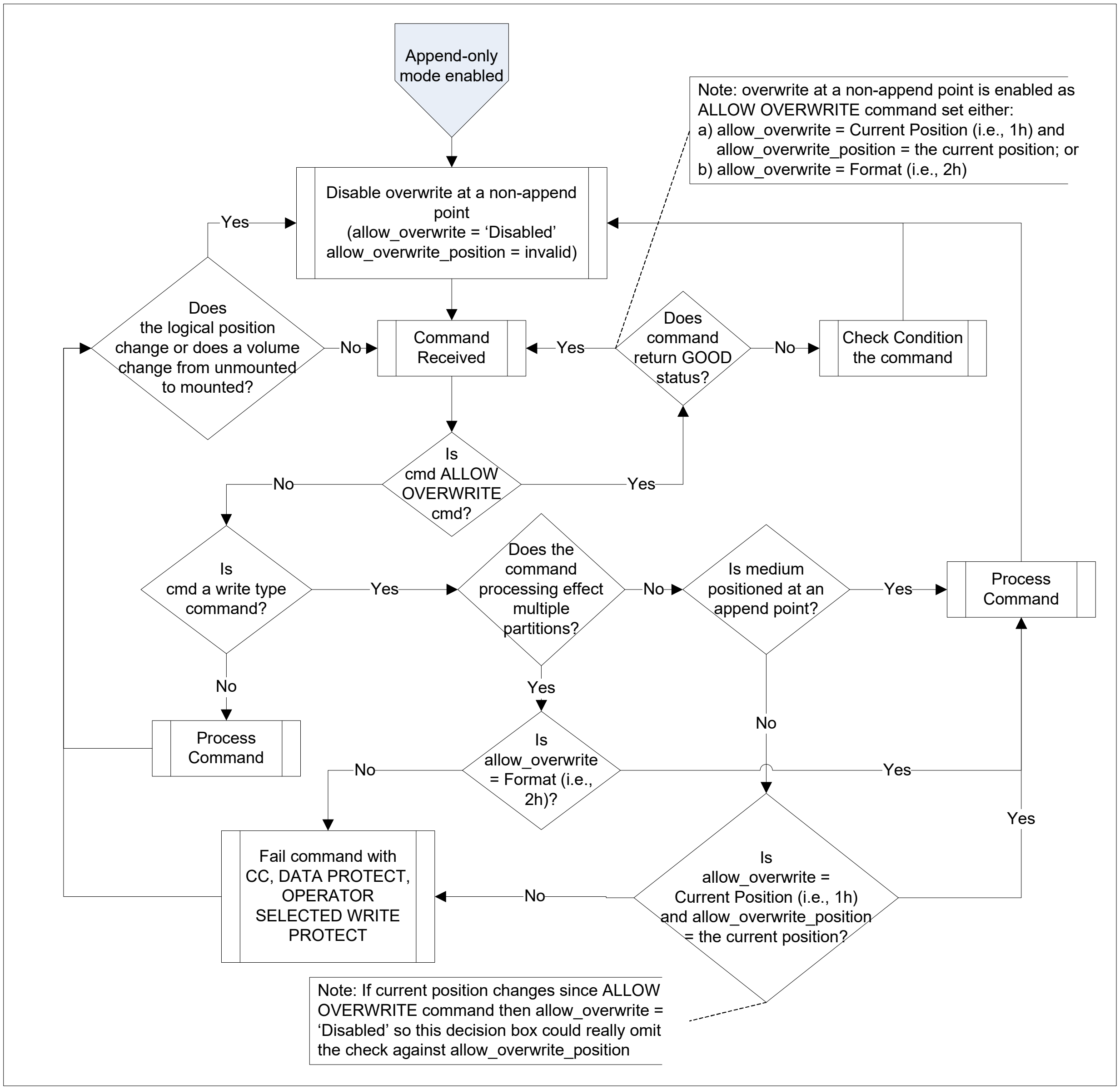
a) sets the allow_overwrite variable to the value in the ALLOW OVERWRITE field of the ALLOW OVERWRITE command; and
b) sets the allow_overwrite_position variable to the current position.

An ALLOW OVERWRITE command that returns a CHECK CONDITION:

a) sets the allow_overwrite variable to Disabled (i.e., 0h); and
b) sets the allow_overwrite_position to invalid.

If append-only mode is enabled, the mounted volume is a WORM volume, and the allow_overwrite variable is not set to Disabled, then a write type command is processed following the WORM rules. Figure 8 shows a representative flowchart of append-only mode behavior.

Figure 8 — Append-only mode flowchart



If the ALLOW OVERWRITE command is received by the device server and append-only mode is not enabled, the command is rejected with a CHECK CONDITION status and the Sense Key is set to ILLEGAL REQUEST and the additional sense code is set to ILLEGAL COMMAND WHEN NOT IN APPEND-ONLY MODE.

When in append-only mode the allow_overwrite variable shall be set to Disabled (i.e., 0h) and the allow_overwrite_position variable shall be set to invalid if:

a) the WRITE MODE field of the Device Configuration Extension mode page changes to a value of 01h (i.e., The write-type operation shall only allow appends as specified for the append-only mode in 4.18.3);
b) a change in logical position occurs;
c) a volume changes state from unmounted to mounted;
d) the CDB of a write type command is validated and the write processing begins; or
e) an ALLOW OVERWRITE command returns a CHECK CONDITION.

4.19 Volume partitioning

4.19.1 Volume partitioning overview

Starting with TS1140 (i.e., 3592 E07) volume partitioning is supported by the device on certain media types. A volume is recorded in the same format for the entire volume as indicated by the primary density code (see 5.2.32) but each partition may have different secondary density codes. An entire partition is recorded in the same format as indicated by the secondary density code.

The drive supports three different types of partitioning:

a) wrap-wise partitioning not using FastSync (see 4.19.2);
b) wrap-wise partitioning using FastSync (see 4.19.2.2); and
c) longitudinal partitioning (see 4.19.3).

This clause and its subclauses describe partitioning and its relationship to:

a) media types (see 4.19.4);
b) reformatting (see 4.19.6); and
c) encryption (see 4.19.7).

4.19.2 Wrap-wise Partitioning

Wrap-wise partitioning uses the full length of the medium for each partition as shown logically in figure 9.

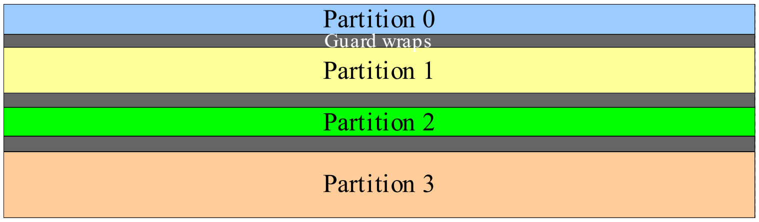


Figure 9 — Wrap-wise partitioning

Wrap-wise partitioning generally uses a minimum of two wraps and generally has a guard wrap of two wraps between each partition. This may reduce the usable capacity by as much as 3% per partition boundary.

When using wrap-wise partitioning a maximum of four partitions with any number of partitions between one and four inclusive are supported. See Partitioning and WORM volumes (see 4.19.5 on page 75) for restrictions that apply to WORM volumes.

There are two partitioning types that use wrap-wise partitioning. The two partitioning types trade-off between maximizing capacity by not using FastSync (i.e., Virtual Backhitch) and improving small file performance by using FastSync. The two partitioning types are:

a) Wrap-wise Partitioning with no FastSync (see 4.19.2.1 on page 63); and
b) Wrap-wise Partitioning with FastSync (see 4.19.2.2 on page 68).

4.19.2.1 Wrap-wise Partitioning with no FastSync

This type of partitioning maximizes capacity by sacrificing the small file performance improvement that would have been available with FastSync (i.e., Virtual Backhitch).

Table 15, and table 16 show the partition sizes that result from a MODE SELECT of the Medium partition mode page (see 6.6.13) with the indicated field settings.

Table 15 — Partition sizes for wrap-wise partitioning with no FastSync (selection fields)

Ref[a]	FDP	SDP	IDP	ADDITIONAL PARTITIONS DEFINED[b]	PARTITION SIZE			
					(FIRST)	(SECOND)	(THIRD)	(LAST)
row1	1	0	0	X	X	X	X	X
row2	0	1	0	00h				
row3				01h				
row4				02h				
row5				03h				
row6	0	0	1	00h	FFFFh	0	0	0
row7					s[c]			
row8				01h	s	FFFFh		
row9					FFFFh	t		
row10					s[c]	t[c]		
row11				02h	s	t	FFFFh	
row12					s	FFFFh	u	
row13					FFFFh	t	u	
row14					s[c]	t[c]	u[c]	
row15				03h	s	t	u	FFFFh
row16					s	t	FFFFh	v
row17					s	FFFFh	u	v
row18					FFFFh	t	u	v
row19					s[c]	t[c]	u[c]	v[c]
others	All other combinations							

[a] The Ref column is the reference that ties the rows in this table to the corresponding rows in table 16.

[b] When more than one partition is defined there may be overhead that results in a loss of capacity

[c] Value must be exact partition size allowed and when summed with other values in the row equal full capacity. This permits a MODE SENSE followed by a MODE SELECT with no change. It is highly recommended that this method only be used in the case where the MODE SELECT data is a return of the MODE SENSE data.

Table 16 — Partition sizes for wrap-wise partitioning with no FastSync (resultant sizes) (part 1 of 2)

Ref[a]	Partition 0[b]	Partition 1[b]	Partition 2[b]	Partition 3[b]
row1	$s=K*n$; where $n=N_2$	$t=K$	-	-
row2	C_{MAX}	-		
row3	$s=K*n$; where n=integer of $\{(N_2+1)/2\}$	$t=K*m$; where $m=N_2+1-n$	-	
row4	$s=K*n$; where n=integer of $\{(N_3+2)/3\}$	$t=K*m$; where m=integer of $\{(N_3+2)/3\}$	$s=K*u$; where $u=N_3+2-n-m$	-
row5	$s=K*n$; where n=integer of $\{(N_4+3)/4\}$	$t=K*m$; where m=integer of $\{(N_4+3)/4\}$	$s=K*u$; where u=integer of $\{(N_4+3)/4\}$	$s=K*v$; where $v=N_4+3-n-m-u$
row6	C_{MAX}	-		
row7	C_{MAX}			
row8	$s=K*n$; where $1<=n<=N_2$	C_{MAX} - overhead-(partition size 0)	-	
row9	C_{MAX} - overhead-(partition size 1)	$t=K*m$; where $1<=m<=N_2$		
row10	$s=K*n$; where $1<=n<=N_2$ and $n+m=N_2+1$	$t=K*m$; where $1<=m<=N_2$ and $n+m=N_2+1$		
row11	$s=K*n$; where $1<=n<=N_3$	$t=K*m$; where $1<=m<=N_3$	C_{MAX} - overhead-(partition size 0)-(partition size 1)	-
row12		C_{MAX} - overhead-(partition size 0)-(partition size 1)	$u=K*p$; where $1<=p<=N_3$	
row13	C_{MAX} - overhead-(partition size 1)-(partition size 2)	$t=K*m$; where $1<=m<=N_3$		
row14	$s=K*n$; where $1<=n<=N_3$ and $n+m+p=N_3+2$	$t=K*m$; where $1<=m<=N_3$ and $n+m+p=N_3+2$	$u=K*p$; where $1<=p<=N_3$ and $n+m+p=N_3+2$	

[a] The values in the Ref column refer back to the associated row in table 15.

[b] The values for C_{MAX}, K, N_2, N_3, N_4, and the sum of existing partitions are specified in table 17.

Table 16 — Partition sizes for wrap-wise partitioning with no FastSync (resultant sizes) (part 2 of 2)

Ref[a]	**Partition 0**[b]	**Partition 1**[b]	**Partition 2**[b]	**Partition 3**[b]
row15	s=K*n; where 1<=n<=N_4	t=K*m; where 1<=m<=N_4	u=K*p; where 1<=p<=N_4	C_{MAX} - overhead- (partition size 0)- (partition size 1)- (partition size 2)
row16			C_{MAX} - overhead- (partition size 0)- (partition size 1)- (partition size 2)	v=K*q; where 1<=q<=N_4
row17		C_{MAX} - overhead- (partition size 0)- (partition size 2)- (partition size 3)	u=K*p; where 1<=p<=N_4	
row18	C_{MAX} - overhead- (partition size 1)- (partition size 2)- (partition size 3)	t=K*m; where 1<=m<=N_4		
row19	s=K*n; where 1<=n<=N_4 and n+m+p+q=N_4+3	t=K*m; where 1<=m<=N_4 and n+m+p+q=N_4+3	u=K*p; where 1<=p<=N_4 and n+m+p+q=N_4+3	v=K*q; where 1<=q<=N_4 and n+m+p+q=N_4+3
others	Check Condition, Illegal Request, Invalid Field in Parameter Data			

[a] The values in the Ref column refer back to the associated row in table 15.

[b] The values for C_{MAX}, K, N_2, N_3, N_4, and the sum of existing partitions are specified in table 17.

Table 17 — Partition values for wrap-wise partitioning with no FastSync by density 54h and media type

Parameter in table 22	Primary Density Code = 54h		
	Media Type		
	JB	JC/JY [b]	JK
C_{MAX}	1.6 TB	4.0 TB	500 GB
K^1	57.142 GB	100.000 GB	12.500 GB
N_2	26	37	37
N_3	24	34	34
N_4	22	31	31
Sum of all partitions [a] s+t+[u]+[v]	s+t=1 542.857 GB s+t+u=1 485.714 GB s+t+u+v=1 428.571 GB	s+t=3 800.000 GB s+t+u=3 600.000 GB s+t+u+v=3 400.000 GB	s+t=475.000 GB s+t+u=450.000 GB s+t+u+v=425.000 GB

[a] The precision of capacity values able to be expressed is limited by the 2-byte PARTITION SIZE field and the value of the PARTITION UNITS field of the Medium Partition mode page (see 6.6.13). Actual size on medium is not limited by the precision of the fields in the mode page.

[b] WORM volumes (i.e., JY) support a maximum of two partitions

Table 18 — Partition values for wrap-wise partitioning with no FastSync by density 55h and media type

Parameter in table 22	Primary Density Code = 55h			
	Media Type			
	JC/JY [b]	JK	JD/JZ [b]	JL
C_{MAX}	7.0 TB	900 GB	10.0 TB	2.0 TB
K^1	97.222 GB	12.500 GB	125.000 GB	25.000 GB
N_2	70	70	78	78
N_3	68	68	76	76
N_4	66	66	74	74
Sum of all partitions [a] s+t+[u]+[v]	s+t= 6902.777 GB s+t+u= 6805.555 GB s+t+u+v= 6708.333 GB	s+t= 887.500 GB s+t+u= 875.000 GB s+t+u+v= 862.500 GB	s+t= 9875.000 GB s+t+u= 9750.000 GB s+t+u+v= 9625.000 GB	s+t= 1975.000 GB s+t+u= 1950.000 GB s+t+u+v= 1925.000 GB

[a] The precision of capacity values able to be expressed is limited by the 2-byte PARTITION SIZE field and the value of the PARTITION UNITS field of the Medium Partition mode page (see 6.6.13). Actual size on medium is not limited by the precision of the fields in the mode page.

[b] WORM volumes (i.e., JY and JZ) support a maximum of two partitions

Table 19 — Partition values for wrap-wise partitioning with no FastSync by density 56h and media type

Parameter in table 22	Primary Density Code = 56h	
	Media Type	
	JD/JZ [b]	JL
C_{MAX}	15.0 TB	3.0 TB
K^1	125.000 GB	25.000 GB
N_2	117	117
N_3	114	114
N_4	111	111
Sum of all partitions [a] s+t+[u]+[v]	s+t= 14 750.000 GB s+t+u= 14 500.000 GB s+t+u+v= 14 250.000 GB	s+t= 2 950.000 GB s+t+u= 2 900.000 GB s+t+u+v= 2 850.000 GB

[a] The precision of capacity values able to be expressed is limited by the 2-byte PARTITION SIZE field and the value of the PARTITION UNITS field of the Medium Partition mode page (see 6.6.13). Actual size on medium is not limited by the precision of the fields in the mode page.

[b] WORM volumes (i.e., JZ) support a maximum of two partitions

Table 20 — Partition values for wrap-wise partitioning with no FastSync by density 57h and media type

Parameter in table 22	Primary Density Code = 57h [b]	
	Media Type	
	JE	JM
C_{MAX}	20.0 TB	5.0 TB
K^1	147.058 GB	36.764 GB
N_2	133	133
N_3	130	130
N_4	127	127
Sum of all partitions [a] s+t+[u]+[v]	s+t= 19 705.882 GB s+t+u= 19 411.764 GB s+t+u+v= 19 117.647 GB	s+t= 4 926.470 GB s+t+u= 4 852.941 GB s+t+u+v= 4 779.411 GB

[a] The precision of capacity values able to be expressed is limited by the 2-byte PARTITION SIZE field and the value of the PARTITION UNITS field of the Medium Partition mode page (see 6.6.13). Actual size on medium is not limited by the precision of the fields in the mode page.

[b] GB is 10^9 and TB is 10^{12}

4.19.2.2 Wrap-wise Partitioning with FastSync

This type of partitioning is optimized for small file performance by allowing FastSync (i.e., Virtual Backhitch) which sacrifices some capacity.

Table 21, and table 22 show the partition sizes that result from a MODE SELECT of the Medium partition mode page (see 6.6.13) with the indicated field settings.

Table 21 — Partition sizes for wrap-wise partitioning with FastSync (selection fields)

Ref[a]	FDP	SDP	IDP	ADDITIONAL PARTITIONS DEFINED[b]	PARTITION SIZE			
					(FIRST)	(SECOND)	(THIRD)	(LAST)
row1	1	0	0	X	FDP set to one dictates Wrap-wise Partitioning with no FastSync (see 4.19.2.1 on page 63).			
row2	0	1	0	00h	X	X	X	X
row3				01h				
row4				02h				
row5				03h				
row6	0	0	1	00h	FFFFh	0	0	0
row7					s[c]			
row8				01h	s	FFFFh		
row9					FFFFh	t		
row10					s[c]	t[c]		
row11				02h	s	t	FFFFh	
row12					s	FFFFh	u	
row13					FFFFh	t	u	
row14					s[c]	t[c]	u[c]	
row15				03h	s	t	u	FFFFh
row16					s	t	FFFFh	v
row17					s	FFFFh	u	v
row18					FFFFh	t	u	v
row19					s[c]	t[c]	u[c]	v[c]
others	All other combinations							

[a] The Ref column is the reference that ties the rows in this table to the corresponding rows in table 22.

[b] When more than one partition is defined there may be overhead that results in a loss of capacity

[c] Value must be exact partition size allowed and when summed with other values in the row equal full capacity. This permits a MODE SENSE followed by a MODE SELECT with no change. It is highly recommended that this method only be used in the case where the MODE SELECT data is a return of the MODE SENSE data.

Table 22 — Partition sizes for wrap-wise partitioning with FastSync (resultant sizes) (part 1 of 2)

Ref [a]	Partition 0[b]	Partition 1[b]	Partition 2[b]	Partition 3[b]
row1	FDP set to one dictates Wrap-wise Partitioning with no FastSync (see 4.19.2.1 on page 63).			
row2	C_{MAX}	-		
row3	s=K*n; where n=integer of {(N_2+1)/2}	t=K*m; where m=N_2+1-n	-	
row4	s=K*n; where n=integer of {(N_3+2)/3}	t=K*m; where m=integer of {(N_3+2)/3}	s=K*u; where u=N_3+2-n-m	-
row5	s=K*n; where n=integer of {(N_4+3)/4}	t=K*m; where m=integer of {(N_4+3)/4}	s=K*u; where u=integer of {(N_4+3)/4}	s=K*v; where v=N_4+3-n-m-u
row6	C_{MAX}	-		
row7	C_{MAX}			
row8	s=K*n; where 1<=n<=N_2	C_{MAX} - overhead-(partition size 0)	-	
row9	C_{MAX} - overhead-(partition size 1)	t=K*m; where 1<=m<=N_2		
row10	s=K*n; where 1<=n<=N_2 and n+m=N_2+1	t=K*m; where 1<=m<=N_2 and n+m=N_2+1		
row11	s=K*n; where 1<=n<=N_3	t=K*m; where 1<=m<=N_3	C_{MAX} - overhead-(partition size 0)-(partition size 1)	-
row12		C_{MAX} - overhead-(partition size 0)-(partition size 1)	u=K*p; where 1<=p<=N_3	
row13	C_{MAX} - overhead-(partition size 1)-(partition size 2)	t=K*m; where 1<=m<=N_3		
row14	s=K*n; where 1<=n<=N_3 and n+m+p=N_3+2	t=K*m; where 1<=m<=N_3 and n+m+p=N_3+2	u=K*p; where 1<=p<=N_3 and n+m+p=N_3+2	

[a] The values in the Ref column refer back to the associated row in table 21.

[b] The values for C_{MAX}, K, N_2, N_3, N_4, and the sum of existing partitions are specified in table 23.

Table 22 — Partition sizes for wrap-wise partitioning with FastSync (resultant sizes) (part 2 of 2)

Ref [a]	Partition 0[b]	Partition 1[b]	Partition 2[b]	Partition 3[b]
row15	s=K*n; where 1<=n<=N_4	t=K*m; where 1<=m<=N_4	u=K*p; where 1<=p<=N_4	C_{MAX} - overhead- (partition size 0)- (partition size 1)- (partition size 2)
row16			C_{MAX} - overhead- (partition size 0)- (partition size 1)- (partition size 2)	v=K*q; where 1<=q<=N_4
row17		C_{MAX} - overhead- (partition size 0)- (partition size 2)- (partition size 3)	u=K*p; where 1<=p<=N_4	
row18	C_{MAX} - overhead- (partition size 1)- (partition size 2)- (partition size 3)	t=K*m; where 1<=m<=N_4		
row19	s=K*n; where 1<=n<=N_4 and n+m+p+q=N_4+3	t=K*m; where 1<=m<=N_4 and n+m+p+q=N_4+3	u=K*p; where 1<=p<=N_4 and n+m+p+q=N_4+3	v=K*q; where 1<=q<=N_4 and n+m+p+q=N_4+3
others	Check Condition, Illegal Request, Invalid Field in Parameter Data			

[a] The values in the Ref column refer back to the associated row in table 21.

[b] The values for C_{MAX}, K, N_2, N_3, N_4, and the sum of existing partitions are specified in table 23.

Table 23 — Partition values for wrap-wise partitioning with FastSync by density 54 and media type

Parameter in table 22	Primary Density Code = 54h[b]		
	Media Type		
	JB	JC	JK
C_{MAX}	1.6 TB	4.0 TB	500 GB
K [a]	57.142 GB	100.000 GB	12.500 GB
N_2	24	34	34
N_3	22	31	31
N_4	20	28	28
Sum of all partitions[a] s+t+[u]+[v]	s+t=1428.571 GB s+t+u=1371.428 GB s+t+u+v=1314.285 GB	s+t=3 500.000 GB s+t+u=3 300.000 GB s+t+u+v=3 100.000 GB	s+t=437.500 GB s+t+u=412.500 GB s+t+u+v=387.500 GB

[a] The precision of capacity values able to be expressed is limited by the 2-byte PARTITION SIZE field and the value of the PARTITION UNITS field of the Medium Partition mode page (see 6.6.13). Actual size on medium is not limited by the precision of the fields in the mode page.

[b] GB is 10^9 and TB is 10^{12}

Table 24 — Partition values for wrap-wise partitioning with FastSync by density 55h and media type

Parameter in table 22	Primary Density Code = 55h[b]			
	Media Type			
	JC	JK	JD	JL
C_{MAX}	7.0 TB	900 GB	10.0 TB	2.0 TB
K^1	97.222 GB	12.500 GB	125.000 GB	25.000 GB
N_2	68	68	76	76
N_3	66	66	74	74
N_4	64	64	72	72
Sum of all partitions[a] s+t+[u]+[v]	s+t= 6708.333 GB s+t+u= 6611.111 GB s+t+u+v= 6513.888 GB	s+t= 862.500 GB s+t+u= 850.000 GB s+t+u+v= 837.500 GB	s+t= 9625.000 GB s+t+u= 9500.000 GB s+t+u+v= 9375.000 GB	s+t= 1925.000 GB s+t+u= 1900.000 GB s+t+u+v= 1875.000 GB

[a] The precision of capacity values able to be expressed is limited by the 2-byte PARTITION SIZE field and the value of the PARTITION UNITS field of the Medium Partition mode page (see 6.6.13). Actual size on medium is not limited by the precision of the fields in the mode page.

[b] GB is 10^9 and TB is 10^{12}

Table 25 — Partition values for wrap-wise partitioning with FastSync by density 56h and media type

Parameter in table 22	Primary Density Code = 56h[b]	
	Media Type	
	JD	JL
C_{MAX}	15.0 TB	3.0 TB
K^1	125.000 GB	25.000 GB
N_2	114	114
N_3	111	111
N_4	108	108
Sum of all partitions[a] s+t+[u]+[v]	s+t= 14 375.000 GB s+t+u= 14 125.000 GB s+t+u+v= 13 875.000 GB	s+t= 2 875.000 GB s+t+u= 2 825.000 GB s+t+u+v= 2 775.000 GB

[a] The precision of capacity values able to be expressed is limited by the 2-byte PARTITION SIZE field and the value of the PARTITION UNITS field of the Medium Partition mode page (see 6.6.13). Actual size on medium is not limited by the precision of the fields in the mode page.

[b] GB is 10^9 and TB is 10^{12}

Table 26 — Partition values for wrap-wise partitioning with FastSync by density 57h and media type

Parameter in table 22	Primary Density Code = 57h [b]	
	Media Type	
	JE	JM
C_{MAX}	20.0 TB	5.0 TB
K^1	147.058 GB	36.764 GB
N_2	130	133
N_3	127	130
N_4	124	127
Sum of all partitions [a] s+t+[u]+[v]	s+t= 19 264.705 GB s+t+u= 18 970.588 GB s+t+u+v= 18 676.470 GB	s+t= 4 816.176 GB s+t+u= 4 742.647 GB s+t+u+v= 4 669.117 GB

[a] The precision of capacity values able to be expressed is limited by the 2-byte PARTITION SIZE field and the value of the PARTITION UNITS field of the Medium Partition mode page (see 6.6.13). Actual size on medium is not limited by the precision of the fields in the mode page.

[b] GB is 10^9 and TB is 10^{12}

4.19.3 Longitudinal Partitioning

Longitudinal partitioning is provided for better random access performance. Longitudinal partitioning dissects the tape into two sections of tape, the first spanning from beginning of tape toward end of tape until enough tape has been spanned to accommodate the requested size and uses all wraps for the first partition. A guard gap is created between the first section and the second section which results in less than 1% capacity loss. The second section uses the remaining length of tape and all wraps in that section for the second partition. A logical representation of a longitudinally partitioned volume is shown in figure 10. There is a minimum length of tape to use for a partition and this requirement is enforced with automatic rounding if needed during creation of the partition.

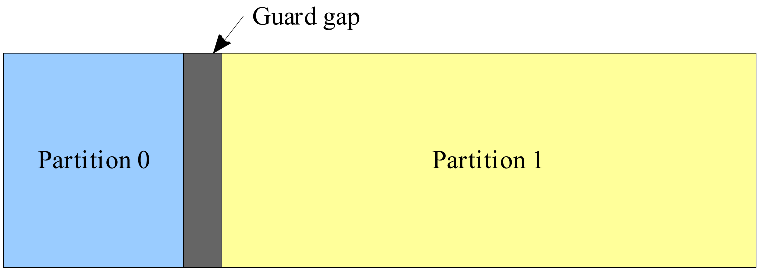


Figure 10 — Longitudinal partitioning

When using longitudinal partitioning a maximum of two partitions are supported.

Table 27, and table 28 show the partition sizes that result from a MODE SELECT of the Medium partition mode page (see 6.6.13) with the indicated field settings.

Table 27 — Partition sizes for longitudinal partitioning (selection fields)

FDP	SDP	IDP	ADDITIONAL PARTITIONS DEFINED[a]	PARTITION SIZE		Resultant Sizes	
				(FIRST)	(SECOND)	Partition 0[c]	Partition 1[c]
1	0	0	X	FDP set to one dictates wrap-wise partitioning (see 4.19.2).			
0	1	0	00h	X	X	C_{MAX}	-
			01h			$\frac{\sum \mathit{Partitions}}{2}$	$\frac{\sum \mathit{Partitions}}{2}$
0	0	1	00h	FFFFh	0	C_{MAX}	-
				s[c]			
			01h	s	FFFFh	P_{MIN} <= s <= P_{MAX}	C_{MAX} - (size of partition 0) - overhead
				FFFFh	t	C_{MAX} - (size of partition 1) - overhead	P_{MIN} <= t <= P_{MAX}
				s[b, d]	t[b, d]	**P_{MIN} <= s <= P_{MAX}**	P_{MIN} <= t <= P_{MAX}
All other combinations						Check Condition, Illegal Request, Invalid Field in Parameter Data	

[a] When more than one partition is defined a guard gap is created between the partitions which results in less than 1% capacity loss.

[b] Value must be exact partition size allowed and when summed with other values in the row equal full capacity. This permits a MODE SENSE followed by a MODE SELECT with no change. It is highly recommended that this method only be used in the case where the MODE SELECT data is a return of the MODE SENSE data.

[c] The values of C_{MAX}, P_{MIN}, P_{MAX}, and $\sum \mathit{Partitions}$ are specified in table 28.

[d] s+t = C_{MAX} - overhead

Table 28 — Longitudinal partition values by density and media type

Parameter in table 27[a]	Primary Density Code = 54h[b]		Primary Density Code = 55h[b]		Primary Density Code = 56h[b]
	Media Type		Media Type[c]		Media Type[c]
	JB	JC	JC	JD	JD
C_{MAX}	1.6 TB	4.0 TB	7.0 TB	10.0 TB	15.0 TB
P_{MIN}	95.860 GB	219.726 GB	439.102 GB	579.008 GB	868.512 GB
P_{MAX}	1 376.051 GB	3 447.812 GB	6 303.227 GB	9 085.451 GB	13 628.177 GB
$\sum \mathit{Partitions}$	1 471.911 GB	3 667.538 GB	6 742.329 GB	9 664.460 GB	14 496.690 GB

[a] The precision of capacity values able to be expressed is limited by the 2-byte PARTITION SIZE field and the value of the PARTITION UNITS field of the Medium Partition mode page (see 6.6.13). Actual size on medium is not limited by the precision of the fields in the mode page.

[b] GB is 10^9 and TB is 10^{12}

[c] Media Types JE/JM do not support longitudinal partitioning.

Note 1 - Density Code 57h does not support longitudinal partitioning.

4.19.4 Partitioning and media types

Partitioning of volumes is supported on media in native 3592 E07 logical formats and later only (i.e., primary density code = 5nh; where $n \geq 4$ and a secondary density code = 5nh or 7nh; where $n \geq 4$).

Reciept of a FORMAT MEDIUM command may cause an implicit re-format to occur depending on the settings described in 4.19.6.

4.19.5 Partitioning and WORM volumes

Partitioning of WORM volumes is supported on JY and JZ media types only. A WORM volume may have up to two partitions and only supports using Wrap-wise Partitioning with no FastSync (see 4.19.2.1 on page 63).

A WORM volume with no data on it may be partitioned, but once data is written on it, the partitioning may not be changed.

4.19.6 Partitioning and reformatting

Medium Reuse at Higher (or Lower) Densities (see 4.16 on page 57) describes how a partitioned volume may be reused at lower or higher densities.

When not subject to reformatting due to the conditions described in 4.16, partitions are created and destroyed using FORMAT MEDIUM - 04h (beginning with E07) (see 5.2.7 on page 112). How a volume is formatted depends on the media type (see 6.6.2.1), if the volume is scaled (see 6.6.19), the PENDING WRITE DENSITY AT BOP 0 field of mode page 25h (see 6.6.21), the settings in the Medium Partition mode page (see 6.6.13), and the FORMAT MEDIUM command (see page 112) settings. The FORMAT MEDIUM command specifies how to format the volume and the interactions of these conditions using the FORMAT field.

The Medium Partition mode page is used to specify the group of medium partitions. The partitioning of the mounted volume is not changed until a subsequent FORMAT MEDIUM command is issued while the volume is mounted.

The device ensures consistency of the partitioning values set in Medium Partition mode page by causing a CHECK CONDITION status with the sense key set to ILLEGAL REQUEST and the additional sense code set to PARAMETER VALUE INVALID to be returned to a subsequent FORMAT MEDIUM command attempting to use the values set in Medium Partition mode page if values in those values become inconsistent between when they were set and when the FORMAT MEDIUM command is received. The invalidation of the values in this page is accomplished by setting the FDP, SDP, and IDP bits to zero and the other values in this page to:

a) the values dictated by the format of the currently mounted volume, if a volume is mounted; or
b) the default values present after power-on, if no volume is mounted.

The invalidation of values in this page occurs if:

a) THE PENDING WRITE DENSITY AT BOP 0 field of MP 25h: Read/Write Control (see 6.6.21 on page 427) is modified;
b) the volume is capacity scaled (see MP 23h: Medium Sense (see 6.6.19 on page 414)); or
c) other events that are determined to make the values in this page inconsistent.

4.19.7 Partitioning and encryption

The relationship between partitioning and encryption is described in Device Hardware Encryption (some models) (see 4.20 on page 76).

4.20 Device Hardware Encryption (some models)

Certain versions and revisions of this device contain hardware which performs user data write encryption and read decryption, protecting all user data written to the medium from unauthorized use [provided it is integrated into a secure system design]. Device support for encryption may be determined by reading mode page MP 24h: Initiator-Specific Extensions (see 6.6.20) with the Mode Sense command, and medium support for encryption may be determined by reading MP 23h: Medium Sense (see 6.6.19) with the Mode Sense command.

> NOTE 12 - It should be noted that encryption is managed on a partition level. Encryption must be enabled before the initial write/label process, and must be maintained throughout the write/read usage of the medium.
>
> NOTE 13 - It should be noted that encryption enabled cartridges, can only be read by devices with encryption hardware.

Multiple ways of controlling encryption settings is supported on devices that support encryption. The 3592 supported encryption prior to any standardization of a method for controlling encryption. Because of this, 3592 had to devise its own proprietary method for controlling encryption settings and passing keys into the device. IBM did work with the T10 standards committee to derive a standards based method, but those standards were just starting their work when the 3592 with encryption support was first released. As soon as the standards methods were devised 3592 added support from the standards based methods. These encryption control methodologies are called:

a) Encryption Control - IBM Proprietary Protocol (IPP); and
b) Encryption Control - T10 Standards.

On volumes with multiple partitions, the drive handles encryption in each partition differently and may report a different density code. The density code reported for a partition is determined when a write occurs at BOP:

a) If logical block 0 in a partition is encrypted using IBM proprietary methods (see 4.20.1) all blocks within the partition are enforced to be encrypted and the density code reported is 74h. If EEDKs are used, then each partition has separate EEDKs;
b) If logical block 0 in a partition is not encrypted, then the density code reported is 54h; and
c) if the encryption method in the drive is set to AME-T10 (see 4.20.2), then an intermix of encrypted and unencrypted blocks are allowed and the density code reported is 54h.

4.20.1 Encryption Control - IBM Proprietary Protocol (IPP)

The following terms are used to describe the methods of control that fall into the IPP:

a) Library Managed Encryption (LME);
b) System Managed Encryption (SME); and
c) Application Managed Encryption - IBM (AME-IBM).

(see 6.6.21.0.1) provides extensive information on this method.

When a device is enabled to perform encryption using one of the IBM Proprietary Protocols (i.e., LME, SME, or AME-IBM) encryption parameters are determined at first write from BOP. On volumes with multiple partitions this means that on a write from BOP(i.e. LBA 0) of each partition the encryption parameters are determined. Writes away from BOP use the existing encryption parameters. If a logical block is encrypted, then all logical blocks are encrypted. If a logical block is not encrypted then all logical blocks are not encrypted. When a partition change occurs the encryption parameters are cleared.

Please see IBM for additional information on IPP.

4.20.2 Encryption Control - T10 Standards

The 3592 device has added support for controlling encryption and passing clear text keys using T10 standards methods. These methods are described in SSC-4 as well as in this document. Note that not all methods described in SSC-4 are supported.

When a device is enabled to perform encryption using AME-T10 the encryption parameters are set by the application. When the encryption parameters are set to encrypt, logical blocks are encrypted. When encryption parameters are set to not encrypt, logical blocks are not encrypted. Changing partitions when enabled for AME-T10 does not necessarily effect the encryption parameters.

The 3592 device uses the term Application Managed Encryption - T10 (AME-T10) to signify that it is using this standards based method.

The 3592 devices supports the T10 method of passing the key in clear text. For specifics on support refer to SECURITY PROTOCOL IN (SPIN) - A2h (see 5.2.39), SECURITY PROTOCOL OUT (SPOUT) - B5h (see 5.2.40), and Security Protocol Parameters (SPP) (see 6.7).

4.21 Device Data Recovery (DDR) using Recover Buffered Data (RBD)

In some system environments it might be more practical to perform error recovery at a low layer which may not have easy access to the source(s) of data which were written to the buffer in previous command(s). When a write error occurs and the device is unable to transfer buffered data to the medium, it might be desirable for the host to read the unwritten data in the device buffer back into the host system. This may be accomplished using the Recover Buffered Data command (see 5.2.30).

The recover buffered data command has two operating modes, LIFO and FIFO. In FIFO mode the data is read in a first-in-first-out method (the order it was written in), from lowest logical block number to highest (most recent). In LIFO mode the data is read in a last-in-first-out method (the opposite order it was written in), from highest (most recent) logical block number to lowest. This method is controlled with the RBO (Recover Buffer Order) field (see "MP 10h: Device Configuration" on page 394).

Additionally, a Buffer Association mode affects when the RBD command can be used. This is controlled by the Buffer Association Enablement field (see "MP 24h: Initiator-Specific Extensions" on page 424).

4.21.1 No Association (Legacy Method)

When no associations are set, any buffered data is retained until the next Rewind command, Load/Unload command, or unload operation. This requires any desirable data to be read from the buffer while the drive is still in the error state. This requires all data to be read and maintained in the host before most error recovery. An example of a DDR operation using this method might be:

1) Application is writing buffered data
2) Permanent Write Error Occurs
3) Drive prepares for RBD and determines amount of data in buffer
4) Application queries the amount of data in the buffer
5) DDR begins, host uses RBD to read data into host memory
6) Tape is unloaded
7) Another drive is selected and same tape is loaded
8) Seek to point of (or before) perm
9) Data is written from host memory
10) Job resumes at point of failure

4.21.2 Unload with Write Error Association

When this association is set, the device host data buffer remains unaltered in the write error case after the medium has been unloaded. This allows the buffered data to be available for recovery after the cartridge has been removed. An example of a DDR operation using this method might be:

1) Application is writing buffered data
2) Permanent Write Error Occurs
3) Drive prepares for RBD and determines amount of data in buffer
4) Application queries the amount of data in the buffer (could be after unload, but probably drives some aspect of the DDR and the unload is special on the DDR path)
5) Tape is unloaded (without rewind, etc), and drive is held reserved/allocated
6) Another drive is selected and same tape is loaded
7) Seek to point of (or before) perm
8) DDR begins, host uses RBD to read data into host memory then writes to ready drive (until complete)
9) Drive which experienced perm is released (can be used for other jobs)
10) Job resumes at point of failure

4.22 String Search Function (not J1A)

A function is provided to have the device perform searches on the medium for various data patterns. This can be configured with MP 37h: String Search (not J1A) (see 6.6.23). The search operation itself is invoked with a vendor-unique String Search command (see 5.2.45), a variant of the Space command (see 5.2.44), a vendor-unique variation of the Read command (see 5.2.20), or a Send Diagnostic command (see 5.2.41). This function allows searches of up to eight (8) strings of up to thirty two (32) bytes each that can be masked or selected on a bit and byte level. All logical AND and OR combinations of these strings within a single logical block can be used as a search match criteria.

> NOTE 14 - Only eight (8) strings of sixteen (16) bytes are supported. Longer criteria may be specified, but only the first sixteen (16) bytes of each string are significant for matches.

4.22.1 Truth Table Usage

The Truth Table is a bitwise field which supports all AND/OR/NOT combinations of the eight (8) [Masked] Search Strings. The table is easiest to understand with the following representation, where each bit represents an OR combination. A successful match will occur if exactly ALL (AND) strings listed in any cell corresponding to a set bit is found. Since combinatorial matches may occur, all bits with the AND conditions should be set, even if they include other values that are not part of the search.

For example, is the desire is to search for string (5 AND 6) or (7 AND 8), all bits with 56 and all bits with 78 should be set. This would result in a Truth Table of: 11h, 1Fh, 11h, 1Fh, 11h, 1Fh, 11h, 1Fh, 11h, 1Fh, 11h, 1Fh, 11h, 1Fh, 11h, 1Fh, 11h, 1Fh, 11h, 1Fh, 11h, 1Fh, 11h, 1Fh, 11h, 1Fh, 11h, 1Fh, 11h, 1Fh, 11h, 1Fh.

All combinations are possible, but some are tricky. This is especially true depending on the values of the byte masks. Special care should be taken with combinatorial sets like (1 AND 2 AND 3 AND 4) OR (1 AND 2 AND 5 AND 6). All combinations of 1,2,3,4,5,6 which include all of 1234 or 1256 must be set.

If String Search Byte Mask(s) are set to 00h then a string match will not result. Due to this, bits in the Truth Table which have strings which are not part of the search may not need to be included in the Truth Table setup. The search would also work correctly those which do not have unused strings set, (provided the String Search Byte Mask(s) are set to 00h). The same search of (5 AND 6) or (7 AND 8) could be properly executed with a Truth Table of: 11h, 1Fh, 11h, 1Fh, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h. If strings 1 through 4 had specified valid criteria, the preceding truth table represents the expression (NOT(1) AND NOT(2) AND NOT(3) AND NOT(4) AND 5 AND 6) OR (NOT(1) AND NOT(2) AND NOT(3) AND NOT(4) AND 7 AND 8).

Special care should be taken with combinatorial sets like (1 AND 2 AND 3 AND 4) OR (1 AND 2 AND 5 AND 6). All combinations of 1,2,3,4,5,6 which include all of 1234 or 1256 must be set. If strings and masks 7 and 8 are set to 00h then only those which do not include 7 or 8 need be set.

Excluded strings may also be set for string matches by insuring that truth table bit entries which include the NOT string are set to 0b. This may be applied to any combination of AND or OR logic.

> NOTE 15 - For programmers wanting to take an algorithmic approach, the table may be viewed as a bit array of 256 bits (numbered 0 to 255 corresponding to the table respectively from byte 0 bit 7... byte 0 bit 0, byte 1 bit 7... byte 31 bit 0). Each AND set can calculate a value by setting all of the 2^(8-string) bits in that AND group. Any bit

array offset which has all of these string bits set should be set to 1b, The same process should be repeated for each OR clause.

Table 29 — String Search Truth Table

Bit Byte	7 msb	6	5	4	3	2	1	lsb 0
0	-	8	7	78	6	68	67	678
1	5	58	57	578	56	568	567	5678
2	4	48	47	478	46	468	467	4678
3	45	458	457	4578	456	4568	4567	45678
4	3	38	37	378	36	368	367	3678
5	35	358	357	3578	356	3568	3567	35678
6	34	348	347	3478	346	3468	3467	34678
7	345	3458	3457	34578	3456	34568	34567	345678
8	2	28	27	278	26	268	267	2678
9	25	258	257	2578	256	2568	2567	25678
10	24	248	247	2478	246	2468	2467	24678
11	245	2458	2457	24578	2456	24568	24567	245678
12	23	238	237	2378	236	2368	2367	23678
13	235	2358	2357	23578	2356	23568	23567	235678
14	234	2348	2347	23478	2346	23468	23467	234678
15	2345	23458	23457	234578	23456	234568	234567	2345678
16	1	18	17	178	16	168	167	1678
17	15	158	157	1578	156	1568	1567	15678
18	14	148	147	1478	146	1468	1467	14678
19	145	1458	1457	14578	1456	14568	14567	145678
20	13	138	137	1378	136	1368	1367	13678
21	135	1358	1357	13578	1356	13568	13567	135678
22	134	1348	1347	13478	1346	13468	13467	134678
23	1345	13458	13457	134578	13456	134568	134567	1345678
24	12	128	127	1278	126	1268	1267	12678
25	125	1258	1257	12578	1256	12568	12567	125678
26	124	1248	1247	12478	1246	12468	12467	124678
27	1245	12458	12457	124578	12456	124568	124567	1245678
28	123	1238	1237	12378	1236	12368	12367	123678
29	1235	12358	12357	123578	12356	123568	123567	1235678
30	1234	12348	12347	123478	12346	123468	123467	1234678
31	12345	123458	123457	1234578	123456	1234568	1234567	12345678

Each digit represents one of the strings set by Mode Page 37h Subpage (criteria) - String Descriptor 10h. Each set of digits in a specific cell represents an AND condition of all the specified strings, while any digits not present in the cell indicates either a string that is not setup, or must not appear. Each cell that is selected is OR'ed with the other cells that are selected. If the NOT logic is used, then the converse is also true.

4.23 Medium auxiliary memory (beginning with E07)

Some types of media, especially removable media, include a non-volatile memory referred to as MAM (medium auxiliary memory). Medium auxiliary memory is used to store data describing the media and its contents. This standard supports medium auxiliary memory with the READ ATTRIBUTE command (see 5.2.21) and the WRITE ATTRIBUTE command (see 5.2.49). These commands are used to retrieve and store information in the medium auxiliary memory in the form of MAM attributes.

A MAM attribute is represented in a format described in 6.5.1 on page 363 and is composed of:

a) an attribute identifier;
b) an attribute format code;
c) a bit indicating whether the identified attribute is read only;
d) an attribute length specifying the number of bytes in the identified attribute value; and
e) the value of the identified attribute.

There are three types of MAM attributes (see table 30).

Table 30 — Types of MAM attributes

Attribute Type	Attribute Source	Example	Readable with READ ATTRIBUTE	Writable with WRITE ATTRIBUTE
Medium	Permanently stored in the medium auxiliary memory during manufacture.	Media Serial Number	Yes	No
Device	Maintained by the device server.	Load Count	Yes	No
Host	Maintained by the application client.	Backup Date	Yes	Yes

Depending on that attribute type, MAM attributes have the states shown in table 31.

Table 31 — MAM attribute states

Attribute Type	Attribute State	Description
Medium or Device	Read Only	An application client may read the contents of the MAM attribute with the READ ATTRIBUTE command, but an attempt to clear or change the MAM attribute using the WRITE ATTRIBUTE command shall result in the command being terminated with CHECK CONDITION status. When the READ ONLY bit (see 6.5.1) is one, the attribute is in the read only state.
	Unsupported	The device server does not support the MAM attribute and shall not return it in response to a READ ATTRIBUTE command.
Host	Nonexistent	A host attribute does not exist in the medium auxiliary memory until a WRITE ATTRIBUTE command creates it.
	Read/Write	The MAM attribute has been created using the WRITE ATTRIBUTE command. After the MAM attribute has been created, the contents may be altered using subsequent WRITE ATTRIBUTE commands. A read/write MAM attribute may be returned to the nonexistent state using a WRITE ATTRIBUTE command with the attribute length set to zero. When the READ ONLY bit (see 6.5.1) is zero, the MAM attribute is in the read/write state.

4.24 Volume Coherency (beginning with E07)

An application client may need to be able to determine if all logical objects on a volume are coherent with the last time an application client wrote to this volume. The VOLUME COHERENCY INFORMATION attribute (see 6.5.2.4.11) of MAM is provided for an application client to collect and save information necessary for this determination.

The VOLUME COHERENCY INFORMATION attribute for each partition is written to MAM by the application client when it has completed a write job (e.g., the volume is demounted). The VOLUME COHERENCY INFORMATION attribute contains references to a volume coherency set that the application client has written to logical objects on a partition. An application client should not create a VOLUME COHERENCY INFORMATION attribute unless it has written a volume coherency set to that partition. The volume coherency set shall include a volume coherency count. The application client shall maintain one volume coherency count for an entire volume and shall monotonically increase the volume coherency count when the state of the volume coherency set changes (e.g., writing identical volume coherency sets on each partition does not force a change of volume coherency count). When the application client writes the VOLUME COHERENCY INFORMATION attribute to MAM for a specific partition the VOLUME CHANGE REFERENCE VALUE field of the VOLUME COHERENCY INFORMATION attribute for a partition shall contain the value returned in the ATTRIBUTE VALUE field of the VOLUME CHANGE REFERENCE attribute after the last volume coherency set was written to the volume. The VOLUME COHERENCY COUNT field of the VOLUME COHERENCY INFORMATION attribute shall contain the volume coherency count that was written to the last volume coherency set written to that partition. The VOLUME COHERENCY SET IDENTIFIER field of the VOLUME COHERENCY INFORMATION attribute for a partition contains the logical object identifier of the first byte of the last volume coherency set written to that partition. The APPLICATION CLIENT SPECIFIC INFORMATION field of the VOLUME COHERENCY INFORMATION attribute for a partition contains information the application client binds with the coherency set referenced by the VOLUME COHERENCY SET IDENTIFIER field.

> NOTE 16 - The application client needs to guarantee that no other application client updates the logical objects on the volume between the time it completes writing and the time it updates the MAM parameter (e.g., use reservations)

An application client may verify that the volume coherency set written in a partition has not changed since the VOLUME COHERENCY INFORMATION attribute was written when the application client reads the VOLUME COHERENCY INFORMATION attribute for a partition (e.g., when a volume is mounted) and compares the value in the VOLUME CHANGE REFERENCE VALUE field with the value returned in the ATTRIBUTE VALUE field of the VOLUME CHANGE REFERENCE attribute. If the values match, then the volume coherency set written in that partition is unchanged.

To find the most recently written volume coherency set, the application client searches the VOLUME COHERENCY INFORMATION attributes of the partitions for which the volume coherency set is unchanged and finds the largest value in the VOLUME COHERENCY COUNT field. The application client then verifies the largest value in the VOLUME COHERENCY COUNT field with the volume coherency count stored in the volume coherency set beginning at the logical object specified by the VOLUME COHERENCY SET IDENTIFIER field. If this matches, then this is the volume coherency set that was most recently written.

The APPLICATION CLIENT SPECIFIC INFORMATION field may also be used by the application client as part of this coherency check. If the information verifies for a partition, then the volume is coherent with the last access by this application. If the information does not verify for a partition, then the volume is not coherent with the last access by this application.

4.25 Device Clocks

The drive supports a Device Clock that maintains a timestamp for various items. This timestamp gets recorded in drive error logs.

The TIMESTAMP ORIGIN is one of those specified in table 32.

Table 32 — TIMESTAMP ORIGIN

Value	Definition
000b	Timestamp initialized to zero at power-on
001b	Reserved
010b	Timestamp initialized by the SET TIMESTAMP command or MP 21h: TOD Control (see 6.6.17 on page 411)
011b	Timestamp initialized by the Library over the Library Drive Interface (i.e. RS-422)
100b - 111b	Reserved

Once a timestamp is initialized it begins counting from that time forward. Once the timestamp is initialized it remains in effect until one of the following occurs:

a) A MODE SELECT is processed with MP 21h: TOD Control (see 6.6.17 on page 411);
b) A SET TIMESTAMP command (see 5.2.42) is processed;
c) An LDI command is processed that modifies the timestamp; or
d) A Hard Reset event occurs.

The method used is indicated in the Extended Control mode page.

The Timestamp is not affected by an I_T nexus loss or a Logical Unit reset.

The TIMESTAMP is specified in table 33.

Table 33 — TIMESTAMP Format

Byte	Bit							
	7	6	5	4	3	2	1	0
0	(MSB)			TIMESTAMP				
5								(LSB)

The TIMESTAMP field contains the value established at the last action that set the timestamp incremented by one for every millisecond that has elapsed since the timestamp was set.

4.26 Dynamic runtime information (beginning with E07)

4.26.1 Dynamic runtime information overview

Dynamic runtime information allows an initiator to set dynamic runtime attributes (DRA) about itself into a device server. The device server then associates those attributes to the I_T_L nexus and uses the information and associations for enhanced data collection and debugging. This information and the associations are added to device error logs (e.g., drive dump) and are provided for retrieval by an application client through the READ DYNAMIC RUNTIME ATTRIBUTE command (see 5.2.24).

The model E07 and later devices support dynamic runtime attributes with the READ DYNAMIC RUNTIME ATTRIBUTE command (see 5.2.24) and the WRITE DYNAMIC RUNTIME ATTRIBUTE command (see 5.2.51). These commands are used to retrieve and store information in the form of dynamic runtime attributes.

A DRA is represented in a format described in 6.2.1 and is composed of:

a) an I_T nexus index;
b) an attribute identifier;
c) an attribute format code;
d) a bit indicating whether the identified attribute is read only;
e) an attribute length specifying the number of bytes in the identified attribute value; and
f) the value of the identified attribute.

There are three types of DRA attributes (see table 34).

Table 34 — Types of DRA attributes

Attribute Type	Focus	Attribute Source	Readable with READ DYNAMIC RUNTIME ATTRIBUTE	Writable with WRITE DYNAMIC RUNTIME ATTRIBUTE	Reference
Logical unit	Device	Set by the device server.	Yes	No	6.2.2.2
Target	I_T nexus	Set by the device server.	Yes	No	6.2.2.3
Initiator	I_T nexus	Set by the application client	Yes	Yes	6.2.2.4

DRA attributes have the states shown in table 35.

Table 35 — DRA attribute states

Attribute State	Description
Read Only	An application client may read the contents of the DRA attribute with the READ DYNAMIC RUNTIME ATTRIBUTE command, but an attempt to clear or change the DRA attribute using the WRITE DYNAMIC RUNTIME ATTRIBUTE command shall result in the command being terminated with CHECK CONDITION status with the sense key set to DATA PROTECT and the additional sense code set to WRITE PROTECTED. When in the read only state the READ ONLY bit (see 6.2.1) is one.
Unsupported	The device server does not support the DRA attribute and shall not return it in response to a READ DYNAMIC RUNTIME ATTRIBUTE command.
Nonexistent	An initiator attribute does not exist in the dynamic runtime attributes until a WRITE DYNAMIC RUNTIME ATTRIBUTE command creates it.
Read/Write	The DRA attribute has been created using the WRITE DYNAMIC RUNTIME ATTRIBUTE command. After the DRA attribute has been created, the contents may be altered using subsequent WRITE DYNAMIC RUNTIME ATTRIBUTE commands. A Read/Write DRA attribute may be returned to the nonexistent state using a WRITE DYNAMIC RUNTIME ATTRIBUTE command with the attribute length set to zero. When in the Read/Write state the READ ONLY bit (see 6.2.1) is zero.

4.26.2 Dynamic runtime information timestamp

Some dynamic runtime attributes have a timestamp associated with them. The timestamp used is described in Device Clocks (see 4.25 on page 83). If no timestamp is set by either a SCSI command (i.e., SET TIMESTAMP - A4h[0Fh] (see 5.2.42 on page 205) command or MP 21h: TOD Control (see 6.6.17 on page 411) mode page) or by the library, then the timestamp will be power-on time and may not be able to be correlated to external logs (e.g., device driver logs, application logs).

4.26.3 Setting dynamic runtime information into the drive

An application client may set attributes into the drive using the WRITE DYNAMIC RUNTIME ATTRIBUTE command (see 5.2.51) to set one or more of the initiator type attributes defined in 6.2.2.4. The application client may write these values at any time and may change these values at any time. If an application client attempts to create a new attribute by writing an attribute that was previously in the non-existent state and the device server does not have the resources necessary to create that attribute the device server shall reject the command with a CHECK CONDITION with the sense code set to ILLEGAL REQUEST and the additional sense code set to INSUFFICIENT RESOURCES (i.e., 5h / 5503h).

4.26.4 Retrieving dynamic runtime information from the drive

An application client may read attributes by using the READ DYNAMIC RUNTIME ATTRIBUTE command (see 5.2.24). The application client may request a single attribute or multiple attributes in a single command. The application client may read any existent attribute (see 6.2).

4.26.5 Management of dynamic runtime information

Dynamic runtime attributes have either a focus of a logical unit (i.e., logical unit type attributes) or a focus of I_T nexus (i.e., target type attributes and initiator type attributes). This relationship is shown in figure 11.

Figure 11 — Dynamic runtime attributes focus

For each dynamic runtime attribute (see 6.2) that the device server supports, if a command is received that should cause an update of one or more of the dynamic runtime attributes (e.g., Reserve, Persistent Reserve Out, Prevent/Allow Medium Removal), then the device server shall update that dynamic runtime attribute. If one or more of the Initiator type attributes that are supposed to be used to update the dynamic runtime attribute is in the nonexistent state, then all information that is known is used. The TransportID of the I_T_L nexus and the target port identifier of the I_T_L nexus is always known from the transport layer and are presented as target type attributes.

4.26.5.1 Dynamic Runtime Information Lifetime

Dynamic Runtime Attributes are maintained separate from the device server's management of I_T nexuses. The I_T_L nexus identifying information (see 6.2.2.2.1) remains in existence inside dynamic runtime attributes even after the I_T nexus referenced is no longer known by the SCSI target port.

Logical unit type attributes (see 6.2.2.2) are created or destoryed as described in the description of each logical unit type attribute.

Target type attributes (see 6.2.2.3) are created by the drive if it detects communication from a new I_T nexus (e.g., Fibre Channel PLOGI/PRLI sequence) and are destroyed by the drive if it detects the disappearance of the I_T nexus (e.g., Fibre Channel LOGO).

Initiator type attributes (see 6.2.2.4) are created if a WRITE DYNAMIC RUNTIME ATTRIBUTE command is received requesting a new attribute and are destroyed if:

a) a WRITE DYNAMIC RUNTIME ATTRIBUTE command is received indicating that attribute with a long of zero;
b) the drive detects the destruction of the target type attribute associated with the I_T nexus that created the Initiator type attribute; or
c) for vendor-specific reasons (e.g. resource management).

Dynamic Runtime Attributes do not persist across a device power off.

4.27 Recommended access order (RAO) (beginning with E07)

4.27.1 Recommended access order overview

A feature of the 3592E07 and later drives is the ability to accept a list of User Data Segments (see 4.27.1.1) and reorder those User Data Segments into a recommended access order that minimizes the time to read those User Data Segments. A User Data Segment (UDS) is defined as a grouping of contiguous logical objects (i.e., logical blocks and filemarks) and is described by partition number, beginning logical object identifier, and ending logical object identifier.

4.27.1.1 User data segments in a partition

Within a partition that has recorded logical objects a contiguous sequence of logical blocks or logical files may be referenced as a user data segment (UDS).

4.27.1.2 User data segment descriptors

UDS descriptors (see 5.2.29.1.3) are used to describe attributes of the UDS and contain the following:

a) an application client specified name;
b) a partition number;
c) a beginning logical object identifier;
d) an ending logical object identifier;
e) in a returned RAO list, an estimate of the time to locate from the end of the current UDS to the beginning of the next UDS to access. This does not include the time required to read the UDS as this has variability dictated by the application, the load on the server, and other unknown factors; and
f) optionally, in a returned RAO list, the physical geometry of the UDS if requested in the GRAO command.

4.27.2 Recommended access order features

A device server may accept a list of User Data Segments (see 4.27.1) in a GRAO parameter list and reorder those UDS into a recommended access order that reduces the time required to read the list of UDS by minimizing the time to locate to each UDS in the list.

The list of UDS is sent to the drive using the Generate Recommended Access Order (GRAO) command (see 5.2.8). The GRAO command allows the user to specify the process to use in generating the Recommended Access Order (RAO) list. The GRAO command supports thousands of User Data Segments in the list. The RRAO command (see 5.2.29) with the UDS_LIMITS bit set to one may be used to determine the number of supported UDS's for the type of RAO list to generate (i.e., for the specific setting of the UDS_TYPE field) as well as the maximum size of each UDS for the type of RAO list to generate. At the time this document was published, the maximum number of supported UDS's for all values of UDS_TYPE is 2730.

The drive processes the GRAO command and returns GOOD status when the RAO list is complete. The RAO list may then be read with one or more Receive Recommended Access Order (RRAO) commands (see 5.2.29). The RRAO command allows retrieval of the ROA list with or without geometry information. The RRAO command may request the number of supported UDS in the ROA list.

The processes for generating a recommended access order list are defined in table 36

Table 36 — PROCESS for generating recommended access order

Value	Description
000b	Drive does not reorder the UDS's passed in the GRAO parameter list, and does not calculate the estimated locate time to UDS (see page 174) but instead sets the ESTIMATED LOCATE TIME TO UDS field in each of the UDS descriptors of the RAO list to the estimated time it would take to locate from the position of the medium at the time the GRAO command is processed to the beginning of that UDS (i.e., all the times are calculated from the same starting position). WARNING This process is not supported. WARNING
001b	Drive does not reorder the UDS's passed in the GRAO parameter list, but does calculate the estimated locate time to UDS (see page 174) for each UDS in the list.
010b	Drive reorders the UDS's passed in the GRAO parameter list into the recommended access order and calculates the estimated locate time to UDS (see page 174) for each UDS in its resultant position.
011b-111b	Reserved

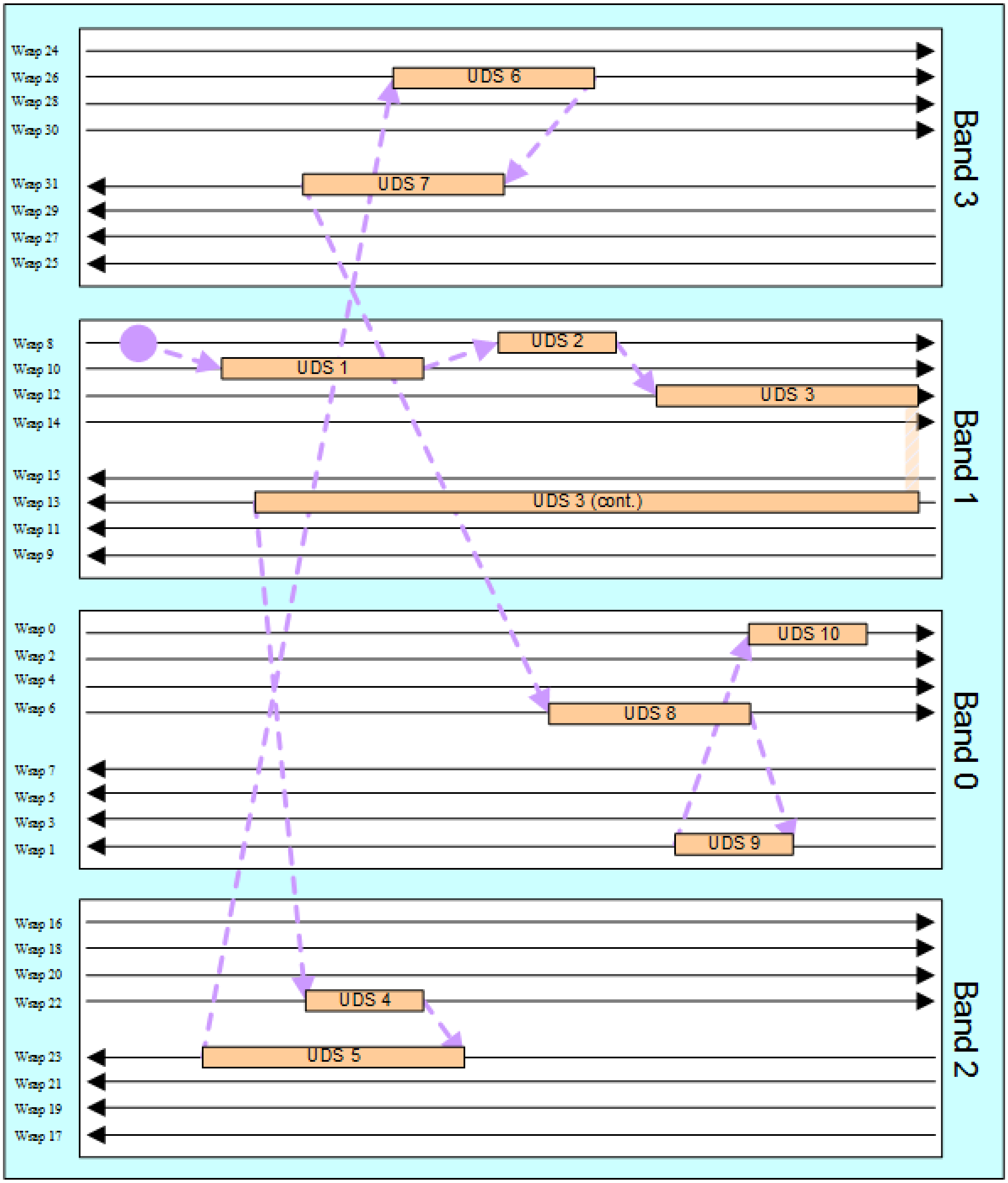
4.27.3 Recommended access order usage

The RRAO command returns the RAO list generated in the last successful GRAO command. The RAO list that is generated is valid for the state of the currently mounted volume (i.e., logical position, logical objects on media, etc.) at the time the list is generated. If the logical position of the medium is changed, or if logical objects are written or erased, then the RAO list becomes out of date. However, the device server takes no action to invalidate the list or to enforce a specific sequence of operation before returning an RAO list. Therefore, the responsibility of ensuring the RAO list has not been invalidated by commands since the processing of the GRAO command rests with the application.

An example of how an application client may use the recommended access order model (see 4.27.2) is to:

1) Read the UDS limits (see 5.2.29) to determine the number of supported UDS's (see 4.27.1.1);
2) Compose a list of UDS's to be accessed;
3) Generate an RAO list (see 4.27.2) from the list of UDS's to be accessed using the GRAO command (see 5.2.8);
4) Read a portion of the RAO list using the RRAO command (see 5.2.29) with the RAO LIST OFFSET field set to zero and the ALLOCATION LENGTH field set as appropriate for the Data-In Buffer;
5) Check the RAO PROCESS field and the STATUS field of the RAO list (see 5.2.29.1.2) to confirm that the RAO list was generated as expected;
6) Locate to the UDS described in the first user data segment descriptor of this portion of the RAO list;
7) Read that UDS;
8) Locate to the UDS described in the next user data segment descriptor of this portion of the RAO list;
9) Read that UDS;
10) Repeat steps 8) and 9) for all user data segment descriptors returned in this portion of the RRAO list;
11) if the value in the RAO list RAO DESCRIPTOR LIST LENGTH field returned in step 4) is larger than the sum of the value in the RAO LIST OFFSET field and the size of the portion of the RAO list returned in response to the RAO command, then read another portion of the RAO list using the RRAO command with the RAO LIST OFFSET field and the ALLOCATION LENGTH field set as appropriate for the Data-In Buffer;
12) Repeat steps 6) through 11) as necessary until all UDS's have been read.

An example of the resulting order after a GRAO with the RAO PROCESS set to 010b is shown in figure 12. This simplified example only shows 8 wraps per data band for a total of 32 wraps.



Note: The UDS numbers represent the sorted output order (i.e. the recommended access order). The sort is based on the current position (i.e., the light purple circle in the figure). The order of the resultant RAO list is independent of the order in the Generate Recommended Access Order parameter list.

Figure 12 — Example Reordering of UDS's

4.27.3.1 User data segment geometry usage

If the RAO list generated contains UDS's with geometry (see 5.2.8), then the geometry descriptors (see 5.2.29.1.3.1) may be used to build a representation of the physical layout of the UDS's on tape. This may be useful for visual feedback or for an application to create its' own algorithm for UDS retrieval based on physical location.

4.28 Extended Copy (i.e., Tape to tape copy) (E08 and later devices)

4.28.1 Overview

Some versions of this device support a sub-set of the T10/SPC-4 third-party copy feature. This support is indicated by the 3PC bit in Inquiry Standard Data (see 5.2.9.1 on page 117). Additionally, tape specific extensions were proposed for inclusion into T10/SPC-4 and IBM is working to get them accepted as part of the standard. This extended copy feature may be used to directly copy sequences of tape logical objects (i.e., logical blocks and/or filemarks) directly from one tape drive to another, without moving the data through a host system thereby reducing the load on the host. This device does not support some features described in SPC-4, such as:

a) byte stream oriented copy sequences;
b) inline copy data;
c) held data;
d) RODs; or
e) non-tape devices.

Supported features and implementation considerations are detailed throughout this document. Applications may discover the supported commands and features by querying the IP 8Fh: Third-party Copy (beginning with E08) (see 6.3.6 on page 271) vital product data inquiry page.

The source tape drive, destination tape drive, source volume, and destination volume of a copy segment operation may be disparate types or generations. Only the hosting drive needs to have extended copy support. The other drive only needs to have basic T10 SSC command set capabilities (check with IBM for which drives are officially supported). The drives must be able to communicate with each other via the host interface. The copy manager is hosted within a supporting device and that supporting device must be either the source or destination of each copy segment operation.

This device’s copy manager adds a few extensions to ensure the integrity of the data transfers. These include:

a) automatically using T10/SSC-4 logical block protection (LBP) during the copy step(s) when supported by both the source drive and the destination drive; and
b) checking the positioning between segment processing.

If during a copy a logical block is encountered on the copy source which is larger than the maximum stream device transfer size (i.e., 0020 0000h), then the copy is failed with COPY ABORTED, COPY TARGET DEVICE DATA OVERRUN (A/0D04). If the copy source contains the copy manager, then the ending source logical position is on the BOP side of that logical block. If the copy source does not contain the copy manager, then the ending source logical position is on the EOP side of that logical block.

The SILI bit is typically set to one by this copy manager.

This device terminates a copy operation if the copy manager receives a response containing a sense key of RECOVERED error from a device that does not contain the copy manager. For this reason it is recommended that the PER bit of the Read-Write Error Recovery mode page (see 6.6.5) be set to zero on the devices that do not contain the copy manager which is requested to perform the extended copy.

It is recommended that the drive which is the destination of the copy data is also the device which contains the copy manager (i.e., the XCOPY command is issued to the destination drive). If the copy manager is contained in the destination device, then the destination device implicitly protects against commands not associated with the copy operation from modifying or changing the logical position of the destination volume. If the copy manager is not contained in the destination device, then the implicit protection provided by the blocking of the command queue does not exist.

The following ordered list of actions is an example of using extended copy with different configurations as shown in figure 13 with numbers that refer to the steps in this list):

1) Any pre-copy setup is performed on both drives as described in Pre-copy setup (see 4.28.3.1 on page 93);
2) The EXTENDED COPY (LID4) (i.e., XCOPY) – 83h[01h] (see 5.2.5 on page 110) command is then sent to the drive containing the copy manager;
3)
    A) The copy manager performs the operations defined by the segment descriptors as described in XCOPY command processing (see 4.28.4.2 on page 95); and
    B) The application client monitors the operations (see 4.28.3.3);

    and
4) The application client performs any cleanup (see 4.28.3.5).

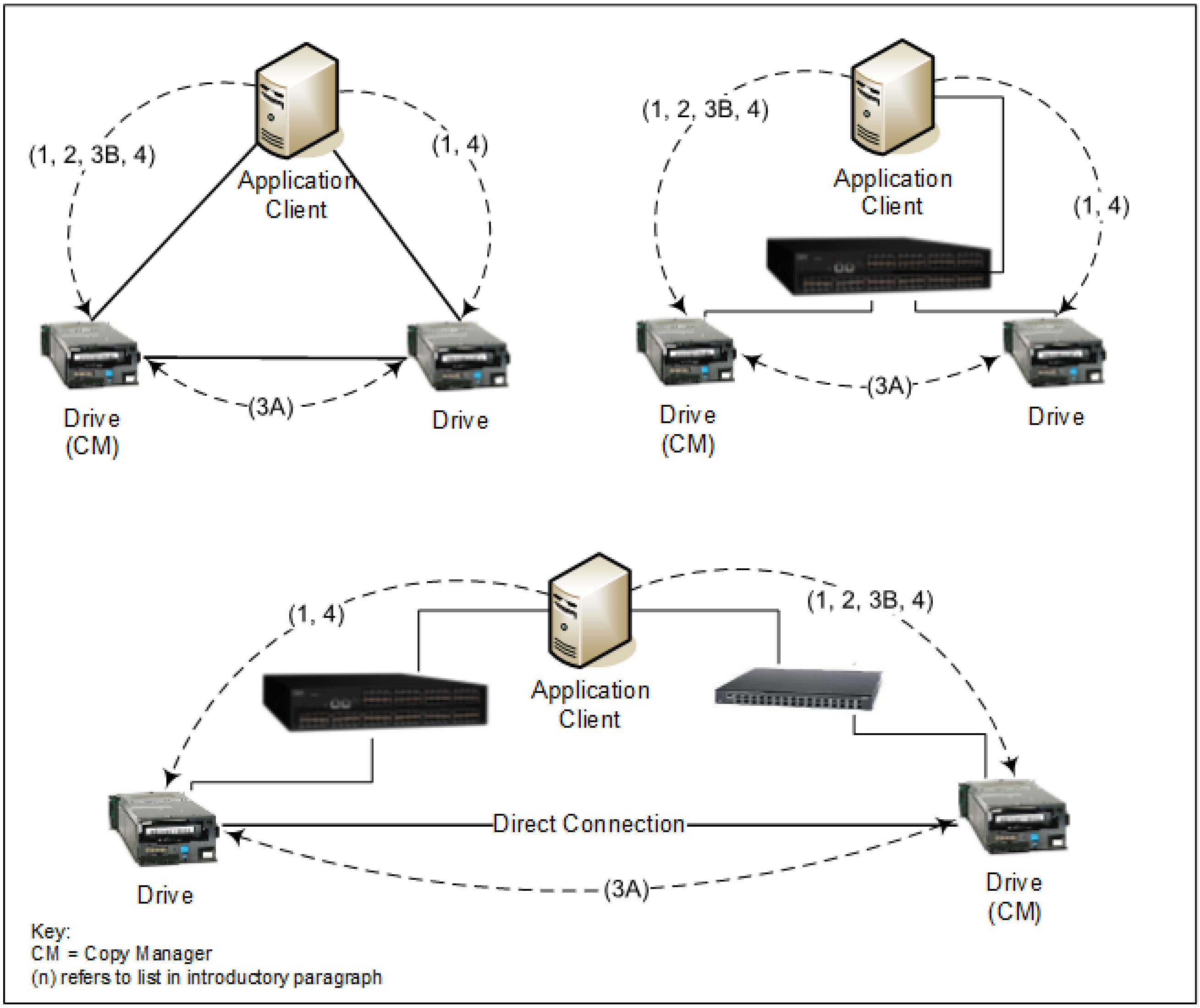


Figure 13 — Example configuration of devices for Extended Copy

4.28.2 Extended copy commands and parameters supported by this device

The commands and parameters associated with extended copy are:

a) the IP 8Fh: Third-party Copy (beginning with E08) (see 6.3.6 on page 271) vital product data inquiry page;

b) the EXTENDED COPY (LID4) (i.e., XCOPY) – 83h[01h] (see 5.2.5 on page 110);
c) the RECEIVE COPY STATUS(LID4) – 84h[05h] (see 5.2.27 on page 167); and
d) the COPY OPERATION ABORT – 83h[1Ch] (see 5.2.3 on page 107).

4.28.3 Extended copy command usage

4.28.3.1 Pre-copy setup

Before XCOPY commands (see 5.2.5) are sent to the copy manager, the application client is required to prepare both the copy source and the copy destination. This preparation may include:

a) loading, mounting, and positioning volumes;
b) configuring device and/or volumes (e.g., sending MODE SELECT commands);
c) reserving devices with persistent reservations; and
d) sending synchronizing commands (e.g., WRITE FILEMARK) prior to the copy if buffer synchronization is important to the application client.

4.28.3.2 Issuing the XCOPY command

4.28.3.2.1 Issuing the XCOPY command overview

After any desired pre-copy setup is performed by the application client, an XCOPY command specifying the desired copy functions (see 4.28.4) is sent to the copy manager. The XCOPY command is processed and a copy operation is started as specified in 4.28.4 on page 94.

4.28.3.2.2 Identifying copy operations originated by XCOPY commands

It is recommended that application clients set the IMMED bit to one in the XCOPY command parameter data due to the long processing time.

The EXTENDED COPY (LID4) (i.e., XCOPY) – 83h[01h] (see 5.2.5 on page 110) command contains a LIST IDENTIFIER field which uniquely identifies the copy operation. The list identifier is used by the RECEIVE COPY STATUS(LID4) – 84h[05h] (see 5.2.27 on page 167) command and the COPY OPERATION ABORT – 83h[1Ch] (see 5.2.3 on page 107) command to identify the copy operation originated by an XCOPY command.

The list identifier is used to uniquely identify the copy operation from the time it is started until the next copy operation is started.

4.28.3.3 Monitoring copy operations

The status and results of a copy operation (see 4.28.3.2.2) may be retrieved using the RECEIVE COPY STATUS(LID4) command (see 5.2.27).

The contents of the COPY OPERATION STATUS field (see table 99 in 5.2.27.1) in the parameter data header indicate the current processing status of the specified copy operation.

4.28.3.4 Copy operation aborts

4.28.3.4.1 Copy operation aborts overview

Copy operations (see 4.28.3.2.2) may be aborted as a result of:

a) a COPY OPERATION ABORT – 83h[1Ch] (see 5.2.3 on page 107);
b) a task management function;
c) a PERSISTENT RESERVE OUT command (see 5.2.17) with a service action of PREEMPT AND ABORT; or
d) certain SCSI events.

All commands and data transfers generated by copy operations are terminated and are no longer transferring data upon completion of an abort.

4.28.3.4.2 Copy operation aborts as a result of the COPY OPERATION ABORT command

The COPY OPERATION ABORT – 83h[1Ch] (see 5.2.3 on page 107) provides an abort capability that is equivalent to an ABORT TASK task management function for any copy operation without regard for whether the copy operation is being performed in the foreground (i.e., a command exists for the copy operation) or in the background (i.e., the XCOPY command set the IMMED bit to one). If the copy operation was originated by an XCOPY command with the IMMED bit set to one, then that copy operation is unable to be aborted using a task management function and is only able to be aborted using the other methods described (e.g., the COPY OPERATION ABORT command, etc).

The copy operation specified by a COPY OPERATION ABORT command is aborted and GOOD status is returned upon successful completion of the abort.

If the copy operation specified by a COPY OPERATION ABORT command specifies a foreground copy operation, then the XCOPY command is also terminated with the sense key set to COPY ABORTED and the additional sense code set to COMMAND CLEARED BY DEVICE SERVER.

4.28.3.4.3 Copy operation aborts as a result of a task management function

Certain task management functions abort a copy operation which is running in the foreground (i.e., the XCOPY command has the IMMED bit set to zero). These task management functions are:

a) ABORT TASK;
b) ABORT TASK SET; and
c) CLEAR TASK SET.

4.28.3.4.4 Copy operation aborts as a result of PERSISTENT RESERVE OUT—PREEMPT AND ABORT

A PERSISTENT RESERVE OUT command (see 5.2.17) with a service action of PREEMPT AND ABORT performs an abort on each copy operation (i.e., does the equivalent of a COPY OPERATION ABORT to each copy operation) regardless of whether the copy operation is running in the foreground or background.

4.28.3.4.5 Copy operation aborts as a result of SCSI events

Certain SCSI events, if detected, abort copy operations. These SCSI events that perform the equivalent of a COPY OPERATION ABORT on each copy operation are:

a) Logical unit reset;
b) Hard reset; and
c) Power on.

4.28.3.5 Extended copy cleanup

After the copy operation completes (see 4.28.3.3), the application client performs any desired cleanup which may include:

a) synchronizing any data that may be left in the copy destination's buffer;
b) demounting and unloading volumes; and
c) releasing any reservations.

4.28.4 The EXTENDED COPY (XCOPY) command

4.28.4.1 XCOPY parameter list

The XCOPY command as supported by this device is the EXTENDED COPY(LID4) command (see 5.2.5). The XCOPY command has a parameter list which contains the copy source, copy destination, and segment descriptors that specify the copy operation. T10/SPC-4 has a mechanism for inline data, but this device does not support that feature and consequently expects the length field of the inline data to be zero.

The header and CSCD descriptors in an XCOPY command are used to provide information to be used by the segment descriptors.

The segment descriptors specify copy functions and the nature of processing (e.g., number of blocks to read and write, where to position medium, etc.) to be performed by the copy manager. Processing requirements specific to each segment descriptor are contained in the section describing the descriptor (see 6.9.2.1).

4.28.4.2 XCOPY command processing

The XCOPY command processing occurs in two stages. The first stage is the parameter list qualification which occurs prior to performing the processing requested by the segment descriptors.

The copy manager performs implicit verify operations prior to processing any segments and determines whether a TEST UNIT READY check should be performed based on all the segments usage for each given CSCD.

Once this qualification is completed, the processing requested by the segment descriptors begins. This processing requested by the segment descriptors is called the copy operation. If the IMMED bit is set to one, then status is returned to the XCOPY command before the copy operation is started. If the IMMED bit is set to zero, then status is returned to the XCOPY command after the copy operation has completed.

4.28.4.2.1 Copy operation

A copy operation is the processing requested by the list of segment descriptors in an XCOPY command. The copy operation performs its processing in the foreground (i.e., during the XCOPY command) if the IMMED bit is set to zero or in the background (i.e., after status has been returned to the XCOPY command) if the IMMED bit is set to one.

A copy operation begins after the parameter list qualification completes and ends after all segments in the parameter list that are going to be processed have been processed (i.e., if an error occurs, it may be that one or more segments are not processed).

Writes are performed using whole-block transfer lengths determined by the block size read from the source.

The copy operation may change the logical position of the copy source and/or copy destination.

The copy operation itself does not explicitly synchronize the data. Therefore, if buffer synchronization is important to the application client, then actions to synchronize the data should be explicitly taken by the application client either before issuing the XCOPY command, as a segment in the XCOPY command, or after the copy operation.

Buffering on the copy destination is affected by the location of the copy manager and may be affected by the settings of the mode parameter header:

a) if the copy manager is contained in the destination device, then the data is buffered;
b) if the copy manager is not contained in the destination device and the BUFFER MODE field in the mode parameter header is managed on a device basis, then the data is buffered according to the settings of the BUFFER MODE field in the mode parameter header; and
c) if the copy manager is not contained in the destination device and the BUFFER MODE field in the mode parameter header is managed on a per I_T nexus basis, then the data is buffered according to the device default buffer mode.

Errors that occur during a copy operation set the sense key to COPY ABORTED.

4.28.4.2.2 XCOPY command errors detected before processing of the copy operation

This device qualifies the parameter list prior to starting the copy operation. This qualification also occurs prior to returning status to an XCOPY command with the IMMED bit set to one. Errors detected in this phase are rejected with a CHECK CONDITION status to the XCOPY command and sense data to indicate the error.

4.28.4.2.3 XCOPY command errors detected during processing of the copy operation

Once the copy manager begins processing copy operations (i.e., after completion of the parameter data qualification and, if an immediate command, the return of GOOD status) there may be errors not related to the qualification of the parameter data structures. These errors may result in the copy operation (see 4.28.3.2.2) being terminated with:

a) COPY ABORTED, THIRD PARTY DEVICE FAILURE (A/0D01) if a CHECK CONDITION or other unexpected status from a CSCD is received. The status and sense data, if any, associated with the exception condition is copied into the sense data as described later in this section;
b) COPY ABORTED, COPY TARGET DEVICE NOT REACHABLE (A/0D02) if it is not possible to complete processing of a segment because the copy manager is unable to communicate with a CSCD;
c) COPY ABORTED, INCORRECT COPY TARGET DEVICE TYPE (A/0D03) if the device type does not match the type in the CSCD descriptor;
d) COPY ABORTED, THIRD PARTY DEVICE FAILURE (A/0D01) if a command other than INQUIRY is issued by the copy manager to a CSCD and the CSCD fails to respond;
e) COPY ABORTED, COPY TARGET DEVICE DATA UNDERRUN (A/0D04) if a CSCD responds with less data than expected;
f) COPY ABORTED, COPY TARGET DEVICE DATA OVERRUN (A/0D05) if a CSCD responds with more data than expected.

After an exception condition is detected during a copy operation:

a) the copy manager indicates the segment that was being processed at the time of the exception by writing the segment number to the third and fourth bytes of the COMMAND-SPECIFIC INFORMATION field. The segment number is based on the relative position of the segment descriptor in the parameter list (see 4.28.4.1) (i.e., the first segment descriptor in the parameter list is assigned descriptor number zero, the second is assigned one, etc.);
b) if the segment being processed at the time the error occurred indicates data to be written, and the residual count is able to be determined, then the residual count for the segment is placed in the INFORMATION field, and the VALID bit is set to one. If the residual count is unable to be determined, then the copy manager indicates that the INFORMATION field does not contain valid data by setting the VALID bit to zero in the sense data. Segment descriptors that do not specify a transfer count do not have a valid residual count.

   The residual count is computed by subtracting the number of units successfully written during the processing of the current segment from the number of units which would have been written if all commands had completed with GOOD status and all READ commands had returned the full data length requested. While computing the residual count, the copy manager includes only the results of commands successfully completed by a copy destination.

   The units in which to report the residual count is determined by the following ordered list:

   1) if the segment being processed is ECD 18h: Tape device logical object copy function (see 6.9.2.1.2.10 on page 513) with the CODE field set to:
      a) a value of 000b, then in units of logical objects;
      b) a value of 001b, then in units of logical blocks; or
      c) a value of 010b, then in units of logical files;

   or

   2) in units of copy destination blocks;
c) if the exception condition is reported by the copy source, then the first byte of the COMMAND-SPECIFIC INFORMATION field is set to the starting byte number, relative to the first byte of sense data, of an area that contains the status byte and sense data delivered to the copy manager by the copy source. The status byte and sense data are not modified by the copy manager. A zero value indicates that no status byte and sense data is being returned for the copy source;
d) if the exception condition is reported by the copy destination, then the second byte of the COMMAND-SPECIFIC INFORMATION field is set to the starting byte number, relative to the first byte of sense data, of an area that contains the status byte and sense data delivered to the copy manager by the copy destination. The status byte and sense data are not modified by the copy manager. A zero value indicates that no status byte and sense data is being returned for the copy destination;

e) if segment processing is terminated because a CSCD is unreachable or as the result of a failure in a command sent to a CSCD, then the SENSE-KEY SPECIFIC field is set as described in SPC-4 for Segment pointer sense key specific information, with the FIELD POINTER field indicating the first byte of the CSCD descriptor that specifies the CSCD; and
f) the copy manager preserves the status of the copy operation for the RECEIVE COPY STATUS(LID4) – 84h[05h] (see 5.2.27 on page 167).

The sense data format resulting from this behavior is described in Sense data format for COPY ABORTED (i.e., Sense Key Ah) (see 6.8.3 on page 491).

5. SCSI Commands

This chapter describes the SCSI commands supported (LUN 0).

The following SCSI command descriptions have a table describing the fields in the Command Descriptor Block (CDB), similar to the style used in the *American National Standard of the National Committee for Information Technology Standards (NCITS)* documents.

Any data required by each command follow these descriptions and are described in a “term-definition” format. In this format, the bits or bytes to be described are highlighted and listed on the left. The definition for the bits or bytes is to the right (not highlighted).

Parameters are described in *6.*

5.1 SCSI Commands Overview

5.1.1 Unsupported SCSI Commands

Certain commands or features of some commands defined in SPC-4 or SSC-3 are not currently supported but may be in the future.

The *Read Buffer* and *Write Buffer* commands are supported but not all buffers are described in this document because many buffers are intended only to be read or written by the Service Representative or by Manufacturing. OEM customers who intend to support host microcode download on a new platform should contact IBM for a complete description of the *Write Buffer* command for this purpose. Note that new microcode may also be loaded without requiring the use of the SCSI *Write Buffer* command, by using the Field Microcode Replacement (FMR) tape process described in the maintenance information manual for this product.

5.1.2 Supported SCSI Commands

The 3592 tape drive accepts commands on LUN 0 for drive related commands (i.e., Peripheral Device Type returned in Standard Inquiry is Sequential Access Device). When in a library the 3592 tape drive also accepts commands on LUN 1 for library related commands (i.e., Peripheral Device Type returned in Standard Inquiry is Medium Changer Device).

5.1.2.1 Supported SCSI Commands on LUN 1

The commands supported on LUN 1 is dependant on the Library in which the drive is contained. To determine the model of the library in which the drive is contained use the Inquiry command. Bytes 8-15 contain the T10 Vendor Identification. Bytes 16 - 31 contain the Product Identification.

This document contains no additional information about commands supported on LUN 1. For a list of all commands supported please see the appropriate Library SCSI Reference.

5.1.2.2 Supported SCSI Commands on LUN 0

Table 37 provides a list of all commands supported by this product for the sequential access devices (i.e., LUN 0). For each command, the operation code, reference page for this specification, recommended host command time-out, type of support required for the command as defined by the SCSI-3 standard, and applicability of certain conditions to the command are shown in table 37.

It is strongly recommended that device drivers or host software implement device reservations using the Reserve or Persistent Reserve commands. Due to the sequential nature of tape devices, many host commands are serialized, and command time-outs consequently have an additive effect. Using reservations will prevent this from causing application disruptions in a multi-initiator or SAN environment. Similar additive time-out effects can occur if the host is using command queuing (simple queuing).

Command time-out values may be found in the command description for REPORT SUPPORTED OPERATION CODES - A3h (beginning with E07) (see 5.2.34 on page 184)

Table 37 — Drive Commands (LUN 0) (part 1 of 3)

Command Name	SCSI	Op Code	Service Action	See Page	Applicable Conditions:							
					RVC[α]	UAT	NRD	WRP	MFC	DCC	DEA	CK1[ν]
Allow Overwrite	SSC-4	82h	-----	104	Y	Y	Y	-	Y	-	-	Y
Change Definition	SCSI-2	40h	-----	106	Y	Y	-	-	-	-	-	-
Copy Operation Abort	SPC-4	83h	001Ch	107	Y	Y	-	-	-	Y	Y	Y
Display Message	VU	C0h	-----	108	Y	Y	-	-	-	-	-	Y
Extended Copy(LID4) (i.e., XCOPY)	SPC-4	83h	0001h	110	Y	Y	-	-	Y	Y	Y	Y
Erase (long = 0)	SSC	19h	-----	111	Y	Y	Y	Y	Y	Y	Y	Y
Erase (long = 1)	SSC		-----									
Format Medium	SSC-4	04h	-----	112	Y	Y	Y	Y	Y	Y	Y	Y
Generate Recommended Access Order (GRAO)	SSC-5	A4h	001Dh	114	Y	Y	Y	-	Y	Y	N	Y
Inquiry	SPC	12h	-----	117	-	-	-	-	-	-	-	-
Load/Unload	SSC	1Bh	-----	122	Y	Y	Y[d]	-	Y[e]	Y	Y	Y
Locate	SSC	2Bh	-----	124	Y	Y	Y	-	Y	Y	Y	Y
Log Select	SPC	4Ch	-----	126	Y	Y	-	-	-	-	-	Y
Log Sense	SPC	4Dh	-----	128	Y	-	-	-	-	-	-	-
Mode Select(6/10) (not page 23h)	SPC	15h/55h	-----	129	Y	Y	-	-[i]	-	Y[h]	-	Y
Mode Select (6/10) (page 23h)	SPC		-----									
Mode Sense (6/10)	SPC	1Ah/5Ah	-----	131	-	Y	-	-	-	-	-	Y
Persistent Reserve In	SPC	5Eh	-----	133	-	Y	-	-	-	-	-	-
Persistent Reserve Out	SPC	5Fh	-----	136	-[a, b]	Y	-	-	-	-	-	-
Prevent Allow Medium Removal	SSC	1Eh	-----	138	Y	Y	-	-	-	-	-	Y
Read	SSC	08h	-----	139	Y	Y	Y	-	Y	Y	Y	Y
Read (String Search) (not J1A)	VU	08h	-----	140	Y	Y	Y	-	Y	Y	Y	Y
Read Attribute	SPC	8Ch	-----	141	Y	Y	-	-	-	Y	-	-
Read Block Limits	SSC	05h	-----	146	Y	Y	-	-	-	-	-	Y
Read Buffer	SPC	3Ch	-----	148	Y	-	-	-	-	-	-	-
Read Dynamic Runtime Attributes	SSC	A3h	001Eh	154	-	-	-	-	-	-	-	-
Read Dynamic Runtime Attributes	VU	D1h	-----	154	-	-	-	-	-	-	-	-
Read Position	SSC	34h	-----	159	Y	Y	-	-	-	-	-	Y
Read Reverse	SSC	0Fh	-----	166	Y	Y	Y	-	Y	Y	Y	Y
Receive Copy Status (LID4)	SPC-4	84h	0005h	167	-	-	-	-	-	-	-	Y
Receive Diagnostic Results	SPC	1Ch	-----	170	Y	Y	-	-	-	-	-	Y
Receive Recommended Access Order (RRAO)	SSC-5	A3h	001Dh	171	Y	Y	Y	-	Y	Y	N	Y
Recover Buffered Data	SSC	14h	-----	177	Y	Y	Y[o]	-	-	Y	Y	Y

Table 37 — Drive Commands (LUN 0) (part 2 of 3)

Command Name	SCSI	Op Code	Service Action	See Page	Applicable Conditions:							
					RVC[a]	UAT	NRD	WRP	MFC	DCC	DEA	CK1[v]
Release Unit (6)[p]	SPC	17h	-----	178	-[b]	Y	-	-	-	-	-	-
Report Density Support	SSC	44h	-----	179	Y	Y	-	-	-	-	-	Y
Report LUNs	SPC	A0h	-----	183	-	-	-	-	-	-	-	-
Report Supported Operation Codes	SPC	A3h	000Ch	184	-	-	-	-	-	-	-	-
Report Timestamp	SPC	A3h	000Fh	196	-	Y	-	-	-	-	-	-
Request Sense	SPC	03h	-----	198	-	-	-	-	-	-	-	-
Reserve Unit (6)[q]	SPC	16h	-----	199	Y[c]	Y	-	-	-	-	-	-
Rewind	SSC	01h	-----	200	Y	Y	Y	-	Y[e]	Y	Y	Y
Security Protocol In (SPIN)	SPC	A2h	-----	201	y	y	-	-	-	-	-	y
Security Protocol Out (SPOUT)	SPC	B5h	-----	202	y	y	-	-	-	-	-	y
Send Diagnostic	SPC	1Dh	-----	203	Y	Y	Y[j]	-	Y[f]	-	-	Y
Set Timestamp	SPC	A4h	000Fh	205	Y	Y	-	-	-	-	-	Y
Space(6/16)	SSC	11h/91h	-----	207	Y	Y	Y	-	Y	Y	Y	Y
Space(6/16) (String Search)	VU	11h/91h	-----	209	Y	Y	Y	-	Y	Y	Y	Y
String Search	VU	E3h	-----	211	Y	Y	Y	-	-	Y	Y	-
Test Unit Ready	SPC	00h	-----	213	Y	Y	Y	-	-	Y[k]	-	Y
Verify (VTE=0 and VTF=0)	SSC	13h	-----	214	Y	Y	Y	-	Y	Y	Y	Y
Verify (VTE=1 or VTF=1)	SSC		-----									
Write	SSC	0Ah	-----	216	Y	Y	Y	Y	Y	Y	Y	Y
Write Attribute	SPC	8Dh	-----	217	Y	Y	-	-	-	Y	-	-
Write Buffer	SPC	3Bh	-----	220	Y	Y	-	-	-	-	-	-
Write Dynamic Runtime Attributes	SSC	A4h	001Eh	223	-	-	-	-	-	-	-	-
Write Dynamic Runtime Attributes	VU	D2h	-----	223	-	-	-	-	-	-	-	-
Write Filemarks	SSC	10h	-----	226	Y	Y	Y	Y	Y	Y	Y	Y

Table 37 — Drive Commands (LUN 0) (part 3 of 3)

Command Name	SCSI	Op Code	Service Action	See Page	Applicable Conditions:							
					RVC[a]	UAT	NRD	WRP	MFC	DCC	DEA	CK1[n]

Legend

M	Mandatory
O	Optional
VU	Vendor-Unique
-	Not Applicable
NS	Not Supported
SPC	SPC-n
SSC	SSC-n

RVC	Reservation Conflict status
UAT	CHECK CONDITION status for Unit Attention
NRD	CHECK CONDITION status for Not Ready
WRP	CHECK CONDITION status for Write Protected
MFC	CHECK CONDITION status for Medium Format Corrupted
DCC	Deferred CHECK CONDITION
CK1	CHECK CONDITION B/4400 for Post Check 1 State
DEA	Deferred Error (DCC) affinity (see 4.13.3)
Y	Yes (Condition applies)
Y^n	Yes (Condition applies per note n below)

Notes:

[a] If an I/O process consists of linked commands and begins with a command that is not subject to the RVC condition, subsequent commands in the I/O process may be subject to Reservation Conflict status, if a linked command is subject to the RVC condition and a reservation conflict exists. Linked commands are not supported by this device.

[b] Performs no operation if the logical unit is reserved to another initiator.

[c] Condition applies if the logical unit is reserved to another initiator.

[d] CHECK CONDITION status for a not ready device is not presented to a Load Unload command that requests the load function when a volume is present. CHECK CONDITION status is presented to a Load Unload command that requests the unload function when a volume is not present.

[e] The command is not subject to the condition unless the medium format corrupted condition has not yet been reported to the initiator on some prior command.

[f] This CHECK CONDITION is diagnostic dependent. Refer to the specific diagnostic in question.

[g] The medium must be contain a factory written servo format. If the media is completely blank (i.e., degaussed with a strong degausser) then it is rendered unusable and cannot be reformatted with the Format Medium command.

[h] The deferred CHECK CONDITION and CHECK CONDITION status for Write Protected only applies to “MP 23h: Medium Sense” on page 414

[i] Only “MP 23h: Medium Sense” on page 414 is treated as write protected. All other mode pages are not.

[j] Some diagnostics require either READY (cartridge loaded) or NOT READY (no loaded cartridge) states prior to their invocation. Refer to the specific diagnostic in question.

[k] Reporting of deferred CHECK CONDITION status for the Test Unit Ready command is optional based on a vendor-unique field in the CDB.

[l] These commands are not supported by this device.

[m] Reservation Conflict is reported as appropriate for the type of Service Action and Reservation Type requested, and the current reservation state of the drive.

[n] If the drive is in a Post Check 1 State (after a fatal error has occurred), the drive will repeatedly report B/4400 to these commands. This condition may be cleared by a Logical Unit or Target Reset. This insures that the application or driver acknowledges the condition and can gather and log any debug or error information.

[o] If a write error has occurred, the buffer state is unchanged (unwritten data is in the buffer), and Unload with Write Error Association is set to 1b (see “MP 24h: Initiator-Specific Extensions” on page 424), then these commands will not post a Not Ready condition, even if the drive is Not Ready (not loaded). The command will be processed in this case.

[p] If the drive sets the CRH bit to one in the parameter data returned by the REPORT CAPABILITIES service action of the Persistent Reserve In command, then in the presence of a Persistent Reservation, a Release(6) command will complete with GOOD status, but the persistent reservation will not be released, if the command is received from:
a) An I_T nexus that is a persistent reservation holder; or
b) An I_T nexus that is registered if a registrants only type persistent reservation is present.
In all other cases, the command will be processed as defined in SPC-2.

[q] A Reserve Unit (6) command will complete with GOOD status, but no reservation will be established and the persistent reservation will not be changed, if the command is received from:
a) An I_T nexus that is a persistent reservation holder; or
b) An I_T nexus that is registered if a registrants only type persistent reservation is present.
In all other cases, the command will be processed as defined in SPC-2.

5.1.3 Control Byte Definition

This description of the control byte fields is to be used for all of the supported commands. The control byte occurs in the last byte of a command, that is, byte 5 (6-byte commands), byte 9 (10-byte commands), byte 11 (12-byte commands), or byte 15 (16-byte commands). Table 38 shows the bit significance of the control byte.

Table 38 — Control Byte Definition

Bit Byte	7 msb	6	5	4	3	2	1	0 lsb
5, 9, or 11	Vendor Specific 00		Reserved 0000				Flag 0	Link 0

NOTE 17 - This device does not support command linking. Therefore both the Flag and Link bit are required to be 0b.

5.2 SCSI Commands Listing

5.2.1 ALLOW OVERWRITE - 82h (beginning with E07)

When Append-only mode (beginning with E07) (see 4.18.3 on page 60) is enabled an application client may issue the ALLOW OVERWRITE command to enable the overwrite of the medium at a non-append point. The processing of the ALLOW OVERWRITE command sets the allow_overwrite and allow_overwrite_position variables as specified in 4.18.3.

Table 39 — ALLOW OVERWRITE command

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	OPERATION CODE (82h)							
1	Reserved							
2	Reserved				ALLOW OVERWRITE			
3	PARTITION							
4	(MSB)	LOGICAL OBJECT IDENTIFIER						
...								
11								(LSB)
12	Reserved							
13								
14								
15	CONTROL							

The ALLOW OVERWRITE field specifies what type of overwrite is allowed. Table 40 defines the actions for the value specified in the ALLOW OVERWRITE field.

Table 40 — ALLOW OVERWRITE field definition

Value	Definition
0h	The allow_overwrite variable shall be set to Disabled
1h	The allow_overwrite variable shall be set to Current Position
2h	The allow_overwrite variable shall be set to Format
3h-Fh	The command shall be rejected with CHECK CONDITIONstatus with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

If the ALLOW OVERWRITE field is set to Current Position (i.e., 1h), then:

a) the PARTITION field is set to the active partition; and
b) the LOGICAL OBJECT IDENTIFIER field is set to the current position.

If the ALLOW OVERWRITE field is set to Current Position (i.e., 1h), then the allow_overwrite_position variable is set to the current position.

If the ALLOW OVERWRITE field is not set to Current Position (i.e., 1h), then the PARTITION field and LOGICAL OBJECT IDENTIFIER field are ignored.

The device server terminates the command with CHECK CONDITION status with the sense key set to ILLEGAL REQUEST and the additional sense code set to SEQUENTIAL POSITIONING ERROR if:

a) the ALLOW OVERWRITE field is set to Current Position (i.e., 1h); and
b) the logical position of the medium is not at the location specified by the PARTITION field and LOGICAL OBJECT IDENTIFIER field.

If the device server terminates that command with any status other than GOOD, then the allow_overwrite variable is set to Disabled and the allow_overwrite_position variable is set to invalid.

5.2.2 CHANGE DEFINITION - 40h

The Change Definition command is defined in SCSI-2 (see http://www.t10.org/cgi-bin/ac.pl?t=f&f=s2-r10l.pdf). This clause specifies the specific implementation.

Table 41 — Change Definition CDB

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Operation code (40h)							
1	Obsolete (LUN)			Reserved				
2	Reserved							Save
3	Reserved	Definition Parameter						
4	Reserved							
7								
8	Parameter Data Length							
9	Control Byte (see 5.1.3)							

The following parameters apply:

- Save: 0
- Definition Parameter:

  An initiator can request one of several operating definitions to be established. If the command is accepted, the new operating definition applies to all initiators. After a power on condition, the drive sets its operating definition to its default value as determined by non-volatile configuration. Any other hard reset condition does not affect the current operating definition.

  The operating modes available for this command are intended for manufacturing, engineering, or specialized controller or OEM environments. Contact IBM for additional information on these modes.

  Definition Parameter description:

Value	Description
00h	Use Current Operating Definition
60h	Clear Post Check 1 State

NOTE 18 - This parameter is used to clear the Post Check 1 State, which is repeatedly reported as a B/4400 to eligible commands (see Table 1 on page 6) until cleared by this command or by a LUN Reset or Target Reset action. The drive will report a 6/2900 if this state was cleared by this Change Definition, just in case any initiator had cleared but was unable to fully process its original unit attention. It should be noted that this is not a true bus reset and normal clearing effects may or may not have occurred.

Parameter Data Length: 00h

5.2.3 COPY OPERATION ABORT – 83h[1Ch]

The COPY OPERATION ABORT command (see table 42) is a third-party copy command (see 4.28.2) that requests that the copy manager abort the specified copy operation(see 4.28.4.2.1) as described in 4.28.3.4. This command terminates the copy operation as quickly as possible at a clean juncture (e.g., the copy manager waits until any outstanding command completes) before returning status.

Table 42 — COPY OPERATION ABORT command

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (83h)							
1	Reserved			SERVICE ACTION (1Ch)				
2	(MSB)							
…	LIST IDENTIFIER							
5								(LSB)
6								
…	Reserved							
14								
15	CONTROL							

The following parameters apply:

- LIST IDENTIFIER

  Specifies the copy operation to be aborted (see 4.28.3.2.2).

5.2.4 DISPLAY MESSAGE - C0h

Table 43 — Display Message CDB

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Operation code (C0h)							
1	Obsolete (LUN)			Reserved				
2	Reserved							
3								
4	Parameter List Length							
5	Control Byte (see 5.1.3)							

The Display Message command allows the initiator to use the device to display messages to the operator regarding the status and the needs of the device. Having this information at the device allows for more efficient use of the device.

The Display Message command is a vendor-unique command and, therefore, is not described in the SCSI standard. This document describes the fields in the Parameter list in general terms, such as the SCSI standard might, and follows each item with the device implementation of that field.

- Parameter List Length: 18h

  The parameter list length field specifies the length in bytes of the message display parameter list that is transferred from the initiator to the target.

The Display Message parameter list follows:

Parameter List

Byte **Description**

0 Display Type
This field is not meaningful. Any value is allowed (and ignored).

1 Message Processing
The Message Processing field specifies the types of messages and how to process the messages.

Bit **Description**

7-5 Message Type

Value	Description
0	General Status Message Message 0, Message 1, or both are displayed according to bits 4-2, until the drive next initiates tape motion or the message is updated with a new message.
1	Demount/Verify Message Message 0, Message 1, or both are displayed according to bits 4-2, until the current volume is unloaded. If the volume is currently unloaded, the message display is not changed and the command performs no operation.
2	Mount with Immediate Action Indicator Message 0, Message 1, or both are displayed according to bits 4-2, until the volume is loaded. An attention indicator is activated. If the volume is currently loaded, the message display is not changed and the command performs no operation.
3-6	Vendor-Reserved
7	Demount/Mount with Immediate Action Indicator When Message Control bits 4-2 are set to a value of 4 (100), Message 0 and Message 1 are displayed alternately until the currently mounted volume, if any, is

unloaded. When Message Control bits 4-2 are set to any other value, Message 0 is displayed until the currently mounted volume, if any, is unloaded. Message 1 is displayed from the time the volume is unloaded (or immediately, if the volume is already unloaded) until another volume is loaded. An attention indicator is activated.

4-2 Message Control

Value	**Description**
0	Display Message 0
1	Display Message 1
2	Flash Message 0
3	Flash Message 1
4	Alternate Message 0 and Message 1
5-7	Vendor-Reserved (Invalid)

The life and sequences of each message must interact with the requirements of other messages, both sent or internally generated by the device.

1-0 Vendor-Reserved

2-5 Vendor-Reserved

6-7 Message Length: 0010h

8-15 Message 0

Eight-character ASCII message. If both Message 0 and Message 1 consist entirely of blanks, all messages are cleared, except for ATTN, FID, and CLEAN messages.

The Message 0 field contains the data to be displayed. Characters in the message are limited to uppercase alphabetic, numeric, blank, and the following special characters:

@ $ #,. / ' ( ) * & + - =%: _ < >?; ¢

All lowercase alphabetic characters are converted to uppercase. All other characters not listed above, including nulls (00), are displayed as if they had been blanks. Real blanks (20) must be used to force the message clearing function described above.

16-23 Message 1 (see Message 0 description above)

5.2.5 EXTENDED COPY (LID4) (i.e., XCOPY) – 83h[01h]

5.2.5.1 XCOPY command introduction

The XCOPY command is the term used by this document to describe the EXTENDED COPY(LID4) command. The XCOPY command (see table 44) provides a means to copy data from one set of copy sources (e.g., a set of source logical units) to a set of copy destinations (e.g., a set of destination logical units). The transfers requested by an XCOPY command are managed by a copy manager (see 4.28.1).

Table 44 — XCOPY command

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (83h)							
1	Reserved			SERVICE ACTION (01h)				
2	Reserved							
…								
9								
10	(MSB)	PARAMETER LIST LENGTH						
…								
13								(LSB)
14	Reserved							
15	CONTROL							

The following parameters apply:

- PARAMETER LIST LENGTH

  If the parameter list length causes truncation of the parameter list, then no data is transferred and the XCOPY command is terminated with ILLEGAL REQUEST, PARAMETER LIST LENGTH ERROR (5/1A00).

XCOPY Parameters (see 6.9 on page 495) describes the parameter data.

5.2.6 ERASE - 19h

The Erase command is defined in SSC (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause specifies the specific implementation.

Table 45 — Erase CDB

Byte	**Bit**							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Operation code (19h)							
1	Obsolete (LUN)			Reserved			Immed	Long
2	Reserved							
4								
5	Control Byte (see 5.1.3)							

The following parameters apply:

- Immed (Immediate)

Value	**Description**
0b	return status when the erase operation has completed.
1b	return status when the CDB has been validated and the buffer flushed.

- Long

Value	**Description**
0b	All remaining medium in the current partition is logically erased beginning at the current logical position.
1b	All remaining medium in the current partition is physically erased and overwritten beginning at the current logical position. This operation may take an extended amount of time to complete.

The Erase command performs either a logical or a physical medium erase from the current position to the end of the current or only partition.

5.2.7 FORMAT MEDIUM - 04h (beginning with E07)

The FORMAT MEDIUM command (see table 46) is used to prepare the medium for use by the logical unit. On some media types this includes a characterization of the medium. If buffered logical objects are stored by the device server when processing of a FORMAT MEDIUM command begins, the command shall be rejected with CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and the additional sense code shall be set to POSITION PAST BEGINNING OF MEDIUM.

See Partitioning and reformatting (see 4.19.6 on page 75) for restrictions related to scaled volumes.

The FORMAT MEDIUM command is subject to:

1) the WORM Write/Append General Behavior (see 4.15.1 on page 53) rules at BOP;
2) the Partitioning and WORM volumes (see 4.19.5 on page 75) rules; and
3) the Append-only mode (beginning with E07) (see 4.18.3 on page 60) rules.

If a WORM cartridge is loaded when a FORMAT MEDIUM command is received and overwriting from BOP is not allowed, then the command will be rejected with CHECK CONDITION status. The sense key shall be set to DATA PROTECT and the additional sense code shall be set to WRITE APPEND POSITION ERROR (WORM).

Table 46 — FORMAT MEDIUM command

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (04h)							
1	Reserved						VERIFY	IMMED
2	Reserved				FORMAT			
3	(MSB)	TRANSFER LENGTH						
4								(LSB)
5	CONTROL							

The FORMAT MEDIUM command shall be accepted only when the medium is at beginning-of-partition 0 (BOP 0). If the medium is logically at any other position, the command shall be rejected with CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and the additional sense code shall be set to POSITION PAST BEGINNING OF MEDIUM.

At the successful completion of a FORMAT MEDIUM command, the medium shall be positioned at BOP 0.

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

a) in response to all commands except REQUEST SENSE and INQUIRY, the device server shall return CHECK CONDITION with a sense key of NOT READY and the additional sense code set to LOGICAL UNIT NOT READY, FORMAT IN PROGRESS unless a reservation conflict exists. In that case RESERVATION CONFLICT status shall be returned; or
b) in response to the REQUEST SENSE command, assuming no error has occurred, the device server shall return a sense key of NOT READY and the additional sense code shall be set to LOGICAL UNIT NOT READY, FORMAT IN PROGRESS, with the sense key specific bytes set for process indication (see SPC-4).

The following parameters apply:

- IMMED (immediate):

  An IMMED bit of zero specifies the device server shall not return status until the FORMAT MEDIUM command has completed. An IMMED bit of one specifies the device server shall return status as soon as the valid medium location has been verified and the command descriptor block of the FORMAT MEDIUM command has been validated. If CHECK CONDITION status is returned for a FORMAT MEDIUM command with an IMMED bit of one, the format operation shall not be performed.

- VERIFY: (0b)

The drive does not perform any verification of the format.

- FORMAT: Dictates what actions to be taken and whether or not to use the settings in mode page 11h.

<table>
<tr><th>Value</th><th>Description</th></tr>
<tr><td>0h</td><td>Default format:
(Re)format the volume to a single unscaled partition using the density specified in the PENDING WRITE DENSITY AT BOP 0 field of mode page 25h (see 6.6.21) and all data on the volume is lost.</td></tr>
<tr><td>1h</td><td>Partition volume:
<table>
<tr><th>Volume State</th><th>Behavior</th></tr>
<tr><td>Scaled</td><td>Return a CHECK CONDITION status with the sense key set to ILLEGAL REQUEST and the additional sense code set to PARAMETER VALUE INVALID</td></tr>
<tr><td>Unscaled</td><td>(Re)format the volume as specified in the Medium Partition mode page (see 6.6.19) using the density specified in the PENDING WRITE DENSITY AT BOP 0 field of mode page 25h. All data on the volume is lost.
If the PENDING WRITE DENSITY AT BOP 0 field of mode page 25h specifies a density that does not support partitioning, a CHECK CONDITION status with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB is returned.</td></tr>
</table>
See MP 23h: Medium Sense (see 6.6.19 on page 414) for a description of scaled volumes.</td></tr>
<tr><td>2h</td><td>Partition volume from default format:
Unscale the volume and (re)format the volume as specified in the Medium Partition mode page using the density specified in the PENDING WRITE DENSITY AT BOP 0 field of mode page 25h and all data on the volume is lost. If the PENDING WRITE DENSITY AT BOP 0 field of mode page 25h specifies a density that does not support partitioning, a CHECK CONDITION status with the sense key set to ILLEGAL REQUEST and the additional sense code set to PARAMETER VALUE INVALID is returned.
See MP 23h: Medium Sense (see 6.6.19 on page 414) for a description of scaled volumes.</td></tr>
</table>

When the FORMAT field contains 1h or 2h, some errors related to mode page field contents may not be detected until the FORMAT MEDIUM command is processed. Therefore, some error conditions described in MP 11h: Medium Partition Page (see 6.6.13 on page 399) may be returned in response to a FORMAT MEDIUM command with 1h or 2h in the FORMAT field.

For an example but not an inclusive list, a CHECK CONDITION status with the sense key set to ILLEGAL REQUEST and the additional sense code set to PARAMETER VALUE INVALID is returned if:

  a) THE PENDING WRITE DENSITY AT BOP 0 field of MP 25h: Read/Write Control (see 6.6.21 on page 427) has been modified since MP 11h: Medium Partition Page (see 6.6.13 on page 399) has been updated;

A) the FDP, SDP, and IDP bits of MP 11h: Medium Partition Page (see 6.6.13 on page 399) are all set to zero; and

B) other unspecified conditions.

Additional information on how the FORMAT MEDIUM command interacts with partitioning, capacity scaling, and reformatting are specified in Partitioning and reformatting (see 4.19.6 on page 75).

- TRANSFER LENGTH: (0000h)

5.2.8 GENERATE RECOMMENDED ACCESS ORDER (GRAO) - A4h[1Dh] (beginning with E07)

The GENERATE RECOMMENDED ACCESS ORDER (GRAO) command is used by an application client to request the device server generate a recommended access order for the User Data Segments that are sent in this command.

> NOTE 19 - After a GRAO command completes, use the RECEIVE RECOMMENDED ACCESS ORDER (RRAO) command (see 5.2.29) to receive the results in the form of an RAO list. When a GRAO command is received and the device server begins validation of the fields in the CDB the previous RAO list is cleared. If the GRAO command is rejected for ILLEGAL REQUEST or any error other than RESERVATION CONFLICT, Deferred Error, or Unit Attention, then the previous results are invalidated.

Table 47 — GENERATE RECOMMENDED ACCESS ORDER CDB

Byte	**Bit**							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	OPERATION CODE (A4h)							
1	Reserved			SERVICE ACTION (1Dh)				
2	Reserved					PROCESS		
3	Reserved					UDS_TYPE		
4	Reserved							
5								
6	MSB	PARAMETER LIST LENGTH						
…								
9								LSB
10	Reserved							
11	Control Byte							

The following parameters apply:

- PROCESS - Requested process to generate the contents of the RAO list (see 4.27.2).
- UDS_TYPE - Type of User Data Segment descriptor to generate (see 4.27.3)

Value	**Description**
000b	User Data Segment without geometry
001b	User Data Segment with geometry
010b-111b	Reserved

- PARAMETER LIST LENGTH: This field specifies the length in bytes of the parameter list that is transferred from the initiator to the target. A parameter list length value of zero indicates that no data is transferred and that the RAO list is to be cleared. This condition is not considered an error. If the specified parameter list length results in truncation of one or more User Data Segments, the drive returns CHECK CONDITION status with associated sense data of 5/2400 (ILLEGAL REQUEST, INVALID FIELD IN CDB). If the parameter length is eight (i.e., no descriptors are transferred), then the RAO list is cleared. This is not considered an error. If the parameter length is less than eight, then the drive returns CHECK CONDITION status with associated sense data of 5/2400 (ILLEGAL REQUEST, INVALID FIELD IN CDB)

5.2.8.1 GRAO Parameter Data

The parameter data sent with the GENERATE RECOMMENDED ACCESS ORDER command is defined in table 48.

Table 48 — GRAO Parameter List

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Reserved							
…								
3								
4	(MSB)	ADDITIONAL DATA (n-7)						
…								
7								(LSB)
User Data Segment descriptors								
	User Data Segment descriptor (first)							
x								
⋮								
y	User Data Segment descriptor (last)							
n								

The following parameters apply:

- ADDITIONAL DATA - This field specifies the amount of data to follow.
- User Data Segment descriptors - A list of User Data Segments for the drive to recommend an access order. The User Data Segment descriptor is defined in 5.2.8.1.1. A CHECK CONDITION WITH THE SENSE KEY SET TO ILLEGAL REQUEST AND THE ADDITIONAL SENSE CODE SET TO INVALID FIELD IN PARAMETER LIST IS RETURNED AND ANY PREVIOUS RECOMMENDED ACCESS ORDER LIST IS INVALIDATED IF:
  - A) the number of User Data Segment descriptors sent is larger than the maximum number supported;
  - B) a User Data Segment specified by one of the descriptors does not exist (i.e., the BEGINNING LOGICAL OBJECT LOCATION - PARTITION NUMBER combination does not exist or the ENDING LOGICAL OBJECT LOCATION - PARTITION NUMBER DOES NOT EXIST); OR
  - C) A USER DATA SEGMENT specified by one of the descriptors IS MAL-FORMED (E.G., THE ENDING LOGICAL OBJECT LOCATION IS LOCATED AT A SMALLER LOCATION THAN THE BEGINNING LOGICAL OBJECT LOCATION).

5.2.8.1.1 GRAO User Data Segment descriptor

The User Data Segment descriptor to be sent is defined in table 49.

Table 49 — GRAO - User Data Segment descriptor

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	(MSB)	DESCRIPTOR LENGTH (1Eh)						
1								(LSB)
2	Reserved							
3	Reserved							
4	Reserved							
5	UDS NAME							
…								
14								
15	PARTITION NUMBER							
16	(MSB)	BEGINNING LOGICAL OBJECT IDENTIFIER						
…								
23								(LSB)
24	(MSB)	ENDING LOGICAL OBJECT IDENTIFIER						
…								
31								(LSB)

The following parameters apply:

Byte	Description
0-1	DESCRIPTOR LENGTH - length of data to follow.
2-4	Reserved
5-14	UDS NAME - Identifier given to this User Data Segment by the application for the applications use. Not used by the drive.
15	PARTITION NUMBER - Number of the partition in which this User Data Segment is located.
16-23	BEGINNING LOGICAL OBJECT IDENTIFIER - LOGICAL OBJECT IDENTIFIER OF THE BEGINNING LOGICAL OBJECT OF THE USER DATA SEGMENT.
24-31	ENDING LOGICAL OBJECT IDENTIFIER - LOGICAL OBJECT IDENTIFIER OF THE ENDING LOGICAL OBJECT OF THE USER DATA SEGMENT.

5.2.9 INQUIRY - 12h

The Inquiry command is defined in SPC (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause specifies the specific implementation.

Table 50 — Inquiry CDB

Byte	**Bit**							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Operation code (12h)							
1	Obsolete (LUN)			Reserved			Obsolete	EVPD
2	Page Code							
3	(MSB)	ALLOCATION LENGTH						
4								(LSB)
5	Control Byte (see 5.1.3)							

The following parameters apply:

- EVPD

Value	**Description**
0b	Inquiry Standard Data is returned: “Inquiry Standard Data: Valid LUN (Logical Unit Number)” on page 118 “Inquiry Standard Data: Invalid LUN” on page 120
1b	Vital Product Data is returned

- PAGE CODE

  If EVPD bit is set to one, then the Page Code field refers to the Vital Product Data page being requested. Supported pages are shown in IP 00h: Supported Inquiry Pages (see 6.3.1 on page 261).

- ALLOCATION LENGTH

NOTE 20 - The allocation length field in TS1130 and earlier was one byte long (i.e., byte 4)

5.2.9.1 Inquiry Standard Data

The standard INQUIRY data format is shown in table 51. Inquiry Standard Data is returned with table 51 populated in one of two different ways. If the command is received on a valid LUN then the data returned is shown in “Inquiry Standard Data: Valid LUN (Logical Unit Number)” on page 118. If the command is received on an invalid LUN then the data returned is shown in “Inquiry Standard Data: Invalid LUN” on page 120.

Table 51 — Standard INQUIRY data format

Bit Byte	7	6	5	4	3	2	1	0
0	PERIPHERAL QUALIFIER			PERIPHERAL DEVICE TYPE				
1	RMB	Reserved						
2	VERSION							
3	Obsolete	Obsolete	NORMACA	HISUP	RESPONSE DATA FORMAT			
4	ADDITIONAL LENGTH (n-4)							
5	SCCS	ACC	TPGS		3PC	Reserved		PROTECT
6	Obsolete	ENCSERV	VS	MULTIP	Obsolete	Obsolete	Obsolete	ADDR16
7	Obsolete	Obsolete	WBUS16	SYNC	Obsolete	Obsolete	CMDQUE	VS
8	(MSB)	T10 VENDOR IDENTIFICATION						
15								(LSB)
16	(MSB)	PRODUCT IDENTIFICATION						
31								(LSB)
32	(MSB)	PRODUCT REVISION LEVEL						
35								(LSB)
36		IBM PLANT OF MANUFACTURE CODE						
37								
38		SERIAL NUMBER						
49								
50		PORT NUMBER						
51								
52	Equipment Flags							
	ENCR_C	ENCR_E	Reserved	A-SHARE	IMCI	LIB ATTACH	ACF	MD ATTACH
53	SCSI CUSTOMIZATION							
54	LIBRARY TYPE							
55	MESSAGE DISPLAY TYPE							

5.2.9.1.1 Inquiry Standard Data: Valid LUN (Logical Unit Number)

The following parameters apply to this request:

- EVPD (Enable Vital Product Data): 0b
- Page Code: 00h
- Allocation Length: 38h (56) bytes available

For a logical unit number (LUN) that is associated with an installed device (see 4.2), the following standard inquiry data is returned (character fields are in ASCII):

Byte **Description**

0

Bit	Description
7-5	Peripheral Qualifier: 000b
4-0	Peripheral Device Type: 01h (Sequential Access Device)

1

Bit	Description
7	RMB (Removable Medium Bit): 1b
6-0	Reserved

2 VERSION: 06H

3

Bit	Description
7	Obsolete: 0b
6	Obsolete: 0b
5	NormACA (Normal ACA Supported): 0b
4	HiSupport (Hierarchical Support): 0b
3-0	Response Data Format: 0010b

4 Additional Length (n-4): 33h

5

Bit	Description
7	SCCS (An SCC Supported): 0b
6	ACC (Access Controls Coordinator): 0b
5-4	TPGS (Target Port Group Support): 00b
3	3PC (Third-Party Copy): This bit indicates support for third-party copy as described in Extended Copy (i.e., Tape to tape copy) (E08 and later devices) (see 4.28 on page 91).

Value	Description
0b	This device does not support third-party copy.
1b	This device supports third-party copy.

Bit	Description
2-1	Reserved
0	PROTECT (Logical Block Protection - see 4.7): 1b

6 This byte supports SCSI-3 changes. In SCSI-2, this byte was Reserved.

Bit	Description
7	Obsolete: 0b
6	EncServ (Enclosure Service): 0b
5	VS: 0b
4	MULTIP (Multi-Port): 1b
3	Obsolete: 0b
2	Obsolete: 0b
1	Obsolete: 0b
0	Addr16: 0b

7

Bit	Description
7	Obsolete: 0b
6	Obsolete: 0b
5	WBUS16 (Wide Bus 16): 0b
4	SYNC (Synchronous Transfer): 0b
3	Obsolete: 0b
2	Obsolete: 0b
1	CMDQUE (Command Queuing): 0b
0	VS: 0b

8-15 T10 VENDOR IDENTIFICATION (Manufacturer): ‘IBM ‘ (in ASCII)

16-31 PRODUCT IDENTIFICATION: ‘03592xxx ‘ (in ASCII)

This is 03592 followed by xxx; where xxx is ‘J1A’, 'E05', ‘E06’, ‘E07’, ‘E08’, ‘55F’, or ‘60F’.

32-35 Product Revision Level (Drive Microcode Revision Level in ASCII)

36-37 IBM Plant of Manufacture Code

38-49 Serial Number of device, right justified with leading zeroes, in ASCII

50-51 PORT NUMBER: For Port 0: ‘ 0’ (in ASCII); For Port 1: ‘ 1’ (in ASCII)

52 Equipment Flags

Bit **Description**

7 ENCR_C: Device Supports Encryption - Capable

Value	**Description**
0b	Device does not support encryption (does not have encryption hardware)
1b	Device supports encryption (encryption interface(s) are not necessarily enabled)

6 ENCR_E: Device Supports Encryption - Enabled

Value	**Description**
0b	Device does not support encryption
1b	Device supports encryption (encryption interface(s) are enabled)
Value	**Description**

5 Vendor-Reserved

4 A-SHARE: Auto-Share Feature Installed (see 4.4)

Value	**Description**
0b	Indicates Auto-Share feature is not installed
1b	Indicates Auto-Share feature is installed

3 IMCI: Independent Medium Changer Installed (see byte 0, bits 4-0)

Value	**Description**
0b	a Medium Changer is not addressable at LUN 1
1b	a Medium Changer is addressable at LUN 1

2 LIB ATTACH: Library Attached attached

Value	**Description**
0b	the device is not attached to a library facility (3494)
1b	the device is attached to a library facility (3494)

1 ACF Attached

Value	**Description**
0b	the device does not support an ACF

0 MD ATTACH: Message Display Attached: 1b

Value	**Description**
0b	a message display is not attached
1b	a message display is attached

53 SCSI Customization: 00h

54 Library Type

55 Message Display Type: 81h

5.2.9.1.2 Inquiry Standard Data: Invalid LUN

The following parameters apply to this request:

- EVPD (Enable Vital Product Data): 0b
- Page Code: 00h
- Allocation Length: 24h (36) bytes available

For a LUN that is not associated with an installed device (see 4.2), the following standard inquiry data is returned (character fields are in ASCII):

Byte **Description**

0

Bit	**Description**
7-5	Peripheral Qualifier: 011b
4-0	Peripheral Device Type: 1Fh (no device type)

1

Bit	**Description**
7	RMB (Removable Medium Bit): 0b
6-0	Reserved

2

Bit	Description
7-6	ISO/IEC Version: 00b
5-3	ECMA Version: 000b
2-0	ANSI Approved Version: 011b

3

Bit	Description
7	AERC (Asynchronous Event Reporting Capability): 0b
6	Obsolete: 0b
5	NormACA (Normal ACA Supported): 0b
4	HiSupport (Hierarchical Support): 0b
3-0	Response Data Format: 0010b

4 Additional Length (n-4): 1Fh (31)

5

Bit	Description
7	SCCS (An SCC Supported): 0b
6-0	Reserved

6 This byte supports SCSI-3 changes. In SCSI-2, this byte was Reserved.

Bit	Description
7	BQue (Basic Queueing): 0b
6	EncServ (Enclosure Service): 0b
5	BarC: 0b
4	MultiP (Multi-Port): 1b
3	Mchngr (Medium Changer): 0b
2	AckReqQ: 0b
1	Addr32: 0b
0	Addr16: 0b

7

Bit	Description
7	RelAdr (Relative Addressing): 0b
6	WBus32 (Wide Bus 32): 0b
5	WBus16 (Wide Bus 16): 0b
4	Sync (Synchronous Transfer): 0b
3	Linked: 0b
2	TranDis (Transfer Disable): 0b
1	CmdQue (Command Queuing): 0b
0	SftRe (Soft Reset): 0b

8-15 Manufacturer: ‘IBM   ‘ (in ASCII)

16-31 Device Type and Model Number: (all ASCII blanks)

32-35 Product Revision Level: (all ASCII blanks)

5.2.10 LOAD UNLOAD - 1Bh

The Load Unload command is defined in SSC (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause specifies the specific implementation. In E07 and later products if there is no cartridge present in the drive, then the command is presented with CHECK CONDITION status and associated sense data of 2/3A00 (NOT READY, MEDIUM NOT PRESENT)

Table 52 — Load Unload CDB

Byte	**Bit**							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Operation code (1Bh)							
1	Obsolete (LUN)			Reserved				IMMED
2	Reserved							
3								
4	Reserved				HOLD	EOT	RETEN	LOAD
5	Control Byte (see 5.1.3)							

The following parameters apply:

- IMMED (Immediate)

Value	**Description**
0b	Indicates the drive is to present status when the command is completed.
1b	Indicates the drive is to present status as soon as all buffered commands have completed execution and the CDB of the Load Unload command has been validated. With the exception of Inquiry, Request Sense, and Test Unit Ready, subsequent commands are queued until the load/unload operation is complete. The completion status of the load/unload operation may be polled by sending a Request Sense command until the sense data returned is no longer 2/0407 (Not Ready, Logical Unit Not Ready, Operation in Progress).

- HOLD: Requests MAM to become accessible and the volume not be positioned for access (i.e., volume to be positioned at tray down, medium not threaded). See table 53 for interaction with other bits.
- EOT (End of Tape): 0b
- RETEN (Retention): In products that support Archive mode unthread (E08+) (see 4.9), requests Archive mode unthread. See table 53 for interaction with other bits.
- LOAD: Requests a load or unload be performed. See table 53 for interaction with other bits.

Table 53 — Behavior for the combinations of the RETEN, LOAD, HOLD bits

Volume position	HOLD	LOAD[a]	RETEN	Description
U or M	0b	0b	-	Unload the cartridge from the drive. Upon completion of the command, MAM is not accessible. If the cartridge is already unloaded, GOOD Status is returned.
T	0b	0b	0b	
T	0b	0b	1b	Perform an Archive mode unthread (E08+) (see 4.9) and then unload the cartridge from the drive. Upon completion of the command, MAM is not accessible.
U or M	0b	1b	-	Load the cartridge and become READY.
T	0b	1b	-	The logical position is set to logical block 0 of partition 0 (i.e., BOP 0) (this is not equivalent to a Rewind command as the active partition is set to partition 0).
U or M	1b	-	-	The cartridge is moved to the seated position with MAM accessible and the tape not threaded.
T	1b	-	0b	
T	1b	-	1b	An Archive mode unthread (E08+) (see 4.9) is performed and then the cartridge is moved to the seated position with MAM accessible and the tape not threaded.

Key:
U - Unloaded
M - MAM accessible not threaded
T - Threaded
- = Don’t care

[a] The LOAD UNLOAD command with the LOAD bit set to 0b is sometimes called an unload command. The LOAD UNLOAD command with the LOAD bit set to 1b is sometimes called a reload command.

NOTE 21 - In certain automation environments (such as 3494) this command cannot be processed successfully if the cartridge is in the unloaded position. In this case, the command is presented with CHECK CONDITION status and associated sense data of 2/0403 (NOT READY, MANUAL INTERVENTION REQUIRED).

5.2.11 LOCATE (10/16) - 2Bh/92h

The Locate commands are defined in SSC (see http://www.t10.org/drafts.htm#SSC_Family). This clause specifies the specific implementation. The LOCATE(16) command (see table 55) adds the ability to position the medium to a logical object or to a logical file, whereas the LOCATE(10) command (see table 54) only allows positioning to a logical object. It is recommended that the LOCATE(16) command be used for all new implementations.

Table 54 — LOCATE(10) command

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (2Bh)							
1	Reserved					BT	CP	IMMED
2	Reserved							
3	(MSB)	LOGICAL IDENTIFIER						
4								
5								
6								(LSB)
7	Reserved							
8	PARTITION							
9	CONTROL (see 5.1.3)							

Table 55 — LOCATE(16) command

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (92h)							
1	Reserved			DEST_TYPE		Rsvd	CP	IMMED
2	Reserved							BAM (0b)
3	PARTITION							
4	(MSB)	LOGICAL IDENTIFIER						
5								
6								
7								
8								
9								
10								
11								(LSB)
12	Reserved							
13	Reserved							
14	Reserved							
15	CONTROL (see 5.1.3)							

The following parameters apply:

- BT (Block address Type): 0b - LOCATE(10) only

Value	**Description**
0b	use standard locate positioning

- DEST_TYPE (Destination Type) - LOCATE(16) only

  This is used in conjunction with the LOGICAL IDENTIFIER field to locate to the appropriate position of the medium.

Value	**Description**
00b	The LOGICAL IDENTIFIER field specifies to which logical object identifier on whose BOP side the medium be located
01b	The LOGICAL IDENTIFIER field specifies to which logical file identifier on whose BOP side the medium be located
011b	The LOGICAL IDENTIFIER field is ignored and the logical position upon completion is EOD (end-of-data) of the partition specified in the PARTITION field if the CP bit is set to 1b or EOD of the current partition if the CP bit is set to 0b.
others	Reserved

- CP (Change Partition):

Value	**Description**
0b	no partition change is to be made; locate to the specified block address within the current partition. The PARTITION field is to be ignored.
1b	change to the partition specified by the PARTITION field prior to locating to the specified Block Address within the partition.

- IMMED (Immediate):

Value	**Description**
0b	present status when command is completed.
1b	present status when all buffered commands have completed execution and the CDB of the Locate command is validated.

- BAM (Block Addess Mode type): 0b (process this command as an implicit address command)
- LOGICAL IDENTIFIER: The destination of the locate operation. In the LOCATE(10) command this field specifies the logical object identifier. In the LOCATE(16) command, the DEST_TYPE field specifies if the locate is of logical objects or logical files.
- PARTITION:

  The partition field specifies the partition to select, when the CP field is 1b.

If the drive encounters End-of-Data (EOD) while processing this command and the DEST_TYPE is not set to 011b (i.e., EOD), then the command is terminated at the EOD position and CHECK CONDITION status is returned with associated sense data of 8/0005 (Blank Check, End-of-Data Detected).

5.2.12 LOG SELECT - 4Ch

The Log Select command is defined in SPC (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause specifies the specific implementation.

Table 56 — Log Select CDB

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	OPERATION CODE (4Ch)							
1	Reserved						PCR	SP
2	PC		PAGE CODE					
3	SUBPAGE CODE							
4	Reserved							
6								
7	MSB	PARAMETER LIST LENGTH						
8								LSB
9	Control Byte (see 5.1.3)							

The following parameters apply:

- PCR (Parameter Code Reset):

Value	Description
0b	Indicates that the log parameters will not be reset.
1b	If the parameter list length is zero, all cumulative and threshold log counter values will be reset to their default values as specified in that pages reset behavior section. If the parameter list length is not zero, the command is terminated with CHECK CONDITION status and associated sense data of 5/2400 (Illegal Request, Invalid Field in CDB).

- SP (Save Parameters): 0b (Saving of the Log Select parameters is not supported)
- PC (Page Control):

Value	Description
00b	(Threshold Values): Supported for all log pages with log counters (LP field set to 0 in the Log Parameter Control Byte) except for LP 3Dh: Subsystem Statistics (see 6.4.23).

NOTE 22 - This device treats each threshold value as a maximum value for the log counter field. Generally, when a threshold/maximum is reached, all log counters in that specific log page are locked (no longer updated) until a subsequent reset via a Log Select command.

NOTE 23 - Only the overflowed log counter is locked forLP 38h: Blocks/Bytes Transferred (see 6.4.18) - all other log counters continue incrementing for this log page.

NOTE 24 - Most log counters forLP 3Dh: Subsystem Statistics (see 6.4.23) lock at maximum values and cannot be reset.

NOTE 25 - If the RLEC bit is set to 1b in “MP 0Ah: Control Mode” on page 387 and a log counter reaches its threshold/maximum, the drive reports a deferred CHECK CONDITION status with associated sense data of 1/5B02 (Recovered Error, Log Counter at Maximum) on the next command eligible for a deferred check condition (see table 37 on page 100). The drive does not report error sense associated with the threshold condition being met.

01b	(Cumulative Values): Supported for all log pages with log counters (LP field set to 0 in the Log Parameter Control Byte) except for LP 3Dh: Subsystem Statistics (see 6.4.23).
10b	(Default Threshold Values): The default threshold value for all log counter fields is the maximum value (every byte is FF).
11b	(Default Cumulative Values): The default cumulative value for all log counter fields is zero (every byte is 00h). If the PCR field is set to 1b, the PC field is ignored.

- PAGE CODE
- SUBPAGE CODE: Supported PAGE CODEs and SUBPAGE CODEs are described in the log page sections (see 6.4). Log Parameters (see 6.4 on page 287) contains a list of supported log pages and their reset conditions.
- PARAMETER LIST LENGTH:

  This field specifies the length in bytes of the parameter list that is to be transferred to the drive. A parameter list length of zero indicates that no pages are to be transferred. If the parameter list length is zero and the PC field is set to 00b (Current Threshold Values), the current threshold parameters are set to the default threshold values. If the parameter list length is zero and the PC field is set to 01b (Current Cumulative Values), the current cumulative parameters are set to the default cumulative values (zero).

  NOTE 26 - If the PCR field is set to 1b, this field must be set to zero.

  If the parameter list length results in the truncation of any log parameter, the command is terminated with CHECK CONDITION status and associated sense data of 5/2400 (Illegal Request, Invalid Field in CDB).

Only one log page is accepted for each Log Select command. For each log page, any combination of the supported log parameters may be sent. If multiple log parameters are sent, they must be sent in ascending order by parameter code value. Only the Parameter Value field may be changed from the log parameters that are returned from Log Sense (see “Log Parameter Format” on page 289). Changes to the Log Parameter Control Byte are not supported.

NOTE 27 - Applications should issue a Log Sense command prior to issuing a Log Select command to determine supported log parameter fields.

If a parameter list is received with an unsupported log page, a log parameter code out of order, or a change to a log parameter field other than the Parameter Value, the command is terminated with CHECK CONDITION status and associated sense data of 5/2600 (Illegal Request, Invalid Field in Parameter List).

5.2.13 LOG SENSE - 4Dh

The Log Sense command is defined in SPC (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause specifies the specific implementation.

Table 57 — Log Sense CDB

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	OPERATION CODE (4Dh)							
1	Obsolete (LUN)			Reserved			PPC	SP
2	PC		PAGE CODE					
3	SUBPAGE CODE							
4	Reserved							
5	MSB	PARAMETER POINTER						
6								LSB
7	MSB	ALLOCATION LENGTH						
8								LSB
9	Control Byte (see 5.1.3)							

The following parameters apply:

- PPC (Parameter Pointer Control): 0b
- SP (Save Parameters): 0b
- PC (Page Control):

Value	**Description**
00b	(Threshold Values): Supported for all log pages with log counters (LP field set to 0b in the Log Parameter Control Byte). For additional information, see Threshold Values page 126.
01b	(Cumulative Values): Supported for all log pages.
10b	(Default Threshold Values): Supported for all log pages with log counters. The default threshold value for all 2-byte log counter fields is FFFFh. The default threshold value for all 4-byte log counter fields is FFFF FFFFh.
11b	(Default Cumulative Values): The default cumulative value for all 2-byte log counter fields is 0000h. The default cumulative value for all 4-byte log counter fields is 0000 0000h.

- PAGE CODE
- SUBPAGE CODE: Supported PAGE CODEs and SUBPAGE CODEs are described in the log page sections (see 6.4). Log Parameters (see 6.4 on page 287) contains a list of supported log pages and their reset conditions.

WARNING WARNING

The parameters in log pages should be dynamically parsed as some parameters may not be present, new parameters may be inserted, and parameter sizes may change. The listed size is for the most recent generation at the time of publication. Some sizes have changed from previous generations. The relative location of parameters have changed and are anticipated to continue to change.

WARNING WARNING

5.2.14 MODE SELECT (6/10) - 15h/55h

The Mode Select commands are defined in SPC (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause specifies the specific implementation.

> NOTE 28 - In the future, the length of the mode parameter list for Mode Sense Page Code 3Fh (All Pages) may exceed 255 bytes. At that time, use of the Mode Select (10) and Mode Sense (10) commands will be required in order to transfer all mode pages with one command. Some mode pages today exceed 255 bytes. For this reason, use of the Mode Select (6) and Mode Sense (6) commands is not recommended.

Table 58 — Mode Select (6) CDB

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Operation code (15h)							
1	Obsolete (LUN)			PF	Reserved			SP
2	Reserved							
3								
4	Parameter List Length							
5	Control Byte (see 5.1.3)							

Table 59 — Mode Select (10) CDB

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Operation code (55h)							
1	Obsolete (LUN)			PF	Reserved			SP
2	Reserved							
6								
7	MSB	PARAMETER LIST LENGTH						
8								LSB
9	Control Byte (see 5.1.3)							

The following standards-based mode pages are supported for the Mode Select command:

“MP 01h: Read-Write Error Recovery” on page 383
“MP 02h: Disconnect-Reconnect” on page 385
“MP 0Ah: Control Mode” on page 387
“MP 0Fh: Data Compression” on page 392
“MP 10h: Device Configuration” on page 394
“MP 18h: Fibre Channel Logical Unit Control” on page 405
“MP 19h: Fibre Channel Port Control” on page 406
“MP 1Ch: Informational Exceptions Control” on page 408

The following vendor-specific mode pages are also supported for the Mode Select command:

“MP 21h: TOD Control” on page 411
“MP 22h: Language” on page 413
“MP 23h: Medium Sense” on page 414
“MP 24h: Initiator-Specific Extensions” on page 424
“MP 25h: Read/Write Control” on page 427
“MP 37h: String Search (not J1A)” on page 444
“MP 3Eh: Engineering Support” on page 455

NOTE 29 - Mode page 3Eh is for engineering use only.

The following parameters apply:

- PF (Page Format): 1b

  The PF (Page Format) bit is explicitly not checked.

- SP (Save Pages):

  The SP bit indicates whether to save the mode parameters.

Value	**Description**
0b	Current values of the mode page are updated. No mode parameters are saved to non-volatile memory.
1b	The behavior depends on the value of the PS bit returned in the mode page in response to a MODE SENSE command and if the parameters are supported as Current values and Saveable values or as Saveable values only.

PS	**Description**
0b	Current values of the mode parameters are updated if Current values are supported. No mode parameters are saved to non-volatile memory. If Current values are not supported, the command is terminated with a CHECK CONDITION status.
1b	Current values of the mode parameters are updated and saveable mode parameters are saved in non-volatile memory.

- PARAMETER LIST LENGTH:

  This field specifies the length in bytes of the mode parameter list that is transferred from the initiator to the target. A parameter list length of zero indicates that no data is transferred. This condition is not considered as an error.

  The target terminates the command with CHECK CONDITION status with associated sense data of 5/1A00 (Illegal Request, Parameter List Length Error) if the parameter list length results in the truncation of the mode parameter header, the mode parameter block descriptor, or any mode page.

NOTE 30 - Issuing a Mode Sense for current values before a Mode Select is generally recommended to avoid accidentally attempting to set fields that cannot be changed by the initiator.

Changing some MP 23h: Medium Sense or MP 10h: Device Configuration parameters causes an implicit write to tape. At Beginning Of Tape, any tape written in one format may be changed to another format when an Erase, Write, or Write Filemarks command is issued. If an attempt to change any such parameters when an incompatible format tape is loaded (even when at Beginning Of Tape), CHECK CONDITION status is returned with associated sense data of 5/3005 (Illegal Request, Cannot Write Medium - Incompatible Format).

Listed below are the Mode Parameters that cause implicit writes:

Mode Page 10, Byte 15, Bit 1 PERSWP
Mode Page 10, Byte 15, Bit 0 PRMWP
Mode Page 23, Byte 10, Bit 4 Persistent Write Protect
Mode Page 23, Byte 10, Bit 3 Reset Persistent Write Protect
Mode Page 23, Byte 10, Bit 0 Permanent Write Protect
Mode Page 23, Byte 11, Bit 0 CapScalV

NOTE 31 - For Reserved and Vendor-Reserved fields, appropriate values to issue on a Mode Select may be non-zero. Mode Selects to pages with these fields should use a value obtained by issuing a Mode Sense just prior to the Mode Select.

Clause 6.6 on page 373 has a listing of all mode parameters.

5.2.15 MODE SENSE (6/10) - 1Ah/5Ah

The Mode Sense commands are defined in SPC (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause specifies the specific implementation.

NOTE 32 - In the future, the length of the mode parameter list for Mode Sense Page Code 3Fh (All Pages) may exceed 255 bytes. At that time, use of the Mode Select (10) and Mode Sense (10) commands will be required in order to transfer all mode pages with one command. For this reason, use of the Mode Select (6) and Mode Sense (6) commands is not recommended.

NOTE 33 - See "MODE SENSE (6/10) - 1Ah/5Ah" on page 131 for a description of the supported mode pages, recommended command usage, and the behavior of parameters for this command.

Table 60 — Mode Sense (6) CDB

Byte	**Bit**							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Operation code (1A)							
1	Obsolete (LUN)			Reserved	DBD	Reserved		
2	PC		Page Code					
3	Subpage Code							
4	Allocation Length							
5	Control Byte (see 5.1.3)							

Table 61 — Mode Sense (10) CDB

Byte	**Bit**							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Operation code (5A)							
1	Obsolete (LUN)			Reserved	DBD	Reserved		
2	PC		Page Code					
3	Subpage Code							
4	Reserved							
6								
7	MSB	Allocation Length						
8								LSB
9	Control Byte (see 5.1.3)							

The following parameters apply:

- DBD (Disable Block Descriptors): 0b or 1b. (See Block Descriptor Length, below.)
- PC (Page Control): 00b, 01b; or 10b supported
- Page Code

The following standards-based mode pages are supported

“MP 01h: Read-Write Error Recovery” on page 383
“MP 02h: Disconnect-Reconnect” on page 385
“MP 0Ah: Control Mode” on page 387
“MP 0Fh: Data Compression” on page 392
“MP 10h: Device Configuration” on page 394
“MP 18h: Fibre Channel Logical Unit Control” on page 405
“MP 19h: Fibre Channel Port Control” on page 406
“MP 1Ch: Informational Exceptions Control” on page 408

The following vendor-specific mode pages are also supported

“MP 21h: TOD Control” on page 411
“MP 22h: Language” on page 413
“MP 23h: Medium Sense” on page 414
“MP 24h: Initiator-Specific Extensions” on page 424
“MP 25h: Read/Write Control” on page 427
“MP 37h: String Search (not J1A)” on page 444
“MP 3Eh: Engineering Support” on page 455
“Mode Page 3Fh: All Pages”

NOTE 34 - Mode page 37h is not included in page 3Fh.

NOTE 35 - Mode page 3Eh is for engineering use only and is not included in page 3Fh

- Subpage Code
- Allocation Length: The maximum number of bytes to be transferred.

  If the allocation length specified is less than the amount available, then the allocated amount is transferred and no error is reported.

Clause 6.6 on page 373 has a listing of all mode parameters.

5.2.16 PERSISTENT RESERVE IN - 5Eh

The Persistent Reserve In command is defined in SPC (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause specifies the specific implementation.

Table 62 — Persistent Reserve In Command

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Opcode: 5Eh							
1	Reserved			Service Action				
2	Reserved							
5								
6	MSB	Allocation Length						
8								LSB
9	Control Byte (see 5.1.3)							

The following parameters apply:

- Service Action:

Value	Description
00h	Reads all registered Reservation Keys
01h	Reads all current persistent reservations
02h	Returns capability information, (Not supported on earlier code levels).
03h	Reads complete information about all registrations and the persistent reservations, if any. (Not supported on earlier code levels).

- Allocation Length: The maximum number of bytes to be transferred.

The Persistent Reserve In parameter data for Read Keys is defined below. (Not supported on earlier code levels):

Byte	Description
0-3	Generation: Counter for Persistent Reserve Out Command requests
4-7	Additional length: A count of the number of bytes in the Reservation key list
8-15	First Reservation Key
16-n	Additional Reservation keys: a maximum of 1 reservation key per initiator is supported

The Persistent Reserve In parameter data for Read Reservations is defined below:

Byte	Description
0-3	Generation: Counter for Persistent Reserve Out Command requests
4-7	Additional length: A count of the number of bytes in the Reservation key list
8-n	Reservation descriptors: (defined below)

The Persistent Reserve In Read Reservations Descriptor is defined below:

Byte	Description
0-7	Reservation Key
8-11	Scope-specific address: 00000000h
12	Reserved

13

Bit	Description
7-4	Scope: persistent reservation applies to the entire logical unit: 0h
4-0	Type:

Value	Descriptor
3h	Exclusive Access
6h	Exclusive Access, Registrants only

14-15 Extent Length: 0000h

The Persistent Reserve In Read Capabilities Descriptor is defined below (Not available on earlier code levels):

Byte	Description
0-1	Length: 0008h
2	

Bit	Description
7-5	Reserved
4	Compatible Reservation Handling (CRH): 1b
3	Specify Initiator Ports Capable (SIP_C): 1b
2	All Target Ports Capable (ATP_C): 1b
1	Reserved 0Persist Through Power Loss Capable (PTPL_C): 0b

3

Bit	Description
7	Type Mask Valid (TMV): 1b
6-1	Reserved
0	Persist Through Power Loss Activated (PTPL_A): 0b

4 Persistent Reservation Type Mask (byte 1)

Bit	Description
7	Write Exclusive - All Registrants (WR_EX_AR): 0b
6	Exclusive Access - Registrants Only (EX_AC_RO): 1b
5	Write Exclusive - Registrants Only (WR_EX_RO): 0b
4	Reserved
3	Exclusive Access (EX_AC): 1b
2	Reserved
1	Write Exclusive (WR_EX): 0b
0	Reserved

5 Persistent Reservation Type Mask (byte 2)

Bit	Description
7-1	Reserved
0	Exclusive Access - Registrants Only (EX_AC_AR): 0b

6-7 Reserved

The Persistent Reserve In Read Full Status Descriptor is defined below

Byte	Description
0-3	PRGeneration
4-7	Additional Length (n-7)
7-p	First full status descriptor
…	
q-n	Last full status descriptor

The Persistent Reserve In Full Status Descriptor format is defined below:

Byte	Description
0-7	Reservation Key
8-11	Reserved
12	Persistent Reservation Type Mask (byte 2)

Bit	Description
7-2	Reserved
1	All Target Ports (All_TG_PT)
0	Reservation Holder (R_HOLDER)

Byte	Description
13	

Bit	Description
7-4	Scope
3-0	Type Byte Description

Byte	Description
14-17	Reserved
18-19	Relative Target Port Identifier
20-23	Additional Descriptor Length (n-23)
24-n	TransportID

5.2.17 PERSISTENT RESERVE OUT - 5Fh

The Persistent Reserve Out command is defined in SPC (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause specifies the specific implementation.

Table 63 — Persistent Reserve Out Command

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Opcode: 5Fh							
1	Reserved			Service Action				
2	Scope				Type			
3	Reserved							
6								
7	MSB	Parameter List Length						
8								LSB
9	Control Byte (see 5.1.3)							

The following parameters apply:

- Service Action:

Value	Description
00h	REGISTER: Register a reservation key with the device server
01h	RESERVE: Create a persistent reservation using a reservation key
02h	RELEASE: Release a persistent reservation
03h	CLEAR: Clear all reservation keys and all persistent reservations
04h	PREEMPT: Preempt persistent reservations and/or removes registrations
05h	PREEMPT AND ABORT: Preempt persistent reservations and/or removes registrations and clear the task set for the preempted initiator
06h	REGISTER AND IGNORE: Register a reservation key without needing to provide current regestration key.
07h	REGISTER AND MOVE: Register And Move the registration to another I_T nexus.
08h-1Fh	Reserved

- Scope: 0h
- Type:

Value	Description
3h	Exclusive Access
6h	Exclusive Access, Registrants only

- Parameter List Length: 0018h for all Service Actions except Register and Move (07h)

  Parameter List Length: variable for Service Action Register and Move (07h)

The Persistent Reserve Out parameter list for all Service Actions except Register and Move (07h) is defined below:

Byte	Description
0-7	Reservation Key
8-15	Service Action Reservation Key
16-19	Scope-specific address: 00000000h
20	

Bit	Description
7-1	Reserved
0	APTPL: Activate Persist Through Power Loss: 0b

21	Reserved
22-23	Obsolete: 0000h

The Persistent Reserve Out parameter list for the Register and Move (07h) Service Action is defined below (Not available on earlier code levels):

Byte	Description
0-7	Reservation Key
8-15	Service Action Reservation Key
16	Reserved
17	
18-19	Relative Target Port Identifier
20-23	Additional Descriptor Length (n-23)
24-n	TransportID

Byte 17:

Bit	Description
7-2	Reserved
1	UNREG: Unregister
0	APTPL: Activate Persist Through Power Loss: 0b

5.2.18 PREVENT ALLOW MEDIUM REMOVAL - 1Eh

The Prevent Allow Medium Removal command is defined in SSC-3 (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause specifies the specific implementation.

The Prevent Allow Medium Removal command is supported only for the Prevent Cartridge Removal option. Table 64 shows the command format.

Table 64 — Prevent Allow Medium Removal Command

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Operation code (1Eh)							
1	Obsolete (LUN)			Reserved				
2	Reserved							
3								
4	Reserved						Prevent	
5	Control Byte (see 5.1.3)							

The following parameters apply:

- Prevent:

Value	**Description**
00b	Allow Cartridge Removal
01b	Prevent Cartridge Removal prevent cartridge
10b	(unsupported)
11b	(unsupported)

The device supports Prevent Cartridge Removal by removing the Unload option from the CE service panel menu. The option is still available via the button on the front of the drive, but pressing this when prevented will not cause the cartridge to be ejected. Cartridge removal is enabled again when the initiator issues the Prevent Allow Medium Removal command with the Prevent field set to 00b (Allow Cartridge Removal). A reset (bus device reset, reset message, or power on reset) also restores the drive to the allow removal state. In the case of the bus device reset or the reset message, the reset restores the allow state only if the reset occurs on the same port that originally set Prevent.

NOTE 36 - The Prevent Allow Medium Removal command has no effect on any library commands that may be received.

5.2.19 READ - 08h

The Read command is defined in SSC (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause specifies the specific implementation.

Table 65 — Read CDB

Byte	**Bit**							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Operation code (08h)							
1	Obsolete (LUN)			Reserved			SILI	Fixed
2	MSB	Transfer Length						
4								LSB
5	Control Byte (see 5.1.3)							

The following parameters apply:

- SILI (Suppress Incorrect Length Indicator) (per SCSI-2 standard) (see 4.10)
- Fixed (see 4.8)
- Transfer Length(see 4.8)

For more information, see “General Read-Type Handling” on page 40

5.2.20 READ 08h (String Search) (not J1A)

See “MP 37h: String Search (not J1A)” on page 444, and “String Search Function (not J1A)” on page 79 for additional information.

Table 66 — Read - 08h (String Search) (not J1A)

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Operation Code: (X'08')							
1	Obsolete (LUN)			Reserved			SILI	Fixed
2	MSB	Transfer Length						
4								LSB
5	Search b'1'	FMKS	Control Byte (see 5.1.3)					

The device logical position may be changed by this command.

If no valid search criteria exists, the command presents CHECK CONDITION status with sense key 5/2C00 (Illegal Request - Command Sequence Error).

If Fixed is set to 1b it only affects the read of records after the first matching record is found. Blocks which are not the same length of the current set Block Length are fully checked for matches, but will not stop the search portion of the Read operation. The data returned includes the matching record and the sequentially following records (not necessarily matching search criteria).

The following parameters differ from the standard Read command:

- Search:

Value	**Description**
0b	this command is a standard Read command
1b	this command is a Read (String Search) command

- FMKS:

Value	**Description**
0b	filemarks stop the search operation with the logical position left after the filemark and are reported per the standard Read command.
1b	filemarks will be ignored (treated as a logical block with no data) - will not match any criteria If a matching record is found, the command will return the record data with GOOD STATUS and the device will be logically positioned after the matching record (another Read (String Search) command may be issued to read the next matching record).

If a matching record is not found before the criteria of the command is fulfilled, the command presents NON-DEFERRED CHECK CONDITION status with sense key 0/1401 (No Sense - Record Not Found).

If a matching record is not found before Search Time elapses, the command presents NON-DEFERRED CHECK CONDITION status with sense key 0/2E00 (No Sense - Insufficient Time for Operation).

Other reportable statuses for the Read command also apply in conditions where the command stops due to other issues before a matching record is found (i.e., filemark is encountered if FMKS is 0b, EOD encountered, etc.)

The method to get results depends on the active operating mode. For snoop operations, individual matches could be reported during the operation via a check condition as described above sections. Summary results (first/last record and match count) may be found by reading the Search String mode page.

Detailed results (Match List) can be read using Read Buffer ID 40h, which will present data in the format detailed in the String Search command returned data description.

5.2.21 READ ATTRIBUTE - 8Ch (beginning with E07)

The READ ATTRIBUTE command is defined in SPC-4 (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause decribes the specific implementation on E07 and later. The READ ATTRIBUTE command (see table 67) allows an application client to read attribute values from medium auxiliary memory.

Table 67 — READ ATTRIBUTE command

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	OPERATION CODE (8Ch)							
1	Reserved			SERVICE ACTION				
2	Obsolete							
4								
5	LOGICAL VOLUME NUMBER							
6	Reserved							
7	PARTITION NUMBER							
8	(MSB)	FIRST ATTRIBUTE IDENTIFIER						
9								(LSB)
10	(MSB)	ALLOCATION LENGTH						
13								(LSB)
14	Reserved							CACHE
15	Control Byte (see 5.1.3)							

If cached attribute information is not reported (e.g., the CACHE bit is set to zero or the CACHE bit is set to one and there is no cached attributes available) and there is no medium present, then the command is terminated with CHECK CONDITION status, with NOT READY, MEDIUM NOT PRESENT (2/3A00).

If cached attribute information is not available (e.g., the CACHE bit is set to zero) and the medium is present but the medium auxiliary memory is not accessible, then the READ ATTRIBUTE command is terminated with CHECK CONDITION status, with MEDIUM ERROR, LOGICAL UNIT NOT READY, AUXILIARY MEMORY NOT ACCESSIBLE (3/0410).

If the medium auxiliary memory is not operational, the READ ATTRIBUTE command is terminated with CHECK CONDITION status, with MEDIUM ERROR, AUXILIARY MEMORY READ ERROR (3/1112) and the Memory Chip in Cartridge Failure TapeAlert (i.e., 0Fh) is asserted.

The following parameters apply:

- SERVICE ACTION:

Value	**Description**
00h	ATTRIBUTE VALUES: Return attribute values as specified in ATTRIBUTE VALUES service action (see 5.2.21.1 on page 142)
01h	ATTRIBUTE LIST: Return a list of available attribute identifiers – identifiers that are in the read only state or in the read/write state (see 4.23) as specified in ATTRIBUTE LIST service action (see 5.2.21.2 on page 143)
02h	LOGICAL VOLUME LIST: Return a list of known logical volume numbers as defined in LOGICAL VOLUME LIST service action (see 5.2.21.3 on page 144)
03h	PARTITION LIST: Return a list of known partition numbers as defined in PARTITION LIST service action (see 5.2.21.4 on page 144)
05h	SUPPORTED ATTRIBUTES: Return a list of supported attribute identifiers – identifiers that are in the read only state, in the read/write state, or in the nonexistent state (see 4.23) as defined in SUPPORTED ATTRIBUTES service action (see 5.2.21.5 on page 145)

- LOGICAL VOLUME NUMBER: 00h
- PARTITION NUMBER: The partition of the attribute to be accessed.
- FIRST ATTRIBUTE IDENTIFIER: The attribute identifier of the first attribute to be returned.
- ALLOCATION LENGTH: The number of bytes allowed to be returned
- CACHE: Specifies whether or not to report attribute information cached from the most recently mounted volume. This bit is ignored if there is a volume mounted. Attribute information from cache is the complete set of attribute information from the most recently mounted volume. Cached attribute information is cleared at the start of a volume load.

Value	**Description**
0b	Do not report cached attribute information.
1b	Report cached attribute information.

5.2.21.1 ATTRIBUTE VALUES service action

The READ ATTRIBUTE command with ATTRIBUTE VALUES service action returns parameter data containing the attributes that are in the read state or read/write state (see 4.23) specified by the PARTITION NUMBER, LOGICAL VOLUME NUMBER, and FIRST ATTRIBUTE IDENTIFIER fields in the CDB. The returned parameter data shall contain the requested attributes in ascending numerical order by attribute identifier value and in the format shown in table 68.

Table 68 — READ ATTRIBUTE with ATTRIBUTE VALUES service action parameter list format

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	(MSB)	AVAILABLE DATA (n-3)						
3								(LSB)
	Attribute(s)							
4	Attribute 0 (see 6.5.1)							
	⋮							
	Attribute x (see 6.5.1)							
n								

The following parameters apply to the READ ATTRIBUTE command with ATTRIBUTE VALUES service action parameter list:

Byte	Description
0-3	The AVAILABLE DATA field shall contain the number of bytes of attribute information in the parameter list.
4-n	The attribute values for the attributes being returned. Attribute format (see 6.5.1 on page 363) describes the format of the attributes.

5.2.21.2 ATTRIBUTE LIST service action

The READ ATTRIBUTE command with ATTRIBUTE LIST service action returns parameter data containing the attribute identifiers for the attributes that are in the read only state or in the read/write state (see 4.23) in the specified partition and volume number. The contents of FIRST ATTRIBUTE IDENTIFIER field in the CDB shall be ignored. The returned parameter data shall contain the requested attribute identifiers in ascending numerical order by attribute identifier value and in the format shown in table 69.

Table 69 — READ ATTRIBUTE with ATTRIBUTE LIST service action parameter list format

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	(MSB)			AVAILABLE DATA (n-3)				
3								(LSB)
				Attribute identifiers				
4				ATTRIBUTE IDENTIFIER 0				
5								
				⋮				
n-1				ATTRIBUTE IDENTIFIER x				
n								

The following parameters apply to the READ ATTRIBUTE command with ATTRIBUTE LIST service action parameter list:

Byte	Description
0-3	The AVAILABLE DATA field shall contain the number of bytes of attribute identifiers in the parameter list.
4-n	An ATTRIBUTE IDENTIFIER field is returned for each attribute that is in the read only state or in the read/write state (see 4.23) in the specified partition and volume number. Attribute identifier values (see 6.5.2 on page 364) provides a description of the attribute identifier values.

5.2.21.3 LOGICAL VOLUME LIST service action

The READ ATTRIBUTE command with LOGICAL VOLUME LIST service action returns parameter data (see table 70) identifying the supported number of logical volumes. The contents of LOGICAL VOLUME NUMBER, PARTITION NUMBER, and FIRST ATTRIBUTE IDENTIFIER fields in the CDB shall be ignored.

Table 70 — READ ATTRIBUTE with LOGICAL VOLUME LIST service action parameter list format

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	(MSB)	AVAILABLE DATA (0002h)						
1								(LSB)
2	FIRST LOGICAL VOLUME NUMBER							
3	NUMBER OF LOGICAL VOLUMES AVAILABLE							

The following parameters apply to the READ ATTRIBUTE command with LOGICAL VOLUME LIST service action parameter list:

Byte	**Description**
0-1	The AVAILABLE DATA field shall contain two.
2	The FIRST LOGICAL VOLUME NUMBER field indicates the first volume available. Logical volume numbering should start at zero.
3	The NUMBER OF LOGICAL VOLUMES AVAILABLE field indicates the number of volumes available.

5.2.21.4 PARTITION LIST service action

The READ ATTRIBUTE command with PARTITION LIST service action returns parameter data (see table 71) identifying the number of partitions supported in the specified logical volume number. The contents of PARTITION NUMBER and FIRST ATTRIBUTE IDENTIFIER fields in the CDB shall be ignored.

Table 71 — READ ATTRIBUTE with PARTITION LIST service action parameter list format

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	(MSB)	AVAILABLE DATA (0002h)						
1								(LSB)
2	FIRST PARTITION NUMBER							
3	NUMBER OF PARTITIONS AVAILABLE							

The following parameters apply to the READ ATTRIBUTE command with PARTITION LIST service action parameter list:

Byte	**Description**
0-1	The AVAILABLE DATA field shall contain two.
2	The FIRST PARTITION NUMBER field indicates the first partition available on the specified logical volume number. Partition numbering starts at zero.
3	The NUMBER OF PARTITIONS AVAILABLE field indicates the number of partitions available on the specified logical volume number.

5.2.21.5 SUPPORTED ATTRIBUTES service action

The READ ATTRIBUTE command with SUPPORTED ATTRIBUTES service action returns parameter data containing the attribute identifiers for the attributes that are in the read only state, in the read/write state, or in the nonexistent state (see 4.23) in the specified partition and volume number. The contents of FIRST ATTRIBUTE IDENTIFIER field in the CDB shall be ignored. The returned parameter data shall contain the requested attribute identifiers in ascending numerical order by attribute identifier value and in the format shown in table 72.

Table 72 — READ ATTRIBUTE with SUPPORTED ATTRIBUTES service action parameter list format

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	(MSB)	AVAILABLE DATA (n-3)						
3								(LSB)
	Attribute identifiers							
4	ATTRIBUTE IDENTIFIER 0							
5								
	⋮							
n-1	ATTRIBUTE IDENTIFIER x							
n								

The following parameters apply to the READ ATTRIBUTE command with SUPPORTED ATTRIBUTES service action parameter list:

Byte	**Description**
0-3	The AVAILABLE DATA field shall contain the number of bytes of attribute identifiers in the parameter list.
4-n	An ATTRIBUTE IDENTIFIER field is returned for each attribute that is in the read only state, in the read/write state, or in the nonexistent state (see 4.23) in the specified partition and volume number. Attribute identifier values (see 6.5.2 on page 364) describes the attribute identifier values.

5.2.22 READ BLOCK LIMITS - 05h

The Read Block LImits command is defined in SSC (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause specifies the specific implementation.

Table 73 — Read Block Limits CDB

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Operation code (05h)							
1	Obsolete (LUN)			Reserved				MLOI
2	Reserved							
4								
5	Control Byte (see 5.1.3)							

The following parameters apply:

- maximum logical object identifier (MLOI):

Value	**Description**
0b	The data returned is the READ BLOCK LIMITS block length data (see 5.2.22.1).
1b	The data returned is the READ BLOCK LIMITS maximum logical object identifier data (see 5.2.22.2).

5.2.22.1 READ BLOCK LIMITS block length data

The format of the data returned in the READ BLOCK LIMITS Descriptor is shown in table 74.

Table 74 — READ BLOCK LIMITS block length data

Byte	Bit							
	7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	Reserved			GRANULARITY (0h)				
1	(MSB)	MAXIMUM BLOCK LENGTH LIMIT						
3								(LSB)
4	(MSB)	MINIMUM BLOCK LENGTH LIMIT						
5								(LSB)

The following parameters apply to the Read Block Limits block length data:

Byte	**Description**
0	

Bit	**Description**
7-5	Reserved
4-0	GRANULARITY (0h)

1-3	MAXIMUM BLOCK LENGTH LIMIT: 200000h (2,097,152 bytes
4-5	MINIMUM BLOCK LENGTH LIMIT: 0001h (1 byte)

Any block length in the range of MINIMUM BLOCK LENGTH LIMIT to MAXIMUM BLOCK LENGTH LIMIT is supported.

For further explanation, see “Data Transfer, Block Limits, and Fixed Block Option” on page 38

5.2.22.2 READ BLOCK LIMITS maximum logical object identifier data

The READ BLOCK LIMITS maximum logical object identifier data (see table 75) specifies the maximum value of the logical object identifier the logical unit supports.

Table 75 — READ BLOCK LIMITS maximum logical object identifier data

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Reserved							
…								
11								
12	(MSB)	MAXIMUM LOGICAL OBJECT IDENTIFIER						
…								
19								(LSB)

The following parameters apply to the Read Block Limits maximum logical object identifier data:

Byte	Description
0-11	Reserved
12-19	MAXIMUM LOGICAL OBJECT IDENTIFIER: The maximum value the device server supports in a logical object identifier field. This field is set to 0000_0000_FFFF_FFFFh.

NOTE 37 - The drive establishes early warning at a constant number of blocks prior to the MAXIMUM LOGICAL OBJECT IDENTIFIER. At the time this document was published, the drive assumes a block size of 32 KiB for these calculations, but this may change at any time. Additionally, programmable early warning, if configured, is returned prior to the early warning value using an assumed block size of 32 KiB.

5.2.23 READ BUFFER - 3Ch

The Read Buffer command is defined in SPC (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause specifies the specific implementation.

Table 76 — Read Buffer CDB

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Operation code (3C)							
1	Obsolete (LUN)			Reserved		Mode		
2	Buffer ID							
3	MSB	Buffer Offset						
5								LSB
6	MSB	Allocation Length						
8								LSB
9	Control Byte (see 5.1.3)							

The following parameters apply:

- Mode:

Value	**Description**
1h	MODE[01h] – Vendor Specific (see 5.2.23.1.1 on page 148)
2h	MODE[02h] – Data (see 5.2.23.1.2 on page 148)
3h	MODE[03h] – Descriptor (see 5.2.23.1.3 on page 149)
7h	MODE[07h] – Descriptor with algorithmic offset boundary (see 5.2.23.1.4 on page 150)

- Buffer ID: The supported buffers are described in table 79.
- Buffer Offset: In mode 1h and 2h, this is the starting address in the buffer to be read. For mode 3h and 7h, this field must be 000000h. See the mode description for how to interpret the BUFFER OFFSET field.

  NOTE 38 - If the Buffer Offset is not on the boundary specified in table 79, the device returns CHECK CONDITION status with associated sense data of 5/2400 (Illegal Request, Invalid Field in CDB).

- Allocation Length: The maximum number of bytes to be transferred starting at the offset specified in Buffer Offset.

  The device transfers the number of bytes specified in the Allocation Length field or the number of bytes in the header or buffer being read, whichever is less. This is not an error. The host may use mode 3h to query the amount of data in the specified buffer prior to reading such data with mode 1h or 2h.

Each buffer image has its own unique format, describing where certain key data may be found. Certain buffers contain embedded data in the buffer image describing the length of the total buffer image, and a CRC field that checks the total buffer image. Uploading the microcode buffer is one such example.

5.2.23.1 Buffer modes for READ BUFFER

5.2.23.1.1 MODE[01h] – Vendor Specific

Same as MODE[02h] – Data (see 5.2.23.1.2 on page 148).

5.2.23.1.2 MODE[02h] – Data

In this mode, the Data-In Buffer is filled only with logical unit buffer data. The BUFFER ID field of the command CDB specifies a buffer within the logical unit from which data shall be transferred.

The BUFFER OFFSET field of the command CDB contains the byte offset within the specified buffer from which data shall be transferred. In a WRITE BUFFER command the application client should conform to the offset boundary requirements returned in one of the READ BUFFER descriptors (MODE[03h] – Descriptor (see 5.2.23.1.3 on page 149) or MODE[07h] – Descriptor with algorithmic offset boundary (see 5.2.23.1.4 on page 150)). If the device server is unable to accept the specified buffer offset, the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST, and the additional sense code set to INVALID FIELD IN CDB.

5.2.23.1.3 MODE[03h] – Descriptor

In this mode, a maximum of four bytes of READ BUFFER descriptor information is returned. The BUFFER OFFSET field is reserved in this mode. The allocation length should be set to four or greater. The READ BUFFER descriptor is defined as shown in table 77.

Table 77 — READ BUFFER descriptor

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	OFFSET BOUNDARY							
1	(MSB)							
	BUFFER CAPACITY							
3								(LSB)

The following parameters apply:

Byte Description

0 OFFSET BOUNDARY:

For READ BUFFER commands, the OFFSET BOUNDARY field applies to the following modes:

a) data (i.e., 02h) (see 5.2.23.1.2).

For WRITE BUFFER commands, the OFFSET BOUNDARY field applies to the following modes:

a) MODE[02h] – Data (see 5.2.50.2 on page 221);
b) MODE[06h] – Download microcode with offsets and activate (see 5.2.50.5 on page 222); and
c) MODE[07h] – Download microcode with offsets, save, and activate (see 5.2.50.6 on page 222).

For data mode (i.e., 02h), the boundary alignment indicated by the OFFSET BOUNDARY field applies only to the buffer specified by the BUFFER ID field. For modes other than data to which the OFFSET BOUNDARY field applies, the boundary alignment applies regardless of the buffer specified by the BUFFER ID field.

Value	Description
others	Multiples of 2^{code} (e.g., 00h means multiples of 1 byte or no offset restrictions, 01h means multiples of 2 bytes or even offsets, 02h means multiples of 4 bytes)
FFh	000000h is the only supported buffer offset

1-3 BUFFER CAPACITY: The maximum size in bytes of the buffer specified by the BUFFER ID field for the:

a) READ BUFFER command with data mode (i.e., 02h); and
b) WRITE BUFFER command with data mode (i.e., 02h).

5.2.23.1.4 MODE[07h] – Descriptor with algorithmic offset boundary

In this mode, a maximum of four bytes of READ BUFFER descriptor information is returned. The BUFFER OFFSET field is reserved in this mode. The allocation length should be set to four or greater. The READ BUFFER descriptor is defined as shown in table 78.

Table 78 — READ BUFFER descriptor

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	SHIFT	NUMBER OF BITS TO SHIFT						
1	(MSB)	BUFFER CAPACITY						
3								(LSB)

The following parameters apply:

Byte Description

0

Bit Description

7 SHIFT: Indication of how to determine the bit shift used with the BUFFER CAPACITY field to determine the buffer capacity

Value	Description
0b	The value in the NUMBER OF BITS TO SHIFT field shall be ignored and the bit shift shall be 6
1b	The value in the NUMBER OF BITS TO SHIFT field indicates the bit shift

1-3 BUFFER CAPACITY: The actual buffer capacity is determined by the value in the BUFFER CAPACITY field left shifted by the bit shift amount which is indicated by the SHIFT bit and the NUMBER OF BITS TO SHIFT field

The BUFFER OFFSET field in the CDB shall be the desired offset shifted right by the bit shift amount.

5.2.23.2 Supported Buffers

Table 79 lists the supported buffers and their IDs:

Table 79 — Supported Buffer IDs (part 1 of 2)

ID[α]		RB modes	WB modes	Buffer Description	Offset Boundary
any[α]		-	4,5,6,7,Dh,Fh[α]	Microcode	4
00h		1,2,3	-	Dump Data - Microcode Dump	4
01h				Microcode	5
09h		1,2,3	-	Dump Data - Tapemap	4
0Ah		1,2,3	1,2	Test Buffer	4
20h		1,2,3	-	Cartridge Memory	4
21h		1,2,3	-	Cartridge Memory from EOD	2
30h		--	1,2	Microcode (do not reset)	4

Table 79 — Supported Buffer IDs (part 2 of 2)

ID[α]		RB modes	WB modes	Buffer Description	Offset Boundary
40h		1,2,3	-	String Search Match List[χ δ]	4
50h		1,2,3	-	Active IP addresses[δ]	FFh
81h		1,2,3	-	Dump Data - Tapemap (same as 09h)	4

Legend
- Not Applicable
NS Not Supported
[a] For Write Buffer, If Mode is 4, 5, 6, 7, Dh, or Fh then Buffer IDs other than 30h are ignored and the command is treated as a Buffer ID 01h - Microcode.
[b] For this buffer, the Buffer Offset must be 000000h.
[c] Depending on the search methodology, the size of the search buffer results may be returned as a constant and may be larger than the amount of data actually returned. The structure of the returned data contains sufficient descriptive lengths for proper parsing and is defined in 5.2.45 on page 211.
[d] Supported in E07 and later only.
[e]

5.2.23.3 Buffer ID 50h: Active IP addresses

The Active IP addresses buffer returns a list of IP addresses that have successfully completed negotiation on the Ethernet link and are active. This list does not contain IP addresses that the drive attempts but cannot use due to conflicts or other errors. The format of the Data field of the active IP addresses buffer depends on the value in byte 0. If byte 0 contains a non-zero value, then the format is defined in Active IP addresses fixed buffer (see 5.2.23.3.1). If byte 0 contains a zero value, then the format is defined in the Active IP addresses variable buffer (see 5.2.23.3.2).

5.2.23.3.1 Active IP addresses fixed buffer

The format of the Data field of the active IP addresses fixed buffer is described in table 80

.

Table 80 — Active IP addresses fixed buffer format

Byte	Bit							
	7	6	5	4	3	2	1	0
0	NUMBER OF ETHERNET PORTS							
List of Ethernet port descriptors								
1	Ethernet port fixed descriptor [first]							
x								
⋮								
y	Ethernet port fixed descriptor [last]							
n								

The following parameters apply:

Byte	Description
0	NUMBER OF ETHERNET PORTS - The number of physical Ethernet ports functioning in the drive.
1-n	List of Ethernet port descriptors. There will be one Ethernet port descriptor (see table 81) for each physical Ethernet port functioning in the drive. The size of each descriptor may vary from Ethernet port to Ethernet port.

The Ethernet port descriptor format is defined in table 81

Table 81 — Ethernet port fixed descriptor format

Byte	Bit 7	6	5	4	3	2	1	0
0	SIZE OF SOCKADDR (M-1)							
1	number of sockaddr structures							
List of sockaddr structures (see 6.6.22.3.1.1)								
2	sockaddr [first]							
m								
⋮								
n-m	sockaddr [last]							
n								

The following parameters apply:

Byte	Description
0	SIZE OF SACKADDR - The size of each sockaddr. All sockaddr's are the same size
1	number of sockaddr - The number of sockaddr structures returned.
2-n	List of sockaddr structures. Each sockaddr structure describes one active IP address. The sockaddr structure is defined in clause 6.6.22.3.1.1.

5.2.23.3.2 Active IP addresses variable buffer

The format of the Data field of the active IP addresses buffer is described in table 80

.

Table 82 — Active IP addresses variable buffer format

Byte	Bit 7	6	5	4	3	2	1	0
0	Reserved							
1								
2	ACTIVE IP ADDRESSES BUFFER LENGTH (N-3)							
3								
4	Reserved							
6								
7	NUMBER OF ETHERNET PORTS							
List of Ethernet port descriptors								
1	Ethernet port variable descriptor [first]							
⋮								
	Ethernet port variable descriptor [last]							
n								

The following parameters apply:

Byte	Description
0	NUMBER OF ETHERNET PORTS - The number of physical Ethernet ports functioning in the drive.
1-n	List of Ethernet port descriptors. There will be one Ethernet port descriptor (see table 83) for each physical Ethernet port functioning in the drive. The size of each descriptor may vary from Ethernet port to Ethernet port.

The Ethernet port variable descriptor format is defined in table 83

Table 83 — Ethernet port variable descriptor format

Byte	Bit							
	7	6	5	4	3	2	1	0
0	DESCRIPTOR LENGTH (M-1)							
1								
2	PORT IDENTIFIER							
3								
4	Reserved							
6								
7	number of sockaddr structures							
List of Ethernet socket address descriptors 6.6.22.3.1.1								
8	Ethernet socket address descriptor [first]							
⋮								
	Ethernet socket address descriptor [last]							
m								

The following parameters apply:

Byte	Description
0	SIZE OF SACKADDR - The size of each sockaddr. All sockaddr's are the same size
1	number of sockaddr - The number of sockaddr structures returned.
2-n	List of Ethernet socket address descriptors. Each Ethernet socket address descriptor describes one active IP address and has the RELATIVE SOCKET ADDRESS IDENTIFIER field set to zero. The Ethernet socket address descriptor is defined in 6.6.22.3.1.1.

5.2.24 READ DYNAMIC RUNTIME ATTRIBUTE - A3h[1Eh] or D1h (beginning with E07)

The READ DYNAMIC RUNTIME ATTRIBUTE (DRA) command has a legacy format shown in table 84 and the standardized format shown in table 84.

Table 84 — READ DRA CDB (legacy)

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (D1h)							
1	Reserved			SERVICE ACTION				
2	FIRST DYNAMIC RUNTIME ATTRIBUTE IDENTIFIER							
3								
4	LAST DYNAMIC RUNTIME ATTRIBUTE IDENTIFIER							
5								
6	(MSB)							
…	ALLOCATION LENGTH							
9								(LSB)
10	Reserved							
11	Control							

Table 85 — READ DYNAMIC RUNTIME ATTRIBUTE CDB (standardized - supported after GA)

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (A3h)							
1	ATTRIBUTE REPORT TYPE			SERVICE ACTION (1Eh)				
2	FIRST DYNAMIC RUNTIME ATTRIBUTE IDENTIFIER							
3								
4	LAST DYNAMIC RUNTIME ATTRIBUTE IDENTIFIER							
5								
6	(MSB)							
…	ALLOCATION LENGTH							
9								(LSB)
10	Reserved							
11	Control							

The format of parameter data returned by the READ DYNAMIC RUNTIME ATTRIBUTE command depends on the service action for the legacy D1h command or the attribute report type for the standardized A3h[1Eh] command as specified in (see 5.2.24.1).

The FIRST DYNAMIC RUNTIME ATTRIBUTE IDENTIFIER field specifies the dynamic runtime attribute identifier of the first attribute to be returned. Only attributes with a dynamic runtime attribute identifier greater than or equal to the value specified in the FIRST DYNAMIC RUNTIME ATTRIBUTE IDENTIFIER field and that are not in the nonexistent or unsupported state shall be reported. It shall not be considered an error if the specified dynamic runtime attribute is in the unsupported or nonexistent state.

The LAST DYNAMIC RUNTIME ATTRIBUTE IDENTIFIER field specifies the dynamic runtime attribute identifier of the last attribute to be returned. Only attributes with a dynamic runtime attribute identifier less than or equal to the value specified in the LAST DYNAMIC RUNTIME ATTRIBUTE IDENTIFIER field and that are not in the nonexistent or unsupported state shall be reported. It shall not be considered an error if the specified dynamic runtime attribute is in the unsupported or nonexistent state. If the attribute identifier specified in the LAST DYNAMIC RUNTIME ATTRIBUTE IDENTIFIER field is less than the attribute identifier specified in the FIRST DYNAMIC RUNTIME ATTRIBUTE

IDENTIFIER field, then the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST, and the additional sense code set to INVALID FIELD IN CDB.

If the ALLOCATION LENGTH specified causes an attribute to be truncated this shall not be considered an error. All data up to the allocation length shall be returned.

5.2.24.1 READ DYNAMIC RUNTIME ATTRIBUTE Service Action

The actions defined for the READ DYNAMIC RUNTIME ATTRIBUTE command are shown in Table 86..

Table 86 — READ DYNAMIC RUNTIME ATTRIBUTE Service Action codes

Standardized (OPERATION CODE **A3h[1Eh]**)	**Legacy** (OPERATION CODE **D1h**)	**Name**	**Description**	**Ref**
ATTRIBUTE REPORT TYPE	SERVICE ACTION			
000b	00h	SUPPORTED ATTRIBUTES	Return a list of dynamic runtime attribute identifiers that the device server supports. No indication of attribute state is implied.	5.2.24.2
001b	10h	ATTRIBUTE VALUES FOR THIS I_T NEXUS	Return values for: a) all logical unit type attributes; b) all target type attributes associated with the I_T nexus through which the command was received; and c) all initiator type attributes associated with the I_T nexus through which the command was received.	5.2.24.3
010b	11h	ATTRIBUTE VALUES FOR ALL I_T NEXUSES	Return values for: a) all logical unit type attributes; b) all target type attributes associated with all I_T nexuses; and c) all initiator type attributes associated with all I_T nexuses.	5.2.24.4
others		Reserved		

5.2.24.2 SUPPORTED ATTRIBUTES service action

The READ DYNAMIC RUNTIME ATTRIBUTE command with SUPPORTED ATTRIBUTES selected (SERVICE ACTION field set to 00h in legacy command or ATTRIBUTE REPORT TYPE set to 000b in standardized command) returns parameter data containing the attribute identifiers that the device server supports. The returned parameter data shall contain the requested attribute identifiers in ascending numerical order by attribute identifier and in the format shown in Table 87.

Table 87 — READ DRA with SUPPORTED ATTRIBUTES service action parameter list format

Bit Byte	7	6	5	4	3	2	1	0
0	ATTRIBUTE REPORT TYPE			SERVICE ACTION				
1	Reserved							
…								
3								
4	AVAILABLE DATA (n-7)							
…								
7								
	Attribute Identifiers							
8	Attribute Identifier 0							
9								
⋮								
n-1	Attribute Identifier x							
n								

The SERVICE ACTION field and ATTRIBUTE REPORT TYPE field shall contain the values shown in table 88

Table 88 — Byte one of the parameter list data

Field	Legacy command	Standardized command
SERVICE ACTION	The value of the SERVICE ACTION field in the CDB	00h
ATTRIBUTE REPORT TYPE	000b	The value of the ATTRIBUTE REPORT TYPE field in the CDB

The AVAILABLE DATA field shall contain the number of bytes of attribute identifiers in the parameter list. The AVAILABLE DATA field shall not be adjusted by the CDB ALLOCATION LENGTH field.

An attribute identifier is returned for each attribute that the device server supports. No indication of the current state of the reported attributes is made. See 6.2.2 for a description of the attribute identifier.

5.2.24.3 ATTRIBUTE VALUES FOR THIS I_T NEXUS service action

The READ DYNAMIC RUNTIME ATTRIBUTE command with ATTRIBUTE VALUES FOR THIS I_T NEXUS selected (SERVICE ACTION field set to 10h in legacy command or ATTRIBUTE REPORT TYPE set to 001b in standardized command) returns parameter data containing the attributes for the I_T_L Nexus through which this command is received starting with the FIRST ATTRIBUTE IDENTIFIER field in the CDB and ending with the LAST ATTRIBUTE IDENTIFIER field in the CDB.

The returned parameter data shall contain the requested attributes in the format shown in table 89 and in ascending numerical order by I_T nexus index then attribute identifier

Table 89 — READ DRA with ATTRIBUTE VALUES FOR THIS I_T NEXUS service action parameter list format

Bit Byte	7	6	5	4	3	2	1	0
0	ATTRIBUTE REPORT TYPE			SERVICE ACTION				
1	Reserved							
…								
3								
4	AVAILABLE DATA (n-7)							
…								
7								
	Attribute(s)							
8	Attribute 0 (see 6.2.1)							
…								
⋮								
	Attribute x (see 6.2.1)							
…								
n								

The SERVICE ACTION field and ATTRIBUTE REPORT TYPE field shall contain the values shown in table 88

Table 90 — Byte one of the parameter list data

Field	Legacy command	Standardized command
SERVICE ACTION	The value of the SERVICE ACTION field in the CDB	00h
ATTRIBUTE REPORT TYPE	000b	The value of the ATTRIBUTE REPORT TYPE field in the CDB

The AVAILABLE DATA field shall contain the number of bytes of attribute information in the parameter list. The value in the AVAILABLE DATA field shall not be adjusted by the CDB ALLOCATION LENGTH field.

The format of the attributes is described in 6.2.1.

5.2.24.4 ATTRIBUTE VALUES FOR ALL I_T NEXUSES service action

The READ DYNAMIC RUNTIME ATTRIBUTE command with ATTRIBUTE VALUES FOR ALL I_T NEXUSES service action (i.e., the SERVICE ACTION field set to 11h) returns parameter data containing the attributes for all known I_T nexuses and starting with the FIRST ATTRIBUTE IDENTIFIER field in the CDB and ending with the LAST ATTRIBUTE IDENTIFIER field in the CDB.

The returned parameter data shall contain the requested attribute values for all I_T nexus in ascending numerical order by I_T nexus index then attribute identifier and in the format shown in table 91.

Table 91 — READ DRA with ATTRIBUTE VALUES FOR ALL I_T NEXUSES service action parameter list format

Bit Byte	7	6	5	4	3	2	1	0
0	ATTRIBUTE REPORT TYPE			SERVICE ACTION				
1	Reserved							
…								
3								
4	AVAILABLE DATA (n-7)							
…								
7								
	Attribute(s)							
8	Attribute 0 (see 6.2.1)							
…								
⋮								
	Attribute x (see 6.2.1)							
…								
n								

The SERVICE ACTION field and ATTRIBUTE REPORT TYPE field shall contain the values shown in table 88

Table 92 — Byte one of the parameter list data

Field	Legacy command	Standardized command
SERVICE ACTION	The value of the SERVICE ACTION field in the CDB	00h
ATTRIBUTE REPORT TYPE	000b	The value of the ATTRIBUTE REPORT TYPE field in the CDB

The AVAILABLE DATA field shall contain the number of bytes to follow. The value in the AVAILABLE DATA field shall not be adjusted by the CDB ALLOCATION LENGTH field.

The format of the attributes is described in 6.2.1.

5.2.25 READ POSITION - 34h

5.2.25.1 READ POSITION command description

The Read Position command is defined in SSC (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause describes the specific implementation.

The READ POSITION command (see table 93) reports the current position and provides information about logical objects contained in the object buffer. No medium movement shall occur as a result of responding to the command.

Table 93 — READ POSITION command

Byte	**Bit**							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
	OPERATION CODE (34h)							
	Reserved			SERVICE ACTION				
2 6	Reserved							
7 8	(MSB)			ALLOCATION LENGTH				(LSB)
9	Control Byte (see 5.1.3)							

The following parameters apply:

- SERVICE ACTION:

Value	**Description**
00h	SHORT FORM-- BLOCK ID: Device server shall return 20 bytes of data with the FIRST LOGICAL OBJECT LOCATION and LAST LOGICAL OBJECT LOCATION fields as logical object identifier values, relative to a partition (see 5.2.25.2). The ALLOCATION LENGTH field shall be zero.

WARNING

The SHORT FORM -- BLOCK ID may become obsolete in future standards.

WARNING

06h	LONG FORM: Device server shall return 32 bytes of data (see 5.2.25.3). The ALLOCATION LENGTH field shall be zero.
08h	EXTENDED FORM: Device server shall return 32 bytes of data up to the maximum length specified by the ALLOCATION LENGTH field (see 5.2.25.4).
others	The command is terminated with CHECK CONDITION status with the sense key set to ILLEGAL REQUEST, and the additional sense code set to INVALID FIELD IN CDB.

- ALLOCATION LENGTH: Length of the data to return. Shall be set to 0000h if the SERVICE ACTION field is set to 00h or 06h.

5.2.25.2 READ POSITION data format, short form

Table 94 specifies the READ POSITION data that shall be returned if the SERVICE ACTION field is 00h.

The short form is included for legacy applications. It is highly recommended that the LONG FORM (06h) (see 5.2.25.3) or the EXTENDED FORM (08h) (see 5.2.25.4) be used instead.

WARNING

The short form breaks when there are greater than 2^{32} logical objects on medium and may become obsolete in future standards.

WARNING

Table 94 — READ POSITION data format, short form

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Position information/validity							
	BOP	EOP	LOCU	BYCU	Rsvd	LOLU	PERR	BPEW
1	PARTITION NUMBER							
2	Reserved							
3	Reserved							
4	(MSB)	FIRST LOGICAL OBJECT LOCATION						
7								(LSB)
8	(MSB)	LAST LOGICAL OBJECT LOCATION						
11								(LSB)
12	RESERVED							
13	(MSB)	NUMBER OF LOGICAL OBJECTS IN OBJECT BUFFER						
15								(LSB)
16	(MSB)	NUMBER OF BYTES IN OBJECT BUFFER						
19								(LSB)

Byte **Description**

0 Position Information/Validity:

Bit **Description**

7 BOP (beginning of partition):

Value	**Description**
0b	the current logical position is not at the beginning of partition.
1b	the device is at the beginning of the current partition.

6 EOP (end of partition)

Value	**Description**
0b	the device is not between early warning and end of partition.
1b	the device is positioned between early warning and end of the current partition.

5 LOCU (logical object count unknown)

Value	**Description**
0b	block count is exact
1b	block count is an estimate

4 BYCU (byte count unknown)

Value	**Description**
0b	byte count is exact
1b	byte count is an estimate

3 Reserved

2 LOLU (logical object location unknown)

Value	**Description**
0b	block position is exact
1b	block position is an estimate

1 PERR (position error):

Value	**Description**
0b	An overflow has not occurred in any of the returned position data fields.

1b	An overflow has occurred in at least one of the returned position data fields. The application should use the LONG FORM (06h) (see 5.2.25.3) to obtain the current position or the application should use the EXTENDED FORM (08h) (see 5.2.25.4) to obtain the current position and number of bytes in the object buffer.

0 BPEW (beyond programmable early warning)

Value	**Description**
0b	The LOLU bit is set to one, the PEWS field of the MP 10h[01h]: Device Configuration Extension (see 6.6.12 on page 397) is set to zero, or the logical object location is not in a PEWZ or on the EOP side of EW.
1b	The logical object location is in a PEWZ or on the EOP side of EW.

1 PARTITION NUMBER:

Reports the partition number for the current logical position. When the volume has only one partition, this field is set to 00h.

2-3 Reserved

4-7 FIRST LOGICAL OBJECT LOCATION:

Specifies the logical object address associated with the current logical position, in the range 0000 0000h to FFFF FFFFh. The value indicates the logical object address of the next data block or filemark to be transferred between the initiator and the target if a READ or WRITE command is issued.

8-11 LAST LOGICAL OBJECT LOCATION:

After a write command, this field specifies the logical object address associated with the next logical object to be transferred from the buffer to the medium, in the range 0000 0000h to FFFF FFFFh. After a read command, this field specifies the logical object address associated with the last (most recent) data block or filemark to be transferred from the medium to the buffer. For any case where the buffer no longer contains a whole block of data or is empty, the value reported for the LAST LOGICAL OBJECT LOCATION is equal to the value reported for the FIRST LOGICAL OBJECT LOCATION.

12 Reserved

13-15 NUMBER OF LOGICAL OBJECTS IN THE OBJECT BUFFER:

The number of data blocks and filemarks in the buffer that have not been written to the medium. (This value is zero if the device is reading rather than writing.)

16-19 NUMBER OF BYTES IN THE OBJECT BUFFER: The total number of write data bytes (before compaction) in the buffer that have not been written to the medium.

5.2.25.3 READ POSITION data format, long form

Table 95 specifies the format of the READ POSITION data that shall be returned if the SERVICE ACTION field is 06h.

Table 95 — READ POSITION data format, long form

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Position information/validity							
	BOP	EOP	Reserved		MPU	LONU	Rsvd	BPEW
1	Reserved							
3								
4	(MSB)	PARTITION NUMBER						
7								(LSB)
8	(MSB)	LOGICAL OBJECT NUMBER						
15								(LSB)
16	(MSB)	LOGICAL FILE IDENTIFIER						
23								(LSB)
24	(MSB)	Obsolete						
31								(LSB)

Byte **Description**

0 Position Information/Validity:

Bit **Description**

7 BOP (beginning of partition):

Value	**Description**
0b	the current logical position is not at the beginning of partition.
1b	the device is at the beginning of the current partition.

6 EOP (end of partition)

Value	**Description**
0b	the device is not between early warning and end of partition.
1b	the device is positioned between early warning and end of the current partition.

5-4 Reserved

3 MPU (mark position unknown)

Value	**Description**
0b	the LOGICAL FILE IDENTIFIER field contains valid position information.
1b	the logical file identifier is not known or accurate reporting is not currently available.

2 LONU (logical object number unknown)

Value	**Description**
0b	the LOGICAL OBJECT NUMBER and PARTITION NUMBER fields contain exact information.
1b	The logical object number is an estimate.

1 Rsvd (Reserved)

0 BPEW (beyond programmable early warning)

Value	**Description**
0b	The LOLU bit is set to one, the PEWS field in the MP 10h[01h]: Device Configuration Extension (see 6.6.12 on page 397) is set to zero, or the logical object location is not in a PEWZ or on the EOP side of EW.
1b	The logical object location is in a PEWZ or on the EOP side of EW.

1-3 Reserved

4-7	PARTITION NUMBER: The partition number for the current logical position.
8-15	LOGICAL OBJECT NUMBER: The number of logical objects between beginning-of-partition and the current logical position. A filemark counts as one logical object.
16-23	LOGICAL FILE NUMBER: The number of filemarks between beginning-of-partition and the current logical position. This value is the current logical file identifier.
24-31	Obsolete.

5.2.25.4 READ POSITION data format, extended form

Table 96 specifies the format of the READ POSITION data that shall be returned if the SERVICE ACTION field is 08h.

Table 96 — READ POSITION data format, extended form

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Position information/validity							
	BOP	EOP	LOCU	BYCU	Rsvd	LOLU	PERR	BPEW
1	PARTITION NUMBER							
2	(MSB)	ADDITIONAL LENGTH (1Ch)						
3								(LSB)
4	Reserved							
5	(MSB)	NUMBER OF LOGICAL OBJECTS IN OBJECT BUFFER						
7								(LSB)
8	(MSB)	FIRST LOGICAL OBJECT LOCATION						
15								(LSB)
16	(MSB)	LAST LOGICAL OBJECT LOCATION						
23								(LSB)
24	(MSB)	NUMBER OF BYTES IN OBJECT BUFFER						
31								(LSB)

The fields are defined the same as for the corresponding fields in the READ POSITION data format, short form (see table 94).

The ADDITIONAL LENGTH field shall contain 1Ch. If the information transferred to the Data-In Buffer is truncated because of an insufficient ALLOCATION LENGTH value, the ADDITIONAL LENGTH field shall not be altered to reflect the truncation.

Byte Description

0 Position Information/Validity:

Bit Description

7 BOP (beginning of partition):

Value	Description
0b	the current logical position is not at the beginning of partition.
1b	the device is at the beginning of the current partition.

6 EOP (end of partition)

Value	Description
0b	the device is not between early warning and end of partition.
1b	the device is positioned between early warning and end of the current partition.

5 LOCU (logical object count unknown)

Value	Description
0b	block count is exact
1b	block count is an estimate

4 BYCU (byte count unknown)

Value	Description
0b	byte count is exact
1b	byte count is an estimate

3 Reserved

2 LOLU (logical object location unknown)

Value	Description
0b	block position is exact
1b	block position is an estimate

1 PERR (position error):

Value	Description
0b	An overflow has not occurred in any of the returned position data fields.
1b	An overflow has occurred in at least one of the returned position data fields.

0 BPEW (beyond programmable early warning)

Value	Description
0b	The LOLU bit is set to one, the PEWS field in MP 10h[01h]: Device Configuration Extension (see 6.6.12 on page 397) is set to zero, or the logical object location is not in a PEWZ or on the EOP side of EW.
1b	The logical object location is in a PEWZ or on the EOP side of EW.

1 PARTITION NUMBER:

Reports the partition number for the current logical position. When the volume has only one partition, this field is set to 00h.

2-3 ADDITIONAL LENGTH: 1Ch

4 Reserved

5-7 NUMBER OF LOGICAL OBJECTS IN THE OBJECT BUFFER:

The number of data blocks and filemarks in the buffer that have not been written to the medium. (This value is zero if the device is reading rather than writing.)

8-15 FIRST LOGICAL OBJECT LOCATION:

Specifies the logical object address associated with the current logical position, in the range 0000 0000h to FFFF FFFFh. The value indicates the logical object address of the next data block or filemark to be transferred between the initiator and the target if a READ or WRITE command is issued.

16-23 LAST LOGICAL OBJECT LOCATION:

After a write command, this field specifies the logical object address associated with the next logical object to be transferred from the buffer to the medium, in the range 0000 0000h to FFFF FFFFh. After a read command, this field specifies the logical object address associated with the last (most recent) data block or filemark to be transferred from the medium to the buffer. For any case where the buffer no longer contains a whole block of data or is empty, the value reported for the LAST LOGICAL OBJECT LOCATION is equal to the value reported for the FIRST LOGICAL OBJECT LOCATION.

24-31 NUMBER OF BYTES IN THE OBJECT BUFFER: The total number of write data bytes (before compaction) in the buffer that have not been written to the medium.

5.2.26 READ REVERSE - 0Fh

The Read Reverse command is defined in SSC (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause specifies the specific implementation.

Table 97 — Read Reverse CDB

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Operation code (0Fh)							
1	Obsolete (LUN)			Reserved		Byte Order	SILI	Fixed
2	MSB	Transfer Length						
4								LSB
5	Control Byte (see 5.1.3)							

The following parameters apply:

- Byte Order field: 1b

  This field defines the order in which bytes are transferred to the initiator.

  Byte Order to the initiator is in the logical forward direction (first byte written is transferred to the initiator before the last byte written).

NOTE 39 - The Byte Order field is Vendor-Unique and not specified in any standards document. In those documents this bit is Reserved and is a 0b. The current product does not support reversed byte order and the bit is required to be set to 1b. An attempt to set this bit to 0b results in a check condition with associated sense data of 5/2400 (Illegal Request, Invalid Field in CDB). Nevertheless, as specified in the standard, the ending position of the medium is before the last block transferred.

- SILI (Suppress Incorrect Length Indicator) (per SCSI-2 standard) (see 4.10)
- Fixed (see 4.8)
- Transfer Length(see 4.8)

For more information, see “General Read-Type Handling” on page 40

5.2.27 RECEIVE COPY STATUS(LID4) – 84h[05h]

The RECEIVE COPY STATUS(LID4) command (see table 98) returns status for the copy operation (see 4.28.4.2.1) specified by the LIST IDENTIFIER field (see 4.28.4.2.2) in the CDB.

Table 98 — RECEIVE COPY STATUS(LID4) command

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (84h)							
1	Reserved			SERVICE ACTION (05h)				
2	(MSB)							
…	LIST IDENTIFIER							
5								(LSB)
6								
…	Reserved							
9								
10	(MSB)							
…	ALLOCATION LENGTH							
13								(LSB)
14	Reserved							
15	CONTROL							

The following parameters apply:

- LIST IDENTIFIER: Specifies the copy operation about which information is to be transferred (see 4.28.4.2.2).
- ALLOCATION LENGTH: The maximum number of bytes the application client is able to accept.

5.2.27.1 Parameter data for the RECEIVE COPY STATUS(LID4) command

The data returned by a RECEIVE COPY STATUS(LID4) command is available from the time the copy operation (see 4.28.4.2.1) starts until the device starts to validate the parameter data for the next XCOPY command.

After the completion of a copy operation (i.e., whenever the COPY OPERATION STATUS field is set to 01h, 02h, or 60h), the copy manager preserves all data returned by a RECEIVE COPY STATUS(LID4) command until a subsequent copy operation is started.

Table 99 shows the format of the parameter data returned by the copy manager in response to the RECEIVE COPY STATUS(LID4) command.

Table 99 — Parameter data for the RECEIVE COPY STATUS(LID4) command

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)	AVAILABLE DATA (n-3)						
…								
3								(LSB)
4	Reserved			RESPONSE TO SERVICE ACTION				
5	Reserved	COPY OPERATION STATUS						
6	(MSB)	OPERATION COUNTER						
7								(LSB)
8	(MSB)	ESTIMATED STATUS UPDATE DELAY						
…								
11								(LSB)
12	EXTENDED COPY COMPLETION STATUS							
13	LENGTH OF THE SENSE DATA FIELD (X-31)							
14	SENSE DATA LENGTH							
15	TRANSFER COUNT UNITS							
16	(MSB)	TRANSFER COUNT						
…								
23								(LSB)
24	(MSB)	SEGMENTS PROCESSED						
25								(LSB)
26	Reserved							
…								
31								
32	SENSE DATA							
…								
x								

Byte	Description
0-3	AVAILABLE DATA: The number of bytes that follow in the parameter data. The contents of the AVAILABLE DATA field are not altered based on the ALLOCATION LENGTH.
4	

Bit	Description
7-5	Reserved
4-0	RESPONSE TO SERVICE ACTION: The value in the SERVICE ACTION field of the XCOPY command (see 5.2.5) which started the copy operation specified by the LIST IDENTIFIER field in the CDB.

5

Bit	Description
7	Reserved
6-0	COPY OPERATION STATUS: The status of the copy operation specified by the LIST IDENTIFIER field in the CDB.

Value	Description
01h	Operation completed without errors
02h	Operation completed with errors
10h	Operation in progress, foreground or background unknown
11h	Operation in progress in foreground
12h	Operation in progress in background
60h	Operation terminated (e.g., by the preemption of a persistent reservation (see 4.28.3.4)). This operation did not complete.

6-7 The OPERATION COUNTER: a wrapping counter of the number of SCSI commands, or equivalent, that the copy manager has sent to a copy source or copy destination as part of processing the copy operation specified by the LIST IDENTIFIER field in the CDB.

8-11 ESTIMATED STATUS UPDATE DELAY: The number of milliseconds that the copy manager recommends that the application client wait before sending another RECEIVE COPY STATUS(LID4) command on the same I_T nexus with the same list identifier.

Value	Description
FFFF FFFEh	The COPY OPERATION STATUS field is set to 01h, 02h, or 60h.
FFFF FFFFh	The copy manager is unable to recommend a delay interval.
other	The copy manager may return the same parameter data if a RECEIVE COPY STATUS(LID4) command is received sooner than the indicated time

12 EXTENDED COPY COMPLETION STATUS: The status code, established for the completed copy operation specified by the LIST IDENTIFIER field.

If the IMMED bit, is set to one in the third-party copy command that originated the copy operation, then the contents of the EXTENDED COPY COMPLETION STATUS field may be different than the status returned by the originating third-party copy command.

If the COPY OPERATION STATUS field is set to 10h, 11h, or 12h then this field is reserved.

13 LENGTH OF THE SENSE DATA FIELD: The number of bytes in the SENSE DATA field. Sense data is padded to a 4-byte boundary.

14 SENSE DATA LENGTH: The number of bytes in the SENSE DATA field that contain sense data, if the COPY OPERATION STATUS field is set to 01h or 02h. This field is reserved otherwise.

15 TRANSFER COUNT UNITS: The units for the TRANSFER COUNT field.

Value	Description (See 2.2.2)
00h	Bytes
01h	Kibibytes
02h	Mebibytes
03h	Gibibytes
04h	Tebibytes
05h	Pebibytes
06h	Exbibytes

16-23 TRANSFER COUNT: The amount of data written to a copy destination for the copy operation specified by the LIST IDENTIFIER field in the CDB prior to receiving the RECEIVE COPY STATUS(LID4) command.

24-25 SEGMENTS PROCESSED: The number of segments the copy manager has processed for the copy operation specified by the LIST IDENTIFIER field in the CDB including the segment currently being processed. The SEGMENTS PROCESSED field is set to zero if the copy manager has not yet begun processing segment descriptors.

26-31 Reserved

32-x SENSE DATA: Sense data, if any, for the copy operation specified by the LIST IDENTIFIER field. The sense data format for sense keys other than COPY ABORTED (i.e., Ah) is described in Sense Data Format for most Sense Keys (see 6.8.2 on page 481). The sense data format for sense key COPY ABORTED (i.e., Ah) is described in Sense data format for COPY ABORTED (i.e., Sense Key Ah) (see 6.8.3 on page 491). This is padded to a 4-byte boundary, if necessary.

5.2.28 RECEIVE DIAGNOSTIC RESULTS - 1Ch

The Receive Diagnostic Results command is defined in SPC (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause specifies the specific implementation.

After a Send Diagnostic command completes, use the Receive Diagnostic Results command to receive the results.

As allowed by the SCSI standards, devices may implement Send Diagnostic and handle status and return information for diagnostics in somewhat different manners. To work correctly with all common variations, the following sequence should be used by the host.

The Send Diagnostic command is issued. If a CHECK CONDITION status occurs, and a non-deferred permanent error results, then the test failed (or was not started if sense key is ILLEGAL REQUEST). If GOOD status is returned, then a Receive Diagnostics Results command should be issued, and:

a) the “blocked” and “error” bit should be checked (byte 8, bits 2 and 0 respectively), and if either is set, the diagnostic was either not run or failed; and
b) if byte 8 bit 1 is set, the SIM/MIM data in the results is valid and contains additional information relating to the failure

NOTE 40 - This process is required by this device to determine diagnostics results for those diagnostics which indicate diagnostics results data is returned. GOOD status returned from the Send Diagnostic only indicates that the diagnostic was accepted but does not indicate the ultimate result of its execution.

NOTE 41 - For diagnostics that produce diagnostic results, the device should be reserved using the Reserve or Persistent Reserve Out command to protect the results from actions of other initiators.

Table 100 shows the command format.

Table 100 — Receive Diagnostic Results CDB

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Operation code (1Ch)							
1	Obsolete (LUN)			Reserved				
2	Reserved							
3	MSB	Allocation Length						
4								LSB
5	Control Byte (see 5.1.3)							

The following parameters apply:

- Allocation Length is the maximum number of bytes to be returned in the page of data following the command (if any).

Clause *6.1* has a listing of all diagnostic parameters.

5.2.29 RECEIVE RECOMMENDED ACCESS ORDER (RRAO) - A3h[1Dh] (beginning with E07)

The RECEIVE RECOMMENDED ACCESS ORDER (RRAO) command is used to retrieve a recommended access order of User Data Segments. Recommended access order (RAO) (beginning with E07) (see 4.27 on page 87) describes the purpose of the recommended access order.

After a GENERATE RECOMMENDED ACCESS ORDER (GRAO) command (see 5.2.8) completes, the RRAO command may be used to receive the results. The results are not cleared when read. See Recommended access order usage (see 4.27.3 on page 88) for the recommended usage and for restrictions on the validity of the RAO list returned in this command.

NOTE 42 - Table 101 shows the command format.

Table 101 — RECEIVE RECOMMENDED ACCESS ORDER CDB

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	OPERATION CODE (A3h)							
1	UDS_LIMITS	Reserved		SERVICE ACTION (1Dh)				
2								
…	RAO LIST OFFSET							
5								
6	MSB							
…	ALLOCATION LENGTH							
9								LSB
10	Reserved					UDS_TYPE		
11	Control Byte							

The following parameters apply:

- UDS_LIMITS - Specifies if the UDS Limits page is to be returned of if the RAO parameter list is to be returned.

Value	**Description**
0b	Return the RAO parameter list (see 5.2.29.1.2)
1b	Return the UDS Limits page using the settings of the other fields in the CDB to determine the values supported (see 5.2.29.1.1). If this field is set to 1b, then the RAO LIST OFFSET field shall be set to zero. If the RAO LIST OFFSET field is not set to zero, then the command is terminated with CHECK CONDITION status with the Sense Key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB (i.e., 5/2400).

- RAO LIST OFFSET - The offset into the RAO list from which to begin returning data. This allows the RAO list to be retrieved in chunks. This value is required to be either zero or a value greater than or equal to eight and on a 4-byte boundary (i.e., 0 or 4+4n where $n \geq 1$). If this value is not a multiple of four, the command is terminated with CHECK CONDITION status with the Sense Key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB (i.e., 5/2400).
- ALLOCATION LENGTH is the maximum number of bytes to be returned.
- UDS_TYPE - If the UDS_LIMITS bit is set to 1b, then this is the type of User Data Segment descriptor (see 5.2.8) to use in calculating the maximum number of UDS supported. If the UDS_LIMITS bit is set to 0b, then this field is ignored.

5.2.29.1 RRAO parameter data

If the UDS_LIMITS bit is set to 1b (i.e., return Maximum number of User Data Segments supported for these settings), then the RRAO parameter data is defined in 5.2.29.1.1. If the UDS_LIMITS bit is set to 0b (i.e., return Recommended Access Order list), then the RRAO parameter data is defined in 5.2.29.1.2.

5.2.29.1.1 UDS Limits page

The parameter data received when the UDS_LIMITS bit of the RRAO command is set to 1b (i.e., return UDS Limits page). This returns values supported for the requested setting of UDS_TYPE and is defined in table 102.

Table 102 — UDS Limits page

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	(MSB)	MAXIMUM UDS SUPPORTED						
1								(LSB)
2	(MSB)	MAXIMUM UDS SIZE						
3								(LSB)

The following parameters apply:

Byte	**Description**
0-1	MAXIMUM UDS SUPPORTED - This field specifies the maximum number of User Data Segments that the drive is capable of returning for the specified UDS TYPE.
2-3	MAXIMUM UDS SIZE - This field specifies the maximum size of the UDS descriptor (see 5.2.29.1.3) for the specified UDS TYPE.

5.2.29.1.2 RAO list

The parameter data received when the UDS_LIMITS bit of the RRAO command is set to 0b (i.e., return Recommended Access Order list) is defined in table 103.

Table 103 — RAO List

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Reserved					PROCESS		
1	Reserved					STATUS		
2	Reserved							
3								
4	(MSB)	ADDITIONAL DATA (n-7)						
…								
7								(LSB)
User Data Segment descriptors								
8	User Data Segment descriptor (first)							
x								
⋮								
y	User Data Segment descriptor (last)							
n								

The following parameters apply:

Byte **Description**

0

- **Bit** **Description**
- 7-3 Reserved
- 2-0 PROCESS - Requested process to generate the contents of the RAO list (see 4.27.2)

1

- **Bit** **Description**
- 7-3 Reserved
- 2-0 STATUS - Status of the RAO list

Value	**Description**
000b	The RAO list does not contain a valid list (e.g., the GRAO command invalidated the list)
001b	The RAO list contains a valid list generated using the process specified in the PROCESS field.
010b	The RAO list contains a list matching the order of the list passed in the GRAO command but has not processed the list due to an inability to do so at this time. The ESTIMATED LOCATE TIME TO UDS field and any Additional information descriptors are present but do not contain valid data.
others	Reserved.

2-3 Reserved

4-7 ADDITIONAL DATA - This field specifies the amount of data to follow. If the RAO list is invalid (i.e., the STATUS field is 000b), then this field is set to zero.

8-n User Data Segment descriptors - A list of User Data Segments in the order specified by the last successful GRAO command (see 5.2.8). User Data Segments are defined in 5.2.29.1.3.

5.2.29.1.3 User Data Segment descriptor

The User Data Segment descriptor is defined in table 104.

Table 104 — User Data Segment descriptor

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	(MSB)	UDS DESCRIPTOR LENGTH (n-1)						
1								(LSB)
2	Reserved							
3	Reserved							
4	ESTIMATED LOCATE TIME TO UDS							
5		UDS NAME						
…								
14								
15	PARTITION NUMBER							
16	(MSB)	BEGINNING LOGICAL OBJECT IDENTIFIER						
…								
23								(LSB)
24	(MSB)	ENDING LOGICAL OBJECT IDENTIFIER						
…								
31								(LSB)
Additional information descriptors								
32	Additional information descriptor (beginning)							
x								
⋮								
y	Additional information Geometry descriptor (ending)							
n								

The following parameters apply:

Byte	**Description**
0-1	UDS DESCRIPTOR LENGTH - length of data to follow.
2	Reserved
3	Reserved
4	ESTIMATED LOCATE TIME TO UDS - An estimation of the nominal time in seconds required to change position:

a) if this is the first UDS in the parameter list, then from the current logical position to the BEGINNING LOGICAL OBJECT IDENTIFIER of this UDS; or

b) if this is not the first UDS in the parameter list, then from the ENDING LOGICAL OBJECT IDENTIFIER from the previous UDS descriptor to the BEGINNING LOGICAL OBJECT IDENTIFIER of this UDS.

This value is not guaranteed and may be affected by:

a) read-ahead operations;
b) application client behaviors;
c) error recovery procedures; and

d) other conditions.

If the time to position to this UDS is unknown, cannot be estimated, the STATUS field of the RAO list is set to 010b, or if the drive does not support ESTIMATED LOCATE TIME TO UDS then this field is set to zero.

If the time to position to this UDS is more than FEh seconds, then this field shall be set to FFh.

5-14 UDS NAME - Name given to this User Data Segment by the application

15 PARTITION NUMBER - Number of the partition in which this User Data Segment is located.

16-23 BEGINNING LOGICAL OBJECT IDENTIFIER - LOGICAL OBJECT IDENTIFIER OF THE BEGINNING LOGICAL OBJECT OF THE USER DATA SEGMENT.

24-31 ENDING LOGICAL OBJECT IDENTIFIER - LOGICAL OBJECT IDENTIFIER OF THE ENDING LOGICAL OBJECT OF THE USER DATA SEGMENT.

32-n Additional information descriptors - Descriptors that describe additional information. Additional information descriptors are returned if the GRAO process selected requested that those descriptors be generated as part of the RAO list generation.

At the time this document was published, there is one geometry descriptor (see 5.2.29.1.3.1) that describes the geometry at the beginning logical object identifier and one geometry descriptor that describes the geometry at the ending logical object identifier.

5.2.29.1.3.1 Additional information descriptor

Additional information descriptors are used to provide additional information related to the UDS, such as its' geometry (see 5.2.29.1.3.1). The additional information descriptor is defined in table 105.

Table 105 — Additional information descriptor

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	(MSB)	ADDITIONAL INFORMATION DESCRIPTOR LENGTH						
1								(LSB)
2	ADDITIONAL INFORMATION TYPE							
3	DIR	Reserved					BAND	
4	(MSB)	RPOS						
…								
7								(LSB)
8	WRAP							
9								

The following parameters apply:

Byte Description

0-1 ADDITIONAL INFORMATION DESCRIPTOR LENGTH - Number of additional bytes that follow in the additional information descriptor. Applications should check this length and if it is longer than expected, then use the bytes and fields understood and skip any extra bytes that are not understood. Growth may occur in the descriptor if new items are requested or found useful.

2 ADDITIONAL INFORMATION TYPE - The type of additional information descriptor.

Value Description

00h Begin Point Geometry - Describes the geometry of the logical object pointed to by the BEGINNING LOGICAL OBJECT IDENTIFIER field.

01h End Point Geometry - Describes the geometry of the logical object pointed to by the ENDING LOGICAL OBJECT IDENTIFIER field.

NOTE 43 - No assumption should be made about the geometry of the data between the Begin Point Geometry and the End Point Geometry since the order of wrap traversal may be non-sequential.

others Reserved for future use.

3 Flags

Bit	**Description**
0-1	BAND - The band in which the logical object is recorded. If the STATUS field of the RAO list (see 5.2.29.1.2) is set to 010b, then this field is set to zero.
2-6	Reserved
7	DIR - The direction of medium travel for the logical object. If the STATUS field of the RAO list (see 5.2.29.1.2) is set to 010b, then this field is set to zero.

Value	**Description**
0b	Forward direction (i.e,. from BOT to EOT)
1b	Backward direction (i.e., from EOT to BOT)

4-7 RPOS - Estimated relative longitudinal position (in mm) of the logical object pointed to by the LOGICAL OBJECT IDENTIFIER RELATIVE TO THE EARLIEST PHYSICAL RECORDING START (I.E. LP3). THIS IS THE DISTANCE BETWEEN LP3 AND THE LOGICAL BLOCK EXPRESSED IN MILLIMETERS. If the STATUS field of the RAO list (see 5.2.29.1.2) is set to 010b, then this field is set to zero.

8-9 WRAP - Physical wrap in which the LOGICAL OBJECT IDENTIFIER is located. If the STATUS field of the RAO list (see 5.2.29.1.2) is set to 010b, then this field is set to zero.

5.2.30 RECOVER BUFFERED DATA - 14h

The Recover Buffered Data command is defined in SSC (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause specifies the specific implementation.

Table 106 — Recover Buffered Data CDB

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Operation code (14)							
1	Obsolete (LUN)			Reserved			SILI	Fixed
2	MSB	Transfer Length						
4								LSB
5	Control Byte (see 5.1.3)							

The following parameters apply:

- SILI (Suppress Incorrect Length Indicator) (per SCSI-2 standard) (see 4.10)
- Fixed (see 4.8)
- Transfer Length (see 4.8)

For more information, see “Data Transfer, Block Limits, and Fixed Block Option” on page 38.

NOTE 44 - When no permanent write error condition has occurred, this command performs a synchronize and attempts to write all buffered data to media. No data will be returned.

NOTE 45 - Having begun to recover data through the use of the Recover Buffer Data command, the initiator should not change the RBO (Recover Buffer Order) field in mode page 10h, until all the data in the buffer is read or until the device has received and successfully executed a Locate command, a Load Unload command, or a Rewind command. If the initiator attempts to change the RBO field while the device still has data in the buffers, the device rejects the command with associated sense data of with 5/2602 (Illegal Request, Parameter Value Invalid).

5.2.31 RELEASE UNIT - 17h

The Release Unit command is defined in SPC-2 (see http://www.t10.org/cgi-bin/ac.pl?t=f&f=spc2r20.pdf). This clause specifies the specific implementation.

Table 107 — Release Unit CDB

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Operation code (17)							
1	Obsolete (LUN)			3rdPty	Third Party Device ID			Reserved
2	Reserved							
4								
5	Control Byte (see 5.1.3)							

The following parameters apply:

- 3rdPty (Third Party): 0b
- Third Party Device ID: 000b

5.2.32 REPORT DENSITY SUPPORT - 44h

The Report Density Support command is defined in SSC (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause specifies the specific implementation.

Table 108 — Report Density Support CDB

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Operation code (44)							
1	Reserved							MEDIA
2	Reserved							
6								
7	MSB	Allocation Length						
8								LSB
9	Control Byte (see 5.1.3)							

The following parameters apply:

- MEDIA:

Value	**Description**
0b	report all supported densities for all supported medium types
1b	report all supported densities for the current medium in the drive. If the device is not ready, the drive will return 2/0400 (Not Ready, Not Ready Cause Not Reportable).

- Allocation Length is the maximum number of bytes to be transferred.

5.2.32.1 Report Density Support data format

5.2.32.1.1 Density descriptor overview

Density descriptors contain information that may be returned by the drive. While the information is reported in a manner useful to applications, SSC-4 specifies fields that have been found to not be useful by applications in selecting which density to use. Instead applications have been found to ignore the Bits Per MM field and the Tracks field. This device does not change the values returned in these fields by the format type media type combinations and instead returns the largest value for the set of supported format type media type combinations. The Capacity field is used by applications and does report values based on the format type - media type combinations. The Density code fields, the Assigning Organization field, the Density Name field, and the Description field are constant per density and are correctly reported.

This device may return different combinations of the descriptors in the Report Density Support data depending on drive generation, medium loaded, setting of the media bit, and settings of various mode parameters related to densities.

The Report Density Support data format is shown in table 109.

Table 109 — REPORT DENSITY SUPPORT data format

Byte	Bit 7	6	5	4	3	2	1	0
0	(MSB)	AVAILABLE DENSITY SUPPORT LENGTH						
1								(LSB)
2	Reserved							
3	Reserved							
	Density support data block descriptors							
4	Density support data block descriptor [first]							
	⋮							
	Density support data block descriptor [last]							
n								

The Density support data block descriptor format is shown in table 110.

Table 110 — Density support data block descriptor format

Byte	Bit 7	6	5	4	3	2	1	0
0	PRIMARY DENSITY CODE							
1	SECONDARY DENSITY CODE							
2	WRTOK	DUP	DEFLT	Reserved				DLV
3	(MSB)	DESCRIPTOR LENGTH						
4								(LSB)
5	(MSB)	BITS PER MM						
7								(LSB)
8	(MSB)	MEDIA WIDTH						
9								(LSB)
10	(MSB)	TRACKS						
11								(LSB)
12	(MSB)	CAPACITY						
15								(LSB)
16	ASSIGNING ORGANIZATION							
23								
24	DENSITY NAME							
31								
32	DESCRIPTION							
51								

5.2.32.1.2 Density information

The density information is shown in table 111 and table 112.

Table 111 — Density information for primary density codes 51h to 53h

FIELD	PRIMARY DENSITY CODE		
	51h [1]	**52h [2]**	**53h [3]**
PRIMARY DENSITY CODE	51h	52h	53h
SECONDARY DENSITY CODE	51h (cleartext) 71h (encrypted)	52h (cleartext) 72h (encrypted)	53h (cleartext) 73h (encrypted)
WRTOK	0b The device cannot write this format 1b The device can write this format		
DUP	0	0	0
DEFLT	0b This density is not currently selected for use on a write from BOP 1b This density is currently selected for use on a write from BOP		
DLV	0	0	0
DESCRIPTOR LENGTH	0000h	0000h	0000h
BITS PER MM	002E18h	002E18h	00348Ch
MEDIA WIDTH (MM)	000Dh	000Dh	000Dh
TRACKS	0200h	0380h	0480h
CAPACITY (10^6)	0004_93E0h [0,JA/JW] 0000_EA60h [JJ/JR]	000A_AE60h [0,JB/JX] 0007_A120h [JA/JW] 0001_86A0h [JJ/JR]	000F_4240h [0,JB/JX] 0009_C400h [JA/JW] 0001_F400h [JJ/JR]
ASSIGNING ORGANIZATION (ASCII)	‘IBM ‘	‘IBM ‘	‘IBM ‘
DENSITY NAME (ASCII)	‘3592A1 ‘	‘3592A2 ‘	‘3592A3 ‘
DESCRIPTION (ASCII)	‘ ‘ (20 ASCII spaces)	‘ ‘ (20 ASCII spaces)	‘ ‘ (20 ASCII spaces)

[0] VALUE WHEN MEDIA = 0B

[1] Density descriptor (51h/71h) may be returned on J1A, E05, and E06

[2] Density descriptor (52h/72h) may be returned on E05, E06 and E07

[3] Density descriptor (53h/73h) may be returned on E06 and E07

[JA/JW] VALUE WHEN MEDIA =1B AND Standard Cartridge (JA) or Standard WORM Cartridge (JW) is loaded

[JJ/JR] VALUE WHEN MEDIA =1B AND Economy Cartridge (JJ) or Economy WORM Cartridge (JR) is loaded

[JB/JX] VALUE WHEN MEDIA =1B AND Extended Cartridge (JB) or Extended WORM Cartridge (JX) is loaded

Table 112 — Density information for primary density codes 54h to 57h

Field	PRIMARY DENSITY CODE			
	54h [1]	55h [2]	56h [3]	57h [4]
PRIMARY DENSITY CODE	54h	55h	56h	57h
SECONDARY DENSITY CODE	54h (cleartext@LBA0) 74h (encrypted@LBA0)	55h (cleartext@LBA0) 75h (encrypted@LBA0)	56h (cleartext@LBA0) 76h (encrypted@LBA0)	57h (cleartext@LBA0) 77h (encrypted@LBA0)
WRTOK	0b The device cannot write this format 1b The device can write this format			
DUP	0	0	0	0
DEFLT	0b This density is not currently selected for use on a write from BOP 1b This density is currently selected for use on a write from BOP			
DLV	0	0	0	0
DESCRIPTOR LENGTH	0000h	0000h	0000h	0000h
BITS PER MM	004CE6h	0050BEh	0050BEh	00 555Ah [JE/JV/JM]
MEDIA WIDTH (MM)	000Dh	000Dh	000Dh	000Dh
TRACKS	0A00h [JC/JY, JK] 0700h [JB/JX]	1400h [JD/JZ, JL] 1200h [JC/JY, K]	1E00h	2200h [JE/JV, JM]
CAPACITY (10^6)	003D_0900h [0,JC/JY] 0007_A120h [JK] 0018_6A00h [JB/JX]	0098 9680h [0,JD/JZ] 001E 8480h [JL] 006A CFC0h [JC/JY] 000D BBA0h JK	00E4 E1C0h [0,JD/JZ] 002D C6C0h [JL]	0131 2D00h [0,JE/JV] 004C 4B40h [JM]
ASSIGNING ORGANIZATION (ASCII)	‘IBM ‘	‘IBM ‘	‘IBM ‘	‘IBM ‘
DENSITY NAME (ASCII)	‘3592A4 ‘	‘3592A5 ‘	‘3592B5 ‘	‘3592A6 ‘
DESCRIPTION (ASCII)	‘ ‘ (20 ASCII spaces)	‘ ‘ (20 ASCII spaces)	‘ ‘ (20 ASCII spaces)	‘ ‘ (20 ASCII spaces)

[0] Value when MEDIA = 0b

[1] Density descriptor (54h/74h) may be returned on products E07, E08 and 55F

[2] Density descriptor (55h/75h) may be returned on products E08, 55F and 60F

[3] Density descriptor (56h/76h) may be returned on products 55F and 60F

[4] Density descriptor (57h/77h) may be returned on product 60F

[JB/JX] Value when MEDIA =1b and Extended Cartridge (JB) or Extended WORM Cartridge (JX) is loaded

[JC/JY] Value when MEDIA =1b and Advanced Type C Cartridge (JC) or Advanced Type C WORM Cartridge (JY) is loaded

[JK] Value when MEDIA =1b and Advanced Type C Economy Cartridge (JK) is loaded

[JL] Value when MEDIA =1b and Advanced Type D Economy Cartridge (JL) is loaded

[JD/JZ] Value when MEDIA =1b and Advanced Type D (JD or JZ) Cartridge is loaded

[JM] Value when MEDIA =1b and Advanced Type E Economy Cartridge (JM) is loaded

[JE/JV] Value when MEDIA =1b and Advanced Type E (JE or JM) Cartridge is loaded

5.2.33 REPORT LUNS - A0h

The Report LUNs command is defined in SPC (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause specifies the specific implementation.

Table 113 — Report LUNs CDB

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Operation Code (A0h)							
1	Reserved							
5								
6	MSB	Allocation Length						
9								LSB
10	Reserved							
11	Control Byte (see 5.1.3)							

The following parameters apply:

- Allocation Length is the maximum number of bytes to be transferred.

5.2.33.1 Report LUNs data format

Byte	Description
0-3	LUN List Length:

Value	Description
00000008h	LUN 1 does not exist; bytes 16-23 are not returned
00000010h	LUN 1 exists; bytes 16-23 are returned.

Byte	Description
4-7	Reserved
8-15	LUN 0: 0000000000000000h
16-23	LUN 1: 0001000000000000h (returned only if LUN 1 exists)

5.2.34 REPORT SUPPORTED OPERATION CODES - A3h (beginning with E07)

5.2.34.1 REPORT SUPPORTED OPERATION CODES command introduction

E07 and later support the REPORT SUPPORTED OPERATION CODES command. The REPORT SUPPORTED OPERATION CODES command (see table 114) requests information on commands the addressed logical unit supports. An application client may request a list of all operation codes and service actions supported by the logical unit or the command support data for a specific command.

The REPORT SUPPORTED OPERATION CODES command is a service action of the MAINTENANCE IN command.

Table 114 — REPORT SUPPORTED OPERATION CODES command

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (A3h)							
1	Reserved			SERVICE ACTION (0Ch)				
2	RCTD	Reserved				REPORTING OPTIONS		
3	REQUESTED OPERATION CODE							
4	(MSB)	REQUESTED SERVICE ACTION						
5								(LSB)
6	(MSB)	ALLOCATION LENGTH						
9								(LSB)
10	Reserved							
11	CONTROL							

A return command timeouts descriptor (RCTD) bit set to one specifies that the command timeouts descriptor (see 5.2.34.4) shall be included in each command descriptor (see 5.2.34.2) that is returned or in the one_command parameter data (see 5.2.34.3) that is returned. A RCTD bit set to zero specifies that the command timeouts descriptor shall not be included in any parameter data returned.

The REPORTING OPTIONS field (see table 115) specifies the information to be returned in the parameter data.

Table 115 — REPORT SUPPORTED OPERATION CODES REPORTING OPTIONS field

Code	Description	Parameter Data Reference
000b	A list of all operation codes and service actions supported by the logical unit shall be returned in the all_commands parameter data format. The REQUESTED OPERATION CODE CDB field and REQUESTED SERVICE ACTION CDB field shall be ignored.	5.2.34.2
001b	The command support data for the operation code specified in the REQUESTED OPERATION CODE field shall be returned in the one_command parameter data format. The REQUESTED SERVICE ACTION CDB field shall be ignored. If the REQUESTED OPERATION CODE field specifies an operation code that has service actions, then the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST, and the additional sense code set to INVALID FIELD IN CDB.	5.2.34.3
010b	The command support data for the operation code and service action specified in the REQUESTED OPERATION CODE CDB field and REQUESTED SERVICE ACTION CDB field shall be returned in the one_command parameter data format. If the REQUESTED OPERATION CODE CDB field specifies an operation code that does not have service actions, then the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST, and the additional sense code set to INVALID FIELD IN CDB.	5.2.34.3
011b to 111b	Reserved	

The REQUESTED OPERATION CODE field specifies the operation code of the command to be returned in the one_command parameter data format (see 5.2.34.3).

The REQUESTED SERVICE ACTION field specifies the service action of the command to be returned in the one_command parameter data format.

The ALLOCATION LENGTH field specifies the maximum number of bytes or blocks that an application client has allocated in the Data-In Buffer.

5.2.34.2 All_commands parameter data format

The REPORT SUPPORTED OPERATION CODES all_commands parameter data format (see table 116) begins with a four-byte header that contains the length in bytes of the parameter data followed by a list of supported commands. Each command descriptor contains information about a single supported command CDB (i.e., one operation code and service action combination, or one non-service-action operation code). The list of command descriptors shall contain all commands supported by the logical unit.

Table 116 — All_commands parameter data

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)							
	COMMAND DATA LENGTH (n-3)							
3								(LSB)
	Command descriptors							
4	Command descriptor 0 (see table 117)							
	⋮							
n	Command descriptor x (see table 117)							

The COMMAND DATA LENGTH field indicates the length in bytes of the command descriptor list.

Each command descriptor (see table 117) contains information about a single supported command CDB.

Table 117 — Command descriptor format

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE							
1	Reserved							
2	(MSB)	SERVICE ACTION						
3								(LSB)
4	Reserved							
5	Reserved						CTDP	SERVACTV
6	(MSB)	CDB LENGTH						
7								(LSB)
8	Command timeouts descriptor, if any (see 5.2.34.4)							
19								

The OPERATION CODE field contains the operation code of a command supported by the logical unit.

The SERVICE ACTION field contains a supported service action of the supported operation code indicated by the OPERATION CODE field. If the operation code indicated in the OPERATION CODE field does not have a service actions, the SERVICE ACTION field shall be set to 00h.

A command timeouts descriptor present (CTDP) bit set to one indicates that the command timeouts descriptor (see 5.2.34.4) is included in this command descriptor. A CTDP bit set to zero indicates that the command timeouts descriptor is not included in this command descriptor.

A service action valid (SERVACTV) bit set to zero indicates the operation code indicated by the OPERATION CODE field does not have service actions and the SERVICE ACTION field contents are reserved. A SERVACTV bit set to one indicates the operation code indicated by the OPERATION CODE field has service actions and the contents of the SERVICE ACTION field are valid.

The CDB LENGTH field contains the length of the command CDB in bytes for the operation code indicated in the OPERATION CODE field, and if the SERVACTV bit is set to the service action indicated by the SERVICE ACTION field.

If the RCTD bit is set to one in the REPORT SUPPORTED OPERATION CODES CDB (see 5.2.34.1), the command timeouts descriptor (see table 117 in 5.2.34.4) shall be included. If the RCTD bit is set to zero, the command timeouts descriptor shall not be included.

5.2.34.3 One_command parameter data format

The REPORT SUPPORTED OPERATION CODES one_command parameter data format (see table 118) contains information about the CDB and a usage map for bits in the CDB for the command specified by the REPORTING OPTIONS, REQUESTED OPERATION CODE, and REQUESTED SERVICE ACTION fields in the REPORT SUPPORTED OPERATION CODES CDB.

Table 118 — One_command parameter data

Bit Byte	7	6	5	4	3	2	1	0
0	Reserved							
1	CTDP	Reserved				SUPPORT		
2	(MSB)	CDB SIZE (n-3)						
3								(LSB)
4	CDB USAGE DATA							
n								
n+1	Command timeouts descriptor, if any (see 5.2.34.4)							
n+12								

A command timeouts descriptor present (CTDP) bit set to one indicates that the command timeouts descriptor (see 5.2.34.4) is included in the parameter data. A CTDP bit set to zero indicates that the command timeouts descriptor is not included in the parameter data.

The SUPPORT field is defined in table 119.

Table 119 — SUPPORT values

Support	Description
000b	Data about the requested SCSI command is not currently available. All data after byte 1 is not valid. A subsequent request for command support data may be successful.
001b	The device server does not support the requested command. All data after byte 1 is undefined.
010b	Reserved
011b	The device server supports the requested command in conformance with a SCSI standard. The parameter data format conforms to the definition in table 118.
100b	Reserved
101b	The device server supports the requested command in a vendor specific manner. The parameter data format conforms to the definition in table 118.
110b to 111b	Reserved

The CDB SIZE field contains the size of the CDB USAGE DATA field in the parameter data, and the number of bytes in the CDB for command being queried (i.e., the command specified by the REPORTING OPTIONS, REQUESTED OPERATION CODE, and REQUESTED SERVICE ACTION fields in the REPORT SUPPORTED OPERATION CODES CDB).

The CDB USAGE DATA field contains information about the CDB for the command being queried. The first byte of the CDB USAGE DATA field shall contain the operation code for the command being queried. If the command being queried contains a service action, then that service action code shall be placed in the CDB USAGE DATA field in the same location as the SERVICE ACTION field of the command CDB. All other bytes of the CDB USAGE DATA field shall contain a usage map for bits in the CDB for the command being queried.

The bits in the usage map shall have a one-for-one correspondence to the CDB for the command being queried. If the device server evaluates a bit in the CDB for the command being queried, the usage map shall contain a one in the corresponding bit position. If any bit representing part of a field is returned as one, all bits for the field shall be returned as one. If the device server ignores or treats as reserved a bit in the CDB for the command being queried, the usage map shall contain a zero in the corresponding bit position. The usage map bits for a given CDB field all shall have the same value.

For example, the CDB usage bit map for the REPORT SUPPORTED OPERATION CODES command is: A3h, 0Ch, 87h, FFh, FFh, FFh, FFh, FFh, FFh, FFh, 00h, 07h. This example assumes that the logical unit only supports the low-order three bits of the CDB CONTROL byte. The first byte contains the operation code, and the second byte contains three reserved bits and the service action. The remaining bytes contain the usage map.

If the RCTD bit is set to one in the REPORT SUPPORTED OPERATION CODES CDB (see 5.2.34.1), the command timeouts descriptor (see table 120 in 5.2.34.4) shall be included. If the RCTD bit is set to zero, the command timeouts descriptor shall not be included.

5.2.34.4 Command timeouts descriptor

5.2.34.4.1 Overview

The command timeouts descriptor (see table 120) returns timeout information for commands supported by the logical unit based on the time from the start of processing for the command to its reported completion.

Values returned in the command timeouts descriptor do not include times that are outside the control of the device server (e.g., prior commands with the IMMED bit set to one in the CDB, concurrent commands from the same or different I_T nexuses, manual unloads, power-on self tests, prior aborted commands, commands that force cache synchronization, delays in the service delivery subsystem).

For commands that cause a change in power condition, values returned in the command timeouts descriptor do not include the power condition transition time (e.g., the time to thread the media).

Values returned in the command timeouts descriptor should not be used to compare products.

Table 120 — Command timeouts descriptor format

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)	DESCRIPTOR LENGTH (0Ah)						
1								(LSB)
2	Reserved							
3	COMMAND SPECIFIC							
4	(MSB)	NOMINAL COMMAND PROCESSING TIMEOUT						
7								(LSB)
8	(MSB)	RECOMMENDED COMMAND TIMEOUT						
11								(LSB)

The DESCRIPTOR LENGTH field indicates the number of bytes that follow in the command timeouts descriptor.

The COMMAND SPECIFIC field contains timeout information (see table 121) that is specific to one or more commands

Table 121 — Command timeouts descriptor COMMAND SPECIFIC field usage

Command	Reference
WRITE BUFFER	5.2.34.4.2

A non-zero value in the NOMINAL COMMAND PROCESSING TIMEOUT field indicates the minimum amount of time in seconds the application client should wait prior to querying for the progress of the command identified by the parameter data that contains this command timeouts descriptor. A value of zero in the NOMINAL COMMAND PROCESSING TIMEOUT field indicates that no timeout is indicated.

NOTE 46 - The value contained in the NOMINAL COMMAND PROCESSING TIMEOUT field may include time required for typical device error recovery procedures expected to occur on a regular basis.

A non-zero value in the RECOMMENDED COMMAND TIMEOUT field specifies the recommended time in seconds the application client should wait prior to timing out the command identified by the parameter data that contains this command timeouts descriptor. A value of zero in the RECOMMENDED COMMAND TIMEOUT field indicates that no time is indicated.

The device server should set the recommended command timeout to a value greater than or equal to the nominal command processing timeout.

Application clients should use the Report Supported Operation Codes command to discover command timoeut values. Table 122 lists the command timeout values at the time this document was published for the commands supported at the time the document was published. These values may have changed since publication.

Due to the sequential nature of tape devices, many host commands are serialized, and command timeouts consequently have an additive effect. Using reservations prevents this from causing application disruptions in a multi-initiator or SAN environment. Similar additive timeout effects may occur if the host is using command queuing (simple queuing).

NOTE 47 - The time-outs in the following table are based on the time from the start of processing of the command, to its reported completion. Since applications are generally concerned with the time from the command being issued, to its reported completion, it should be noted that this overall time may be affected by currently processing operations. If these currently processing operations are very long (e.g., long erase, verify with VTE, extended copy) there could be significant effects to the system that need to be accounted for. Some of these conditions include:

a) A prior command was issued with the Immediate bit set
b) Multiple concurrent commands with Simple queuing are processed
c) Multi-initiator configurations without reservations
d) Non-host operations, such as manual unloads, power-on self tests, etc.
e) Commands issued shortly after certain aborted commands
f) Commands which force flushes when unwritten write data is in the buffer

Table 122 — 3592 command timeout values (at publication) (part 1 of 4)

OpCode	Service Action	Command Name	Command Timeout (seconds)							
			E07		E08		55F		60F	
			Nom	Rec	Nom	Rec	Nom	Rec	Nom	Rec
00h	-----	TEST UNIT READY	1	30	1	30	1	30	1	30
01h	-----	REWIND [a] [b]	130	550	150	560	150	560	180	600
03h	-----	REQUEST SENSE	1	30	1	30	1	30	1	30
04h	-----	FORMAT MEDIUM	530	3000	530	3 100	530	3 100	540	3 060
05h	-----	READ BLOCK LIMITS	1	30	1	30	1	30	1	30
08h	-----	READ [d]	360	2 100	410	2 400	500	2 400	480	2 340
0Ah	-----	WRITE	360	1 200	410	1 500	500	1 500	480	1 440
0Fh	-----	READ REVERSE	360	2 100	410	2 400	500	2 400	480	2 340
10h	-----	WRITE FILEMARKS [a]	220	1 100	220	1 400	220	1 400	240	1 380
11h	-----	SPACE(6) [a] [b]	340	2 000	390	2 300	390	2 300	420	2 280
12h	-----	INQUIRY	1	30	1	30	1	30	1	30
13h	-----	VERIFY (VTE=1b or VBF=1b) [a] [b]	26 500	38 100	34 000	46 700	50 600	68 900	47 460	65 820
14h	-----	RECOVER BUFFERED DATA	30	60	30	60	30	60	30	60
15h	-----	MODE SELECT(6)	300	900	300	900	300	900	300	900

Table 122 — 3592 command timeout values (at publication) (part 2 of 4)

OpCode	Service Action	Command Name	Command Timeout (seconds)							
			E07		E08		55F		60F	
			Nom	Rec	Nom	Rec	Nom	Rec	Nom	Rec
16h	-----	RESERVE UNIT(6) [b]	300	900	300	900	300	900	300	900
17h	-----	RELEASE UNIT(6)	30	30	30	30	30	30	30	30
19h	-----	ERASE [a] [b]	26 500	36 900	34 000	45 800	50 600	68 000	47 460	64 860
1Ah	-----	MODE SENSE(6)	1	30	1	30	1	30	1	30
1Bh	-----	LOAD/UNLOAD [b]	400	720	400	900	400	900	400	900
1Ch	-----	RECEIVE DIAGNOSTIC RESULTS	1	30	1	30	1	30	1	30
1Dh	-----	SEND DIAGNOSTIC [c]	780	2 100	780	21 00	780	2 100	780	2 100
1Eh	-----	PREVENT ALLOW MEDIUM REMOVAL	1	30	1	30	1	30	1	30
2Bh	-----	LOCATE (10) [a] [b]	340	2 000	390	2 300	390	2 300	420	2 280
34h	0000h	READ POSITION — [SHORT FORM-- BLOCK ID]	1	30	1	30	1	30	1	30
34h	0006h	READ POSITION — [LONG FORM]	1	30	1	30	1	30	1	30
34h	0008h	READ POSITION — [EXTENDED FORM]	1	30	1	30	1	30	1	30
3Bh	-----	WRITE BUFFER	60	540	60	540	60	540	60	540
3Ch	-----	READ BUFFER	60	480	60	480	60	480	60	480
40h	-----	CHANGE DEFINITION	1	30	1	30	1	30	1	30
44h	-----	REPORT DENSITY SUPPORT	1	30	1	30	1	30	1	30
4Ch	-----	LOG SELECT	1	30	1	30	1	30	1	30
4Dh	-----	LOG SENSE	1	30	1	30	1	30	1	30
55h	-----	MODE SELECT(10)	300	900	300	900	300	900	300	900
56h	-----	RESERVE UNIT (10)	300	900	300	900	300	900	300	900
57h	-----	RELEASE UNIT (10)	30	30	30	30	30	30	30	30
5Ah	-----	MODE SENSE(10)	1	30	1	30	1	30	1	30
5Eh	0000h	PERSISTENT RESERVE IN — [READ KEYS]	1	30	1	30	1	30	1	30
5Eh	0001h	PERSISTENT RESERVE IN — [READ RESERVATION]	1	30	1	30	1	30	1	30
5Eh	0002h	PERSISTENT RESERVE IN — [REPORT CAPABILITIES]	1	30	1	30	1	30	1	30
5Eh	0003h	PERSISTENT RESERVE IN — [READ FULL STATUS]	1	30	1	30	1	30	1	30
5Fh	0000h	PERSISTENT RESERVE OUT — [REGISTER] [b]	300	900	300	900	300	900	300	900
5Fh	0001h	PERSISTENT RESERVE OUT — [RESERVE] [b]	300	900	300	900	300	900	300	900
5Fh	0002h	PERSISTENT RESERVE OUT — [RELEASE] [b]	300	900	300	900	300	900	300	900

Table 122 — 3592 command timeout values (at publication) (part 3 of 4)

OpCode	Service Action	Command Name	Command Timeout (seconds)							
			E07		E08		55F		60F	
			Nom	Rec	Nom	Rec	Nom	Rec	Nom	Rec
5Fh	0003h	PERSISTENT RESERVE OUT — [CLEAR] [b]	300	900	300	900	300	900	300	900
5Fh	0004h	PERSISTENT RESERVE OUT — [PREEMPT] [b]	300	900	300	900	300	900	300	900
5Fh	0005h	PERSISTENT RESERVE OUT — [PREEMPT AND ABORT] [b]	300	900	300	900	300	900	300	900
5Fh	0006h	PERSISTENT RESERVE OUT — [REGISTER AND IGNORE EXISTING KEY] [b]	300	900	300	900	300	900	300	900
5Fh	0007h	PERSISTENT RESERVE OUT — [REGISTER AND MOVE] [b]	300	900	300	900	300	900	300	900
82h	-----	ALLOW OVERWRITE	1	30	1	30	1	30	1	30
83h	0001h	XCOPY (i.e., EXTENDED COPY(LID4))	-----	-----	18 000	21 600	50 600	68 900	50 600	68 900
83h	001Ch	COPY OPERATION ABORT	-----	-----	1	1 080	1	1 080	1	1 080
84h	0005h	RECEIVE COPY STATUS(LID4)	-----	-----	1	30	1	30	1	30
8Ch	0000h	READ ATTRIBUTE — [ATTRIBUTE VALUES]	60	60	60	60	60	60	60	60
8Ch	0001h	READ ATTRIBUTE — [ATTRIBUTE LIST]	60	60	60	60	60	60	60	60
8Ch	0002h	READ ATTRIBUTE — [LOGICAL VOLUME LIST]	60	60	60	60	60	60	60	60
8Ch	0003h	READ ATTRIBUTE — [PARTITION LIST]	60	60	60	60	60	60	60	60
8Ch	0005h	READ ATTRIBUTE — [SUPPORTED ATTRIBUTES]	60	60	60	60	60	60	60	60
8Dh	-----	WRITE ATTRIBUTE	60	60	60	60	60	60	60	60
91h	-----	SPACE(16) [a] [b]	340	2 00	390	2 300	390	2 300	420	2 280
92h	-----	LOCATE (16) [a] [b]	340	2 000	390	2 300	390	2 300	420	2 280
A0h	-----	REPORT LUNS	1	30	1	30	1	30	1	30
A2h	-----	SECURITY PROTOCOL IN (SPIN)	300	300	300	300	300	300	300	300
A3h	000Ch	REPORT SUPPORTED OPERATION CODE	1	30	1	30	1	30	1	30
A3h	000Fh	REPORT TIMESTAMP	1	30	1	30	1	30	1	30
A3h	001Dh	RECEIVE RECOMMENDED ACCESS ORDER (RRAO)	1	30	1	30	1	30	1	30
A4h	000Fh	SET TIMESTAMP	1	30	1	30	1	30	1	30
A4h	001Dh	GENERATE RECOMMENDED ACCESS ORDER (GRAO)	5	30	5	30	5	30	5	30
B5h	-----	SECURITY PROTOCOL OUT (SPOUT)	300	300	300	300	300	300	300	300
C0h	-----	DISPLAY MESSAGE	1	30	1	30	1	30	1	30

Table 122 — 3592 command timeout values (at publication) (part 4 of 4)

OpCode	Service Action	Command Name	Command Timeout (seconds)							
			E07		E08		55F		60F	
			Nom	Rec	Nom	Rec	Nom	Rec	Nom	Rec
D1h	0000h	READ DYNAMIC RUNTIME ATTRIBUTE — [SUPPORTED ATTRIBUTES]	1	30	1	30	1	30	1	30
D1h	0010h	READ DYNAMIC RUNTIME ATTRIBUTE — [ATTRIBUTE VALUES FOR THIS I_T NEXUS]	1	30	1	30	1	30	1	30
D1h	0011h	READ DYNAMIC RUNTIME ATTRIBUTE — [ATTRIBUTE VALUES FOR ALL I_T NEXUSES]	1	30	1	30	1	30	1	30
D2h	-----	READ DYNAMIC RUNTIME ATTRIBUTE — [SUPPORTED ATTRIBUTES]	1	30	1	30	1	30	1	30
E3h	-----	STRING SEARCH [d]	350	1 600	410	1 700	410	1 700	410	1 700

Key:

----- **N/A**
Nom **Nominal**
Rec **Recommended**

Notes:

The Command Specific field is set to 00h except in the WRITE BUFFER command where it is set to 3Ch

[a] If the immediate bit is set, status is returned quickly. It is recommended that the application polls for completion with the Request Sense command. Other commands may not begin to process until completion of the immediate command, resulting in an additive effect of the command time-outs.

[b] A media synchronize operation may be performed during the processing of this command when unwritten buffered write data is present when the command is received.

[c] The time required for diagnostics can vary widely based on the particular diagnostic invoked.

[d] The time required can vary based on options and an optional time may be specified.

Table 123 lists the command timeout values for abnormal conditions or for options that are too long to make them useful with the values returned in table 122. This contains the commands supported at the time the document was published. These values may have changed since publication.

Table 123 — 3592 command timeout values (at publication) not returned to the command

OpCode	Service Action	Command Name	Command Timeout (seconds)							
			E07		E08		55F		60F	
			Nom	Rec	Nom	Rec	Nom	Rec	Nom	Rec
11h	-----	SPACE(6) (slow—e.g., lost directory) [a]	26 500	38 100	34 000	46 700	50 600	68 900	47 460	65 820
13h	-----	VERIFY (VTE=0b and VBF=0b) (e.g., single block verify)	300	1 300	300	1 600	300	1 600	300	1 560
2Bh	-----	LOCATE (10) (slow—e.g., lost directory) [a]	26 500	38 100	34 000	46 700	50 600	68 900	47 460	65 820
91h	-----	SPACE(16) (slow—e.g., lost directory) [a]	26 500	38 100	34 000	46 700	50 600	68 900	47 460	65 820
92h	-----	LOCATE (16) (slow—e.g., lost directory) [a]	26 500	38 100	34 000	46 700	50 600	68 900	47 460	65 820

Key:

-----	N/A
CS	Command Specific
Nom	Nominal
Rec	Recommended

Notes:

[a] Spacing and locate commands will automatically rebuild the portions of the High Resolution Tape Directory if required. This operation can be completed within the recommended time-out if the Cartridge Memory (CM) is valid. Otherwise linear processing of tape is needed and the time-out increases by up to 220 minutes.

Table 124 lists the command timeout values for device products prior to E07. The REPORT SUPPORTED OPERATION CODES command is not supported prior to E07.

Table 124 — 3592 command timeout values for Models prior to E07 (part 1 of 3)

Op Code	Service Action	Command Name	Command time-out in seconds
82h	-----	Allow Overwrite	30
40h	-----	Change Definition	30
C0h	-----	Display Message	30
19h	-----	Erase (long = 0)	900 [b, c]
	-----	Erase (long = 1)	14 100 [b, c]
04h	-----	Format Medium	1 500
12h	-----	Inquiry	30
1Bh	-----	Load (Unloaded to BOT)	720 [c]
		Loaded (EOT to BOT)	480 [c]
		Unload (BOT to Cartridge Eject)	600 [c]
		Unload (EOT to Cartridge Eject)	660 [c]
2Bh	-----	Locate	840 [a, b, c]
4Ch	-----	Log Select	30
4Dh	-----	Log Sense	30

Table 124 — 3592 command timeout values for Models prior to E07 (part 2 of 3)

Op Code	Service Action	Command Name	Command time-out in seconds
15h/55h	-----	Mode Select(6/10) (not page 23h)	30
		Mode Select(6/10) (page 23h)	900
1Ah/5Ah	-----	Mode Sense(6/10)	30
5Eh	-----	Persistent Reserve In	30
5Fh	-----	Persistent Reserve Out	900 [c]
1Eh	-----	Prevent Allow Medium Removal	30
08h	-----	Read	1 080
08h	-----	Read (String Search) (not J1A)	1 080+ [e]
8Ch	-----	Read Attribute	60
05h	-----	Read Block Limits	30
3Ch	-----	Read Buffer	300
D1h	-----	Read Dynamic Runtime Attributes	30
34h	-----	Read Position	30
0Fh	-----	Read Reverse	1 080
1Ch	-----	Receive Diagnostic Results	30
14h	-----	Recover Buffered Data	60
17h	-----	Release Unit (6)	30
44h	-----	Report Density Support	30
A0h	-----	Report LUNs	30
A3h	000Ch	Report Supported Operation Codes	30
A3h	000Fh	Report Timestamp	30
03h	-----	Request Sense	30
16h	-----	Reserve Unit (6)	900 [c]
01h	-----	Rewind	480 [b, c]
A2h	-----	Security Protocol In (SPIN)	300
B5h	-----	Security Protocol Out (SPOUT)	300
1Dh	-----	Send Diagnostic	2 100 [d]
A4h	000Fh	Set Timestamp	30
11h/91h	-----	Space(6/16)	840 [a, b, c]
11h/91h	-----	Space(6/16) (String Search)	1 080+ [a, b, c, e]
E3h	-----	String Search	1 080+ [e]
00h	-----	Test Unit Ready	30
13h	-----	Verify (VTE=0 and VTF=0)	1 080
		Verify (VTE=1 or VTF=1)	21 600 [b, c]
0Ah	-----	Write	1 080
8Dh	-----	Write Attribute	60
3Bh	-----	Write Buffer	480
D2h	-----	Write Dynamic Runtime Attributes	30
10h	-----	Write Filemarks	1 080 [b]

Table 124 — 3592 command timeout values for Models prior to E07 (part 3 of 3)

Op Code	Service Action	Command Name	Command time-out in seconds

Notes:

[a] Spacing and locate commands will automatically rebuild the portions of the High Resolution Tape Directory if required. This operation can be completed within the recommended time-out if the Cartridge Memory (CM) is valid. Otherwise linear processing of tape is needed and the time-out increases by up to 220 minutes.

[b] When the immediate bit is set, the command will return quickly. It is recommended that the host issues Request Sense commands to poll for completion. Other commands may not begin execution until completion of the immediate command, resulting in an additive effect of the command time-outs.

[c] Depending on the current state of the drive and the command parameters, a media synchronize operation may be performed during the execution of this command. This primarily relates to conditions where unwritten buffered write data is present when the command is issued.

[d] The time required for diagnostics can vary widely based on the particular diagnostic invoked.

[e] The time required can vary based on options and an optional time may be specified.

[f]

5.2.34.4.2 WRITE BUFFER command timeouts descriptor COMMAND SPECIFIC field usage

For the WRITE BUFFER command, the COMMAND SPECIFIC field usage is reserved for all modes except the following:

a) Download microcode mode (04h);
b) Download microcode and save mode (05h);
c) Download microcode with offsets mode (06h);
d) Download microcode with offsets and save mode (07h);
e) Download microcode with offsets and defer activation mode (0Eh) only if the microcode is activated by an event other than an activate deferred microcode mode; and
f) Activate deferred microcode mode (0Fh).

If the command timeouts descriptor describes one of the WRITE BUFFER modes listed in this subclause, then the COMMAND SPECIFIC field indicates the maximum time, in one second increments, that access to the SCSI device is limited or not possible through any SCSI ports associated with a logical unit that processes a WRITE BUFFER command that specifies one of the named modes. A value of zero in the COMMAND SPECIFIC field indicates that the no maximum time is indicated.

5.2.35 REPORT TIMESTAMP - A3h[0Fh]

The REPORT TIMESTAMP command (see table 125) is defined in SPC (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause decribes the specific implementation.

Table 125 — REPORT TIMESTAMP CDB

Byte	Bit							
	7	6	5	4	3	2	1	0
0	OPERATION CODE (A3h)							
1	Reserved			SERVICE ACTION (0Fh)				
2 5	Reserved							
6 9	(MSB)	ALLOCATION LENGTH						(LSB)
10	Reserved							
11	Control Byte (see 5.1.3)							

The following parameters apply:

- SERVICE ACTION: 0Fh
- ALLOCATION LENGTH: The number of bytes that have been allocated for the returned parameter data.

5.2.35.1 REPORT TIMESTAMP parameter data

The format of the parameter data returned by the REPORT TIMESTAMP command is shown in table 126.

Table 126 — Timestamp Descriptor

Byte	Bit							
	7	6	5	4	3	2	1	0
0 1	(MSB)	TIMESTAMP PARAMETER DATA LENGTH (0Ah)						(LSB)
2	Reserved					TIMESTAMP ORIGIN		
3	Reserved							
4 9	(MSB)	TIMESTAMP						(LSB)
10 11	Reserved							

The parameters are defined as follows:

Byte	Description
0-1	TIMESTAMP PARAMETER DATA LENGTH: 0Ah
2	

Bit	Description
7-3	Reserved
2-0	TIMESTAMP ORIGIN: Device Clocks (see 4.25 on page 83) defines the TIMESTAMP ORIGIN

3	Reserved
4-9	TIMESTAMP: Device Clocks (see 4.25 on page 83) defines the timestamp.
10-11	Reserved

5.2.36 REQUEST SENSE - 03h

The Request Sense command is defined in SPC (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause specifies the specific implementation.

Table 127 — Request Sense CDB

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Operation code (03h)							
1	Obsolete (LUN)			Reserved				
2	Reserved							
3								
4	Allocation Length							
5	Control Byte (see 5.1.3)							

The following parameters apply:

- Allocation Length: The maximum number of bytes to be transferred.

  This device generates up to 96 bytes of sense data. If the allocation length specified is less than the generated sense data length, then the allocated amount is transferred, the remaining sense data is lost, and no error is reported. If the allocated length specified is greater, then the entire sense data is transferred and no error is reported.

  In read ILI conditions only 18 bytes of sense data may be generated. While processing sense data, the host should use byte 7 to determine the amount of sense data generated to insure that only valid transferred fields are examined.

NOTE 48 - Since this device uses autosense, 96 bytes of unsolicited sense is always generated using this command. Autosense may generate and return other lengths (e.g., 18 bytes).

NOTE 49 - More than 96 bytes of data may be returned (e.g., EXTENDED COPY (LID4) (i.e., XCOPY) – 83h[01h] (see 5.2.5 on page 110)).

Clause 6.8 describes the different formats of sense data.

5.2.37 RESERVE UNIT - 16h

The Reserve Unit command is defined in SPC-2 (see http://www.t10.org/cgi-bin/ac.pl?t=f&f=spc2r20.pdf). This clause specifies the specific implementation.

Table 128 — Reserve Unit CDB

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Operation code (16h)							
1	Obsolete (LUN)			3rdPty	Third Party Device ID			Reserved
2	Reserved							
4								
5	Control Byte (see 5.1.3)							

The following parameters apply:

- 3rdPty (Third Party): 0b
- Third Party Device ID: 000b

NOTE 50 - Reserves are honored across initiator as well as port boundaries. For additional information, see “Multiple Port Behavior” on page 263.

5.2.38 REWIND - 01h

The Rewind command is defined in SSC (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause specifies the specific implementation.

Table 129 — Rewind CDB

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Operation code (01h)							
1	Obsolete (LUN)			Reserved				Immed
2	Reserved							
4								
5	Control Byte (see 5.1.3)							

The following parameters apply:

- Immed (Immediate):

Value	**Description**
0b	present status when command is completed.
1b	present status when all buffered data is successfully written to the media.

5.2.39 SECURITY PROTOCOL IN (SPIN) - A2h

The Security Protocol In command is defined in SPC-4 (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause specifies the specific implementation.

The SECURITY PROTOCOL IN command (see table 130) is:

7) supported in encryption capable drives that are configured for AME-T10 mode (see 4.20.2); and
8) is used to retrieve security protocol information (see 5.2.40) or the results of one or more SECURITY PROTOCOL OUT commands (see 5.2.40).

Table 130 — Security Protocol In - A2h CDB

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Operation code (A2h)							
1	Security Protocol							
2	Security Protocol Specific							
3								
4	INC_512 (0b)	Reserved						
5	Reserved							
6	(MSB)	Allocation Length						
9								(LSB)
10	Reserved							
11	Control Byte (see 5.1.3)							

The following parameters apply:

- Security Protocol

Table 131 — Security Protocol Definitions

Code	Description	Reference
00h	Security protocol information	see *6.7.1*
20h	Tape Data Encryption	see *6.7.2*
all others	Reserved	

- Security Protocol Specific - The contents depend on the protocol specified by the Security Protocol field (see table 131).
- Allocation Length

Clause *6.7* has a listing of all security protocol parameters.

5.2.40 SECURITY PROTOCOL OUT (SPOUT) - B5h

The Security Protocol Out command is defined in SPC-4 (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause specifies the specific implementation.

The SECURITY PROTOCOL OUT command (see Table 132) is:

a) supported in encryption capable drives that are configured for AME-T10 mode (see 4.20.2); and
b) used to send data to the logical unit. The data sent specifies one or more operations to be performed by the logical unit. The format and function of the operations depends on the contents of the SECURITY PROTOCOL field (see table 178). Depending on the protocol specified by the SECURITY PROTOCOL field, the application client may use the SECURITY PROTOCOL IN command (see 5.2.39) to retrieve data derived from these operations.

NOTE 51 - The operation code, B5h has been recovered from a seldom used media changer (i.e. LUN 1) command (Request Volume Element Address). If the device driver being used still uses the LUN field of the CDB from SCSI-2 days, this command will be routed to the incorrect LUN. Because it is a DATA OUT type command, whereas the Request Volume Element Address is a DATA IN type command this can cause strange system behaviors.

Table 132 — Security Protocol Out B5h CDB

Byte	**Bit**							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Operation code (B5h)							
1	Security Protocol							
2	Security Protocol Specific							
3								
4	INC_512 (0b)	Reserved						
5	Reserved							
6	(MSB)	Allocation Length						
9								(LSB)
10	Reserved							
11	Control Byte (see 5.1.3)							

The following parameters apply:

- Security Protocol

Value	**Description**
20h	Tape Data Encryption security protocol

- Security Protocol Specific - The contents depend on the protocol specified by the Security Protocol field.
- Allocation Length

Clause *6.7* has a listing of all security protocol parameters.

5.2.41 SEND DIAGNOSTIC - 1Dh

The Send Diagnostic command is defined in SPC (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause specifies the specific implementation.

After a Send Diagnostic command completes, use the Receive Diagnostic Results command to receive the results.

For diagnostics that produce diagnostic results, the LUN should be reserved to allow the I_T nexus that issued the Send Diagnostic command to issue the Receive Diagnostic Results command before a different I_T nexus clears the results by reading them.

Table 133 — Send Diagnostic CDB

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Operation code (1Dh)							
1	Obsolete (LUN)			PF	Reserved	SelfTest	DevOfl	UnitOfl
2	Reserved							
3	MSB	Parameter List Length						
4								LSB
5	Control Byte (see 5.1.3)							

The following parameters apply:

- PF (Page Format): 1b
- SelfTest:

Value	**Description**
0b	Perform the diagnostic operation specified in the parameter list.
1b	Perform self test. See *6.1.3* for details of this diagnostic.

NOTE 52 - When Self Test is requested, no other diagnostic may be simultaneously requested.

- DevOfL (Device Off Line):

Value	**Description**
0b	is supported and prohibits any diagnostic operations that may be detected by subsequent I/O processes.
1b	is supported and grants permission to the target to perform diagnostic operations that may affect all the logical units on a target; that is, alteration of reservations, log parameters, or sense data.
x	in the diagnostic description indicates that either 0b or 1b may be used with identical effects.

NOTE 53 - DevOfl and UnitOfl are set by the system. These bits grant permission to the target to perform vendor-specific diagnostic operations on the target that may be visible to attached initiators. Thus, by preventing operations that are not enabled by these bits, the target assists the operating system in protecting its resources.

- UnitOfL (Unit Off Line):

Value	**Description**
0b	is supported and prohibits any diagnostic operations that may be detected by subsequent I/O processes.
1b	is supported and grants permission to the target to perform diagnostic operations that may affect the user medium on the logical unit; for example, write operations to the

	user-accessible medium, or operations that reposition the medium on sequential access devices.
x	in the diagnostic description indicates that either 0b or 1b may be used with identical effects.

- Parameter List Length: This field specifies the length in bytes of the parameter list that is transferred from the initiator to the target. A parameter list length value of zero specifies that no data is transferred. This condition is not considered an error. If the specified parameter list length results in truncation of one or more pages (PF bit set to 1b), the target returns CHECK CONDITION status with associated sense data of 5/2400 (Illegal Request, Invalid Field in CDB).

Clause *6.1* has a listing of all diagnostic parameters.

5.2.42 SET TIMESTAMP - A4h[0Fh]

The SET TIMESTAMP command is defined in SPC-4 (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause decribes the specific implementation.

The SET TIMESTAMP command (see table 134) requests the drive to initialize the timestamp (see 4.25), if the SCSIP bit is set to one or the TCMOS bit is set to one in MP 0Ah[01h]: Control Extension (see 6.6.8 on page 389). If the SCSIP bit is set to zero, the SET TIMESTAMP command is terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST, and the additional sense code set to INVALID FIELD IN CDB.

Table 134 — SET TIMESTAMP CDB

Byte	Bit							
	7	6	5	4	3	2	1	0
0	OPERATION CODE (A4h)							
1	Reserved				SERVICE ACTION (0Fh)			
2	Reserved							
5								
6	(MSB)	PARAMETER LIST LENGTH						
9								(LSB)
10	Reserved							
11	Control Byte (see 5.1.3)							

The following paramters apply:

- PARAMETER LIST LENGTH: The length in bytes of the SET TIMESTAMP parameters that is transferred from the application client to the device server. A PARAMETER LIST LENGTH of zero indicates that no data is transferred, and that no change is made to the timestamp.

5.2.42.1 SET TIMESTAMP Parameter List

The format for the parameter data for the SET TIMESTAMP command is shown in table 135.

Table 135 — SET TIMESTAMP parameter list format

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Reserved							
3								
4	(MSB)	TIMESTAMP						
9								(LSB)
10	Reserved							
11								

The definition of the parameters follows:

Byte	**Description**
0-3	Reserved
4-9	TIMESTAMP: The initial value of the timestamp in the format defined in Device Clocks (see 4.25 on page 83). The timestamp should be the number of milliseconds that have elapsed since midnight, 1

January 1970 UT. If the high order byte in the TIMESTAMP field is greater than F0h, the command is terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST, and the additional sense code set to INVALID FIELD IN PARAMETER LIST.

10-11 Reserved

On successful completion of a SET TIMESTAMP command the drive generates a UNIT ATTENTION condition for the initiator port associated with every I_T nexus except the I_T nexus on which the SET TIMESTAMP command was received, with the additional sense code set to TIMESTAMP CHANGED.

5.2.43 SPACE (6/16) - 11h/91h

The Space commands are defined in SSC (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause specifies the specific implementation.

The Space command is implemented similar to the Locate command, causing the tape to move at maximum speed when appropriate.

Table 136 — Space (6) CDB

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Operation code (11h)							
1	Obsolete (LUN)			Reserved		Code		
2	MSB	Count						
4								LSB
5	Control Byte (see 5.1.3)							

The Space(16) command (see table 137), operates identically to the Space(6) command, but allows specifying a Count field up to eight bytes in length. The Space (16) command is not supported on J1A devices. Following completion of a Space(16) command a Read Position command should be issued to obtain positioning information.

This command is newly added to the standards. The only exceptions to SSC-3 for this command are that the explicit command set is not supported and the list of values supported in the Code field.

Table 137 — Space(16) command (not J1A)

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Operation Code (91h)							
1	Reserved				Code			
2	Reserved							
3								
4	(MSB)	Count						
11								(LSB)
12	(MSB)	Parameter Length (0000h)						
13								
14	Reserved							
15	Control							

The following parameters apply:

- Code

Value	Description
000b	Blocks
001b	Filemarks
010b	Sequential Filemarks
011b	End of Data

- Count

  When spacing over blocks or filemarks, the count field specifies the number of blocks or filemarks to be spaced over in the current partition. A positive value N in the count field causes forward positioning (toward End of Partition) over N blocks or filemarks ending on the End of Partition side of the last block or filemark. A zero value in the count field causes no change of logical position. A negative value -N (two's complement notation) in the count field causes reverse positioning (toward Beginning Of Partition) over N blocks or filemarks ending on the Beginning of Partition side of the last block or filemark.

- Parameter Length

  If the Parameter Length field is set to any value other than zero, the command is terminated with a Check Condition status. The sense key is set to Illegal Request and the additional sense code is set to Invalid Field In CDB (5/2400h).

If the drive encounters End-of-Data (EOD) while processing this command, the command is terminated at the EOD position and CHECK CONDITION status is returned with associated sense data of 8/0005 (Blank Check, End-of-Data Detected).

5.2.44 SPACE (6/16) - 11h/91h (String Search) (not J1A)

See “MP 37h: String Search (not J1A)” on page 444, and “String Search Function (not J1A)” on page 79 for additional information.

Table 138 — Space (6) command (String Search)

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Operation Code: 11h							
1	Obsolete (LUN)			Reserved (00b)		Code		
2	MSB	Count						
4								LSB
5	Search 1b	FMKS	Control Byte (see 5.1.3)					

The Space (16) (String Search) command allows specifying a Count field up to eight bytes in length. Following completion of a Space(16) command, a Read Position command should be issued to obtain positioning information.

Table 139 — Space (16) command (String Search)

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Operation Code (91h)							
1	Reserved				Code			
2	Reserved							
3								
4	(MSB)	Count						
11								(LSB)
12	(MSB)	Parameter Length (0000h)						
13								
14	Reserved							
15	Search 1b	FMKS	Control Byte (see 5.1.3)					

NOTE 54 - The device logical position may be changed by this command.

NOTE 55 - Reverse direction Space (String Search) operations run very slowly and are not recommended.

NOTE 56 - If no valid search criteria exists, the command presents CHECK CONDITION status with sense key 5/2C00 (Illegal Request - Command Sequence Error).

The following parameters differ from the standard Space command:

- Search:

Value	Description
0b	this command is a standard Space command
1b	this command is a Space -String Search command

- FMKS:

Value	Description
0b	filemarks will be handled as per the standard Space command.
1b	filemarks will be ignored (treated as a logical block with no data - will not match any criteria) (only meaningful when Code field is 000b)

- Parameter Length

This field shall be set to 0. If the Parameter Length field is set to any other value, the command is terminated with a Check Condition status. The sense key is set to Illegal Request and the additional sense code is set to Invalid Field In CDB (5/2400h).

If a matching record is found, the command will return GOOD STATUS and the device will be logically positioned before the matching record (a Read command may be issued to read the matching record).

If a matching record is not found before the criteria of the Space command is fulfilled, the command presents NON-DEFERRED CHECK CONDITION status with sense key 0/1401 (No Sense - Record Not Found).

If a matching record is not found before Search Time elapses, the command presents NON-DEFERRED CHECK CONDITION status with sense key 0/2E00 (No Sense - Insufficient Time for Operation).

Other reportable statuses for the space command also apply in conditions where the Space stops due to other issues before a matching record is found (i.e., filemark is encountered on space blocks, BOP or EOD encountered, etc.).

5.2.45 STRING SEARCH - E3h (not J1A)

Snoop searches are performed during normal command processing, according to search parameters as setup in the String Search Mode Page (Search Control)

Explicit searches are performed using the new vendor unique command.

Table 140 — String Search - E3h

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Operation Code: E3h							
1	Obsolete (LUN)			Reserved		SMKS	FMKS	Immed
2	MSB	Search Count						
4								LSB
5	MSB	Match Count						
6								LSB
7	MSB	Allocation Length						
8								LSB
9	Control Byte (see 5.1.3)							

NOTE 57 - The device logical position may be changed by this command.

NOTE 58 - If no valid search criteria exists, the command presents CHECK CONDITION status with sense key 5/2C00 (Illegal Request - Command Sequence Error).

The following parameters apply:

- Immed (Immediate) (not supported):

Value	**Description**
0b	present status when command is completed
1b	present status when command is started, results must be acquired through Mode Sense or Read Buffer

- FMKS:

0b	stop when a filemark is encountered
1b	filemarks will be ignored (treated as a logical block with no data - will not match any criteria)

- Search Count: The maximum number of records to search. If this field is set to 000000h, then all records from the current location to EOD are searched (limited to Match Count and Search Time).
- Match Count: The maximum number of matches to gather in the Match List.

Value	**Description**
0000h	If Match Count is set to 0000h, then the search will stop on the next record (within Search Count) which matches the search criteria. If a matching record is found, the command will return GOOD STATUS and the device will be logically positioned before the matching record (a Read command may be issued to read the matching record). If a matching record is not found before the criteria of the Search command is fulfilled, the command presents NON-DEFERRED CHECK CONDITION status with sense key 0/1401 (No Sense - Record Not Found). If a matching record is not found before Search Time elapses, the command presents NON-DEFERRED CHECK CONDITION status with sense key 0/2E00 (No Sense - Insufficient Time for Operation). Other reportable statuses for the space command also

apply in conditions where the Space stops due to other issues before a matching record is found (i.e., filemark is encountered with FMKS set to 0b, EOD encountered, etc.).

other	If Match Count is non-zero, the search will stop once the Match List is full or Search Time elapses. The device will be logically positioned after the last record processed. If a matching record is not found before Search Time elapses, the command presents NON-DEFERRED CHECK CONDITION status with sense key 0/2E00 (No Sense - Insufficient Time for Operation). Other reportable statuses for the space command also apply in conditions where the operation stops due to other issues before a matching record is found (i.e., filemark is encountered with FMKS set to 0b, EOD encountered, etc.).

- Allocation Length: The length of the record found list, if zero, the search is performed but no results are returned. Use Mode Sense (String Search page) for summary results, or Read Buffer for full results.

The Search String method will return results via the return data. Results will also remain in Read Buffer ID 40h.

5.2.45.1 String Search Results Buffer (Match List)

Byte	Description
0	Page Code: 40h
1	Control Field
2-3	Page Length
5-63	Same as bytes 5-63 as defined by “MP 37h: String Search (not J1A)” on page 444
64-123	Vendor Reserved
124	Match List Entry Type
125	Match List Entry Length: size in bytes of each element 00h, 004h, 008h, or 10h
126-127	Match List Length: size of the results which follow
128-n	Match List: entries for matching block(s) in format determined by Match List Entry Type as specified below

The Match List entry for Match List Entry Type of 01h is defined below:

Byte	Description
0-3	Matching logical block number

The Match List entry for Match List Entry Type of 02h is defined below:

Byte	Description
0-3	Matching logical block number
4-7	Matching file number

The Match List entry for Match List Entry Type of 03h is defined below:

Byte	Description
0-7	Matching logical block number

The Match List entry for Match List Entry Type of 04h is defined below:

Byte	Description
0-7	Matching logical block number
8-15	Matching file number

5.2.46 TEST UNIT READY - 00h

The Test Unit Ready command is defined in SPC (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause specifies the specific implementation.

Table 141 — Test Unit Ready CDB

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Operation code (00h)							
1	Obsolete (LUN)			Reserved				
2	Reserved							EDCC
3	Reserved							
4								
5	Control Byte (see 5.1.3)							

The following parameters apply:

- EDCC (Enable Deferred CHECK CONDITION):

Value	**Description**
0b	Deferred CHECK CONDITION status is not to be reported for this command.
1b	Deferred CHECK CONDITION status may be reported for this command.

NOTE 59 - This is a vendor unique field and behavior which is different than the SPC standards.

5.2.47 VERIFY(6) - 13h (beginning with E05)

The VERIFY command is defined in SSC (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause describes the specific implementation. The VERIFY(6) command (see table 142) requests that the device server verify one or more logical block(s) or one or more logical file(s) beginning at the current logical position. Prior to performing the verify operation, the device server performs a synchronize operation (see 4.2.11).

Table 142 — VERIFY (6) CDB

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	OPERATION CODE (13h)							
1	Reserved		VTE	VLBPM	VBF	IMMED	BYTCMP	FIXED
2	(MSB)							
	VERIFICATION LENGTH							
4								(LSB)
5	Control Byte (see 5.1.3)							

The following parameters apply:

- VTE - If the verify to end-of-data (VTE) bit is set to one, then all logical blocks shall be verified between the current position and EOD (see 4.7.8). The behavior related to encrypted logical blocks is the same as if they are read with a READ command. The logical position upon successful completion of a verify with the VTE bit set to one shall be at EOD. If the VTE bit is set to one and the verification of a logical block fails, then the command shall terminate with CHECK CONDITION status, the SENSE VALID bit shall be set to zero, the INFORMATION field shall be set to zero, and the sense key, sense code, and additional sense code shall be set the same as it would be on a read. Upon termination, the medium shall be positioned after the logical block on which the verification failed (i.e., end-of-partition side). If the VTE bit is set to one, then the BYTCMP bit and the VBF bit shall be set to zero. If the VTE bit is set to one, then the VERIFICATION LENGTH field shall be ignored. If the VTE bit is set to one and the FIXED bit is set to one, then the length of each logical block shall be verified to be the current block length reported in the mode parameters block descriptor. If the VTE bit is set to one and the FIXED bit is set to zero, then the length of each logical block shall not be verified. If the VTE bit is set to zero, then a verify to EOD is not requested.
- VLBPM - If the verify logical block protection method (VLBPM) bit is set to one then the verification being performed also includes a verification that each logical block uses the logical block protection method specified in the Control Data Protection mode page. If the VLBPM bit is set to one and the verification fails because a logical block did not use the logical block protection method specified by the Control Data Protection mode page, then the device server terminates the command with CHECK CONDITION status, with the sense key set to MISCOMPARE and the additional sense code set to LOGICAL BLOCK PROTECTION METHOD ERROR. Upon termination, the medium is positioned after the logical block on which the verification failed (end-of-partition side). If the VLBPM bit is set to zero then the verification being performed does not require a verification that each logical block uses the logical block protection method specified in the Control Data Protection mode page. If a logical block is encrypted, then the behavior is the same as attempting to read an ecnrypted logical block with the READ command.
- VBF - If the verify by filemarks (VBF) bit is set to one, then the VERIFICATION LENGTH field contains a count of filemarks to be traversed. All logical blocks starting at the current position and continuing to the n^{th} filemark, where n is the value in the VERIFICATION LENGTH field shall be verified (see 4.7.7). The behavior related to encrypted logical blocks is the same as it is for a READ command. The logical position upon successful completion of a verify with the VBF bit set to one shall be after the n^{th} filemark. If the VBF bit is set to one and the verification of a logical block fails, then the command shall terminate with CHECK CONDITION status, the SENSE VALID bit shall be set to one, the INFORMATION field shall be set to the requested verification length minus the actual number of filemarks successfully traversed, and the sense key, sense code, and additional sense code values shall be the same as for a READ ocmmand. Upon termination, the medium shall be positioned after the logical block on which the verification failed (end-of-partition side). If the VBF bit is set to one, then the BYTCMP bit and the VTE bit shall be set to zero. If the VBF bit is set to one

and the FIXED bit is set to one, then the length of each logical block shall be verified to be the current block length reported in the mode parameters block descriptor. If the VBF bit is set to one and the FIXED bit is set to zero, then the length of each logical block shall not be verified. If the VBF bit is set to zero, then the VERIFICATION LENGTH field does not contain a count of filemarks to be traversed.

- IMMED: 0b

Value	**Description**
0b	The command does not return status until the verify operation has completed.
1b	Status is returned as soon as the command descriptor block has been validated.

- BYTCMP: 0b

  The verification is a verification of logical blocks on the medium (e.g., CRC, ECC). No data is transferred from the application client to the device server.

- FIXED (see 4.8)
- VERIFICATION LENGTH (see 4.23): The amount of data to verify in, logical files, logical blocks or bytes, as specified by the VBF bit and the FIXED bit. If the VTE bit is one, then the VERIFICATION LENGTH field shall be ignored. If the VERIFICATION LENGTH field is zero and the VTE bit is zero, then no data is verified and the current logical position is not changed. This condition is not considered an error.

The VERIFY(6) command terminates as follows:

a) when conditions related to encryption are met;
b) when the verification length has been satisfied and the VTE bit is zero;
c) when an incorrect-length logical block is encountered;
d) when a filemark is encountered and the VBF bit and VTE bit are zero;
e) when end-of-data is encountered;
f) when the end-of-partition is encountered;
g) when early-warning is encountered (if the REW bit is one in the Device Configuration mode page) and the VTE bit is zero;
h) when the logical block protection method used by the logical block is not the logical block protection method specified in the Control Data Protection mode page and the VLBPM bit is one; or
i) when an unrecoverable read error is encountered.

General Read-Type Handling (see 4.10.1 on page 40) provides additional information.

5.2.48 WRITE - 0Ah

The Write command is defined in SSC (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause specifies the specific implementation.

Table 143 — Write CDB

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Operation code (0Ah)							
1	Obsolete (LUN)			Reserved				Fixed
2	MSB	Transfer Length						
4								LSB
5	Control Byte (see 5.1.3)							

The following parameters apply:

- Fixed (see 4.8)
- Transfer Length(see 4.8)

5.2.49 WRITE ATTRIBUTE - 8Dh (beginning with E07)

The WRITE ATTRIBUTE command is defined in SPC-4 (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause describes the specific implementation.

The WRITE ATTRIBUTE command (see table 144) allows an application client to write attributes to medium auxiliary memory. The READ ATTRIBUTE command (see 5.2.13) is used to read these attribute. Application clients should issue READ ATTRIBUTE commands prior to using this command to discover device server support for medium auxiliary memory.

Table 144 — WRITE ATTRIBUTE CDB

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (8Dh)							
1	Reserved							WTC
2	Reserved							
4								
5	VOLUME NUMBER (00h)							
6	Reserved							
7	PARTITION NUMBER							
8	Reserved							
9								
10	(MSB)	PARAMETER LIST LENGTH						
13								(LSB)
14	Reserved							
15	Control Byte (see 5.1.3)							

The following parameters apply:

Byte Description

0 OPERATION CODE (8Dh)

1 Byte 1

Bit	Description
7-1	Reserved
0	WTC - Write-through cache

Value	Description
0b	The attributes in the parameter list may be cached.
1b	The attributes in the parameter list shall be synchronized with the medium auxiliary memory during the processing of the WRITE ATTRIBUTE command and GOOD status shall not be returned until the attributes have been synchronized with the medium auxiliary memory.

2-4 Reserved

5 VOLUME NUMBER (00h)

6 Reserved

7 PARTITION NUMBER - The number of the partition to which these attributes belong. This shall be zero if there is only one partition on the volume. The number of partitions of the medium auxiliary memory equals that of the attached medium.

8-9 Reserved

10-13 PARAMETER LIST LENGTH - The length in bytes of the parameter list contained in the Data-Out Buffer. A parameter list length of zero specifies that no parameter data is present; this shall not be considered an error. If the parameter list length results in the truncation of an attribute, the WRITE ATTRIBUTE command is terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST, and the additional sense code set to PARAMETER LIST LENGTH ERROR.

14 Reserved

15 CONTROL (see 5.1.3)

The parameter list shall have the format shown in table 145. Attributes shall be sent in ascending numerical order. If the attributes are not in order, then no attributes are changed and the WRITE ATTRIBUTE command is terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST, and the additional sense code set to INVALID FIELD IN PARAMETER LIST.

Table 145 — WRITE ATTRIBUTE parameter list format

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)	PARAMETER DATA LENGTH (n-3)						
3								(LSB)
	Attribute(s)							
4	Attribute 0 (see 6.5.1)							
	⋮							
n	Attribute x (see 6.5.1)							

The PARAMETER DATA LENGTH field should contain the number of bytes of attribute data and shall be ignored by the device server.

The format of the attributes is described in “READ ATTRIBUTE” on page 252.

If there is not enough space to write the attributes to the medium auxiliary memory, then no attributes shall be changed and the WRITE ATTRIBUTE command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST, and the additional sense code set to AUXILIARY MEMORY OUT OF SPACE.

If the medium auxiliary memory is not accessible because there is no medium present, then no attributes shall be changed and the WRITE ATTRIBUTE command shall be terminated with CHECK CONDITION status, with the sense key set to NOT READY, and the additional sense code set to MEDIUM NOT PRESENT.

If the medium is present but the medium auxiliary memory is not accessible, then no attributes shall be changed and the WRITE ATTRIBUTE command shall be terminated with CHECK CONDITION status, with the sense key set to MEDIUM ERROR, and the additional sense code set to LOGICAL UNIT NOT READY, AUXILIARY MEMORY NOT ACCESSIBLE.

If the medium auxiliary memory is not operational (e.g., bad checksum), the WRITE ATTRIBUTE command shall be terminated with CHECK CONDITION status, with the sense key set to MEDIUM ERROR, and the additional sense code set to AUXILIARY MEMORY WRITE ERROR.

If the WRITE ATTRIBUTE command parameter data contains an attribute with an ATTRIBUTE LENGTH field set to zero, then one of the following actions shall occur:

a) if the attribute state is unsupported or read only (see 4.23), then no attributes shall be changed and the WRITE ATTRIBUTE command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST, and the additional sense code set to INVALID FIELD IN PARAMETER LIST;
b) if the attribute state is read/write, the attribute shall be changed to the nonexistent state. This attribute shall not be returned in response to a READ ATTRIBUTE command and not be reported by the READ ATTRIBUTE command with ATTRIBUTE LIST service action; or
c) if the attribute state is nonexistent, the attribute in the WRITE ATTRIBUTE command parameter list shall be ignored; this shall not be considered an error.

No attributes shall be changed, the WRITE ATTRIBUTE command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST, and the additional sense code set to INVALID FIELD IN PARAMETER LIST if the parameter data contains any of the following:

a) an attempt to change an attribute in the read only state;
b) an attribute with incorrect ATTRIBUTE LENGTH field contents; or
c) an attribute with unsupported ATTRIBUTE VALUE field contents.

5.2.50 WRITE BUFFER - 3Bh

The Write Buffer command is defined in SPC (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause specifies the specific implementation.

The Write Buffer command is supported, but not all buffers are described in this document because most buffers are intended only to be written by the service representative or by manufacturing. Note that new microcode may also be loaded without requiring the use of the SCSI Write Buffer command, by using the Field Microcode Replacement (FMR) tape process described in the maintenance information manual for this product. See "Supported Buffers" on page 150 for a list of the buffers supported by the drive. Table 146 shows the command format.

Table 146 — Write Buffer CDB

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Operation code (3Bh)							
1	MODE SPECIFIC			MODE				
2	Buffer ID							
3	MSB			Buffer Offset				
5								LSB
6	MSB			Parameter List Length				
8								LSB
9	Control Byte (see 5.1.3)							

The following parameters apply:

- MODE SPECIFIC: This field is valid for MODE[0Dh] – Download microcode with offsets, select activation, save, and defer activate mode (beginning with E07) (see 5.2.50.7 on page 222) only. It is Reserved for other modes.

Bit	**Description**
7	PO_ACT (1b): Activate on Power On reset is required to be set to 1b, since deferred microcode always activates on power on reset.
6	HR_ACT (0b): Activate on Hard Reset is required to be set to 0b, since deferred microcode does not activate on a hard reset.
5	VSE_ACT: Selects whether or not to activate on medium removal.

Value	**Description**
0b	Do not set deferred microcode to be activated on medium removal (i.e., MPRSNT bit of VHF data transitions from 1b to 0b)
1b	Set deferred microcode to be activated on medium removal (i.e., MPRSNT bit of VHF data transitions from 1b to 0b). On control path drives (i.e., LUN 1 is enabled) this setting is allowed and ignored (i.e., an explicit activate with the

mode field set 0Fh or a power on reset is required to activate the deferred microcode).

- MODE:

Value	Description
01h	MODE[01h] – Vendor Unique (see 5.2.50.1 on page 221)
02h	MODE[02h] – Data (see 5.2.50.2 on page 221).
04h	MODE[04h] – Download microcode and activate (see 5.2.50.3 on page 221).
05h	MODE[05h] – Download microcode, save, and activate (see 5.2.50.4 on page 221).
06h	MODE[06h] – Download microcode with offsets and activate (see 5.2.50.5 on page 222): use of strictly increasing offsets is required.
07h	MODE[07h] – Download microcode with offsets, save, and activate (see 5.2.50.6 on page 222): use of strictly increasing offsets is required.
0Dh	MODE[0Dh] – Download microcode with offsets, select activation, save, and defer activate mode (beginning with E07) (see 5.2.50.7 on page 222): use of strictly increasing offsets is required.
0Eh	Not Supported.
0Fh	MODE[0Fh] – Activate deferred microcode mode (beginning with E07) (see 5.2.50.8 on page 222).

- BUFFER ID: If the MODE field is set to 04h, 05h, 06h, 07h, 0Dh, or 0Fh, then this field is ignored for all Buffer ID's other than 30h. The supported buffers are described in table 79 on page 150. If an unsupported buffer ID code is selected, the command is terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST, and the additional sense code set to INVALID FIELD IN CDB (5/2400h).
- BUFFER OFFSET: The relative byte location within the buffer to write the data transferred by this command.
- PARAMETER LIST LENGTH: If the BUFFER OFFSET and PARAMETER LIST LENGTH fields specify a transfer in excess of the buffer capacity, then the command is terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST, and the additional sense code set to INVALID FIELD IN CDB (5/2400h).

Supported Buffers (see 5.2.23.2 on page 150) describes the supported Buffer ID's.

5.2.50.1 MODE[01h] – Vendor Unique

This mode is the same as MODE[02h] – Data (see 5.2.50.2 on page 221).

5.2.50.2 MODE[02h] – Data

In this mode, the Data-Out Buffer contains buffer data destined for the device. The BUFFER ID field identifies a specific buffer within the device.If an unsupported buffer ID code is selected, the command is terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST, and the additional sense code set to INVALID FIELD IN CDB (5/2400h).

The MODE SPECIFIC field is reserved.

The BUFFER OFFSET field and PARAMETER LIST LENGTH field are as specified in the "The following parameters apply:" (see page 220) description above.

5.2.50.3 MODE[04h] – Download microcode and activate

This mode behaves the same as MODE[07h] – Download microcode with offsets, save, and activate (see 5.2.50.6 on page 222).

5.2.50.4 MODE[05h] – Download microcode, save, and activate

This mode behaves the same as MODE[07h] – Download microcode with offsets, save, and activate (see 5.2.50.6 on page 222).

5.2.50.5 MODE[06h] – Download microcode with offsets and activate

This mode behaves the same as MODE[07h] – Download microcode with offsets, save, and activate (see 5.2.50.6 on page 222).

5.2.50.6 MODE[07h] – Download microcode with offsets, save, and activate

In this mode, microcode is transferred to the device, saved to nonvolatile storage, and activated.

The MODE SPECIFIC field is reserved.

The BUFFER ID field is ignored.

The BUFFER OFFSET field and PARAMETER LIST LENGTH field are as specified in the “The following parameters apply:” (see page 220) description above.

5.2.50.7 MODE[0Dh] – Download microcode with offsets, select activation, save, and defer activate mode (beginning with E07)

In this mode, microcode is transferred to the device server using one or more WRITE BUFFER commands, saved to nonvolatile storage, and considered deferred. The deferred microcode is activated if a WRITE BUFFER command with MODE[0Fh] – Activate deferred microcode mode (beginning with E07) (see 5.2.50.8 on page 222) is processed.

The MODE SPECIFIC field specifies additional events that may be selected to activate the deferred microcode.

The BUFFER ID field, BUFFER OFFSET field, and PARAMETER LIST LENGTH field are defined in MODE[07h] – Download microcode with offsets, save, and activate (see 5.2.50.6 on page 222).

5.2.50.8 MODE[0Fh] – Activate deferred microcode mode (beginning with E07)

In this mode, deferred microcode is activated.

The MODE SPECIFIC field is reserved.

The the BUFFER ID field, the BUFFER OFFSET field, and the PARAMETER LIST LENGTH field are ignored.

If there is no deferred microcode the WRITE BUFFER command is terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST, and the additional sense code set to COMMAND SEQUENCE ERROR (5/2C00h).

5.2.51 WRITE DYNAMIC RUNTIME ATTRIBUTE - A4h[1Eh] or D2h (beginning with E07)

The WRITE DYNAMIC RUNTIME ATTRIBUTE command has a legacy format shown in table 147 and a standardized format shown in table 148.

Table 147 — WRITE DYNAMIC RUNTIME ATTRIBUTE command (legacy)

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (D2h)							
1	Reserved							
…								
5								
6	(MSB)	PARAMETER LIST LENGTH						
…								
9								(LSB)
10	Reserved							
11	Control							

Table 148 — WRITE DYNAMIC RUNTIME ATTRIBUTE CDB (standardized - supported after GA)

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (A4h)							
1	Reserved			SERVICE ACTION (1Eh)				
2	Reserved							
…								
5								
6	(MSB)	PARAMETER LIST LENGTH						
…								
9								(LSB)
10	Reserved							
11	Control							

5.2.51.1 WRITE DYNAMIC RUNTIME ATTRIBUTE parameter list

The parameter list shall have the format shown in table 149. Attributes that may be changed in a WRITE DYNAMIC RUNTIME ATTRIBUTE command are the initiator type attributes listed in 6.2.2.4.

Table 149 — WRITE DYNAMIC RUNTIME ATTRIBUTE parameter list

Bit Byte	7	6	5	4	3	2	1	0
0	RESERVED							
…								
7								
	Attribute(s) (see 6.2)							
8	Attribute 0 (see 6.2.1)							
…								
	Attribute x (see 6.2.1)							
…								
n								

If an attribute that is not an initiator type is sent in the list of attributes, then this shall not be considered an error, the attribute shall be ignored, and the remaining attributes shall be processed normally.

The device server shall process attributes in the order received. Attributes shall be sent in ascending order by attribute identifier. If the attributes are not in order, then no attributes shall be changed and the WRITE DYNAMIC RUNTIME ATTRIBUTE command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST, and the additional sense code set to INVALID FIELD IN PARAMETER LIST.

If an attribute that requires creation of that attribute is sent in the list of attributes and there are no available resources to create that attribute, then all attributes for which there are resources shall be changed and the WRITE DYNAMIC RUNTIME ATTRIBUTE command shall be terminated with a CHECK CONDITION with the sense code set to RECOVERED ERROR and the additional sense code set to INSUFFICIENT RESOURCES (i.e., 1h / 5503h).

If the WRITE DYNAMIC RUNTIME ATTRIBUTE command parameter data contains an attribute with an ATTRIBUTE LENGTH field (see 6.2.1) set to zero, then one of the following actions shall occur:

a) if the attribute state is unsupported or read only (see 6.2.1), then the attribute shall not be changed and the WRITE DYNAMIC RUNTIME ATTRIBUTE command shall continue processing normally; this shall not be considered an error;
b) if the attribute state is read/write, the attribute shall be changed to the nonexistent state. This attribute shall not be returned in response to a READ DYNAMIC RUNTIME ATTRIBUTE command; or
c) if the attribute state is nonexistent, the attribute in the WRITE ATTRIBUTE command parameter list shall be ignored; this shall not be considered an error.

If the WRITE DYNAMIC RUNTIME ATTRIBUTE command parameter data contains an attribute with an ATTRIBUTE LENGTH set to a non-zero value other than that specified in 6.2.2.4, then one of the following actions shall occur:

a) if the FORMAT is not ASCII, then the attribute shall be ignored; or
b) if the FORMAT is ASCII, then the attribute shall be:
   A) truncated to the length specified in 6.2.2.4; or
   B) ignored.

If the parameter list length results in the truncation of an attribute, the command shall be processed normally but the truncated attribute shall be ignored.

5.2.52 WRITE FILEMARKS - 10h

The Write Filemarks command is defined in SSC (see the latest version at http://www.t10.org/drafts.htm#SCSI3_CMNDSETS). This clause specifies the specific implementation.

Table 150 — Write Filemarks CDB

Byte	**Bit**							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Operation code (10h)							
1	Obsolete (LUN)			Reserved			WSmk	Immed
2	MSB	Transfer Length						
4								LSB
5	Control Byte (see 5.1.3)							

The following parameters apply:

- WSmk (Write Setmark): 0b
- Immed (Immediate):

Value	**Description**
0b	present status when command is completed.
1b	present status when command is verified.

- Transfer Length:

  Because the WSmk is set to 0b, the field indicates the number of filemarks to be written. After any buffered write operation completes, the initiator can issue a Write Filemarks command with the Immed bit set to 0b and the Transfer Length set to 000000b to ensure that all buffered data and filemarks are successfully written to the medium (synchronized).

6. Parameters for SCSI Commands

This clause describes the parameters used in SCSI commands that are supported on LUN 0.

Parameters are described in a “term-definition” format. In this format, the bits or bytes to be described are highlighted and listed on the left. The definition for the bits or bytes is to the right (not highlighted).

The following is a list of abbreviations used to describe the various parameters referenced in this document as well as a reference to the parameter clause in which those specific parameters are defined:

Term	**Description**
Daig	Diagnostic Parameters (see 6.1 on page 229)
DRA	Dynamic runtime attributes (DRA) (see 6.2 on page 253)
IP	Inquiry Vital Product Data Parameters (see 6.3 on page 261)
LP	Log Parameters (see 6.4 on page 287)
MAM	Medium auxiliary memory attributes (MAM) (see 6.5 on page 363)
MP	Mode Parameters (see 6.6 on page 373)
SPP	Security Protocol Parameters (SPP) (see 6.7 on page 457)
Sense	Sense data (see 6.8 on page 481)
XCOPY	XCOPY Parameters (see 6.9 on page 495)

6.1 Diagnostic Parameters

Diagnostic parameters are used with the Send Diagnostic command (see 5.2.41) and the Receive Diagnostic Results command (see 5.2.28).

Terms used in this clause are:

Term	Description
CD	Cryptographic Diagnostic
Diag	Diagnostic
Page	Page Code
Parm	Parameter
RcvDiag	Receive Diagnostic Results
SendDaig	Send Diagnostic

6.1.1 Diag Page Formats

The drive supports two diagnostic pages for normal use: Page 00h and 80h. Page 81h is for engineering use only.

6.1.1.1 Page 00h

6.1.1.2 SendDiag Data - Page 00h

The format for the Send Diagnostic command follows:

Byte	Description
0	Page Code: 00h
1	Reserved (00h)
2-3	Page Length: 0000h

6.1.1.3 RcvDiag - Page 00h

The format for the Receive Diagnostic Results command follows:

Byte	Description
0	Page Code: 00h
1	Reserved
2-3	Page Length: 0002h
4	Page Code Supported: 00h
5	Page Code Supported: 80h

6.1.1.4 Page 80h

Page Code 80h is a general purpose page for sending flags and diagnostic parameters to the target.

6.1.1.4.1 SendDiag Command - Page 80h

The format for the Send Diagnostic command follows:

Byte	Description
0	Page Code: 80h
1	Reserved

2-3	Page Length (n-3)
4-5	Diagnostic ID This field specifies the diagnostic that is to be run
6	Flags (Send Diagnostic command)

Bits	**Description**
7-1	Reserved
0	Cartridge Required Set to 1b when a cartridge is required for a diagnostic. When 1b, a cartridge must be loaded and ready for the Send Diagnostic command to be accepted. See specific diagnostic descriptions for cartridge use: some diagnostics require this bit to be set to 1b, some require it to be set to 0b, and some do not require a specific bit setting. • 0b: No cartridge required • 1b: Cartridge required

7	Reserved
8-n	Diagnostic Parameters The Diagnostic Parameters field contains the parameters required to run the diagnostic.

See 6.1.2 for a list of supported diagnostic page 80h routines.

6.1.1.5 RcvDiag - Page 80h

The format for the Receive Diagnostic Results command follows:

Byte	**Description**
0	Page Code: 80h
1	Reserved
2-3	Page Length (n-3)
4-5	Diagnostic ID This field contains the same value as that sent with the Send Diagnostic command for which this response is associated.
6	Flags Set to 00h for the Receive Diagnostic Results command
7	Reserved
8-n	Diagnostic Results The Diagnostic Results field contains the results from the diagnostic.

See the individual Send Diagnostic parameter descriptions for the field contents. Refer to Supported Page 80h Diags (see 6.1.2) for a list of diagnostic parameters supported by the drive.

6.1.1.6 Page 81h

This is a special page for media manufacturing use only (Arbitrary Commands).

For more information on this page, please contact IBM.

6.1.2 Supported Page 80h Diags

Table 151 shows the supported diagnostic page 80h routines and indicates values required in the Send Diagnostic CDB. These diagnostics reside in the device. (See "SEND DIAGNOSTIC - 1Dh" on page 203 and "RECEIVE DIAGNOSTIC RESULTS - 1Ch" on page 170 for additional information on the commands.) Individual diagnostic descriptions follow table 151.

Table 151 — Supported Page 80 Diagnostic Routines

ID	Name/Description	See Page	Parm Length	Result Length[β]	Self Test	Dev Ofl	Unit Ofl	Cart Req'd
None	Self Test	203	0000h	-	1	0	0	-
0090h	Primary Port Wrap Test	233	0008h	0051h	0	X	X	X
0100h	POST A Diagnostic	234	0008h	0051h	0	X	X	X
0101h	POST B Diagnostic[a]	236	0008h	0051h	0	X	1	1
010Ah	Retension Medium	237	000Eh	000Eh	0			1
0160h	Force Dump	239	0008h	-	0	1	X	X
0161h	Write Dump to Cartridge[a]	240	0008h	0051h	0	X	1	1
0170h	Create FMR Cartridge[a]	241	0008h	0051h	0	X	1	1
0190h	Set Traps	243	000Ah	-	0	X	X	X
0191h	Remove Traps	244	000Ah	-	0	X	X	X
0210h	Terminate Immediate Command	245	000Ah	-	0	X	1	1
2001h	Reset Volume SARS	247	0008h	-	0	1	X	1
2002h	Reset Drive (Soft)	248	0008h	-	0	X	X	0

Legend

- Not Applicable

[a] These diagnostics will destroy all data on the currently mounted cartridge.

[b] GOOD status is returned for diagnostics which expect result data based solely on validation of the diagnostic. The Receive Diagnostics Results command must be used to determine the success or failure of the actual execution of such diagnostics. Diagnostic procedures are recommended above.

[c]

6.1.3 Diag - SelfTest: Self Test

When the SelfTest bit is 1b in the Send Diagnostic command (See “SEND DIAGNOSTIC - 1Dh” on page 203 and “RECEIVE DIAGNOSTIC RESULTS - 1Ch” on page 170 for additional information on the commands), the target runs the normal power-on self test (POST A) diagnostics that occur at bring-up. No diagnostic results are returned.

6.1.3.1 SendDiag Command - Self Test

Table 152 shows the Send Diagnostic command format to specify Self Test (the SelfTest bit is set to 1b).

Table 152 — SendDiag CDB - Self Test

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Operation code (1Dh)							
1	Obsolete (LUN)			PF (1b)	Reserved (0b)	SelfTest (1b)	DevOfl (0b)	UnitOfl (0b)
2	Reserved (00h)							
3	MSB	Parameter List Length (0000h)						
4								LSB
5	Control Byte (see 5.1.3)							

When the SelfTest bit is 1b in the Send Diagnostic command, the target will execute its default power-on self test. If the self test successfully passes, the command is terminated with Good status. If the self test detects a failure, the command is terminated with CHECK CONDITION status and the sense key is set to Hardware Error.

6.1.3.2 RcvDiag Data - Self Test

There are no diagnostic results for the self test.

6.1.4 Diag - 0090h: Primary port wrap test

This test will perform a wrap test on the specified primary port. A wrap tool must be attached prior to running this command. See SEND DIAGNOSTIC - 1Dh (see 5.2.41 on page 203) and RECEIVE DIAGNOSTIC RESULTS - 1Ch (see 5.2.28 on page 170) for additional information on the commands.

6.1.4.1 SendDiag Parm Data - Primary port wrap test

Table 153 shows the parameter data for the Send Diagnostic command.

Table 153 — Primary Port Wrap Test SendDiag Parm Data

Byte	7	6	5	4	3	2	1	0
0	PAGE CODE (80h)							
1	Reserved							
2	(MSB)	PAGE LENGTH (0004h)						
3								(LSB)
4	DIAGNOSTIC ID (0090h)							
5								
6	FLAGS (0000000b)							CR (0b)
7	PORT IDENTIFIER							

The following parameters apply:

- CR (cartridge required): 0b
- PORT IDENTIFIER: This field is identical to the PORT IDENTIFIER field described in IP 83h: Device Identification (see 6.3.4 on page 264). If the value of the PORT IDENTIFIER is zero, then the wrap test will be performed on all primary ports. If the value is a valid port identifier, the wrap test is performed on the port indicated by the PORT IDENTIFIER field. If an invalid port identifier value is used, the drive will respond with a CHECK CONDITION for INVALID FIELD IN PARAMETER LIST.

6.1.4.2 RcvDiag Data - Primary port wrap test

Table 154 shows the diagnostic results data received from the attempted Primary port wrap test diagnostic.

Table 154 — Primary Port Wrap Test Receive Diag Parm Data

Byte	7	6	5	4	3	2	1	0
0	PAGE CODE (80h)							
1	Reserved							
2	(MSB)	PAGE LENGTH (004Dh)						
3								(LSB)
4	DIAGNOSTIC ID (0090h)							
5								
6	FLAGS							
7	Reserved							
8	FLAGS (00000b)					BLOCKED	SIM/MIM PRESENT	ERROR
9	SIM/MIM Message or All Zeros							
80								

6.1.5 Diag - 0100h: POST A

This diagnostic runs the POST A (power-on self test) diagnostics, as does running the Self Test diagnostic by setting the SelfTest bit to 1b in a Send Diagnostic command. However, unlike Self Test, the POST A diagnostic returns data through the Receive Diagnostic Results command. (See “SEND DIAGNOSTIC - 1Dh” on page 203 and “RECEIVE DIAGNOSTIC RESULTS - 1Ch” on page 170 for additional information on the commands)

6.1.5.1 SendDiag Parm Data - POST A

Table 155 shows the parameter data for the Send Diagnostic command.

Table 155 — SendDiag Parm Data - POST A

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Page Code (80h)							
1	Reserved (00h)							
2	MSB	Page Length (0004h)						
3								LSB
4	MSB	Diagnostic ID (0100h)						
5								LSB
6	Flags (0000000b)							Cartridge Required [a] (x)
7	Reserved (00h)							

[a] The Cartridge Required flag can be set to 0b or 1b. If a cartridge is in the drive when this diagnostic is received, some diagnostics will not run. If the diagnostic is blocked because a cartridge is loaded in the drive or for any other reason, the Diagnostic Blocked bit is set in the Receive Diagnostics Results data.

6.1.5.2 RcvDiag Data - POST A

Table 156 shows the diagnostic results data received from the POST A diagnostic.

Table 156 — RcvDiag Data - POST A

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Page Code (80h)							
1	Reserved (00h)							
2	MSB	Page Length (0004h)						
3								LSB
4	MSB	Diagnostic ID (0100h)						
5								LSB
6	Flags (00h)							
7	Reserved (00h)							
8	Reserved (00000b)					Diagnostic Blocked	SIM/MIM Present	Error
9	SIM/MIM message or all zeros							
80								

[a] The Error bit in byte 8 is set when the diagnostic detects an error.
[b] The SIM/MIM Present bit in byte 8 is set when a SIM or MIM message is contained in the diagnostic results.
[c] SIM/MIM messages are defined exactly as described in LP 31h: SIM/MIM (see 6.4.13); the SIM/MIM is not valid if the SIM/MIM Present bit is 0b.
[d] The Diagnostic Blocked bit is set when the diagnostic cannot run all its tests. This occurs if a cartridge is in the drive.

6.1.6 Diag - 0101h: POST B

This diagnostic causes all of the device read, write, and motion test diagnostics to be executed. Thus, a cartridge is required to be loaded to run the diagnostic. See "SEND DIAGNOSTIC - 1Dh" on page 203 and "RECEIVE DIAGNOSTIC RESULTS - 1Ch" on page 170 for additional information on the commands.

6.1.6.1 SendDiag Parm Data - POST B

Table 157 shows the parameter data for the Send Diagnostic command.

Table 157 — SendDiag Parm Data - POST B

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Page Code (80h)							
1	Reserved (00h)							
2	MSB	Page Length (0004h)						
3								LSB
4	MSB	Diagnostic ID (0101h)						
5								LSB
6	Flags (0000000b)							Cartridge Required (1b)
7	Reserved (00h)							

[a] Cartridge Required=1b, a cartridge must be loaded and ready.

6.1.6.2 RcvDiag Data - POST B

Table 158 shows the diagnostic results data received from the POST B diagnostic.

Table 158 — RcvDiag Data - POST B

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Page Code (80h)							
1	Reserved (00h)							
2	MSB	Page Length (004Dh)						
3								LSB
4	MSB	Diagnostic ID (0101h)						
5								LSB
6	Flags (00h)							
7	Reserved (00h)							
8	Reserved (00000b)					Diagnostic Blocked	SIM/MIM Present	Error
9	SIM/MIM message or all zeros							
80								

[a] The Error bit in byte 8 is set when the diagnostic detects an error.
[b] The SIM/MIM Present bit in byte 8 is set when a SIM or MIM message is contained in the diagnostic results.
[c] SIM/MIM messages are defined exactly as described in LP 31h: SIM/MIM (see 6.4.13); the SIM/MIM is not valid if the SIM/MIM Present bit is 0.

6.1.7 Diag -010Ah: Retension Medium

This diagnostic causes the media to be located to end of tape and back while the drive performs a servo verify to generate a Media Quality status. A cartridge is required to be loaded to run this diagnostic. See “SEND DIAGNOSTIC - 1Dh” on page 203 and “RECEIVE DIAGNOSTIC RESULTS - 1Ch” on page 170 for additional information on the commands.

6.1.7.1 SendDiag Parm Data - Retension Medium

Table 159 shows the parameter data for the Send Diagnostic command.

Table 159 — SendDiag Parm Data - Retension Medium

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Page Code (80h)							
1	Reserved (00h)							
2	Page Length (000Ah)							
3								
4	Diagnostic ID (010Ah)							
5								
6	Flags (0000000b)							Cart Req’d (1b)
7	Refresh Type (01h)							
8	Reserved (00h)							
9	Reserved (0000h)							
10								

[a] Cartridge Required=1b, a cartridge must be loaded and ready.
[b] Refresh Type=01h, Media Refresh with servo verify

6.1.7.2 RcvDiag Data - Retension Medium

Table 160 shows the diagnostic results data received from the Retension Medium diagnostic.

Table 160 — RcvDiag Data - Retension Medium

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Page Code (80h)							
1	Reserved (00h)							
2	Page Length (000Ah)							
3								
4	Diagnostic ID (010Ah)							
5								
6	Reserved (00000000h)							
9								
10	Media Quality							
11	Reserved (00000b)					Step complete flag (0b)	Seq complete flag (0b)	Fail flag
12	Reserved (0000h)							
13								

[a] Media Quality: media quality metric value from 00h to FFh, where 00h is the best quality and FFh is the poorest quality. If the Fail Flag is active, the Media Quality number is invalid

[b] Fail Flag: indicates whether the Media Quality number exceeds the fail quality threshold.

[c] Pass = 0b, Fail = 1b.

6.1.8 Diag - 0160h: Force Dump

This diagnostic forces a dump. The dump data is stored in device control storage and can be read by the Read Buffer command (Buffer ID of 00h) (see 5.2.23)

When a higher priority dump has been generated automatically by the drive but has not yet been read, the drive will ignore this Send Diagnostic command and return GOOD status.

See “SEND DIAGNOSTIC - 1Dh” on page 203 and “RECEIVE DIAGNOSTIC RESULTS - 1Ch” on page 170 for additional information on the commands.

6.1.8.1 SendDiag Parm Data - Force Dump

Table 161 shows the parameter data for the Send Diagnostic command.

Table 161 — SendDiag Parm Data - Force Dump

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Page Code (80h)							
1	Reserved (00h)							
2	MSB	Page Length (0004h)						
3								LSB
4	MSB	Diagnostic ID (0160h)						
5								LSB
6	Flags (0000000b)							Cartridge Required (x)
7	Reserved (00h)							

[a] The Cartridge Required flag may be set to 0b or 1b.

6.1.8.2 RcvDiag Data - Force Dump

There are no diagnostic results for this function.

6.1.9 Diag - 0161h: Write Dump to Cartridge

This diagnostic causes dump information residing in the device control storage to be written to a cartridge without the need to retrieve the dump data across the host interface. A cartridge is required to be loaded to run the diagnostic. See “SEND DIAGNOSTIC - 1Dh” on page 203 and “RECEIVE DIAGNOSTIC RESULTS - 1Ch” on page 170 for additional information on the commands.

6.1.9.1 SendDiag Parm Data - Write Dump to Cartridge

Table 162 shows the parameter data for the Send Diagnostic command.

Table 162 — SendDiag Parm Data - Write Dump to Cartridge

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Page Code (80h)							
1	Reserved (00h)							
2	MSB	Page Length (0004h)						
3								LSB
4	MSB	Diagnostic ID (0161h)						
5								LSB
6	Flags (0000000b)							Cartridge Required (1b)
7	Reserved (00h)							

[a] Cartridge Required=1b, a cartridge must be loaded and ready.

6.1.9.2 RcvDiag Data - Write Dump to Cartridge

Table 163 shows the diagnostic results data received from the Write Dump to Cartridge diagnostic.

Table 163 — RcvDiag Data - Write Dump to Cartridge

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Page Code (80h)							
1	Reserved (00h)							
2	MSB	Page Length (004Dh)						
3								LSB
4	MSB	Diagnostic ID (0161h)						
5								LSB
6	Flags (00h)							
7	Reserved (00h)							
8	Reserved (00000b)					Diagnostic Blocked	SIM/MIM Present	Error
9	SIM/MIM message or all zeros							
80								

[a] The Error bit in byte 8 is set when the diagnostic detects an error.

[b] The SIM/MIM Present bit in byte 8 is set when a SIM or MIM message is contained in the diagnostic results.

[c] SIM/MIM messages are defined exactly as described in LP 31h: SIM/MIM (see 6.4.13); the SIM/MIM is not valid if the SIM/MIM Present bit is 0b.

6.1.10 Diag - 0170h: Create FMR Cartridge

See “SEND DIAGNOSTIC - 1Dh” on page 203 and “RECEIVE DIAGNOSTIC RESULTS - 1Ch” on page 170 for additional information on the commands. This diagnostic causes the device microcode to be written to a cartridge, creating a field microcode replacement (FMR) cartridge. A cartridge is required to be loaded to run the diagnostic.

This copies the functional microcode load onto a cartridge for transporting to another drive when a FMR cartridge is not available.

NOTE 60 - This function is also available from the CE service panel.

6.1.10.1 SendDiag Parm Data - Create FMR Cartridge

Table 164 shows the parameter data for the Send Diagnostic command.

Table 164 — SendDiag Parm Data - Create FMR Cartridge

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Page Code (80h)							
1	Reserved (00h)							
2	MSB	Page Length (0004h)						
3								LSB
4	MSB	Diagnostic ID (0170h)						
5								LSB
6	Flags (0000000b)							Cartridge Required (1b)
7	Reserved (00h)							

[a] Cartridge Required=1b, a cartridge must be loaded and ready.

6.1.10.2 RcvDiag Data - Create FMR Cartridge

Table 165 shows the diagnostic results data received from the Create FMR Cartridge diagnostic.

Table 165 — RcvDiag Data - Create FMR Cartridge

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Page Code (80h)							
1	Reserved (00h)							
2	MSB	Page Length (004Dh)						
3								LSB
4	MSB	Diagnostic ID (0170h)						
5								LSB
6	Flags (00h)							
7	Reserved (00h)							
8	Reserved (000000b)					Diagnostic Blocked	SIM/MIM Present	Error
9		SIM/MIM message or all zeros						
80								

[a] The Error bit in byte 8 is set when the diagnostic detects an error.

[b] The SIM/MIM Present bit in byte 8 is set when a SIM or MIM message is contained in the diagnostic results.

[c] SIM/MIM messages are defined exactly as described in LP 31h: SIM/MIM (see 6.4.13); the SIM/MIM is not valid if the SIM/MIM Present bit is 0b.

6.1.11 Diag - 0190h: Set Traps

See "SEND DIAGNOSTIC - 1Dh" on page 203 and "RECEIVE DIAGNOSTIC RESULTS - 1Ch" on page 170 for additional information on the commands. This diagnostic permits a SCSI interface user to set a microcode trap that causes a dump to occur when the trap is sprung. The drive continues to operate after the dump completes.

6.1.11.1 SendDiag Parm Data - Set Traps

Table 166 shows the parameter data for the Send Diagnostic command.

Table 166 — SendDiag Parm Data - Set Traps

Byte	**Bit**							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Page Code (80h)							
1	Reserved (00h)							
2	MSB	Page Length (0006h)						
3								LSB
4	MSB	Diagnostic ID (0190h)						
5								LSB
6	Flags (0000000b)							Cartridge Required (x)
7	Reserved (00h)							
8	MSB	Fault Symptom Code						
9								LSB

[a] Cartridge Required=1, a cartridge must be loaded and ready before the diagnostic is run.

6.1.11.2 RcvDiag Data - Set Traps

There are no diagnostic results for this function.

6.1.12 Diag - 0191h: Remove Traps

See “SEND DIAGNOSTIC - 1Dh” on page 203 and “RECEIVE DIAGNOSTIC RESULTS - 1Ch” on page 170 for additional information on the commands. This diagnostic permits a SCSI interface user to remove a microcode trap that was set either via the CE service panel or the SCSI Set Traps Diagnostic.

6.1.12.1 SendDiag Parm Data - Remove Traps

Table 167 shows the parameter data for the Send Diagnostic command.

Table 167 — SendDiag Parm Data - Remove Traps

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Page Code (80h)							
1	Reserved (00h)							
2	MSB	Page Length (0006h)						
3								LSB
4	MSB	Diagnostic ID (0191h)						
5								LSB
6	Flags (0000000b)							Cartridge Required (x)
7	Reserved (00h)							
8	MSB	Fault Symptom Code						
9								LSB

[a] Cartridge Required=1, a cartridge must be loaded and ready before the diagnostic is run.

6.1.12.2 RcvDiag Data - Remove Traps

There are no diagnostic results for this function.

6.1.13 Diag 0210h: Terminate Immediate Command (Not J1A)

This diagnostic terminates all processing associated with the specified command that had been previously issued with the IMMED bit set to one. The processing to be terminated is specified by the OPERATION CODE / SERVICE ACTION pair of the command that initiated the processing. If the processing specified to terminate exists, then the drive returns Good status after the termination of the processing is complete (i.e., termination of processing is not instantaneous). If the processing specified to terminate does not exist, then the drive behaves as specified by the setting of the CC bit. If an error occurs during the processing of this Send Diagnostic command, Good status is returned and the error is reported as a deferred error to the next eligible command.

6.1.13.1 Send Data – Terminate Immed Command

Table 3 shows the Send Diagnostic parameter data for the Terminate Immed Command diagnostic.

Table 168 — Send Data – (Diag 0210h) Terminate Immed Command

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	PAGE CODE (80h)							
1	Reserved (00h)							
2	(MSB)	PAGE LENGTH (0006h)						
3								(LSB)
4	(MSB)	DIAGNOSTIC ID (0210h)						
5								(LSB)
6	FLAGS (0000000b)							CR(1b)
7	Reserved (00h)							
8	OPERATION CODE							
9	CC	Reserved (00b)		SERVICE ACTION				

The following parameters apply:

Byte	**Description**
0	PAGE CODE - Selects the diagnostic page as 80h which is the IBM general purpose page.
1	Reserved
2-3	PAGE LENGTH - The length in bytes of the diagnostic parameters that follow this field.
4-5	DIAGNOSTIC ID - The ID of the specific general purpose diagnostic.
6	FLAGS - Flags specific to this diagnostic.

Bit	**Description**
7-1	Reserved
0	CR - Cartridge Required (1b)

Value	**Description**
0b	A cartridge is not required to be loaded and ready.
1b	A cartridge must be loaded and ready.

7	Reserved
8	OPERATION CODE - The operation code of the command to terminate.

9

Bit	Description
7	CC - Check Condition bit
6-4	Reserved
4-0	SERVICE ACTION - The service action of the command to terminate. If there is no service action, then the service action field shall be set to zero

Value	Description
0b	Return Good status if the command specified by the OPERATION CODE / SERVICE ACTION pair is not currently active.
1b	Return Check Condition – Invalid Field in Parameter Data (5/2600) with Field pointer set to the OPERATION CODE field if the command specified by the OPERATION CODE / SERVICE ACTION pair is not currently active.

Table 169 lists the commands supported by the Terminate Immediate Command diagnostic.

Table 169 — Supported Commands in the Terminate Immediate Command diagnostic

Command	Op Code	Service Action	State of Drive after Command is Terminated
Erase	19h	00h	The drive stops processing and leaves the medium positioned at the stopping point. Note - This cancels the long erase. If the long erase was being used as part of a process to securely erase the medium (e.g., TSM Secure Erase), then this process must be redone in order to achieve the desired complete erasure.
Send Diagnostic	1Dh	00h	The drive stops processing, clears the buffer, and, if the logical position of the medium was changed, rewinds to BOP.
VERIFY	13h	00h	The drive stops processing and leaves the medium positioned at the stopping point. The READ POSITION command should be used to determine the current logical position of the medium. Note - The stopping point may be anywhere between the starting point and the logical position that would have resulted had the processing from the original command completed processing.

6.1.13.2 Results Data – Terminate Immed Command

There is no Receive Diagnostic Results data for this function.

6.1.14 Diag - 2001h: Reset Volume SARS

See “SEND DIAGNOSTIC - 1Dh” on page 203 and “RECEIVE DIAGNOSTIC RESULTS - 1Ch” on page 170 for additional information on the commands. This diagnostic resets the VOLUME SARS data in the currently loaded cartridge.

6.1.14.1 SendDiag Parm Data - Reset Volume SARS

Table 170 shows the parameter data for the Send Diagnostic command.

Table 170 — SendDiag Parm Data - Reset Volume SARS

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Page Code (80h)							
1	Reserved (00h)							
2	MSB	Page Length (0004h)						
3								LSB
4	MSB	Diagnostic ID (2001h)						
5								LSB
6	Flags (0000000b)							Cartridge Required (1b)
7	Reserved (00h)							

6.1.14.2 RcvDiag Data - Reset Volume SARS

There are no diagnostic results for this function.

6.1.15 Diag - 2002h: Reset Drive

See “SEND DIAGNOSTIC - 1Dh” on page 203 and “RECEIVE DIAGNOSTIC RESULTS - 1Ch” on page 170 for additional information on the commands. This diagnostic aborts all current drive operations and restarts the functional microcode. This reset is equivalent to a power on reset. (This function can also be specified from the CE service panel.)

6.1.15.1 SendDiag Parm Data - Reset Drive

Table 171 shows the parameter data for the Send Diagnostic command.

Table 171 — SendDiag Parm Data - Reset Drive

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Page Code (80h)							
1	Reserved (00h)							
2	MSB	Page Length (0004h)						
3								LSB
4	MSB	Diagnostic ID (2002h)						
5								LSB
6	Flags (0000000b)							Cartridge Required (0b)
7	Reserved (00h)							

6.1.15.2 RcvDiag Command - Reset Drive

There are no diagnostic results for this function.

6.1.16 Diag - 3000h: String Search (not J1A)

See “SEND DIAGNOSTIC - 1Dh” on page 203 and “RECEIVE DIAGNOSTIC RESULTS - 1Ch” on page 170 for additional information on the commands. This diagnostic performs a String Search operation as specified. See “MP 37h: String Search (not J1A)” on page 444, and “String Search Function (not J1A)” on page 79 for additional information.

NOTE 61 - The device logical position may be changed by this command.

NOTE 62 - If valid search criteria is not specified, the command presents CHECK CONDITION status with sense key 5/2C00 (Illegal Request - Command Sequence Error).

NOTE 63 - Mode Page 37h and its subpage(s) may be altered by this command. See “Search Criteria 01h - String Descriptor 10h” on page 447 for a detailed description of descriptors.

6.1.16.1 SendDiag Parm Data - String Search

Table 172 — SendDiag Parm Data - String Search

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Page Code: 80h							
1	Reserved (00h)							
2	MSB	Page Length (n-3)						
3								LSB
4	MSB	Diagnostic ID (3000h)						
5								LSB
6	Flags (00000000b)							Cart req'd (1b)
7	Reserved					SMKS	FMKS	Immed
8	MSB	Search Count						
11								LSB
12	MSB	Match Count						
15								LSB
16	Match List Entry Type							
17	MSB	Reserved						
27								LSB
28	MSB	Search Time						
29								LSB
30	Reserved							
31	Search Method (01h)							
32		Search Criteria 01h Descriptor(s)						
n								

The following parameters apply:

- FMKS:

Value	**Description**
0b	stop when a filemark is encountered
1b	filemarks will be ignored (treated as logical blocks with no data - will not match any criteria

- Search Count: The maximum number of records to search. If this field is set to 00000000h, then all records from the current location to EOD are searched (limited to Match Count).
- Match Count: The maximum number of matches to gather in the Match List.

Value	**Description**
0000h	If Match Count is set to 0000h, then the search will stop on the next record (within Search Count) which matches the search criteria. If a matching record is found, the command will return GOOD STATUS and the device will be logically positioned before the matching record (a Read command may be issued to read the matching record). If a matching record is not found before the criteria of the Search command is fulfilled, the command presents NON-DEFERRED CHECK CONDITION status with sense key 0/1401 (No Sense - Record Not Found). If a matching record is not found before Search Time elapses, the command presents NON-DEFERRED CHECK CONDITION status with sense key 0/2E00 (No Sense - Insufficient Time for Operation). Other reportable statuses also apply in conditions where

the operation stops due to other issues before a matching record is found (i.e., filemark is encountered with FMKS set to 0b, EOD encountered, etc.).

Other If Match Count is non-zero, the search will stop once the Match List is full or Search Time elapses. The device logical position may be changed, the command will return good status, and any results must be acquired with Receive Diagnostics Results and/or Read Buffer (Buffer ID 40h).

- Match List Entry Type
- Search Time
- Search Criteria - as per MP 37h[01h]: String Search Criteria (see 6.6.23.2). Minimal descriptor forms are recommended.

6.1.16.2 RcvDiag Data - String Search

The Send Diagnostic method will return results via the Receive Diagnostics Result command. Results will also remain in Read Buffer ID 40h

Table 173 — RcvDiag Command- String Search

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Page Code: 80h							
1	Reserved (00h)							
2	MSB	Page Length (n-3)						
3								LSB
4	MSB	Diagnostic ID (3000h)						
5								LSB
6	Flags (00000000b)							
7	Reserved (00h)							
8	Reserved (00000b)					Blocked	SIM/MIM	Error
9	String Search Results Buffer (Match List)							
n								

String Search Results Buffer (Match List) - for a complete definition of this structure see 5.2.45.1 on page 212.

NOTE 64 - If an error is encountered during a search operation, the Error bit is set to 1b and the SIM/MIM may be set, and a SIM/MIM with additional error information may be presented. These fields should be checked to determine search result validity before string search results are examined

6.1.17 Cryptographic Diags

See “SEND DIAGNOSTIC - 1Dh” on page 203 and “RECEIVE DIAGNOSTIC RESULTS - 1Ch” on page 170 for additional information on the commands. The cryptographic diagnostics are specified in the Encryption version of the SCSI Reference.

6.2 Dynamic runtime attributes (DRA)

6.2.1 Attribute format

Each dynamic runtime attribute shall be communicated between the application client and device server in the format shown in table 174. This format shall be used in the parameter data for the WRITE DYNAMIC RUNTIME ATTRIBUTE command (see 5.2.51) and the READ DYNAMIC RUNTIME ATTRIBUTE command (see 5.2.24). The attribute format implies nothing about the actual representation of the attribute in the drive.

Table 174 — DRA ATTRIBUTE format

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)	I_T NEXUS INDEX						
1								(LSB)
2	(MSB)	ATTRIBUTE IDENTIFIER						
3								(LSB)
4	READ ONLY	Reserved					FORMAT	
5	(MSB)	ATTRIBUTE LENGTH (n-8)						
…								
8								(LSB)
9	ATTRIBUTE VALUE							
…								
n								

The I_T NEXUS INDEX field contains an index associated to the I_T_L nexus by the device server. How the I_T nexus index is maintained is not specified by this standard. The I_T nexus index association may change from one command to another. On a read the value of 0000h in the I_T NEXUS INDEX field indicates that there is no I_T_L nexus associated with the attribute specified by the ATTRIBUTE IDENTIFIER field. The device server shall set the I_T NEXUS INDEX field to 0000h in the logical unit type attributes. On a write the I_T NEXUS INDEX field should be set to 0000h by the application client and the device server shall ignore the value and fill in the I_T NEXUS INDEX field of the stored attribute with the value of the associated I_T_L nexus through which the command arrived. A value of FFFFh is reserved.

The ATTRIBUTE IDENTIFIER field contains a code value identifying the attribute (see 6.2.2).

The READ ONLY bit indicates whether the attribute is in the read only state (see 6.2.1). If an attribute is not in the non-existent state and the READ ONLY bit is set to one, then the attribute is in the Read Only state. If an attribute is not in the non-existent state and the READ ONLY bit is set to zero, then the attribute is in the Read/Write state.

The FORMAT field (see table 175) specifies the format of the data in the ATTRIBUTE VALUE field.

Table 175 — DRA attribute FORMAT field

Format	Name	Description
00b	BINARY	The ATTRIBUTE VALUE field contains binary data.
01b	ASCII	The ATTRIBUTE VALUE field contains left-aligned ASCII data (see 4.4.1).
10b-11b		Reserved

The ATTRIBUTE LENGTH field specifies the length in bytes of the ATTRIBUTE VALUE field. If the ATTRIBUTE LENGTH field is set to zero, then there is no ATTRIBUTE VALUE field.

The ATTRIBUTE VALUE field contains the current value, for the READ DYNAMIC RUNTIME ATTRIBUTE command (see 5.2.24), or intended value, for the WRITE DYNAMIC RUNTIME ATTRIBUTE command (see 5.2.51), of the attribute (see 6.2.2).

6.2.2 Attribute identifier values

6.2.2.1 Attribute identifier values overview

The values in the ATTRIBUTE IDENTIFIER field (see 6.2.1) are assigned according to the attribute type (see 6.2) (see table 176).

Table 176 — DRA attribute identifier range assignments

Attribute Identifiers	Attribute Type	Subclause
0000h to 07FFh	Logical unit	6.2.2.2
1000h to 13FFh	Target	6.2.2.3
1800h to 1BFFh	Initiator	6.2.2.4
others	Reserved	

Devices that support this feature accept and process a WRITE DYNAMIC RUNTIME ATTRIBUTE command containing Initiator type attribute identifier values (i.e., 1800h to 1BFFh) and may be checked as described in 6.2.2.4.

6.2.2.2 Logical unit type attributes

Logical unit type attributes (see table 177) shall be maintained and updated by the device server. All supported logical unit type attributes shall be in the Read Only state (see 6.2).

Table 177 — Logical unit type attributes

Attribute Identifier	Name	Attribute Length (in bytes)	Format	Subclause
0000h	Number of I_T nexuses supported by the Dynamic Runtime Attributes	2	BINARY	6.2.2.2.2
0001h	Timestamp when processed	12	BINARY	6.2.2.2.3
0010h	Reservation Information	V	BINARY	6.2.2.2.4
0011h	Registration Information	V	BINARY	6.2.2.2.5
0012h	Prevent Allow Medium Removal Information	V	BINARY	6.2.2.2.6
0013h	Last failed reservation	V	BINARY	6.2.2.2.7
others	Reserved			
V - Variable				

6.2.2.2.1 I_T_L nexus identifying information descriptor

Attributes may contain one or more I_T_L nexus identifying information descriptors. The format of the I_T_L nexus identifying information descriptor is defined in table 178.

Table 178 — I_T_L nexus identifying information format (part 1 of 2)

Bit Byte	7	6	5	4	3	2	1	0
0	I_T_L NEXUS IDENTIFYING INFORMATION LENGTH (n-3)							
…								
3								

Table 178 — I_T_L nexus identifying information format (part 2 of 2)

Bit Byte	7	6	5	4	3	2	1	0
4	(MSB)	ATTRIBUTE CREATION TIME						
…								
15								(LSB)
16	TARGET TYPE ATTRIBUTES LIST LENGTH (x-19)							
…								
19								
	Target type attributes(s) (see 6.2.2.3)							
20	Target type attributes [first]							
…								
	Target type attributes [last]							
…								
x								
x+1	INITIATOR TYPE ATTRIBUTES LIST LENGTH (n-(x+4))							
…								
x+4								
	Initiator type attributes(s) (see 6.2.2.4)							
x+5	Initiator type attributes [first]							
…								
	Inititator type attributes [last]							
…								
n								

The I_T_L NEXUS IDENTIFYING INFORMATION LENGTH field specifies the amount of data to follow.

The ATTRIBUTE CREATION TIME field has the format of the REPORT TIMESTAMP parameter data format (see 5.2.35.1) and contains the value of the REPORT TIMESTAMP parameter data at the time the attribute is created.

The TARGET TYPE ATTRIBUTES LIST LENGTH field specifies the length of the following target type attributes.

The target type attributes shall be listed in order by I_T NEXUS INDEX and ATTRIBUTE IDENTIFIER. The I_T NEXUS INDEX in this list is from the time the attribute was created and may be a different value than the I_T NEXUS INDEX associated with this I_T nexus at the time the READ DYNAMIC RUNTIME ATTRIBUTE is processed.

The INITIATOR TYPE ATTRIBUTES LIST LENGTH field specifies the length of the following initiator type attributes.

The initiator type attributes shall be listed in order by I_T NEXUS INDEX and ATTRIBUTE IDENTIFIER. The I_T NEXUS INDEX in this list is from the time the attribute was created and may be a different value than the I_T NEXUS INDEX associated with this I_T nexus at the time the READ DYNAMIC RUNTIME ATTRIBUTE is processed.

6.2.2.2.2 Number of I_T nexuses supported by dynamic runtime attributes: Indicates the maximum number of instances of target type attributes and initiator type attributes.

6.2.2.2.3 Timestamp when processed: Timestamp when the READ DYNAMIC RUNTIME ATTRIBUTE command that returns this attribute is processed. The DYNAMIC RUNTIME ATTRIBUTE VALUE field of the timestamp when processed attribute format is the value that would be returned by the REPORT TIMESTAMP parameter data format (see 5.2.35.1).

6.2.2.2.4 Reservation Information attribute: The DYNAMIC RUNTIME ATTRIBUTE VALUE field of the Reservation Information attribute is the list of I_T_L nexus identifying information for each I_T_L nexus that is a reservation holder. The DYNAMIC RUNTIME ATTRIBUTE VALUE field of the Reservation Information attribute is shown in table 179.

Table 179 — Reservation Information Dynamic Runtime Attribute value format

Bit Byte	7	6	5	4	3	2	1	0
0	RESERVATION TYPE							
1	Reserved							
	I_T_L nexus identifying information descriptor(s) (see 6.2.2.2.1)							
2	I_T_L nexus identifying information (see 6.2.2.2.1) (first)							
...								
m								
⋮								
x	I_T_L nexus identifying information (see 6.2.2.2.1) (last)							
...								
y								

The RESERVATION TYPE field shall contain the reservation type as defined in table 180.

Table 180 — RESERVATION TYPE values

Code	Reservation Type
Persistent Reservations are in the range 00h - 0Fh	
00h	Persistent Reserve - Obsolete
01h	Persistent Reserve - Write Exclusive
02h	Persistent Reserve - Obsolete
03h	Persistent Reserve - Exclusive Access
04h	Persistent Reserve - Obsolete
05h	Persistent Reserve - Write Exclusive – Registrants Only
06h	Persistent Reserve - Exclusive Access – Registrants Only
07h	Persistent Reserve - Write Exclusive – All Registrants
08h	Persistent Reserve - Exclusive Access – All Registrants
09h - 0Fh	Persistent Reserve - Reserved
10h	SPC-2 Reserve
11h - FFh	Reserved

Each I_T_L nexus identifying information descriptor is a snapshot of the I_T_L nexus identifying information for an I_T_L nexus when a reservation is created by that I_T_L nexus or when that I_T_L nexus joins the reservation as a reservation holder. The Reservation Information attribute is created and an I_T_L nexus identifying information descriptor is created and added to the list when an I_T_L nexus reserves the logical unit with a PERSISTENT RESERVE OUT command or an SPC-2 RESERVE. Other I_T_L nexus identifying information descriptors are created for each I_T_L nexus that is a reservation holder, if any. This may be due to receipt of a PERSISTENT RESERVE OUT command or due to already registered I_T_L nexuses when an ALL REGISTRANTS type reseration is created. When an I_T_L nexus is no longer a reservation holder either due to the receipt of a PERSISTENT RESERVE OUT command to unregister the I_T_L nexus or the removal of the reservation holder as a side effect to some event that occurs (e.g., PREEMPT) the I_T_L nexus identifying information descriptor related to that I_T_L nexus shall be removed from the list. When the last I_T_L nexus identifying information descriptor has been removed and the reservation is removed the Reservation Information attribute shall be destroyed.

6.2.2.2.5 Registration Information attribute: The DYNAMIC RUNTIME ATTRIBUTE VALUE field of the Registration Information dynamic runtime attribute contains the list of I_T_L nexus identifying information descriptors for each I_T_L nexus that is registered for a persistent reservation. The DYNAMIC RUNTIME ATTRIBUTE VALUE field of the Registration Information dynamic runtime attribute is shown in table 181.

Table 181 — Registration Information Dynamic Runtime Attribute value format

Bit Byte	7	6	5	4	3	2	1	0
0	I_T_L nexus identifying information (see 6.2.2.2.1) (first)							
…								
m								
⋮								
x	I_T_L nexus identifying information (see 6.2.2.2.1) (last)							
…								
y								

Each I_T_L nexus identifying information descriptor is created and added to the list when an I_T_L nexus registers with a PERSISTENT RESERVE OUT command. When an I_T_L nexus is no longer registered either due to a PERSISTENT RESERVE OUT command is recieved to unregister or the registration is removed as a side effect to some event that occurs the I_T_L nexus identifying information descriptor related to that I_T_L nexus shall be removed from the list. When the last I_T_L nexus identifying information descriptor has been removed the Registration Information attribute shall be destroyed.

6.2.2.2.6 Prevent Allow Medium Removal Information attribute: The DYNAMIC RUNTIME ATTRIBUTE VALUE field of the Prevent Allow Medium Removal Information attribute is the list of I_T_L nexus identifying information for each I_T_L nexus that has prevented media removal. The DYNAMIC RUNTIME ATTRIBUTE VALUE field of the Prevent Allow Medium Removal Information attribute is shown in table 182.

Table 182 — Prevent Allow Medium Removal Information Dynamic Runtime Attribute value format

Bit Byte	7	6	5	4	3	2	1	0
0	I_T_L nexus identifying information (see 6.2.2.2.1) (first)							
…								
m								
⋮								
x	I_T_L nexus identifying information (see 6.2.2.2.1) (last)							
…								
y								

Each I_T_L nexus identifying information descriptor is created and added to the list when a volume's removal is prevented due to a PREVENT ALLOW MEDIUM REMOVAL command received from that I_T_L nexus identifying information (i.e., the ATTRIBUTE CREATION TIME field is set to the time the PREVENT ALLOW REMOVAL command with the PREVENT bit set to one is received through that I_T_L nexus.) When an I_T_L nexus no longer prevents medium removal (e.g., a PREVENT ALLOW MEDIUM REMOVAL command with the PREVENT bit set to zero is received through that I_T_L nexus) the I_T_L nexus identifying information descriptor related to that I_T_L nexus shall be removed from the list. When the last I_T_L nexus identifying information descriptor has been removed from the list the Prevent Allow Medium Removal Information attribute shall be destroyed.

6.2.2.2.7 Last failed reservation attribute: Indicates the I_T_L nexus that last failed to get a reservation due to a reservation conflict. The DYNAMIC RUNTIME ATTRIBUTE VALUE field of the Last failed reservation attribute contains the I_T_L nexus identifying information (see 6.2.2.2.1) for the I_T_L nexus that last received a RESERVATION CONFLICT to one of the following commands:

a) PERSISTENT RESERVE OUT;
b) PERSISTENT RESERVE IN;
c) RESERVE (see SPC-2); and
d) RELEASE (see SPC-2).

6.2.2.3 Target type attributes

Target type attributes (see table 183) shall be maintained and updated by the device server. All supported target type attributes shall have a status of read only (see 6.2)..

Table 183 — Target type attributes

Attribute Identifier	Name	Attribute Length (in bytes)	Format	Subclause
1000h	TransportID	24	BINARY	6.2.2.3.1
1001h	Target port ID (relative port identifier)	2	BINARY	6.2.2.3.2
1002h	Last access time	12	BINARY	6.2.2.3.3
1003h to 13FFh	Reserved			
V - Variable				

6.2.2.3.1 TransportID: Indicates the TransportID of the initiator port associated with this I_T_L nexus.

6.2.2.3.2 Target port ID: Indicates the relative target port identifier of the target port associated with this I_T_L nexus.

6.2.2.3.3 Last access time: Time of most recent command that effects the volume received through this I_T_L nexus.This attribute has the format of the REPORT TIMESTAMP parameter data format (see 5.2.35.1) and contains the value of the REPORT TIMESTAMP parameter data at the time the most recent command was received through this I_T_L nexus that is not in the following list:

a) INQUIRY;
b) LOG SENSE;
c) MODE SENSE;
d) READ DYNAMIC RUNTIME ATTRIBUTE;
e) PERSISTENT RESERVE IN;
f) REPORT LUNS;
g) REQUEST SENSE; or
h) TEST UNIT READY.

6.2.2.4 Initiator type attributes

Application clients may use the WRITE DYNAMIC RUNTIME ATTRIBUTE and READ DYNAMIC RUNTIME ATTRIBUTE commands to maintain initiator type attributes. All existent initiator type attributes shall follow the definition specified in table 184. See Dynamic Runtime Information Lifetime (see 4.26.5.1 on page 86) for when attributes are created and destroyed.

Table 184 — Initiator type attributes

Attribute Identifier	Name	Attribute Length (in bytes)	Format	Subclause
1800h	Device special file name (DSFN)	1..32	ASCII	6.2.2.4.1
1801h	Data path failover enabled path	1..4	ASCII	6.2.2.4.2
1802h	Host name (HN)	1..32	ASCII	6.2.2.4.3
1803h	Operating system (OS)	1..16	ASCII	6.2.2.4.4
1804h	Operating system version (OS_V)	1..32	ASCII	6.2.2.4.5
1805h	Device driver name (DD_N)	1..16	ASCII	6.2.2.4.6
1806h	Device driver version (DD_V)	1..16	ASCII	6.2.2.4.7
1807h	Process ID	1..8	ASCII	6.2.2.4.8
1808h to 1BFFh	Reserved			

6.2.2.4.1 Device special file name: Indicates the device special file name used by the application client to identify the I_T_L nexus (e.g., “\\.\tape0”, “/dev/rmt0”, “/dev/sg0”).

6.2.2.4.2 Data path failover enabled path: The path that is enabled for use in the device driver when data path failover (DPF) is enabled and is being used by the thread that issued this command.

6.2.2.4.3 Host name: Indicates the host name of the server that contains the initiator port of the I_T_L nexus (e.g., “foobar”).

6.2.2.4.4 Operating system: Indicates the operating system being used by the application client.

6.2.2.4.5 Operating system version: Indicates the version of the operating system specified in 6.2.2.4.4.

6.2.2.4.6 Device driver name: Indicates the name of the operating system device driver.

6.2.2.4.7 Device driver version: Indicates the version of the operating system device driver specified in 6.2.2.4.6.

6.2.2.4.8 Process ID: The process ID of the thread that is sending commands through this I_T nexus.

6.3 Inquiry Vital Product Data Parameters

Inquiry vital product data parameters are returned to the Inquiry command. INQUIRY - 12h (see 5.2.9) describes how to request these pages.

6.3.1 IP 00h: Supported Inquiry Pages

INQUIRY - 12h (see 5.2.9) describes how to request this page. The following parameters apply to this request:

- EVPD (Enable Vital Product Data): 1b
- Page Code: 00h

6.3.1.1 Returned Data - IP 00h: Supported Inquiry Pages

For a LUN that is associated with an installed device (see 4.2) the following data is returned:

Byte	Description
0	Peripheral Data

Bit	Description
7-5	Peripheral Qualifier: 000b
4-0	Peripheral Device Type: 01h

Byte	Description
1	Page Code: 00h
2	Reserved
3	Page Length (n-3)
4-n	Supported Inquiry Pages: This field is a list of 1-byte long page codes and may include some or all of the following:

Code	Inquiry Page
00h	IP 00h: Supported Inquiry Pages (see 6.3.1 on page 261)
03h	IP 03h: ASCII Information (see 6.3.2 on page 262)
80h	IP 80h: Unit Serial Number (see 6.3.3 on page 263)
83h	IP 83h: Device Identification (see 6.3.4 on page 264)
86h	IP 86h: Extended INQUIRY Data (see 6.3.5 on page 268)
8Fh	IP 8Fh: Third-party Copy (beginning with E08) (see 6.3.6 on page 271)
B0h	IP B0h - Sequential-Access device capabilities (E07 and later) (see 6.3.7 on page 280)
B1h	IP B1h - Manufacturer-assigned Serial Number (see 6.3.8 on page 281)
B5h	IP B5h: Logical Block Protection (see 6.3.9 on page 282)
C0h	IP C0h: Drive Component Revision Levels (see 6.3.10 on page 284)
C1h	IP C1h: Drive Serial Numbers (see 6.3.11 on page 286)
D0h-DFh	Vendor-Reserved (Attachment Specification Information)

6.3.2 IP 03h: ASCII Information

INQUIRY - 12h (see 5.2.9) describes how to request this page. The following parameters apply to this request:

- EVPD (Enable Vital Product Data): 1b
- Page Code: 03h

6.3.2.1 Returned Data - IP 03h: ASCII Information

For a LUN that is associated with an installed device (see 4.2) the following data is returned:

Byte	**Description**
0	Peripheral Data **Bit** **Description** 7-5 Peripheral Qualifier: 000b 4-0 Peripheral Device Type: 01h
1	Page Code: 03h
2	Reserved
3	Page Length: 21h
4	ASCII Length: 00h
5-7	Reserved
8-11	Load ID The Load ID of microcode, represented by eight hex characters, is used to determine if the microcode (i.e., firmware) to be downloaded is compatible with the device electronics (see Annex D.).
12-15	Rev Level The Revision Level of the device microcode, represented with four ASCII characters, is used to determine if the latest level of microcode is downloaded. As each change is implemented, the Rev Level chosen must be numerically larger than any previous Rev Level used, as determined by converting the ASCII characters to numerical format and subtracting. For example, if the old Rev Level is ASCII ‘032C’, and the new Rev Level is ASCII ‘03B1’, that is equivalent to 3033323Ch and 30333B31h, respectively. When the two values are compared numerically, the new Rev Level is greater than the old Rev Level, which satisfies the requirement.
16-19	PTF Number: (unsupported)
21-23	Patch Number: (unsupported)
24-31	RU Name Used by the attaching system. This is an 8-byte ASCII field that may be used to determine if the microcode (i.e., firmware) to be downloaded is compatible with the device electronics (see Annex D.).
32-36	Library Sequence Number (in ASCII). This field applies to the IBM 3494 Library only. This field matches the information returned in RS/422 initialization response, bytes 38-42.

6.3.3 IP 80h: Unit Serial Number

INQUIRY - 12h (see 5.2.9) describes how to request this page. The following parameters apply to this request:

- EVPD (Enable Vital Product Data): 1b
- Page Code: 80h

6.3.3.1 Returned Data - IP 80h: Unit Serial Number

For a LUN that is associated with an installed device (see 4.2) the following data is returned:

Byte	Description
0	Peripheral Data **Bit** **Description** 7-5 Peripheral Qualifier: 000b 4-0 Peripheral Device Type: 01h
1	Page Code: 80h
2	Reserved
3	Page Length: 0Ch
4-15	Serial Number of device, right-justified with leading zeroes, in ASCII (same as Inquiry Standard Data bytes 38-49)

6.3.4 IP 83h: Device Identification

INQUIRY - 12h (see 5.2.9) describes how to request this page. The following parameters apply to this request:

- EVPD (Enable Vital Product Data): 1b
- Page Code: 83h

6.3.4.1 Returned Data on Fibre Channel devices - IP 83h: Device Identification

For a LUN that is associated with an installed device (see 4.2) the following data is returned:

Byte	**Description**	
0	Peripheral Data	
	Bit	**Description**
	7-5	Peripheral Qualifier: 000b
	4-0	Peripheral Device Type: 01h
1	Page Code: 83h	
2	Reserved	
3	Page Length: 5Ch	
4	Device ID (T10 vendor identification)	
	Bit	**Description**
	7-4	Reserved
	3-0	Code Set: 2h (Identifier is all ASCII)
5		
	Bit	**Description**
	7-6	Reserved
	5-4	Association: 00b
	3-0	Identifier Type: 1h
6	Reserved	
7	Identifier Length: 24h	
8-15	Vendor ID (same as Inquiry Standard Data bytes 8-15)	
16-31	Device Type and Model Number (same as Inquiry Standard Data bytes 16-31)	
32-43	Serial Number of device (same as Inquiry Standard Data bytes 38-49)	
44	World Wide Node Name (NAA)	
	Bit	**Description**
	7-4	Reserved
	3-0	Code Set: 1h
45		
	Bit	**Description**
	7-6	Reserved
	5-4	Association Type: 00b
	3-0	Identifier Type: 3h
46	Reserved	
47	Identifier Length: 8h	
48-55	World Wide Node Name	
56	World Wide Port Name (NAA)	
	Bit	**Description**
	7-4	Reserved
	3-0	Code Set: 1h

57

Bit	**Description**
7-6	Reserved
5-4	Association Type: 01b
3-0	Identifier Type: 3h

58 Reserved

59 Identifier Length: 8h

60-67 World Wide Port Name

NOTE 65 - This relates to the port on which the Inquiry command was received.

68 Port Identifier (Relative target port)

Bit	**Description**
7-4	Reserved
3-0	Code Set: 1h

69

Bit	**Description**
7-6	Reserved
5-4	Association Type: 01b
3-0	Identifier Type: 4h

70 Reserved

71 Identifier Length: 4h

72-75 Port Identifier

Value	**Description**
00000001h	Primary interface port 0
00000002h	Primary interface port 1

NOTE 66 - This relates to the port on which the Inquiry command was received.

76 Alternate World Wide Port Name (NAA)

Bit	**Description**
7-4	Reserved
3-0	Code Set: 1h

77

Bit	**Description**
7-6	Reserved
5-4	Association Type: 10b
3-0	Identifier Type: 3h

78 Reserved

79 Identifier Length: 8h

80-87 World Wide Port Name

NOTE 67 - This relates to a different port than the port on which the Inquiry command was received.

88 Alternate Port Identifier (Relative target port)

Bit	**Description**
7-4	Reserved
3-0	Code Set: 1h

89

Bit	**Description**
7-6	Reserved
5-4	Association Type: 10b
3-0	Identifier Type: 4h

90 Reserved

91 Identifier Length: 4h

92-95 Port Identifier

Value	**Description**
00000001h	Primary interface port 0
00000002h	Primary interface port 1

NOTE 68 - This relates to a different port than the port on which the Inquiry command was received.

6.3.4.2 Returned Data on Ethernet devices - IP 83h: Device Identification

For a LUN that is associated with an installed device (see 4.2) the following data is returned:

Byte **Description**

0 Peripheral Data

Bit	**Description**
7-5	Peripheral Qualifier: 000b
4-0	Peripheral Device Type: 01h

1 Page Code: 83h

2 Reserved

3 Page Length: 7Ch

4 Device ID (T10 vendor identification)

Bit	**Description**
7-4	Reserved
3-0	Code Set: 2h (Identifier is all ASCII)

5

Bit	**Description**
7-6	Reserved
5-4	Association: 00b
3-0	Identifier Type: 1h

6 Reserved

7 Identifier Length: 24h

8-15 Vendor ID (same as Inquiry Standard Data bytes 8-15)

16-31 Device Type and Model Number (same as Inquiry Standard Data bytes 16-31)

32-43 Serial Number of device (same as Inquiry Standard Data bytes 38-49)

44 World Wide Node Name (NAA)

Bit	**Description**
7-4	Reserved
3-0	Code Set: 1h

45

Bit	**Description**
7-6	Reserved
5-4	Association Type: 00b
3-0	Identifier Type: 3h

46 Reserved

47 Identifier Length: 8h

48-55 World Wide Node Name

56 World Wide Port Name (NAA)

Bit	**Description**
7-4	Reserved
3-0	Code Set: 1h

57

Bit	Description
7-6	Reserved
5-4	Association Type: 01b
3-0	Identifier Type: 4h

58 Reserved

59 Identifier Length: 4h

60-63 Port Identifier

Value	Description
00000001h	Primary interface port 0
00000002h	Primary interface port 1

NOTE 69 - This relates to the port on which the Inquiry command was received.

64 Internet SCSI (iSCSI) Target Device Identifier

Bit	Description
7-4	PROTOCOL IDENTIFIER: 5h
3-0	Code Set: 3h

65

Bit	Description
7	PIV: 1b
6	Reserved
5-4	Association Type: 10b
3-0	Identifier Type: 8h

66 Reserved

67 Identifier Length: 18h

68-91 SCSI name string naa identifier

92 Internet SCSI (iSCSI) Target Port Identifier

Bit	Description
7-4	PROTOCOL IDENTIFIER: 5h
3-0	Code Set: 3h

93

Bit	Description
7	PIV: 1b
6	Reserved
5-4	Association Type: 01b
3-0	Identifier Type: 8h

94 Reserved

95 Identifier Length: 20h

96-128 SCSI name string naa identifier

NOTE 70 - This relates to the port on which the Inquiry command was received.

6.3.5 IP 86h: Extended INQUIRY Data

The Extended INQUIRY Data VPD page (see table 185) provides the application client with a means to obtain information about the logical unit.

Table 185 — Extended INQUIRY Data VPD page

Bit Byte	7	6	5	4	3	2	1	0
0	PERIPHERAL QUALIFIER (000B)			PERIPHERAL DEVICE TYPE (01H)				
1	PAGE CODE (86h)							
2	Reserved							
3	PAGE LENGTH (3Ch)							
4	ACTIVATE MICROCODE		SPT			GRD_CHK (0B)	APP_CHK (0B)	REF_CHK (0B)
5	Reserved		UASK_SUP (0B)	GROUP_SUP (0B)	PRIOR_SUP (0B)	HEADSUP (0B)	ORDSUP (0B)	SIMPSUP (1B)
6	Reserved				WU_SUP (0B)	CRD_SUP (0B)	NV_SUP (0B)	V_SUP (0B)
7	Reserved			P_I_I_SUP (0B)	Reserved			LUICLR (0B)
8	Reserved			R_SUP (0B)	Reserved			CBCS (0B)
9	Reserved				MULTI I_T NEXUS MICROCODE DOWNLOAD			
10	(MSB)	EXTENDED SELF-TEST COMPLETION MINUTES						
11								(LSB)
12	POA_SUP	HOA_SUP	VSA_SUP	Reserved				
13	MAXIMUM SUPPORTED SENSE DATA LENGTH							
14 63	Reserved							

The following data is returned.

Byte	Description
0	

Bit	Description
7-5	PERIPHERAL QUALIFIER: 000b
4-0	PERIPHERAL DEVICE TYPE: 01h (Sequential Access Device)

1	PAGE CODE (86h)
2	Reserved
3	PAGE LENGTH (3Ch)

4

<table>
<tr><th>Bit</th><th>Description</th></tr>
<tr><td>7-6</td><td>ACTIVATE MICROCODE (00B)

The ACTIVATE MICROCODE field indicates how the device server activates microcode and establishes a unit attention condition when a WRITE BUFFER command (see 5.2.50) with the download microcode mode set to 05h or 07h is processed.

<table>
<tr><th>Value</th><th>Description</th></tr>
<tr><td>00b</td><td>The actions of the device server may or may not be as defined for values 01b or 10b.</td></tr>
<tr><td>01b-10b</td><td>Not supported</td></tr>
<tr><td>11b</td><td>Reserved</td></tr>
</table></td></tr>
<tr><td>5-3</td><td>SPT

A supported protection type (SPT) field indicates the type of protection the logical unit supports. The SPT field is reserved if the PROTECT bit (see 5.2.9.1) is set to zero.

<table>
<tr><th>Value</th><th>Protection type supported</th></tr>
<tr><td>001b</td><td>Logical block protection</td></tr>
<tr><td>others</td><td>Reserved</td></tr>
</table></td></tr>
<tr><td>2</td><td>GRD_CHK (0b)</td></tr>
<tr><td>1</td><td>APP_CHK (0b)</td></tr>
<tr><td>0</td><td>REF_CHK (0b)</td></tr>
</table>

5

Bit	Description
7-6	Reserved
5	UASK_SUP (0b)
4	GROUP_SUP (0b)
3	PRIOR_SUP (0b)
2	HEADSUP (0b)
1	ORDSUP (0b)
0	SIMPSUP (1b) - The device server supports simple queing.

6

Bit	Description
7-4	Reserved
3	WU_SUP (0b)
2	CRD_SUP (0b)
1	NV_SUP (0b)
0	V_SUP (0b)

7

Bit	Description
7-4	Reserved
3	P_I_I_SUP (0b)
2-1	Reserved
0	LUICLR (0b)

8

Bit	Description
7-4	Reserved
3	R_SUP (0b)
2-1	Reserved
0	CBCS (0b)

9

Bit	Description
7-4	Reserved
3-0	MULTI I_T NEXUS MICROCODE DOWNLOAD (0B)

The MULTI I_T NEXUS MICROCODE DOWNLOAD field indicates how the device server handles concurrent attempts to download microcode using the WRITE BUFFER command (see 5.2.38) from multiple I_T nexuses.

Value	Description
0h	The handling of concurrent WRITE BUFFER download microcode operations from multiple I_T nexus is vendor specific.
1h-3h	Not Supported
4h-Fh	Reserved.

10-11 EXTENDED SELF-TEST COMPLETION MINUTES (0000h): Not supported.

12

Bit	Description
7	POA_SUP (Power On Activate Supported)(1b): A WRITE BUFFER command with the MODE field set to 0Dh (see 5.2.50.7) and the PO_ACT bit set to one is supported.
6	HRA_SUP (Hard Reset Activate Supported)(0b): A WRITE BUFFER command with the MODE field set to 0Dh (see 5.2.50.7) and the HR_ACT bit set to one is not supported.
5	VSA_SUP (Vendor Specific Event Activate Supported)(1b): A WRITE BUFFER command with the MODE field set to 0Dh (see 5.2.50.7) and the VSE_ACT bit set to one is supported. The vendor specific event is the detection of medium removal.
4-0	Reserved.

13 MAXIMUM SUPPORTED SENSE DATA LENGTH (0000h): Not supported.

14-63 Reserved

6.3.6 IP 8Fh: Third-party Copy (beginning with E08)

6.3.6.1 Third-party Copy VPD page overview

The Third-party Copy VPD page (see table 186) provides a means to retrieve descriptors that indicate the capabilities supported by the copy manager (see 4.28.1).

Table 186 — Third-party Copy VPD page

Byte	**Bit**							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	PERIPHERAL QUALIFIER			PERIPHERAL DEVICE TYPE				
1	PAGE CODE (8Fh)							
2	(MSB)	PAGE LENGTH (n-3)						
3								(LSB)
	Third-party copy descriptors							
4	Third-party copy descriptor [first]							
…								
	⋮							
	Third-party copy descriptor [last]							
…								
n								

Byte	**Description**
0	Peripheral Data **Bit** **Description** 7-5 PERIPHERAL QUALIFIER: 000b 4-0 PERIPHERAL DEVICE TYPE: 01h
1	Page Code: 8Fh
2-3	Page Length: Number of bytes that follow.
4-n	Third-party copy descriptors: These are returned in ascending order based on third-party copy descriptor type values (see 6.3.6.2).

6.3.6.2 Third-party copy descriptor format

Each third-party copy descriptor has the format shown in table 187.

Table 187 — Third-party copy descriptor format

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)	THIRD-PARTY COPY DESCRIPTOR TYPE						
1								(LSB)
2	(MSB)	THIRD-PARTY COPY DESCRIPTOR LENGTH (n-3)						
3								(LSB)
4	Third-party copy parameters							
…								
n								

Byte	Description
0-1	THIRD-PARTY COPY DESCRIPTOR TYPE: Indicates which third-party copy descriptor is being returned.

Value	Description
0001h	Supported Commands third-party copy descriptor (see 6.3.6.3 on page 273)
0004h	Parameter Data third-party copy descriptor (see 6.3.6.4 on page 275)
0008h	Supported Descriptors third-party copy descriptor (see 6.3.6.5 on page 276)
000Ch	Supported CSCD IDs third-party copy descriptor (see 6.3.6.6 on page 277)
8001h	General Copy Operations third-party copy descriptor (see 6.3.6.7 on page 278)
9101h	Stream Copy Operations third-party copy descriptor (see 6.3.6.8 on page 279)

Byte	Description
2-3	THIRD-PARTY COPY DESCRIPTOR LENGTH: The number of bytes of third-party copy parameters that follow.
4-n	Third-party copy parameters contents are indicated by the contents of the THIRD-PARTY COPY DESCRIPTOR TYPE field.

6.3.6.3 Supported Commands third-party copy descriptor

6.3.6.3.1 Supported Commands third-party copy descriptor overview

The Supported Commands third-party copy descriptor (see table 188) indicates which combinations of operation code and service action the copy manager supports. The information provided by the Supported Commands third-party copy descriptor is equivalent to information returned by the REPORT SUPPORTED OPERATION CODES command (see 5.2.34).

Table 188 — Supported Commands third-party copy descriptor format

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)			THIRD-PARTY COPY DESCRIPTOR TYPE (0001h)				
1								(LSB)
2	(MSB)			THIRD-PARTY COPY DESCRIPTOR LENGTH (n-3)				
3								(LSB)
4				COMMANDS SUPPORTED LIST LENGTH (m-4)				
				Commands supported list				
5				Command support descriptor (see table 189) [first]				
…								
				⋮				
				Command support descriptor (see table 189) [last]				
…								
m								
m+1				DESCRIPTOR PAD (if needed)				
…								
n								

Byte	Description
0-1	THIRD-PARTY COPY DESCRIPTOR TYPE: 0001h.
2-3	THIRD-PARTY COPY DESCRIPTOR LENGTH: The number of bytes of third-party copy parameters that follow.
4	COMMANDS SUPPORTED LIST LENGTH: The length, in bytes, of the list of supported commands that follow.
5-m	List of command support descriptors: Each command support descriptor (see 6.3.6.3.2) lists the service actions that the copy manager supports for a specific operation code.
m+1 to n	DESCRIPTOR PAD: zero to three bytes set to zero such that the total length of the Supported Commands third-party copy descriptor is a multiple of four.

6.3.6.3.2 Command support descriptor format

Each command support descriptor (see table 189) indicates an operation code that the copy manager supports and the services actions for that operation code that are supported.

Table 189 — Command support descriptor format

Bit Byte	7	6	5	4	3	2	1	0
0	SUPPORTED OPERATION CODE							
1	SUPPORTED SERVICE ACTIONS LIST LENGTH (m-1)							
2	List of supported service actions							
…								
m								

Byte	Description
0	SUPPORTED OPERATION CODE: The operation code for which a list of supported service actions is being returned.
1	SUPPORTED SERVICE ACTIONS LIST LENGTH: The number of supported service action values that follow.
2-m	List of supported service actions: One byte for each service action of the operation code indicated by the SUPPORTED OPERATION CODE field that the copy manager supports, with a unique supported service action in each byte. The service actions appear in the list in ascending numerical order.

Table 190 contains the list of command support descriptors at the time of publication.

Table 190 — List of command support descriptors (at publication)

Op Code	Service Action	Command
83h	01h	EXTENDED COPY (LID4) (i.e., XCOPY) – 83h[01h] (see 5.2.5 on page 110)
83h	1Ch	COPY OPERATION ABORT – 83h[1Ch] (see 5.2.3 on page 107)
84h	05h	RECEIVE COPY STATUS(LID4) – 84h[05h] (see 5.2.27 on page 167)

6.3.6.4 Parameter Data third-party copy descriptor

The Parameter Data third-party copy descriptor (see table 191) indicates the limits that the copy manager places on the contents of the XCOPY command parameter data.

Table 191 — Parameter Data third-party copy descriptor format

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)	THIRD-PARTY COPY DESCRIPTOR TYPE (0004h)						
1								(LSB)
2	(MSB)	THIRD-PARTY COPY DESCRIPTOR LENGTH (001Ch)						
3								(LSB)
4		Reserved						
…								
7								
8	(MSB)	MAXIMUM CSCD DESCRIPTOR COUNT (0001h)						
9								(LSB)
10	(MSB)	MAXIMUM SEGMENT DESCRIPTOR COUNT (1000h)						
11								(LSB)
12	(MSB)	MAXIMUM DESCRIPTOR LIST LENGTH (0001 0000h)						
…								
15								(LSB)
16	(MSB)	MAXIMUM INLINE DATA LENGTH (0000 0000h)						
…								
19								(LSB)
20		Reserved						
…								
31								

Byte	**Description**
0-1	THIRD-PARTY COPY DESCRIPTOR TYPE: 0004h
2-3	THIRD-PARTY COPY DESCRIPTOR LENGTH: 001Ch
4-7	Reserved
8-9	MAXIMUM CSCD DESCRIPTOR COUNT: 0001h
10-11	MAXIMUM SEGMENT DESCRIPTOR COUNT: The maximum number of segment descriptors that the copy manager allows in an XCOPY parameter list (see 4.28.4.1).
12-15	MAXIMUM DESCRIPTOR LIST LENGTH: The maximum length, in bytes, of the CSCD descriptor list and segment descriptor list that the copy manager allows in an XCOPY command parameter list (see 4.28.4.1).
16-19	MAXIMUM INLINE DATA LENGTH: 0000 0000h
20-31	Reserved

6.3.6.5 Supported Descriptors third-party copy descriptor

The Supported Descriptors third-party copy descriptor (see table 192) indicates which CSCD descriptors (see 6.9.2.1.1.1) and segment descriptors (see 6.9.2.1.2.1) the copy manager supports.

Table 192 — Supported Descriptors third-party copy descriptor format

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)	THIRD-PARTY COPY DESCRIPTOR TYPE (0008h)						
1								(LSB)
2	(MSB)	THIRD-PARTY COPY DESCRIPTOR LENGTH (n-3)						
3								(LSB)
4	SUPPORTED DESCRIPTOR LIST LENGTH (m-4)							
5	List of supported descriptor type codes							
…								
m								
m+1	DESCRIPTOR PAD (if needed)							
…								
n								

Byte Description

0-1 THIRD-PARTY COPY DESCRIPTOR TYPE: 0008h

2-3 THIRD-PARTY COPY DESCRIPTOR LENGTH: The number of bytes that follows.

4 SUPPORTED DESCRIPTOR LIST LENGTH: The length, in bytes, of the list of supported descriptor type codes that follows.

5-m List of supported descriptor type codes: One byte for each CSCD descriptor and segment descriptor DESCRIPTOR TYPE CODE value (see 6.9.2.1) supported by the copy manager, with a unique supported DESCRIPTOR TYPE CODE value in each byte. The descriptor type code values appear in the list in ascending numerical order.

The list at publication time follows.

Value	Description
07h	ECD 07h: Verify CSCD function (see 6.9.2.1.2.2 on page 504)
10h	ECD 10h: Write filemarks function (see 6.9.2.1.2.3 on page 505)
11h	ECD 11h: Space function (see 6.9.2.1.2.4 on page 506)
12h	ECD 12h: Locate function (see 6.9.2.1.2.5 on page 507)
13h	ECD 13h: Tape device image copy function (see 6.9.2.1.2.6 on page 507)
14h	ECD 14h: Register persistent reservation key function (see 6.9.2.1.2.7 on page 508)
15h	ECD 15h: Third party persistent reservations source I_T nexus function (see 6.9.2.1.2.8 on page 509)
17h	ECD 17h: Positioning function (see 6.9.2.1.2.9 on page 511)
18h	ECD 18h: Tape device logical object copy function (see 6.9.2.1.2.10 on page 513)
E0h	ECD E0h: Fibre Channel N_Port_Name CSCD descriptor format (see 6.9.2.1.1.2 on page 499)
E2h	ECD E2h: Fibre Channel N_Port_ID With N_Port_Name Checking CSCD descriptor format (see 6.9.2.1.1.3 on page 500)
E4h	ECD E4h: Identification Descriptor CSCD descriptor format (see 6.9.2.1.1.4 on page 501)

m+1 to n DESCRIPTOR PAD: Zero to three bytes set to zero such that the total length of the Supported Descriptors third-party copy descriptor is a multiple of four.

6.3.6.6 Supported CSCD IDs third-party copy descriptor

The Supported CSCD IDs third-party copy descriptor (see table 193) indicates which CSCD IDs (see 6.9.2.1.1.1) other than 0000h to 07FFh the copy manager supports.

Table 193 — Supported CSCD IDs third-party copy descriptor format

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)	THIRD-PARTY COPY DESCRIPTOR TYPE (000Ch)						
1								(LSB)
2	(MSB)	THIRD-PARTY COPY DESCRIPTOR LENGTH (n-3)						
3								(LSB)
4	(MSB)	SUPPORTED CSCD IDS LIST LENGTH (m-5)						
5								(LSB)
	Supported CSCD ID list							
6	(MSB)	SUPPORTED CSCD ID [first]						
7								(LSB)
m-1	(MSB)	SUPPORTED CSCD ID [last]						
m								(LSB)
m+1	DESCRIPTOR PAD (if needed)							
…								
n								

Byte Description

0-1 THIRD-PARTY COPY DESCRIPTOR TYPE: 000Ch

2-3 THIRD-PARTY COPY DESCRIPTOR LENGTH: The number of bytes that follows.

4-5 SUPPORTED CSCD IDS LIST LENGTH: The length, in bytes, of the list of supported CSCD IDs that follows.

6-m Supported CSCD ID list: Each SUPPORTED CSCD ID field indicates one unique CSCD ID with a value greater than 07FFh that is supported by the copy manager. The CSCD IDs appear in the list in ascending numerical order.

The supported CSCD descriptor ID list at publication follows:

Value	Description
FFFFh	The copy source or copy destination is the logical unit that contains the copy manager that is processing the XCOPY command

m+1 to n DESCRIPTOR PAD: Zero to three bytes set to zero such that the total length of the Supported CSCD IDs Descriptors third-party copy descriptor is a multiple of four.

6.3.6.7 General Copy Operations third-party copy descriptor

The General Copy Operations third-party copy descriptor (see table 194) indicates the limits that the copy manager places on processing of copy operations (see 4.28.4.2.1).

Table 194 — General Copy Operations third-party copy descriptor format

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)	THIRD-PARTY COPY DESCRIPTOR TYPE (8001h)						
1								(LSB)
2	(MSB)	THIRD-PARTY COPY DESCRIPTOR LENGTH (0020h)						
3								(LSB)
4	(MSB)	TOTAL CONCURRENT COPIES (0000 0001h)						
…								
7								(LSB)
8	(MSB)	MAXIMUM IDENTIFIED CONCURRENT COPIES (0000 0001h)						
…								
11								(LSB)
12	(MSB)	MAXIMUM SEGMENT LENGTH (0000 0000h)						
15								(LSB)
16	DATA SEGMENT GRANULARITY (log 2) (00h)							
17	INLINE DATA GRANULARITY (log 2) (00h)							
18	Reserved							
…								
35								

Byte	Description
0-1	THIRD-PARTY COPY DESCRIPTOR TYPE: 8001h
2-3	THIRD-PARTY COPY DESCRIPTOR LENGTH: 0020h
4-7	TOTAL CONCURRENT COPIES: 0000 0001h The maximum number of third-party copy commands that are supported for concurrent processing by the copy manager.
8-11	MAXIMUM IDENTIFIED CONCURRENT COPIES: 0000 0001h The maximum number of third-party copy commands that are not an XCOPY command with the LIST ID USAGE field set to 11b that are supported for concurrent processing by the copy manager.
12-15	MAXIMUM SEGMENT LENGTH: 0000 0000h
16	DATA SEGMENT GRANULARITY: 00h
17	INLINE DATA GRANULARITY: 00h
18-35	Reserved

6.3.6.8 Stream Copy Operations third-party copy descriptor

The Stream Copy Operations third-party copy descriptor (see table 195) indicates the limits that the copy manager places on processing of copy operations (see 4.28.4.2.1).

Table 195 — Stream Copy Operations third-party copy descriptor format

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)	THIRD-PARTY COPY DESCRIPTOR TYPE (9101h)						
1								(LSB)
2	(MSB)	THIRD-PARTY COPY DESCRIPTOR LENGTH (000Ch)						
3								(LSB)
4	(MSB)	MAXIMUM STREAM DEVICE TRANSFER SIZE (0020 0000h)						
…								
7								(LSB)
8	Reserved							
…								
15								

Byte	Description
0-1	THIRD-PARTY COPY DESCRIPTOR TYPE: 9101h
2-3	THIRD-PARTY COPY DESCRIPTOR LENGTH: 000Ch
4-7	MAXIMUM STREAM DEVICE TRANSFER SIZE: 0020 0000h The maximum transfer size, in bytes, supported.
8-15	Reserved

6.3.7 IP B0h - Sequential-Access device capabilities (E07 and later)

INQUIRY - 12h (see 5.2.9 on page 117) describes how to request this page. The following parameters apply to this request:

- EVPD (Enable Vital Product Data): 1b
- Page Code: B0h

6.3.7.1 Returned Data - IP B0h: Sequential-Access device capabilities

For LUN 0, the following data is returned:This page provides the application client with the means to determine if the features specified in this page are supported by the drive.

Table 196 — Table 23. Sequential-Access Device Capabilities Page

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Peripheral Qualifier (000b)			Peripheral Device Type (01h)				
1	Page Code (B0h)							
2	Page Length (2)							
3								
4	Reserved							WORM
5	Reserved							

If the write once read many (WORM) bit is set to one, the device server supports WORM mode operation (see 4.2.24.3). If the WORM bit is set to zero, the device server does not support WORM mode operation.

6.3.8 IP B1h - Manufacturer-assigned Serial Number

INQUIRY - 12h (see 5.2.9 on page 117) describes how to request this page. The following parameters apply to this request:

- EVPD (Enable Vital Product Data): 1b
- Page Code: B1h

6.3.8.1 Returned Data - IP B1h: Manufacturer-assigned Serial Number

Table 197 specifies the Manufacturer-assigned Serial Number VPD page.

Table 197 — Manufacturer-assigned Serial Number VPD page

Bit Byte	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	PERIPHERAL QUALIFIER (000b)			PERIPHERAL DEVICE TYPE (01h)				
1	PAGE CODE (B1h)							
2	(MSB)	PAGE LENGTH (000Ch)						
3								(LSB)
4	(MSB)	MANUFACTURER-ASSIGNED SERIAL NUMBER						
15								(LSB)

Byte	**Description**
0	Peripheral Data

Bit	**Description**
7-5	PERIPHERAL QUALIFIER: 000b
4-0	PERIPHERAL DEVICE TYPE: 01h

1	PAGE CODE: B1h
2-3	PAGE LENGTH: 0Ch
4-15	MANUFACTURER-ASSIGNED SERIAL NUMBER: Right-aligned ASCII data that is the manufacturer-assigned serial number. If the manufacturer-assigned serial number is not available, the device returns ASCII spaces (20h) in this field. If the manufacturer-assigned serial number differs from the value in the UNIT SERIAL NUMBER field of IP 80h: Unit Serial Number (see 6.2.3), then the value in the UNIT SERIAL NUMBER field is used in building the T10 vendor ID descriptor.

6.3.9 IP B5h: Logical Block Protection

INQUIRY - 12h (see 5.2.4 on page 81) describes how to request this page. The following parameters apply to this request:

- EVPD (Enable Vital Product Data): 1b
- Page Code: B5h

6.3.9.1 Returned Data - IP B5h: Logical Block Protection

Table 198 specifies the Logical Block Protection VPD page.

Table 198 — IP 85h Logical Block Protection VPD page

Bit Byte	7	6	5	4	3	2	1	0
0	PERIPHERAL QUALIFIER (000b)			PERIPHERAL DEVICE TYPE (01h)				
1	PAGE CODE (B5h)							
2	(MSB)	PAGE LENGTH (n-3)						
3								(LSB)
4	Reserved							
7								
Logical block protection method descriptor list								
	Logical block protection method descriptor for LBP disabled (E07+)							
8	LOGICAL BLOCK PROTECTION METHOD DESCRIPTOR LENGTH (07h)							
9	LOGICAL BLOCK PROTECTION METHOD (00h)							
10	Reserved		LOGICAL BLOCK PROTECTION INFORMATION LENGTH (00h)					
11	LBP_W_C (0b)	LBP_R_C (0b)	RBDP_C (0b)	Reserved				
12	Reserved							
15								
Logical block protection method descriptor for RS-CRC (E07+)								
16	LOGICAL BLOCK PROTECTION METHOD DESCRIPTOR LENGTH (07h)							
17	LOGICAL BLOCK PROTECTION METHOD (01h)							
18	Reserved		LOGICAL BLOCK PROTECTION INFORMATION LENGTH (04h)					
19	LBP_W_C (1b)	LBP_R_C (1b)	RBDP_C (1b)	Reserved				
20	Reserved							
23								
Logical block protection method descriptor for CRC32C (E08+)								
24	LOGICAL BLOCK PROTECTION METHOD DESCRIPTOR LENGTH (07h)							
25	LOGICAL BLOCK PROTECTION METHOD (02h)							
26	Reserved		LOGICAL BLOCK PROTECTION INFORMATION LENGTH (04h)					
27	LBP_W_C (1b)	LBP_R_C (1b)	RBDP_C (1b)	Reserved				
28	Reserved							
31								

Byte	Description
0	Peripheral Data

Bit	Description
7-5	PERIPHERAL QUALIFIER: 000b
4-0	PERIPHERAL DEVICE TYPE: 01h

1	PAGE CODE: B5h
2-3	PAGE LENGTH: n-3
4-7	Reserved
8-n	The logical block protection method descriptor list contains a list of descriptors which describe each supported logical block protection method.The descriptors are returned in ascending order by logical block protection method code starting with logical block protection method zero (i.e., LBP not enabled). The list of logical block protection method descriptors depends on the product.

For x=1 to the number of descriptors:

Byte	Description
8x+0	LOGICAL BLOCK PROTECTION METHOD DESCRIPTOR LENGTH: 07h
8x+1	LOGICAL BLOCK PROTECTION METHOD: (see 6.6.9)
8x+2	

Bit	Description
7-6	Reserved
5-0	LOGICAL BLOCK PROTECTION INFORMATION LENGTH:

8x+3	

Bit	Description
7	LBP_W_C (logical blocks protected during write capable):
6	LBP_R_C (logical block protected during read capable):
5	RBDP_C (recover buffered data protected capable):
4-0	Reserved

(8x+4) to (8x+7)	Reserved

6.3.10 IP C0h: Drive Component Revision Levels

INQUIRY - 12h (see 5.2.9) describes how to request this page. The following parameters apply to this request:

- EVPD (Enable Vital Product Data): 1b
- Page Code: C0h

6.3.10.1 Returned Data - IP C0h: Drive Component Revision Levels

For LUN 0, the following data is returned:

Table 199 — IP C0h: Drive Component Revision Levels

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	PERIPHERAL QUALIFIER (000B)			PERIPHERAL DEVICE TYPE (01h)				
1	PAGE CODE (C0H)							
2	Reserved							
3	PAGE LENGTH (27h)							
4	CODE NAME							
15								
16	TIME(HHMMSS)							
22								
23	DATE (YYYYMMDD)							
30								
31	PLATFORM							
42								

Byte	Description
0	Peripheral Data **Bit** **Description** 7-5 PERIPHERAL QUALIFIER: 000b 4-0 PERIPHERAL DEVICE TYPE: 01h
1	PAGE CODE: C0h
2	Reserved
3	PAGE LENGTH: 27h
4-15	CODE NAME - The code name definition is not published.
16-22	TIME - Time the code was built in HHMMSS format with a trailing NULL (i.e., 00h). Prior to May 2010, this field was set to ASCII zeros (i.e., 30h)
23-30	DATE - The date the code was built.

31-42 PLATFORM - ASCII characters containing the <protocol>_<package>[_<variant>] for the drive. This is left-aligned with ASCII spaces padded at the end.

Table 200 — PLATFORM definition

Symbol	Description	
<protocol>	**Value**	**Transport Protocol**
	fcp	Fibre Channel (FC)
	fca	Fibre Channel (FC)
	ena	Ethernet (RoCEv2)
<package>	**Value**	**Type**
	fj	3592
<variant>	**Value**	**Type**
	f	FIPS

6.3.11 IP C1h: Drive Serial Numbers

INQUIRY - 12h (see 5.2.9) describes how to request this page. The following parameters apply to this request:

- EVPD (Enable Vital Product Data): 1b
- Page Code: C1h

6.3.11.1 Returned Data - IP C1h: Drive Serial Numbers

For LUN 0, the following data is returned:

WARNING WARNING

The length of this page should be dynamically parsed. The size of this page has increased in the past and may increase in the future (i.e., additional fields may be added).

WARNING WARNING

Table 201 — IP C1h: Drive Serial Numbers

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Peripheral Qualifier (000b)			Peripheral Device Type (01h)				
1	Page Code (C1h)							
2	Reserved							
3	Page Length (24h)							
4	Manufacturing Full Assembly Serial Number							
15								
16	Reported Full Assembly Serial Number							
27								
28	Manufacturing Drive Brick Serial Number							
39								

Byte	**Description**
0	Peripheral Data **Bit** **Description** 7-5 Peripheral Qualifier: 000b 4-0 Peripheral Device Type: 01h
1	Page Code: C1h
2	Reserved
3	Page Length: 24h
4-15	Manufacturing Full Assembly Serial Number: Full Assembly Serial Number set at time of manufacture, right-justified with leading zeros, in ASCII. In earlier code levels this field presented the Manufacturing Drive Brick Serial Number (i.e., information in bytes 28-39) on LUN0 and the Manufacturing Full Assembly Serial Number on LUN2 (i.e., to the medium changer).
16-27	Reported Full Assembly Serial Number: Full Assembly Serial Number used as the serial number in the Unit Serial Number Inquiry page (i.e., page 80h), right-justified with leading zeros, in ASCII. This value may be over-ridden by a library or FRU process.
28-39	Manufacturing Drive Brick Serial Number: This is the drive brick serial number set at time of manufacture, right-justified with leading zeros, in ASCII.

6.4 Log Parameters

Log parameters are used in relation to LOG SELECT - 4Ch (see 5.2.12) commands and LOG SENSE - 4Dh (see 5.2.13) commands.

A list of all log pages, their reset behaviors, their access path policy, and links can be found in table 202.

WARNING WARNING

Log pages may have different reset behaviors. Care should be taken to ensure counters are processed appropriately, including consideration of their reset behaviors. While reset behaviors may be different between products, an attempt is made to maintain compatibility between generations of a given product.

WARNING WARNING

Table 202 — Supported log Pages

Code	Log Page	T10 Std	Pol [a]	Reset		
				Sel [b]	Sns [c]	Lod [d]
00h	LP 00h: Supported Log Pages (see 6.4.3 on page 290)	SPC	S	-	-	-
02h	LP 02h: Write Error Counters (see 6.4.4 on page 291)	SPC	H,L	H,L	H,L	ALL
03h	LP 03h: Read Error Counters (see 6.4.5 on page 292)	SPC	H,L	H,L	H,L	ALL
06h	LP 06h: Non-Medium Errors (see 6.4.6 on page 293)	SPC	H,L	H,L	H,L	-
0Ch	LP 0Ch: Sequential-Access Device (see 6.4.7 on page 294)	SSC	H,L	H,L	H,L	ALL
0Dh[01h]	LP 0Dh[01h]: Environmental Reporting (see 6.4.8 on page 296)	SPC	S	-	-	-
11h	LP 11h: DT Device Status (not J1A) (see 6.4.9 on page 300)	ADC	S	-	-	-
17h	LP 17h: Volume Statistics (see 6.4.10 on page 315)	SSC	S	-	-	-
2Eh	LP 2Eh: TapeAlerts (see 6.4.11 on page 327)	SSC	ITN	-	ITN	-
30h	LP 30h: Tape Usage (see 6.4.12 on page 330)	VU	S	-	-	-
31h	LP 31h: SIM/MIM (see 6.4.13 on page 331)	VU	S	-	Y	-
32h	LP 32h: Write Errors (see 6.4.14 on page 335)	VU	H,L	H,L	H,L	ALL
34h	LP 34h: Read Forward Errors (see 6.4.15 on page 337)	VU	H,L	H,L	H,L	ALL
36h	LP 36h: Read Backward Errors (see 6.4.16 on page 339)	VU	H,L	H,L	H,L	ALL
37h	LP 37h: Performance Characteristics (not J1A) (see 6.4.17 on page 340)	VU	S	Y	-	-
38h	LP 38h: Blocks/Bytes Transferred (see 6.4.18 on page 352)	VU	H,L	H,L	H,L	ALL
39h	LP 39h: Host Port 0 Interface Errors (see 6.4.19 on page 354)	VU	H,L	H,L	H,L	-

Key:

-	No
Y	Yes
S	Single shared aspect
H	Aspect related to access through the host primary ports
L	Aspect related to access through the library port
ITN	Aspect related to access through an I_T nexus (i.e., access path)
ALL	All aspects supported by the log page

[a] Policy—Which accesses path(s)/type(s) have common counters
NOTE: Prior to certain levels of code H and L aspects are shared (e.g., resets by either access path affect both)

[b] Aspects which may be reset using a LOG SELECT through that aspect

[c] Aspects reset when parameter(s) are read by a LOG SENSE through that aspect

[d] Aspects reset on a LOAD (regardless of cause of load)

[e] This log page is for engineering use only and is not reported as a supported log page

[f] See the page description for counters that do not follow the general behavior described in this table.

Table 202 — Supported log Pages

Code	Log Page	T10 Std	Pol[a]	Reset		
				Sel[b]	Sns[c]	Lod[d]
3Ah	LP 3Ah: Host Port 1 Interface Errors (see 6.4.20 on page 355)	VU	H,L	H,L	H,L	-
3Bh	LP 3Bh: Equipment Check Errors (see 6.4.21 on page 356)	VU	H,L	H,L	H,L	-
3Ch	LP 3Ch: Drive Control Statistics (see 6.4.22 on page 357)	VU	S	Y	-	Y
3Dh	LP 3Dh: Subsystem Statistics (see 6.4.23 on page 358)	VU	S	-	-	-
3Eh	LP 3Eh: Engineering Use (see 6.4.24 on page 361)[e]	VU	H,L	H,L[f]	H,L[f]	-[f]

Key:

-	No
Y	Yes
S	Single shared aspect
H	Aspect related to access through the host primary ports
L	Aspect related to access through the library port
ITN	Aspect related to access through an I_T nexus (i.e., access path)
ALL	All aspects supported by the log page

[a] Policy—Which accesses path(s)/type(s) have common counters
NOTE: Prior to certain levels of code H and L aspects are shared (e.g., resets by either access path affect both)

[b] Aspects which may be reset using a LOG SELECT through that aspect

[c] Aspects reset when parameter(s) are read by a LOG SENSE through that aspect

[d] Aspects reset on a LOAD (regardless of cause of load)

[e] This log page is for engineering use only and is not reported as a supported log page

[f] See the page description for counters that do not follow the general behavior described in this table.

6.4.1 Log Page Format

Each log page begins with a four-byte page header followed by zero or more variable-length log parameters defined for that log page. The log page format is defined in table 203.

Table 203 — Log page format

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	DS	SPF	PAGE CODE					
1	SUBPAGE CODE							
2	(MSB)	PAGE LENGTH (n-3)						
3								(LSB)
	Log parameter(s) (see 6.4.2)							
4	Log parameter (First) (Length x)							
x+3								
	. .							
n-y+1	Log parameter (Last) (Length y)							
n								

If the SPF bit is set to 0 then the Subpage Code field is reserved and is set to zero. If the SPF bit is set to one, then the subpage format is being used and the Subpage Code field is used to determine which log parameters are to be returned.

6.4.2 Log Parameter Format

Each log parameter begins with a 4-byte parameter header, followed by 1 or more bytes of parameter data. Table 204 shows the log parameter format. The fields of byte 2 are described under clause 6.4.2.1.

Table 204 — Log Parameter Format

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	MSB	Parameter Code						
1								LSB
2	DU	DS	TSD	ETC	TMC		Reserved	LP
3	Parameter Length (n-3)							
4	MSB	Parameter Value						
n								LSB

6.4.2.1 Log Parameter Byte 2 — Control Byte

A Log Parameter Control Byte is returned for each parameter code described in the log pages. The Log Parameter Control Byte is described here one time only. Any parameters using a different Log Parameter Control Byte have that byte described within that parameter.

The contents of this byte are constant; the user cannot set these values. This byte is part of the returned data described in the SCSI standard; it is described in the following list:

Bit	**Description**
7	DU (Disable Update): 0b
6	DS (Disable Save): 1b
5	TSD (Target Save Disable): 1b
4	ETC (Enable Threshold Comparison): 0b
3-2	TMC (Threshold Met Comparison): 00b
1	Reserved
0	LP (List Parameter): 0b (indicates this is a log counter)

6.4.3 LP 00h: Supported Log Pages

See LOG SELECT - 4Ch (see 5.2.12) and LOG SENSE - 4Dh (see 5.2.13) for directions on how to use this page. This log page returns the list of log pages supported. There may be log pages which can be read or reset which are not included in this list. Such pages are for engineering or manufacturing use and are not intended for general use. This page does not contain any log parameters.

Byte	Description
0	**Bit** **Description** 7-6 Reserved 5-0 Page Code (000000b)
1	Reserved
2-3	Page Length (n-3)
4-n	Supported Log Pages: This field is a list of 1-byte log page codes and may include some of the log pages listed in Table 202, “Supported log Pages,” on page 287

6.4.4 LP 02h: Write Error Counters

See LOG SELECT - 4Ch (see 5.2.12) and LOG SENSE - 4Dh (see 5.2.13) for directions on how to use this page. This page is for Write Errors.

6.4.4.1 Parameter Definitions

WARNING WARNING

Log pages may have different reset behaviors. Care should be taken to ensure counters are processed appropriately, including consideration of their reset behaviors. While reset behaviors may be different between products, an attempt is made to maintain compatibility between generations of a given product. See "Log Parameters" on page 287.

WARNING WARNING

WARNING WARNING

The parameters in log pages should be dynamically parsed as some parameters may not be present, new parameters may be inserted, and parameter sizes may change. The listed size is for the most recent generation at the time of publication. Some sizes have changed from previous generations. The relative location of parameters have changed and are anticipated to continue to change..

WARNING WARNING

Table 205 — Write Error Counters log parameter codes

Code	Counter: Description	Size
0002h	**Total Write Errors {02h:0002h}:** The sum of the Total Corrected Write Errors and Total Uncorrected Write Errors	2
0003h	**Total Corrected Write Errors {02h:0003h}:** These errors are corrected by ECC and do not require error recovery procedures (ERPs). Each count represents one block in error that was corrected and written.	2
0005h	**Total Write Kibibytes Processed {02h:0005h}:** Each count represents a kibibyte (2^{10}) of data processed across the host interface during write-type commands. The count does not include ERP retries. This field is identical to the Host Write Kibibytes Processed field of Page Code 38h, parameter code 0001h.	6
0006h	**Total Uncorrected Write Errors {02h:0006h}:** The total number of write errors that could not be corrected by ECC, no servo error was reported, and the error was not a transient error. Each count represents one block in error that was not corrected, but was recovered by ERPs and successfully written.	2

6.4.5 LP 03h: Read Error Counters

See LOG SELECT - 4Ch (see 5.2.12) and LOG SENSE - 4Dh (see 5.2.13) for directions on how to use this page. This page is for Read Errors.

6.4.5.1 Parameter Definitions

WARNING WARNING

Log pages may have different reset behaviors. Care should be taken to ensure counters are processed appropriately, including consideration of their reset behaviors. While reset behaviors may be different between products, an attempt is made to maintain compatibility between generations of a given product. See “Log Parameters” on page 287.

WARNING WARNING

WARNING WARNING

The parameters in log pages should be dynamically parsed as some parameters may not be present, new parameters may be inserted, and parameter sizes may change. The listed size is for the most recent generation at the time of publication. Some sizes have changed from previous generations. The relative location of parameters have changed and are anticipated to continue to change..

WARNING WARNING

Table 206 — Read Error Counters log parameter codes

Code	Counter: Description	Size
0002h	**Total Read Errors {03h:0002h}:** The sum of the Total Corrected Read Errors and the Total Uncorrected Read Errors.	2
0003h	**Total Corrected Read Errors {03h:0003h}:** These are errors that are corrected by ECC and do not require error recovery procedures (ERPs). Each count represents one block in error that was corrected and read.	2
0005h	**Total Read Kibibytes Processed {06h:0005h}:** Each count represents a kibibyte (2^{10}) processed across the host interface during read-type commands. The count does not include ERP retries. This field is identical to the Host Read Kibibytes Processed field of Page Code 38h, parameter code 0003h.	6
0006h	**Total Uncorrected Read Errors {03h:0006h}:** The total number of read errors that could not be corrected by ECC, no servo error was reported, and the error was not a transient error. Each count represents one block in error that was not corrected, but was recovered by ERPs and successfully read.	2

6.4.6 LP 06h: Non-Medium Errors

See LOG SELECT - 4Ch (see 5.2.12) and LOG SENSE - 4Dh (see 5.2.13) for directions on how to use this page. This page sums the occurrences of error events other than write or read failures. Parameter codes do not discriminate among the various types of events.

6.4.6.1 Parameter Definitions

WARNING WARNING

Log pages may have different reset behaviors. Care should be taken to ensure counters are processed appropriately, including consideration of their reset behaviors. While reset behaviors may be different between products, an attempt is made to maintain compatibility between generations of a given product. See "Log Parameters" on page 287.

WARNING WARNING

WARNING WARNING

The parameters in log pages should be dynamically parsed as some parameters may not be present, new parameters may be inserted, and parameter sizes may change. The listed size is for the most recent generation at the time of publication. Some sizes have changed from previous generations. The relative location of parameters have changed and are anticipated to continue to change..

WARNING WARNING

Table 207 — Non-Medium Errors log parameter codes

Code	Counter: Description	Size
0000h	**Total Non-Medium Error Count {06h:0000h}:**	4

6.4.7 LP 0Ch: Sequential-Access Device

See LOG SELECT - 4Ch (see 5.2.12) and LOG SENSE - 4Dh (see 5.2.13) for directions on how to use this page. This page is for counters specific to tape drives.

6.4.7.1 Parameter Definitions

WARNING WARNING

Log pages may have different reset behaviors. Care should be taken to ensure counters are processed appropriately, including consideration of their reset behaviors. While reset behaviors may be different between products, an attempt is made to maintain compatibility between generations of a given product. See “Log Parameters” on page 287.

WARNING WARNING

WARNING WARNING

The parameters in log pages should be dynamically parsed as some parameters may not be present, new parameters may be inserted, and parameter sizes may change. The listed size is for the most recent generation at the time of publication. Some sizes have changed from previous generations. The relative location of parameters have changed and are anticipated to continue to change.

WARNING WARNING

Table 208 — LP 0Ch: Sequential-Access Device log parameters (part 1 of 2)

Code	Counter: Description	Length
0000h	**Total Channel Write Bytes {0Ch:0000h}:** The total number of bytes of data written from the host on this mount.	8
0001h	**Total Device Write Bytes {0Ch:0001h}:** The total number of bytes of data written to tape on this mount, not counting ECC and formatting overhead. This is the number of data bytes after compression.	8
0002h	**Total Device Read Bytes {0Ch:0002h}:** The total number of bytes of data read from tape on this mount, not counting ECC and formatting overhead. This is the number of compressed data bytes read from media before decompression.	8
0003h	**Total Channel Read Bytes {0Ch:0003h}:** The total number of bytes of data read to the host on this mount.	8
0004h	**Approximate native capacity from BOP to EOD {0Ch:0004h}:** This is in megabytes (10^6). This is not sensitive to the current position of the medium. The approximate native capacity between EOD and EW is the difference of parameter 0005h and this parameter. Conditions may occur that reduce the amount of data that is written before reaching EW. EOD may be beyond LEOP. A value of all bits set to one indicates that this information is invalid (e.g., no volume is mounted, EOD information needs to be rebuilt).	4
0005h	**Approximate native capacity between BOP and EW of the current partition {0Ch:0005h}:** This is in megabytes (10^6). If no volume is mounted or this value is unknown the device server shall set all bits in this parameter to one.	4
0006h	**Minimum native capacity between EW and LEOP of the current partition {0Ch:0006h}:** This is in megabytes (10^6). If no volume is mounted the device server shall set all bits in this parameter to one.	4

Table 208 — LP 0Ch: Sequential-Access Device log parameters (part 2 of 2)

Code	Counter: Description	Length
0007h	**Approximate native capacity from BOP to the current position of the medium {0Ch:0007h}:** This is in megabytes (10^6). If no volume is mounted the device server shall set all bits in this parameter to one.	4
0008h	**Maximum native capacity that is currently allowed to be in the device object buffer {0Ch:0008h}:** This is in megabytes (10^6). This value may change depending on the current position of the medium (e.g., available native capacity may decrease as the current position of the medium approaches LEOP).	4
0100h	**Cleaning Requested {0Ch:0100h}:** A non-zero value indicates a cleaning action is requested by the drive.	1
8003h	**Remaining Capacity {0Ch:8002h}:** Nominal remaining unwritten capacity of the mounted media in bytes. This is not sensitive to current position. NOTE 71 - When the tape does not have a valid EOD, or if a tape is not loaded, a value of ’all ones’ (-1) is returned.	6

6.4.8 LP 0Dh[01h]: Environmental Reporting

See LOG SELECT - 4Ch (see 5.2.12) and LOG SENSE - 4Dh (see 5.2.13) for directions on how to use this page. This page is for counters specific to tape drives.

The Environmental Reporting log page (i.e.,log page 0Dh subpage 01h) is included in E06 and later. It is not included in earlier generation drives.

6.4.8.1 Parameter Definitions

WARNING WARNING

Log pages may have different reset behaviors. Care should be taken to ensure counters are processed appropriately, including consideration of their reset behaviors. While reset behaviors may be different between products, an attempt is made to maintain compatibility between generations of a given product. See "Log Parameters" on page 287.

WARNING WARNING

WARNING WARNING

The parameters in log pages should be dynamically parsed as some parameters may not be present, new parameters may be inserted, and parameter sizes may change. The listed size is for the most recent generation at the time of publication. Some sizes have changed from previous generations. The relative location of parameters have changed and are anticipated to continue to change.

WARNING WARNING

Table 209 — LP 0Dh[01h]: Environmental Reporting log parameters

Code	Counter: Description	Length
0000h	**Deck Temperature {0Dh[01h]:0000h}:** Information related to the reporting of temperature. See Temperature Report parameter data (see 6.4.8.1.1 on page 297) for the format of this parameter.	8
0001h to 00FFh	**Additional Component Temperatures {0Dh[01h]:00001h to 00FFh}:** Some products may report additional component temperatures at any parameter in this range. See Temperature Report parameter data (see 6.4.8.1.1 on page 297) for the format of these parameters.	8
0100h	**Deck Relative Humidity {0Dh[01h]:0100h}:** Information related to the reporting of relative humidity. See Relative Humidity Report parameter data (see 6.4.8.1.2 on page 298) for the format of this parameter.	8

6.4.8.1.1 Temperature Report parameter data

The Temperature Report parameter data has the format shown in table 210.

Table 210 — Temperature Report parameter data format

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Reserved							MTV
1	TEMPERATURE							
2	LIFETIME MAXIMUM TEMPERATURE							
3	LIFETIME MINIMUM TEMPERATURE							
4	MAXIMUM TEMPERATURE SINCE POWER ON							
5	MINIMUM TEMPERATURE SINCE POWER ON							
6	MAXIMUM MOUNTED TEMPERATURE							
7	MINIMUM MOUNTED TEMPERATURE							

The temperature values reported in the Temperature Report parameter data indicate a temperature in degrees Celsius. Negative values are indicated by two’s complement notation. A value of -128 (i.e., 80h) specifies that temperature is not valid.

Byte Description

0

Bit	**Description**
7-1	Reserved
0	MTV (mounted temperature valid): Indicates if the MAXIMUM MOUNTED TEMPERATURE field and the MINIMUM MOUNTED TEMPERATURE field are valid.

Value	**Description**
0b	Fields are invalid
1b	Fields are valid

1 TEMPERATURE: The temperature in degrees Celsius most recently read by the temperature sensor.

Value	**Description**
80h	There is no valid temperature to report.
other	The temperature in two’s complement notation (i.e., signed).

2 LIFETIME MAXIMUM TEMPERATURE: The maximum temperature in degrees Celsius read by the temperature sensor over the life of the drive.

Value	**Description**
80h	There is no valid temperature to report.
other	The temperature in two’s complement notation (i.e., signed).

3 LIFETIME MINIMUM TEMPERATURE: The minimum temperature in degrees Celsius read by the temperature sensor over the life of the drive.

Value	**Description**
80h	There is no valid temperature to report.
other	The temperature in two’s complement notation (i.e., signed).

4 MAXIMUM TEMPERATURE SINCE POWER ON: The maximum temperature in degrees Celsius read by the temperature sensor since power on (i.e., the last power cycle).

Value	**Description**
80h	There is no valid temperature to report.
other	The temperature in two’s complement notation (i.e., signed).

5 MINIMUM TEMPERATURE SINCE POWER ON The minimum temperature in degrees Celsius read by the temperature sensor since power on (i.e., the last power cycle).

Value	**Description**
80h	There is no valid temperature to report.
other	The temperature in two's complement notation (i.e., signed).

6 MAXIMUM MOUNTED TEMPERATURE: The meaning of this field depends on the value of the MTV bit.

MTV	**Meaning**
0b	This field is ignored.
1b	The maximum temperature detected from the most recent time that the volume was mounted until: A) the current time, if the volume has not been demounted; or B) the time at which the volume was demounted.

7 MINIMUM MOUNTED TEMPERATURE: The meaning of this field depends on the value of the MTV bit.

MTV	**Meaning**
0b	This field is ignored.
1b	The minimum temperature detected from the most recent time that the volume was mounted until: A) the current time, if the volume has not been demounted; or B) the time at which the volume was demounted.

6.4.8.1.2 Relative Humidity Report parameter data

The Relative Humidity Report parameter data has the format shown in table 211.

Table 211 — Relative Humidity Report parameter data format

Byte	**Bit**							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Reserved							MRHV
1	RELATIVE HUMIDITY							
2	LIFETIME MAXIMUM RELATIVE HUMIDITY							
3	LIFETIME MINIMUM RELATIVE HUMIDITY							
4	MAXIMUM RELATIVE HUMIDITY SINCE POWER ON							
5	MINIMUM RELATIVE HUMIDITY SINCE POWER ON							
6	MAXIMUM MOUNTED RELATIVE HUMIDITY							
7	MINIMUM MOUNTED RELATIVE HUMIDITY							

Byte **Description**

0

Bit	**Description**
7-1	Reserved
0	MRHV (mounted relative humidity valid): Indicates if the MAXIMUM MOUNTED RELATIVE HUMIDITY field and the MINIMUM MOUNTED RELATIVE HUMIDITY field are valid.

Value	**Description**
0b	Fields are invalid
1b	Fields are valid

1 RELATIVE HUMIDITY: The relative humidity most recently read from the humidity sensor.

Value	**Description**
0-100	Relative humidity
255	No valid relative humidity to report
others	Reserved

2 LIFETIME MAXIMUM RELATIVE HUMIDITY: The maximum relative humidity read by the humidity sensor over the life of the drive.

Value	Description
0-100	Relative humidity
255	No valid relative humidity to report
others	Reserved

3 LIFETIME MINIMUM RELATIVE HUMIDITY: The minimum relative humidity read by the humidity sensor over the life of the drive.

Value	Description
0-100	Relative humidity
255	No valid relative humidity to report
others	Reserved

4 MAXIMUM RELATIVE HUMIDITY SINCE POWER ON: The maximum relative humidity read by the humidity sensor since power on (i.e., the last power cycle).

Value	Description
0-100	Relative humidity
255	No valid relative humidity to report
others	Reserved

5 MINIMUM RELATIVE HUMIDITY SINCE POWER ON: The minimum relative humidity read by the humidity sensor since power on (i.e., the last power cycle).

Value	Description
0-100	Relative humidity
255	No valid relative humidity to report
others	Reserved

6 MAXIMUM MOUNTED RELATIVE HUMIDITY: The meaning of this field depends on the value of the MRHV bit.

MRHV	Meaning
0b	This field is ignored.
1b	The maximum relative humidity detected from the most recent time that the volume was mounted until: A) the current time, if the volume has not been demounted; or B) the time at which the volume was demounted.

7 MINIMUM MOUNTED RELATIVE HUMIDITY: The meaning of this field depends on the value of the MRHV bit.

MRHV	Meaning
0b	This field is ignored.
1b	The minimum relative humidity detected from the most recent time that the volume was mounted until: A) the current time, if the volume has not been demounted; or B) the time at which the volume was demounted.

6.4.9 LP 11h: DT Device Status (not J1A)

See LOG SELECT - 4Ch (see 5.2.12) and LOG SENSE - 4Dh (see 5.2.13) for directions on how to use this page.

The DT Device Status log page (see table 212) defines log information pertaining to the DT device (i.e. tape drive) and DT device primary ports.

Table 212 — LP 11h: DT Device Status log page

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Reserved		PAGE CODE (11h)					
1	Reserved							
2	(MSB)	PAGE LENGTH (n-3)						
3								(LSB)
4	DT Device Status log parameters							
n								

Byte	Description
0	

Bit	Description
7-6	Reserved
5-0	Page Code (11h)

1	Reserved
2-3	Page Length (n-3)
4-n	DT Device Status log parameters(see 6.4.9.1).

6.4.9.1 Parameter Definitions

WARNING WARNING

Log pages may have different reset behaviors. Care should be taken to ensure counters are processed appropriately, including consideration of their reset behaviors. While reset behaviors may be different between products, an attempt is made to maintain compatibility between generations of a given product. See "Log Parameters" on page 287.

WARNING WARNING

WARNING WARNING

The parameters in log pages should be dynamically parsed as some parameters may not be present, new parameters may be inserted, and parameter sizes may change. The listed size is for the most recent generation at the time of publication. Some sizes have changed from previous generations. The relative location of parameters have changed and are anticipated to continue to change.

WARNING WARNING

DT Device Status log parameters are shown in table 213.

Table 213 — DT Device Status log parameters

Parameter Code	Description
0000h	**Very high frequency data:** (see 6.4.9.1.1)
0001h	**Very high frequency polling delay:** (see 6.4.9.1.2)
0101h - 0102h	**DT device primary port status:** (see 6.4.9.1.3)
0301h to 03FFh	**DT device primary port physical interface Information:** (see 6.4.9.1.5)

Table 213 — DT Device Status log parameters

Parameter Code	Description
8000h	**Medium VolSer:** (see 6.4.9.1.6)
8001h	**Medium Status Data:** (see 6.4.9.1.7)
8100h	**Drive Status data:** (see 6.4.9.1.8)
9101h	Reserved for Primary Port Features (first port)
9102h	Reserved for Primary Port Features (second port)
E000h	**Encryption Control Descriptor:** (see 6.4.9.1.9)

6.4.9.1.1 Very high frequency data log parameter

The very high frequency data log parameter format is shown in Table 214.

Table 214 — Very high frequency data log parameter format

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	(MSB)	PARAMETER CODE (0000h)						
1								(LSB)
2	DU (0)	Obsolete	TSD (0)	ETC (0)	TMC (0)		LBIN (1)	LP (1)
3	PARAMETER LENGTH (04h)							
4	VHF data descriptor							
7								

Byte Description

0-1 PARAMETER CODE: (0000h)

2 Parameter list control byte - binary format list log parameter

Bit Description

7 DU: 0b
6 Obsolete
5 TSD: 0b
4 ETC: 0b
3-2 TMC: 00b
1 LBIN: 1b
0 LP: 1b

3 PARAMETER LENGTH: (04h)

Transfer of the complete parameter is required.

4-7 VHF data descriptor

The VHF data descriptor is defined in table 215. Returned data shall reflect the last known values since the DT device initialized.

Table 215 — VHF data descriptor

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
4	PAMR	HIU	MACC	CMPR	WRTP	CRQST	CRQRD	DINIT
5	INXTN	Rsvd	RAA	MPRSNT	Rsvd	MSTD	MTHRD	MOUNTED
6	DT DEVICE ACTIVITY							
7	VS	Rsvd	TDDEC	EPP	ESR	RRQST	INTFC	TAFC

NOTE 72 - In addition to reliance on indication of initialization completion, reliance on returned values should also take into consideration conditions indicated by changes in Tape Alert flag status, and process those first as needed.

Byte	**Description**
4	

Bit	**Description**
7	PAMR(prevent/allow medium removal):

Value	**Description**
0b	Medium removal is not prevented as the result of a PREVENT/ALLOW MEDIUM REMOVAL command on LUN 0.
1b	Medium removal is prevented as the result of a PREVENT/ALLOW MEDIUM REMOVAL command on LUN 0.

6 HIU(host initiated unload):

Value	**Description**
0b	Volume is not in an unload state due to a LOAD UNLOAD command.
1b	Volume is unloaded due to a LOAD UNLOAD command on LUN 0.

5 MACC (medium auxiliary memory accessible):

Value	**Description**
0b	Medium Auxiliary Memory (MAM) is not accessible.
1b	Medium Auxiliary Memory (MAM) is accessible.

4 CMPR (compress):

Value	**Description**
0b	Data compression is not enabled.
1b	Data compression is enabled.

3 WRTP (write protect): only valid if the MPRSNT bit is set to one.

Value	**Description**
0b	Volume is not physically write protected.
1b	Volume is physically write protected.

2 CRQST (cleaning requested):

Value	**Description**
0b	Cleaning not requested.
1b	Cleaning requested.

1 CRQRD (cleaning required):

Value	**Description**
0b	Cleaning not required; normal operation possible without cleaning.
1b	Cleaning required; normal operation may not be possible until drive is cleaned.

0 DINIT (DT device initialized):

Value	**Description**
0b	DT device initialization is required or incomplete. VHF data not valid.
1b	VHF data valid

5

Bit **Description**

7 INXTN (in transition): Indicates the stability of the other bits in this byte (i.e., byte 5) and whether state transitions are taking place.

Value	Description
0b	Device is in the state reflected by the remaining bits in this byte and is making no attempt to leave this state.
1b	Device is transitioning to another state; other bits in this byte are in transition.

6 Rsvd (Reserved)

5 RAA (robotic access allowed):

Value	Description
0b	Library or Medium Changer should not move a volume to or from the device.
1b	Library or Medium Changer may move a volume to or from the device.

4 MPRSNT (medium present):

Value	Description
0b	The device does not detect a volume present.
1b	The device detects a volume present.

3 Rsvd (Reserved)

2 MSTD (medium seated):

Value	Description
0b	Cartridge is not seated; further mechanical motion remains in order to complete the loading process, exclusive of tape threading.
1b	Cartridge is mechanically seated within the loading mechanism (i.e., the physical loading process has completed).

1 MTHRD (medium threaded):

The value of the MTHRD bit may or may not correspond to the device responding with a status of GOOD to a TEST UNIT READY command, as additional processing may be required by the device after threading before the logical unit becomes ready.

Value	Description
0b	Medium is not threaded.
1b	Medium has been threaded; tape motion operations are possible. The value of the MTHRD bit may or may not correspond to the DT device responding with a status of GOOD to a TEST UNIT READY command, as additional processing may be required by the DT device after threading before the logical unit becomes ready.

0 MOUNTED:

Value	Description
0b	Volume is not mounted.
1b	Volume is mounted. The drive may be able to respond to a TEST UNIT READY command with GOOD status, however when a cleaning cartridge or microcode update cartridge is loaded the drive may respond to a TEST UNIT READY command with a CHECK CONDITON with the sense key set to NOT READY.

6 DT DEVICE ACTIVITY: This field is used to describe the current activity of the device

Value	Description
00h	No DT device activity
01h	Cleaning operation in progress
02h	Medium is being loaded
03h	Medium is being unloaded
04h	Other medium activity
05h	Reading from medium
06h	Writing to medium
07h	Locating medium
08h	Rewinding medium
09h	Erasing medium
0Ah	Formatting medium
0Bh	Calibrating medium
0Ch	Other DT device activity
0Dh	Microcode update in progress

7

Bit	Description
7	VS: (0b)
6-3	Reserved
5	TDDEC (tape diagnostic data entry created):

Value	Description
0b	The device has not created a new Tape Diagnostic Data log page entry since the last retrieval of any of the parameters from the Tape Diagnostic Data log page by this I_T nexus.
1b	The device has created a new Tape Diagnostic Data log page entry since the last retrieval of any of the parameters from the Tape Diagnostic Data log page by this I_T nexus.

4 EPP (encryption parameters present):

Value	Description
0b	The device does not have a set of saved data encryption parameters with either the ENCRYPTION MODE field set to a value other than DISABLE or the DECRYPTION MODE field set to a value other than DISABLE
1b	The device has a set of saved data encryption parameters with either the ENCRYPTION MODE field set to a value other than DISABLE or the DECRYPTION MODE field set to a value other than DISABLE

3 ESR (encryption service request):

Value	Description
0b	At least one bit in the SERVICE REQUEST INDICATORS field in the DT device ADC data encryption control status log parameter has been set to one since the last retrieval of the DT device ADC data encryption control status log parameter by this I_T nexus and at least one bit in the SERVICE REQUEST INDICATORS field in the DT device ADC data encryption control status log parameter is set to one.
1b	No bits in the SERVICE REQUEST INDICATORS field in the DT device ADC data encryption control status log parameters have been set to one since the last retrieval of the DT device ADC data encryption control status log parameter by

this I_T nexus or all of the bits in the SERVICE REQUEST INDICATORS field in the DT device ADC data encryption control status log parameter are set to zero.

2 RRQST (recovery requested):

Value	**Description**
0b	No recovery procedure is requested.
1b	Device has detected an error and one or more requested recovery procedures are available via the ADC Requested Recovery log page.

1 INTFC (interface changed):

Value	**Description**
0b	No fields in the DT device primary port status log parameters have changed since the last retrieval of any of the DT device primary port status log parameters from the DT Device Status log page over this I_T nexus.
1b	One or more fields in the DT device primary port status log parameters have changed since the last retrieval of any of the DT device primary port status log parameters from the DT Device Status log page over this I_T nexus.

0 TAFC (TapeAlert state flag changed):

Value	**Description**
0b	No TapeAlert state flag has changed since the last retrieval of the TapeAlert Response log page over this I_T nexus.
1b	At least one TapeAlert state flag has changed since the last retrieval of the TapeAlert Response log page over this I_T nexus.

6.4.9.1.2 Very high frequency polling delay log parameter

The very high frequency polling delay log parameter format is shown in Table 216.

Table 216 — Very high frequency polling delay log parameter format

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	(MSB)	PARAMETER CODE (0001h)						
1								(LSB)
2	DU (0)	Obsolete	TSD (0)	ETC (0)	TMC (00)		LBIN (1)	LP (1)
3	PARAMETER LENGTH (02h)							
4	(MSB)	VHF POLLING DELAY						
5								(LSB)

Byte Description

0-1 PARAMETER CODE: (0001h)

2 Parameter list control byte - binary format list log parameter

Bit	**Description**
7	DU: 0b
6	Obsolete
5	TSD: 0b
4	ETC: 0b
3-2	TMC: 00b
1	LBIN: 1b
0	LP: 1b

3 PARAMETER LENGTH: (02h)

Transfer of the complete parameter is required.

4-7 VHF POLLING DELAY: The minimum delay in milliseconds before another DT Device Status log page should be requested.

6.4.9.1.3 Primary port status log parameter(s)

There is a primary port status log parameter for each primary port of the device. The format is shown in table 217.

Table 217 — Primary port status log parameter(s) format

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	(MSB)	PARAMETER CODE						
1								(LSB)
2	DU (0)	Obsolete	TSD (0)	ETC (0)	TMC (00)		LBIN (1)	LP (1)
3	PARAMETER LENGTH (n-3)							
4	(MSB)	Primary port status data						
n								(LSB)

Byte Description

0-1 PARAMETER CODE:

0100h plus the value of the Relative Target Port field associated with that port. The Relative Target Port is the same as the relative target port value defined in the VPD pages of inquiry

Value	Description
0101h	Primary port 1; traditionally known as port 0.
0102h	Primary port 2; traditionally known as port 1.

2 Parameter list control byte - binary format list log parameter

Bit	Description
7	DU: 0b
6	Obsolete
5	TSD: 0b
4	ETC: 0b
3-2	TMC: 00b
1	LBIN: 1b
0	LP: 1b

3 PARAMETER LENGTH:

4-7 Primary port status data: This is determined by the protocol of the port with which the parameter is associated. The protocol of the port is reported in the PROTOCOL IDENTIFIER field of the Relative target port identifier designation descriptor of the Device identification VPD page for the associated port.

PROTOCOL	Description
0h	Fibre Channel port status data (see 6.4.9.1.4 on page 307)

6.4.9.1.4 Fibre Channel port status data

The format of the primary port status data for a Fibre Channel port is shown in table 218. This descriptor reports the current operating points of the specified port.

Table 218 — Fibre Channel port status data format

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	CURRTOP	CURRENT SPEED			LC	CONFLICT	SIGNAL	PIC
1	CURRENT N_PORT_ID							
3								
4	ECSV	Reserved						
5	Reserved				EXTENDED CURRENT SPEED			
6	Reserved							
7	Rsvd	CURRENT FC-AL LOOP ID						
8	CURRENT PORT NAME							
15								
16	CURRENT NODE NAME							
23								

Byte Description

0

Bit Description

7 CURRTOP (current topology):

This field is undefined when the PIC field is set to zero

Value	Description
0b	Port is currently operating in arbitrated loop mode.
1b	Port is currently operating in point to point mode.

6-4 CURRENT SPEED: This field is undefined when the PIC field is set to zero

Value	Description
000b	1 Gb/sec.
001b	2 Gb/sec.
010b	4 Gb/sec.
011b	8 Gb/sec.
100b	Not Supported
others	Reserved

3 LC (login complete):

Value	Description
0b	No host is currently logged in to the drive through this port .
1b	At least one host is currently logged in to the drive through this port (i.e., has successfully completed PRLI and still has an active session).

2 CONFLICT:

Value	Description
0b	No AL_PA conflict exists on this port.
1b	The required Hard AL_PA is in use by another device or no AL_PA is available for this port.

1 SIGNAL:

Value	Description
0b	Signal (i.e., light) is not detected on this port.
1b	Signal (i.e., light) is detected on this port.

0 PIC (port initialization complete):

Value	Description
0b	The FC_Port state machine is not in the ACTIVE state (if port is operating in point-to-point topology), or has not successfully completed the most recent LIP.
1b	The FC_Port state machine is in the ACTIVE state (if port is operating in point-to-point topology), or the most recent LIP has completed successfully.

1-3 CURRENT N_PORT_ID: The 24-bit N_Port_ID that is currently assigned to this port. This field is undefined when the PIC field is set to zero.

4

Bit	Description
7	ECSV (extended current speed valid):

Value	Description
0b	EXTENDED CURRENT SPEED field is not valid. The speed is reported in the CURRENT SPEED field.
1b	EXTENDED CURRENT SPEED field is valid. The speed is reported in the EXTENDED CURRENT SPEED field.

Bit	Description
6-0	Reserved

5

Bit	Description
7-4	Reserved
3-0	EXTENDED CURRENT SPEED:

This field is not valid when the ECSV field is set to zero.

Value	Description
0000b	Not reported
0001b	2 Gb/sec
0010b	4 Gb/sec
0011b	8 Gb/sec
0101b	16 Gb/sec
0110b	Not Supported
0111b	Not Supported
others	Reserved

6 Reserved

7

Bit	Description
7	Reserved
6-0	CURRENT FC-AL LOOP ID: The loop identifier assigned to this port. This field is ignored ignored when the PIC bit is set to zero or when the CURRTOP bit is set to one.

8-15 CURRENT PORT NAME: The port's name identifier (i.e., WWPN).

16-23 CURRENT NODE NAME: The device's node name identifier (i.e., WWNN).

6.4.9.1.5 DT device primary port physical interface information

The DT device primary port physical interface information log parameter(s) format is shown in table 219.

WARNING WARNING

The information from the SFP is read during power on. If SFPs are hot swapped there is no guarantee that the data will be current.

WARNING WARNING

Table 219 — DT device primary port status log parameter(s) format

Byte	**Bit**							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	(MSB)	PARAMETER CODE						
1								(LSB)
2	Parameter list control byte - binary format list log parameter							
	DU	Obsolete	TSD	Obsolete			FORMAT AND LINKING	
3	PARAMETER LENGTH (84h)							
4	PHYSICAL INFORMATION FORMAT							
5	Reserved							
6								
7								
8	DT device primary port physical interface information data							
135								

Byte **Description**

0-1 PARAMETER CODE:

0300h plus the value of the Relative Target Port field associated with that port. The Relative Target Port is the same as the relative target port value defined in the VPD pages of inquiry

Value	**Description**
0301h	Primary port 1; traditionally known as port 0.
0302h	Primary port 2; traditionally known as port 1.

2 Parameter list control byte - binary format list log parameter

Bit	**Description**
7	DU: 0b
6	Obsolete
5	TSD: 0b
4 to 2	Obsolete
1 to 0	FORMAT AND LINKING: 11b

3 PARAMETER LENGTH: (84h)

4 PHYSICAL INFORMATION FORMAT: The format of the DT device primary port physical interface information data

Code	**Description**
00h	No information is available. The DT device primary port physical interface information data is not valid.
01h	The DT device primary port physical interface information data is set to bytes 0 through 127 of the SFF-8472 physical interface memory map address A0h.

5 to 7 Reserved

8 to 135 DT device primary port physical interface information data (see the PHYSICAL INFORMATION FORMAT field).

6.4.9.1.6 Medium VolSer log parameter

The Medium VolSer indicates the Volume Label Serial Number (VolSer) recorded in the CM of the currently mounted volume. (This is not to be confused with the manufacturer’s cartridge serial number also maintained in the CM.) The drive reports the VolSer according to the following prioritized order:

1) the VolSer, if any, transmitted by the library and recorded in the CM via the WRITE BUFFER command;
2) the VolSer, if any, transmitted by the host application via the WRITE ATTRIBUTE command and recorded into a standard field in CM; or
3) all ASCII blanks.

The first seven characters of the VolSer are transferred in the REQUEST SENSE data. The format of this log parameter is shown in table 220.

Table 220 — Medium Volume Label Serial Number log parameter format

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	(MSB)	PARAMETER CODE (8000h)						
1								(LSB)
2	DU (0)	Obsolete	TSD (0)	ETC (0)	TMC (0)		LBIN (1)	LP (1)
3	PARAMETER LENGTH (08h)							
4	VOLSER							
11								

Byte Description

0-1 PARAMETER CODE: (8000h)

2 Parameter list control byte - binary format list log parameter

Bit	**Description**
7	DU: 0b
6	Obsolete
5	TSD: 0b
4	ETC: 0b
3-2	TMC: 00b
1	LBIN: 1b
0	LP: 1b

3 PARAMETER LENGTH: (08h)

4-11 VOLSER: Volume Label Serial Number in ASCII

6.4.9.1.7 Medium Status Data log parameter

The Medium Status Data log parameter provides information related to the currently mounted volume. The format of this log parameter is shown in table 221.

Table 221 — Medium Status Data log parameter format

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	(MSB)	PARAMETER CODE (8001h)						
1								(LSB)
2	DU (0)	Obsolete	TSD (0)	ETC (0)	TMC (0)		LBIN (1)	LP (1)
3	PARAMETER LENGTH (n-3)							
4	Reserved			TAPE CARTRIDGE TYPE				
5	Rsvd	MEC	Reserved			VAE	MES	
6	Reserved							
n								

Byte	Description
0-1	PARAMETER CODE: (8001h)
2	Parameter list control byte - binary format list log parameter

Bit	Description
7	DU: 0b
6	Obsolete
5	TSD: 0b
4	ETC: 0b
3-2	TMC: 00b
1	LBIN: 1b
0	LP: 1b

Byte	Description
3	PARAMETER LENGTH: (n-3)
4	TAPE CARTRIDGE TYPE:

Value	Description
00010b	Cleaner cartridge
00111b	Invalid or unknown cartridge type
01001b	3592 JA, JJ, JR, or JW cartridge
01011b	3592 JB or JX cartridge
01110b	3592 JC, JY, or JK cartridge
10000b	3592 JD, JZ, or JL cartridge

5

Bit	Description
7	Reserved
6	MEC (Medium is Encryption Capable): Indicates the currently loaded medium may be used for encryption purposes.
5-3	Reserved
2	VAE (Valid After Eject): Indicates the validity of the Medium Status Data after the volume is ejected.

Value	Description
0b	The data is cleared when the cartridge is ejected (extracted) from the drive.
1b	The data is valid after the cartridge is ejected.

Bit	Description
1-0	MES (Medium Encryption Status): Indicates the encryption status of the currently mounted volume.

Value	Description
00b	Unable to determine if all data on all partitions is encrypted or if all data on all partitions is unencrypted
01b	Some data on at least one partition is unencrypted
10b	All data on all partitions is encrypted
11b	Reserved

6-n Reserved

6.4.9.1.8 Drive Status Data log parameter

The Drive Status Data log parameter provides information related to the drive. The format of this log parameter is shown in table 222.

Table 222 — Drive Status Data log parameter format

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	(MSB)	PARAMETER CODE (8100h)						
1								(LSB)
2	DU (0)	Obsolete	TSD (0)	ETC (0)	TMC (0)		LBIN (1)	LP (1)
3	PARAMETER LENGTH (n-3)							
4	Reserved							POST
5	Reserved						EE	EC
6	Reserved							
n								

Byte	Description
0-1	PARAMETER CODE: (8100h)
2	Parameter list control byte - binary format list log parameter

Bit	Description
7	DU: 0b
6	Obsolete
5	TSD: 0b
4	ETC: 0b
3-2	TMC: 00b
1	LBIN: 1b
0	LP: 1b

Byte	Description
3	PARAMETER LENGTH: (n-3)

4

Bit	**Description**
7-1	Reserved
0	POST (POWER-ON SELF TEST): Indicates if the drive has completed its initialization or self test diagnostics.

Value	**Description**
0b	POST has completed.
1b	POST has not completed.

5

Bit	**Description**
7-2	Reserved
1	EE (Encryption Enabled): Indicates if the drive is enabled to perform encryption operations.

Value	**Description**
0b	The drive is not enabled to perform encryption operations.
1b	The drive is enabled to perform encryption operations.

Bit	**Description**
1-0	EC (Encryption Capable): Indicates if the drive contains hardware capable of performing encryption operations. The drive may or may not be enabled for encryption.

Value	**Description**
0b	The drive does not contain hardware capable of performing encryption operations.
1b	The drive does contain hardware capable of performing encryption operations.

6-n Reserved

6.4.9.1.9 Encryption Control Descriptor

The Encryption Control Descriptor is used to convey encryption control information to the automation device during certain phases of the encryption process. Details of such events are outside the scope of this document. The format of this log parameter is shown in table 223.

Table 223 — Encryption Control Descriptor log parameter format

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	(MSB)	PARAMETER CODE (E000h)						
1								(LSB)
2	DU (0)	Obsolete	TSD (0)	ETC (0)	TMC (0)		LBIN (1)	LP (1)
3	PARAMETER LENGTH (n-3)							
4	ENCRYPTION SEQUENCE IDENTIFIER							
7								
8	Encryption Control Descriptor Parameter Data							
n								

Byte	**Description**
0-1	PARAMETER CODE: (E000h)
2	Parameter list control byte - binary format list log parameter

Bit	**Description**
7	DU: 0b
6	Obsolete
5	TSD: 0b
4	ETC: 0b
3-2	TMC: 00b
1	LBIN: 1b
0	LP: 1b

3 PARAMETER LENGTH: (08h)

4-7 ENCRYPTION SEQUENCE IDENTIFIER: An identifier for the endryption data related to the Encryption Control Descriptor Parameter Data that is sent back to the drive under certain asynchronously driven encryption events. The Encryption Sequence Identifier shall be returned back to the drive without modification in certain Encryption commands (documented outside the scope of this document).

8-n Encryption Control Descriptor Parameter Data is described in *IBM Automation Drive Interface Specification with Encryption Support*.

6.4.10 LP 17h: Volume Statistics

See LOG SELECT - 4Ch (see 5.2.12) and LOG SENSE - 4Dh (see 5.2.13) for directions on how to use this page. This page contains non-volatile information relating to volume usage.

The Volume Statistics log page is included in E06 and later. It is not included in earlier generation drives.

The Volume Statistics log page *(see table 224)* defines data parameters associated with utilization of the tape volume and the medium within the volume. Volume statistics for the most recent mounted volume are reported in the volume Statistics log page parameters. Volume statistics for previously mounted volumes may be reported in Volume Statistics log subpages (see table 225). A device server that implements the Volume Statistics log page shall implement support for the most recent mounted volume and the defined parameters as shown in table 226. The parameters shall not be set to zero or changed with the use of a LOG SELECT command.

If a supported log subpage is requested for a mount which has not occurred, then all bytes in the parameter data fields shall be set to 00h.

NOTE 73 - An application client may detect if parameter values in the page are valid by testing for parameter value 0000h, (i.e., page valid) set to zero.

Table 224 — Volume Statistics log page

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	DS	SPF	PAGE CODE (17h)					
1	SUBPAGE CODE							
2	(MSB)	PAGE LENGTH (N-3)						
3								(LSB)
	Volume Statistics log parameters							
4	Volume Statistics log parameters [first]							
	. . .							
n	Volume Statistics log parameters [last]							

See table 225 for the definition of the SUBPAGE CODE field.

Table 225 — Volume Statistics log subpage codes

Code	Description
00h	Reserved
01h-0Fh	Volume statistics for previously mounted volumes 1 through 15. Use supage FFh (i.e., Supported subpages log page) to determine which log subpages are supported. This will also indicate how many previous mounts are supported by the drive.
10h-FEh	Reserved
FFh	Supported subpages log page. The supported subpages log page reports the number of subpages supported for volume statitistics log pages saved for previous mounts.

6.4.10.1 Parameter Definitions

WARNING WARNING

Log pages may have different reset behaviors. Care should be taken to ensure counters are processed appropriately, including consideration of their reset behaviors. While reset behaviors may be different between products, an attempt is made to maintain compatibility between generations of a given product. See "Log Parameters" on page 287.

WARNING WARNING

WARNING WARNING

The parameters in log pages should be dynamically parsed as some parameters may not be present, new parameters may be inserted, and parameter sizes may change. The listed size is for the most recent generation at the time of publication. Some sizes have changed from previous generations. The relative location of parameters have changed and are anticipated to continue to change.

WARNING WARNING

Table 226 — Volume statistics log parameters (part 1 of 7)

Parameter Code	Description	Type	Size
0000h	**Page valid {17h:0000h}:** A value of 01h indicates that the values reported in the parameters to follow are valid. A value of 00h indicates the values reported in the parameters that follow are invalid	C	1
0001h	**Volume Mounts {17h:0001h}:** Number of mounts for the current volume (i.e., Thread Count)	C	4
0002h[a]	**Volume Datasets Written {17h:0002h}:** The total number of data sets written to the medium in the volume over the lifetime of the volume. (i.e., Total Datasets Written)	C	8
0003h[a]	**Volume Recovered Write Data Errors {17h:0003h}:** The total number of recovered write data correction errors (e.g., write temps) for the lifetime of the volume. (i.e., Total Write Retries)	C	4
0004h[a]	**Volume Unrecovered Write Data Errors (i.e., Write Perms) {17h:0004h}:** The total number of times that a write type command was terminated with CHECK CONDITION status and a sense key of MEDIUM ERROR or HARDWARE ERROR over the lifetime of the volume. (i.e., Total Unrecovered Write Errors)	C	2

Type Key:
C - Volume statistics data counter log parameter (see 6.4.10.2.1)
S - Volume statistics string data log parameter (see 6.4.10.2.2)
P - Volume statistics partition record log parameter (see 6.4.10.2.3). The size field is the size of the PARTITION RECORD DATA COUNTER FIELD OF EACH VOLUME STATISTICS PARTITION RECORD DESCRIPTOR.
M - Mount history log parameter (see 6.4.10.2.4)

Size Key:
v - variable size

Note:
If a parameter value is unknown but other parameter values are known, then parameter for the unknown value is not returned. If none of the values are known, then parameter 0000h is set to 00h.

Footnotes:

[a] Not supported on E06

[b] The value of this parameter may be unknown for some volume personalities. When this occurs, this parameter is not returned.

[c] The size field for a counter of Type P is the size of the PARTITION RECORD DATA COUNTER FIELD OF EACH VOLUME STATISTICS PARTITION RECORD DESCRIPTOR. The size of this counter depends on the number of partitions and is number_of_partitions * (4 bytes internal header + value in Size column).

Table 226 — Volume statistics log parameters (part 2 of 7)

Parameter Code	Description	Type	Size
0005h[a]	**Volume Write Servo Errors (updated on both temp and perm) {17h:0005h}:** The total number of times that the device suspended a write due to detection of a servo condition which could result in an incorrectly written track and attempted to write the data at a different location. If a logical block is interrupted more than once this parameter shall only be incremented once. (i.e., Total Number of Suspended Writes)	C	2
0006h [a]	**Volume Unrecovered Write Servo Errors {17h:0006h}:** The total number of times that the device suspended a write due to detection of a servo condition which could result in an incorrectly written track and was unable to write the data at a different location. (i.e., Total Number of Fatal Suspended Writes)	C	2
0007h[a]	**Volume Datasets Read {17h:0007h}:** The total number of data sets read from the medium in the volume over the lifetime of the volume. (i.e., Total Data Sets Read)	C	8
0008h[a]	**Volume Recovered Read Errors {17h:0008h}:** The total number of recovered read errors (e.g., read temps) for the lifetime of the volume. (i.e., Total Read Retries)	C	4
0009h[a]	**Volume Unrecovered Read Errors (i.e., Read Perms) {17h:0009h}:** The total number of times that a read type command was terminated with CHECK CONDITION status and a sense key of MEDIUM ERROR or HARDWARE ERROR over the lifetime of the volume. (i.e., Total Unrecovered Read Errors)	C	2
000Ah-000Bh	Not Supported		
000Ch[a]	**Last mount unrecovered write errors {17h:000Ch}:** Count of the number times a write type command was terminated with status of CHECK CONDITION and a sense key of HARDWARE ERROR, or MEDIA ERROR during the last mount	C	2
000Dh[a]	**Last mount unrecovered read errors {17h:000Dh}:** Count of the number times a read type command was terminated with status of CHECK CONDITION and a sense key of HARDWARE ERROR, or MEDIA ERROR during the last mount	C	2
000Eh[a]	**Last mount megabytes written {17h:000Eh}:** Count of the number of megabytes (10^6 bytes) of logical objects that were written to the medium after compression during the last mount. The value reported is rounded up to the next megabyte. The value reported contains bytes written as part of the process of writing a filemark. This value is calculated using information from the current format of the volume. If the volume has been reformatted, then this value may not be accurate.	C	4

Type Key:
C - Volume statistics data counter log parameter (see 6.4.10.2.1)
S - Volume statistics string data log parameter (see 6.4.10.2.2)
P - Volume statistics partition record log parameter (see 6.4.10.2.3). The size field is the size of the PARTITION RECORD DATA COUNTER FIELD OF EACH VOLUME STATISTICS PARTITION RECORD DESCRIPTOR.
M - Mount history log parameter (see 6.4.10.2.4)

Size Key:
v - variable size

Note:
If a parameter value is unknown but other parameter values are known, then parameter for the unknown value is not returned. If none of the values are known, then parameter 0000h is set to 00h.

Footnotes:
[a] Not supported on E06
[b] The value of this parameter may be unknown for some volume personalities. When this occurs, this parameter is not returned.
[c] The size field for a counter of Type P is the size of the PARTITION RECORD DATA COUNTER FIELD OF EACH VOLUME STATISTICS PARTITION RECORD DESCRIPTOR. The size of this counter depends on the number of partitions and is number_of_partitions * (4 bytes internal header + value in Size column).

Table 226 — Volume statistics log parameters (part 3 of 7)

Parameter Code	Description	Type	Size
000Fh[a]	**Last mount megabytes read {17h:000Fh}:** Count of the number of megabytes (10^6 bytes) of logical objects that were read from the medium before decompression during the last mount. The value reported is rounded up to the next megabyte. The value reported contains bytes read as part of a filemark. This value is calculated using information from the current format of the volume. If the volume has been reformatted, then this value may not be accurate.	C	4
0010h[a]	**Lifetime megabytes written {17h:0010h}:** Count of the number of megabytes (10^6 bytes) of logical objects that have been written to the medium after compression during the lifetime of the volume. The value reported is rounded up to the next megabyte. The value reported contains bytes written as part of the process of writing a filemark. This value is calculated using information from the current format of the volume. If the volume has been reformatted, then this value may not be accurate.	C	8
0011h[a]	**Lifetime megabytes read {17h:0011h}:** Count of the number of megabytes (10^6 bytes) that have been read from the medium before decompression during the lifetime of the volume. The value reported is rounded up to the next megabyte. The value reported contains bytes read as part of a filemark. This value is calculated using information from the current format of the volume. If the volume has been reformatted, then this value may not be accurate.	C	8
0012h[a]	**Last load write compression ratio {17h:0012h}:** (number of bytes transferred out of the logical object buffer to an application client ÷ the number of bytes in logical objects read from the medium) x 100	C	2
0013h[a]	**Last load read compression ratio {17h:0017h}:** (number of bytes transferred from an application client into the logical object buffer ÷ the number of bytes in logical objects written to the medium) x 100	C	2
0014h[a]	**Medium mount time {17h:0014h}:** Time in milliseconds from the time when the device server would first report GOOD status to a TEST UNIT READY command upon successful completion of a load operation until the device server did not detect a volume present.	C	6
0015h[a]	**Medium ready time {17h:0015h}:** Time in milliseconds from the time the device server was able to process medium access commands until the device server started the processing of an unload operation.	C	6
0016h	**Total native capacity {17h:0016h}:** The sum of the total native capacity of all partitions in megabytes (10^6 bytes) from BOP to EOP. A data counter value with all bytes set to FFh in the partition record data counter field (see 6.4.10.2.3) indicates that the total native capacity is unknown.	C	4

Type Key:
C - Volume statistics data counter log parameter (see 6.4.10.2.1)
S - Volume statistics string data log parameter (see 6.4.10.2.2)
P - Volume statistics partition record log parameter (see 6.4.10.2.3). The size field is the size of the PARTITION RECORD DATA COUNTER FIELD OF EACH VOLUME STATISTICS PARTITION RECORD DESCRIPTOR.
M - Mount history log parameter (see 6.4.10.2.4)
Size Key:
v - variable size
Note:
If a parameter value is unknown but other parameter values are known, then parameter for the unknown value is not returned. If none of the values are known, then parameter 0000h is set to 00h.
Footnotes:
[a] Not supported on E06
[b] The value of this parameter may be unknown for some volume personalities. When this occurs, this parameter is not returned.
[c] The size field for a counter of Type P is the size of the PARTITION RECORD DATA COUNTER FIELD OF EACH VOLUME STATISTICS PARTITION RECORD DESCRIPTOR. The size of this counter depends on the number of partitions and is number_of_partitions * (4 bytes internal header + value in Size column).

Table 226 — Volume statistics log parameters (part 4 of 7)

Parameter Code	Description	Type	Size
0017h	**Total used native capacity {17h:0017h}:** The sum of the used native capacity of all partitions in megabytes (10^6 bytes) from BOP to EOD. A data counter value with all bytes set to FFh in the partition record data counter field (see 6.4.10.2.3) indicates that the total used native capacity is unknown	C	4
0018h-003Fh	Reserved		
0040h	**Volume serial number {17h:0040h}:** The volume serial number parameter contains the value from the serial number field of the cartridge manufacturer's information page in the CM.	S	32
0041h[a]	**Tape lot identifier {17h:0041h}:** The tape lot identifier field may contain the tape pancake identifier value extracted from the servo writer manufacturer field in the media manufacturer's page of the CM or may be extracted from the manufacturer's word in the servo pattern.	S	8
0042h[a]	**Volume barcode {17h:0042h}:** The value from MAM attribute 0806h (i.e., BARCODE, see SPC-4)	S	32
0043h[a]	**Volume manufacturer {17h:0043h}:** The volume manufacturer parameter contains the value from the cartridge manufacturer field of the cartridge manufacturer's information page of the CM.	S	8
0044h[a]	**Volume license code {17h:0044h}:** ASCII code to represent the license under which this volume was manufactured. The volume license code parameter contains the value from the cartridge license code parameter of the cartridge manufacturer's information page of the CM.	S	4
0045h[a]	**Volume personality {17h:0045h}:** The volume personality parameter contains an ASCII string to identify the combination of physical volume type and density formatted on the medium. The intent is to provide a designator sufficient for successful volume interchange . The value is in the format "3592"+<Volume label cartridge type (see bytes 81 and 82 of sense data)>+<identifier of written density>. Identifiers used for the written format are: 3592A1 density - G1 3592A2 density - G2 3592A3 density - G3 3592A4 density - G4 3592A5 density - G5 3592B5 density - A5 3592A6 density - G6 As an example, a volume that is a JD cartridge written in 3592B5 density is "3592JDA5" The <identifier of written density> is the density written not the generation of drive.	S	8

Type Key:
C - Volume statistics data counter log parameter (see 6.4.10.2.1)
S - Volume statistics string data log parameter (see 6.4.10.2.2)
P - Volume statistics partition record log parameter (see 6.4.10.2.3). The size field is the size of the PARTITION RECORD DATA COUNTER FIELD OF EACH VOLUME STATISTICS PARTITION RECORD DESCRIPTOR.
M - Mount history log parameter (see 6.4.10.2.4)

Size Key:
v - variable size

Note:
If a parameter value is unknown but other parameter values are known, then parameter for the unknown value is not returned. If none of the values are known, then parameter 0000h is set to 00h.

Footnotes:

[a] Not supported on E06

[b] The value of this parameter may be unknown for some volume personalities. When this occurs, this parameter is not returned.

[c] The size field for a counter of Type P is the size of the PARTITION RECORD DATA COUNTER FIELD OF EACH VOLUME STATISTICS PARTITION RECORD DESCRIPTOR. The size of this counter depends on the number of partitions and is number_of_partitions * (4 bytes internal header + value in Size column).

Table 226 — Volume statistics log parameters (part 5 of 7)

Parameter Code	Description	Type	Size
0046h	**Volume manufacture date {17h:0046h}:** The volume manufacture date parameter contains the date of manufacture of the volume. The format is YYYYMMDD (i.e., four numeric ASCII characters for the year followed by two numeric ASCII characters for the month followed by two numeric ASCII characters for the day with no intervening spaces).	S	8
0047h-007Fh	Reserved		
0080h[a]	**Write protect {17h:0080h}:** An indication if the volume mounted is write protected. a) A value of 01h indicates that a write protection state that is persistent with this volume is set; b) A value of 00h indicates that no write protection state that is persistent with this volume is set; and c) A value of FFh indicates that it is unknown if there is a persistent write protection condition set.	C	1
0081h[a]	**WORM {17h:0081h}:** An indication if the mounted volume is a WORM volume. a) A value of 01h indicates that the volume is a WORM volume; b) A value of 00h indicates that the volume is not a WORM volume; and c) A value of FFh indicates that it is unknown if the volume is a WORM volume.	C	1
0082h[a]	**Maximum recommended tape path temperature exceeded {17h:0082h}:** An indication if the recommended tape path temperature has been exceeded at some point in the past. a) A value of 01h indicates a drive has detected at some point in the past that the maximum recommended tape path temperature has been exceeded; b) A value of 00h indicates no drive has detected at any point in the past that the maximum recommended tape path temperature has been exceeded; and c) A value of FFh indicates that it is not known if at any point in the past that the maximum recommended tape path temperature has been exceeded.	C	1
0083h-00FFh	Reserved		
0100h[a, b]	**Volume write mounts {17h:}:** The volume write mounts parameter contains a count of the number of mounts in which logical objects on the medium were written or modified.	C	4
0101h[a, b]	**Beginning of medium passes {17h:0101h}:** Count of the total number of times the beginning of medium position has passed over the head. The value from the LP3 passes field of the usage information page of the CM.	C	4

Type Key:
C - Volume statistics data counter log parameter (see 6.4.10.2.1)
S - Volume statistics string data log parameter (see 6.4.10.2.2)
P - Volume statistics partition record log parameter (see 6.4.10.2.3). The size field is the size of the PARTITION RECORD DATA COUNTER FIELD OF EACH VOLUME STATISTICS PARTITION RECORD DESCRIPTOR.
M - Mount history log parameter (see 6.4.10.2.4)

Size Key:
v - variable size

Note:
If a parameter value is unknown but other parameter values are known, then parameter for the unknown value is not returned. If none of the values are known, then parameter 0000h is set to 00h.

Footnotes:

[a] Not supported on E06

[b] The value of this parameter may be unknown for some volume personalities. When this occurs, this parameter is not returned.

[c] The size field for a counter of Type P is the size of the PARTITION RECORD DATA COUNTER FIELD OF EACH VOLUME STATISTICS PARTITION RECORD DESCRIPTOR. The size of this counter depends on the number of partitions and is number_of_partitions * (4 bytes internal header + value in Size column).

Table 226 — Volume statistics log parameters (part 6 of 7)

Parameter Code	Description	Type	Size
0102h[a, b]	**Middle of tape passes {17h:0102h}:** count of the total number of times that the physical middle of the user data region on the tape has passed over the head (e.g., (EOM – BOM) ÷ 2). The value from the (LP5 – LP3) ÷ 2 passes field of the usage information page of the CM.	C	4
0103h-01FFh	Reserved		
0200h[a]	**First encrypted logical object identifier(s) {17h:0200h}:** The logical object identifier(s) for the first logical object on the medium which has been encrypted for each partition on the medium. Each byte in the partition record data counter field(s) is set to FFh if there are no encrypted logical objects or if the partition does not exist. The least significant byte in the partition record data counter field (see 6.4.10.2.3) is set to FEh and all other bytes is set to FFh if it is not known if there are encrypted logical objects on the medium.	P	6[c]
0201h[a]	**First unencrypted logical object on the EOP side of the first encrypted logical object identifier(s) {17h:0201h}:** The logical object identifiers for the first logical object on the medium which is not encrypted and is on the EOP side of the first encrypted logical object for each partition on the medium. Each byte in the partition record data counter field (see 6.4.10.2.3) is set to FFh if there are no unencrypted logical objects on the EOP side of the first encrypted logical object identifier or if the partition does not exist. The least significant byte in the partition record data counter field is set to FEh and all other bytes is set to FFh if: a) it is not known if there are unencrypted logical objects on the EOP side of the first encrypted logical object; b) all bytes in the first encrypted logical object identifier are set to FFh; or c) the least significant byte in the first encrypted logical object identifier is set to FEh and all other bytes are set to FFh.	P	6[c]
0202h[a]	**Approximate native capacity of partition(s) {17h:0202h}:** The native capacity of the partition(s) in megabytes (10^6 bytes) from BOP to EOP. A data counter value with all bytes set to FFh in the partition record data counter field (see 6.4.10.2.3) indicates that the native capacity of the partition is unknown.	P	4[c]
0203h[a]	**Approximate used native capacity of partition(s) {17h:0203h}:** The used native capacity of the partition in megabytes (10^6 bytes) from BOP to EOD. A data counter value with all bytes set to FFh in the partition record data counter field (see 6.4.10.2.3) indicates that the used native capacity of the partition is unknown.	P	4[c]

Type Key:
C - Volume statistics data counter log parameter (see 6.4.10.2.1)
S - Volume statistics string data log parameter (see 6.4.10.2.2)
P - Volume statistics partition record log parameter (see 6.4.10.2.3). The size field is the size of the PARTITION RECORD DATA COUNTER FIELD OF EACH VOLUME STATISTICS PARTITION RECORD DESCRIPTOR.
M - Mount history log parameter (see 6.4.10.2.4)

Size Key:
v - variable size

Note:
If a parameter value is unknown but other parameter values are known, then parameter for the unknown value is not returned. If none of the values are known, then parameter 0000h is set to 00h.

Footnotes:

[a] Not supported on E06

[b] The value of this parameter may be unknown for some volume personalities. When this occurs, this parameter is not returned.

[c] The size field for a counter of Type P is the size of the PARTITION RECORD DATA COUNTER FIELD OF EACH VOLUME STATISTICS PARTITION RECORD DESCRIPTOR. The size of this counter depends on the number of partitions and is number_of_partitions * (4 bytes internal header + value in Size column).

Table 226 — Volume statistics log parameters (part 7 of 7)

Parameter Code	Description	Type	Size
0204h[a]	**Approximate remaining native capacity to early warning of partition(s) {17h:0204h}:** The approximate remaining native capacity of the partition(s) in megabytes (10^6 bytes) that is less than or equal to the native capacity from EOD to EW. The value reported in this parameter shall be zero once EOD is at or beyond EW. A data counter value with all bytes set to FFh in the PARTITION RECORD DATA COUNTER field (see 6.4.10.2.3) indicates that the remaining native capacity of the partition is unknown.	P	4[c]
0205h-02FFh	Reserved		
0300h	**Mount History {17h:0300h}:** The mount history parameter contains the T10 vendor identification and unit serial number of the most recent devices into which this volume was mounted.	M	v
0301h-EFFFh	Reserved		
F000h-FFFFh	Vendor specific		

Type Key:
C - Volume statistics data counter log parameter (see 6.4.10.2.1)
S - Volume statistics string data log parameter (see 6.4.10.2.2)
P - Volume statistics partition record log parameter (see 6.4.10.2.3). The size field is the size of the PARTITION RECORD DATA COUNTER FIELD OF EACH VOLUME STATISTICS PARTITION RECORD DESCRIPTOR.
M - Mount history log parameter (see 6.4.10.2.4)
Size Key:
v - variable size
Note:
If a parameter value is unknown but other parameter values are known, then parameter for the unknown value is not returned. If none of the values are known, then parameter 0000h is set to 00h.
Footnotes:
[a] Not supported on E06
[b] The value of this parameter may be unknown for some volume personalities. When this occurs, this parameter is not returned.
[c] The size field for a counter of Type P is the size of the PARTITION RECORD DATA COUNTER FIELD OF EACH VOLUME STATISTICS PARTITION RECORD DESCRIPTOR. The size of this counter depends on the number of partitions and is number_of_partitions * (4 bytes internal header + value in Size column).

6.4.10.2 Parameter formats

6.4.10.2.1 Data counter log parameter format

The volume statistics data counter log parameter is used for reporting parameters specified as volume statistics data counter log parameters in Table 226, Volume statistics log parameters. The volume statistics data counter log parameter format is specified in Table 227, Volume statistics data counter log parameter format.

Table 227 — Volume statistics data counter log parameter format

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	(MSB)	PARAMETER CODE						
1								(LSB)
2	DU	OBSOLETE	TSD (0B)	ETC(0B)	TMC		FORMAT AND LINKING (11B)	
3	PARAMETER LENGTH (N-3)							
4	(MSB)	VOLUME STATISTICS DATA COUNTER						
n								(LSB)

The PARAMETER CODE field is defined in Table 226, Volume statistics log parameters.

See SPC-4 for descriptions of the DU bit, TSD bit, ETC bit, TMC field and FORMAT AND LINKING field. The TSD bit and FORMAT AND LINKING field is set to the values specified in Table 227, Volume statistics data counter log parameter format.

The PARAMETER LENGTH field indicates the number of bytes in the VOLUME STATISTICS DATA COUNTER field that follows.

The VOLUME STATISTICS DATA COUNTER field is the value of the data counter associated with the parameter code.

6.4.10.2.2 Volume statistics string data log parameter format

The volume statistics string data log parameter is used for reporting parameters specified as volume statistics string data log parameters in Table 226, Volume statistics log parameters. The volume statistics string data log parameter format is specified in Table 228, Volume statistics string data log parameter format. The volume statistics string data log parameter shall be a multiple of 4 bytes.

Table 228 — Volume statistics string data log parameter format

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	(MSB)	PARAMETER CODE						
1								(LSB)
2	DU(0B)	OBSOLETE	TSD (0B)	ETC(0B)	TMC(00B)		FORMAT AND LINKING (01B)	
3	PARAMETER LENGTH (N-3)							
4	STRING DATA							
n								

The PARAMETER CODE field is defined in Table 226, Volume statistics log parameters.

See SPC-4 for descriptions of the DU bit, TSD bit, ETC bit, TMC field and FORMAT AND LINKING field. These fields shall be set to the values specified in Table 228, Volume statistics string data log parameter format.

The PARAMETER LENGTH field indicates the number of bytes of data that follows.

The STRING DATA field contains an ASCII string describing the volume statistics parameter specified by the PARAMETER CODE value. The STRING DATA field is an ASCII data field (see SPC-4).

6.4.10.2.3 Volume statistics partition record log parameter format

The volume statistics partition record log parameter is used for reporting parameters specified as volume statistics partition record log parameters in Table 226, Volume statistics log parameters. The volume statistics partition record log parameter format is specified in Table 229, Volume statistics partition log parameter format.

Table 229 — Volume statistics partition log parameter format

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	(MSB)	PARAMETER CODE						
1								(LSB)
2	DU	OBSOLETE	TSD (0B)	ETC(0B)	TMC(00B)		FORMAT AND LINKING (11B)	
3	PARAMETER LENGTH (N-3)							
	VOLUME STATISTICS PARTITION RECORD DESCRIPTOR(S)							
4	VOLUME STATISTICS PARTITION RECORD DESCRIPTOR [FIRST]							
	. . .							
n	VOLUME STATISTICS PARTITION RECORD DESCRIPTOR [LAST]							

The PARAMETER CODE field is defined in Table 226, Volume statistics log parameters.

See SPC-4 for descriptions of the DU bit, TSD bit, ETC bit, TMC field and FORMAT AND LINKING field. The TSD bit and FORMAT AND LINKING field shall be set to the values specified in Table 229, Volume statistics partition log parameter format.

The PARAMETER LENGTH field indicates the number of bytes in the volume statistics partition record log descriptors.

The volume statistics partition record descriptor format is specified in Table 230, Volume statistics partition record descriptor format.

Table 230 — Volume statistics partition record descriptor format

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	PARTITION RECORD DESCRIPTOR LENGTH (n-1)							
1	Reserved							
2	(MSB)	PARTITION NUMBER						
3								(LSB)
4	(MSB)	PARTITION RECORD DATA COUNTER						
n								(LSB)

The PARTITION RECORD DESCRIPTOR LENGTH field specifies the number of bytes that follow.

The PARTITION NUMBER field indicates the number of the partition that the following counter is associated with.

The PARTITION RECORD DATA COUNTER field is the value of the data counter associated with the parameter code and associated with the specified partition.

6.4.10.2.4 Mount history

The mount history parameter contains the T10 vendor identification and unit serial number of the most recent devices into which this volume was mounted. The mount history log parameter format is specified in table 229.

Table 231 — Mount history log parameter format

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	(MSB)	PARAMETER CODE						
1								(LSB)
2	DU(0b)	OBSOLETE	TSD (0b)	ETC(0b)	TMC(00b)		FORMAT AND LINKING (11b)	
3	PARAMETER LENGTH (N-3)							
	MOUNT HISTORY DESCRIPTOR(S)							
4	MOUNT HISTORY DESCRIPTOR [FIRST]							
	⋮							
n	MOUNT HISTORY DESCRIPTOR [LAST]							

The PARAMETER CODE field is defined in Table 226, Volume statistics log parameters.

See SPC-4 for descriptions of the DU bit, TSD bit, ETC bit, TMC field and FORMAT AND LINKING field. The TSD bit and FORMAT AND LINKING field shall be set to the values specified in table 229.

The PARAMETER LENGTH field indicates the number of bytes that follow in the mount history log descriptors.

There is one mount history descriptor for each of the last four devices into which the volume was mounted. If the volume has been mounted in fewer than four devices there will be fewer than four descriptors (i.e., a descriptor is only returned if the volume has been mounted in that many previous drives).

The mount history descriptor format is specified in table 230.

Table 232 — Mount history descriptor format

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	MOUNT HISTORY DESCRIPTOR LENGTH (n)							
1	Reserved							
2	(MSB)	MOUNT HISTORY INDEX						
3								(LSB)
4	(MSB)	MOUNT HISTORY VENDOR ID						
11								(LSB)
12	(MSB)	MOUNT HISTORY UNIT SERIAL NUMBER						
n								(LSB)

The MOUNT HISTORY DESCRIPTOR LENGTH field specifies the number of bytes that follow. The MOUNT HISTORY DESCRIPTOR LENGTH field shall contain a value that is a multiple of two.

The MOUNT HISTORY INDEX field indicates which previous mount was to the device specified in this descriptor. The value is incrementing starting at zero (e.g., The current device, or most recent device if not currently mounted, is numbered 0000h, the previous device is numbered 0001h, etc.).

The MOUNT HISTORY VENDOR ID field indicates the value returned in the T10 VENDOR IDENTIFICATION field of the standard inquiry data associated with this mount history index.

The MOUNT HISTORY UNIT SERIAL NUMBER field indicates the value of the PRODUCT SERIAL NUMBER field of the Unit Serial Number vital product data page associated with this mount history index. The MOUNT HISTORY UNIT SERIAL NUMBER field shall be right-aligned and the unused bytes shall be filled with ASCII space characters (i.e., 20h).

6.4.11 LP 2Eh: TapeAlerts

See LOG SELECT - 4Ch (see 5.2.12) and LOG SENSE - 4Dh (see 5.2.13) for directions on how to use this page. This page is used to track significant errors or conditions.

Byte	**Description**
0	
	Bit **Description**
	7-6 Reserved
	5-0 Page Code: 2Eh
1	Reserved
2-3	Page length: 0140h

Page Code 2Eh is used to report conditions in the tape drive. The TapeAlert log page is unique in that the parameter value is required to be 0000h or 0001h in the Log Sense command. A result of this requirement is that all parameters are requested, so the page length field is always returned 0140h.

6.4.11.1 Parameter Reset Behavior

WARNING WARNING

Log pages may have different reset behaviors. Care should be taken to ensure counters are processed appropriately, including consideration of their reset behaviors. While reset behaviors may be different between products, an attempt is made to maintain compatibility between generations of a given product. See “Log Parameters” on page 287.

WARNING WARNING

Specific flags may be cleared when corrective actions have removed the condition that caused the flag to be set (see table 233).

6.4.11.2 Parameter Definitions

There are 64 parameters numbered from 01h to 40h. However, for this product only parameters listed in table 233 have meaning. The remaining parameters always return a value of 00h.

All parameters are one byte long. Each parameter is either 00h to indicate the corresponding condition has not occurred or 01h to indicate that the condition has occurred. See “LP 30h: Tape Usage” on page 330 for parameter clearing behavior.

The following parameters are supported for each TapeAlert. Table 233 lists the supported TapeAlerts and additional fields as specified in SSC may be supported in the future.

Parameter description:

Byte	**Description**
0-1	Parameter Code
2	Parameter control byte
	Bit **Description**
	7 DU (Disable Update): 0b
	6 DS (Disable Save): 1b
	5 TSD (Target Save Disable): 0b
	4 ETC (Enable Threshold Comparison): 0b
	3-2 TMC (Threshold Met Comparison): 00b
	1 Reserved
	0 LO (List Parameter): 0b
3	Parameter length: 01h

4 TapeAlert Flag

Value	**Description**
01h	condition occurred
00h	condition did not occur

Table 233 — Supported TapeAlerts (part 1 of 2)

Code	Description	Set	Clear[1]	Type
0001h	**Read Warning {2Eh:0001h}:**		R	Warning
0002h	**Write Warning {2Eh:0002h}:**		R	Warning
0003h	**Hard Error {2Eh:0003h}:**		R	Warning
0004h	**Media {2Eh:0004h}:**		R	Critical
0005h	**Read Failure {2Eh:0005h}:**		R	Critical
0006h	**Write Failure {2Eh:0006h}:**		R	Critical
0007h	**Media Life {2Eh:0007h}:**		L	Warning
0008h	**Not Data Grade {2Eh:0008h}:**		R	Warning
0009h	**Write Protect {2Eh:0009h}:**		R	Critical
000Ah	**No Removal {2Eh:000Ah}:**		R	Informational
000Bh	**Cleaning Media {2Eh:000Bh}:**		R	Informational
000Ch	**Unsupported Format {2Eh:000Ch}:**		R	Informational
000Dh	**Recoverable Mechanical Cartridge Failure** {2Eh:000Dh}:		R	Critical
000Eh	**Unrecoverable Snapped Tape {2Eh:000Eh}:**		R	Critical
000Fh	**Memory Chip in Cartridge Failure {2Eh:000Fh}:**		R	Warning
0010h	**Forced Eject {2Eh:0010h}:**		L	Critical
0011h	**Read Only Format {2Eh:0011h}:**		R	Warning
0012h	**Tape Directory Corrupted {2Eh:0012h}:**		R	Warning
0013h	**Nearing Media Life {2Eh:0013h}:**		R	Informational
0014h	**Cleaning Required (clean for performance) {2Eh:0014h}:**		C	Critical
0015h	**Cleaning Requested (clean for usage) {2Eh:0015h}:**		C	Warning
0016h	**Expired Cleaning Media {2Eh:0016h}:**	C	C	Critical
0017h	**Invalid cleaning volume {2Eh:0017h}:**	C	L	Warning
0019h	**Host Channel Failure {2Eh:0019h}:**			Warning
001Ah	**Cooling Fan Failure {2Eh:001Ah}:**	S		Warning
001Bh	**Power Supply Failure {2Eh:001Bh}:**	S		Warning
001Eh	**Hardware A {2Eh:001Eh}:**			Critical
001Fh	**Hardware B {2Eh:001Fh}:**			Critical
0020h	**Interface {2Eh:0020h}:**			Warning
0021h	**Eject Media {2Eh:0021h}:**		U,R	Critical
0022h	**Download Fault {2Eh:0022h}:**			Warning
0024h	**Drive Temperature {2Eh:0024h}:**	S		Warning
0025h	**Drive Voltage {2Eh:0025h}:**	S		Warning
0026h	**Predictive Failure {2Eh:0026h}:**			Critical
0027h	**Diagnostics Required {2Eh:0027h}:**			Warning
0031h	**Diminished native capacity {2Eh:0031h}:**		L	Informational

Table 233 — Supported TapeAlerts (part 2 of 2)

Code	Description	Set	Clear[1]	Type
0032h	**Lost Statistics {2Eh:0032h}:**		R	Warning
0033h	**Tape Directory Invalid at Unload {2Eh:0033h}:**		L,R	Warning
0034h	**Tape System Area Write Failure {2Eh:0034h}:**		L,R	Critical
0035h	**Tape System Area Read Failure {2Eh:0035h}:**		R	Critical
0037h	**Loading Failure {2Eh:0037h}:**		R	Critical
0038h	**Unrecoverable Unload Failure {2Eh:0038h}:**		R	Critical
003Bh	**WORM Medium - Integrity Check Failed {2Eh:003Bh}:**		R	Warning
003Ch	**WORM Medium - Overwrite Attempted {2Eh:003Ch}:**		R	Warning

Legend

-	Not set/supported
L	Load - medium is loaded
C	Clean - cleaner tape is loaded
U	Unload - medium is ejected
E	Error - error code is posted
R	Removal - medium is FULLY removed
S	Sensor - sensor check

6.4.12 LP 30h: Tape Usage

See LOG SELECT - 4Ch (see 5.2.12) and LOG SENSE - 4Dh (see 5.2.13) for directions on how to use this page. This page contains non-volatile information relating to volume usage.

6.4.12.1 Parameter Definitions

WARNING WARNING

Log pages may have different reset behaviors. Care should be taken to ensure counters are processed appropriately, including consideration of their reset behaviors. While reset behaviors may be different between products, an attempt is made to maintain compatibility between generations of a given product. See “Log Parameters” on page 287.

WARNING WARNING

WARNING WARNING

The parameters in log pages should be dynamically parsed as some parameters may not be present, new parameters may be inserted, and parameter sizes may change. The listed size is for the most recent generation at the time of publication. Some sizes have changed from previous generations. The relative location of parameters have changed and are anticipated to continue to change..

WARNING WARNING

Table 234 — Tape Usage log parameter codes

Code	Counter: Description	Size
0001h	**Volume Mounts {30h:0001h}:** Number of mounts for the current volume	4
0002h	**Volume Datasets Written {30h:0002h}:** Total number of datasets written	8
0003h	**Volume Write Retries {30h:0003h}:** Total number of write retries	4
0004h	**Volume Write Perms {30h:0004h}:** Total write perms	2
0005h	**Volume Suspended Writes {30h:0005h}:**	2
0006h	**Volume Fatal Suspended Writes {30h:0006h}:**	2
0007h	**Volume Datasets Read {30h:0007h}:** Total number of datasets read	8
0008h	**Volume Read Retries {30h:0008h}:** Total number of read retries	4
0009h	**Volume Read Perms {30h:0009h}:** Total read perms	2
000Ah	**Volume Suspended Reads {30h:000Ah}:**	2
000Bh	**Volume Fatal Suspended Reads {30h:000Bh}:**	2
0100h	**Volume Write Mounts {30h:0100h}:** Number of Mounts in which the current volume was modified.	2
0101h	**Head Pass BOT Count {30h:0101h}:** Number of times the head passes logical beginning of tape (i.e., LP3)	4
0102h	**Head Pass MOT Count {30h:0102h}:** Number of times the head passes physical middle of tape (i.e., [LP3 + LP5]/2)	4

6.4.13 LP 31h: SIM/MIM

See LOG SELECT - 4Ch (see 5.2.12) and LOG SENSE - 4Dh (see 5.2.13) for directions on how to use this page. This page is designed to support SIMs (Service Information Messages), and MIMs (Medium Information Messages) reporting. SIMs and MIMs provide the initiator and operator details on service problems encountered by the device. The first 9 bytes are common to both the SIMs and the MIMs.

6.4.13.1 Parameter Reset Behavior

WARNING WARNING

Log pages may have different reset behaviors. Care should be taken to ensure counters are processed appropriately, including consideration of their reset behaviors. While reset behaviors may be different between products, an attempt is made to maintain compatibility between generations of a given product. See “Log Parameters” on page 287.

WARNING WARNING

When read with a Log Sense command, only the returned SIM or MIM is no longer available for subsequent retrieval. SIMs and MIMs are stored in and retrieved from a first-in first-out (FIFO) queue. Only 1 SIM or MIM is returned for each Log Sense command that requests Page Code 31h.

6.4.13.2 Parameter Definitions

6.4.13.2.1 SIM/MIM Header Data

Byte	Description	
0	Page Code	
	Bit	**Description**
	7-6	Reserved
	5-0	Page Code: 31h
1	Reserved	
2-3	Page Length: 0044h	
4-5	Parameter Code: 0000h	
6	Parameter control byte	
	Bit	**Description**
	7	DU (Disable Update): 0b
	6	DS (Disable Save): 1b
	5	TSD (Target Save Disable): 1b
	4	ETC (Enable Threshold Comparison): 0b
	3-2	TMC (Threshold Met Comparison): 00b
	1	Reserved
	0	LP (List Parameter): 1b
7	Parameter length: 40h	
8	SIM/MIM Indicator	
	Value	**Description**
	00h	Bytes 9 through 71 are invalid.
	01h	Bytes 9 through 71 are a SIM message. See “SIM Messages” on page 332.
	02h	Bytes 9 through 71 are a MIM message. See “MIM Messages” on page 333.
	03-FF	Bytes 9 through 71 are invalid.

6.4.13.2.2 SIM Messages

The following data are the parameters for the hardware SIM message:

Byte	**Description**
9-15	Vendor-Reserved
16-19	Engineering Data
20-21	SIM Message Code codes

Value (ASCII)	**Description**
‘00’	No Message
‘41’	Device Degraded. Call for Service
‘42;	Device Hardware Failure. Call for Service
‘43’	Service Circuits Failed, Operations Not Affected. Call for Service
‘55’	Drive Needs Cleaning. Load Cleaning Cartridge
‘57’	Drive Has Been Cleaned
All Others	Device Message message

Byte	**Description**
22-23	Vendor-Reserved
24	Exception Message Code

Value (ASCII)	**Description**
‘0’	Vendor-Reserved
‘1’	Effect of Failure Is Unknown
‘2’	Device Exception. No Performance Impact
‘3;	Exception on Host Interface xx See bytes 28-29 (Exception Data xx) in this SIM record for the xx value.
‘4’	Device Exception on ACF
‘5’	Device Exception on Operator Panel
‘6’	Device Exception on Tape Path
‘7’	Device Exception in Drive
‘8’	Cleaning Required
‘9’	Cleaning Done
‘A’-’F’	Vendor-Reserved

Byte	**Description**
25	Service Message Code

Value (ASCII)	**Description**
‘0’	Vendor-Reserved
‘1’	Repair Impact is Unknown
‘2’-’6’	Vendor-Reserved
‘7’	Repair Will Disable Access to Device serno (serno refers to serial number in bytes 52-63 of this SIM record)
‘8’	Repair Will Disable Message Display IDs on Device
‘9’	Clean Device
‘A’	Device Cleaned
‘B’	Device Cleaning For Performance Reasons is Required
‘C’-’F’	Vendor-Reserved

Byte	**Description**
26	Service Message Severity Code

Value (ASCII)	**Description**
‘0’	SIM severity code “Service”
‘1’	SIM severity code “Moderate”
‘2’	SIM severity code “Serious”
‘3’	SIM severity code “Acute”
‘4’-’9’, ‘A’-’F’	Vendor-Reserved

27 Vendor-Reserved

28-29 Exception Data

Interface Data is returned when byte 24 (Exception Message Code) in this SIM record contains the ASCII value ‘3’.

Value (ASCII)	**Description**
‘00’	interface 0 is indicated
‘01’	interface 1 is indicated

30-33 FRU Identifier

34-37 First FSC

38-41 Last FSC

42-45 Product ID: ‘8000’ (these 4 bytes define “TAPE”)

46-63 Product Identifier

Broken out into constituent bytes:

Byte	**Description**
46-48	Manufacturer: “IBM”
49-50	Plant of Manufacture
51	‘-’ (Dash symbol)
52-63	Serial Number

64-71 Device Type and Model Number (same as Inquiry Standard Data bytes 16-23)

6.4.13.2.3 MIM Messages

Medium Information Messages (MIMs) are supported by this device. The following data are the parameters for the MIM:

Byte **Description**

9-15 Vendor-Reserved

16-19 Expert Systems Data (microcode link level)

20-21 MIM Message Code codes

Value (ASCII)	**Description**
‘00’	No Message
‘60’	Bad Media, Read Only permitted
‘61’	Rewrite Media if possible
‘62’	Tape Directory Invalid. Re-read media if possible
‘64’	Bad Media-Cannot Read or Write
‘72’	Replace Cleaner Cartridge
Others	Vendor-Reserved

22-23 Engineering Data-First Failing Test

24 Exception Message Code

Value (ASCII)	**Description**
‘2’	Data Degraded
‘4’	Medium Degraded
‘6’	Block 0 Error
‘7’	Medium Exception
Others	Vendor-Reserved

25 Media Message Code

26 Media Message Modifier Severity Code

Value (ASCII)	**Description**
‘0’	Service
‘1’	Moderate-High Temp Read/Write Errors Detected
‘2’	Serious-Permanent Read/Write Errors Detected
‘3’	Acute-Block 0 Error
Others	Vendor-Reserved

27-29 Vendor-Reserved

30-33 Fault Symptom Code (FSC)

34-39 VOLID (in ASCII). Only valid if indicated by VOLID Valid Flag (byte 40)

40 VOLID Valid Flag

Value (ASCII)	**Description**
‘0’	VOLID (bytes 34-39) not valid
‘1’	VOLID valid, obtained from tape
‘3’	VOLID valid, obtained from library
Others	Vendor-Reserved for future use (odd number will always indicate VOLID valid)

41 Vendor-Reserved

46-63 Product Identifier

Broken out into constituent bytes:

Byte	**Description**
46-48	Manufacturer: “IBM”
49-50	Plant of Manufacture
51	‘-’ (Dash symbol)
52-63	Serial Number

64-71 Device Type and Model Number (same as Inquiry Standard Data bytes 16-23)

6.4.14 LP 32h: Write Errors

See LOG SELECT - 4Ch (see 5.2.12) and LOG SENSE - 4Dh (see 5.2.13) for directions on how to use this page. This page contains detailed counters related to write operations.

6.4.14.1 Parameter Definitions

NOTE 74 - When multiple errors occur on a dataset, the counter that is updated is generally based on the first error detected.

WARNING WARNING

Log pages may have different reset behaviors. Care should be taken to ensure counters are processed appropriately, including consideration of their reset behaviors. While reset behaviors may be different between products, an attempt is made to maintain compatibility between generations of a given product. See “Log Parameters” on page 287.

WARNING WARNING

WARNING WARNING

The parameters in log pages should be dynamically parsed as some parameters may not be present, new parameters may be inserted, and parameter sizes may change. The listed size is for the most recent generation at the time of publication. Some sizes have changed from previous generations. The relative location of parameters have changed and are anticipated to continue to change..

WARNING WARNING

Table 235 — Write Errors log parameter codes (part 1 of 2)

Code	Counter: Description	Size
0000h	**Datasets Corrected {32h:0000h}:** ECC is done by hardware. This is driven by an excessive CQs rewritten condition (TBD). Each count represents one dataset in error that was successfully corrected and written.	2
0001h	**Servo Transients {32h:0001h}:** ERP action was required because of a servo detected error and the first retry was successfully in place (stop write without backhitch, i.e, servo write skip). Each count represents one dataset in error that was successfully recovered and written.	2
0002h	**Data Transients {32h:0002h}:** ERP action was required because of a readback check or ECC detected error and the first retry was successfully in place (no backhitch). Each count represents one dataset in error that was successfully recovered and written.	2
0003h	**Velocity Events {32h0003h:}:** A velocity control problem occurred. Each count represents one occurrence, not just the count of affected datasets. Counts may include occurrences from both temporary and permanent errors.	2
0004h	**Servo Acquisition Temps {32h:0004h}:** A servo error (servo dropout or off-track shutdown) was detected while trying to acquire a DSS or dataset at the beginning of a write append sequence (motion); ERP action was required, and servo transient condition criteria were not met. Each count represents one dataset in error that was successfully recovered and written.	2
0005h	**Data Acquisition Temps {32h:0005h}:** During read-back check, the read channel failed to acquire a DSS or dataset at the beginning of a write append sequence and no servo error was reported, ERP action was required, and read/ECC transient condition criteria were not met. Each count represents one dataset in error that was successfully recovered and written.	2
0006h	**Servo Temps {32h:0006h}:** A servo error (servo dropout or off-track shutdown) was detected while writing data, ERP action was required, and servo transient condition criteria were not met. Each count represents one dataset in error that was successfully recovered and written.	2

Table 235 — Write Errors log parameter codes (part 2 of 2)

Code	Counter: Description	Size
0007h	**Data Temps {32h:0007h}:** An uncorrectable error, CRC error, instantaneous speed variation (ISV) error, or no ending burst error occurred during readback check of a dataset, and no servo error was reported; ERP action was required, and readback/ECC transient condition criteria were not met. Each count represents one dataset in error that was successfully recovered and written.	2
0008h	**Total Retries {32h:0008h}:** The count of the total number of ERP actions. Each count represents one occurrence, not just one time per dataset. Counts may include occurrences from both temporary and permanent errors.	2
0009h	**Vendor-Reserved {32h:0009h}:**	2
000Ah	**Match Filter ERP (write mode use) {32h:000Ah}:**	2
000Bh	**Servo Skip Events {32h:000Bh}:** The count of long servo write skips, extended DSS or long spaces between datasets written. This is generally servo write skips, but may also include other write scenarios. Each count represents one occurrence, not one count per block. Counts may include occurrences from both temporary and permanent errors.	2
000Ch	**Housekeeping Events {32h:000Ch}:** The count of write problems in the Housekeeping Dataset Region. Each count represents one occurrence, not just one time per dataset. Counts may include occurrences from both temporary and permanent errors.	2
000Dh	**FID Events {32h:000Dh}:** The count of write problems while processing the FID. Each count represents one occurrence, not just one time per dataset. Counts may include occurrences from both temporary and permanent errors.	2
000Eh	**Vendor-Reserved (Blocks Lifted) {32h:000Eh}:**	2
000Fh	**Dataset Underrun {32h:000Fh}:** The number of times that the drive overran the buffer processing capability and had to stop and restart during a write. Each count represents one occurrence, not just one time per write.	2
0010h	**Vendor-Reserved {32h:0010h}:**	2
0011h	**Servo Position Events {32h:0011h}:** The number of servo detected positional compare discrepancies.	2

6.4.15 LP 34h: Read Forward Errors

See LOG SELECT - 4Ch (see 5.2.12) and LOG SENSE - 4Dh (see 5.2.13) for directions on how to use this page. This page contains detailed counters related to read operations.

6.4.15.1 Parameter Definitions

NOTE 75 - When multiple errors occur on a dataset, the counter that is updated is generally based on the first error detected. ERP counters indicate which specific ERP methods were successfully employed.

WARNING WARNING

Log pages may have different reset behaviors. Care should be taken to ensure counters are processed appropriately, including consideration of their reset behaviors. While reset behaviors may be different between products, an attempt is made to maintain compatibility between generations of a given product. See “Log Parameters” on page 287.

WARNING WARNING

WARNING WARNING

The parameters in log pages should be dynamically parsed as some parameters may not be present, new parameters may be inserted, and parameter sizes may change. The listed size is for the most recent generation at the time of publication. Some sizes have changed from previous generations. The relative location of parameters have changed and are anticipated to continue to change.

WARNING WARNING

Table 236 — Read Error Counters log parameter codes (part 1 of 2)

Code	Counter: Description	Size
0000h	**Datasets Corrected {34h:0000h}:** ECC is done by hardware. Each count represents one dataset in error that was successfully corrected and read.	2
0001h	**Servo Transients {34h:0001h}:** ERP action was required because of a servo detected error and the first retry was successfully in place. Each count represents one dataset in error that was successfully recovered and read.	2
0002h	**Data Transients {34h:0002h}:** ERP action was required because of a read channel or ECC detected error and the first retry was successfully in place. Each count represents one dataset in error that was successfully recovered and read.	2
0003h	**Velocity Events {34h:0003h}:** A velocity control problem occurred. Each count represents one occurrence, not just the count of affected datasets. Counts may include occurrences from both temporary and permanent errors.	2
0004h	**Servo Acquisition Temps {34h:0004h}:** A servo error (servo dropout or off track shutdown) was detected while trying to acquire an initial DSS or dataset, ERP action was required, and servo transient condition criteria were not met. Each count represents one dataset in error that was successfully recovered and read.	2
0005h	**Data Acquisition Temps {34h:0005h}:** The read channel failed to acquire an initial DSS or dataset, and no servo error was reported; ERP action was required, and read/ECC transient condition criteria were not met. Each count represents one dataset in error that was successfully recovered and read.	2
0006h	**Servo Temps {34h:0006h}:** A servo error (servo drop out) was detected while reading a dataset; ERP action was required, and servo transient condition criteria were not met. Each count represents one dataset in error that was successfully recovered and read.	2
0007h	**Data Temps {34h:0007h}:** An uncorrectable error, CRC error, or no ending burst error occurred while reading a dataset, and no servo error was reported; ERP action was required, and read/ECC transient condition criteria were not met. Each count represents one dataset in error that was successfully recovered and read.	2

Table 236 — Read Error Counters log parameter codes (part 2 of 2)

Code	Counter: Description	Size
0008h	**Sequence Errors {34h:0008h}:** A dataset number out of sequence was encountered, and no Servo or read/ECC error reported; ERP action was required, and no transient condition criteria were not met. Each count represents one dataset in error that was successfully recovered and read.	2
0009h	**Vendor-Reserved (ERP Read Opposite) {34h:0009h}:**	2
000Ah	**Vendor-Reserved (ERP Tension Adjust Hi) {34h:000Ah}:**	2
000Bh	**Vendor-Reserved (ERP Tension Adjust Lo) {34h:000Bh}:**	2
000Ch	**ERP Servo Adjust Hi {34h:000Ch}:** The dataset was recovered by reading with servo off-track variations. Each count represents one dataset in error that was successfully recovered and read.	2
000Dh	**ERP Servo Adjust Lo {34h:000Dh}:** The dataset was recovered by reading with servo off-track variations. Each count represents one dataset in error that was successfully recovered and read.	2
000Eh	**Vendor-Reserved (ERP Dead Reckon Nominal) {34h:000Eh}:**	2
000Fh	**Vendor-Reserved (ERP Dead Reckon Hi) {34h:000Fh}:**	2
0010h	**Servo AGA Gain ERP (read only mode) {34h:0010h}:**	2
0011h	**Vendor-Reserved (ERP Filter Coefficients) {34h:0011h}:**	2
0012h	**Servo Opposite Gap ERP (read only mode) {34h:0012h}:**	2
0013h	**Vendor-Reserved (ERP Dataflow Clock Adjust) {34h:0013h}:**	2
0014h	**Vendor-Reserved {34h:0014h}:**	2
0015h	**Total Retries {34h:0015h}:** The count of the total number of ERP actions. Each count represents one occurrence, not just one time per dataset. Counts may include occurrences from both temporary and permanent errors.	2
0016h	**Match Filter ERP (read mode use) {34h:0016h}:**	2
0017h	**Housekeeping Events {34h:0017h}:** The count of read problems in the Housekeeping Dataset Region. Each count represents one occurrence, not just one time per dataset. Counts may include occurrences from both temporary and permanent errors.	2
0018h	**Vendor-Reserved (Cartridge Init Errors) {34h:0018h}:**	2
0019h	**Dataset Overrun {34h:0019h}:** The number of times that the drive overran the buffer processing capability and had to stop and restart during a read. Each count represents one occurrence, not just one time per read.	2
001Ah	**Vendor-Reserved {34h:001Ah}:**	2
001Bh	**Servo Skip Events {34h:001Bh}:** The count of extended DSS or long spaces between datasets read. This may include servo write skips, but may also include other write scenarios. Each count represents one occurrence, not one count per block. Counts may include occurrences from both temporary and permanent errors.	2
001Ch	**Vendor-Reserved {34h:001Ch}:**	2
001Dh	**FID Events {34h:001Dh}:** The count of read problems while processing the FID. Each count represents one occurrence, not just one time per dataset. Counts may include occurrences from both temporary and permanent errors.	2
001Eh	**Servo Position Events {34h:001Eh}:** The number of servo detected positional compare discrepancies.	2

6.4.16 LP 36h: Read Backward Errors

See LOG SELECT - 4Ch (see 5.2.12) and LOG SENSE - 4Dh (see 5.2.13) for directions on how to use this page. This page contains parameters related to the device performing Read Backward operations rather than Read Forward operations. The read backwards (physical read reverse) function is not currently implemented in this device, but may be in the future. All host commands and function will not be affected by potential future implementation of this feature.

6.4.16.1 Parameter Definitions

WARNING WARNING

Log pages may have different reset behaviors. Care should be taken to ensure counters are processed appropriately, including consideration of their reset behaviors. While reset behaviors may be different between products, an attempt is made to maintain compatibility between generations of a given product. See "Log Parameters" on page 287.

WARNING WARNING

The parameters supported for "LP 36h: Read Backward Errors" on page 339 are identical to those described in "LP 34h: Read Forward Errors" on page 337, except that this data is recorded when the device is performing Read Backward operations rather than Read Forward operations.

6.4.17 LP 37h: Performance Characteristics (not J1A)

See LOG SELECT - 4Ch (see 5.2.12) and LOG SENSE - 4Dh (see 5.2.13) for directions on how to use this page. This page includes various performance and capacity measurements across the operation of the drive. Some fields are normalized qualitative measures while others are quantitative.

6.4.17.1 Parameter Reset Behavior

Each parameter has different reset characteristics which are described under the Subpage Code field description for bits 7-6 Scope.

6.4.17.2 Parameter Definitions

This page uses the Subpage Code mechanism (see 5.2.13) to select which groups of counters to return. This page has three scopes controlled by the Subpage Code field.

The subpage field in LOG SENSE - 4Dh and Log Parameter Format is used as follows:

Bit	**Description**
7-6	Scope

Value	**Description**
00b	Transient values: reset on Log Select [all Subpages are reset]
01b	Mount values: reset on load
10b	Lifetime values: reset on device power on or device reset (not target reset)
11b	Vendor-Reserved

5-4	Level

Value	**Description**
00b	Return summary counters
01b	Return basic counters
10b	Return advanced counters
11b	Return development counters

3-0	Group

Value	**Description**
0h	All groups
1h	Host Interface
2h	Buffer
3h	Medium
4h	Capacity
5h	Load/Unload
6h	Servo

The Subpage field in LOG SENSE - 4Dh may be set to 00h when the page code field is 37h. This operation will reset all group and local counters in the transient scope. Other scope cannot be explicitly reset.

The individual log subpage and parameter codes are described in the following table. Note that the counters which are returned depends on the Level and Group fields in the subpage. A group value of 0h will return all counters of a level less than or equal to that specified.

In the following tables, multiple counter codes may be represented by a single row. There will be an aspect symbol in the counter code such as 'p', 'q', 's', or '?'. The Aspect(s) column indicates which of the following values applies to the given code(s).

Aspect	**Definition**
p=0	primary interface (fibre) all ports (totals)
p=1	primary interface (fibre) port 0
p=2	primary interface (fibre) port 1
p=A	automation interface (RS-422)
q=1	Non-Ready: NOTE: These commands include ALL commands which are processed when the drive is in a Not Ready state.
q=2	Head-of-Queue: NOTE: These commands are commands which may be processed in any order. Such commands include: Inquiry, Report LUNs, Test Unit Ready and Request Sense. These counts are updated only when the drive is in a Ready state.
q=3	Read: NOTE: This aspect has more features detailed below.
q=4	Write: NOTE: This aspect has more features detailed below.
q=5	Sync: NOTE: This aspect has more features detailed below.
q=6	Seek: NOTE: This aspect has more features detailed below.
q=7	Non-Medium: NOTE: These commands are command issued to LUN 0 which are not in any other applicable category. These include many commands such as Log Sense, Log Select,

	Read Buffer, Reserve, etc. These counts are updated only when the drive is in a Ready state.
q=8	Non-LUN0, Non-Ready: NOTE: These commands include ALL commands which are processed by a LUN other than LUN 0 when the drive is in a Not Ready state.
q=9	Non-LUN0, Head-of-Queue: NOTE: These commands are commands which may be processed in any order. Such commands include: Inquiry, Report LUNs, Test Unit Ready and Request Sense. These counts are updated only when the drive is in a Ready state.
q=A	Non-LUN0: NOTE: These commands include any commands processed by a LUN other than LUN 0. These counts are updated only when the drive is in a Ready state.
s=1	Speed 1: Highest read/write speed
s=2	Speed 2: Second highest read/write speed
s=3	Speed 3: Third highest read/write speed
s=4	Speed 4: Fourth highest read/write speed
s=5	Speed 5: Fifth highest read/write speed
s=6	Speed 6: Sixth highest read/write speed
s=7	Speed 7: Seventh highest read/write speed
s=8	Speed 8: Eighth highest read/write speed
s=9	Speed 9: Ninth highest read/write speed
s=A	Speed 10: Tenth highest read/write speed
s=B	Speed 11: Eleventh highest read/write speed
s=C	Speed 12: Twelfth highest read/write speed
s=D	Speed 13: Thirteenth highest read/write speed
s=E	Speed 14: Fourteenth highest read/write speed
s=F	High speed locate (not read/write capable)
t=0	Partition 0
t=1	Partition 1
t=2	Partition 2
t=3	Partition 3
?=1	write - write phase without host holdoff. NOTE: if no (paused) data is supported for a particular counter, this aspect will include all write information
?=2	write (paused) - write phase while the host is being held off (buffer full)
?=3	read - read phase without host holdoff. NOTE: if no (paused) data is supported for a particular counter, this aspect will include all read information
?=4	read (paused) - read phase while the host is being held off (buffer empty)
?=5	position - during the processing of a seek operation
?=6	load - during the processing of an load operation
?=7	unload - during the processing of an unload operation
?=F	other - not in an above phase

WARNING WARNING

Log pages may have different reset behaviors. Care should be taken to ensure counters are processed appropriately, including consideration of their reset behaviors. While reset behaviors may be different between products, an attempt is made to maintain compatibility between generations of a given product. See “Log Parameters” on page 287.

WARNING WARNING

WARNING WARNING

The parameters in log pages should be dynamically parsed as some parameters may not be present, new parameters may be inserted, and parameter sizes may change. The listed size is for the most recent generation at the time of publication. Some sizes have changed from previous generations. The relative location of parameters have changed and are anticipated to continue to change.

WARNING WARNING

Table 237 — LP 37h: Performance Characteristics: Quality Summary

Code	Aspect(s)	Name: Description	Unit	Size	Level	Group
0000h		**Drive Efficiency {37h:0000h}:** Overall measure of the drive's condition. 00 is unknown, from 01 (best) to FF (worst)	rating	1	0	0
0001h		**Media Efficiency {37h:0001h}:** Overall measure of the currently mounted media's condition. X'00' is unknown, from X'01' (best) to X'FF' (worst)	rating	1	0	0
0010h		**Primary Interface Efficiency {37h:0010h}:** Overall measure of the interface (to the host) condition. 00 is unknown, from 01 (best) to FF(worst)	rating	1	0	0
0011h		**Primary Interface Port 0 Efficiency {37h:0011h}:** Overall measure of the per port interface (to the host) condition. X'00' is unknown, from X'01' (best) to X'FF' (worst)	rating	1	0	0
0012h		**Primary Interface Port 1 Efficiency {37h:0012j}:** Overall measure of the per port interface (to the host) condition. 00 is unknown, from 01(best) to FF (worst)	rating	1	0	0
001Ah		**Library Interface Efficiency {37h:001Ah}:** Overall measure of the interface (to the library) condition. 00 is unknown, from 01 (best) to FF (worst)	rating	1	0	0

Table 238 — LP 37h: Performance Characteristics: Device Usage (part 1 of 2)

Code	Aspect(s)	Name: Description	Unit	Size	Level	Group
0040h		**Time {37h:0040h}:** Amount of entire sample duration	msec	6	2	5
0041h		**Medium Empty Time {37h:0041h}:** Duration without a tape present	msec	6	2	5
0042h		**Medium Insert Time {37h:0042h}:** Duration from cartridge insert to load	msec	6	2	5
0043h		**Medium Mount Time {37h:0043h}:** Total time from start of cartridge load until cartridge ejected	msec	6	2	5
0044h		**Medium Load Time {37h:0044h}:** Total time from start of cartridge load to load complete (ready)	msec	6	2	5
0045h		**Medium Ready Time {37h:0045h}:** Total time from load complete (ready) to start of unload	msec	6	2	5
0046h		**Medium Eject Time {37h:0046h}:** Time from start of unload to unload complete	msec	6	2	5
0047h		**Medium Extract Time {37h:0047h}:** Time from cartridge unloaded to removed	msec	6	2	5
0048h		**Medium Dwell Time {37h:0048h}:** Time from cartridge unloaded to (re)loaded. Note - This may include time which cannot be determined as dwell or extract (when time is queried with a cartridge remaining in the unloaded position)	msec	6	2	5
0049h		**Medium Clean Time {37h:0049h}:** Time from cleaner recognized to eject complete	msec	6	2	5

Table 238 — LP 37h: Performance Characteristics: Device Usage (part 2 of 2)

Code	Aspect(s)	Name: Description	Unit	Size	Level	Group
0051h		**Medium Empty Count {37h:0051h}:** Number of times tape was fully removed	count	4	2	5
0052h		**Medium Insert Count {37h:0052h}:** Number of cartridge insertions to load position detected	count	4	2	5
0053h		**Medium Mount Count {37h:0053h}:** Number of mount operations	count	4	2	5
0054h		**Medium Load Count {37h:0054h}:** Number of load operations	count	4	2	5
0055h		**Medium Ready Count {37h:0055h}:** Number of ready transitions	count	4	2	5
0056h		**Medium Eject Count {37h:0056h}:** Number of unloads	count	4	2	5
0057h		**Medium Extract Count {37h:0057h}:** Number of times tape was extracted	count	4	2	5
0058h		**Medium Dwell Count {37h:0058h}:** Number of times tape was reloaded (from unload)	count	4	2	5
0059h		**Medium Clean Count {37h:0059h}:** Number of recognized cleaner loads (does not indicate successful cleans, tape may be expired)	count	4	2	5

Table 239 — LP 37h: Performance Characteristics: Host Commands (part 1 of 4)

Code	Aspect(s)	Name: Description	Unit	Size	Level	Group
0q00h	12(3456)789A	**Count {37h:0q00h}:**	count	4	2	1
0q01h	12(3456)789A	**Timing {37h:0q01h}:**	msec	6	2	1
0q02h	12(3456)789A	**Relative Time {37h:0q01h}:**	% * 65536	4	1	1
0q04h	12(3456)789A	**Transfer Count to Host (in) {37h:0q04h}:**	count	4	2	1
0q05h	12(3456)789A	**Transfer Byte Count to Host (in) {37h:0q05h}:**	bytes	8	2	1
0q06h	12(3456)789A	**Transfer Timing to Host (in) {37h:0q06h}:**	msec	6	2	1
0q08h	12(4)789A	**Transfer Count from Host (out) {37h:0q08h}:**	count	4	2	1
0q09h	12(4)789A	**Transfer Byte Count from Host (out) {37h:0q09h}:**	bytes	8	2	1
0q0Ah	12(4)789A	**Transfer Timing from Host (out) {37h:0q0Ah}:**	msec	6	2	1
Note - Read type host commands include Read, Verify and Read Reverse (not all of these may be supported). These counts are updated only when the drive is in a Ready state						
0300h		**Read Count {37h:0300h}:** Number of blocks processed to the host by read type commands	blocks	6	2	1
0301h		**Read Timing {37h:0301h}:** Amount of time processing read type commands. Note - Due to device specific performance path resources, this may not reflect the actual time spent processing commands, but may reflect the amount of time where read commands could be processed.	msec	6	2	1
0302h		**Read Relative Time {37h:0302h}:** Ratio of time spent reading with respect to Medium Ready Time	% * 65536	4	1	1
0304h		**Transfer Count to Host (in) {37h:0304h}:**	count	6	2	1

Table 239 — LP 37h: Performance Characteristics: Host Commands (part 2 of 4)

Code	Aspect(s)	Name: Description	Unit	Size	Level	Group
0305h		**Transfer Byte Count to Host (in) {37h:0305h}:**	bytes	8	2	1
0306h		**Transfer Timing to Host (in) {37h:0306h}:**	msec	6	2	1
03D0h		**Read Performance Efficiency {37h:03D0h}:** Ratio of performance read type commands with respect to all read type commands	% * 65536	4	1	1
03D4h		**Read Filemark (perf) Relative Time {37h:03D4h}:** Amount of time spent sending filemark encountered status to the host with respect to time Read Timing.	% * 65536	4	2	1
Note - Write type host commands include Write and Write Filemarks [not including any synchronization portion]. These counts are updated only when the drive is in a Ready state.						
0400h		**Write Count {37h:0400h}:** Number of blocks processed from the host by write type commands	blocks	6	2	1
0401h		**Write Timing {37h:0401h}:** Amount of time processing write type commands. Note - Due to device specific performance path resources, this may not reflect the actual time spent processing commands, but may reflect the amount of time where write commands could be processed.	msec	6	2	1
0402h		**Write Relative Time {37h:0402h}:** Ratio of time spent writing with respect to Medium Ready Time	% * 65536	4	1	1
0404h		**Transfer Count to Host (in) {37h:0404h}:**	count	6	2	1
0405h		**Transfer Byte Count to Host (in) {37h:0405h}:**	bytes	8	2	1
0406h		**Transfer Timing to Host (in) {37h:0406h}:**	msec	6	2	1
0408h		**Transfer Count from Host (out) {37h:0408h}:**	count	6	2	1
0409h		**Transfer Byte Count from Host (out) {37h:0409h}:**	bytes	8	2	1
040Ah		**Transfer Timing from Host (out) {37h:040Ah}:**	msec	6	2	1
04D0h		**Write Performance Efficiency {37h:04D0h}:** Ratio of performance write commands with respect to all write type commands	% * 65536	4	1	1
04D4h		**Write Filemark Relative Time {37h:04D4j}:** Amount of time spent writing filemarks.	% * 65536	4	2	1
Note - Sync type host commands include Write Filemarks [non-immediate]. Implicit sync type commands include mode changes while writing, non-buffered mode and idle time based syncs. These counts are updated only when the drive is in a Ready state.						
0500h		**Sync Count [Host] {37h:0500h}:** Number of host sync operations (non-immediate Write Filemarks, non-buffered writes)	count	4	2	1
0501h		**Sync Timing [Host] {37h:0501h}:** Amount of time processing host sync commands	msec	6	2	1
0502h		**Sync Relative Time [Host] {37h:0502h}:** Ratio of time spent processing host sync commands with respect to Medium Ready Time	% * 65536	4	1	1
0504h		**Transfer Count to Host (in) {37h:0504h}:**	count	4	2	1
0505h		**Transfer Byte Count to Host (in) {37h:0505h}:**	bytes	8	2	1
0506h		**Transfer Timing to Host (in) {37h:0506h}:**	msec	6	2	1

Table 239 — LP 37h: Performance Characteristics: Host Commands (part 3 of 4)

Code	Aspect(s)	Name: Description	Unit	Size	Level	Group
05D1h		**Sync Count [Implicit] {37h:05D1h}:** Number of implicit sync commands (time based flushes, mode change flushes)	count	4	2	1
05D2h		**Sync Timing [Implicit] {37h:05D2h}:** Amount of time processing implicit sync commands	msec	6	2	1
05D4h		**Sync Relative Time [Implicit] {37h:05D4h}:** Ratio of time spent processing implicit sync commands with respect to Medium Ready Time	% * 65536	4	1	1
Note - Seek type host commands include Space, Locate, and Rewind. These counts are updated only when the drive is in a Ready state.						
0600h		**Seek Count {37h:0600h}:** Number of positioning host commands	count	4	2	1
0601h		**Seek Timing {37h:0601h}:** Amount of time spent processing host positioning commands	msec	6	2	1
0602h		**Seek Relative Time {37h:0602h}:** Ratio of time spend processing host seek commands with respect to Medium Ready Time	% * 65536	4	1	1
0604h		**Transfer Count to Host (in) {37h:0604h}:**	count	4	2	1
0605h		**Transfer Byte Count to Host (in) {37h:0605h}:**	bytes	8	2	1
0606h		**Transfer Timing to Host (in) {37h:0606h}:**	msec	6	2	1
06D1h		**Seek Block Count {37h:06D1h}:** Number of blocks processed in host positioning commands	blocks	4	2	1
0Cp0h	012A	**Command Count {37h:0Cp0h}:**	count	6	2	1
0Cp1h	012A	**Command Timing {37h:0Cp1h}:**	msec	6	2	1
0Cp2h	012A	**Command Relative Time {37h:0Cp2h}:**	% * 65536	4	2	1
0Cp4h	012A	**Command Transfer Count to Host (in) {37h:0Cp4h}:**	count	6	2	1
0Cp5h	012A	**Command Transfer Byte Count to Host (in) {37h:0Cp5h}:**	bytes	8	2	1
0Cp6h	012A	**Command Transfer Timing to Host (in) {37h:0Cp6h}:**	msec	6	2	1
0Cp8h	012A	**Command Transfer Count from Host (out) {37h:0Cp8h}:**	count	6	2	1
0Cp9h	012A	**Command Transfer Byte Count from Host (out) {37h:0Cp9h}:**	bytes	8	2	1
0CpAh	012A	**Command Transfer Timing from Host (out) {37h:0CpAh}:**	msec	6	2	1
0CpCh	012A	**Command Queue Count {37h:0CpCh}:**	count	6	2	1
0CpDh	012A	**Command Queue Latency {37h:0CpDh}:**	msec	6	2	1
0CpEh	012A	**Command Queue Relative Time {37h:0CpEh}:**	% * 65536	4	2	1
0Dp1h	12A	**Port Throughput Rate Maximum Bursting {37h:0Dp1h}:**	bytes/sec	6	2	1
0Dp2h	12A	**Port Throughput Rate Maximum Sustained {37h:0Dp2h}:**	bytes/sec	6	2	1
0Dp3h	12A	**Port Throughput Rate {37h:0Dp3h}:**	bytes/sec	6	2	1
0Dp4h	12A	**Port Throughput Efficiency {37h:0Dp4h}:**	% * 65536	4	1	1

Table 239 — LP 37h: Performance Characteristics: Host Commands (part 4 of 4)

Code	Aspect(s)	Name: Description	Unit	Size	Level	Group
0Dp7h	12A	**Port Rate Changes {37h:0Dp7h}:**	count	4	2	1
0DF0h		**Average Command Latency {37h:0DF0h}:** Average amount of time commands were queued waiting for execution with respect to all commands (including unqueued commands which processed immediately).	usec	4	1	1
0DF1h		**Average Dequeue Latency {37h:0DF1h}:** Average amount of time commands were queued waiting for execution with respect to commands which where queued (not processed immediately).	usec	4	1	1
0DF8h		**Long Queue Latency Count [>1 sec] {37h:0DF8h}:**	count	4	2	1
0DF9h		**Long Queue Latency Count [>10 sec] {37h:0DF9h}:**	count	4	2	1
0DFAh		**Long Queue Latency Count [>100 sec] {37h:0DFAh}:**	count	4	2	1
0DFBh		**Long Queue Latency Count [>1000 sec] {37h:0DFBh}:**	count	4	2	1

Table 240 — LP 37h: Performance Characteristics: Host Initiators

Code	Aspect(s)	Name: Description	Unit	Size	Level	Group
Note - In dual port configurations it is possible for the same host to be identified as a different initiator when using a different logical or physical path. This can occur in failover or load balancing applications.						
0E00h		**Active Initiator Count {37h:0E00h}:** Number of initiators which processed one or more commands.	count	4	2	1
0E01h		**Primary Initiator {37h:0E01h}:** Ratio of commands issued by the initiator which is issuing the most commands with respect to all initiators.	% * 65536	4	2	1
0E02h		**Secondary Initiator {37h:0E02h}:** Ratio of commands issued by the initiator which is issuing the second most commands with respect to all initiators.	% * 65536	4	2	1
0E03h		**Current Initiator {37h:0E03h}:** Ratio of commands issued by this (the querying) initiator with respect to all initiators.	% * 65536	4	2	1

Table 241 — LP 37h: Performance Characteristics: Host Recovery (by port)

Code	Aspect(s)	Name: Description	Unit	Size	Level	Group
0Fp0h	12A	**Transfer Recoveries [by port] {37h:0Fp0h}:**	count	4	2	1
0Fp1h	12A	**Transfer Recover Time [by port] {37h:0Fp1h}:**	msec	6	2	1
0Fp2h	12A	**Resource Recoveries [by port] {37h:0Fp2h}:**	count	4	2	1
0Fp3h	12A	**Reset Count [by port] {37h:0Fp3h}:**	count	4	2	1
0Fp8h	12A	**Abort Count [by port] {37h:0Fp8h}:**	count	4	2	1
0Fp9h	12A	**Abort Time [by port] {37h:0Fp9h}:**	msec	6	2	1

Table 242 — LP 37h: Performance Characteristics: Mode Phase Timing Windows (part 1 of 3)

Code	Aspect(s)	Name: Description	Unit	Size	Level	Group
1000h		**Write Cycles {37h:1000h}:**	count	4	2	2
1001h		**Write Pauses {37h:1001h}:**	count	4	2	2
1010h		**Write Cycle Time {37h:1010h}:**	msec	6	2	2
1020h		**Write Cycle Relative Time {37h:1020h}:** Ratio of time in write mode with respect to Medium Ready Time.	% * 65536	4	1	2
1021h		**Write Setup Relative Time {37h:1021h}:**	% * 65536	4	2	2
1022h		**Write Ready Relative Time {37h:1022h}:**	% * 65536	4	1	2
1023h		**Write Pause Relative Time {37h:1023h}:**	% * 65536	4	1	2
1024h		**Write Exit Relative Time {37h:1024h}:**	% * 65536	4	2	2
1200h		**Read Cycles {37h:1200h}:**	count	4	2	2
1201h		**Read Pauses {37h:1201h}:**	count	4	2	2
1210h		**Read Cycle Time {37h:1210h}:**	msec	6	2	2
1214h		**Read Exit Time {37h:1214h}:**	msec	6	3	2
1215h		**Read Traverse EM Time {37h:1215h}:**	msec	6	3	2
1220h		**Read Cycle Relative Time {37h:1220h}:** Ratio of time in write mode with respect to Medium Ready Time.	% * 65536	4	1	2
1221h		**Read Setup Relative Time {37h:1221h}:**	% * 65536	4	2	2
1222h		**Read Ready Relative Time {37h:1222h}:**	% * 65536	4	1	2
1223h		**Read Pause Relative Time {37h:1223h}:**	% * 65536	4	1	2
1224h		**Read Exit Relative Time {37h:1224h}:**	% * 65536	4	2	2
1225h		**Read Traverse EM Relative Time {37h:1225h}:**	% * 65536	4	2	2
1400h		**Position Count {37h:1400h}:**	count	4	2	2
1410h		**Position Time {37h:1410h}:**	msec	6	2	2
1420h		**Position Relative Time {37h:1420h}:** Ratio of time spent physically and logically positioning with respect to Medium Ready Time.	% * 65536	4	1	2
1430h		**Position Relative Rate {37h:1430h}:**	bytes/sec	4	1	2
1480h		**Position Count (Media) {37h:1480h}:**	count	4	2	2
1490h		**Position Time (Media) {37h:1490h}:**	msec	6	2	2
14A0h		**Position Relative Time (Media) {37h:14A0h}:** Ratio of time spent physically positioning media with respect to Medium Ready Time.	% * 65536	4	1	2
14B0h		**Position Relative Rate (Media) {37h:14B0h}:**	bytes/sec	4	1	2
14F0h		**Position Buffer Hits {37h:14F0h}:** Ratio of positioning operations where targets were already present in the buffer.	% * 65536	4	1	2
1500h		**Flush Count {37h:1500h}:** Number of low level buffer write flush operations. These may include operations which only affect the buffer and do not involve media motion.	count	4	1	2
1510h		**Flush Time {37h:1510h}:** Time spend executing operations counted by Flush Count.	msec	6	2	2

Table 242 — LP 37h: Performance Characteristics: Mode Phase Timing Windows (part 2 of 3)

Code	Aspect(s)	Name: Description	Unit	Size	Level	Group
1520h		**Flush Relative Time {37h:1520h}:** Ratio of time spent flushing with respect to Medium Ready Time.	% * 65536	4	1	2
1580h		**Flush Count (Media) {37h:1580h}:** Number of low level buffer write flush operations which involve [or continue] media motion.	count	4	1	2
1590h		**Flush Time (Media) {37h:1590h}:** Time spent executing operations counted by Flush Count (Media).	msec	6	2	2
15A0h		**Flush Relative Time (Media) {37h:15A0h}:** Ratio of time spent flushing to media with respect to Write Cycle Time.	% * 65536	4	1	2
15F0h		**Flush Buffer Hits {37h:15F0h}:** Ratio of flush operations which required media motion.	% * 65536	4	1	2
2000h		**Media Idle {37h:2000h}:**	msec	6	2	3
2001h		**Media Write {37h:2001h}:**	msec	6	2	3
2002h		**Media Read {37h:2002h}:**	msec	6	2	3
2003h		**Media Erase {37h:2003h}:**	msec	6	2	3
2004h		**Media Position {37h:2004h}:**	msec	6	2	3
20?0h	1234567F	**Media Phase Timing {37h:20?0h}:**	msec	6	2	3
20?1h	1234567F	**Media Phase Cycles {37h:20?1h}:**	count	4	2	3
2?10h	1234567F	**Wrap Change Count {37h:2?10h}:** Total number of wrap changes.	count	4	2	3
2?11h	1234567F	**Band Change Count {37h:2?11h}:** Total number of changes to different servo bands.	count	4	2	3
2?50h	13	**Datarate Performance Impacting ERPs {37h:2?50h}:**	% * 65536	4	1	3
2?51h	13567F	**Performance Impacting ERPs {37h:2?51h}:**	% * 65536	4	1	3
2?52h	1234567F	**Performance Impact by ERPs {37h:2?52h}:**	% * 65536	4	1	3
2?60h	135	**Uncompressed Data {37h:2?60h}:**	bytes	8	2	2
2?61h	135	**Compressed Data {37h:2?61h}:**	bytes	8	2	2
2?62h	135	**Padded Data {37h:2?62h}:**	bytes	8	2	2
2?63h	135	**Degate Data {37h:2?63h}:**	bytes	8	2	2
2?68h	135	**Datasets Processed {37h:2?68h}:**	datasets	4	2	2
2?6Ch	13	**Compression Ratio {37h:2?6Ch}:**	% * 65536	4	1	2
2?71h	13F	**Compressed Data (Medium) {37h:2?71h}:**	bytes	8	2	3
2?72h	13F	**Padded Data (Medium) {37h:2?72h}:**	bytes	8	2	3
2?80h	13	**Maximum Host Transfer Rate {37h:2?80h}:**	bytes/sec	4	1	2
2?81h	13	**Average Host Transfer Rate {37h:2?81h}:**	bytes/sec	4	1	2
2?82h	13	**Average Host Buffer Rate {37h:2?82h}:**	bytes/sec	4	1	2
2?83h	13	**Window Host Buffer Rate {37h:2?83h}:**	bytes/sec	4	1	2
2?84h	13	**Host Buffer Efficiency {37h:2?84h}:**	% * 65536	4	1	2
2?85h	13	**Window Buffer Efficiency {37h:2?85h}:**	% * 65536	4	1	2
2?88h	13	**Average Host Transfer Length {37h:2?88h}:**	bytes	4	2	2
2?8Ch	1	**Average Host Sync Length {37h:2?8Ch}:**	bytes	6	2	2

Table 242 — LP 37h: Performance Characteristics: Mode Phase Timing Windows (part 3 of 3)

Code	Aspect(s)	Name: Description	Unit	Size	Level	Group
2?90h	13	**Maximum Comp Transfer Rate {37h:2?90h}:**	bytes/sec	4	1	2
2?91h	13	**Average Comp Transfer Rate {37h:2?91h}:**	bytes/sec	4	1	2
2?92h	13	**Average Comp Buffer Rate {37h:2?92h}:**	bytes/sec	4	1	2
2?93h	13	**Window Comp Buffer Rate {37h:2?93h}:**	bytes/sec	4	1	2
2?94h	13	**Comp Buffer Efficiency {37h:2?94h}:**	% * 65536	4	1	2
2?95h	13	**Window Comp Buffer Efficiency {37h:2?95h}:**	% * 65536	4	1	2
2?98h	13	**Average Comp Transfer Length {37h:2?98h}:**	bytes	4	2	2
2?9Ch	1	**Average Comp Sync Length {37h:2?9Ch}:**	bytes	6	2	2
2?A0h	13	**Maximum Tape Transfer Rate {37h:2?A0h}:**	bytes/sec	4	1	2
2?A1h	13	**Average Tape Buffer Rate {37h:2?A1h}:**	bytes/sec	4	2	2
2?A2h	13	**Window Tape Buffer Rate {37h:2?A2h}:**	bytes/sec	4	2	2
2?A3h	13	**Moving Tape Buffer Rate {37h:2?A3h}:**	bytes/sec	4	2	2
2?A4h	13	**Window Tape Buffer Efficiency {37h:2?A4h}:**	% * 65536	4	2	2
2?A5h	13	**Moving Tape Buffer Efficiency {37h:2?A5h}:**	% * 65536	4	2	2
2?A6h	13	**Tape Buffer Efficiency {37h:2?A6h}:** Ratio of amount of time we are usefully moving and ready with respect to amount of time the buffer is able to process data. A ratio larger than 1 indicates the compressed host data is arriving faster than the native device rate. Lower values indicate the device has under utilized host bandwidth.	% * 65536	4	2	2
2?A7h	13F	**Tape Thrashing {37h:2?A7h}:** Ratio of amount of time we are accelerating, decelerating or backhitching with respect to the time in mode.	% * 65536	4	2	2
2?A8h	13F	**Tape Efficiency {37h:2?A8h}:** Ratio of amount of time we are usefully moving and ready with respect to the time in mode.	% * 65536	4	1	2
2?F0h	13	**Speed Changes {37h:2?F0h}:**	count	4	2	2
2?F1h	13	**Speed Forced {37h:2?F1h}:**	count	4	2	2

Table 243 — LP 37h: Performance Characteristics: Servo Speed Characteristics

Code	Aspect(s)	Name: Description	Unit	Size	Level	Group
5Fs0h	123456789ABCDEF	**Servo Speed Relative Time {37h:5Fs0h}:**	% * 65536	4	2	6

Table 244 — LP 37h: Performance Characteristics: Static Capacity

Code	Aspect(s)	Name: Description	Unit	Size	Level	Group
7000h		**Static Capacity Efficiency {37h:7000h}:**	% * 65536	4	1	4
7010h		**Static Datasets Media {37h:7010h}:**	datasets	4	2	4

Table 244 — LP 37h: Performance Characteristics: Static Capacity

Code	Aspect(s)	Name: Description	Unit	Size	Level	Group
7011h		**Static Datasets Used {37h:7011h}:**	datasets	4	2	4
7020h		**Static Distance Media {37h:7020h}:**	mm	8	2	4
7021h		**Static Distance Used {37h:7021h}:**	mm	8	2	4
7030h		**Static Remaining Capacity in SkipSync Buffer {37h:7030h}:**	% * 65536	4	2	4

Table 245 — LP 37h: Performance Characteristics: Active Capacity

Code	Aspect(s)	Name: Description	Unit	Size	Level	Group
7?00h	13F	**Active Capacity Efficiency {37h:7?00h}:**	% * 65536	4	1	4
7?10h	13F	**Active Sync Loss {37h:7?10h}:**	% * 65536	4	2	4
7?11h	13F	**Active Skip Loss {37h:7?11h}:**	% * 65536	4	2	4
7?12h	13F	**Active DSS Loss {37h:7?12h}:**	% * 65536	4	2	4
7?13h	13F	**Active CQs Loss (on-the-fly) {37h:7?13h}:**	% * 65536	4	2	4
7?21h	13F	**Active Distance Skip {37h:7?21h}:**	mm	8	2	4
7?22h	13F	**Active Distance DSS {37h:7?22h}:**	mm	8	2	4
7?23h	13F	**Active Distance CQs (on-the-fly) {37h:7?23h}:**	mm	8	2	4
7?2Fh	13F	**Active Distance Total {37h:7?2Fh}:**	mm	8	2	4

Table 246 — Log Page 37h: Performance Characteristics: Static Capacity per Partition

Code	Aspect(s)	Name: Description	Unit	Size	Level	Group
8t00h	01	**Static Capacity Efficiency (Partition t) {37h:8t00h}:**	% * 65536	4	1	4
8t10h	01	**Static Datasets Media (Partition t) {37h:8t10h}:**	datasets	4	2	4
8t11h	01	**Static Datasets Used (Partition t) {37h:8t11h}:**	datasets	4	2	4
8t20h	01	**Static Distance Media (Partition t) {37h:8t20h}:**	mm	8	2	4
8t21h	01	**Static Distance Used (Partition t) {37h:8t21h}:**	mm	8	2	4

6.4.18 LP 38h: Blocks/Bytes Transferred

See LOG SELECT - 4Ch (see 5.2.12) and LOG SENSE - 4Dh (see 5.2.13) for directions on how to use this page.

6.4.18.1 Parameter Definitions

WARNING WARNING

Log pages may have different reset behaviors. Care should be taken to ensure counters are processed appropriately, including consideration of their reset behaviors. While reset behaviors may be different between products, an attempt is made to maintain compatibility between generations of a given product. See “Log Parameters” on page 287.

WARNING WARNING

WARNING WARNING

The parameters in log pages should be dynamically parsed as some parameters may not be present, new parameters may be inserted, and parameter sizes may change. The listed size is for the most recent generation at the time of publication. Some sizes have changed from previous generations. The relative location of parameters have changed and are anticipated to continue to change..

WARNING WARNING

Table 247 — LP 38h: Blocks/Bytes Transferred log parameter codes (part 1 of 2)

Code	Counter: Description	Size
0000h	**Host Write Blocks Processed {38h:0000h}:** Each count represents a block processed across the host interface during a Write. The count does not include ERP retries.	8
0001h	**Host Write Kibibytes Processed {38h:0001h}:** Each count represents a kibibyte (2^{10}) processed across the host interface during a Write. The count does not include ERP retries. This count may be divided by Device Write Kibibytes Processed, 0005, to calculate an approximate write compression ratio.	6
0002h	**Host Read Blocks Processed {38h:0002h}:** Each count represents a block processed across the host interface during a Read. The count does not include ERP retries.	8
0003h	**Host Read Kibibytes Processed {38h:0003h}:** Each count represents a kibibyte (2^{10}) processed across the host interface during a Read. The count does not include ERP retries. This count may be divided by Device Read Kibibytes Processed, 0007, to calculate an approximate read compression ratio.	6
0004h	**Device Write Datasets Processed {38h:0004h}:** Each count represents a dataset processed on the medium. The count does not include ERP retries.	4
0005h	**Device Write Kibibytes Processed {38h:0005h}:** Each count represents a kibibyte (2^{10}) processed on the medium. The count does not include ERP retries or any tape formatting overhead bytes.	6
0006h	**Device Read Datasets Processed {38h:0006h}:** Each count represents a dataset processed from the medium. The count does not include ERP retries.	4
0007h	**Device Read Kibibytes Processed {38h:0007h}:** Each count represents a kibibyte (2^{10}) processed from the medium. The count does not include ERP retries or any tape formatting overhead bytes.	6
0008h	**Device Write Datasets Transferred {38h:0008h}:** Each count represents a dataset processed on the medium. The count includes ERP retries.	4
0009h	**Device Write Kibibytes Transferred {38h:0009h}:** Each count represents a kibibyte (2^{10}) processed on the medium. The count includes ERP retries and any tape formatting overhead bytes.	6
000Ah	**Device Read Datasets Transferred {38h:000Ah}:** Each count represents a dataset processed from the medium. The count includes ERP retries.	4

Table 247 — LP 38h: Blocks/Bytes Transferred log parameter codes (part 2 of 2)

Code	Counter: Description	Size
000Bh	**Device Read Kibibytes Transferred {38h:000Bh}:** Each count represents a kibibyte (2^{10}) processed from the medium. The count includes ERP retries and any tape formatting overhead bytes.	6
000Ch	**Nominal Capacity of Partition {38h:000Ch}:** The nominal capacity of the current partition in kibibytes (2^{10}).	8
000Dh	**Fraction of Partition Traversed {38h:000Dh}:** The fractional part of the current partition traversed (N/255).	1
000Eh	**Nominal Capacity of Volume {38h:000Eh}:** The nominal capacity of the mounted volume in kibibytes (2^{10}). This is determined by the sum of the Nominal Capacity of Partition parameter for each partition.	8
000Fh	**Fraction of Volume Traversed {38h:000Fh}:** The fractional part of the mounted volume traversed (N/255). This reports the value that would be reported by the Fraction of Partition Traversed if the volume were a single partition.	1
0010h	**Remaining Capacity of Volume {38h:0010h}:** The nominal unwritten remaining capacity of the mounted volume in kibibytes(2^{10}). This is not sensitive to current position. This is determined by the sum of the Remaining Capacity of Partition for each partition. If the tape does not have a valid EOD in at least one existing partition, or if a tape is not loaded, then a value of ’all ones’ (-1) is returned.	8
0011h	**Remaining Capacity of Partition {38h:0011h}:** The nominal unwritten remaining capacity of the current partition in kibibytes(2^{10}). This is not sensitive to current position. If the tape does not have a valid EOD in the current partition, or if a tape is not loaded, then a value of ’all ones’ (-1) is returned.	8

6.4.19 LP 39h: Host Port 0 Interface Errors

See LOG SELECT - 4Ch (see 5.2.12) and LOG SENSE - 4Dh (see 5.2.13) for directions on how to use this page. The count of errors occurring on host interface port 0 (while the device is active on the interface).

6.4.19.1 Parameter Definitions

WARNING WARNING

Log pages may have different reset behaviors. Care should be taken to ensure counters are processed appropriately, including consideration of their reset behaviors. While reset behaviors may be different between products, an attempt is made to maintain compatibility between generations of a given product. See “Log Parameters” on page 287.

WARNING WARNING

WARNING WARNING

The parameters in log pages should be dynamically parsed as some parameters may not be present, new parameters may be inserted, and parameter sizes may change. The listed size is for the most recent generation at the time of publication. Some sizes have changed from previous generations. The relative location of parameters have changed and are anticipated to continue to change..

WARNING WARNING

Table 248 — LP 39h: Host Port Interface Errors log parameter codes

Code	Counter: Description	Size
0000h	**Host Protocol Errors {39h:0000h}:** Each count represents one occurrence. Counts may include occurrences from both temporary and permanent errors.	2
0007h	**Host Aborts {39h:0007h}:** Each count represents one occurrence. Counts may include occurrences from both temporary and permanent errors.	2
0008h	**Host Resets {39h:0008h}:** Each count represents one occurrence. Counts may include occurrences from both temporary and permanent errors.	2
0009h	**Vendor-Reserved {39h:0009h}:**	2
000Ah	**Vendor-Reserved {39h:000Ah}:**	2
0010h	**Host Recoveries {39h:0010h}:** Each count represents one occurrence. Counts may include occurrences from both temporary and permanent errors. An example of a host recoveries is a Fibre Channel Sequence Retransmission Request (SRR).	4

6.4.20 LP 3Ah: Host Port 1 Interface Errors

See LOG SELECT - 4Ch (see 5.2.12) and LOG SENSE - 4Dh (see 5.2.13) for directions on how to use this page. The count of errors occurring on host port 1 (while the device is active on the interface).

6.4.20.1 Parameter Definitions

NOTE 76 - The parameters are identical to those found in "LP 39h: Host Port 0 Interface Errors" on page 354, except this data is recorded for incidents which occur on host interface port 1.

WARNING WARNING

Log pages may have different reset behaviors. Care should be taken to ensure counters are processed appropriately, including consideration of their reset behaviors. While reset behaviors may be different between products, an attempt is made to maintain compatibility between generations of a given product. See "Log Parameters" on page 287.

WARNING WARNING

WARNING WARNING

The parameters in log pages should be dynamically parsed as some parameters may not be present, new parameters may be inserted, and parameter sizes may change. The listed size is for the most recent generation at the time of publication. Some sizes have changed from previous generations. The relative location of parameters have changed and are anticipated to continue to change..

WARNING WARNING

6.4.21 LP 3Bh: Equipment Check Errors

See LOG SELECT - 4Ch (see 5.2.12) and LOG SENSE - 4Dh (see 5.2.13) for directions on how to use this page. The following counters all deal with errors where a problem in the drive hardware is suspected (as opposed to media processing or host interface quality related problems).

6.4.21.1 Parameter Definitions

WARNING WARNING

Log pages may have different reset behaviors. Care should be taken to ensure counters are processed appropriately, including consideration of their reset behaviors. While reset behaviors may be different between products, an attempt is made to maintain compatibility between generations of a given product. See "Log Parameters" on page 287.

WARNING WARNING

WARNING WARNING

The parameters in log pages should be dynamically parsed as some parameters may not be present, new parameters may be inserted, and parameter sizes may change. The listed size is for the most recent generation at the time of publication. Some sizes have changed from previous generations. The relative location of parameters have changed and are anticipated to continue to change..

WARNING WARNING

Table 249 — LP 3Bh: Equipment Check Errors log parameter codes

Code	Counter: Description	Size
0001h	**Panel Errors {3Bh:0001h}:** Each count represents one occurrence. Counts may include occurrences from both temporary and permanent errors.	2
0002h	**Host Protocol Chip Errors {3Bh:0002h}:** Each count represents one occurrence. Counts may include occurrences from both temporary and permanent errors.	2
0003h	**Host Buffer Errors {3Bh:0003h}:** Each count represents one occurrence. Counts may include occurrences from both temporary and permanent errors.	2
0004h	**Data Compression Errors {3Bh:0004h}:** Each count represents one occurrence. Counts may include occurrences from both temporary and permanent errors.	2
0005h	**Format Buffer Errors {3Bh:0005h}:** Each count represents one occurrence. Counts may include occurrences from both temporary and permanent errors.	2
0006h	**Dataflow Hardware Errors {3Bh:0006h}:** Each count represents one occurrence. Counts may include occurrences from both temporary and permanent errors.	2
0007h	**ECC Hardware Errors {3Bh:0007h}:** Each count represents one occurrence. Counts may include occurrences from both temporary and permanent errors.	2
0008h	**Channel Hardware Errors {3Bh:0008h}:** Each count represents one occurrence. Counts may include occurrences from both temporary and permanent errors.	2
0009h	**Internal Interface Errors {3Bh:0009h}:** Each count represents one occurrence. Counts may include occurrences from both temporary and permanent errors.	2
000Ah	**Loader Hardware Errors {3Bh:000Ah}:** Each count represents one occurrence. Counts may include occurrences from both temporary and permanent errors.	2

6.4.22 LP 3Ch: Drive Control Statistics

See LOG SELECT - 4Ch (see 5.2.12) and LOG SENSE - 4Dh (see 5.2.13) for directions on how to use this page. This page is for special drive control mode statistics. While is is included in the list of supported pages (page code 00h), this page is not intended for general use.

6.4.22.1 Parameter Definitions

WARNING WARNING

Log pages may have different reset behaviors. Care should be taken to ensure counters are processed appropriately, including consideration of their reset behaviors. While reset behaviors may be different between products, an attempt is made to maintain compatibility between generations of a given product. See “Log Parameters” on page 287.

WARNING WARNING

WARNING WARNING

The parameters in log pages should be dynamically parsed as some parameters may not be present, new parameters may be inserted, and parameter sizes may change. The listed size is for the most recent generation at the time of publication. Some sizes have changed from previous generations. The relative location of parameters have changed and are anticipated to continue to change.

WARNING WARNING

6.4.23 LP 3Dh: Subsystem Statistics

See LOG SELECT - 4Ch (see 5.2.12) and LOG SENSE - 4Dh (see 5.2.13) for directions on how to use this page. The following counters all deal with subsystem statistics and errors.

6.4.23.1 Parameter Reset Behavior

WARNING WARNING

Log pages may have different reset behaviors. Care should be taken to ensure counters are processed appropriately, including consideration of their reset behaviors. While reset behaviors may be different between products, an attempt is made to maintain compatibility between generations of a given product. See "Log Parameters" on page 287.

WARNING WARNING

Most of the counters on this page are never reset. Most counters are maintained in VPD and persist across Log Select, Log Sense, Power On Resets, and even microcode download. Lifetime values are written to VPD every eight operating hours when the drive is in a not ready state. The user may also save these counters to VPD at other times (such as just prior to powering off) by selecting the *Save* option from the CE service panel Statistics menu. The counters lock at maximum values.

6.4.23.2 Parameter Definitions

WARNING WARNING

The parameters in log pages should be dynamically parsed as some parameters may not be present, new parameters may be inserted, and parameter sizes may change. The listed size is for the most recent generation at the time of publication. Some sizes have changed from previous generations. The relative location of parameters have changed and are anticipated to continue to change..

WARNING WARNING

Table 250 — LP 3Dh: Subsystem Statistics log parameter codes (part 1 of 3)

Code	Counter: Description	Size
0020h	**Volume Lifetime Mounts {3Dh:0020h}:** The total number of successful cartridge unloads performed during the lifetime of a cartridge. This field may not be updated for mounts that occur with the volume physically write-protected.	4
0021h	**Volume Lifetime Megabytes Written {3Dh:0021h}:** The total amount of data in Megabytes (10^6) written during the lifetime of the cartridge. On each unload, an approximate value is calculated and stored by rounding up to the nearest Megabyte. These bytes are counted as they are processed to the medium (compressed bytes), not at the host interface. This field may not be updated during mounts that occur with the volume physically write-protected.	8
0022h	**Volume Lifetime Megabytes Read {3Dh:0022h}:** The total amount of data in Megabytes (10^6) read during the lifetime of the cartridge. On each unload, an approximate value is calculated and stored by rounding up to the nearest Megabyte. These bytes are counted as they are processed from the medium (compressed bytes), not at the host interface. This field may not be updated during mounts that occur with the volume physically write-protected.	8
0040h	**Drive Lifetime Mounts {3Dh:0040h}:** The total number of successful cartridge unloads performed during the lifetime of the drive.	4
0041h	**Drive Lifetime Megabytes Written {3Dh:0041h}:** The total amount of data in Megabytes (10^6) written during the lifetime of the drive. On each unload, an approximate value is calculated and stored by rounding up to the nearest Megabyte. These bytes are counted as they are processed to the medium (compressed bytes), not at the host interface.	8
0042h	**Drive Lifetime Megabytes Read {3Dh:0042h}:** The total amount of data in Megabytes (10^6) read during the lifetime of the drive. On each unload, an approximate value is calculated and stored by rounding up to the nearest Megabyte. These bytes are counted as they are processed from the medium (compressed bytes), not at the host interface.	8

Table 250 — LP 3Dh: Subsystem Statistics log parameter codes (part 2 of 3)

Code	Counter: Description	Size
0060h	**Clean Lifetime Mounts {3Dh:0060h}:** The total number of successful cleaner cartridge operations performed during the lifetime of the drive.	4
0061h	**Megabytes Written since Clean {3Dh:0061h}:** The total amount of data in Megabytes (10^6) written since the last successful clean operation. On each unload, an approximate value is calculated and stored by rounding up to the nearest Megabyte. These bytes are counted as they are processed to the medium (compressed bytes), not at the host interface.	8
0062h	**Megabytes Read since Clean {3Dh:0062h}:** The total amount of data in Megabytes (10^6) read since the last successful clean operation. On each unload, an approximate value is calculated and stored by rounding up to the nearest Megabyte. These bytes are counted as they are processed from the medium (compressed bytes), not at the host interface.	8
0063h	**Mounts since Clean {3Dh:0063h}:** The total number of mounts performed since the last successful clean operation.	4
0080h	**Library Interface Messages Received {3Dh:0080h}:** This counter is not stored in VPD and reflects messages since reset.	4
0081h	**Library Interface Messages Transmitted {3Dh:0081h}:** This counter is not stored in VPD and reflects message since reset.	4
0082h	**Library Interface Resets {3Dh:0082h}:** Count of hardware reset or logical reinitializations during normal operation.	4
0083h	**Library Interface Buffer Errors {3Dh:0083h}:** This includes buffer overrun or underrun conditions.	4
0084h	**Library Interface Sync Errors {3Dh:0084h}:**	4
0085h	**Library Interface Framing Errors {3Dh:0085h}:**	4
0086h	**Library Interface Protocol Errors {3Dh:0086h}:**	4
0087h	**Library Interface Logical Errors {3Dh:0087h}:**	4
0088h	**Library Interface Loader Failures {3Dh:0088h}:** This counter reflects load attempts when the drive is in an incorrect state or was otherwise unable to attempt requested loader action.	4
0090h	**Drive Lifetime Write Perms {3Dh:0090h}:** Total number of write permanent errors which occurred on this drive.	4
0091h	**Drive Lifetime Read Perms {3Dh:0091h}:** Total number of read permanent errors which occurred on this drive.	4
0092h	**Drive Lifetime Load Perms {3Dh:0092h}:** Total number of load permanent errors which occurred on this drive.	4
0093h	**Drive Lifetime Unload Perms {3Dh:0093h}:** Total number of unload permanent errors which occurred on this drive.	4
00A0h	**Drive Lifetime Write Temps {3Dh:00A0h}:** Total number of write temporary errors which occurred on this drive. Note - This is not a count of ERP actions taken, and certain transient errors may not be included in this count.	4
00A1h	**Drive Lifetime Read Temps {3Dh:00A1h}:** Total number of read temporary errors which occurred on this drive. Note - This is not a count of ERP actions taken, and certain transient errors may not be included in this count.	4
00A2h	**Drive Lifetime Load Temps {3Dh:00A2h}:** Total number of load temporary errors which occurred on this drive. Note - This is not a count of ERP actions taken, and certain transient errors may not be included in this count.	4

Table 250 — LP 3Dh: Subsystem Statistics log parameter codes (part 3 of 3)

Code	Counter: Description	Size
00A3h	**Drive Lifetime Unload Temps {3Dh:00A3h}:** Total number of unload temporary errors which occurred on this drive. Note - This is not a count of ERP actions taken, and certain transient errors may not be included in this count.	4
0100h	**Lifetime Power On Seconds {3Dh:0100h}:** Cumulative number of seconds which the drive has been powered on. Note - Since this time is only periodically updated in non-volatile storage, it is possible that this time may not be entirely accurate to the full resolution of the counter.	4
0101h	**Power On Seconds {3Dh:0101h}:** Number of seconds since the drive was powered on or has undergone a hard reset condition.	4
0102h	**Reset Seconds {3Dh:0102h}:** Number of seconds since the drive has undergone a soft reset condition.	4

6.4.24 LP 3Eh: Engineering Use

See LOG SELECT - 4Ch (see 5.2.12) and LOG SENSE - 4Dh (see 5.2.13) for directions on how to use this page. This page is for engineering use only and is not included in the list of supported pages (i.e., page code 00h). As such, the counters on this page are not intended for general use.

6.4.24.1 Parameter Definitions

WARNING WARNING

Log pages may have different reset behaviors. Care should be taken to ensure counters are processed appropriately, including consideration of their reset behaviors. While reset behaviors may be different between products, an attempt is made to maintain compatibility between generations of a given product. See “Log Parameters” on page 287.

WARNING WARNING

WARNING WARNING

The parameters in log pages should be dynamically parsed as some parameters may not be present, new parameters may be inserted, and parameter sizes may change. The listed size is for the most recent generation at the time of publication. Some sizes have changed from previous generations. The relative location of parameters have changed and are anticipated to continue to change..

WARNING WARNING

6.5 Medium auxiliary memory attributes (MAM)

6.5.1 Attribute format

Each medium auxiliary memory attribute shall be communicated between the application client and device server in the format shown in table 251. This format shall be used in the parameter data for the WRITE ATTRIBUTE command (see 5.2.49) and the READ ATTRIBUTE command (see 5.2.13). The attribute format in this standard implies nothing about the physical representation of an attribute in the medium auxiliary memory.

Table 251 — MAM ATTRIBUTE format

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	(MSB)	ATTRIBUTE IDENTIFIER						
1								(LSB)
2	READ ONLY	Reserved					FORMAT	
3	(MSB)	ATTRIBUTE LENGTH (n-4)						
4								(LSB)
5	ATTRIBUTE VALUE							
n								

The ATTRIBUTE IDENTIFIER field contains a code value identifying the attribute (see 6.5.2).

The READ ONLY bit indicates whether the attribute is in the read only state (see 4.23). If the READ ONLY bit is set to one, the attribute is in the read only state. If the READ ONLY bit is set to zero, the attribute is in the read/write state.

The FORMAT field (see table 252) specifies the format of the data in the ATTRIBUTE VALUE field.

Table 252 — MAM attribute FORMAT field

Format	Name	Description
00b	BINARY	The ATTRIBUTE VALUE field contains binary data.
01b	ASCII	The ATTRIBUTE VALUE field contains left-aligned ASCII data.
10b	TEXT	The attribute contains textual data. The character set is as described in the TEXT LOCALIZATION IDENTIFIER attribute (see 6.5.2.4.6).
11b		Reserved

The ATTRIBUTE LENGTH field specifies the length in bytes of the ATTRIBUTE VALUE field.

The ATTRIBUTE VALUE field contains the current value, for the READ ATTRIBUTE - 8Ch (beginning with E07) (see 5.2.21 on page 141) command, or intended value, for the WRITE ATTRIBUTE - 8Dh (beginning with E07) (see 5.2.49 on page 217) command, of the attribute.

6.5.2 Attribute identifier values

6.5.2.1 Attribute identifier values overview

The values in the ATTRIBUTE IDENTIFIER field (see 6.5.1) are assigned according to the attribute type (see 4.23) and whether the attribute is standard or vendor specific (see table 253).

Table 253 — MAM attribute identifier range assignments

Attribute Identifiers	Attribute Type	Standardized	Subclause
0000h to 03FFh	Device	Yes	6.5.2.2
0400h to 07FFh	Medium	Yes	6.5.2.3
0800h to 0BFFh	Host	Yes	6.5.2.4
0C00h to 0FFFh	Device	Vendor specific	
1000h to 13FFh	Medium	Vendor specific	
1400h to 17FEh	Host	Vendor specific	6.5.2.5
17FFh	Vendor-reserved		
1800h to FFFFh	Reserved		

Device servers may accept and process a WRITE ATTRIBUTE command containing standardized host type attribute identifier values (i.e., 0800h-0BFFh) or vendor specific host type attribute identifier values (i.e., 1400h-17FFh). Standardized host type attribute identifier values may be checked as described in 6.5.2.4.

6.5.2.2 Device type attributes

Device type attributes (see table 254) shall be maintained and updated by the device server when the medium and associated medium auxiliary memory are present. All supported medium type attributes shall have a status of read only (see 4.23).

Table 254 — Device type attributes

Attribute Identifier	Name	Attribute Length (in bytes)	Format	Subclause
0000h	REMAINING CAPACITY IN PARTITION	8	BINARY	6.5.2.2.1
0001h	MAXIMUM CAPACITY IN PARTITION	8	BINARY	6.5.2.2.1
0002h	TAPEALERT FLAGS	8	Binary	
0003h	LOAD COUNT	8	BINARY	6.5.2.2.3
0004h	MAM SPACE REMAINING	8	BINARY	6.5.2.2.4
0005h	ASSIGNING ORGANIZATION	8	ASCII	
0006h	FORMATTED DENSITY CODE	1	Binary	
0007h	INITIALIZATION COUNT	2	BINARY	6.5.2.2.5
0008h	Not Supported (VOLUME IDENTIFIER)	32	ASCII	
0009h	VOLUME CHANGE REFERENCE	4	BINARY	6.5.2.2.6
000Ah to 0209h	Reserved			
020Ah	DEVICE VENDOR/SERIAL NUMBER AT LAST LOAD	40	ASCII	6.5.2.2.7
020Bh	DEVICE VENDOR/SERIAL NUMBER AT LOAD-1	40	ASCII	6.5.2.2.7
020Ch	DEVICE VENDOR/SERIAL NUMBER AT LOAD-2	40	ASCII	6.5.2.2.9
020Dh	DEVICE VENDOR/SERIAL NUMBER AT LOAD-3	40	ASCII	6.5.2.2.10
020Eh to 021Fh	Reserved			
0220h	TOTAL MBYTES WRITTEN IN MEDIUM LIFE	8	BINARY	6.5.2.2.12
0221h	TOTAL MBYTES READ IN MEDIUM LIFE	8	BINARY	6.5.2.2.12
0222h	TOTAL MBYTES WRITTEN IN CURRENT/LAST LOAD	8	BINARY	6.5.2.2.13
0223h	TOTAL MBYTES READ IN CURRENT/LAST LOAD	8	BINARY	6.5.2.2.13
0224h	LOGICAL POSITION OF FIRST ENCRYPTED BLOCK	8	BINARY	6.5.2.2.15
0225h	LOGICAL POSITION OF FIRST UNENCRYPTED BLOCK AFTER THE FIRST ENCRYPTED BLOCK	8	BINARY	6.5.2.2.16
0226h to 033Fh	Reserved			
0340h	Not Supported (MEDIUM USAGE HISTORY)	90	BINARY	
0341h	Not Supported (PARTITION USAGE HISTORY)	60	BINARY	
0342h to 03FFh	Reserved			

6.5.2.2.1 REMAINING CAPACITY IN PARTITION: Native capacity (i.e., assuming no data compression for the specified medium partition) remaining in partition. This value is expressed in MiB.

6.5.2.2.2 MAXIMUM CAPACITY IN PARTITION: Native capacity (i.e., assuming no data compression for the specified medium partition) of the partition. This value is expressed in MiB.

6.5.2.2.3 LOAD COUNT: Indicates how many times this medium has been fully loaded. This attribute should not be reset to zero by any action of the device server. The load counter is a saturating counter.

6.5.2.2.4 MAM SPACE REMAINING: Indicates the space currently available in the medium auxiliary memory. The total medium auxiliary memory capacity is reported in the MAM CAPACITY attribute (see 6.5.2.3.7).

NOTE 77 - It may not always be possible to utilize all of the available space in a given medium auxiliary memory implementation.

6.5.2.2.5 INITIALIZATION COUNT: Indicates the number of times that a device server has logically formatted the medium. This value is cumulative over the life of the medium and is not reset to zero. The initialization counter is a saturating counter.

6.5.2.2.6 VOLUME CHANGE REFERENCE: The VOLUME CHANGE REFERENCE attribute indicates changes in the state of the medium related to logical objects or format specific symbols of the currently mounted volume. There is one value for the volume change reference and the VOLUME CHANGE REFERENCE attribute for each partition shall use the same value. The VOLUME CHANGE REFERENCE attribute value shall:

a) be written to non-volatile medium auxiliary memory before the change on medium is valid for reading; and
d) change in a non-repeating fashion (i.e., never repeat for the life of the volume) and defined to be in a consistent manner per volume format.

The VOLUME CHANGE REFERENCE attribute value shall change when:

a) the first logical object for each mount is written on the medium in any partition;
e) the first logical object is written after GOOD status has been returned for a READ ATTRIBUTE command with the service action field set to ATTRIBUTE VALUES (i.e., 00h) and the first attribute identifier field set to VOLUME CHANGE REFERENCE (i.e, 0009h);
f) any logical object on the medium (i.e., in any partition) is overwritten; or
g) the medium is formatted.

The VOLUME CHANGE REFERENCE attribute may change at other times when the contents on the medium change.

The VOLUME CHANGE REFERENCE attribute should not change if the logical objects on the medium do not change.

A value of zero in the VOLUME CHANGE REFERENCE attribute indicates that the medium has not had any logical objects written to it (i.e., the volume is blank and has never been written to) or the value is unknown.

A value of all ones (e.g., 0xFFFFh) in the VOLUME CHANGE REFERENCE attribute indicates that all values have been used. This value indicates the VOLUME CHANGE REFERENCE value is no longer able to indicate changes to the volume. The device server does not allow further modifications of the medium.

When adding or modifying logical objects the VOLUME CHANGE REFERENCE attribute should only be read after all writing to the volume has completed and been synchronized.

6.5.2.2.7 DEVICE VENDOR/SERIAL NUMBER AT LAST LOAD: The DEVICE VENDOR/SERIAL NUMBER AT LOAST LOAD attributes (this one and the next three) give a history of the last drive's in which the medium has been loaded. The format of the attribute is shown in table 255.

Table 255 — DEVICE VENDOR/SERIAL NUMBER attribute format

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)	T10 VENDOR IDENTIFICATION						
7								(LSB)
8	(MSB)	PRODUCT SERIAL NUMBER						
39								(LSB)

The T10 VENDOR IDENTIFICATION field shall be the same value returned in the Standard INQUIRY data.

The PRODUCT SERIAL NUMBER field contains ASCII data that is a vendor specific serial number. If the product serial number is not available, the PRODUCT SERIAL NUMBER field shall contain ASCII spaces (20h).

6.5.2.2.8 DEVICE VENDOR/SERIAL NUMBER AT LOAD –1: (see 6.5.2.2.7)

6.5.2.2.9 DEVICE VENDOR/SERIAL NUMBER AT LOAD –2: (see 6.5.2.2.7)

6.5.2.2.10 DEVICE VENDOR/SERIAL NUMBER AT LOAD –3: (see 6.5.2.2.7)

6.5.2.2.11 TOTAL MBYTES WRITTEN IN MEDIUM LIFE: Indicate the total number of data bytes that are transferred to the medium, after any data compression has been applied, over the entire medium life. These values are cumulative and shall not be reset to zero. These values are expressed in increments of 1 048 576 bytes (e.g., a value of one means 1 048 576 bytes and a value of two means 2 097 152 bytes).

6.5.2.2.12 TOTAL MBYTES READ IN MEDIUM LIFE: Indicate the total number of data bytes that are transferred from the medium, after any data compression has been applied, over the entire medium life. These values are cumulative and shall not be reset to zero. These values are expressed in MiB.

6.5.2.2.13 TOTAL MBYTES WRITTEN IN CURRENT/LAST LOAD: Indicate the total number of data bytes that are transferred to the medium, after any data compression has been applied, during the current load if the medium is currently loaded, or the last load if the medium is currently unloaded. The drive sets these attributes to zero when the medium is loaded. These values are expressed in MiB.

6.5.2.2.14 TOTAL MBYTES READ IN CURRENT/LAST LOAD: Indicate the total number of data bytes that are transferred from the medium, after any data compression has been applied, during the current load if the medium is currently loaded, or the last load if the medium is currently unloaded. The drive sets these attributes to zero when the medium is loaded. These values are expressed in MiB.

6.5.2.2.15 LOGICAL POSITION OF FIRST ENCRYPTED BLOCK: Indicates the address of the first logical block on the medium that contains encrypted data.

6.5.2.2.16 LOGICAL POSITION OF FIRST UNENCRYPTED BLOCK AFTER THE FIRST ENCRYPTED BLOCK: Indicates the address of the first logical block in the partition that contains unencrypted data and follows the first logical block in the partition that contains encrypted data. If this attribute is supported, then the LOGICAL POSITION OF FIRST ENCRYPTED BLOCK (see 7.3.2.2.9) attribute shall be supported. The attribute value shall be set to FFFF FFFF FFFF FFFFh if the attribute value for the LOGICAL POSITION OF FIRST ENCRYPTED BLOCK is set to:

a) FFFF FFFF FFFF FFFFh; or
h) any value other than FFFF FFFF FFFF FFFFh or FFFF FFFF FFFF FFFEh and no logical block in the partition after the first encrypted logical block contains unencrypted data.

The attribute value shall be set to FFFF FFFF FFFF FFFEh if the attribute value for the LOGICAL POSITION OF FIRST ENCRYPTED BLOCK is set to:

a) FFFF FFFF FFFF FFFEh; or
i) any value other than FFFF FFFF FFFF FFFFh or FFFF FFFF FFFF FFFEh and it is unknown whether any logical block in the partition after the first encrypted logical block contains unencrypted data.

6.5.2.3 Medium type attributes

Medium type attributes (see table 256) are stored in the medium auxiliary memory by the manufacturer. The device server shall not alter medium type attributes. All supported medium type attributes shall have a status of read only (see 4.23).

Table 256 — Medium type attributes

Attribute Identifier	Name	Attribute Length (in bytes)	Format	Subclause
0400h	MEDIUM MANUFACTURER	8	ASCII	6.5.2.3.1
0401h	MEDIUM SERIAL NUMBER	32	ASCII	6.5.2.3.2
0402h	MEDIUM LENGTH	4	Binary	6.5.2.3.3
0403h	MEDIUM WIDTH	4	Binary	6.5.2.3.4
0404h	ASSIGNING ORGANIZATION	8	ASCII	6.5.2.3.5

Table 256 — Medium type attributes

Attribute Identifier	Name	Attribute Length (in bytes)	Format	Subclause
0405h	MEDIUM DENSITY CODE	1	Binary	
0406h	MEDIUM MANUFACTURE DATE	8	ASCII	6.5.2.3.6
0407h	MAM CAPACITY	8	BINARY	6.5.2.3.7
0408h	MEDIUM TYPE	1	BINARY	6.5.2.3.8
0409h	MEDIUM TYPE INFORMATION	2	BINARY	6.5.2.3.8
040Ah	Not Supported (NUMERIC MEDIUM SERIAL NUMBER)			
040Bh to 07FFh	Reserved			

6.5.2.3.1 MEDIUM MANUFACTURER: Contains eight bytes of left-aligned ASCII data identifying the vendor of the media.

6.5.2.3.2 MEDIUM SERIAL NUMBER: Contains the manufacturer's serial number for the medium.

6.5.2.3.3 MEDIUM LENGTH: Specifies the length of the medium in meters. A value of 0h specifies that the length of the medium is undefined.

6.5.2.3.4 MEDIUM WIDTH: Specifies the width of the medium supported by this density. This attribute has units of tenths of millimeters. The value in this attribute shall be rounded up if the fractional value of the actual value is greater than or equal to 0,5. The MEDIUM WIDTH attribute may vary for a given density depending on the mounted volume. A value of 0h specifies the width of the medium is undefined.

6.5.2.3.5 ASSIGNING ORGANIZATION: Identifies the organization responsible for the specifications defining the values in the MEDIUM DENSITY CODE attribute. The ASSIGNING ORGANIZATION attribute contains "LTO-CVE".

6.5.2.3.6 MEDIUM MANUFACTURE DATE: Contains the date of manufacture of the medium. The format is YYYYMMDD (i.e., four numeric ASCII characters for the year followed by two numeric ASCII characters for the month followed by two numeric ASCII characters for the day with no intervening spaces).

6.5.2.3.7 MAM CAPACITY: Is the total capacity of the medium auxiliary memory, in bytes, at manufacture time. It does not indicate the available space of an unused medium auxiliary memory because some of the medium auxiliary memory space may be reserved for device-specific use making it inaccessible to the application client.

6.5.2.3.8 MEDIUM TYPE and MEDIUM TYPE INFORMATION: Give information about non-data media and other types of media. The MEDIUM TYPE INFORMATION attribute is interpreted according to the type of medium indicated by the MEDIUM TYPE (see table 257).

Table 257 — MEDIUM TYPE and MEDIUM TYPE INFORMATION attributes

MEDIUM TYPE	Description	MEDIUM TYPE INFORMATION
00h	Data medium	Reserved
01h	Cleaning medium	Maximum number of cleaning cycles permitted
02h to 7Fh	Reserved	Reserved
80h	Write-once medium	Reserved
81h to FFh	Reserved	Reserved

6.5.2.4 Host type attributes

Application clients may use the WRITE ATTRIBUTE and READ ATTRIBUTE commands to maintain the attributes shown in table 258. All existent host type attributes shall have a status of read/write (see 4.23).

Table 258 — Host type attributes

Attribute Identifier	Name	Attribute Length (in bytes)	Format	Subclause
0800h	APPLICATION VENDOR	8	ASCII	6.5.2.4.1
0801h	APPLICATION NAME	32	ASCII	6.5.2.4.2
0802h	APPLICATION VERSION	8	ASCII	6.5.2.4.3
0803h	USER MEDIUM TEXT LABEL	160	TEXT	6.5.2.4.4
0804h	DATE AND TIME LAST WRITTEN	12	ASCII	6.5.2.4.5
0805h	TEXT LOCALIZATION IDENTIFIER	1	BINARY	6.5.2.4.6
0806h	BARCODE	32	ASCII	6.5.2.4.7
0807h	OWNING HOST TEXTUAL NAME	80	TEXT	6.5.2.4.8
0808h	MEDIA POOL	160	TEXT	6.5.2.4.9
0809h	Not Supported (PARTITION USER TEXT LABEL)			
080Ah	Not Supported (LOAD/UNLOAD AT PARTITION)			
080Bh	APPLICATION FORMAT VERSION	16	ASCII	6.5.2.4.10
080Ch	VOLUME COHERENCY INFORMATION		BINARY	6.5.2.4.11
others between 0800h and 0BFFh	Reserved			

6.5.2.4.1 APPLICATION VENDOR: Contains eight bytes of left-aligned ASCII data identifying the manufacturer of the application client (e.g., class driver or backup program) that last sent a WRITE ATTRIBUTE command to the device server while this medium auxiliary memory was accessible. The application vendor shall be a T10 vendor identification assigned by INCITS. A list of assigned T10 vendor identifications is on the T10 web site (http://www.T10.org).

6.5.2.4.2 APPLICATION NAME: Contains the name of the application client.

6.5.2.4.3 APPLICATION VERSION: Contains the version of the application client.

6.5.2.4.4 USER MEDIUM TEXT LABEL: Is the user level identifier for the medium.

6.5.2.4.5 DATE & TIME LAST WRITTEN: Contains when the application client last wrote to the medium auxiliary memory. The format is YYYYMMDDHHMM (i.e., four numeric ASCII characters for the year followed by two numeric ASCII characters for the month followed by two numeric ASCII characters for the day followed by two numeric ASCII characters between 00 and 24 for the hour followed by two numeric ASCII characters for the minute with no intervening spaces).

6.5.2.4.6 TEXT LOCALIZATION IDENTIFIER: Defines the character set (see table 259) used for attributes with a TEXT format (see 6.5.1).

Table 259 — TEXT LOCALIZATION IDENTIFIER attribute values

Value	Meaning
00h	No code specified (ASCII)
01h	ISO/IEC 8859-1 (Europe, Latin America)
02h	ISO/IEC 8859-2 (Eastern Europe)
03h	ISO/IEC 8859-3 (SE Europe/miscellaneous)
04h	ISO/IEC 8859-4 (Scandinavia/Baltic)

Table 259 — TEXT LOCALIZATION IDENTIFIER attribute values

Value	Meaning
05h	ISO/IEC 8859-5 (Cyrillic)
06h	ISO/IEC 8859-6 (Arabic)
07h	ISO/IEC 8859-7 (Greek)
08h	ISO/IEC 8859-8 (Hebrew)
09h	ISO/IEC 8859-9 (Latin 5)
0Ah	ISO/IEC 8859-10 (Latin 6)
0Bh to 7Fh	Reserved
80h	ISO/IEC 10646-1 (UCS-2BE)
81h	ISO/IEC 10646-1 (UTF-8)
82h to FFh	Reserved

6.5.2.4.7 BARCODE: Is contents of a barcode associated with the medium in the medium auxiliary memory.

6.5.2.4.8 OWNING HOST TEXTUAL NAME: Indicates the host from which that USER MEDIUM TEXT LABEL (see 6.5.2.4.4) originates.

6.5.2.4.9 MEDIA POOL: Indicates the media pool to which this medium belongs.

6.5.2.4.10 APPLICATION FORMAT VERSION: Indicates the version of the format being used by the application that set this attribute.

6.5.2.4.11 VOLUME COHERENCY INFORMATION: Contains information used to maintain coherency of information on a volume (see 4.24). The VOLUME COHERENCY INFORMATION attribute ATTRIBUTE VALUE field is defined in table 260

Table 260 — VOLUME COHERENCY INFORMATION attribute format

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	VOLUME CHANGE REFERENCE VALUE LENGTH (n)							
1 n	VOLUME CHANGE REFERENCE VALUE							
n+1 n+8	VOLUME COHERENCY COUNT							
n+9 n+16	VOLUME COHERENCY SET IDENTIFIER							
n+17 n+18	APPLICATION CLIENT SPECIFIC INFORMATION LENGTH (y-(n+18))							
n+19 y	APPLICATION CLIENT SPECIFIC INFORMATION							

The contents of the VOLUME CHANGE REFERENCE VALUE field, the VOLUME COHERENCY SET IDENTIFIER field, the VOLUME COHERENCY COUNT field, and the APPLICATION CLIENT SPECIFIC INFORMATION field are described in 4.24 on page 82.

The VOLUME CHANGE REFERENCE VALUE LENGTH field contains the length of the VOLUME CHANGE REFERENCE VALUE field.

6.5.2.5 Vendor specific host type attributes (E07+)

The amount of space available for vendor specific host type attributes varies by cartridge type. The amount of user data available is reduced by the overhead associated with each ATTRIBUTE ID used as indicated in table 261.

Table 261 — MAM user data space available

Cartridge Type	Bytes user data space available where n is number of unique ATTRIBUTE IDs used
JB/JX	1024 - 4(n+1)
JK	
JC/JY	
JL	3072 - 4(n+1)
JD/JZ	
JM	
JE/JV	

6.6 Mode Parameters

Mode parameters are used with the MODE SELECT (6/10) - 15h/55h (see 5.2.14) commands and the MODE SENSE (6/10) - 1Ah/5Ah (see 5.2.15) commands. Mode Page Behaviors (see 4.5 on page 27) describes this device’s non-standard behaviors related to mode parameters.

6.6.1 Mode Parameter List for Mode Select (6/10)

For Mode Select (6) the mode pages are preceded by a 4-byte mode parameter header below, and an optional 8-byte block descriptor (see 6.6.2.2). Table 262 shows the format of the mode parameter list for Mode Select (6).

Table 262 — Mode Parameter List for Mode Select (6)

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Mode Parameter Header							
3								
4	Block Descriptor (n = 0 or n = 8; if n=0 Block Descriptor does not exist)							
4+n-1								
4+n	Mode Pages							
n								

For Mode Select (10) the mode pages are preceded by a 8-byte mode parameter header (see "Mode Parameter Header for Mode Select (6/10)" on page 374) and an optional 8-byte block descriptor (see "Block Descriptor for Mode Select (6/10)" on page 375). Mode page descriptions begin at "MP 01h: Read-Write Error Recovery" on page 383.

Table 263 shows the format of the mode parameter list for Mode Select (10).

Table 263 — Mode Parameter List for Mode Select (10)

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Mode Parameter Header							
7								
8	Block Descriptor (n = 0 or n = 8; if n=0 Block Descriptor does not exist)							
8+n-1								
8+n	Mode Pages							
n								

6.6.1.1 Mode Parameter Header for Mode Select (6/10)

There is one copy of the mode parameter header for each initiator. Mode parameter header and block descriptor policy (see 4.5.1.1 on page 27) describes this devices non-standard behavior related to fields in the header.

Note that mounting a volume that modifies the value of fields in the mode parameter header does not establish a unit attention condition.

Table 264 shows the format of the mode parameter header for Mode Select (6).

Table 264 — Mode Parameter Header for Mode Select (6)

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Reserved (Mode Data Length)							
1	Medium Type							
2	WP	Buffer Mode			Speed			
3	Block Descriptor Length							

Table 265 shows the format of the mode parameter header for Mode Select (10).

Table 265 — Mode Parameter Header for Mode Select (10)

Byte	**Bit**							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	MSB	Reserved (Mode Data Length)						
1								LSB
2	Medium Type							
3	WP	Buffer Mode			Speed			
4	Reserved							
5								
6	MSB	Block Descriptor Length						
7								LSB

Mode parameter header field descriptions follow:

Byte **Description**

0-1 Reserved (Mode Data Length): must be zero.

2 Medium Type: (changeable-ignored) See Mode Parameter Header for Mode Sense (6/10) (see 6.6.2.1 on page 378)

3 Device-Specific Parameter - Sequential Access Devices

Bit	**Description**
7	WP: (changeable-ignored)
6-4	Buffered Mode: 001b <Shared> (changeable)

Value	**Description**
000b	Good status is reported when data on medium
001b	Good status is reported when data is in buffer
010b	Not Supported
011b - 111b	Reserved

3-0	Speed: 0b (use default speed) <Shared> (changeable)

WARNING WARNING

Setting the SPEED field to a value other than 0h (i.e., selecting a specific speed) is not recommended. The drive is designed to dynamically select the optimal speed to achieve maximum systemic performance. This is based on complex criteria including interface bandwidth, host throughput, data compressibility, etc.

WARNING WARNING

4-5 Reserved

6-7 Block Descriptor Length: <Per I_T nexus> (changeable)

Value	**Description**
0000h	No block descriptor follows
0008h	A single block descriptor follows

6.6.1.2 Block Descriptor for Mode Select (6/10)

The presence of the block descriptor in the Mode Select command depends on the value of the Block Descriptor Length in the mode parameter header. There is one copy of the block descriptor for each initiator. Table 266 shows the format of the block descriptor. The format of the block descriptor is the same for Mode Select (6) and Mode Select (10). Mode parameter header and block descriptor policy (see 4.5.1.1 on page 27) describes this devices non-standard behavior related to fields in the block descriptor.

Note that mounting a volume that modifies the value of fields in the block descriptor does not establish a unit attention condition.

Table 266 — Block Descriptor for Mode Select

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Density Code							
1	MSB	Number of Blocks						
3								LSB
4	Reserved							
5	MSB	Block Length						
7								LSB

The block descriptor field definitions follow:

Byte Description

0 Density Code: (changeable)(ignored on J1A) (see “REPORT DENSITY SUPPORT - 44h” on page 179)

NOTE 78 - Changing density and logical format with this mechanism is now supported. The new density will be applied on the next write-type operation when positioned at BOP (Write, Write Filemarks (>0), Erase, Format Medium, etc.) and will not be reported in the Mode Sense Block Descriptor Density Code field before the format is performed. If this value is changed from the value reported in Mode Sense, that change will be remembered until the next write at BOP or until another change is made. Any pending, but unrealized density change will be cleared on unload. Such a pending density code change may also be read or written using Mode Sense/Select (see “MP 25h: Read/Write Control” on page 427).

NOTE 79 - If a specific density (e.g., 51h) is specified, this value will only have an effect if it is different than the value returned in Mode Sense. If the same density is returned in Mode Sense is desired, then the value FFh should be used. If the same value is specified, then no change is made to default or previously specified density.

Value	**Description**
00h	Medium present unidentified for one of the following reasons: • No medium present • Unknown/Unsupported medium present
51h	Medium present is 3592 Enterprise Tape Cartridge 3592A1 density (only JA/JW; JJ/JR medium)
52h	Medium present is 3592 Enterprise Tape Cartridge 3592A2 density (only JA/JW; JJ/JR; JB/JX medium)
53h	Medium present is 3592 Enterprise Tape Cartridge 3592A3 density (only JA/JW; JJ/JR; JB/JX medium)
54h	Medium present is 3592 Enterprise Tape Cartridge 3592A4 density (only JB/JX; JC/JY/JK) medium)
55h	Medium present is 3592 Enterprise Tape Cartridge 3592A5 density (only JC/JY/JK; JD/JZ/JL) medium)
56h	Medium present is 3592 Enterprise Tape Cartridge 3592B5 density (only JD/JZ/JL) medium)
57h	Medium present is 3592 Enterprise Tape Cartridge 3592A6 density (only JE/JV/JM medium)
71h	Medium present is 3592 Enterprise Tape Cartridge Encrypted 3592A1 density (only JA/JW; JJ/JR medium)
72h	Medium present is 3592 Enterprise Tape Cartridge Encrypted 3592A2 density (only JA/JW; JJ/JR; JB/JX medium)
73h	Medium present is 3592 Enterprise Tape Cartridge Encrypted 3592A3 density (only JA/JW; JJ/JR; JB/JX medium)
74h	Medium present is 3592 Enterprise Tape Cartridge Encrypted 3592A4 density (only JB/JX; JC/JY/JK) medium)
75h	Medium present is 3592 Enterprise Tape Cartridge Encrypted 3592A5 density (only JC/JY/JK; JD/JZ/JL) medium)
76h	Medium present is 3592 Enterprise Tape Cartridge Encrypted 3592B5 density (only JD/JZ/JL) medium)
77h	Medium present is 3592 Enterprise Tape Cartridge Encrypted 3592A6 density (only JE/JV/JM medium)
7Fh	Do not change density (set only - NOOP)
FFh	Use current medium density (set only)

NOTE 80 - On devices which support encryption, secondary encryption density codes may be selected but are interchangeable and identical to primary density codes. Use of any density codes has no effect on the encrypted state of medium and only reports encryption state information and cannot be used to control encryption.

1-3 Number of Blocks: 000000h (non-changeable)

4 Reserved

5-7 Block Length: 000000h <Per I_T nexus> (changeable)

Any value of block length can be specified between the minimum and the maximum block lengths, inclusive, specified in the Read Block Limits command.

A Block Length value of 000000h indicates that the logical block size to be written to or read from the medium must be explicitly specified by the Transfer Length field in the CDB and the fixed bit must be 0b (see “READ - 08h” on page 139 and “WRITE - 0Ah” on page 216). Additionally the read-type overlength ILI reporting will be suppressed (see 4.10.1).

6.6.2 Mode Parameter List for Mode Sense (6/10)

For Mode Sense (6) the mode pages are preceded by a 4-byte mode parameter header below and an optional block descriptor. If the DBD field is 0b, an 8-byte block descriptor follows the mode parameter header (see “Block Descriptor for Mode Sense (6/10)” on page 380). If the DBD field is 1b, the block descriptor is not present and the first mode page follows the mode parameter header. Table 267 shows the format of the mode parameter list.

Table 267 — Mode Parameter List for Mode Sense (6)

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Mode Parameter Header							
3								
4	Block Descriptor (if DBD = 0b, then n = 8 an m=12 else DBD = 1b, n = 0,and m=4)							
4+n-1								
m	Mode Pages							
p								

For Mode Sense (10) the mode pages are preceded by an 8-byte mode parameter header (see “Mode Parameter Header for Mode Sense (6/10)” on page 378) and an optional block descriptor. If the DBD field is 0b, an 8-byte block descriptor follows the mode parameter header (see “Block Descriptor for Mode Select (6/10)” on page 375). If the DBD field is 1b, the block descriptor is not present and the first mode page follows the mode parameter header. Mode page descriptions begin at “MP 01h: Read-Write Error Recovery” on page 383. Table 268 shows the format of the mode parameter list.

Table 268 — Mode Parameter List for Mode Sense (10)

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Mode Parameter Header							
7								
8	Block Descriptor (if DBD = 0b, then n = 8 an m=16 else DBD = 1b, n = 0,and m=8)							
8+n-1								
m	Mode Pages							
p								

6.6.2.1 Mode Parameter Header for Mode Sense (6/10)

There is one copy of the mode parameter header for each initiator. Table 269 shows the format of the mode parameter header for Mode Sense (6).

Note that mounting a volume that modifies the value of fields in the mode parameter header does not establish a unit attention condition.

Table 269 — Mode Parameter Header for Mode Sense (6)

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Mode Data Length							
1	Medium Type							
2	WP	Buffer Mode			Read/Write Speed			
3	Block Descriptor Length							

Table 270 shows the format of the mode parameter header for Mode Sense (10).

Table 270 — Mode Parameter Header for Mode Sense (10)

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	MSB	Mode Data Length						
1								LSB
2	Medium Type							
3	WP	Buffer Mode			Read/Write Speed			
4	MSB	Reserved						
5								LSB
6	MSB	Block Descriptor Length						
7								LSB

Mode parameter header field descriptions follow:

Byte **Description**

0-1 Mode Data Length

The length in bytes of the following data that is available to be transferred. The mode data length does not include itself; that is, the length value is total length of the data available minus the size of this field (1 or 2, depending on Mode Sense (6) or (10), respectively).

2 Medium Type:

Value	**Description**
00h	No medium present or the drive does not support loaded cartridge type
91h	Medium present is 3592 Enterprise Tape — Standard Cartridge (JA)
92h	Medium present is 3592 Enterprise Tape — Extended Cartridge (JB)
93h	Medium present is 3592 Enterprise Tape — Advanced Type C Cartridge (JC)
94h	Medium present is 3592 Enterprise Tape — Advanced Type D Cartridge (JD)
95h	Medium present is 3592 Enterprise Tape — Advanced Type E Cartridge (JE)
A1h	Medium present is 3592 Enterprise Tape — Standard WORM Cartridge (JW)
A2h	Medium present is 3592 Enterprise Tape — Extended WORM Cartridge (JX)
A3h	Medium present is 3592 Enterprise Tape — Advanced Type C WORM Cartridge (JY)
A4h	Medium present is 3592 Enterprise Tape — Advanced Type D WORM Cartridge (JZ)
A5h	Medium present is 3592 Enterprise Tape — Advanced Type E WORM Cartridge (JV)
B1h	Medium present is 3592 Enterprise Tape — Economy Cartridge (JJ)
B2h	Medium present is 3592 Enterprise Tape — Advanced Type C Economy Cartridge (JK)
B3h	Medium present is 3592 Enterprise Tape — Advanced Type D Economy Cartridge (JL)
B4h	Medium present is 3592 Enterprise Tape — Advanced Type E Economy Cartridge (JM)
C1h	Medium present is 3592 Enterprise Tape — Economy WORM Cartridge (JR)

3 Device-Specific Parameter - Sequential Access Devices. See Mode Parameter Header for Mode Select (6/10) (see 6.6.1.1 on page 374).

4-5 Reserved

6-7 Block Descriptor Length:

If DBD = 0, the Block Descriptor Length field is set to 0008h and a block descriptor follows. If DBD = 1, the Block Descriptor Length field is set to 0000h and no block descriptor follows.

6.6.2.2 Block Descriptor for Mode Sense (6/10)

The presence of the block descriptor in the Mode Sense command depends on the value of the DBD bit in the CDB. There is one copy of the block descriptor for each initiator. Table 271 shows the format of the block descriptor.

Note that mounting a volume that modifies the value of fields in the block descriptor does not establish a unit attention condition.

Table 271 — Block Descriptor for Mode Sense (10) or Mode Sense (6)

Byte	**Bit**							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Density Code							
1	MSB	Number of Blocks						
3								LSB
4	Reserved							
5	MSB	Block Length						
7								LSB

The block descriptor is defined in Block Descriptor for Mode Select (6/10) (see 6.6.1.2 on page 375).

6.6.3 Mode Page Format

Table 272 shows the format of mode pages that do not use subpages.

Table 272 — Mode Page Format

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	PS	SPF(0B)	Page Code					
1	Page Length (n-1)							
2	Mode Parameters							
n								

Table 273 shows the format of mode pages that use subpages.

Table 273 — Mode Page Subpage Format

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	PS	SPF(1B)	Page Code					
1	Subpage Code							
2	(MSB)	Page Length (n-3)						
3								(LSB)
4	Mode Parameters							
n								

The individual mode page descriptions that follow include the field descriptions. Each field is non-changeable unless specifically identified otherwise.

6.6.4 Supported Mode Pages

The following standards-based mode pages are supported

MP 01h: Read-Write Error Recovery (see 6.6.5 on page 383)
MP 02h: Disconnect-Reconnect (see 6.6.6 on page 385)
MP 0Ah: Control Mode (see 6.6.7 on page 387)
MP 0Ah[F0h]: Control Data Protection (see 6.6.9 on page 390)
MP 0Fh: Data Compression (see 6.6.10 on page 392)
MP 10h: Device Configuration (see 6.6.11 on page 394)
MP 10h[01h]: Device Configuration Extension (see 6.6.12 on page 397)
MP 11h: Medium Partition Page (see 6.6.13 on page 399)
MP 18h: Fibre Channel Logical Unit Control (see 6.6.14 on page 405)
MP 19h: Fibre Channel Port Control (see 6.6.15 on page 406)
MP 1Ch: Informational Exceptions Control (see 6.6.16 on page 408)

The following vendor-specific mode pages are also supported

MP 21h: TOD Control (see 6.6.17 on page 411)
MP 22h: Language (see 6.6.18 on page 413)
MP 23h: Medium Sense (see 6.6.19 on page 414)
MP 24h: Initiator-Specific Extensions (see 6.6.20 on page 424)
MP 25h: Read/Write Control (see 6.6.21 on page 427)
MP 30h: Device Attribute Settings (see 6.6.22 on page 431)
MP 30h[01h]: Drive MAC address - Device attribute settings (see 6.6.22.3.2 on page 434)
MP 30h[02h]: Drive IP address and subnet mask - Device attribute settings (see 6.6.22.3.3 on page 436)
MP 30h[20h]: Encryption mode - Device Attribute Settings (see 6.6.22.4.1 on page 437)
MP 30h[40h]: SkipSync - Device attribute settings (see 6.6.22.5.1 on page 439)
MP 30h[42h]: End of partition behavior control - Device attribute settings (see 6.6.22.5.2 on page 442)
MP 30h[43h]: Feature switches - Device attribute settings (see 6.6.22.5.3 on page 442)
MP 37h: String Search (not J1A) (see 6.6.23 on page 444)
MP 3Eh: Engineering Support (see 6.6.24 on page 455)
“Mode Page 3Fh: All Pages”

NOTE 81 - Mode page 37h is not included in page 3Fh.

NOTE 82 - Mode page 3Eh is for engineering use only and is not included in page 3Fh

6.6.5 MP 01h: Read-Write Error Recovery

See MODE SELECT (6/10) - 15h/55h (see 5.2.14) for how to set these parameters and MODE SENSE (6/10) - 1Ah/5Ah (see 5.2.15) for how to read these parameters. Mode Page Behaviors (see 4.5 on page 27) describes this device's non-standard behaviors related to mode parameters.

The mode page policy for this page is described in Policy — non-standard (see 4.5.1 on page 27).

The Read-Write Error Recovery mode page (see table 274) specifies the error recovery and reporting parameters that the device server shall use when transferring data between the device and the medium. These parameters do not affect protocol-level recovery procedures or positioning error recovery procedures.

NOTE 83 - The parameters in the Read-Write Error Recovery mode page also apply to verify operations.

Table 274 — Read-Write Error Recovery mode page

<table>
<tr><th rowspan="2">Byte</th><th colspan="8">Bit</th></tr>
<tr><th>7 msb</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0 lsb</th></tr>
<tr><td>0</td><td>PS</td><td>SPF(0)</td><td colspan="6">PAGE CODE (01h)</td></tr>
<tr><td>1</td><td colspan="8">PAGE LENGTH (0Ah)</td></tr>
<tr><td>2</td><td colspan="2">Reserved</td><td>TB</td><td>Rsvd</td><td>EER</td><td>PER</td><td>DTE</td><td>DCR</td></tr>
<tr><td>3</td><td colspan="8">READ RETRY COUNT</td></tr>
<tr><td>4</td><td colspan="8">Reserved</td></tr>
<tr><td>5</td><td colspan="8">Reserved</td></tr>
<tr><td>6</td><td colspan="8">Reserved</td></tr>
<tr><td>7</td><td colspan="8">Reserved</td></tr>
<tr><td>8</td><td colspan="8">WRITE RETRY COUNT</td></tr>
<tr><td>9</td><td colspan="8">Reserved</td></tr>
<tr><td>10</td><td colspan="8">Reserved</td></tr>
<tr><td>11</td><td colspan="8">Reserved</td></tr>
</table>

Byte	Description
0	

Bit	Description
7	PS (Parameter Saveable): 0b. Ignored in MODE SELECT. See Parameter Saveable behavior in 4.5.2.2 on page 29
6	Reserved
5-0	Page Code: 01h

1	Page Length: 0Ah

2

Bit	Description
7-6	Reserved
5	TB (Transfer Block): 1b (non-changeable)
4	Reserved
3	EER (Enable Early Recovery): 1b (non-changeable)
2	PER (Post Error): 0b (changeable)

Value	Description
0b	The device does not create CHECK CONDITION status for recovered errors except for non-deferred sense data of: — 1/0017 (Recovered Error, Drive Needs Cleaning) for a Load Unload command — 1/3700 (Recovered Error, Rounded Parameter) for a Mode Select command, and — 1/8383 (Recovered Error, Drive Has Been Cleaned) for a Load Unload command. For reporting of Housekeeping errors, see “MP 25h: Read/Write Control” on page 427.
1b	The device will report a CHECK CONDITION status for all recovered data and non-data errors with a sense key of 1 in non-deferred sense data as well as deferred sense data.

Bit	Description
1	DTE (Disable Transfer on Error): 0b (non-changeable)
0	DCR (Disable Correction): 0b (non-changeable)

3 Read Retry Limit: FFh (approximate maximum read recovery limit in seconds) (changeable)

Value	Description
05h	Limited error recovery; < 5 seconds.
FFh	Full Recovery Routines allowed (no time limit).
XXh	All other values may be rounded (to non-FFh).

4-7 Reserved

8 Write Retry Limit: FFh (approximate maximum write recovery limit in seconds) (changeable)

Value	Description
02h	Limited error recovery; < 2 seconds.
05h	Limited error recovery; < 5 seconds.
FFh	Full Recovery Routines allowed (no time limit).
XXh	All other values may be rounded (to non-FFh).

9-11 Reserved

6.6.6 MP 02h: Disconnect-Reconnect

See MODE SELECT (6/10) - 15h/55h (see 5.2.14) for how to set these parameters and MODE SENSE (6/10) - 1Ah/5Ah (see 5.2.15) for how to read these parameters.

Mode Page Behaviors (see 4.5 on page 27) describes this device’s non-standard behaviors related to mode parameters.

The mode page policy for this page is described in Policy — non-standard (see 4.1.1 on page 2).

Table 275 — Disconnect-Reconnect mode page

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	PS	SPF (0b)	PAGE CODE (02h)					
1	PAGE LENGTH (0Eh)							
2	BUFFER FULL RATIO							
3	BUFFER EMPTY RATIO							
4	(MSB)	BUS INACTIVITY LIMIT						
5								(LSB)
6	(MSB)	DISCONNECT TIME LIMIT						
7								(LSB)
8	(MSB)	CONNECT TIME LIMIT						
9								(LSB)
10	(MSB)	MAXIMUM BURST SIZE						
11								(LSB)
12	EMDP	FAIR ARBITRATION			DIMM	DTDC		
13	Reserved							
14	(MSB)	FIRST BURST SIZE						
15								(LSB)

Byte **Description**

0

Bit	Description
7	PS: 0b. Ignored in MODE SELECT. See Parameter Saveable behavior in 4.5.2.2 on page 29.
6	Reserved
5-0	Page Code: 02h

Byte	Description
1	Page Length: 0Eh
2	Buffer Full Ratio: 00h (non-changeable)
3	Buffer Empty Ratio: 00h (non-changeable)
4-5	Bus Inactivity Limit: 0000h (no limit) (non-changeable)
6-7	Disconnect Time Limit: 0000h (no limit) (non-changeable)
8-9	Connect Time Limit: 0000h (no limit) (non-changeable)
10-11	Maximum Burst Size: SCSI - 0000h (no limit) (changeable) This value is changeable and remembered, but not used.

12

Bit	**Description**
7	EMDP (Enable Modify Data Pointers): 0b (non-changeable)
6	FARd (Fair Arbitration Read): 0b (non-changeable)
5	FAWrt (Fair Arbitration Write): 0b (non-changeable)
4	FAStat (Fair Arbitration Status): 0b (non-changeable)
3	Dimm (Disconnect Immediate): 0b (non-changeable)
2-0	DTDC (Data Transfer Disconnect Word): 000b (non-changeable)

13 Reserved

14-15 First Burst Size: 0000h (non-changeable)

6.6.7 MP 0Ah: Control Mode

See MODE SELECT (6/10) - 15h/55h (see 5.2.14) for how to set these parameters and MODE SENSE (6/10) - 1Ah/5Ah (see 5.2.15) for how to read these parameters.

Mode Page Behaviors (see 4.5 on page 27) describes this device’s non-standard behaviors related to mode parameters.

The mode page policy for this page is described in Policy — non-standard (see 4.5.1 on page 27)

Table 276 — Control mode page

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	PS	SPF (0b)	PAGE CODE (0Ah)					
1	PAGE LENGTH (0Ah)							
2	TST			Reserved			GLTSD	RLEC
3	QUEUE ALGORITHM MODIFIER				Reserved		QERR	DQUE
4	Reserved	RAC	Reserved			RAERP	UAAERP	EAERP
5	Reserved					AUTOLOAD MODE		
6	(MSB)	READY AEN HOLDOFF PERIOD						
7								(LSB)
8	(MSB)	BUSY TIME-OUT PERIOD						
9								(LSB)
10	(MSB)	Reserved						
11								(LSB)

Byte Description

0

Bit	Description
7	PS: 0b. Ignored in MODE SELECT. See Parameter Saveable behavior in 4.5.2.2 on page 29.
6	SPF: 0b
5-0	Page Code: 0Ah

1 Page Length: 0Ah

2

Bit	Description
7-5	TST (Task Set Type): 000b (non-changeable)
4-2	Reserved
1	GLTSD (Global Logging Target Save Disable): 0b (non-changeable)
0	RLEC (Report Log Exception Condition): 0b (non-changeable)

3

Bit	Description
7-4	Queue Algorithm Modifier: 0000b (non-changeable)
3-2	Reserved
1	QErr (Queue Error): 0b (non-changeable)
0	DQue (Disable Queuing): 1b (non-changeable)

4

Bit	Description
7	Reserved
6	RAC (Report A Check): 0b (non-changeable)
5-3	Reserved
2	RAERP (Ready Asynchronous Event Reporting): 0b (non-changeable)
1	UAAERP (Unit Attention Asynchronous Event Reporting): 0b (non-changeable)
0	EAERP (Error Asynchronous Event Reporting): 0b (non-changeable)

5

Bit	Description
7-3	Reserved
2-0	AUTOLOAD MODE: 000b (changeable)

Some medium changers may silently override AUTOLOAD MODE.

Value	Description
000b	Volume is loaded for full access.
001b	Volume is loaded for MAM access only (i.e., load hold position).
010b	Volume is not allowed to be loaded. ***WARNING:*** This setting may cause problems for medium changers (e.g., fail move medium commands, interference with pickers, etc.)
others	Reserved.

6-7 Ready AEN Holdoff Period: 0000h (non-changeable)

If AEN is disabled (Byte 4 bit 2 = 0b), this field is not meaningful.

8-9 Busy time-out period: FFFFh (non-changeable)

10 Reserved

11 Reserved

6.6.8 MP 0Ah[01h]: Control Extension

See MODE SELECT (6/10) - 15h/55h (see 5.2.14 on page 129) for how to set these parameters and MODE SENSE (6/10) - 1Ah/5Ah (see 5.2.15 on page 131) for how to read these parameters.Mode Page Behaviors (see 4.5 on page 27) describes this device’s non-standard behaviors related to mode parameters.

The mode page policy for this page is described in Policy — non-standard (see 4.5.1 on page 27)

Table 277 — Control Extension mode page

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	PS	SPF (1b)	PAGE CODE (0Ah)					
1	SUBPAGE CODE (01h)							
2	(MSB)	PAGE LENGTH (1Ch)						
3								(LSB)
4	Reserved					TCMOS	SCSIP	IALUAE
5	Reserved				INITIAL COMMAND PRIORITY			
6 31	Reserved							

Byte Description

0

Bit	**Description**
7	PS (Parameter Saveable): 0b. Ignored in MODE SELECT. See Parameter Saveable behavior in 4.5.2.2 on page 29.
6	SPF: 1b
5-0	PAGE CODE: 0Ah

1 SUBPAGE CODE: 01h

2-3 PAGE LENGTH: 1Ch

4

Bit Description

7-3 Reserved

2 TCMOS (Timestamp Changeable by Methods Outside this Standard): 1b (non-changeable)

Value	**Description**
0b	Follows the behavior of being set to 1b.
1b	The timestamp may be initialized by methods other that the SET TIMESTAMP command (for example, The LDI Set Timestamp command).

1 SCSIP (SCSI precedence): 1b (changeable)

Value	**Description**
0b	Methods outside the SET TIMESTAMP command (for example the LDI Set Timestamp command) may change the timestamp and the SET TIMESTAMP command is illegal.
1b	The timestamp may be changed using a SET TIMESTAMP command if not already set by the library.

0 IALUAE (implicit asymmetric logical unit access enabled): 0b (non-changeable)

5

Bit	**Description**
7-4	Reserved
3-0	INITIAL COMMAND PRIORITY: 0h (non-changeable)

6-31 Reserved

6.6.9 MP 0Ah[F0h]: Control Data Protection

See MODE SELECT (6/10) - 15h/55h (see 5.2.14 on page 129) for how to set these parameters and MODE SENSE (6/10) - 1Ah/5Ah (see 5.2.15 on page 131) for how to read these parameters.

The Control Data Protection mode page provides controls that allow selective use of logical block protection. Logical block protection (see 4.7 on page 31) describes how this page is used to control logical block protection.

Mode Page Behaviors (see 4.5 on page 27) describes this device's non-standard behaviors related to mode parameters.

The mode page policy for this page is described in Policy — non-standard (see 4.5.1 on page 27)

Table 278 — Control Data Protection mode page format

Byte	Bit							
	7	6	5	4	3	2	1	0
0	PS	SPF(1b)	PAGE CODE (0Ah)					
1	SUBPAGE CODE (F0h)							
2	(MSB)	PAGE LENGTH (28)						
3								(LSB)
4	LOGICAL BLOCK PROTECTION METHOD							
5	Reserved		LOGICAL BLOCK PROTECTION INFORMATION LENGTH					
6	LBP_W	LBP_R	RBDP	Reserved				
7	Reserved							
8	Reserved							
31								

Byte Description

0

Bit	Description
7	PS: 0b. Ignored in MODE SELECT. See Parameter Saveable behavior in 4.5.2.2 on page 29.
6	SPF: 1b
5-0	PAGE CODE: 0Ah

1 SUBPAGE CODE: F0h

2-3 PAGE LENGTH: 1Ch

4 LOGICAL BLOCK PROTECTION METHOD: (CHANGEABLE)

Value	Description
00h	Do not use logical block protection
01h	Use the Reed-Solomon CRC as defined in ECMA-319 as the logical block protection information.
02h	Use the CRC32C CRC as the logical block protection information transferred between the drive and the host. The CRC32C CRC is transformed to or from the Reed-Solomon CRC by the drive between the block being transferred across the interface and the block saved on the tape in the tape format.
others	Reserved.

5

Bit	Description
7-6	Reserved
5-0	LOGICAL BLOCK PROTECTION INFORMATION LENGTH: 00h (CHANGEABLE)

6

Bit	Description
7	LBP_W (logical blocks protected during write): 0b (CHANGEABLE)

Shall be set to zero if the LOGICAL BLOCK PROTECTION METHOD field is set to zero.

Value	Description
0b	Protection information is not included with logical blocks transferred when writing.
1b	Protection information is included with logical blocks transferred during processing of the commands specified in Protecting logical blocks transferred during writes (see 4.7.4 on page 34).

Bit	Description
6	LBP_R (logical block protected during read): 0b (CHANGEABLE)

Shall be set to zero if the LOGICAL BLOCK PROTECTION METHOD field is set to zero.

Value	Description
0b	Protection information is not included with logical blocks transferred when reading.
1b	Protection information is included with logical blocks transferred during processing of the commands specified in Protecting logical blocks transferred during reads (see 4.7.5 on page 34).

Bit	Description
5	RBDP (recover buffered data protected): 0b (CHANGEABLE)

Shall be set to zero if the LOGICAL BLOCK PROTECTION METHOD field is set to zero.

Value	Description
0b	Protection information is not included with logical blocks transferred during the processing of a RECOVER BUFFERED DATA command (see 5.2.30).
1b	Protection information is included with logical blocks transferred during the processing of a RECOVER BUFFERED DATA command as specified in Protecting logical blocks transferred from the object buffer in response to a RECOVER BUFFERED DATA command (see 4.7.6 on page 35)

Bit	Description
4-0	Reserved

7-31 Reserved

6.6.10 MP 0Fh: Data Compression

See MODE SELECT (6/10) - 15h/55h (see 5.2.14) for how to set these parameters and MODE SENSE (6/10) - 1Ah/5Ah (see 5.2.15) for how to read these parameters.Mode Page Behaviors (see 4.5 on page 27) describes this device's non-standard behaviors related to mode parameters.

The mode page policy for this page is described in Policy — non-standard (see 4.5.1 on page 27)

Table 279 — Data Compression mode page

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	PS	SPF(0)	PAGE CODE (0Fh)					
1	PAGE LENGTH (0Eh)							
2	DCE	DCC	Reserved					
3	DDE	RED		Reserved				
4	(MSB)	COMPRESSION ALGORITHM						
7								(LSB)
8	(MSB)	DECOMPRESSION ALGORITHM						
11								(LSB)
12	Reserved							
15								

Byte **Description**

0

Bit	**Description**
7	PS: 0b. Ignored in MODE SELECT. See Parameter Saveable behavior in 4.5.2.2 on page 29.
6	Reserved
5-0	Page Code: 0Fh

1 Page Length: 0Eh

2

Bit	**Description**
7	DCE (Data Compression Enabled): 1b (changeable)

Value	**Description**
0	Data compression is not enabled
1	Data compression is enabled

NOTE 84 - The only advantage to disabling data compression is predictable full tape capacity (see 3.1.7)

6	DCC (Data Compression Capable): 1b (non-changeable)
5-0	Reserved

3

Bit	**Description**
7	DDE: 1b
6-5	RED (Report Exception on Decompression): 00b (non-changeable)
4-0	Reserved

4-7 Compression Algorithm: 000000FFh (Unregistered algorithm) (non-changeable)
A value of 00000001h which specifies the default algorithm is accepted on MODE SELECT.

8-11	Decompression Algorithm: 000000FFh (Unregistered algorithm) (non-changeable) A value of 00000001h which specifies the default algorithm is accepted on MODE SELECT.
12	Reserved
13	Reserved
14	Reserved
15	Reserved

6.6.11 MP 10h: Device Configuration

See MODE SELECT (6/10) - 15h/55h (see 5.2.14) for how to set these parameters and MODE SENSE (6/10) - 1Ah/5Ah (see 5.2.15) for how to read these parameters.Mode Page Behaviors (see 4.5 on page 27) describes this device's non-standard behaviors related to mode parameters.

The mode page policy for this page is described in Policy — non-standard (see 4.5.1 on page 27)

Table 280 — Device Configuration mode page

Bit Byte	7	6	5	4	3	2	1	0
0	PS	SPF(0b)	PAGE CODE (10h)					
1	PAGE LENGTH (0Eh)							
2	Rsvd	Obsolete	CAF(0b)	ACTIVE FORMAT (00h)				
3	ACTIVE PARTITION							
4	WRITE OBJECT BUFFER FULL RATIO (00h)							
5	READ OBJECT BUFFER EMPTY RATIO (00h)							
6	(MSB)	WRITE DELAY TIME						
7								(LSB)
8	OBR (1)	LOIS (1)	Obsolete	AVC (0)	SOCF (0)		ROBO	REW (0)
9	Obsolete							
10	EOD DEFINED (0h)			EEG (1)	SEW	SWP (0)	BAML(0)	BAM(0)
11	(MSB)	OBJECT BUFFER SIZE AT EARLY WARNING (000000h)						
12								
13								(LSB)
14	SELECT DATA COMPRESSION ALGORITHM							
15	WTRE		OIR	REWIND ON RESET (10B)		ASOCWP(0)	PERSWP(0)	PRMWP(0)

Byte Description

0

Bit	Description
7	PS: 1b. Ignored in MODE SELECT. See Parameter Saveable behavior in 4.5.2.2 on page 29.
6	Reserved
5-0	Page Code: 10h

1 Page Length: 0Eh

2

Bit Description

7 Reserved

6 CAP (Change Active Partition): 0b (non-changeable)

Value	Description
0	No partition change is specified.
1	Change logical partition to the partition specified by Active Partition.

5 CAF (Change Active Format): 0b (non-changeable)

4-0 Active Format: 00000b (non-changeable)

3 Active Partition (non-changeable) (non-changeable)

For Mode Select this field is ignored.

For Mode Sense this field specifies the current logical partition number in use on the volume.

4 Write Buffer Full Ratio: 00h (value not specified) (non-changeable)

5 Read Buffer Empty Ratio: 00h (value not specified) (non-changeable)

6-7 Write Delay Time: 0014h (about 2 sec)

8

Bit	Description
7	DBR (Data Buffer Recovery): 1b (non-changeable)
6	BIS (Block Identifier Supported): 1b (non-changeable)
5	RSmk (Report Setmarks): 0b (non-changeable)
4	AVC (Automatic Velocity Control): 0b (non-changeable)
3-2	SOCF (Stop on Consecutive Filemarks): 00b (non-changeable) 00 (read ahead to fill buffer, without regard for filemarks)
1	RBO (Recover Buffer Order): 0b (changeable)

Value	Description
0	FIFO
1	LIFO

0	REW (Report Early Warning): 0b (non-changeable)

9 Gap Size: 00h (non-changeable)

10

Bit	Description
7-5	EOD Defined (End Of Data Defined): 000b (non-changeable)
4	EEG (Enable EOD Generation): 1b (non-changeable)
3	SEW (Synchronize at Early-Warning): 1b (changeable)

NOTE 85 - Applications which expect to write significant amounts of data after EW should use a programmable early warning mechanism (see 4.6).

Value	Description
0b	In EW, no special behavior related to synchronizing data in the object buffer to the medium is performed. Repeated write commands in EW may result in a significant amount of data in the object buffer which is unable to be written to the medium.
1b	In EW, data in the object buffer is frequently synchronized to medium to minimize the likelihood and amount of data from repeated write commands in EW which may be unable to be successfully written to the medium.

2	SWP (Soft Write Protect): 0b (changeable)
1	BAML (Block Address Mode Lock): 0b (non-changeable)
0	BAM (Block Address Mode): 0b (non-changeable)

11-13 Buffer Size at Early Warning: 000000h (non-changeable)

14 Select Data Compression Algorithm: 01h (changeable)

Value	Description
00h	No compression used

NOTE 86 - The only advantage to disabling data compression is predictable full tape capacity (see 3.1.7)

01h	Use default compression algorithm (ELDC)

NOTE 87 - On a Mode Sense, the value of byte 14 will always be consistent with what is found on Mode Page 0F, Byte 2, Bit 7. If this byte alone is updated on a Mode Select, and Mode Page 0F is not sent, then Mode Page 0F, Byte 2, Bit 7 is updated to according to this field. If both Page 10 and Page 0F are sent, then what is in Page 0F is used to update both fields and any legal value in byte 14 is ignored.

15

Bit **Description**

7-6 WTRE (WORM Tamper Read Enable): 10b (changeable)

Value	**Description**
00b	The device treats a value of 00b the same as if the value were 10b.
01b	Detection of compromised integrity on a WORM medium shall not affect processing of a task.

NOTE 88 - An application client should not set the WTRE bit to 01b except for the recovery of data from a WORM medium where the integrity of the stored data has been compromised.

Value	**Description**
10b	If the drive detects compromised integrity on a WORM medium it returns CHECK CONDITION status with the sense key set to MEDIUM ERROR and the additional sense code set to WORM MEDIUM - INTEGRITY CHECK. (3/300Dh). The position of the medium may have changed.
11b	Reserved

5 OIR (Only If Reserved) 0b (not changeable on earlier code levels) (changeable-saveable)

This field dictates the behavior of commands other than RESERVE, RELEASE, PERSISTENT RESERVE IN, and PERSISTENT RESERVE OUT.

Value	**Description**
0b	Commands are processed per RVC column of Table 1. Drive Commands. No reservation is required to process the commands.
1b	Commands listed with a ‘Y’ in the RVC column of Table 1. Drive Commands are rejected with ILLEGAL REQUEST, NOT RESERVED if received and no reservation is present in the drive. If a reservation is present in the drive, the commands are processed only if a reservation exists that allows access via the I_T nexus from which the command was received.

4-3 REWIND ON RESET: 10b (non-changeable)

The position on medium is not changed due to a logical unit reset.

2 ASOCWP (Associated Write Protect): 0b (changeable)

Value	**Description**
0	No soft write protect is in effect
1	No write type commands will be allowed for the current mount

1 PERSWP (Persistent Write Protect) 0b (from media) (changeable)

Value	**Description**
0	Persistent Write Protect not in effect
1	No write type commands will be allowed on this tape.

0 PRMWP (Permanent Write Protect): 0b (from media) (changeable-special)

Value	**Description**
0	No soft write protect is in effect
1	No write type commands will ever be allowed for the mounted tape. This is permanent and cannot be reset. The media will be unwriteable after this operation is completed. Unlike Persistent Write Protect, **Permanent Write Protect can never be reset except by degaussing. The media has factory written servo formatting and is unusable if degaussed.**

Identical functions are provided by the write protect bits in vendor unique Mode Page 23. On a Mode Sense, the values in byte 15 will always match up with what is found on vendor unique Mode Page 23, Byte 10. If this byte alone is updated on a Mode Select, and Mode Page 23 is not sent, then Mode Page 23, Byte 10 is updated to match what was set in this field. If both Page 10 and Page 23 are sent, then what is in Page 10, byte 15 is used to update both fields and any legal value in Mode Page 23, Byte 10 is ignored.

6.6.12 MP 10h[01h]: Device Configuration Extension

The Device Configuration Extension mode page (see table 281), a subpage of the Device Configuration mode page (see 6.4.9), provides control of the SCSI features specific to sequential-access devices and is supported on E07 and later. Mode Page Behaviors (see 4.5 on page 27) describes this device's non-standard behaviors related to mode parameters.

The mode page policy for this page is described in Policy — non-standard (see 4.5.1 on page 27)

See MODE SELECT (6/10) - 15h/55h (see 5.2.14) for how to set these parameters and MODE SENSE (6/10) - 1Ah/5Ah (see 5.2.15) for how to read these parameters.

Table 281 — Device Configuration Extension mode page

Bit Byte	7	6	5	4	3	2	1	0
0	ps	spf(1b)	page code (10h)					
1	SUBPAGE CODE (01h)							
2	(MSB)	PAGE LENGTH (1Ch)						
3								(LSB)
4	Reserved				TARPF (0)	TASER(0)	TARPC(0)	TAPLSD(0)
5	WRITE MODE				SHORT ERASE MODE (2h)			
6	(MSB)	PEWS						
7								(LSB)
8	Reserved							VCELBRE
9	Reserved							
31								

Byte Description

0

Bit Description

7 PS: 1b. Ignored in MODE SELECT. See Parameter Saveable behavior in 4.5.2.2 on page 29.

6 SPF: 1b

5-0 PAGE CODE: 10h

1 SUBPAGE CODE: 01h

2-3 PAGE LENGTH: 1Ch

4

Bit Description

7-4 Reserved

3 TARPF: 0b (non-changeable)

2 TASER: 0b (non-changeable)

1 TARPC: 0b (non-changeable)

0 TAPLSD: 0b (non-changeable)

5

Bit Description

7-4 WRITE MODE: (0h) (changeable-saveable)

Specifies the write mode (see 4.18) in which to place the device server. Enablement is allowed when no volume is loaded or when positioned at BOP 0. Disablement is allowed when no volume is loaded. The command is rejected with ILLEGAL REQUEST, INVALID FIELD IN PARAMETER LIST if a volume is loaded and:

A) an attempt is made to change the WRITE MODE from 1h (i.e., append-only) to 0h (i.e., overwrite-allowed); or

B) the logical position of the volume is not at BOP 0 and an attempt is made to change the WRITE MODE from 0h (i.e., overwrite-allowed) to 1h (i.e., append-only).

The values supported are:

Value	Description
0h	overwrite-allowed mode (see 4.18.2).
1h	append-only mode (see 4.18.3).

NOTE 89 - Support for a value of 1h has not always been supported. Support may be discovered by examining the changeable bit mask returned in MODE SENSE.

3-0 SHORT ERASE MODE: (2h) (non-changeable)

The device server, when performing a short erase records an EOD indication at the specified location on the medium.

6-7 PEWS (programmable early warning size) (changeable-saveable)

The programmable early warning size (PEWS) field specifies the number of megabytes (10^6) native capacity to use in establishing a PEWZ. See Programmable early warning (see 4.6 on page 30) for a description of programmable early warning. E07 and later support the PEWS field set to any value and does not round the value (e.g., due to volume capacity). The default value is 0000h indicating that there is no PEWZ.

NOTE 90 - It is possible that a partition is set to a capacity less than the size that can be represented by the PEWS field. In this case the first write will get the programmable early warning indication.

NOTE 91 - PEWZ will be created using the amount of medium required to fit PEWS megabytes assuming one-to-one compression (e.g., compression disabled) and the medium is in good condition.

NOTE 92 - The drive employs a design to protect against the number of blocks on medium exceeding a value that can be represented in a 4-byte field. To ensure the programmable early warning indication is reported to an application during this scenario, the drive also uses the number of blocks on medium as a determination of entering PEWZ. The drive calculates how many blocks it will take to fill PEWZ assuming a modest block size of 32 KiB (i.e., PEWS/8000h = number_of_blocks_to_make_PEWZ) and reports programmable early warning when the number of blocks on medium reaches PEWZ (i.e., Early Warning - number_of_blocks_to_make_PEWZ).

8

Bit	Description
7-1	Reserved
0	VCELBRE (volume containing encrypted logical blocks requires encryption) (changeable-ignored)

9-31 Reserved

6.6.13 MP 11h: Medium Partition Page

See MODE SELECT (6/10) - 15h/55h (see 5.2.14) for how to set these parameters and MODE SENSE (6/10) - 1Ah/5Ah (see 5.2.15) for how to read these parameters.

This page is only supported on drives that support partitioning (i.e., 3592 E07 and later)

Mode Page Behaviors (see 4.5 on page 27) describes this device's non-standard behaviors related to mode parameters.

The mode page policy for this page is described in Policy — non-standard (see 4.5.1 on page 27)

The Medium Partition mode page (see table 282) is used to specify the group of medium partitions. Fields in the Medium Partition mode page indicating the current state of the partitions for the medium are changed by the device server to the current medium state on a not ready to ready transition when the medium state changes from demounted to mounted.

The partitioning of the mounted volume is not changed until a subsequent FORMAT MEDIUM command is issued while the volume is mounted. Note that mounting a volume that modifies the value of fields in this page does not establish a unit attention condition.

WARNING WARNING

The fields in this page do not follow normal mode parameter rules. Some fields always return values depending on the mounted volume even after a MODE SELECT command changes them. Other fields return pending values set by a MODE SELECT command instead of values dictated by the mounted volume. Check each field description to understand this behavior.

WARNING WARNING

The device ensures consistency of the partitioning values set in this mode page by causing a CHECK CONDITION status with the sense key set to ILLEGAL REQUEST and the additional sense code set to PARAMETER VALUE INVALID to be returned to a subsequent FORMAT MEDIUM command attempting to use this mode page if values in this page become inconsistent between when they were set and when the FORMAT MEDIUM command is received. The invalidation of the values in this page is accomplished by setting the FDP, SDP, and IDP bits to zero and the other values in this page to:

a) the values dictated by the format of the currently mounted volume, if a volume is mounted; or
b) the default values present after power-on, if no volume is mounted.

The invalidation of values in this page occurs if:

a) THE PENDING WRITE DENSITY AT BOP 0 field of MP 25h: Read/Write Control (see 6.6.21 on page 427) is modified;
b) the volume is capacity scaled (see MP 23h: Medium Sense (see 6.6.19 on page 414)); or
c) other events that are determined to make the values in this page inconsistent.

There is an overhead associated with each additional partition, regardless of the size of the partition, that subtracts from the customer data space on the volume. The larger the number of possible partitions, the more overhead is consumed when the volume is partitioned.

See Partitioning and reformatting (see 4.19.6 on page 75) for restrictions on partitioning scaled volumes.

Table 282 — Medium Partition mode page

Bit Byte	7	6	5	4	3	2	1	0
0	PS(0b)	SPF(0b)	PAGE CODE (11h)					
1	PAGE LENGTH (n-1)							
2	MAXIMUM ADDITIONAL PARTITIONS							
3	ADDITIONAL PARTITIONS DEFINED							
4	FDP	SDP	IDP	PSUM (11B)		POFM (1b))	CLEAR (0B)	ADDP(0B)
5	MEDIUM FORMAT RECOGNITION (03h)							
6	PARTITIONING TYPE				PARTITION UNITS			
7	Reserved							
Partition size descriptor(s)								
8	(MSB)	PARTITION SIZE (first)						
9								(LSB)
n-1	(MSB)	PARTITION SIZE (last)						
n								(LSB)

Byte Description

0

Bit	**Description**
7	PS (Page Save): 0b. Ignored in MODE SELECT. See Parameter Saveable behavior in 4.5.2.2 on page 29.
6	SPF (SubPage Format): 0b
5-0	PAGE CODE: 11h

1 PAGE LENGTH:

In a MODE SENSE command: 0Eh (14).

In a MODE SELECT command:

Condition	**Description**
FDP=1	6 + 2 * k; where 0 <= K <= 4
SDP=1	6 + 2 * k; where 0 <= K <= 4
IDP=1	6 + 2 * k; where (ADDITIONAL PARTITIONS DEFINED + 1) <= K <= 4.

2 MAXIMUM ADDITIONAL PARTITIONS: (non-changeable)

This field specifies the maximum number of additional partitions supported on the loaded volume at the PENDING WRITE DENSITY AT BOP 0. It can be thought of as the value N-1 where N is the maximum number of partitions allowed. In a MODE SENSE command the MAXIMUM ADDITIONAL PARTITIONS field is set to the value indicated by the loaded volume at the PENDING WRITE DENSITY AT BOP 0, or the value reported for the previously loaded volume if there is no volume loaded. If there has been no volume loaded since power-on or the mounted volume does not support partitioning (see 4.19), then the MAXIMUM ADDITIONAL PARTITIONS field is set to 00h.

This value may be changed by the device if:

a) a volume is loaded; or
A) other unspecified events occur.

3 ADDITIONAL PARTITIONS DEFINED: 0h (changeable) on E07 and later

This field specifies the number of additional partitions on the mounted volume (in addition to partition 0). It can be thought of as the value N-1 where N is the total number of partitions.

If SDP is set to one or IDP is set to one, then the values for this field may be;

Value	**Description**
00h	The default partition is the only partition.
01h	There is one additional partition making a total of two.
02h	There are two additional partitions making a total of three. This value is not supported if the mounted volume is a WORM volume.
03h	There are three additional partitions making a total of four. This value is not supported if the mounted volume is a WORM volume.
others	Not supported.

This field is ignored when FDP is set to one (i.e., any value is allowed and ignored).

This field is not allowed to change when the drive is not ready.

The partitioning of the mounted volume is not changed until a subsequent Format Medium command is issued while the volume is mounted. If the logical unit is not ready, the ADDITIONAL PARTITIONS DEFINED field is undefined.

NOTE 93 - If a MODE SENSE command is received after a MODE SELECT command but before a FORMAT MEDIUM command, then the ADDITIONAL PARTITIONS DEFINED value returned is the value sent in the MODE SELECT command.

4

Bit **Description**

7 FDP (Fixed Data Partitions): 0b (changeable)

A FDP bit of one in a MODE SELECT command specifies the logical unit shall partition the medium based on its fixed definition of partitions. Setting this bit to one is mutually exclusive with the SDP and IDP bits. The ADDITIONAL PARTITIONS DEFINED field, the PARTITIONING TYPE field, and the partition size descriptors are ignored by the MODE SELECT command when the FDP bit is set to one. The drive creates two partitions on the volume and assigns one partition as the minimum sized partition and one partition as the remaining available size.

When a MODE SENSE command is received after a MODE SELECT command but before a FORMAT MEDIUM command the FDP bit returned is the value that was set in the MODE SELECT command. When a MODE SENSE command is received and there is no pending change related to a MODE SELECT command for this page the FDP bit is set to zero.

NOTE 94 - The partition size descriptors are present in MODE SENSE data regardless of the settings of the FDP, SDP or IDP fields to give an estimate of the size of each partition.

6 SDP (Select Data Partitions): 0b (changeable)

For Mode Select, either value is allowed

Value	**Description**
0b	The SDP functionality is not used.
1b	The volume is partitioned into the number of partitions as specified by the ADDITIONAL PARTITIONS DEFINED field (n) using partitions as close to equal size as possible. If rounding is required to meet format requirements, partitions will be rounded up from the equal size and the last partition (i.e., n + 1) will use the remaining capacity. The drive partitions the volume into n+1 partitions numbered 0 through n. Setting this bit to one is mutually exclusive with the FDP and IDP fields. The partition size descriptors are ignored by the MODE SELECT command when the SDP bit is set to one.

When a MODE SENSE command is received after a MODE SELECT command but before a FORMAT command the SDP bit returned is the value that was set in the MODE SELECT command. When a MODE SENSE command is received and there is no pending change related to a MODE SELECT command for this page the SDP bit is set to zero.

5 IDP (Initiator Defined Partitions): 0b (changeable)

For Mode Select, either value is allowed

Value	Description
0b	The IDP functionality is not used.
1b	The volume is partitioned as specified by the ADDITIONAL PARTITIONS DEFINED field and the partition size descriptors. Setting this bit to one is mutually exclusive with the FDP and SDP fields. The number of non-zero partition size descriptors received in the Medium Partition mode page shall be one more than the ADDITIONAL PARTITIONS DEFINED value. The size of partition 0 shall be non-zero.

When a MODE SENSE command is received after a MODE SELECT command but before a FORMAT MEDIUM command the IDP bit returned is the value that was set in the MODE SELECT command. When a MODE SENSE command is received and there is no pending change related to a MODE SELECT command for this page the IDP bit is set to one.

4-3 PSUM (Partition Size Unit of Measure): 11b ($10^{(\text{PARTITION UNITS})}$ bytes) (changeable)

2 POFM (Partition on Format Medium): 1b (non-changeable)

This bit indicates that the Mode Select command will not cause changes to the partition sizes or user data, either recorded or buffered. Actual media partitioning occurs with a subsequent Format Medium command using the mode data for this page. Field values specified by a Mode Select command for this page will not be changed by the drive before the volume is unloaded or the drive is reset. Some field checking may be performed by the MODE SELECT command. However, there is no guarantee that any subsequent partitioning during a FORMAT MEDIUM command will complete with no errors.

1 CLEAR (Partition clearing): 0b (non-changeable)

0 ADDP (Adding Partitions): 0b (non-changeable)

When a MODE SENSE command is received after a MODE SELECT command but before a FORMAT MEDIUM command the values returned in the fields in byte 4 are the values that were set in the MODE SELECT command. When a MODE SENSE command is received and there is no pending change related to a MODE SELECT command for this page, then the values returned in the fields in byte 4 depend on the mounted volume.

See Volume partitioning (see 4.19 on page 63) for a detailed description of how to select partition sizes.

5 MEDIUM FORMAT RECOGNITION: 03h (Capable of format and partition recognition) (non-changeable)

6

Bit **Description**

7-4 PARTITIONING TYPE: 0h (changeable)

This field is ignored if the FDP bit is set to one.

The PARTITIONING TYPE field specifies the criteria used to describe the partitions.

Value	Description
0h	The type of partitioning is unknown The drive sets the PARTITIONING TYPE field to a value of 1h when a 0h is received in a MODE SELECT command. If the volume is not partitioned, then a MODE SENSE command returns this value unless there is a pending action from a MODE SELECT
1h	The type of partitioning is optimized for streaming performance (i.e., wrap-wise partitioning with no FastSync). On WORM volumes, when this type is selected, there is a maximum of two partitions. On non-WORM volumes, when this type is selected there is a

	maximum of four partitions. See Wrap-wise Partitioning (see 4.19.2 on page 63) for a detailed description of how to select partition sizes.
2h	The type of partitioning is optimized for random access performance (i.e., longitudinal partitioning). This type of partitioning is not supported on WORM volumes. When this value is selected there may be a maximum of two partitions. See Longitudinal Partitioning (see 4.19.3 on page 73) for a detailed description of how to select partition sizes. A volume that is partitioned with longitudinal partitioning may be detected in a device that has a firmware level that does not support longitudinal partitioning. If a device operating a firmware level that does not support longitudinal partitioning detects that a volume is partitioned with longitudinal partitioning, then the volume is rejected as an unsupported format.
3h	The type of partitioning reduces the total native capacity of the volume as part of optimizing for streaming performance (i.e., wrap-wise partitioning using FastSync) This type of partitioning is not supported on WORM volumes. When this value is selected there may be a maximum of four partitions. See Wrap-wise Partitioning with FastSync (see 4.19.2.2 on page 68) for a detailed description of how to select partition sizes.
4h-Eh	Reserved
Fh	For a MODE SENSE command, this value is not returned because this device does not support multiple types of partitioning on the same volume. For a MODE SELECT command this value is reserved.

When a MODE SENSE command is received after a MODE SELECT command but before a FORMAT MEDIUM command the PARTITIONING TYPE field returned is the value that was set in the MODE SELECT command. When a MODE SENSE command is received and there is no pending change related to a MODE SELECT command for this page the PARTITIONING TYPE field is set depending on how the volume is currently partitioned.

3-0 PARTITION UNITS: (changeable)

The PARTITION UNITS is used in a MODE SELECT to define the value of the PARTITION SIZE descriptors.When a MODE SENSE command is received after a MODE SELECT command but before a FORMAT MEDIUM command the PARTITION UNITS field returned is the value that was set in the MODE SELECT command. When a MODE SENSE command is received and there is no pending change related to a MODE SELECT command for this page, then the PARTITION UNITS field is set depending on how the volume is currently partitioned.

7 Reserved

8-n PARTITION SIZE descriptors: (n=9, 11, 13, or 15) (changeable)

Each PARTITION SIZE descriptor specifies the size of a partition in $10^{\text{PARTITION UNITS}}$ bytes. The device rounds, any value received in a PARTITION SIZE descriptor to the nearest valid partition size. Volume partitioning (see 4.19 on page 63) provides a detailed description of how to select values for each PARTITION SIZE descriptor.

Byte	Description
8-9	PARTITION SIZE descriptor for partition 00h This shall exist and shall be non-zero.
10-11	PARTITION SIZE descriptor for partition 01h, if sent This descriptor shall be present and non-zero if the ADDITIONAL PARTITIONS DEFINED field is 01h or greater. This descriptor may be present and set to zero if the ADDITIONAL PARTITIONS DEFINED field is 00h.
12-13	PARTITION SIZE descriptor for partition 02h, if sent This descriptor shall be present and non-zero if the ADDITIONAL PARTITIONS DEFINED field is 02h or greater. This descriptor may be present and set to zero if the ADDITIONAL PARTITIONS DEFINED field is less than 02h.
14-15	PARTITION SIZE descriptor for partition 03h, if sent This descriptor shall be present and non-zero if the ADDITIONAL PARTITIONS DEFINED field is 03h. This descriptor may be present and set to zero if the ADDITIONAL PARTITIONS DEFINED field is less than 03h.

NOTE 95 - Summing the partition sizes provides a standardized mechanism for an initiator to calculate the volume capacity with MODE SENSE.

In a MODE SELECT command:

  a) a value of FFFFh received in a PARTITION SIZE descriptor, requests that the logical unit allocate all remaining partition space to that partition;

B) if the FDP bit is set to one any values are allowed in the PARTITION SIZE descriptors and ignored;

C) if the SDP bit is set to one any values are allowed in the PARTITION SIZE descriptors and ignored. The PARTITION SIZE descriptors are updated by the drive when an initiator specifies a new number of partitions; and

D) the device server returns CHECK CONDITION status with the Sense Key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN PARAMETER LIST if:
    A) insufficient space exists on the volume for the requested partition sizes; or
    B) multiple partition size descriptors are set to FFFFh.

In a MODE SENSE command:

  a) All four PARTITION SIZE descriptors are always returned in a MODE SENSE command.

E) a value of FFFFh returned in a PARTITION SIZE descriptor, indicates the partition size, in units indicated by PSUM and PARTITION UNITS, is greater than or equal to FFFFh;

F) if the logical unit is not ready, then the PARTITION SIZE descriptors are undefined;

G) if a MODE SELECT command has modified values in this page but a FORMAT MEDIUM command has not yet been processed, then the values in the PARTITION SIZE fields are the values set by the MODE SELECT command rounded to a valid partition size as described in Volume partitioning (see 4.19 on page 63); and

H) if the logical unit is ready and there is no pending change related to a MODE SELECT command for this page, then the PARTITION SIZE fields reflect the size on the volume of the related partition.

NOTE 96 - When more than one partition is defined, the sum of the partition sizes may be less than when only a single partition is defined. Each partition requires a certain amount of overhead space on a volume, which reduces the usable customer data space.

6.6.14 MP 18h: Fibre Channel Logical Unit Control

See MODE SELECT (6/10) - 15h/55h (see 5.2.14) for how to set these parameters and MODE SENSE (6/10) - 1Ah/5Ah (see 5.2.15) for how to read these parameters.

Mode Page Behaviors (see 4.5 on page 27) describes this device's non-standard behaviors related to mode parameters.

The mode page policy for this page is described in Policy — non-standard (see 4.5.1 on page 27)

This page is defined for Fibre Channel attached devices only.

.

Table 283 — Fibre Channel Logical Unit mode page

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	PS	SPF (0b)	PAGE CODE (18h)					
1	PAGE LENGTH (n-1)							
2	Reserved				PROTOCOL IDENTIFIER			
3	Reserved							EPDC
4	Reserved							
n								

Byte **Description**

0

Bit	Description
7	PS (Page Save): 0b. Ignored in MODE SELECT. See Parameter Saveable behavior in 4.5.2.2 on page 29.
6	Reserved
5-0	Page Code: 18h

1 Page Length: 06h

2

Bit	Description
7-1	Reserved
3-0	PROTOCOL IDENTIFIER: 0h (FCP) (non-changeable)

3

Bit	Description
7-1	Reserved
0	EPDC (Enable Precise Delivery Control): 1b (changeable)

4-7 Reserved

6.6.15 MP 19h: Fibre Channel Port Control

See MODE SELECT (6/10) - 15h/55h (see 5.2.14) for how to set these parameters and MODE SENSE (6/10) - 1Ah/5Ah (see 5.2.15) for how to read these parameters.

Mode Page Behaviors (see 4.5 on page 27) describes this device's non-standard behaviors related to mode parameters.

The mode page policy for this page is described in Policy — non-standard (see 4.5.1 on page 27)

This page is defined for Fibre Channel attached devices only.

Table 284 — Fibre Channel Port mode page

Byte	Bit							
	7	6	5	4	3	2	1	0
0	PS (0)	SPF (0)	PAGE CODE (19h)					
1	PAGE LENGTH							
2	Reserved				PROTOCOL IDENTIFIER			
3	DTFD (0)	PLPB (0)	DDIS (0)	DLM (0)	RHA (0)	ALWI (0)	DTIPE (0)	DTOLI (0)
4	Reserved							
5								
6	Reserved					RR_TOV UNITS		
7	Resource Recovery Time Out Value (RR_TOV)							

Byte **Description**

0

Bit	Description
7	PS (Page Save): 0b. Ignored in MODE SELECT. See Parameter Saveable behavior in 4.5.2.2 on page 29.
6	Reserved
5-0	Page Code: 19h

1 Page Length: 06h

NOTE 97 - A Page Length of 0Eh is allowed on Mode Select for backwards compatibility with older devices and levels of the standard. Any fields beyond byte 7 are (changeable-ignored).

2 Reserved

3

Bit	Description
7	DTFD (Disable Target Fabric Discovery): 0b (non-changeable)
6	PLPB (Prevent Loop Port Bypass): 0b (non-changeable)
5	DDIS (Disable Discovery): 0b (non-changeable)
4	DLM (Disable Loop Master): 0b (non-changeable)
3	RHA (Require Hard Address): 0b (non-changeable)
2	ALWI (Allow Login without Loop Initialization): 0b (non-changeable)
1	DTIPE (Disable Target Initiated Port Enable): 0b (non-changeable)
0	DTOLI (Disable Target Originated Loop Initialization): 0b (non-changeable)

4-5 Reserved

6

Bit	Description
7-3	Reserved
2-0	RR_TOV Units (changeable)

Value	Description
000b	No timer is specified
001b	Timer is specified in.001 second units
011b	Timer is specified in.1 second units
101b	Timer is specified in 10 second units

7 RR_TOV (Resource Recovery Time Out Value): (changeable)

NOTE 98 - The default RR_TOV value is 25 seconds.

6.6.16 MP 1Ch: Informational Exceptions Control

See MODE SELECT (6/10) - 15h/55h (see 5.2.14) for how to set these parameters and MODE SENSE (6/10) - 1Ah/5Ah (see 5.2.15) for how to read these parameters. Mode Page Behaviors (see 4.5 on page 27) describes this device’s non-standard behaviors related to mode parameters.

The mode page policy for this page is described in Policy — non-standard (see 4.5.1 on page 27)

Table 285 — Informational Exceptions Control mode page

Byte	Bit							
	7	6	5	4	3	2	1	0
0	PS	SPF (0b)	PAGE CODE (1Ch)					
1	PAGE LENGTH (0Ah)							
2	PERF	Reserved	EBF	EWASC	DEXCPT	TEST	EBACKERR	LOGERR
3	Reserved				MRIE			
4	(MSB)	INTERVAL TIMER						
7								(LSB)
8	(MSB)	REPORT COUNT / TEST FLAG NUMBER						
11								(LSB)

Byte Description

0

Bit	Description
7	PS (Parameter Saveable): 0b. Ignored in MODE SELECT. See Parameter Saveable behavior in 4.5.2.2 on page 29.
6	SPF (Subpage Format): 0b
5-0	PAGE CODE: 1Ch

1 PAGE LENGTH: 0Ah

2

Bit	Description
7	PERF(performance): 0b (non-changeable)
6	Reserved
5	EBF (enable background function): 0b (non-changeable)
4	EWASC (enable warning): 0b (non-changeable)
3	DEXCPT (disable exception control): 1b (changeable)

Value	Description
0b	Exception Reporting is enabled
1b	Exception Reporting is disabled

Bit	Description
2	TEST: 0b (changeable-special)

This bit may be written to 1b and the action described under Value 1b is performed. The value returned to a MODE SENSE command is always 0b.

Value	Description
0b	Commands are processed normally.
1b	The behavior depends on the setting of the DEXCPT bit as follows:

DEXCPT	Description
0b	An exception information condition is asserted according to the REPORT COUNT / TEST FLAG NUMBER field as indicated by the MRIE field.
1b	The command is rejected with 5/2600h (ILLEGAL REQUEST, INVALID FIELD IN PARAMETER DATA).

Bit	Description
1	EBACKERR (enable background error): 0b (non-changeable))
0	LOGERR (log errors): 0b (non-changeable)

3

Bit	Description
7-4	Reserved
3-0	MRIE: 4h (changeable)

Value	Description
0h	If an exception condition exists, it is not reported.
3h	If an exception condition exists and the PER bit in the Read-Write Error Recovery mode page is set to one, unit attention eligible commands (<<see UAT in Table 30 commands table>>) on this I_T nexus that complete with status GOOD have the status modified to CHECK CONDITION and return non-deferred sense data of 1/5D00h (RECOVERED ERROR, FAILURE PREDICTION THRESHOLD EXCEEDED).
4h	If an exception condition exists, unit attention eligible commands (<<see UAT in Table 30 commands table>>) on this I_T nexus that complete with status GOOD have the status modified to CHECK CONDITION and return non-deferred sense data of 1/5D00h (RECOVERED ERROR, FAILURE PREDICTION THRESHOLD EXCEEDED).

4-7 INTERVAL TIMER: 00000000h (non-changeable)

8-11 REPORT COUNT / TEST FLAG NUMBER: 00000000h (changeable-special)

If TEST=0b, this field is not changeable.

If TEST=1b, the value is not retained in the mode page. The following describes the test behavior.

Value	Description
0000 0000h	The device server shall not activate or deactivate any TapeAlert flag. After the MODE SELECT command completes, the device server reports an informational exception condition one time as specified by MRIE except with the additional sense code set to 5DFFh (FAILOVER PREDICTION THRESHOLD EXCEEDED (FALSE)). Then, if an exception condition exists, it is reported as specified by MRIE.
0000 0001h to 0000 0040h	The device server activates the TapeAlert flag specified by the REPORT COUNT/TEST FLAG NUMBER field. After the MODE SELECT completes the device server reports an informational exception condition one time as specified by MRIE except with an addi-

	tional sense code of 5DFFh (FAILOVER PREDICTION THRESHOLD EXCEEDED (FALSE)). Then, if an exception condition exists, it is reported as specified by MRIE. NOTE 99 - If the specified TapeAlert flag was already active, then the behavior may be the same as 0000_0000h. To guarantee that a TapeAlert flag causes a non-test exception condition to exist, the deactivate (i.e., FFFF_FFFFh to FFFF_FFC0h) should be used prior to the activate. A non-test exception condition exists in any case where a LOG SENSE of log page 2Eh returns a non-zero TapeAlert flag. Reading log page 2Eh with LOG SENSE effectively clears the exception condition until a TapeAlert condition is activated. NOTE 100 - While the scope of this page is <Per I_T nexus>, the activated or deactivated TapeAlerts have the full effects of normally occurring TapeAlerts on other initiators, on libraries, etc. After performing a test, TapeAlert flags activated using this mechanism should be deactivated before resuming normal operation.
FFFF FFFFh to FFFF FFC0h	The device server shall deactivate the TapeAlert flag specified by the absolute value of the REPORT COUNT/TEST FLAG NUMBER field. Deactivating the flag in this way is equivalent to performing the specified corrective action for that flag. Then, if an exception condition exists, it is reported as specified by MRIE.

6.6.17 MP 21h: TOD Control

See MODE SELECT (6/10) - 15h/55h (see 5.2.14) for how to set these parameters and MODE SENSE (6/10) - 1Ah/5Ah (see 5.2.15) for how to read these parameters.

The TOD (Time-of-Day) control is used to provide the device with an estimate of the correct time. The device adds the current TOD clock to every block transferred to media. These time traces can then be used by engineering for analysis of medium at a later time. It is strongly recommended to device driver writers with access to system or network clocks to use this mode page to enable system time based device tracing.

The Time-of-Day clock is a binary counter with a 64-bit format: bit 63 being the highest value; bit 0 being the lowest. Bit 32 represents a 1 second clock; that is, the TOD is incremented by 1 in bit position 32, once every second. This gives the TOD clock a cycle time of approximately 136 years. Setting the high order byte to any number greater than F0 will result in parameter rounding down to F0 with CHECK CONDITION status and associated sense data of 1/3700 (Recovered Error, Rounded Parameter). The time of day may be set at any time. Page Code 21 is designed to provide the time-of-day clock setting to the device for 3592 format data.

Mode Page Behaviors (see 4.5 on page 27) describes this device’s non-standard behaviors related to mode parameters.

The mode page policy for this page is described in Policy — non-standard (see 4.5.1 on page 27)

WARNING WARNING

The Time-of-Day clock may be set by the library in which the drive resides. If a library sets the Time-of-Day clock the value set by the library will be used. An initiator may still issue a Mode Select with this page and attempt to set the Time-of-Day clock and will get a GOOD status returned. However, the value set by the library will still be the value used.

WARNING WARNING

Byte	Description
0	

Bit	Description
7	PS (Page Save): 0b. Ignored in MODE SELECT. See Parameter Saveable behavior in 4.5.2.2 on page 29.
6	Reserved
5-0	Page Code: 21h

Byte	Description
1	Page Length: 0Ah
2-9	Time of Day: (changeable) The default time setting at power on is 0000 0000 0000 0000h. The time setting may be preserved across certain reset conditions. The Time of Day (TOD) clock begins counting relative time from that point. When sensed, the TOD field returns the current drive time. If the TOD field has never been set, the TOD field contains the elapsed power on time from the default value of 0000 0000 0000 0000h, which corresponds to a time of January 1, 1970, 0:00 AM, Greenwich Mean Time (GMT).

10 Time-of-Day Flags

Bit	Description
7-3	Vendor-Reserved
2	TOD Reset Valid: 0b (changeable-special)

Value	Description
0b	The TOD clock will not be changed (always on Mode Sense)
1b	The TOD clock will be set to Time of Day (Mode Select only)

Bit	Description
1	SysSet TOD: (non-changeable)

Value	Description
0b	The TOD was not set by an initiator
1b	The TOD was set by an initiator

Bit	Description
0	Relative TOD: (non-changeable)

Value	Description
0b	The TOD was not set by the device
1b	The TOD was set by the device

11 Reserved

6.6.18 MP 22h: Language

See MODE SELECT (6/10) - 15h/55h (see 5.2.14) for how to set these parameters and MODE SENSE (6/10) - 1Ah/5Ah (see 5.2.15) for how to read these parameters.

Mode Page Behaviors (see 4.5 on page 27) describes this device's non-standard behaviors related to mode parameters.

The mode page policy for this page is described in Policy — non-standard (see 4.5.1 on page 27)

> NOTE 101 - Operator languages other than English are not supported for this device. This mode page is supported for backwards compatibility only.

Byte	Description
0	

Bit	Description
7	PS (Page Save): 0b. Ignored in MODE SELECT. See Parameter Saveable behavior in 4.5.2.2 on page 29.
6	Reserved
5-0	Page Code: 22h

Byte	Description
1	Page Length: 02h
2	Current Language (non-changeable) When sensed, this field indicates the current language. The following languages are supported:

Value	Language
00h	U.S. English

Byte	Description
3	Requested Language: (changeable-ignored)

6.6.19 MP 23h: Medium Sense

See MODE SELECT (6/10) - 15h/55h (see 5.2.14) for how to set these parameters and MODE SENSE (6/10) - 1Ah/5Ah (see 5.2.15) for how to read these parameters.

The Medium Sense page provides information about the state of the medium currently associated with the device, if any.

Mode Page Behaviors (see 4.5 on page 27) describes this device's non-standard behaviors related to mode parameters.

The mode page policy for this page is described in Policy — non-standard (see 4.5.1 on page 27)

> NOTE 102 - Issuing a Mode Sense for current values before a Mode Select is generally recommended to avoid accidentally attempting to set fields that cannot be changed by the initiator. *Not all fields in this page can be set by users. All fields other than those explicitly indicated that can be set by users are read-only.*

Byte	**Description**
0	

Bit	**Description**
7	PS (Page Save): 0b. Ignored in MODE SELECT. See Parameter Saveable behavior in 4.5.2.2 on page 29.
6	Reserved
5-0	Page Code: 23h

1	Page Length: 3Ah
2-3	Vendor-Reserved
4-5	Medium Identifier: (non-changeable) When there is an associated medium, this field contains the medium identifier of the associated medium.

Value	**Description**
0000h	No medium present or type is invalid or unknown
0141h	Medium present is 3592 Enterprise Tape — Standard Cartridge (JA)
0142h	Medium present is 3592 Enterprise Tape — Extended Cartridge (JB)
0143h	Medium present is 3592 Enterprise Tape— Advanced Type C Cartridge (JC)
0144h	Medium present is 3592 Enterprise Tape— Advanced Type D Cartridge (JD)
0151h	Medium present is 3592 Enterprise Tape — ECONOMY Cartridge (JJ)
0152h	Medium present is 3592 Enterprise Tape — Advanced Type C Economy Cartridge (JK)
0153h	Medium present is 3592 Enterprise Tape — Advanced Type D Economy Cartridge (JL)
0241h	Medium present is 3592 Enterprise Tape — Standard WORM Cartridge (JW)
0242h	Medium present is 3592 Enterprise Tape — Extended WORM Cartridge (JX)
0243h	Medium present is 3592 Enterprise Tape — Advanced Type C WORM Cartridge (JY)
0244h	Medium present is 3592 Enterprise Tape — Advanced Type D WORM Cartridge (JZ)
0251h	Medium present is 3592 Enterprise Tape - ECONOMY WORM Cartridge (JR)

6 Format Identifier: (non-changeable)

When there is an associated medium, this field contains the format identifier of the associated medium.

Value	Description
00	No medium present of format is invalid or unknown
05	FMR cartridge (any format)
31	Medium present (only JA/JW; JJ/JR medium) is written in 3592A1 density
32	Medium present (only JA/JW; JJ/JR medium) is written in 3592A2 density
33	Medium present (only JA/JW; JJ/JR; JB/JX medium) is written in 3592A3 density
34	Medium present (only JB/JX; JC/JY/JK medium) is written in 3592A4 density
35	Medium present (only JC/JY/JK; JD/JZ/JL medium) is written in 3592A5 density
36	Medium present (only JD/JZ/JL medium) is written in 3592B5 density
37	Medium present (only JE/JV/JM medium) is written in 3592A6 density
71	Medium present (only JA/JW; JJ/JR medium) is written encrypted in 3592A1
72	Medium present (only JA/JW; JJ/JR; JB/JX medium) is written encrypted in 3592A2 density
73	Medium present (only JA/JW; JJ/JR; JB/JX medium) is written encrypted in 3592A3 density
74	Medium present (only JB/JX; JC/JY/JK medium) is written encrypted in 3592A4 density
75	Medium present (only JC/JY/JK; JD/JZ/JL medium) is written encrypted in 3592A5 density
76	Medium present (only JD/JZ/JL medium) is written encrypted in 3592B5 density
77	Medium present (only JE/JV/JM medium) is written encrypted in 3592A6 density

7 Partition Information (non-changeable)

Bit	Description
7-4	PARTITIONING TYPE - The type of partitioning currently in use on the volume (see 6.6.13)
3-0	Reserved

8 ACTIVE PARTITION - The partition number of the current logical position. (non-changeable)

9 WORM Control

Bits	Description
7–5	Reserved
4	Vendor-Reserved (changeable) (unsupported)
3–0	WORM Mode (changeable)

Value	Description
0h	None
2h	Relabel/Reappend (Allow overwrite of working construct)

10 Write Protect Flags

There are two forms of write protect: logical and physical, and three types of logical write protect. Each performs essentially the same function: each protects the customers data from change. Each write protect method performs that function; the only difference is the permanence of the effect (but note the differences with respect to CM validity for Associated Write Protection).

Bits **Description**

7 Physical Write Protect:

This field is also found in “REQUEST SENSE - 03h” on page 198 (byte 24, bit 1).

This field indicates the state of the physical Write Protect switch located on the cartridge. This switch is controlled by the user. When the switch is set to 1, the entire physical volume is set to the write protected state; when the switch is set to 0, the volume is physically write enabled.

Value	Description
0b	The cartridge write protect switch is set to write enabled.
1b	The cartridge write protect switch is set to write protected state.

This field may be changed only by physically changing the state of the switch on the cartridge. After the cartridge is loaded into the device, the write protect switch is not available to the user, and, therefore, does not change states while mounted.

6 Associated Write Protect: (changeable)

An OR condition of the three forms of logical write protect may be found in “REQUEST SENSE - 03h” on page 198 (byte 24, bit 0).

This field allows an initiator to set the logical volume to the Associated Write Protected state. Associated Write Protect protects a volume only while the logical volume is associated with (mounted on) the device.

For a Mode Select command, this field has the following meaning:

Value	Description
0b	Do not change the Associated Write Protect state.
1b	Set this logical volume to the Associated Write Protect state.

For a Mode Sense command, this field has the following meaning:

Value	Description
0b	This logical volume is not set to the Associated Write Protect state.
1b	This logical volume is set to the Associated Write Protect state.

The logical volume can be set to Associated Write Protect at any time. The medium need not be positioned to Beginning of Partition (BOP).

When the initiator accepts the GOOD status from the Mode Select command, the logical volume is write protected. Buffered write data is not necessarily transferred to the medium prior to the completion of the Mode Select command. It is transferred at such time as it would have been had no further writes been issued (in any case prior to cartridge unload).

The Associated Write Protect state remains only as long as the medium is associated with (mounted on) the device. Both Unload and Power Off return the logical volume to the default state of write enabled. The initiator can also write enable the logical volume by issuing the Mode Select command with the Reset Associated Write Protect field set to 1b.

While in the Associated Write Protect state, any attempt to repartition, reformat, or write results in CHECK CONDITION status with associated sense data of 7/2700 (Data Protect, Write Protected).

NOTE 103 - For products with a CE service panel, the write protect icon will appear on the status screen when the medium is physically or logically write protected. Except for the time span covered by cartridge loading and cartridge unloading, the icon will remain on until the initiator requests the write protection status to change or unload or power-off occurs, which reset the status by default. During load and unload the icon will revert to the unprotected indicator. This is due to updates to the tape Housekeeping region, which requires writing. Unlike Physical Write Protection, Persistent Write Protection, or Permanent Write Protection, the CM is subject to change when under Associated Write Protection. This means that unexpected power-offs during the load or unload process while Associated Write Protection is set may result in the CM being in an invalid state, just as is possible when no write protection is active. Subsequent locate and space operations to the volume will automatically rebuild the tape directory.

5 Reset Associated Write Protect (changeable)

A 1h in this field causes the Associated Write Protect state to be reset; that is, to change the state of the logical volume from write protected to write enabled. After being reset, the logical volume again accepts write commands.

For a Mode Select command, this field has the following meaning:

Value	**Description**
0b	Do not change the Associated Write Protect state.
1b	Reset the Associated Write Protect state for this logical volume.

For a Mode Sense command, this field is 0b.

When the device successfully executes a Reset Associated Write Protect, the device immediately resets the write protected state to the write enabled state and allows write commands from that point.

4 Persistent Write Protect (changeable)

An OR condition of the three forms of logical write protect may be found in “REQUEST SENSE - 03h” on page 198 (byte 24, bit 0).

This field allows an initiator to set the logical volume to the write protected state. Unlike the Associated Write Protect, Persistent Write Protect persists across mount cycles because the state is written in the housekeeping area of the volume.

For a Mode Select command, this field has the following meaning:

Value	**Description**
0b	Do not change the Persistent Write Protect state.
1b	Set this logical volume to the Persistent Write Protect state.

For a Mode Sense command, this field has the following meaning:

Value	**Description**
0b	This logical volume is not set to the Persistent Write Protect state.
1b	This logical volume is set to the Persistent Write Protect state.

The volume can be set to Persistent Write Protect only when the medium is positioned to Beginning of Partition 0 (BOP 0). The device writes the Persistent Write Protect field in the CM and presents status to the initiator. If the command fails, persistent write protection cannot be guaranteed.

Unlike Permanent Write Protect (see below), the volume may be write enabled by issuing a Mode Select with the Reset Persistent Write Protect field set to 1b. Similar to Permanent Write Protect, if a partition is set to Persistent Write Protect, the physical volume may not be repartitioned unless all partitions on the volume are write enabled. Any attempt to repartition, reformat, or write results in CHECK CONDITION status with associated sense data of 7/2700 (Data Protect, Write Protected).

3 Reset Persistent Write Protect (changeable-special)

A 1b in this field causes the Persistent Write Protect field to be reset; that is, to change the state of the logical volume from write protected to write enabled. After being reset, the logical volume again accepts write commands.

For a Mode Select command, this field has the following meaning:

Value	**Description**
0b	Do not change the Persistent Write Protect state.
1b	Reset the Persistent Write Protect state for this logical volume.

For a Mode Sense command, this field is 0b.

The device must be at BOP 0 to accept Reset Persistent Write Protect.

2-1 Vendor-Reserved

0 Permanent Write Protect (changeable-special)

An OR condition of the three forms of logical write protect may be found in "REQUEST SENSE - 03h" on page 198 (byte 24, bit 0).

This field allows an initiator to set the logical volume to a permanently write protected state. Similar to the Persistent Write Protect, Permanent Write Protect persists across mount cycle because the state is written in the CM of the physical volume. Unlike Persistent Write Protect, Permanent Write Protect can never be reset except by degaussing. The media has factory written servo formatting and is unusable if degaussed.

For a Mode Select command, this field has the following meaning:

Value	**Description**
0b	Do not change the Permanent Write Protect state.
1b	Set this logical volume to the Permanent Write Protect state. After being set, THIS LOGICAL VOLUME CAN NEVER BE WRITTEN ON AGAIN!

For a Mode Sense command, this field has the following meaning:

Value	**Description**
0b	This logical volume is not set to the Permanent Write Protect state.
1b	This logical volume is set to the Permanent Write Protect state.

The volume can be set to the Permanent Write Protect state only when the medium is positioned to Beginning of Partition 0 (BOP 0). The device writes the Permanent Write Protect field in the CM and presents status to the initiator. Any attempt to repartition, reformat, or write results in CHECK CONDITION status with associated sense data of 7/2700 (Data Protect, Write Protected).

11 Capacity Scaling Control

Bits	**Description**
7-2	Vendor-Reserved
1	SegScalV (Performance Segment Scaling Valid): 0b (non-changeable)

Value	**Description**
0b	Indicates that the device should not segment the medium in accordance with the value in the Segment Scaling field (always on Mode Sense).
1b	Indicates that the device should segment the medium in accordance with the value in the Performance Segment Scaling field.

NOTE 104 - Explicit Performance Segment Scaling of a volume is not currently supported, but may be in the future.

0 CapScalV (Capacity Scaling Valid): 0b (changeable-special)

Value	**Description**
0b	Indicates that the device should not scale the medium in accordance with the value in the Capacity Scaling field (always on Mode Sense).
1b	Indicates that the device should scale the medium in accordance with the value in the Capacity Scaling field.

Capacity or Performance Segment Scaling is accepted only at Beginning of Partition 0 (BOP 0). A valid Scaling request causes all data on the entire physical volume to be lost. If this command is received at other than BOP 0, the command is presented CHECK CONDITION status with associated sense data of 5/2600 (Illegal Request, Invalid Field in Parameter List).

Capacity or Performance Scaling of a volume is not supported for ECONOMY or WORM medium types. See the Support Flags field in this mode page for information on medium which supports scaling. If an attempt is made to perform scaling on medium types which disallow scaling, the command is presented CHECK CONDITION status with associated sense data of 5/2600 (Illegal Request, Invalid Field in Parameter List).

NOTE 105 - On devices that support partitioning, if the volume is scaled, the values in MP 11h: Medium Partition Page (see 6.6.13) are invalidated (e.g., the FDP, SDP, and IDP bits are set to 0b).

12 Capacity Scaling: 00h (from media) (changeable)

This field allows an initiator to logically change the size of partition 0. One effect is faster access to data at the expense of data capacity. This byte indicates or sets the currently formatted medium capacity in relationship to the maximum medium capacity. Maximum medium capacity in this context refers to the amount of data that can be potentially written on the medium, independent of the amount that is currently written or available. The capacity is reduced to a value of n/256*100 percent of this maximum capacity where n is the value in the Capacity Scaling field and ranges between 1 and 256 (01h - FFh, 00h). (The value 00 represents 256, or 100% of capacity.) For example, a Capacity Scaling value of 128 (80h) reduces the capacity of a single partition volume to 50% of its maximum value, but also reduces the average access time to any given data.

This medium is changed as indicated by this field only if the CapScalV bit is set to 1b.

Only certain values are supported. All other values are rounded up to the next supported value. If a value is rounded, the device responds to the Mode Select command with CHECK CONDITION status and associated sense data of 1/3700 (Recovered Error, Rounded Parameter). A Mode Sense command may be used to determine the actual value used. It is recommended that only non-rounding values be used as additional values may be supported in the future and may produce differing results.

Table 286 — Supported Capacity Scaling Values

Requested Capacity Scaling (byte 12)[c d]	Actual Capacity Scaling (byte 12)[b]	Implicit Performance Segment Scaling (byte 19)[a d]	FastSync (i.e., Virtual Backhitch) Supported to LEOP
F1h-FFh,00h	00h	00h	N
EC-F0	F0h	00h	Y
E1h-EBh	E1h-EBh	Y[d]	N
4Bh-E0h	4Bh-E0h	Y[d]	Y
1Dh-4Ah	1Dh-4Ah	00h	Y
16h-1Ch	16h-1Ch	00h	Y
01h-16h	16h	00h	Y

Note - GB is 10^9

[a] Performance Segment Scaling is automatically selected for Capacity Scaled medium at certain scaling values. When this value is non-zero the Performance Segment size is implicitly set to a fixed size of about the values:

Format:	J5AD	J5D	J5C	J4C		
Value:	2 870 GB	1 909GB	1 616GB	720GB		
Format:	J4B	J3B	J3A	J2B	J2A	J1A
Value:	227GB	155GB	135GB	100GB	100GB	60GB.

[b] Medium Capacity is calculated as a fraction of nominal maximum capacity. Scaled medium capacity is approximately equal to the nominal unscaled medium capacity times this value divided by 256. The nominal capacity for 300 GB medium is 292, 968,750 (300GB/1024) and 500 GB medium is 488, 281, 250 (500GB/1024).

[c] Recommended values are 00h (unscaled), F0h (FastSync optimized), EBh (performance segmented), E0h (performance segmented and FastSync optimized), and 35h to achieve:

Format:	J5AD	J5D	J5C	J4C		
Value:	3 100 GB	2 070GB	1 450GB	820GB		
Format:	J4B	J3B	J3A	J2B	J2A	J1A
Value:	330GB	200GB	130GB	140GB	100GB	60GB.

[d] Implicit Segmentation is not used by the 60F format or JE, JV, JM media types.

This command is accepted only at Beginning of Partition 0 (BOP 0). A valid Capacity Scaling command causes all data on the entire physical volume to be lost and only partition 0 to remain at some size. If this command is received at other than BOP 0, the command is presented CHECK CONDITION status with associated sense data of 5/2600 (Illegal Request, Invalid Field in Parameter List).

NOTE 106 - On devices that support partitioning, if the volume is scaled, the values in MP 11h: Medium Partition Page (see 6.6.13) are invalidated (e.g., the FDP, SDP, and IDP bits are set to 0b).

13-16 Medium Capacity (non-changeable)

This field specifies the nominal capacity, in kibibytes (2^{10}), of the program accessible portion of the medium indicated by the Medium Identifier in bytes 4-5. If capacity scaling is in effect, this field reflects the capacity after scaling. If the medium is currently partitioned, the nominal capacity of the medium is the sum of the nominal capacity of all partitions. The value in the field multiplied by 1024 yields the approximate capacity in bytes. The field is set to 0 if the capacity is unknown. *This field is not changeable.* It is for reporting only.

If the value in this field is FFFF_FFFFh, then the nominal capacity of the medium exceeds FFFF_FFFEh kibibytes (approximately 4.39 TB) and this field should not be used to determine the medium capacity.

17 Obsolete (never used)

18 Vendor-Reserved

19 Performance Segment Scaling: 00h (from media) (changeable)

NOTE 107 - Explicit Performance Segment Scaling of a volume is not currently supported, but may be in the future.

NOTE 108 - 3592A6 density or later and JE, JV, or JM media types or later do not support Performance Segment Scaling. Capacity Scaling is performed instead of Performance Segment Scaling.

This field allows an initiator to logically change the size of the Performance Segment. This may affect the data capacity of the medium, and does affect the physical layout of data on medium. The logical layout is not affected. The primary effect is faster access to data located logically near BOT. A secondary effect is a possible improvement in pseudo-random access to data located within the same segment. This byte indicates or sets the currently formatted performance segment capacity in relationship to the currently scaled medium capacity. Currently scaled medium capacity in this context refers to the amount of data that can be potentially written on the currently capacity scaled medium (or the entire medium if not scaled), independent of the amount that is currently written or available. The performance segment capacity is set to a value of n/256*100 percent of this maximum capacity where n is the value in the Performance Segment Scaling field and ranges between 1 and 256 (01h - FFh, 00h). (The value 00h represents 256, or 100% of capacity.) For example, a Segment Scaling value of 128 (80h) set the performance segment capacity to 50% of the writeable medium capacity, but also reduces the average access time to any given data written in that area of media.

This medium is changed as indicated by this field only if the SegScalV bit is set to 1b.

Only certain values are supported. All other values are rounded up to the next supported value. If a value is rounded, the device responds to the Mode Select command with CHECK CONDITION status and associated sense data of 1/3700 (Recovered Error, Rounded Parameter). A Mode Sense command may be used to determine the actual value used. It is recommended that only non-rounding values be used as additional values may be supported in the future and may produce differing results.

This command is accepted only at Beginning of Partition 0 (BOP 0). A valid Performance Segment Scaling command causes all data on the entire physical volume to be lost and only partition 0 to remain at some size. If this command is received at other than BOP 0, the command is presented CHECK CONDITION status with associated sense data of 5/2600 (Illegal Request, Invalid Field in Parameter List).

NOTE 109 - On devices that support partitioning, if the volume is scaled, the values in MP 11h: Medium Partition Page (see 6.6.13) are invalidated (e.g., the FDP, SDP, and IDP bits are set to 0b).

20 Support Flags

Bit	Description
7	Medium Supports Partitioning: 0b (non-changeable)

Value	Description
0b	The medium does not support partitioning.
1b	The medium supports partitioning (see 4.19.4).

Bit	Description
6	Medium Supports Performance Segment Scaling: 0b (non-changeable)
5	Medium Supports Capacity Scaling: (non-changeable)

Value	Description
0b	The medium does not support Capacity Scaling.
1b	The medium supports Capacity Scaling.

NOTE 110 - The only medium type which currently supports scaling is DATA. The ECONOMY, ECONOMY WORM, and WORM types do not support scaling.

Bit	Description
4	Device supports WORM

Value	Description
0b	Medium is not WORM
1b	Medium is WORM

NOTE 111 - For WORM usage implications, see“WORM Behavior” on page 53

3 Medium Supports Encryption (changeable-ignored)

Value	**Description**
0b	Medium does not support encryption
1b	Medium supports encryption (device must also support encryption for this to be set)

2-1 Vendor-Reserved

0 Vendor-Reserved (Edge Track): 0b

21 Internal-VOLID Flags (non-changeable)

Bits **Description**

7 Internal-VOLID Valid

Value	**Description**
0b	Indicates that the fields are not valid.
1b	Indicates that the internal volume-identifier field and the remainder of the internal-VOLID flags field are valid.

6 Tape Directory Valid

Value	**Description**
0b	Indicates that the tape directory is not valid.
1b	Indicates that the tape directory is valid.

5 Vendor-Reserved

4 Internal Volid is EBCDIC

Value	**Description**
0b	Volume label is ASCII
1b	Volume label is EBCDIC

3 Vendor-Reserved

2-0 Internal-VOLID Information Source

Value	**Description**
000b	Internal-VOLID from external source, that is, a library manager.
001b	Internal VOLID from medium, namely the VOL1 record.
010b-111b	Vendor-Reserved

22-27 Internal-Volume Identifier (non-changeable)

If available, this field contains the volume identifier, which is recorded on the medium. The field is left-justified and padded with ASCII or EBCDIC blanks. If the Volume Identifier is not valid, it is set to 0000 0000 0000h.

28-31 Partition Capacity (non-changeable)

This field specifies the nominal capacity, in kibibytes (2^{10}), of the program accessible portion of the currently active partition. The value in the field multiplied by 1024 yields the number of bytes in the partition. The field is set to 0 if the capacity is unknown. This field is not changeable but does reflect changes by other activity. It is for reporting only.

If the value in this field is FFFF_FFFFh, then this partitions capacity exceeds FFFF_FFFEh kibibytes (approximately 4.39 TB) and this field should not be used to determine the partitions capacity.

32-35 Kibibytes Traversed (non-changeable)

This field specifies the current position on the tape measured in kibibytes (2^{10}) traversed (user data only). The value at the logical end of tape should equal the nominal capacity of the tape. All values are rounded down to the nearest kibibyte.

If the value in this field is FFFF_FFFFh, then the current position on tape exceeds FFFF_FFFEh kibibytes (approximately 4.39 TB).

36-47 Vendor-Reserved

48–59 World Wide Unique Cartridge Identifier (non-changeable)

The World Wide Unique identifier used by the WORM cartridge in hexidecimal numbers. If the cartridge is not a WORM cartridge this value may be all zeros.

NOTE 112 - This field should not be used to determine whether the current medium is a WORM cartridge.

NOTE 113 - Byte 8 (byte 56 of this page), bits 7 and 6 are used to indicate special test cartridges. For production WORM media bit 7 is set to b'0' and bit 6 is set to b'1'. To insure the most comprehensive WORM integrity protection, applications should check these bits as part of their medium management strategy for use with production data.

6.6.20 MP 24h: Initiator-Specific Extensions

See MODE SELECT (6/10) - 15h/55h (see 5.2.14) for how to set these parameters and MODE SENSE (6/10) - 1Ah/5Ah (see 5.2.15) for how to read these parameters.

The Initiator-Specific Extensions page provides controls that allow selective use of CRC protection and permit program control of key synchronous data transfer parameters.

Mode Page Behaviors (see 4.5 on page 27) describes this device's non-standard behaviors related to mode parameters.

The mode page policy for this page is described in Policy — non-standard (see 4.5.1 on page 27), except for fields common to all initiators as noted below.

Byte **Description**

0

Bit	Description
7	PS (Page Save): 0b. Ignored in MODE SELECT. See Parameter Saveable behavior in 4.5.2.2 on page 29.
6	Reserved
5-0	Page Code: 24h

1 Page Length: 16h

2 CRC Target Support (checking across host interface) (non-changeable)

This field cannot be changed by the initiator.

Bits	Description
7	CRC32-IEEE Support: 0b

CRC32-IEEE (see E.3. on page 548) describes this CRC Algorithm.

Value	Description
0b	CRC32-IEEE is not supported
1b	CRC32-IEEE is supported

6 Reed-Solomon CRC Support: 1b

Reed-Solomon CRC (see E.1. on page 543) describes this CRC Algorithm

Value	Description
0b	Reed-Solomon CRC is not supported
1b	Reed-Solomon CRC is supported

5 CRC32C Support (changeable-ignored)

CRC32C (Castagnoli) (see E.2. on page 545) describes this CRC Algorithm

Value	Description
0b	CRC32C CRC is not supported (e.g., J1A through E07)
1b	CRC32C CRC is supported (e.g., E08+ only)

4-0 Vendor-Reserved

3 CRC Target Enablement (across host interface) (changeable)

This field can be changed by the initiator.

Value	Description
00h	CRC checking disabled (default setting)
01h	CRC32-IEEE checking enabled (unsupported)
02h	Reed-Solomon CRC checking enabled
03h	CRC32C checking enabled

4 CRC (1) Placement and Length (non-changeable)

Bits	**Description**
7-6	CRC Placement

Value	**Description**
00b	CRC appended to data
01b	CRC prefixed to data

5-0	CRC Length (0-63 bytes): 04h

5 CRC (1) Scope

Bits	**Description**
7	Read data checked: 1b
6	Write data checked: 1b
5	Parameter read data checked: 0b
4	Parameter write data checked: 0b
3	CDB checked: 0b
2	RBD data checked: 1b
1-0	Vendor-Reserved

6 Other CRC (1) Characteristics

Bits	**Description**
7	CDB Transfer Length: 1b

Value	**Description**
0b	CDB Transfer Length does not include CRC.
1b	CDB Transfer length includes CRC.

6	CRC Endian: 0b

Value	**Description**
0b	CRC is Little Endian.
1b	CRC is Big Endian.

5	Read CRC Reporting: 1b

Value	**Description**
0b	Check One on Read CRC Miscompare
1b	Check Condition on Read CRC Miscompare

4	Write CRC Reporting: 1b

Value	**Description**
0b	Check One on Write CRC Miscompare
1b	Check Condition on Write CRC Miscompare

3	Write CRC Check Condition: 0b

Value	**Description**
0b	Deferred Check Condition on Write CRC Miscompare
1b	Immediate Check Condition on Write CRC Miscompare

2-0	Vendor-Reserved

- 7 Support Flags
  - **Bit** **Description**
  - 7 Device Supports Partitioning: 0b
  - 6 Device Supports Performance Segment Scaling: 0b
  - 5 Device Supports Capacity Scaling: 1b
  - 4 Device Supports WORM
    - **Value** **Description**
    - 0b WORM medium is not supported, and will not fully load and come ready.
    - 1b WORM medium is supported.
  - 3 Device Supports Encryption - Enabled (Changeable-Ignored)
    - **Value** **Description**
    - 0b Device does not support encryption
    - 1b Device supports encryption (encryption interface(s) are enabled)
  - 2-1 Vendor-Reserved
  - 0 Device Supports Encryption- Capable (Changeable-Ignored)
    - **Value** **Description**
    - 0b Device does not support encryption (does not have encryption hardware)
    - 1b Device supports encryption (encryption interface(s) are not necessarily enabled)
- 8-12 Vendor-Reserved
- 13 Vendor-Reserved (Transfer Period): FFh
- 14 Vendor-Reserved (REQ/ACK Offset): FFh
- 15 Buffer Association Enablement (DDR Support)

  NOTE 114 - This is persistent across host resets and is common to all initiators.

  - **Bits** **Description**
  - 7 Manual unload Association: 0b
    - **Value** **Description**
    - 0b Association Disabled (RBD must be performed prior to unload)
    - 1b Association Enabled (RBD may be performed before or after unload)

    NOTE 115 - The associated date will be lost if a Rewind or Load command is executed.

  - 6 Manual Rewind Association: 0b
    - **Value** **Description**
    - 0b Association Disabled
    - 1b Association Enabled
  - 5 Unload with Write error Association: 0b (changeable)
    - **Value** **Description**
    - 0b Association Disabled (RBD must be performed prior to unload)
    - 1b Association Enabled (RBD may be performed before or after unload)

    NOTE 116 - The associated data will be lost if a Rewind or Load command is executed.

  - 4-0 Vendor-Reserved
- 16-23 Vendor-Reserved

6.6.21 MP 25h: Read/Write Control

See MODE SELECT (6/10) - 15h/55h (see 5.2.14) for how to set these parameters and MODE SENSE (6/10) - 1Ah/5Ah (see 5.2.15) for how to read these parameters.

Mode Page Behaviors (see 4.5 on page 27) describes this device's non-standard behaviors related to mode parameters.

The mode page policy for this page is described in Policy — non-standard (see 4.5.1 on page 27)

Byte	**Description**		
0			
	Bit	**Description**	
	7	PS (Page Save): 0b. Ignored in MODE SELECT. See Parameter Saveable behavior in 4.5.2.2 on page 29.	
	6	Reserved	
	5-0	Page Code: 25h	
1	Page Length: 1Eh		
2	Ignore Sequence Checks		
	Bit	**Description**	
	7-3	Vendor-Reserved	
	2	Locate command: 0b (changeable)	
		Value	**Description**
		0b	Do not ignore sequence checks
		1b	Ignore sequence checks
	1	Space command for blocks: 0b (changeable)	
		Value	**Description**
		0b	Do not ignore sequence checks
		1b	Ignore sequence checks
	0	Space command for filemarks: 0 (changeable)	
		Value	**Description**
		0b	Do not ignore sequence checks
		1b	Ignore sequence checks
3	Ignore Data Checks		
	Bit	**Description**	
	7-3	Vendor-Reserved	
	2	Locate command: 1b (changeable)	
		Value	**Description**
		0b	Do not ignore data checks
		1b	Ignore data checks
	1	Space command for blocks: 1b (changeable)	
		Value	**Description**
		0b	Do not ignore data checks
		1b	Ignore data checks
	0	Space command for filemarks: 1b (changeable)	
		Value	**Description**
		0b	Do not ignore data checks
		1b	Ignore data checks
4	Vendor-Reserved		
5	Logical End of Partition Method: 00h (changeable)		
	Value	**Description**	
	00h	LEOP is determined by density code where: • density code 51h or 71h uses maximum medium capacity • density codes other than 51h or 71h use constant medium capacity	
	01h	LEOP is determined to maximize medium capacity	
	02h	LEOP is determined to provide constant medium capacity	

6-7 Logical End of Partition: 0000 — Early Warning (LEOP-EW)

This field provides a vehicle for the initiator to enable an early warning indication of the approach of the Logical End of Partition (LEOP). This warning may be used by the initiator to ensure it has sufficient remaining space on the current tape partition to commit all of its internal write buffers. The early warning is provided in the form of a deferred CHECK CONDITION status with associated sense data of 6/0002 (Unit Attention, End-of-Partition/Medium Detected). This CHECK CONDITION status is returned when the first device block is committed to the medium which comes within the specified number of megabytes (10^6) of the LEOP. The normal initiator response may be to stop further writes and flush all initiator buffered data to the drive.

This field is specified in megabytes (10^6). The maximum allowed value is 1000h. A value of 0000h results in no warning being given. Any other value specifies the number of megabytes prior to the LEOP that the warning will occur. Non-zero values less than 008Ch are rounded to 008Ch and report 1/3700 Rounded Parameter. The device makes worst case compression assumption.

8

Bit	**Description**
7	Disable FastSync (i.e., Virtual Backhitch): 0b (changeable)

Value	**Description**
0b	FastSync operation is enabled.
1b	FastSync operation is disabled.

6	Disable SkipSync: 0b (changeable)

0b	SkipSync operation is enabled.
1b	SkipSync operation is disabled.

5-4	Vendor-Reserved
3	Disable Crossing EOD: 0b (changeable)

Value	**Description**
0b	Crossing EOD is enabled. (See “General Read-Type Handling” on page 40)
1b	Crossing EOD is disabled.

2	Disable Crossing Permanent Errors: 0b (changeable)

Value	**Description**
0b	Crossing permanent errors is enabled. (See “General Read-Type Handling” on page 40)
1b	Crossing permanent errors is disabled.

1	Report Segment Early Warning: 0b
0	Report Housekeeping Errors: 0b (changeable)

Value	**Description**
0b	If the Mode Page 01h PER bit is set to 0b, do not report Housekeeping errors.
1b	Report CHECK CONDITION status and deferred sense data of 1/0000 (Recovered Error, No Additional Sense Information) for permanent read/write housekeeping errors including Tape Directory invalid.

9 Default Write Density at BOP 0: X’00’ ((changeable) [not J1A]) (from VPD)

NOTE 117 - Note: If medium that does not support the specifically selected density is loaded, the nearest supported density will be used and reflected in the Pending Active Write Density at BOP field.

NOTE 118 - If an attempt is made to set the value to an value unknown by the device including a value for a later generation product (e.g., the device is an E05 drive and the attempted value is for an E06) the Mode Select will be rejected

Value	**Description**
00h	Use default density code (highest supported) (on unload)
51h	Use 3592A1 density (only JA/JW; JJ/JR medium) (on unload)
52h	Use 3592A2 density (only JA/JW; JJ/JR; JB/JX medium) (on unload)
53h	Use 3592A3 density (only JA/JW; JJ/JR; JB/JX medium) (on unload)
54h	Use 3592A4 density (only JB/JX; JC/JY/JK medium) (on unload)
55h	Use 3592A5 density (only JC/JY/JK; JD/JZ/JL medium) (on unload)
56h	Use 3592B5 density (only JD/JZ/JL medium) (on unload)
57h	Use 3592A6 density (only JE/JV/JM medium)
7Fh	No change to density (including pending density) (on load, use default for uninitialized medium)
FFh	Use currently recorded density (on load use default for uninitialized medium)

10 PENDING WRITE DENSITY AT BOP 0 ((changeable) [not J1A])

The medium is reformatted to this density on the next qualified write operation (write, write filemarks, format, etc.) at logical block 0. This field may be indirectly altered when the Default Write Density field above is changed, when the Mode Select Block Descriptor Density Code field is changed, or according to the Default Write Density field value when medium is unloaded or loaded.

NOTE 119 - This field will be read as explicit density values only.

Value	**Description**
00h	Use default density code (highest supported) (mode select only)
51h	Use 3592A1 density (only JA/JW; JJ/JR medium)
52h	Use 3592A2 density (only JA/JW; JJ/JR; JB/JX medium)
53h	Use 3592A3 density (only JA/JW; JJ/JR; JB/JX medium)
54h	Use 3592A4 density (only JB/JX; JC/JY/JK medium)
55h	Use 3592A5 density (only JC/JY/JK; JD/JZ/JL medium)
56h	Use 3592B5 density (only JD/JZ/JL medium)
57h	Use 3592A6 density (only JE/JV/JM medium)
7Fh	No change to pending density (this value is a no-op) (mode select only)
FFh	Use currently recorded density (on load, use default for uninitialized medium; mode select only)

NOTE 120 - It is illegal for PENDING WRITE DENSITY AT BOP 0 to change from the value reported in Mode Sense if the Density Code field in the Block Descriptor is changed to a value different than either the value returned in Mode Sense, the value X'7F' or the changed value specified for PENDING WRITE DENSITY AT BOP 0. For such an illegal attempt, the device responds with CHECK CONDITION status and associated sense data of 5/2600 (Illegal Request, Invalid Field in Parameter List).

NOTE 121 - On devices which support encryption, secondary encryption density codes may be selected but are interchangeable and identical to primary density codes and are remapped by the device. Use of any density codes has no effect on the encrypted state of medium and cannot be used to control encryption.

NOTE 122 - On devices that support partitioning, if the value of the PENDING WRITE DENSITY AT BOP 0 changes, the values in MP 11h: Medium Partition Page (see 6.6.13) are invalidated (e.g., the FDP, SDP, and IDP bits are set to 0b).

11-15 Reserved
16-27 Vendor-Reserved
28-31 Reserved

6.6.22 MP 30h: Device Attribute Settings

The device attribute settings page and the related subpages are used to query and configure settings used by the drive to control its behavior and configuration in a specified environment. The page_0 mode page 30h is the directory listing of supported subpages. Each subpage is for the query and/or setting of device attributes for a specific function. The persistence of parameters and statement of which parameters are settable or only readable is stated in the section describing each subpage.

See MODE SELECT (6/10) - 15h/55h (see 5.2.14 on page 129) for how to set these parameters and MODE SENSE (6/10) - 1Ah/5Ah (see 5.2.15 on page 131) for how to read these parameters.

6.6.22.1 MP 30h: Directory Listing - Device Attribute Settings

The directory listing - device attribute settings mode page is used to report the list of supported subpages in the device attribute settings mode page and subpages. The page is valid only for a Mode Sense command.

Mode Page Behaviors (see 4.5 on page 27) describes this device's non-standard behaviors related to mode parameters.

The mode page policy for this page is described in Policy — non-standard (see 4.5.1 on page 27)

Table 287 — MP 30h: Directory Listing - Device Attribute Settings mode page format

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	PS	SPF(0)	PAGE CODE (30h)					
1	PAGE LENGTH (n-1)							
Supported subpage list								
2	Supported subpage [first]							
⋮								
n	Supported subpage [last]							

The supported subpage list is a list of supported subpages listed in ascending order. Since the list of supported subpages is anticipated to differ between code levels (i.e., newer code levels may add subpages) the complete list is not described here. A user should read this page to determine which subpages may be used.

The following parameters apply:

Byte **Description**

0

Bit	**Description**
7	PS (Page Saveable): 0b. Ignored in MODE SELECT. See Parameter Saveable behavior in 4.5.2.2 on page 29
6	SPF (SubPage Format): 0b
5-0	PAGE CODE: 30h

1 PAGE LENGTH:

2-n Supported subpage list: Supported subpage list - Device Attribute Settings (see 6.6.22.2 on page 432) describes the subpages that were implemented or planned at the time this document was published.

6.6.22.2 Supported subpage list - Device Attribute Settings

The supported subpage list returns the list of supported subpages. These subpages are organized in logical groupings by function.

Range	Description
01h-03h	MP 30h[01h-03h]: Ethernet attributes - Device attribute settings (see 6.6.22.3 on page 432).
04h-1Fh	Reserved
20h	MP 30h[20h-(20h)]: Encryption Attributes - Device Attribute Settings (see 6.6.22.4 on page 437).
21h-3Fh	Reserved
40h-43h	MP 30h[40h-43h]: Data processing attributes - Device attribute settings (see 6.6.22.5 on page 439)
44h-FEh	Reserved

Byte	Description
2-n	Supported subpages: This field is a list of 1-byte long subpage codes and may include some or all of the following:

Code	Mode Page 30h Subpage
01h	MP 30h[01h]: Drive MAC address - Device attribute settings (see 6.6.22.3.2 on page 434)
02h	MP 30h[02h]: Drive IP address and subnet mask - Device attribute settings (see 6.6.22.3.3 on page 436)
MP 30h[20h]: Encryption mode - Device Attribute Settings (see 6.6.22.4.1 on page 437)	
40h	MP 30h[40h]: SkipSync - Device attribute settings (see 6.6.22.5.1 on page 439)
42h	MP 30h[42h]: End of partition behavior control - Device attribute settings (see 6.6.22.5.2 on page 442)
43h	MP 30h[43h]: Feature switches - Device attribute settings (see 6.6.22.5.3 on page 442)

6.6.22.3 MP 30h[01h-03h]: Ethernet attributes - Device attribute settings

6.6.22.3.1 Ethernet attributes overview

The Ethernet attributes subpages of the device attribute mode page describe information related to the drives Ethernet interface. The subclauses to this subclause list structures used by the Ethernet attribute subpages that are described in the subclauses that are at a peer level to this subclause.

6.6.22.3.1.1 Ethernet socket address descriptor

The IP address and subnet mask is defined in table 288..

Table 288 — Ethernet socket address descriptor

Byte	Bit							
	7	6	5	4	3	2	1	0
0	MSB	ETHERNET SOCKET ADDRESS DESCRIPTOR LENGTH (22H)						
1								LSB
2	MSB	RELATIVE SOCKET ADDRESS IDENTIFIER						
3								LSB
4	SOCKADDR							
31								
32	SUBNET MASK LENGTH							
33	Reserved							
35								

The following parameters apply:

Byte	**Description**
0-1	ETHERNET SOCKET ADDRESS DESCRIPTOR LENGTH - The number of bytes to follow in the Ethernet socket address descriptor.
2-3	RELATIVE SOCKET ADDRESS IDENTIFIER - A unique identifier for this Ethernet socket address.
4-31	SOCKADDR - defined by type of IP address

Type	**Description**
IPv4	6.6.22.3.1.1.1
IPv6	6.6.22.3.1.1.2

32	SUBNET MASK LENGTH - The number of bits set to one in the subnet mask.
33-35	Reserved

6.6.22.3.1.1.1 Sockaddr for an IPv4 IP address

The sockaddr for an IPv4 IP address is defined in table 289.

Table 289 — Sockaddr format for IPv4

Byte	**Bit**							
	7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	ADDRESS LENGTH							
1	ADDRESS FAMILY							
2	MSB	PORT						
3								LSB
4	MSB	INTERNET ADDRESS						
7								LSB
8		Pad bytes						
27								

The following parameters apply:

Byte	**Description**
0	ADDRESS LENGTH (10h)
1	ADDRESS FAMILY (02h)
2-3	PORT - The TCP port number, if any. Zero if there is no TCP port number.
4-7	INTERNET ADDRESS - The IP Address.
8-27	Pad bytes - All bytes set to zero.

6.6.22.3.1.1.2 Sockaddr for an IPv6 address

The sockaddr for an IPv6 IP address is defined in table 290.

Table 290 — Sockaddr format for IPv6

Byte	Bit							
	7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	ADDRESS LENGTH							
1	ADDRESS FAMILY							
2	MSB	PORT						
3								LSB
4	FLOW INFO							
7								
8	MSB	INTERNET ADDRESS						
23								LSB
24	SCOPE ID							
27								

The following parameters apply:

Byte	Description
0	ADDRESS LENGTH (1Ch)
1	ADDRESS FAMILY (0Ah)
2-3	PORT - The TCP port number, if any. Zero if there is no TCP port number.
4-7	FLOW INFO - (0000 0000h)
8-23	INTERNET ADDRESS - The IP Address.
24-27	SCOPE ID - (0000 0000h)

6.6.22.3.2 MP 30h[01h]: Drive MAC address - Device attribute settings

Mode Page Behaviors (see 4.5 on page 27) describes this device’s non-standard behaviors related to mode parameters.

The mode page policy for this page is described in Policy — non-standard (see 4.5.1 on page 27)

The drive MAC address is read only and is defined in table 291.

Table 291 — Drive MAC address

Byte	Bit							
	7	6	5	4	3	2	1	0
	PS (0B)	SPF(1b)	PAGE CODE (30h)					
	SUBPAGE CODE (01h)							
	(MSB)	PAGE LENGTH (n-3)						
								(LSB)
	Reserved							
	Reserved							
	Reserved							
	NUMBER OF DRIVE PORT MAC ADDRESS DESCRIPTORS							
	Drive port MAC address descriptor [first]							
15								
	⋮							
n-7	Drive port MAC address descriptor [last]							
n								

The following parameters apply:

Byte Description

0

Bit	**Description**
7	PS: 0b. Ignored in MODE SELECT. See Parameter Saveable behavior in 4.5.2.2 on page 29
6	SPF (1b)
5-0	PAGE CODE (30h)

1 SUBPAGE CODE (01h)

2-3 PAGE LENGTH

4-6 Reserved

7 NUMBER OF DRIVE PORT MAC ADDRESS DESCRIPTORS (non-changeable)

The number of descriptors to follow

8-n Drive port MAC address descriptors (non-changeable)

The Drive port MAC address descriptors are listed in ascending order by RELATIVE TARGET PORT IDENTIFIER. The Drive port MAC address descriptor is defined in table 292.

Table 292 — Drive port MAC address descriptor

Byte	Bit							
	7	6	5	4	3	2	1	0
0	MSB	RELATIVE TARGET PORT IDENTIFIER						
1								LSB
2	MAC ADDRESS							
7								

Byte Description

0-1 RELATIVE TARGET PORT IDENTIFIER - The relative port value for the Ethernet port.

2-7 MAC ADDRESS - The binary representation of the MAC address for that port.

6.6.22.3.3 MP 30h[02h]: Drive IP address and subnet mask - Device attribute settings

Mode Page Behaviors (see 4.5 on page 27) describes this device's non-standard behaviors related to mode parameters.

The mode page policy for this page is described in Policy — non-standard (see 4.5.1 on page 27)

Note that MODE SELECT parameter data is not required to match the structure of the parameter data returned by MODE SENSE. There is a fixed portion of the mode page that is required to be the same, but the descriptor list may contain a subset of descriptors. The Changeable Values are reported in the same structure as Current Values in MODE SENSE but are not used to positionally validate values received in MODE SELECT.

The drive IP address and subnet mask subpage is defined in table 293.

The SP bit of a MODE SELECT command shall be set to one for this mode page. If the SP bit is set to zero, then the drive rejects the command with a CHECK CONDITION and sets the sense key to ILLEGAL REQUEST and the additional sense code to INVALID FIELD IN CDB.

Table 293 — Drive IP address and subnet mask subpage

Byte	Bit 7	6	5	4	3	2	1	0
0	PS	SPF(1b)	PAGE CODE (30h)					
1	SUBPAGE CODE (02h)							
2	(MSB)	PAGE LENGTH (n-3)						
3								(LSB)
4	Reserved							
5								
6	Restricted							
7	NUMBER OF DRIVE ETHERNET PORT DESCRIPTORS							
8	Drive Ethernet port descriptor [first]							
m								
	⋮							
n-k	Drive Ethernet port descriptor [last]							
n								

The following parameters apply:

Byte	Description
0	**Bit** **Description** 7 PS - 1b. Ignored in MODE SELECT. See Parameter Saveable behavior in 4.5.2.2 on page 29. 6 SPF (1b) 5-0 PAGE CODE (30h)
1	SUBPAGE CODE (02h)
2-3	PAGE LENGTH
4-5	Reserved
6	Restricted
7	NUMBER OF DRIVE PORT DESCRIPTORS (changeable) - May be set to any number between one and the number of drive Ethernet ports inclusive. Changes will only affect the ports for which a drive Ethernet port descriptor is sent.

8-n Drive Ethernet port descriptors.

The drive Ethernet port descriptors shall be sent in ascending order by RELATIVE TARGET PORT IDENTIFIER (see table 294). The drive Ethernet port descriptor is defined in table 294.

Table 294 — Drive Ethernet port descriptor

Byte	Bit 7	6	5	4	3	2	1	0
0	MSB	DRIVE ETHERNET PORT DESCRIPTOR LENGTH (N-1)						
1								LSB
2	MSB	RELATIVE TARGET PORT IDENTIFIER						
3								LSB
4	Reserved							
5								
6	Reserved							DHCP_V4
7	NUMBER OF ETHERNET SOCKET ADDRESS DESCRIPTORS							
8	Ethernet socket address descriptor [first]							
x								
	⋮							
y	Ethernet socket address descriptor [last]							
n								

Byte	Description
0-1	DRIVE ETHERNET PORT DESCRIPTOR LENGTH - The number of bytes to follow in the drive Ethernet port descriptor.
2-3	RELATIVE TARGET PORT IDENTIFIER (changeable) - The relative port identifier of the Ethernet port.
4-5	Reserved
6	

Bit	Description
7-1	Reserved
0	DHCP_V4 (CHANGEABLE-SAVEABLE)

Value	Description
1b	DHCP shall be used to obtain an additional IPv4 address.
0b	DHCP shall not be used to obtain an additional IPv4 address.

7 NUMBER OF ETHERNET SOCKET ADDRESS DESCRIPTORS (changeable) - Shall be set to 01h or 02h.

8-n Ethernet socket address descriptors (CHANGEABLE-SAVEABLE). The Ethernet socket address descriptor is defined in 6.6.22.3.1.1 on page 432.

In each Ethernet socket address descriptor the PORT field of the SOCKADDR (see table 289 and table 290) is reserved and shall be set to zero.

In each Ethernet socket address descriptor the FLOW INFO field of the SOCKADDR (see table 290), if any, is reserved and shall be set to zero.

In each Ethernet socket address descriptor the SCOPE ID field of the SOCKADDR (see table 290), if any, is reserved and shall be set to zero.

6.6.22.4 MP 30h[20h-(20h)]: Encryption Attributes - Device Attribute Settings

6.6.22.4.1 MP 30h[20h]: Encryption mode - Device Attribute Settings

Mode Page Behaviors (see 4.5 on page 27) describes this device's non-standard behaviors related to mode parameters.

The mode page policy for this page is described in Policy — non-standard (see 4.5.1 on page 27)

The Encryption mode - Device attribute settings mode page allows an application client to query the encryption settings in the drive. If the drive receives an Encryption mode - Device Attribute Settings mode page in a MODE SELECT command it is rejected with an ILLEGAL REQUEST, INVALID FIELD IN PARAMETER LIST.

The Encryption mode - Device Attribute Settings mode page may be used in a MODE SENSE command to query the current Encryption settings. Some of the settings likely to be reported are listed in table 295.

Table 295 — Expected Encryption settings

Encryption setting	Method (byte 5)	Key Path (byte 6)	Default Encryption State (byte 7)	Density Reporting (byte 8)
No Encryption	01h	01h	01h	00h
AME (Application Managed Encryption)	03h	01h	01h	00h
LME (Library Managed Encryption) - Barcode policy	04h	01h	03h	00h
LME - Internal label (selects) policy	04h	01h	06h	00h
LME - Internal label (all) policy	04h	01h	07h	00h
LME - Encrypt Always policy	04h	01h	02h	00h
Custom	combinations not listed above			

The Encryption mode - Device attribute settings mode page is defined in table 296.

Table 296 — Encryption mode mode page

Byte	Bit							
	7	6	5	4	3	2	1	0
0	PS	SPF(1b)	PAGE CODE (30h)					
1	SUBPAGE CODE (20h)							
2	(MSB)	PAGE LENGTH (0005h)						
3								(LSB)
4	Reserved							
5	METHOD							
6	KEY PATH							
7	DEFAULT ENCRYPTION STATE							
8	DENSITY REPORTING							

Byte Description

0

Bit Description

7 PS (parameters saveable): 0b. Ignored in MODE SELECT. See Parameter Saveable behavior in 4.5.2.2 on page 29. Note that the parameters do change in response to configuration changes made via other paths (e.g., through the library).

6 SPF (subpage format): 1b

5-0 PAGE CODE: 30h

1 SUBPAGE CODE: 20h

2-3 PAGE LENGTH: 0005h

4 Reserved

5 METHOD: Encryption solution method. (non-changeable)

6 KEY PATH: (non-changeable) The path used for key management communications

7 DEFAULT ENCRYPTION STATE: (non-changeable)
8 DENSITY REPORTING: (non-changeable)

6.6.22.5 MP 30h[40h-43h]: Data processing attributes - Device attribute settings

6.6.22.5.1 MP 30h[40h]: SkipSync - Device attribute settings

See MODE SELECT (6/10) - 15h/55h (see 5.2.14) for how to set these parameters and MODE SENSE (6/10) - 1Ah/5Ah (see 5.2.15) for how to read these parameters.

Mode Page Behaviors (see 4.5 on page 27) describes this device’s non-standard behaviors related to mode parameters.

The mode page policy for this page is described in Policy — non-standard (see 4.5.1 on page 27)

Note that mounting a volume that modifies the value of fields in this page does not establish a unit attention condition.

The SkipSync mode page is defined in table 297. SkipSync is a feature that performs a trade-off between how much data can be written to a volume and how fast that data can be written when the data is a small file size. This mode page allows the feature to be enabled and disabled. It also provides information that describes certain aspects of this trade-off, such as the Target Minimum Capacity. This feature is only available for use on volumes that are not scaled and that are not partitioned.

Table 297 — SkipSync - Device attribute settings mode page format

Bit Byte	7	6	5	4	3	2	1	0
0	PS	SPF(1b)	PAGE CODE (30h)					
1	SUBPAGE CODE (40h)							
2	(MSB)	Page Length 10h						
3								(LSB)
4	Reserved					SV		ENABLE
5	Reserved							
6	Reserved							
7	SkipSync Policy							
8	(MSB)	Target Minimum Capacity						
11								(LSB)
12	(MSB)	Vendor-restricted						
15								(LSB)
16	(MSB)	Vendor-restricted						
19								(LSB)

Byte Description

0

Bit	**Description**
7	PS (Parameter Saveable): 1b. Ignored in MODE SELECT. See Parameter Saveable behavior in 4.5.2.2 on page 29. It is recommended that users set the SP bit of the MODE SELECT command to one.
6	SPF (SubPage Format): 1b
5-0	PAGE CODE: 30h

1 SUBPAGE CODE: 40h

2-3 PAGE LENGTH: 10h

4

Bit	**Description**
7-3	Reserved
2-1	SV (SkipSync Validity): (changeable-ignored) Note that mounting a volume that modifies the value of this field does not establish a unit attention condition.

Value	**Description**
00b	There is no volume mounted, or SkipSync is not valid for mounted volume.
01b	SkipSync valid for mounted volume but SKIPSYNC POLICY field not supported
10b	SkipSync valid for mounted volume and SKIPSYNC POLICY field is supported
11b	Reserved

0 ENABLE: 1b (changeable-saveable)

Value	**Description**
0b	The SkipSync function is disabled. If this bit is changed from 1b to 0b, then the DISABLE SKIPSYNC bit in MP 25h: Read/Write Control (see 6.6.21) is implicitly set to 1b
1b	The SkipSync function is enabled. The drive operates in a manner to increase performance and reduce backhitches while still performing synchronizations on

small file writes. If this bit is changed from 0b to 1b, then the DISABLE SKIPSYNC bit in MP 25h: Read/Write Control (see 6.6.21) is implicitly set to 0b

5-6 Reserved

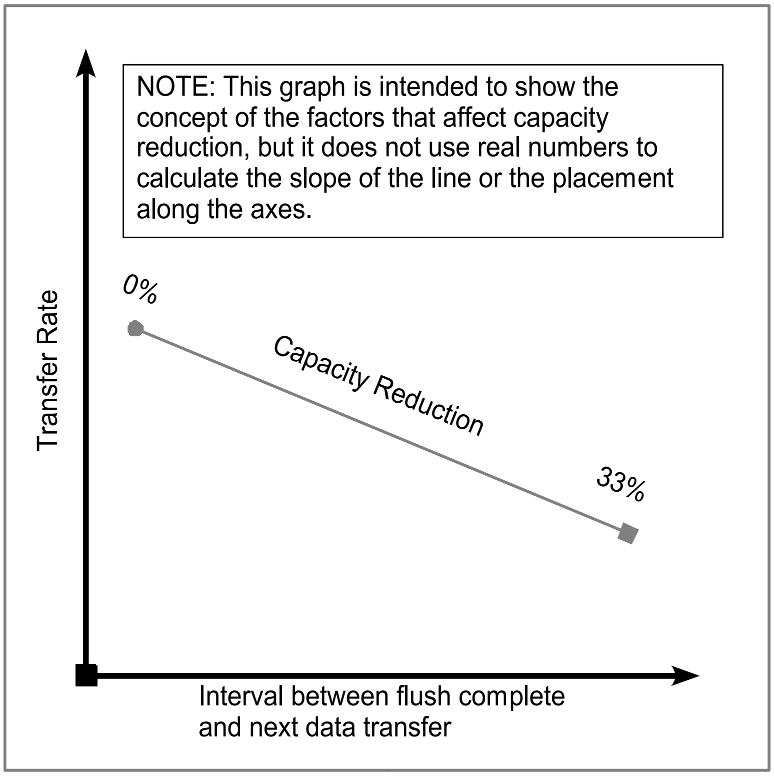
7 SKIPSYNC POLICY: (changeable-saveable)

The SkipSync Policy field indicates the algorithm used in performing the SkipSync operations.

If no volume is mounted or if a volume is mounted and the SV field is set to 10b, then this field is changeable.

If a volume is mounted and the SV field is set to 00b or 01b and an attempt to change the value of this field is made, then the MODE SELECT command is rejected with ILLEGAL REQUEST, INVALID FIELD IN PARAMETER DATA.

Value	**Description**
00h	Drive Default SkipSync Policy (Not returned on MODE SENSE if SV = 10b)
10h	Sync performance with no capacity reduction
20h	Sync performance allowing capacity reduction (a MODE SENSE command may be used to report in other fields the limits that are established by this setting; e.g., TARGET MINIMUM CAPACITY) When this option is selected, the overall capacity may be reduced by up to 33% depending on the transfer characteristics (e.g., transfer rate and the interval between the sync completion and the next data transfer).



8-11 TARGET MINIMUM CAPACITY (MiB): (changeable-ignored)

This field indicates the target capacity in mebibytes (2^{20}) to which the capacity of the current partition when full may be decreased in order to perform SkipSync operations. The total capacity may be smaller than TARGET MINIMUM CAPACITY if the average of transaction sizes is very small. If the mounted volume is not capable of supporting skipsync this field is set to zero. This field reflects the setting of the SKIPSYNC POLICY field.

12-15 Vendor-restricted

16-19 Vendor-restricted

6.6.22.5.2 MP 30h[42h]: End of partition behavior control - Device attribute settings

Mode Page Behaviors (see 4.5 on page 27) describes this device’s non-standard behaviors related to mode parameters.

The mode page policy for this page is described in Policy — non-standard (see 4.5.1 on page 27)

The End of partition behavior control - Device attribute settings mode page is defined in table 291.

Table 298 — End of partition behavior control - Device attribute settings mode page format

Byte	Bit							
	7	6	5	4	3	2	1	0
0	PS	SPF(1b)	PAGE CODE (30h)					
1	SUBPAGE CODE (42h)							
2	(MSB)	PAGE LENGTH (n-3)						
3								(LSB)
4	LEOP METHOD							

The following parameters apply:

Byte	Description
0	Byte zero

Bit	Description
7	PS (parameters saveable): 1b. Ignored in MODE SELECT. See Parameter Saveable behavior in 4.5.2.2 on page 29.
6	SPF (1b)
5-0	PAGE CODE: 30h

Byte	Description
1	SUBPAGE CODE (42h)
2-3	PAGE LENGTH
4	LEOP METHOD - Logical end of partition (LEOP) method: 00h (changeable-saveable)

Value	Description
00h	LEOP is determined by density code

Value	Description
51h	Maximize medium capacity
71h	Maximize medium capacity
others	Constant medium capacity

Value	Description
01h	Maximize medium capacity - LEOP is based on the amount of physical medium available. Early warning is based on the drive's buffer size, the nominal physical dataset length, and some margin (i.e., the drive will accept write request before LEOP if there is tape available).
02h	Constant medium capacity - LEOP is determined to provide constant medium capacity. Early warning is based on the user's logical data capacity already written to the tape (i.e constant capacity). If the medium condition or drive condition is degraded enough to reach a predetermined LPOS tape region, then LEOP may be reached prior to the constant capacity.
others	Reserved

6.6.22.5.3 MP 30h[43h]: Feature switches - Device attribute settings

The Feature switches - Device attribute settings mode page is defined in table 299.

Mode Page Behaviors (see 4.5 on page 27) describes this device’s non-standard behaviors related to mode parameters.

The mode page policy for this page is described in Policy — non-standard (see 4.5.1 on page 27)

Table 299 — Feature switches - Device attribute settings mode page format

Byte	Bit							
	7	6	5	4	3	2	1	0
0	PS	SPF(1b)	PAGE CODE (30h)					
1	SUBPAGE CODE (43h)							
2	(MSB)	PAGE LENGTH (10h)						
3								(LSB)
4	Reserved						E_ARCHIVE	D_BOPC
5	Reserved							
19								

The following parameters apply:

Byte	Description
0	Byte zero

Bit	Description
7	PS - parameters saveable: 1b. Ignored in MODE SELECT. See Parameter Saveable behavior in 4.5.2.2 on page 29.
6	SPF (1b)
5-0	PAGE CODE: 30h

Byte	Description
1	SUBPAGE CODE (43h)
2-3	PAGE LENGTH
4	

Bit	Description
7-2	Reserved
1	E_ARCHIVE (Enable Archive mode unthread): 1b (changeable-saveable) See Archive mode unthread (E08+) (see 4.9)

WARNING WARNING

See Archive mode unthread (E08+) (see 4.9) for warnings.

WARNING WARNING

0 D_BOPC (Disable BOP Cache): 0b (changeable)

This control indicates if the BOP cache information is used during a reposition operation. It does not alter the accumulation of BOP cache information.

Value	Description
0b	Use cached BOP information (if available) to perform positioning requests (see 4.3.2)
1b	Ignore cached BOP information when performing positioning requests (e.g., Perform physical positioning and read data sequentially into buffer from BOP)

5-19 Reserved

6.6.23 MP 37h: String Search (not J1A)

See MODE SELECT (6/10) - 15h/55h (see 5.2.14) for how to set these parameters and MODE SENSE (6/10) - 1Ah/5Ah (see 5.2.15) for how to read these parameters.

This page is used to setup search criteria and query summary status. See 4.22 on page 79 for additional information on string searches.

Mode Page Behaviors (see 4.5 on page 27) describes this device's non-standard behaviors related to mode parameters.

The mode page policy for this page is described in Policy — non-standard (see 4.5.1 on page 27)

This page or related subpage(s) exceeds the maximum page length supported by Mode Sense (6) and Mode Select (6) and cannot be queried or altered with these commands. The Mode Sense (10) and Mode Select (10) must be used with this mode page. This page is not returned in mode page 3Fh (all pages).

Mode Page 37h and Mode Page 37h[01h] should be set under separate Mode Select commands to insure the proper operating setup is used. To do this, the criteria should be setup with Mode Page 37h[01h] before the base page (i.e., Mode Page 37h) enables searching, etc.

6.6.23.1 String Search Control/Status

Byte **Description**

0
- **Bit** **Description**
- 7 PS (Page Save): 0b. Ignored in MODE SELECT. See Parameter Saveable behavior in 4.5.2.2 on page 29.
- 6 Reserved
- 5-0 Page Code: 37h

1 Page Length: 3Eh

2 Search Control
- **Bit** **Description**
- 7 Extended Search Enable: 0b (unsupported)
- 6 Snoop Search Enable:0b (reset when a cartridge is unloaded)

  NOTE 123 - Read and write performance may be affected if Snoop Search Enable is 1b and Match Buffer is also 1b.

  NOTE 124 - Snoop operations may require a flush (Write Filemarks 0) operation to insure the results are correct due to posting latencies.

- 5 Snoop Read: 1b
- 4 Snoop Write: 1b
- 3-0 Vendor-Reserved

3-4 Vendor-Reserved

5 Search Reporting
- **Bit** **Description**
- 7 Vendor-Reserved
- 6 Deferred Sense Key 1: 0b (unsupported)
- 5 Read Sense Key 1: 0b (unsupported)
  - **Value** **Description**
  - 1b Generate a sense with 1/8501 (Recovered Error, Search Snoop Match Found)
- 4 Write Sense Key 1: 0b (unsupported)
- 3-0 Vendor-Reserved

6 Match List Control

Bit **Description**

7 Match Buffer: 0b (changeable)

Value	**Description**
0b	Do not generate match list buffer for snoop/implicit operations
1b	Generate match list buffer for snoop/implicit operations

NOTE 125 - Read and write performance will be affected if Snoop Search Enable is 1b and Match Buffer is also 1b.

6 Match Overflow: 0b (changeable)

Value	**Description**
0b	Match List overwrites oldest entries when list is full
1b	Match List discards new entries when list is full

5-0 Vendor-Reserved

7 Match List Entry Type: 01h (type of match list entry to generate) (changeable)

Value	**Description**
00h	No match list
01h	Use four (4) byte Logical Block Numbers
02h	Use four (4) byte Logical Block Numbers and four (4) byte File Number pairs
03h	Use eight (8) byte Logical Block Numbers
04h	Use eight (8) byte Logical Block Numbers and eight (8) byte File Number pairs

8-9 Search Time: 0000h. The maximum time allowed for a search operation in seconds. If 0000h then there is no search time limit.

NOTE 126 - Device read error recovery actions may cause this time to be exceeded. The device stops the search command when this time has elapsed and only after any record (not necessarily a matching record) has been fully processed.

10 Vendor-Reserved

11 Search Method (Active SubPage): 01h

12 Search Status

Bit **Description**

7 Explicit Results

Value	**Description**
0b	Search Results and Match List reflect implicit (snoop) searches
1b	Search Results and Match List reflect the last explicit search (if Valid Results is 1b)

6-2 Vendor-Reserved

1 Valid Criteria (resettable)

Value	**Description**
0b	Search Criteria has not been setup
1b	Search Criteria has been setup (valid criteria and at least one possible match condition)

0 Valid Results (resettable)

Value	**Description**
0b	Current results are not valid (this will cause the Match Buffer to be cleared)
1b	Current results are valid

13 String Criteria Setup (changeable-ignored)

Bit	**Description**
7	1b: String 1 has valid criteria
6	1b: String 2 has valid criteria
5	1b: String 3 has valid criteria
4	1b: String 4 has valid criteria
3	1b: String 5 has valid criteria
2	1b: String 6 has valid criteria
1	1b: String 7 has valid criteria
0	1b: String 8 has valid criteria

14 Vendor-Reserved (Hardware Match Status Last Record) (changeable-ignored)
15 Vendor-Reserved (Hardware Match Status Last Dataset) (changeable-ignored)
16-23 First Match Logical Block Address (if Valid Results is 1b) (changeable-ignored)
24-31 First Match File Number (if Valid Results is 1b) (changeable-ignored)
32-39 Final Match Logical Block Address (if Valid Results is 1b) (changeable-ignored)
40-47 Final Match File Number (if Valid Results is 1b) (changeable-ignored)
48-55 Match Count (if Valid Results is 1b) (changeable-ignored)

NOTE 127 - It is possible for the Match Count to be larger than the number of Match List Entries

56-63 Check Count (if Valid Results is 1b) (changeable-ignored)

6.6.23.2 MP 37h[01h]: String Search Criteria

This subpage applies to Search Method 01h.

NOTE 128 - This page is specified in the manner that it is returned in Mode Sense (full fields and descriptors are always returned). For Mode Select any of the various Descriptors listed may be omitted, the number of elements in each descriptor may be reduced, or a shorted string length (m) may be used. When this subpage is altered, usage and order sensitive default values will be used for all unspecified/unchanged fields. Given this unique descriptor and default behavior, it may be simpler for the initiator to explicitly generate this subpage and use Mode Select only (without Mode Sense), rather than using the normally recommended standard method of reading the data with Mode Sense, modifying it, and using Mode Select. If the page is explicitly built, care should be taken to insure that the mode headers are not unintentionally altered. Additionally, in order to support a similar simplified method of defaulting descriptors when using the normally recommended standard Mode Sense, alter fields, Mode Select sequence, an “Apply” field is present in each descriptor. The Apply field must be set to 1b when any changes are intended for that descriptor. If changes to a descriptor are detected without the Apply bit set, the device responds with a CHECK CONDITION status and associated sense data of 5/2600 (Illegal Request, Invalid Field in Parameter List).

NOTE 129 - Descriptors are processed in order within the Mode Select. The last descriptor which affects (either by defaulting or explicitly) a given field or set of fields has precedence. The descriptors have been ordered to facilitate the most common behaviors and allow easy explicit overrides with a single Mode Select.

NOTE 130 - Mode Page 37h and Mode Page 37h[01h] should be set under separate Mode Select commands to insure the proper operating setup is used. To do this, the criteria should be setup with Mode Page 37h[01h] before the base page (i.e., Mode Page 37h) enables searching, etc.

NOTE 131 - Model E05 only supports eight (8) strings of sixteen (16) bytes. Strings longer than this may be specified, but the match criteria is only significant up to the first sixteen (16) bytes for each string.

Byte	Description
0	

Bit	Description
7	PS (Page Save): 0b. Ignored in MODE SELECT. See Parameter Saveable behavior in 4.5.2.2 on page 29.
6	SPF (SubPage Format): 1b
5-0	Page Code: 37h

1	SubPage Code: 01h
2-3	Page Length: (sum of descriptors) On Mode Sense: 03B4h

The following descriptors are sent in the order listed.

6.6.23.2.1 Search Criteria 01h - Features Descriptor 00h

This descriptor is normally sent on Mode Selects.

Table 300 — Search Criteria 01h - Features Descriptor 00h

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Descriptor Code: 00h							
	Apply	Vendor-Reserved						
2	MSB	Descriptor Length: 0008h						
3								LSB
4	Span: 0b	Vendor-Reserved						
5	Vendor-Reserved							
11								

- Apply: Always reads as 0b, must be set to 1b if descriptor changes were made or values should be used (see note 128).
- Span: 0b (unsupported)

Value	Description
0b	match does not span record boundaries (all strings must be found in a single logical block)
1b	match may span record boundaries (unsupported)

NOTE 132 - A given spanned match may only be contained in two successive records. If the string is longer than the record length and spans three or more records, it will not be found.

NOTE 133 - Until span support is available, a single sufficiently long string may be split into two strings and ANY (OR) match logic can be used. This may result in more candidate searches, but will allow a search to find simple strings which span records.

6.6.23.2.2 Search Criteria 01h - String Descriptor 10h

This descriptor is normally sent on Mode Selects.

NOTE 134 - Strings specified will generate implicit default search criteria and setup bit masks and byte methods according to the Case, Wild, Any and Not fields corresponding to the values of all specified bytes. These defaults may be overridden by other descriptors with the Apply bit set. Strings are considered valid and will setup default behavior up to the last non-00h character of each string. Strings with one or more byte(s) with the most significant bit (7) set (80h) are considered EBCDIC with respect to Wild and Case. To specify matches of strings with trailing 00h characters the Byte Mask must be explicitly specified. Unspecified strings and partial strings will be non-criteria (byte mask methods of 00b).

Table 301 — Search Criteria 01h - String Descriptor 10h

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Descriptor Code: 10h							
1	Apply	Vendor-Reserved						
2	MSB	Additional Descriptor Length (n-3)						
3								LSB
4	Vendor-Reserved					Case	Wild	Any
5	Not 1	Not 2	Not 3	Not 4	Not 5	Not 6	Not 7	Not 8
6	Vendor-Reserved							
7	String Length: (m; where m <= 20h)							
8	MSB	String (k) (changeable) (where k = 1)						
7+m								LSB
	. . .							
n-m-1	MSB	String (k) (changeable) (where k <= 8)						
n								LSB

- Apply: Always reads as 0b, must be set to 1b if descriptor changes were made or values should be used (see note 114).
- Case: Always reads 0b (changeable)

  This applies to all of the following strings in String (k)

Value	Description
0b	use exact match Byte Mask method
1b	use case insensitive Byte Mask method (for ASCII or EBCDIC characters between A-Z and a-z)

- Wild: Always reads 0b (changeable)

  This applies to all of the following strings in String (k)

Value	Description
0b	do not use wildcard character
1b	use “?” as a wildcard character (match any byte) for ASCII (3Fh) and EBCDIC (6Fh)

- Any: Always reads 0b (changeable)

Value	Description
0b	default Truth Table (see 4.22.1) is built with AND conditions for all specified strings (ALL specified strings must be found)
1b	default Truth Table is built with OR conditions for all specified strings (ANY specified string may be found)

- Not (k): Always reads 0b (changeable)

k = [1,2,3,4,5,6,7,8]

Each Not (k) relates to the corresponding String (k) that follows.

Value	**Description**
0b	default Truth Table is built with specified AND/OR behavior
1b	default Truth Table is built with NOT conditions for the specified string (match cannot contain string) - only valid when Any is set to 0b

6.6.23.2.3 Search Criteria 01h - Bit Mask Descriptor 20h

Table 302 — Search Criteria 01h - Bit Mask Descriptor 20h

Byte	**Bit**							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Descriptor Code: 20h							
1	Apply	Vendor-Reserved						
2	MSB	Additional Descriptor Length (n-3)						
3								LSB
4	Vendor-Reserved							
6								
7	String Bit Mask Length: (m; where m <= 20h)							
8	MSB	String Bit Mask (k) (changeable) (where k = 1)						
7+m								LSB
. . .								
n-m-1	MSB	String Bit Mask (k) (changeable) (where k <= 8)						
n								LSB

NOTE 135 - All Bytes in a String Bit Mask (k) must be set to the same value. Typical values which might be used are 11011111b or 10111111b for case independent matches of ASCII or EBCDIC strings respectively.

- Apply: Always reads as 0b, must be set to 1b if descriptor changes were made or values should be used (see note 114).

6.6.23.2.4 Search Criteria 01h - Byte Mask Descriptor 30h

Table 303 — Search Criteria 01h - Byte Mask Descriptor 30h

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Descriptor Code: 30h							
1	Apply	Vendor-Reserved						
2	MSB	Additional Descriptor Length (n-3)						
3								LSB
4	Vendor-Reserved							
6								
7	Byte Mask Element Length: (m; where m <= 20h)							
8	MSB	Byte Mask Element (k) (changeable) (where k = 1)						
7+m								LSB
. . .								
n-m-1	MSB	Byte Mask Element (k) (changeable) (where k <= 8)						
n								LSB

- Apply: Always reads as 0n, must be set to 1b if descriptor changes were made or values should be used (see note 114).

6.6.23.2.4.1 Search Criteria 01h - Descriptor 30h - Byte Mask Element

Table 304 — Search Criteria 01h - Descriptor 30h - Byte Mask Element

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Byte Method		Vendor-Reserved					
. . .								
m	Byte Method		Vendor-Reserved					

- Byte Method:

Value	**Description**
00b	No valid byte in this location (this terminates the string allowing strings of less than 16 bytes)
01b	A byte must exist in this location, but its value is not checked (this may also be accomplished with Byte Method 10b and a String Bit Mask of 00h
10b	Match byte, but only check String bits which have corresponding bits in the String Bit Mask field set to 1b.
11b	Match byte exactly, bit for bit (do not use String Bit Mask field).

NOTE 136 - If a string is not to be searched for, set 00b in the mask for byte 0. A string will be searched starting at byte 0 and continuing until the first byte mask that is 00b or the end of the string.

6.6.23.2.5 Search Criteria 01h - Minimum Record Offset Descriptor 40h

Table 305 — Search Criteria 01h - Minimum Record Offset Descriptor 40h

<table>
<tr><th rowspan="2">Byte</th><th colspan="8">Bit</th></tr>
<tr><th>7 msb</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0 lsb</th></tr>
<tr><td>0</td><td colspan="8">Descriptor Code: 40h</td></tr>
<tr><td>1</td><td>Apply</td><td colspan="7">Vendor-Reserved</td></tr>
<tr><td>2</td><td>MSB</td><td colspan="6" rowspan="2">Additional Descriptor Length (n-3)</td><td></td></tr>
<tr><td>3</td><td></td><td>LSB</td></tr>
<tr><td>4</td><td colspan="8" rowspan="2">Vendor-Reserved</td></tr>
<tr><td>6</td></tr>
<tr><td>7</td><td colspan="8">String (k) Minimum Byte Offset Length: (04h)</td></tr>
<tr><td>8</td><td>MSB</td><td colspan="6" rowspan="2">String (k) Minimum Byte Offset: (where k = 1)</td><td></td></tr>
<tr><td>11</td><td></td><td>LSB</td></tr>
<tr><td colspan="9">.
.
.</td></tr>
<tr><td>n-3</td><td>MSB</td><td colspan="6" rowspan="2">String (k) Minimum Byte Offset: (where k <= 8)</td><td></td></tr>
<tr><td>n</td><td></td><td>LSB</td></tr>
</table>

- Apply: Always reads as 0b, must be set to 1b if descriptor changes were made or values should be used (see note 114).
- Minimum Byte Offset:

Value	**Description**
00000000h	Beginning of a string

6.6.23.2.6 Search Criteria 01h - Maximum Record Offset Descriptor 50h

Table 306 — Search Criteria 01h - Maximum Record Offset Descriptor 50h

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Descriptor Code: 50h							
1	Apply	Vendor-Reserved						
2	MSB	Additional Descriptor Length (n-3)						
3								LSB
4	Vendor-Reserved							
6								
7	String (k) Maximum Byte Offset Length: (04h)							
8	MSB	String (k) Maximum Byte Offset: FFFF FFFFh (where k = 1)						
11								LSB
. . .								
n-3	MSB	String (k) Maximum Byte Offset: FFFF FFFFh (where k <= 8)						
n								LSB

- Apply: Always reads as 0b, must be set to 1b if descriptor changes were made or values should be used (see note 114).

6.6.23.2.7 Search Criteria 01h - Match Truth Table Descriptor F0h

Table 307 — Search Criteria 01h - Match Truth Table Descriptor F0h

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Descriptor Code: F0h							
1	Apply	Vendor-Reserved						
2	MSB	Additional Descriptor Length: 0024h						
3								LSB
4	Vendor-Reserved							
6								
7	Element Length: 20h							
8	Truth Table (changeable) (see 4.22.1)							
39								

- Apply: Always reads as 0b, must be set to 1b if descriptor changes were made or values should be used (see note 114).

NOTE 137 - If a Truth Table is not specified (descriptor not present or Apply is 0b), then a table will be generated according to the Any field in descriptor 10h and/or by descriptor F1h. See “String Search Function (not J1A)” on page 79 for additional information on the Truth Table.

6.6.23.2.8 Search Criteria 01h - Match String Helper Descriptor F1h

NOTE 138 - This helper descriptor is provided to simplify the generation of the Truth Table (see 4.22). Any number of elements can be used. All bits in each entry specify an AND and AND NOT match criteria. Each entry in the list is an OR match criteria. This table is used for Mode Select only, and it affects other fields and does not retain any values itself (always reads defaults).

Table 308 — Search Criteria 01h - Match String Helper Descriptor F1h

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Descriptor Code: F1h							
1	Apply	Vendor-Reserved						
2	MSB	Additional Descriptor Length: n-3						
3								LSB
4	Vendor-Reserved						Delete	Modify
5	Vendor-Reserved							
6								
7	Element Length: 02h							
AND Match Condition(s) (Element) (2 bytes - repeating) (changeable) (bits as below)								
8	Find 1	Find 2	Find 3	Find 4	Find 5	Find 6	Find 7	Find 8
9	Not 1	Not 2	Not 3	Not 4	Not 5	Not 6	Not 7	Not 8
				. . .				
n-1	Find 1	Find 2	Find 3	Find 4	Find 5	Find 6	Find 7	Find 8
n	Not 1	Not 2	Not 3	Not 4	Not 5	Not 6	Not 7	Not 8

- Apply: Always reads as 0b, must be set to 1b if descriptor changes were made or values should be used (see note 114).
- Modify: Always read as 0b (changeable)

Value	Description
0b	Use AND Match Condition(s) to generate entire Truth Table (clear prior)
1b	Use AND Match Condition(s) to modify existing Truth Table

- Delete: Always read as 0b (changeable)

Value	Description
0b	Add AND Match Condition(s) to Truth Table
1b	Remove AND Match Condition(s) from Truth Table. The Modify field must be set to 1b when Delete is 1b.

- Find (k): Always read as 0b (changeable)

  k = [1,2,3,4,5,6,7,8]

Value	Description
0b	string is a don't care state (if Not is also 0b)
1b	Truth Table is built with AND conditions for the specified string (match must contain string)

- Not (k): Always read as 0b (changeable) k = [1,2,3,4,5,6,7,8]

Value	**Description**
0b	string is a don't care state (if And is also 0b)
1b	Truth Table is built with NOT conditions for the specified string (match cannot contain string)

NOTE 139 - It is illegal for corresponding Find and Not bits to both be set to 1b on any given match condition element.

NOTE 140 - More than one F1h descriptor may be sent. If more than one F1h descriptor is sent, each descriptor is processed before the next descriptor is processed (i.e., they are processed in the order they appear).

6.6.24 MP 3Eh: Engineering Support

See MODE SELECT (6/10) - 15h/55h (see 5.2.14) for how to set these parameters and MODE SENSE (6/10) - 1Ah/5Ah (see 5.2.15) for how to read these parameters.

Mode Page Behaviors (see 4.5 on page 27) describes this device’s non-standard behaviors related to mode parameters.

The mode page policy for this page is described in Policy — non-standard (see 4.5.1 on page 27), but may contain elements which are common to all initiators, as well as elements which are initiator unique.

NOTE 141 - As this page is not for normal application use, it is not returned in mode page 3Fh (all pages). Consequently, this page must be specifically queried and set.

NOTE 142 - For more information on special needs and the usage of this page contact IBM.

<table>
<tr><th>Byte</th><th>Description</th></tr>
<tr><td>0</td><td>
<table>
<tr><th>Bit</th><th>Description</th></tr>
<tr><td>7</td><td>PS (Page Save): 0b. Ignored in MODE SELECT. See Parameter Saveable behavior in 4.5.2.2 on page 29.</td></tr>
<tr><td>6</td><td>Reserved</td></tr>
<tr><td>5-0</td><td>Page Code: 3Eh</td></tr>
</table>
</td></tr>
<tr><td>1</td><td>Page Length: 7Eh</td></tr>
<tr><td>2-95</td><td>Vendor-Reserved</td></tr>
</table>

6.7 Security Protocol Parameters (SPP)

Security Protocol parameters are used by the SECURITY PROTOCOL IN (SPIN) - A2h (see 5.2.39 on page 201)command and by the SECURITY PROTOCOL OUT (SPOUT) - B5h (see 5.2.40 on page 202) command.

The following terms are used in this clause:

Term	Description
SPIN	Security Protocol In
SPOUT	Security Protocol Out

6.7.1 SPIN Pages (00h - Security Protocol Information)

See SECURITY PROTOCOL IN (SPIN) - A2h (see 5.2.39 on page 201) for a description of how to request this page. The Security Protocol Specific field of the SPIN CDB is defined by Table 309

Table 309 — Security Protocol Specific Definitions for Security Protocol 00h

Code	Description	Reference
0000h	Supported security protocol list	see page 458
0001h	Certificate data	see page 458
0002h - FFFFh	Reserved	

6.7.1.1 SPIN (00h[0000h]) - Supported Security Protocols List

See SECURITY PROTOCOL IN (SPIN) - A2h (see 5.2.39 on page 201) for a description of how to request this page. If the SECURITY PROTOCOL field is set to 00h and the SECURITY PROTOCOL SPECIFIC field is set to 0000h in a SECURITY PROTOCOL IN command, the parameter data shall have the format shown in Table 310.

Table 310 — Supported Security Protocols List Structure

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Reserved							
5								
6	(MSB)	SUPPORTED SECURITY PROTOCOL LIST LENGTH (m-7)						
7								(LSB)
8	Supported Security Protocol (first)							
	⋮							
m	Supported Security Protocol (last)							

The Supported Security Protocol list contains the following supported security protocols:

Value	**Description**
00h	Security protocol information (see 6.7.1).
20h	Tape Data Encryption (see 6.7.2).

6.7.1.2 SPIN (00h[0001h]) - Certificate Data

See SECURITY PROTOCOL IN (SPIN) - A2h (see 5.2.39 on page 201) for a description of how to request this page. The drive certificate (if present) is provided in X.509 format via this interface (see SPC-4).

If the Security Protocol field is set to 00h and the Security Protocol Specific field is set to 0001h in a Security Protocol In command, the parameter data shall have the format shown in Table 311.

Table 311 — 0001h - Certificate Data Structure

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	Reserved							
1								
2	(MSB)	CERTIFICATE LENGTH (m-3)						
3								(LSB)
4	CERTIFICATE							
m								

The following parameters apply:

- Certificate Length - The total length, in bytes, of the certificate that follows.
- Certificate - The drive certificate set during manufacturing is returned.

6.7.2 SPIN Pages (20h - Tape Data Encryption)

See SECURITY PROTOCOL IN (SPIN) - A2h (see 5.2.39 on page 201) for a description of how to request this page. The SECURITY PROTOCOL SPECIFIC field specifies the type of report that the application client is requesting. Table 312 shows supported SECURITY PROTOCOL SPECIFIC field values.

Table 312 — Security Protocol Specific Definitions for Security Protocol 20h

Page Code	Description	Reference
0000h	Tape Data Encryption In Support Pages	see page 460
0001h	Tape Data Encryption Out Support Pages	see page 461
0010h	Data Encryption Capabilities	see page 461
0011h	Supported Key Formats	see page 467
0012h	Data Encryption Management Capabilities	see page 467
0020h	Data Encryption Status	see page 468
0021h	Next Block Encryption Status	see page 471
0030h	Random Number page	see page 474

6.7.2.1 SPIN (20h[0000h]) - Tape Data Encryption In Support Pages page

See SECURITY PROTOCOL IN (SPIN) - A2h (see 5.2.39 on page 201) for a description of how to request this page. Supported protocol specific in pages for protocol 20h are indicated above (see SSC-3).

Table 313 — 0000h - Tape Data Encryption In Support Pages Structure

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	(MSB)	PAGE CODE (0000h)						
1								(LSB)
2	(MSB)	PAGE LENGTH (n-3)						
3								(LSB)
Tape Data Encryption In Support page code list								
4	(MSB)	Tape Data Encryption In Support page code (first)						
5								(LSB)
	⋮							
n-1	(MSB)	Tape Data Encryption In Support page code (last)						
n								(LSB)

Table 314 on page 460 show which Tape Data Encryption In page codes are supported:

Table 314 — Tape Data Encryption In page codes

Page Code	Description	Reference
0000h	Tape Data Encryption In Support Pages	see page 460
0001h	Tape Data Encryption Out Support Pages	see page 461
0010h	Data Encryption Capabilities	see page 461
0011h	Supported Key Formats	see page 467
0012h	Data Encryption Management Capabilities	see page 467
0020h	Data Encryption Status	see page 468
0021h	Next Block Encryption Status	see page 471

6.7.2.2 SPIN (20h[0001h]) - Tape Data Encryption Out Support Pages page

See SECURITY PROTOCOL IN (SPIN) - A2h (see 5.2.39 on page 201) for a description of how to request this page. Supported protocol specific out pages for protocol 20h are indicated above (see SSC-3).

Table 315 — 0001h - Tape Data Encryption Out Support Pages Structure

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	(MSB)	Page Code (0001h)						
1								(LSB)
2	(MSB)	Page Length (n-3)						
3								(LSB)
Tape Data Encryption Out Support page code list								
4	(MSB)	Tape Data Encryption Out Support page code (first)						
5								(LSB)
	⋮							
n-1	(MSB)	Tape Data Encryption Out Support page code (last)						
n								(LSB)

Table 316 on page 461 show which Tape Data Encryption Out page codes are supported:

Table 316 — Tape Data Encryption Out page codes

Page Code	Description	Reference
0010h	Set Data Encryption	6.7.3.1 on page 475

6.7.2.3 SPIN (20h[0010h]) - Data Encryption Capabilities page

See SECURITY PROTOCOL IN (SPIN) - A2h (see 5.2.39 on page 201) for a description of how to request this page. Table 97 specifies the format of the Data Encryption Capabilities page.

Table 317 — 0010h - Data Encryption Capabilities page

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	(MSB)	PAGE CODE (0010h)						
1								(LSB)
2	(MSB)	PAGE LENGTH (n-3)						
3								(LSB)
4	Reserved				EXTDECC		CFG_P	
5	Reserved							
19								
Data Encryption Algorithm descriptor list								
20	Data Encryption Algorithm descriptor (first)							
	⋮							
	Data Encryption Algorithm descriptor (last)							
n								

The Data Encryption Capabilities page description follows:

Byte	Description
0-1	PAGE CODE: 0010h
2-3	PAGE LENGTH: The number of bytes that follow
4	

Bit	Description
7-4	Reserved
3-2	EXTDECC (external data encryption control capable): Specifies the external data encryption control capability of the drive.

Value	Description
00b	The external data encryption control capability is not reported.
01b	The drive is not external data encryption control capable.
10b	The drive is external data encryption control capable.
11b	Reserved

Bit	Description
1-0	CFG_P (configuration prevented): Specifies the logical block encryption parameters configuration capabilities for the algorithms reported in the logical block encryption algorithm descriptors.

Value	Description
00b	The logical block encryption configuration capabilities are not reported.
01b	The drive is configured to allow this device server to establish or change logical block encryption parameters.
10b	The drive is configured to not allow this device server to establish or change logical block encryption parameters. In code levels that set this field, this value is reported when in LME mode.
11b	Reserved

Byte	Description
5-19	Reserved
20-n	Data Encryption Algorithm descriptor list. There is one Data Encryption Algorithm Descriptor - Standard Encryption (see 6.7.2.3.1 on page 463) for each supported Data Encryption Algorithm (i.e., one for each density that supports encryption).

Byte	Description
20-43	Data Encryption Algorithm descriptor [First], if any
44-67	Data Encryption Algorithm descriptor [Second], if any
68-91	Data Encryption Algorithm descriptor [Third], if any

Table 318 — Data Encryption Algorithm descriptor list returned

Data Encryption Algorithm descriptor	Model E05 & E06 drives	Model E07 drives	Model E08 drives
First (bytes 20-43)	ALGORITHM INDEX (01h)	ALGORITHM INDEX (01h)	ALGORITHM INDEX (01h)
Second (bytes 44-67)	N/A	ALGORITHM INDEX (02h)	ALGORITHM INDEX (02h)
Third (bytes 68-91)	N/A	N/A	ALGORITHM INDEX (03h)

6.7.2.3.1 Data Encryption Algorithm Descriptor - Standard Encryption

The Standard Encryption Algorithm Descriptor is shown in Table 260 on page 463

Table 260 — Data Encryption Algorithm Descriptor - Standard Encryption Structure

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	ALGORITHM INDEX							
1	Reserved							
2	(MSB)	DESCRIPTOR LENGTH (0014h)						
3								(LSB)
4	AVFMV	SDK_C (0b)	MAC_C (1B)	DELB_C (1B)	DECRYPT_C (2H)		ENCRYPT_C (2H)	
5	AVFCLP		NONCE_C (3H)		KADF_C	VCELB_C (1B)	UKADF (0b)	AKADF (0b)
6	(MSB)	MAXIMUM UNAUTHENTICATED KEY-ASSOCIATED DATA (U-KAD) BYTES						
7								(LSB)
8	(MSB)	MAXIMUM AUTHENTICATED KEY-ASSOCIATED DATA (A-KAD) BYTES						
9								(LSB)
10	(MSB)	LOGICAL BLOCK ENCRYPTION KEY SIZE (0020h)						
11								(LSB)
12	DKAD_C		EEMC_C (2H)		RDMC_C (7h)			EAREM (1b)
13	Reserved							
14		MSDK_COUNT (00000000H)						
15								
16		Reserved						
19								
20	(MSB)	SECURITY ALGROTIHM CODE (00010014h)						
23								(LSB)

Each Data Encryption Algorithm Descriptor is specified by its ALGORITHM INDEX. For fields whose value changes depending on the ALGORITHM INDEX, the value is specified in the description of that field. The Data Encryption Algorithm Descriptor - Standard Encryption Structure description follows:

Byte Description

0 ALGORITHM INDEX: The index of the Data Encryption Algorithm being described. Used by the SECURITY PROTOCOL OUT command Set Data Encryption page to select this algorithm.

ALGORITHM INDEX	**Density Code**
01h	72h (i.e., E05 encrypted format) and 73h (i.e., E06 encrypted format)
02h	54h or 74h (i.e., E07 formats)
03h	55h or 75h (i.e., E08 formats)

1 Reserved

2-3 DESCRIPTOR LENGTH:

4

Bit	Description
7	AVFMV (algorithm valid for mounted volume): Specifies if the algorithm selected by the ALGORITHM INDEX is valid for the currently mounted volume.

Value	Description
0b	There is no volume mounted or the algorithm being described is not valid for the currently mounted volume.
1b	A volume is currently mounted and the encryption algorithm being described is valid for that volume.

Bit	Description
6	SDK_C (supplemental decryption key capable): 0b
5	MAC_C (message authentication code capable): 1b
4	DELB_C (distinguish encrypted logical block capable): 1b
3-2	DECRYPT_C (decryption capabilities):

Value	Description
00b	No capability — The drive has no logical block decryption capability using this algorithm. This value shall be returned if the specified algorithm is disabled.
01b	Software capable — Not Supported.
10b	Hardware capable — The drive has the ability to decrypt logical blocks using this algorithm in hardware.
11b	Capable with external control — The drive has the capability to decrypt logical blocks using this algorithm, but control of the logical block encryption parameters by this device server is prevented.

Bit	Description
1-0	ENCRYPT_C (encryption capabilities):

Value	Description
00b	No capability — The drive has no logical block encryption capability using this algorithm. This value shall be returned if the specified algorithm is disabled.
01b	Software capable — Not Supported.
10b	Hardware capable — The drive has the ability to encrypt logical blocks using this algorithm in hardware.
11b	Capable with external control — The drive has the capability to encrypt logical blocks using this algorithm, but control of the logical block encryption parameters by this device server is prevented.

5

Bit	Description
7-6	AVFCLP (algorithm valid for current logical position): Specifies if the encryption algorithm being specified is valid for writing to the mounted volume at the current logical position

Value	Description
00b	Current logical position is not applicable to the encryption algorithm validity or no volume is loaded.
01b	The encryption algorithm being specified is not valid for writing to the mounted volume at the current logical position.
10b	The encryption algorithm being specified is valid for writing to the mounted volume at the current logical position.
11b	Reserved

Bit	Description
5-4	NONCE_C:

Value	Description
00b-10b	Not Supported.
11b	The drive supports all or part of the nonce value provided by the application client. If the Set Data Encryption page that enables encryption does not include a nonce value descriptor, the drive generates the nonce value.

3 KADF_C (KAD format capable): Indicates if the drive is KAD format capable, that is, it supports:

A) the ENCRYPTION PARAMETERS KAD FORMAT field in the Data Encryption Status page (see 6.7.2.6);

B) the NEXT BLOCK KAD FORMAT field in the Next Block Encryption Status page (see 6.7.2.7); and

C) the KAD FORMAT field in the Set Data Encryption page (see 6.7.3.1).

Value	**Description**
0b	The drive is not KAD format capable as described herein.
1b	The drive is KAD format capable as described herein.

2 VCELB_C (volume contains encrypted logical blocks capable): 1B

1 UKADF (U-KAD fixed): Specifies restrictions on the length of the U-KAD, if present.

Value	**Description**
0b	If the value in the MAXIMUM UNAUTHENTICATED KEY-ASSOCIATED DATA BYTES field is non-zero, then the length of the U-KAD, if present in the parameter data for a SECURITY PROTOCOL OUT command, shall be a value between one and the value in the MAXIMUM UNAUTHENTICATED KEY-ASSOCIATED DATA BYTES field.
1b	The length of the U-KAD in the parameter data for a SECURITY PROTOCOL OUT command shall be MAXIMUM UNAUTHENTICATED KEY-ASSOCIATED DATA BYTES in length and the MAXIMUM UNAUTHENTICATED KEY-ASSOCIATED DATA BYTES field shall contain a non-zero value.

0 AKADF (A-KAD fixed): Specifies restrictions on the length of the A-KAD, if present.

Value	**Description**
0b	If the value in the MAXIMUM AUTHENTICATED KEY-ASSOCIATED DATA BYTES field is non-zero, then the length of the A-KAD, if present in the parameter data for a SECURITY PROTOCOL OUT command, shall be a value between one and the value in the MAXIMUM AUTHENTICATED KEY-ASSOCIATED DATA BYTES field.
1b	The length of A-KAD in the parameter data for a SECURITY PROTOCOL OUT command shall be MAXIMUM AUTHENTICATED KEY-ASSOCIATED DATA BYTES in length and the MAXIMUM AUTHENTICATED KEY-ASSOCIATED DATA BYTES field shall contain a non-zero value.

6-7 MAXIMUM UNAUTHENTICATED KEY-ASSOCIATED DATA BYTES:

Indicates the maximum size of the unauthenticated key-associated data supported by the device server for this algorithm.

ALGORITHM INDEX	**Value**
01h	0000h
02h	0020h
03h	0020h

8-9 MAXIMUM AUTHENTICATED KEY-ASSOCIATED DATA BYTES:

Indicates the maximum size of the authenticated key-associated data supported by the device server for this algorithm. The value for each algorithm follows.

ALGORITHM INDEX	**Value**
01h	000Ch
02h	003Ch
03h	003Ch

10-11 LOGICAL BLOCK ENCRYPTION KEY SIZE:

Indicates the size in bytes of the logical block encryption key required by the algorithm.

12

Bit **Description**

7-6 DKAD_C (decryption KAD capabilties):

Indicates the decryption capabilities when the DECRYPTION MODE field of the Set Data Encryption page (see 6.7.3.1) is set to DECRYPT or MIXED.

Value	**Description**
00b	not specified — No capabilities are specified.
01b	KAD Required — Not Supported.
10b	KAD Not Allowed — Not Supported.
11b	KAD Capable — If the MAXIMUM UNAUTHENTICATED KEY-ASSOCIATED DATA BYTES field is set to a non-zero value or the MAXIMUM AUTHENTICATED KEY-ASSOCIATED DATA BYTES field is set to a non-zero value, then this value is supported.

5-4 EEMC_C (external encryption mode control capabilities):

Indicates the capabilities the encryption algorithm provides to the application client to control write operations that transfer encrypted logical blocks while the encryption mode is set to EXTERNAL.

Value	**Description**
00b	No capabilities are specified.
01b	Not Supported.
10b	The encryption algorithm allows write operations in EXTERNAL encryption mode. The device server does act as a KCDLU for this encryption algorithm.
11b	Reserved

3-1 RDMC_C (raw decryption mode control capabilities):

Indicates the capabilities the encryption algorithm provides to the application client to control read operations that access encrypted logical blocks while the decryption mode is set to RAW.

Value	**Description**
000b	No capabilities are specified.
001b	Not Supported.
010b-011b	Reserved
100b	Not Supported.
101b	Not Supported.
110b	Not Supported.
111b	The encryption algorithm enables read operations in RAW mode by default and does not allow the application client to control RAW reads via the RDMC field in the Set Data Encryption page (see 6.7.3.1). The device server acts as a KCSLU for this encryption algorithm.

0 EAREM (encryption algorithm records encryption mode):

Value	**Description**
0b	The encryption mode is not recorded with each encrypted logical block.
1b	The encryption mode is recorded with each encrypted logical block.

The value returned depends on the ALGORITHM INDEX.

ALGORITHM INDEX	**Value**
01h	0b
02h	1b
03h	1b

13 Reserved

14-15 MSDK_COUNT (maximum supplemental decryption key count): 0000h

16-19 Reserved

20-23 SECURITY ALGORITHM CODE:

Contains a security algorithm code (see SPC-4).

6.7.2.4 SPIN (20h[0011h]) - Supported Key Formats page

See SECURITY PROTOCOL IN (SPIN) - A2h (see 5.2.39 on page 201) for a description of how to request this page.

The structure of the Supported Key Formats page is shown in Table 261 on page 467

Table 261 — 0011h - Supported Key Formats page Structure

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	(MSB)	PAGE CODE (0011h)						
1								(LSB)
2	(MSB)	PAGE LENGTH (n-3)						
3								(LSB)
Supported Key Format list								
4	Supported Key Format [first]							
	⋮							
n	Supported Key Format [last]							

Table 262 — Supported Key Formats

Key Format	Description	Reference
00h	Plaintext Key Format	See page 467

6.7.2.4.1 Plaintext Key Format (00h)

The Plaintext Key Format structure is shown in Table 263 on page 467

Table 263 — 00h - Plaintext Key Format Structure

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	KEY							
n								

6.7.2.5 SPIN (20h[0012h]) - Data Encryption Management Capabilities

See SECURITY PROTOCOL IN (SPIN) - A2h (see 5.2.39 on page 201) for a description of how to request this page.

Table 264 on page 468 specifies the format of the Data Encryption Management Capabilities page.

Table 264 — 0012h - Data Encryption Management Capabilities page

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	(MSB)	PAGE CODE (0012h)						
1								(LSB)
2	(MSB)	PAGE LENGTH (000ch)						
3								(LSB)
4	Reserved							LOCK_C (1b)
5	Reserved					CKOD_C (1b)	CKORP_C (1b)	CKORL_C (1b)
6	Reserved							
7	Reserved					AITN_C (1b)	LOCAL_C (1b)	PUBLIC_C (1b)
8	Reserved							
15								

6.7.2.6 SPIN (20h[0020h]) - Data Encryption Status page

Table 265 specifies the format of the Data Encryption Status page

This is a query of information which was set with Security Protocol Out 0010h - Set Data Encryption and does not reflect the actual state of the medium itself or of any data on medium..

Table 265 — 0020h - Data Encryption Status page

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	(MSB)	PAGE CODE (0020h)						
1								(LSB)
2	(MSB)	PAGE LENGTH (n-3)						
3								(LSB)
4	I_T NEXUS SCOPE			Reserved		LOGICAL BLOCK ENCRYPTION SCOPE		
5	ENCRYPTION MODE							
6	DECRYPTION MODE							
7	ALGORITHM INDEX							
8	(MSB)	KEY INSTANCE COUNTER						
11								(LSB)
12	Reserved	parameters control			VCELB	CEEMS		RDMD
13	ENCRYPTION PARAMETERS KAD FORMAT							
14	(MSB)	ASDK_COUNT (0000h)						
15								(LSB)
16	Reserved							
23								
Key-associated data descriptor list								
24	Key-associated data descriptor (first)							
	Key-associated data descriptor (last)							
n								

The I_T NEXUS SCOPE field shall contain the value from the logical block encryption scope saved for the I_T_L nexus through which this command was received.

The LOGICAL BLOCK ENCRYPTION SCOPE field shall contain the value from the logical block encryption scope in the saved logical block encryption parameters currently associated with the I_T_L nexus on which this command was received.

The ENCRYPTION MODE field shall contain the value from the encryption mode in the saved logical block encryption parameters currently associated with the I_T_L nexus on which this command was received.

The DECRYPTION MODE field shall contain the value from the decryption mode in the saved logical block encryption parameters currently associated with the I_T_L nexus on which this command was received.

The ALGORITHM INDEX field shall contain the value from the algorithm index in the saved logical block encryption parameters currently associated with the I_T_L nexus on which this command was received. If the ENCRYPTION MODE field and the DECRYPTION MODE field are both set to DISABLE, the value in the ALGORITHM INDEX field is undefined.

The KEY INSTANCE COUNTER field contains the value of the logical block encryption parameters key instance counter assigned to the logical block encryption key indicated by the LOGICAL BLOCK ENCRYPTION SCOPE field value.

The PARAMETERS CONTROL field specifies information on how the logical block encryption parameters are controlled. The PARAMETERS CONTROL field values are specified in table 266.

Table 266 — PARAMETERS CONTROL field

Code	Description
000b	Logical block encryption parameters control is not reported. This value is returned on Ultrium 4 devices
001b	Logical block encryption parameters are not exclusively controlled by external data encryption control. This value is returned when in AME encryption mode.
010b	Logical block encryption parameters are exclusively controlled by the sequential-access device server.
011b	Logical block encryption parameters are exclusively controlled by the automation/drive interface device server. This value is returned when in LME or ADC Controlled encryption mode.
100b	Not supported.
101b-111b	Reserved

A volume contains encrypted logical blocks (VCELB) bit set to one indicates that the mounted volume contains an encrypted logical block. A VCELB bit set to zero indicates that either:

a) the mounted volume does not contain any encrypted logical blocks;
b) there is no volume mounted; or
c) the VCELB_C bit in the Data Encryption Capabilities page is set to zero.

The raw decryption mode disabled (RDMD) bit shall be set to one if the device entity is configured to mark each encrypted record as disabled for raw read operations based on the RDMC_C value and the raw decryption mode disable parameter in the saved logical block encryption parameters currently associated with the I_T_L nexus on which the command was received.

The check external encryption mode status (CEEMS) field shall contain the value from the check external encryption mode parameter in the saved logical block encryption parameters currently associated with the I_T_L nexus on which the command was received.

The ENCRYPTION PARAMETERS KAD FORMAT field shall contain the value from the KAD_FORMAT in the saved logical block encryption parameters currently associated with the I_T_L nexus on which this command was received. If the encryption algorithm specified in the ALGORITHM INDEX field reports a KADF_C bit set to zero, then the ENCRYPTION PARAMETERS KAD FORMAT field shall be set to zero.

The available supplemental decryption key count (ASDK_COUNT) field shall bet set to zero because the device server is not capable of supporting supplemental decryption keys.

If the ENCRYPTION MODE field and the DECRYPTION MODE field are both set to DISABLE, the key-associated data descriptors list shall not be included in the page.

If either the ENCRYPTION MODE field or the DECRYPTION MODE field is set to a value other than DISABLE, the key-associated data descriptors list shall include Tape Data Encryption descriptors describing attributes assigned to the logical block encryption key defined by the I_T NEXUS SCOPE and LOGICAL BLOCK ENCRYPTION SCOPE fields at the time the logical block encryption key was established in the device entity (see "Key-Associated Data (KAD) Descriptors" on page 478). If more than one key-associated data descriptor is included, they shall be in increasing numeric order of the value in the KEY DESCRIPTOR TYPE field. Descriptors shall be included as defined by the following paragraphs.

An unauthenticated key-associated data descriptor shall be included if an unauthenticated key-associated data descriptor was included when the logical block encryption key was established in the device entity. The AUTHENTICATED field is reserved. The KEY DESCRIPTOR field shall contain the U-KAD value associated with the logical block encryption key.

An authenticated key-associated data descriptor shall be included if an authenticated key-associated data descriptor was included when the logical block encryption key was established in the device entity. The

AUTHENTICATED field is reserved. The KEY DESCRIPTOR field shall contain the A-KAD value associated with the logical block encryption key.

A nonce value descriptor shall be included if a nonce value descriptor was included when the logical block encryption key was established in the device entity. The AUTHENTICATED field is reserved. The KEY DESCRIPTOR field shall contain the nonce value associated with the logical block encryption key. A nonce value descriptor may be included if no nonce value descriptor was included when the logical block encryption key was established in the device entity. In this case, the KEY DESCRIPTOR field shall be set to the nonce value established by the device entity for use with the selected logical block encryption key.

A metadata key-associated data descriptor shall be included if the metadata key-associated data descriptor was included when the logical block encryption parameters were established. The KEY DESCRIPTOR field shall contain the M-KAD value associated with the logical block encryption key.

6.7.2.7 SPIN (20h[0021h]) - Next Block Encryption Status page

See SECURITY PROTOCOL IN (SPIN) - A2h (see 5.2.39 on page 201) for a description of how to request this page.

NOTE 143 - Next block encryption status may not be available in all situations. When it is not known appropriate values are returned as per the standard. In most situations next block information is available during read operations when read ahead is being performed. This is automatically managed by the device.

Table 267 specifies the format of the Next Block Encryption Status page

Table 267 — 0021h - Next Block Encryption Status page

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	(MSB)	PAGE CODE (0021h)						
1								(LSB)
2	(MSB)	PAGE LENGTH (n-3)						
3								(LSB)
4	(MSB)	LOGICAL OBJECT NUMBER						
11								(LSB)
12	COMPRESSION STATUS (0h)				ENCRYPTION STATUS			
13	ALGORITHM INDEX							
14	Reserved						EMES	RDMDS
15	NEXT BLOCK KAD FORMAT							
Key-associated data descriptor list								
16	Key-associated data descriptor [first]							
	Key-associated data descriptor [last]							
n								

The LOGICAL OBJECT NUMBER field contains the logical object identifier of the next logical object.

The COMPRESSION STATUS field values are specified in table 268.

Table 268 — COMPRESSION STATUS field

Code	Description
0h	The device entity is incapable of determining if the logical object referenced by the LOGICAL OBJECT NUMBER field has been compressed.
1h	Not Supported.
2h	Not Supported.
3h	Not Supported.
4h	Not Supported.
5h-Fh	Reserved

The ENCRYPTION STATUS field values are specified in table 269.

Table 269 — ENCRYPTION STATUS field

Code	Description
0h	The device entity is incapable of determining if the logical object referenced by the LOGICAL OBJECT NUMBER field has been encrypted.
1h	The device entity is capable of determining if the logical object referenced by the LOGICAL OBJECT NUMBER field has been encrypted, but is not able to at this time. Possible reasons are: a) the next logical block has not yet been read into the buffer; b) there was an error reading the next logical block; or c) there are no more logical blocks (i.e., end-of-data).
2h	The device entity has determined that the logical object referenced by the LOGICAL OBJECT NUMBER field is not a logical block.
3h	The device entity has determined that the logical object referenced by the LOGICAL OBJECT NUMBER field is not encrypted.
4h	The device entity has determined that the logical object referenced by the LOGICAL OBJECT NUMBER field is encrypted by an algorithm that is not supported by this device server. The values in the key-associated data descriptors list contain information pertaining to the encrypted logical block.
5h	The device entity has determined that the logical object referenced by the LOGICAL OBJECT NUMBER field is encrypted by an algorithm that is supported by this device server. The values in the ALGORITHM INDEX field and key-associated data descriptors list contain information pertaining to the encrypted logical block.
6h	The device entity has determined that the logical object referenced by the LOGICAL OBJECT NUMBER field is encrypted by an algorithm that is supported by this device server, but the device entity is either not enabled to decrypt or does not have the correct logical block encryption key or nonce value to decrypt the encrypted logical block.
7h-Fh	Reserved

The ALGORITHM INDEX field indicates which of the encryption algorithms reported by the SECURITY PROTOCOL IN command Data Encryption Capabilities page was used to encrypt the logical block. For values in the ENCRYPTION STATUS field (see table 269) that do not indicate the ALGORITHM INDEX field is valid, the algorithm index is undefined.

The encryption mode external status (EMES) bit shall be set to one if:

a) the ENCRYPTION STATUS field is set to either 5h or 6h;
b) the EAREM bit in the algorithm descriptor (see 6.7.2.3) for the algorithm specified by the ALGORITHM INDEX field is set to one; and
c) the next logical block is marked as having been written to the medium while the encryption mode was set to EXTERNAL.

The EMES bit shall be set to zero if:

a) the ENCRYPTION STATUS field is set to a value other than 5h or 6h;
b) the EAREM bit in the algorithm descriptor (see 6.7.2.3) for the algorithm specified by the ALGORITHM INDEX field is set to zero; or
c) the next logical block is marked as having been written to the medium while the encryption mode was set to ENCRYPT.

The raw decryption mode disabled status (RDMDS) bit shall be set to one if:

a) the device server supports raw decryption mode;
b) the ENCRYPTION STATUS field is set to either 5h or 6h; and
c) the next logical block is marked as disabled for raw decryption mode operations (see 6.7.3.1).

The RDMDS bit shall be set to zero if:

a) the device server does not support raw decryption mode;
b) the ENCRYPTION STATUS field is set to a value other than 5h or 6h; or
c) the next logical block is not marked as disabled for raw decryption mode operations.

If the value in the ENCRYPTION STATUS field indicates that the next logical block is encrypted by a supported algorithm, then the NEXT BLOCK KAD FORMAT field shall contain the KAD_FORMAT logical block encryption parameters associated with the encrypted logical block. If the value in the ENCRYPTION STATUS field does not indicate that the next logical object is an encrypted logical block, then the NEXT BLOCK KAD FORMAT field shall be ignored. If the encryption algorithm specified in the ALGORITHM INDEX field reports a KADF_C bit set to zero, then the NEXT BLOCK KAD FORMAT field shall be set to zero.

If the value in the ENCRYPTION STATUS field indicates that the next logical block is encrypted by a supported algorithm, then the device server shall include in the key-associated data descriptor list (see “Key-Associated Data (KAD) Descriptors” on page 478) all key-associated data that is associated with the encrypted logical block. If more than one key-associated data descriptor is included in the Next Block Encryption Status page, then they shall be in increasing numeric order of the value in the KEY DESCRIPTOR TYPE field.

An unauthenticated key-associated data descriptor (see 6.7.3.2.1) shall be included if any unauthenticated key-associated data is associated with the next logical block. The AUTHENTICATED field shall be set to 1. The KEY DESCRIPTOR field shall contain the U-KAD value associated with the encrypted logical block.

An authenticated key-associated data descriptor (see 6.7.3.2.2) shall be included if any authenticated key-associated data is associated with the next logical block. The AUTHENTICATED field shall indicate the status of the authentication done by the device entity. The KEY DESCRIPTOR field shall contain the A-KAD value associated with the encrypted logical block.

The Next Block Encryption Status page may include a nonce value descriptor (see 6.7.3.2.3). If a nonce value descriptor is included, then the AUTHENTICATED field shall indicate the status of the authentication done by the device entity. The KEY DESCRIPTOR field shall contain the nonce value associated with the encrypted logical block.

A metadata key-associated data descriptor (see 6.7.3.2.4) shall be included if any M-KAD is associated with the next logical block and the decryption mode is set to RAW in the saved logical block encryption parameters currently associated with the I_T_L nexus on which this command was received. The KEY DESCRIPTOR field shall contain the M-KAD value associated with the encrypted logical block.

The following table indicates valid combinations of record status, Decryption Mode and returned Key-Associated Descriptors reflecting the currently setup state of the device.

Table 270 — SPIN (20h[0021h]) - KAD Parameters by Mode

Record Information	Decryption Mode	Read Data	Status	Key-Associated Descriptors				Notes
				uKAD 00h	aKAD (DKi) 01h	Nonce 02h	Meta data 03h	
Unknown	any	?	1h	O	n/a	n/a	n/a	
Filemark	any	n/a	2h	n/a	n/a	n/a	n/a	may be unknown
EOD	any	n/a	2h	O	n/a	n/a	n/a	may be unknown
Error	any	n/a	1h	O	n/a	n/a	n/a	may be unknown
Cleartext	0h Disable	C	3h	n/a	n/a	n/a	n/a	
Cleartext	1h Raw	E	3h	n/a	n/a	n/a	n/a	not readable
Cleartext	2h Decrypt	E	3h	n/a	n/a	n/a	n/a	not readable
Cleartext	3h Mixed	C	3h	n/a	n/a	n/a	n/a	
Encrypted	0h Disable	E	4h 5h or 6h	O	Y	N	N	not readable
Encrypted	1h Raw	R		N	N	N	Y	
Encrypted	2h Decrypt	C[1]		O	Y	N	N	
Encrypted	3h Mixed	C[1]		O	Y	N	N	

Legend:
Y: element is required
O: element is optional
N: element is not present
n/a: not applicable (element is not present)
C: cleartext (not encrypted)
R: raw (compressed encoded/encrypted)
E: error condition, record cannot be read

Notes:
1: Data is decrypted
2: uKAD is only supported for 3592A4 and later formats

6.7.2.7.1 Key-Associated Data (KAD) Descriptors

See “Key-Associated Data (KAD) Descriptors” on page 478

6.7.2.8 SPIN (20h[0030h]) - Random Number page

Table 271 specifies the format of the Random Number page.

Table 271 — Random Number page

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	(MSB)	PAGE CODE (0030h)						
1								(LSB)
2	(MSB)	PAGE LENGTH (32)						
3								(LSB)
4	(MSB)	RANDOM NUMBER						
35								(LSB)

The RANDOM NUMBER field contains a secure random number, suitable for use as a random nonce, that is generated by the device server using a source of entropy available within the device. Each request for the Random Number page generates a new secure random number for the RANDOM NUMBER field.

6.7.3 SPOUT Pages (20h - Tape Data Encryption security protocol)

See SECURITY PROTOCOL OUT (SPOUT) - B5h (see 5.2.40 on page 202) for a description of how to send this page.

The Security Protocol Specific field (see Table 132, “Security Protocol Out B5h CDB,” on page 202) specifies the type of page that the application client is sending. Table 272 shows supported values.

Table 272 — Security Protocol Specific Definitions for Security Protocol 20h

Page Code	Description	Reference
0010h	Set Data Encryption	See page 475

6.7.3.1 SPOUT (20h[0010h]) - Set Data Encryption

See SECURITY PROTOCOL OUT (SPOUT) - B5h (see 5.2.40 on page 202) for a description of how to send this page.

Table 273 specifies the format of the Set Data Encryption page.

Table 273 — 0010h - Set Data Encryption page

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	(MSB)	PAGE CODE (0010h)						
1								(LSB)
2	(MSB)	PAGE LENGTH (m-3)						
3								(LSB)
4	SCOPE			Reserved				LOCK
5	CEEM		RDMC (00b)		SDK (0b)	CKOD	CKORP	CKORL
6	ENCRYPTION MODE							
7	DECRYPTION MODE							
8	ALGORITHM INDEX							
9	LOGICAL BLOCK ENCRYPTION KEY FORMAT							
10	KAD FORMAT							
11	Reserved							
17								
18	(MSB)	LOGICAL BLOCK ENCRYPTION KEY LENGTH (n-19)						
19								(LSB)
20	LOGICAL BLOCK ENCRYPTION KEY							
n								
Key-associated data descriptor list								
n+1	Key-associated data descriptor [first]							
	Key-associated data descriptor [last]							
m								

The following parameters apply:

- SCOPE

Value	**Description**
0h	Public
1h	Local
2h	All I_T Nexus

- LOCK
- CEEM (check external encryption mode)

Value	**Description**
00b	Same as 01b.
01b	Do not check the encryption mode that was in use when the logical block was written to the medium.
10b	On read and verify commands, check the encryption mode that was in use when the logical block was written to the medium. Report an error if the logical block was written in EXTERNAL mode
11b	On read and verify commands, check the encryption mode that was in use when the logical block was written to the medium. Report an error if the logical block was written in ENCRYPT mode

A CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST, and the additional sense code set to INVALID FIELD IN PARAMETER DATA is returned if the CEEM field is set to either 10b or 11b, and:

a) the DECRYPTION MODE field is set to DISABLE; or
b) the EAREM bit in the algorithm descriptor (see 6.7.2.3) for the algorithm specified by the ALGORITHM INDEX field is set to zero.

- RDMC (raw decryption mode control) (00b)

This field is ignored if the ENCRYPTION MODE field is not set to ENCRYPT.

Value	**Description**
00b	The device entity shall mark each encrypted logical block per the default setting for the algorithm.
10b	Not Supported.
11b	Not Supported.

A CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense key set to INVALID FIELD IN PARAMETER DATA is returned if:

a) the ENCRYPTION MODE field is set to ENCRYPT;
b) the RDMC field is set to 10b or 11b; and
c) the RDMC_C field in the algorithm descriptor for the encryption algorithm selected by the value in the ALGORITHM INDEX field is set to 1h, 6h, or 7h.

- CKOD (clear key on demount)
- CKORP (clear key on reservation preempt)
- CKORL (clear key on reservation loss)
- ENCRYPTION MODE

Value	**Description**
0h	Disable
1h	External
2h	Encrypt

- DECRYPTION MODE

Value	**Description**
0h	Disable
1h	Raw
2h	Decrypt
3h	Mixed

- ALGORITHM INDEX: Any value is permitted (does not need to match format), and the usage will be dictated by format. Other fields defined by algorithm capabilities (see 6.7.2.3) are checked and enforced when writing. An error will return 7/7407 (i.e., DATA PROTECT, ENCRYPTION PARAMETERS NOT USEABLE)
- LOGICAL BLOCK ENCRYPTION KEY FORMAT
- KAD FORMAT

A non-zero value is only allowed if the algorithm in the ALGORITHM INDEX field reports a KADF_C bit set to one.

Value	**Description**
0h	Unspecified
1h	Binary logical block encryption key name
2h	ASCII logical block encryption key name

If the KAD FORMAT value is non-zero, then:

a) a) only an A-KAD descriptor is provided and the authenticated key-associated data is the key name;
b) b) only a U-KAD descriptor is provided and the unauthenticated key-associated data is the key name; or
c) c) both an A-KAD descriptor and a U-KAD descriptor is provided and the key name is formed by the authenticated key-associated data followed by the unauthenticated key-associated data.

- LOGICAL BLOCK ENCRYPTION KEY LENGTH

Value	**Description**
0000h	When no Key is specified
0020h	When Key is specified using Key Format 00h

- LOGICAL BLOCK ENCRYPTION KEY
- Key-Associated Descriptors List (See “Key-Associated Data (KAD) Descriptors” on page 478)

The following table indicates valid combinations of Encryption Mode and Decryption Mode and mandatory, optional and prohibited Key and Key-Associated Descriptors.

Table 274 — SPOUT (20h[0010h]) - KAD Parameters by Mode

Encryption Mode	Decryption Mode	R/W Data	Key	Key-Associated Descriptors				Notes
				uKAD 00h	aKAD (DKi) 01h	Nonce 02h	Meta data 03h	
0h Disable	0h Disable	C/C	P	P	P	P	P	
0h Disable	1h Raw	R/C	P	P	P	P	P	not recommended
0h Disable	2h Decrypt	C[4]/C	M [2]	P	P	P	P	not recommended
0h Disable	3h Mixed	C[6]/C	M [2]	P	P	P	P	
1h External	0h Disable	C/R	P	P	P	P	M [1]	not recommended
1h External	1h Raw	R/R	P	P	P	P	M [1]	
1h External	2h Decrypt	C[4]/R	M [2]	P	P	P	M [1]	not recommended
1h External	3h Mixed	C[6]/R	M [2]	P	P	P	M [1]	not recommended
2h Encrypt	0h Disable	C/C[4]	M [1]	O [1]	O [1]	O [1,3]	P	
2h Encrypt	1h Raw	R/C[4]	M [1]	O [1]	O [1]	O [1,3]	P	not recommended
2h Encrypt	2h Decrypt	C[5]/C[4]	M	O [1]	O [1]	O [1,3]	P	
2h Encrypt	3h Mixed	C[6]/C[4]	M	O [1]	O [1]	O [1,3]	P	
Legend: M: element is mandatory (required) P: element prohibited (must not be present) O: element is optional (may be device generated) I: element is ignored (may be present) C: cleartext (not encrypted) R: raw (compressed encoded/encrypted)				Notes: 1: Only used for writing 2: Only used for reading 3: May be partially ignored 4: Data is encrypted 5: Data is decrypted 6: Data is decrypted (if needed)				

6.7.3.2 Key-Associated Data (KAD) Descriptors

6.7.3.2.1 KAD 00h - UKAD (Unathenticated KAD)

The UKAD field is an optional field which is used when writing and is recorded with each record.

Table 275 — KAD 00h - UKAD (Unathenticated KAD)

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	KEY DESCRIPTOR TYPE (00h)							
1	Reserved					AUTHENTICATED		
2	(MSB)	KEY DESCRIPTOR LENGTH (n-3)						
3								(LSB)
4	UNAUTHENTICATED DATA							
n								

The following parameters apply:

- AUTHENTICATED

Value	**Description**
0h	Reserved (must be set for Security Protocol Out - 0010h)
1h	Not Covered by Authentication (only Security Protocol In)

- KEY DESCRIPTOR LENGTH: Maximum size is determined by the value in the ALGORITHM INDEX field (see 6.7.2.3.1).
- UNATHENTICATED DATA

6.7.3.2.2 KAD 01h - DKi (Data Key Identifier)

The DKi KAD field is an optional field which is used when writing and is recorded with each record. This is the AKAD field.

Table 276 — KAD 01h - DKi (Data Key Identifier)

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
0	KEY DESCRIPTOR TYPE (01h)							
1	Reserved					AUTHENTICATED		
2	(MSB)	KEY DESCRIPTOR LENGTH (n-3)						
3								(LSB)
4	DKi							
n								

The following parameters apply:

- AUTHENTICATED

Value	**Description**
0h	Reserved (must be set for Security Protocol Out)
2h	No attempt has been made to authenticate (only Security Protocol In)

- KEY DESCRIPTOR LENGTH: Maximum size is determined by the value in the ALGORITHM INDEX field (see 6.7.2.3.1).
- DKi

6.7.3.2.3 KAD 02h - Nonce

The Nonce/IV is not technically KAD. The Nonce may be set to provide the initial value for IV generation for write operations. This field is optional and the device is capable of generating high quality random IV values. When an application specifies nonce values, it is possible that a systemic cryptographic weakness may be introduced into the system. It is strongly recommended that nonce values are not supplied by the application.

> NOTE 144 - The Nonce KAD is only reported by the device in Security Protocol In X'0020' - Data Encryption Status, and the value returned is the exact value specified in X'0010' - Set Data Encryption. This may not reflect the actual nonce or IV used for writing encrypted data.

NOTE 145 - IV values are constructed using only part of the specified Nonce value and are altered for each write in a device dependent manner.

Table 277 — KAD 02h - Nonce

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	Key Descriptor Type (02h)							
1	Reserved					Authenticated		
2	(MSB)	KEY DESCRIPTOR LENGTH (000Ch)						
3								(LSB)
4	NONCE/IV							
n								

The following parameters apply:

- AUTHENTICATED

Value	Description
0h	Reserved (must be set for Security Protocol Out)
1h	Not Covered by Authentication (only Security Protocol In)

- KEY DESCRIPTOR LENGTH: the only supported length is 000Ch
- NONCE/IV

6.7.3.2.4 KAD 03h - MKAD (Metadata)

The MKAD field is used for a keyless copy operation (i.e., RAW decryption mode and EXTERNAL encryption mode).

Table 278 — KAD 03h - MKAD (Metadata)

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	KEY DESCRIPTOR TYPE (03h)							
1	Reserved					AUTHENTICATED		
2	(MSB)	KEY DESCRIPTOR LENGTH (n-3)						
3								(LSB)
4	KEY DESCRIPTOR							
n								

The following parameters apply:

- AUTHENTICATED

Value	Description
0h	Reserved (must be set for Security Protocol Out)
2h	No attempt has been made to authenticate the value in the KEY DESCRIPTOR field (only Security Protocol In)

- KEY DESCRIPTOR LENGTH
- KEY DESCRIPTOR: DATA REQUIRED BY THE ENCRYPTION ALGORITHM FOR A KEYLESS COPY OPERATION.

6.8 Sense data

6.8.1 Sense data overview

Sense data is returned in autosense to a CHECK CONDITION status or in response to a REQUEST SENSE - 03h (see 5.2.36 on page 198) command. Sense data format is returned in one of two formats depending on the sense key. The typical sense data format is for all sense key values except COPY ABORTED (i.e., Ah).

6.8.2 Sense Data Format for most Sense Keys

The sense data format for all sense key's except COPY ABORTED (i.e., Ah) is described in this clause (see table 279). Sense data format for COPY ABORTED (i.e., Sense Key Ah) (see 6.8.3 on page 491) describes the format for a sense key of COPY ABORTED (i.e., Ah).

Table 279 — Sense Data Format for most sense key

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	VALID	RESPONSE CODE						
1	Obsolete (00h)							
2	FILEMARK	EOM	ILI	Reserved	SENSE KEY			
3	INFORMATION							
6								
7	ADDITIONAL SENSE LENGTH (n-7)							
8	COMMAND-SPECIFIC INFORMATION							
…								
11								
12	ADDITIONAL SENSE CODE							
13	ADDITIONAL SENSE CODE QUALIFIER							
14	FIELD REPLACEABLE UNIT CODE							
SKSV (0b)	Vendor-Unique Indicators (i.e., SKSV=0b)							
15	SKSV (0b)	TAPE POSITION INDICATORS				Rsvd	PERMANENT	SIM/MIM
16	FAULT SYMPTOM CODE							
17								
SKSV (1b)	Sense Key Specific (i.e., SKSV=1b)							
15	SKSV (1b)	C/D	Reserved		BPV	BIT POINTER		
16	FIELD POINTER							
17								
18	Rsvd	SMLA	DUMP	TDIR_INV	Rsvd	EOD_PSEG	PSEG	PORT
19	Vendor-Reserved							
20	Vendor-Reserved							
21	Vendor-Reserved							
22	Vendor-Reserved							
23	MEDIUM ACCESS				MEDIUM ASSOCIATION STATE			
24	Medium Descriptor flags							
	INIT_REQD	HK_INTEG	PARTITIONED	PART_INTEG	MEDIUM_CHK	INCOMP_FMT	PWP	LWP
25	Vendor-Reserved							

Table 279 — Sense Data Format for most sense key

Byte	Bit							
	7 msb	6	5	4	3	2	1	0 lsb
26	Vendor-Reserved							
…								
29								
30	FAILING COMMAND							
31	FIRST FAULT SYMPTOM CODE FLAG DATA							
32								
33	SECOND FAULT SYMPTOM CODE							
34								
35	SECOND FAULT SYMPTOM COD FLAG DATA							
36								
37	NEXT-TO-LAST FAULT SYMPTOM CODE							
38								
39	NEXT-TO-LAST FAULT SYMPTOM CODE FLAG DATA							
40								
41	LAST FAULT SYMPTOM CODE							
42								
43	LAST FAULT SYMPTOM COD FLAG DATA							
44								
45	Vendor-Reserved							
46	LOGICAL BLOCK NUMBER							
…								
49								
50	DATASET NUMBER							
…								
52								
53	PHYSICAL WRAP NUMBER							
54	Partition Information							
	PARTITIONING TYPE				Reserved			
55	ACTIVE PARTITION							
56	RELATIVE LPOS							
…								
59								
60	LOGICAL WRAP NUMBER							
61	LAST SPEED INDEX							
62	SARS DRIVE RELATIVE QUALITY							
63	SARS MEDIA RELATIVE QUALITY							
64	Vendor-Reserved							

Table 279 — Sense Data Format for most sense key

<table>
<tr><th rowspan="2">Byte</th><th colspan="8">Bit</th></tr>
<tr><th>7 msb</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0 lsb</th></tr>
<tr><td>65
…
69</td><td colspan="8">Vendor-Reserved</td></tr>
<tr><td>70</td><td>CLN RQD_NM</td><td>CLN_RQD_TR</td><td>H_DUMP</td><td>CAL_RQD</td><td>Reserved</td><td>CLNR_EXP</td><td>CLNR_FAIL</td><td>Reserved</td></tr>
<tr><td>71</td><td colspan="8">Vendor-Reserved</td></tr>
<tr><td>72
…
78</td><td colspan="8">MICROCODE EC NUMBER</td></tr>
<tr><td>79</td><td colspan="8">Vendor-Reserved</td></tr>
<tr><td rowspan="2">80</td><td colspan="8">Volume Label Flags</td></tr>
<tr><td>VLF_VALID</td><td>VL_EXT</td><td>VL_EBCDIC</td><td colspan="5">Reserved</td></tr>
<tr><td>81</td><td colspan="8">VOLUME LABEL CARTRIDGE TYPE — CHARACTER 2</td></tr>
<tr><td>82</td><td colspan="8">VOLUME LABEL CARTRIDGE TYPE — CHARACTER 1</td></tr>
<tr><td>83
…
88</td><td colspan="8">VOLUME LABEL</td></tr>
<tr><td>89
…
95</td><td colspan="8">Vendor-Reserved</td></tr>
</table>

The following parameters apply:

Byte	Description

0

Bit	Description
7	Valid

Value	Description
0b	information bytes 3-6 are not valid
1b	information bytes 3-6 are valid

Bit	Description
6-0	Response Code

Value	Description
70h	current (non-deferred)
71h	deferred

1 Obsolete: 00h

2

Bit	**Description**
7	Filemark (see Table 12, “Information and ILI Behavior Summary,” on page 41)

Value	**Description**
0b	the current command has not encountered a filemark
1b	the current command has encountered a filemark. This device does not report Setmarks, per Mode Page 10h, byte 8, bit 5.

6 EOM (End-Of-Medium):

Value	**Description**
0b	indicates that the device is not at the end of medium.
1b	indicates that the device is at the end of medium.

5 ILI (Incorrect Length Indicator): see Table 12, “Information and ILI Behavior Summary,” on page 41. ILI residual counts are in the Information field.

4 Reserved

3-0 Sense Key (see Annex B.)

Value	**Description**
0h	NO SENSE
1h	RECOVERED ERROR
2h	NOT READY
3h	MEDIUM ERROR
4h	HARDWARE ERROR
5h	ILLEGAL REQUEST
6h	UNIT ATTENTION
7h	DATA PROTECT
8h	BLANK CHECK
9h	Not Supported
Ah	COPY ABORTED Sense data format for COPY ABORTED (i.e., Sense Key Ah) (see 6.8.3 on page 491) describes the sense data format when this sense key is reported.
Bh	ABORTED COMMAND
Ch	Reserved
Dh	VOLUME OVERFLOW
Eh	Not supported
Fh	COMPLETED

3-6 Information: The content of this field varies depending on the failing command and error. This field is only valid when the Valid bit (byte 0, bit 7) is set to 1. This field is generally only valid for non-deferred errors. See Table 12, “Information and ILI Behavior Summary,” on page 41

7Additional Sense Length (n-7):This device returns:

A) 96 bytes of sense data (a value of 58h in the ADDITIONAL SENSE LENGTH field) in most cases;
B) 18 bytes of sense data (a value of 0Ah in the ADDITIONAL SENSE LENGTH field) in association with ILI conditions for read-type commands, filemark reads, and some protocol errors;
C) 115 bytes of sense data (a value of 6Ch in the ADDITIONAL SENSE LENGTH field) in most cases of COPY ABORTED sense key (see 6.8.3); or
D) other lengths.

8-11 COMMAND-SPECIFIC INFORMATION: 0000 0000h

The 3592 drive does not support the commands associated with this field.

12 ADDITIONAL SENSE CODE (ASC) (see Annex B.)

13 ADDITIONAL SENSE CODE QUALIFIER (ASCQ) (see Annex B.)

14 FIELD REPLACEABLE UNIT CODE (FRU)

This field indicates a possible component or area which is related to the error or failure. Since this device is a single FRU product, this does not necessarily mean a replaceable component has been identified. Instead, this is used for extended fault isolation information.

15-17 Sense Key Specific

When the SKSV bit is 1b (often set when sense key is ILLEGAL REQUEST), bytes 15-17 are interpreted as follows:

Byte	Description
15	Sense Key Specific

Bit	Description
7	SKSV (Sense Key Specific Valid): 1b
6	C/D (Control/Data):

Value	Description
0b	specifies that the error is in a data field of the parameter list.
1b	specifies that the error is in a CDB field.

Bit	Description
5-4	Reserved
3	BPV (Bit Pointer Valid):

Value	Description
0b	specifies that the Bit Pointer Field is not valid.
1b	specifies that the Bit Pointer Field is valid.

Bit	Description
2-0	BIT POINTER When BPV is set to 1b, this field points to the bit in error of the field specified by the FIELD POINTER.

Byte	Description
16-17	FIELD POINTER Points to the CDB byte or parameter byte in error.

When the SKSV bit is 0b, bytes 15-17 are interpreted as follows:

Byte	Description
15	Vendor-Unique Indicators

Bit	Description
7	SKSV (Sense Key Specific Valid): 0b
6-3	TAPE POSITION INDICATORS

Value	Description
0h	BOP (Beginning of Partition)
1h	Data between BOP and LEOP-EW
2h	LEOP-EW (Logical End of Partition - Early Warning)
3h	Data between LEOP-EW and LEOP
4h	LEOP (Logical End of Partition)

Bit	Description
2	Rsvd: Vendor-Reserved
1	PERMANENT

Value	Description
0b	Indicates that the error was recovered
1b	Indicates that the error is permanent

Bit	Description
0	SIM/MIM

Value	Description
0b	Indicates that a SIM/MIM is not available
1b	Indicates that a SIM/MIM is available in Log Page 31

Byte	Description
16-17	FAULT SYMPTOM CODE (FSC) of reporting error.

18

Bit **Description**

7 Rsvd: Vendor-Reserved (Record Sense Request)

6 SMLA (Sequential Medium Loader Active) 0b

This device does not support sequential loaders.

5 DUMP: Microcode Dump Available. This dump is generated by user request or drive check-1.

Value	Description
0b	Indicates there is no microcode dump currently available.
1b	Indicates a microcode dump is available. The microcode dump may be retrieved with the Read Buffer command, and can be used by IBM Service personnel to analyze the state of the device.

NOTE 146 - The dump is lost at the next power off.

4 TDIR_INV (Tape Directory Invalid)

Value	Description
0b	Indicates the Tape Directory is valid
1b	Indicates the Tape Directory is invalid

NOTE 147 - The Tape Directory Invalid field may not be valid if the drive is not ready or when the sense data is associated with a CHECK CONDITION status (contingent allegiance or autosense condition). The Request Sense command should be used to return unsolicited sense to insure the field is valid. An invalid Tape Directory will be automatically rebuilt using a performance optimized method during any space or locate operation. There is no need to fully read the media for directory rebuilding.

3 Rsvd: Vendor-Reserved (Media Test Mode)

2 EOD_PSEG: EOD is in Performance Segment

1 PSEG: Current location is in Performance Segment

0 PORT: Port Reporting Sense

Value	Description
0b	Indicates this sense reported on port 0
1b	Indicates this sense reported on port 1

19 Vendor-Reserved (Drive Identifier)

20 Vendor-Reserved (ACF Status - unsupported)

21 Vendor-Reserved (RAC)

22 Vendor-Reserved (BRAC)

23

Bit **Description**

7-4 MEDIUM ACCESS

Value	Description
0h	Position unknown (during power-on initialization or unusual conditions)
1h	Load error (cartridge loaded, but drive not ready)
2h	Unload error (error occurred while attempting to unload)
3h	Path error (only rewind or unload commands allowed)
4h	Cartridge unloaded or removed
5h	Cartridge is currently loading
6h	Cartridge is currently unloading
9h	Cartridge is loaded

NOTE 148 - Other values are possible, but undefined.

3-0 MEDIUM ASSOCIATION STATE

Value	Description
0h	Medium Unassociated
1h	Medium Associated (loaded and Ready)

24 Medium Descriptor flags

Bit **Description**

7 INIT_REQD (Initialization Required)

Value	**Description**
0b	Medium is initialized.
1b	The medium requires initialization. The device cannot detect initialization on the volume at BOT.

6 HK_INTEG (Housekeeping Integrity Check)

Value	**Description**
0b	No error.
1b	An error condition was detected

5 PARTITIONED

Value	**Description**
0b	The volume does not have multiple partitions.
1b	The volume has multiple partitions.

4 PART_INTEG (Partitioning Integrity)

Value	**Description**
0b	No Error.
1b	An error condition is detected in the partition definition of the volume.

3 MEDIUM_CHK (Medium Check)

Value	**Description**
0b	No error.
1b	A medium check was detected for the medium, and the medium cannot be processed by the device.

2 INCOMP_FMT (Incompatible Format)

Value	**Description**
0b	Format is recognized.
1b	The device does not recognize the format of the medium.

1 PWP (Physical Write Protect)

Value	**Description**
0b	The volume Physical Write Protect switch is off.
1b	The volume Physical Write Protect switch is on.

0 LWP (Logical Write Protect)

Value	**Description**
0b	The volume is not logically write protected.
1b	The volume is logically write protected.

25 Vendor-Reserved (LPOS Region)

26-29 Vendor-Reserved (Error Summary)

30 Failing Command

Request under execution at time of error

31-32 FIRST FAULT SYMPTOM CODE FLAG DATA

Optional encoded flag data for bytes 16-17

33-34 SECOND FAULT SYMPTOM CODE

Second error code encountered

35-36 SECOND FAULT SYMPTOM CODE FLAG DATA

Optional encoded flag data that relates to the SECOND FAULT SYMPTOM CODE

37-38 NEXT-TO-LAST FAULT SYMPTOM CODE

Next-to-last error code encountered

39-40 NEXT-TO-LAST FAULT SYMPTOM CODE FLAG DATA

Optional encoded flag data that relates to the NEXT-TO-LAST FAULT SYMPTOM

41-42 LAST FAULT SYMPTOM CODE

Last error code encountered

43-44 LAST FAULT SYMPTOM CODE FLAG DATA

Optional encoded flag data that relates to the LAST FAULT SYMPTOM

45 Vendor-Reserved (Load Status)

46-49 LOGICAL BLOCK NUMBER

The next block that would be accessed in the forward direction

50-52 DATASET NUMBER

The number of the current physical dataset. Valid values are from 000000h to FFFFFFh

53 PHYSICAL WRAP NUMBER

Physical wrap number relates to actual location on media and may change in a non-incrementing fashion due to segment mapping or other drive features. If the value in this field is FFh, then the physical wrap number exceeds 254.

54 PARTITION INFORMATION

Bit	**Description**
7-4	PARTITIONING TYPE - The type of partitioning currently in use on the volume (see 6.6.13)
3-0	Reserved

55 ACTIVE PARTITION - The partition number of the current logical position.

56-59 RELATIVE LPOS

Longitudinal position relative to beginning of user data (LP3)

60 LOGICAL WRAP NUMBER

Logical wrap number operates in a consistently increasing manner in normal use. If the value in this field is FFh, then the logical wrap number exceeds 254.

61 LAST SPEED INDEX

The last used speed index

62 SARS DRIVE RELATIVE QUALITY

Overall SARS Relative Quality determination of the drive where 00h is unknown, and otherwise ranges from best 01h to worst FFh.

63 SARS MEDIA RELATIVE QUALITY

Overall SARS Relative Quality determination of the currently mounted media where 00h is unknown, and otherwise ranges from best 01h to worst FFh.

64 Vendor-Reserved (Library Address)

65-69 Vendor-Reserved (Recovery Summary)

70Bits 7 and 6 are Cleaning Required Indicators (Static) (see Drive Cleaning Indicators (see 4.12 on page 47))

Bit **Value**

7 M_CLN RQD: Cleaning Required - Normal Maintenance

Value	**Description**
0b	No cleaning is required for normal maintenance
1b	Cleaning is required for normal maintenance

6 T_CLN_RQD: Cleaning Required - Threshold Reached

Value	**Description**
0b	No cleaning is required due to a threshold being reached
1b	Cleaning is required due to a threshold being reached

5 H_DUMP: Hidden dump is present: There was a permanent error and the drive generated a dump

Value	**Description**
0b	A drive generated dump does not exist or has already been read
1b	A drive generated dump exists that has not been read

4 CAL_RQD: Calibration required

Value	**Description**
0b	Calibration is not required
1b	Calibration is required (the drive automatically performs a calibration at the next appropriate time)

3 Reserved

2 CLNR_EXP: Cleaner Expired

Value	**Description**
0b	Cleaner is not expired
1b	Cleaner is expired

1 CLNR_FAIL: Cleaner Fail

Value	**Description**
0b	Cleaner did not fail
1b	Cleaner failed

0 Reserved

71 Vendor-Reserved (Library Error Code)

72-78 MICROCODE EC NUMBER (in ASCII)

79 Vendor-Reserved (Library Drive Status)

80 Volume Label Flags

Bit **Description**

7 VLF_VALID: Volume Label Fields Valid

Value	**Description**
0b	Indicates sense bytes 81-88 are not valid.
1b	Indicates sense bytes 81-88 are valid for the media that is loaded.

6 VL_EXT: Volume Label Source External

Value	**Description**
0b	Indicates the source of sense bytes 81-88 is block 0 of the media.
1b	Indicates the source of sense bytes 81-88 is the cartridge bar code label.

5 VL_EBCDIC: Volume Label is EBCDIC

Value	**Description**
0b	Indicates Volume Label is in ASCII
1b	Indicates Volume Label is in EBCDIC

4-0 Reserved

81 VOLUME LABEL CARTRIDGE TYPE — CHARACTER 2

Value (in ASCII)	**Description**
'A'	3592 Enterprise Tape — Standard Cartridge (JA)
'B'	3592 Enterprise Tape — Extended Cartridge (JB)
‘C’	3592 Enterprise Tape — Advanced Type C Cartridge (JC)
‘D’	3592 Enterprise Tape — Advanced Type D Cartridge (JD)
'J'	3592 Enterprise Tape — Economy Cartridge (JJ)
‘K’	3592 Enterprise Tape — Advanced Type C Economy Cartridge (JK)
‘L’	3592 Enterprise Tape — Advanced Type D Economy Cartridge (JL)
'R'	3592 Enterprise Tape — Economy WORM Cartridge (JR)
'W'	3592 Enterprise Tape — Standard WORM Cartridge (JW)
'X'	3592 Enterprise Tape — Extended WORM Cartridge (JX)
‘Y’	3592 Enterprise Tape — Advanced Type C WORM Cartridge (JY)
‘Z’	3592 Enterprise Tape — Advanced Type D WORM Cartridge (JZ)

82 VOLUME LABEL CARTRIDGE TYPE — CHARACTER 1

Value (in ASCII)	**Description**
'J'	3592 Enterprise Tape Cartridge

83-88 VOLUME LABEL (in ASCII or EBCDIC, depending on the source)

89-95 Vendor-Reserved (Error Summary)

6.8.3 Sense data format for COPY ABORTED (i.e., Sense Key Ah)

The sense data format resulting from a COPY ABORTED sense key (i.e., an aborted copy operation (see 4.28.4.2.1)) is described in table 280.

Table 280 — Sense Data Format for Copy Abort sense key

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0	VALID	RESPONSE CODE (70h)						
1	Obsolete (00h)							
2	FILEMARK (0b)	EOM (0b)	ILI (0b)	Reserved	SENSE KEY (Ah)			
3	SEGMENT RESIDUAL COUNT							
6								
7	ADDITIONAL SENSE LENGTH (n-8)							
8	POINTER TO COPY SOURCE STATUS AND SENSE DATA (00h or s)							
9	POINTER TO COPY DESTINATION STATUS AND SENSE DATA (00h or s)							
10	COPY SEGMENT NUMBER							
11								
12	ADDITIONAL SENSE CODE							
13	ADDITIONAL SENSE CODE QUALIFIER							
14	FIELD REPLACEABLE UNIT CODE							
SKSV (0b)	Vendor-Unique Indicators (i.e., SKSV=0b)							
15	SKSV (0b)	TAPE POSITION INDICATORS				Rsvd	PERMANENT	SIM/MIM
16	FAULT SYMPTOM CODE							
17								
SKSV (1b)	Sense Key Specific (i.e., SKSV=1b)							
15	SKSV (1b)	C/D	Reserved		BPV	BIT POINTER		
16	FIELD POINTER							
17								
r	ADDITIONAL VENDOR UNIQUE SENSE DATA, if any							
…								
s-1								
s	SCSI STATUS OF COPY SOURCE/DESTINATION, if exists							
s+1	COPY SOURCE/DESTINATION SENSE DATA, if exists							
…								
n								

The following parameters apply:

Byte	Description
0	

Bit	Description
7	VALID

Value	Description
0b	SEGMENT RESIDUAL COUNT bytes 3-6 are not valid
1b	SEGMENT RESIDUAL COUNT bytes 3-6 are valid

Bit	Description
6-0	RESPONSE CODE

Value	Description
70h	current (non-deferred)
71h	deferred

Byte	Description
1	Obsolete: 00h
2	

Bit	Description
7	FILEMARK (0b)
6	EOM (End-Of-Medium) (0b)
5	ILI (Incorrect Length Indicator) (0b)
4	Reserved
3-0	SENSE KEY: Ah Sense Data Format for most Sense Keys (see 6.8.2 on page 481) describes the sense data format when this field is not set to Ah.

3-6 SEGMENT RESIDUAL COUNT: The content of this field varies depending on the failing command and error. This field is only valid when the Valid bit (byte 0, bit 7) is set to 1b. See item b) on page 96.

7 ADDITIONAL SENSE LENGTH (n-7)

This form of sense data may include embedded status from another device (i.e., the copy source or the copy destination). It is also possible that there may be Additional Vendor Unique Sense Data prior to such embedded status. All of these fields are included in the ADDITIONAL SENSE LENGTH field such that this embedded sense is considered part of the sense data. In both possible instances of sense data the first 18 bytes comply with the T10/SPC-4 fixed format sense data.

This device returns:

A) 115 bytes of sense data (a value of 6Ch in the ADDITIONAL SENSE LENGTH field) in most cases of COPY ABORTED sense key; or

B) other lengths.

8 POINTER TO COPY SOURCE STATUS AND SENSE DATA: See c) on page 96.

9 POINTER TO COPY DESTINATION STATUS AND SENSE DATA: See d) on page 96.

10-11 COPY SEGMENT NUMBER: See a) on page 96.

12 ADDITIONAL SENSE CODE (ASC) (see Annex B.)

13 ADDITIONAL SENSE CODE QUALIFIER (ASCQ) (see Annex B.)

14 FIELD REPLACEABLE UNIT CODE (FRU)

This field indicates a possible component or area which is related to the error or failure. Since this device is a single FRU product, this does not necessarily mean a replaceable component has been identified. Instead, this is used for extended fault isolation information.

15-17 SENSE KEY SPECIFIC

When the SKSV bit is 1b (often set when SENSE KEY is ILLEGAL REQUEST), bytes 15-17 are interpreted as follows:

Byte	Description
15	SENSE KEY SPECIFIC

Bit	Description
7	SKSV (Sense Key Specific Valid): 1b
6	C/D (Control/Data):

Value	Description
0b	specifies that the error is in a data field of the parameter list.
1b	specifies that the error is in a CDB field.

Bit	Description
5-4	Reserved
3	BPV (Bit Pointer Valid):

Value	Description
0b	specifies that the Bit Pointer Field is not valid.
1b	specifies that the Bit Pointer Field is valid.

Bit	Description
2-0	BIT POINTER When BPV is set to 1b, this field points to the bit in error of the field specified by the FIELD POINTER.

Byte	Description
16-17	CSCD DESCRIPTOR POINTER Points to the CSCD as described in e) on page 97. For example, if the exception condition occurred on the non-copy manager CSCD ID (i.e., 0001h) then this is the offset to the CSCD (i.e., 0030h) as indicated by the formula in 6.9.2.1.2.1 on page 502. Exceptions which occur on the copy manager CSCD ID (i.e., FFFFh) have no location defined in the parameter data and consequently do not set SKSV to 1b.

When the SKSV bit is 0b, bytes 15-17 are interpreted as follows:

Byte	Description
15	Vendor-Unique Indicators

Bit	Description
7	SKSV (Sense Key Specific Valid): 0b
6-0	Reserved

Byte	Description
16-17	REPORTING ERROR FAULT SYMPTOM CODE (FSC)

18-(s-1) ADDITIONAL VENDOR UNIQUE SENSE DATA: This is not typically returned but may be in the future. The POINTER TO COPY SOURCE STATUS AND SENSE DATA and POINTER TO COPY DESTINATION STATUS AND SENSE DATA should be used to determine the status and specific cause of copy failures.

s SCSI STATUS OF COPY SOURCE/DESTINATION: The SCSI status byte delivered to the copy manager by:

A) the copy source if the POINTER TO COPY SOURCE STATUS AND SENSE DATA is non-zero; or

B) the copy destination if the POINTER TO COPY DESTINATION STATUS AND SENSE DATA is non-zero.

(s+1)-n COPY SOURCE/DESTINATION SENSE DATA: The sense data associated with the SCSI status in byte s. This is only returned if the SCSI status is CHECK CONDITION (i.e., 02h). The specific format of this sense data is defined by the copy source/copy destination device.

6.9 XCOPY Parameters

6.9.1 XCOPY parameters overview

XCOPY parameters are used in the EXTENDED COPY (LID4) (i.e., XCOPY) – 83h[01h] (see 5.2.5 on page 110) command to define the functions requested for a copy operation (see 4.28.4.2.1).

6.9.2 XCOPY parameter data

The format of the XCOPY parameter list (see 4.28.4.1) is shown in table 281.

Table 281 — XCOPY parameter list (part 1 of 2)

Bit Byte	7	6	5	4	3	2	1	0
0	PARAMETER LIST FORMAT (01h)							
1	Reserved		STR	LIST ID USAGE		PRIORITY		
2	(MSB)	HEADER CSCD DESCRIPTOR LIST LENGTH (0020h)						
3								(LSB)
4		Reserved						
…								
14								
15	Reserved						G_SENSE	IMMED
16	HEADER CSCD DESCRIPTOR TYPE CODE (FFh)							
17		Reserved						
18								
19								
20	(MSB)	LIST IDENTIFIER						
…								
23								(LSB)
24		Reserved						
…								
41								
42	(MSB)	CSCD DESCRIPTOR LIST LENGTH (n-47)(20h)						
43								(LSB)
44	(MSB)	SEGMENT DESCRIPTOR LIST LENGTH (m-n)(m-79)						
45								(LSB)
46	(MSB)	INLINE DATA LENGTH (0000h)						
47								(LSB)
	CSCD descriptor list							
48		CSCD descriptor [ID 1] (see 6.9.2.1.1)						
…								
n(79)								

Table 281 — XCOPY parameter list (part 2 of 2)

Bit Byte	7	6	5	4	3	2	1	0
	Segment descriptor list							
n+1 (80)	Segment descriptor [first] (see 6.9.2.1.2)							
…								
	⋮							
	Segment descriptor [last] (see 6.9.2.1.2)							
…								
m								

The maximum number of CSCD descriptors permitted within a parameter list is indicated by the MAXIMUM CSCD DESCRIPTOR COUNT field in the Third-party Copy VPD page Parameter Data descriptor (see 6.3.6.4). If the number of CSCD descriptors exceeds the allowed number, the command is terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST, and the additional sense code set to TOO MANY TARGET DESCRIPTORS.

Byte Description

0 PARAMETER LIST FORMAT: 01h

1

Bit Description

7-6 Reserved

5 STR: This bit is ignored

4-3 LIST ID USAGE

Value	**Description**
00b or 10b	The contents of the LIST IDENTIFIER field are defined in 4.28.3.2.2. The list identifier value may be used to abort (see 5.2.3) or to request status for a specific command (e.g., using the RECEIVE COPY STATUS(LID4) command (see 5.2.27)). There is a difference between 00b and 10b related to held data, but this device does not support held data.
01b	Reserved
11b	Only valid if the LIST IDENTIFIER field is zero. Only valid if IMMED bit is set to 0b. The behaviors for held data dictated by this selection is not supported by this device. The behaviors for other segments are.

2-0 PRIORITY: This field is ignored.

2-3 HEADER CSCD DESCRIPTOR LIST LENGTH: 0020h

4-14 Reserved

15

Bit	Description
7-2	Reserved
1	G_SENSE (good with sense data): Specifies if the copy manager is required to return sense data with GOOD status.

Value	Description
0b	When this command completes with GOOD status, the copy manager does not associate sense data with that command.
1b	When this command completes with GOOD status, the copy manager associates sense data with that command and: — sets the sense key and additional sense code fields to COMPLETED, EXTENDED COPY INFORMATION AVAILABLE (F/0020), and — sets the COMMAND-SPECIFIC INFORMATION field to the number of segment descriptors the copy manager has processed. Only valid if IMMED is 0b.

0	IMMED (immediate): Specifies whether the copy manager returns status before the copy operation is started (see 4.28.4.2). Processing of the IMMED bit is described in 4.28.4.2.2.

Value	Description
0b	The command is completely processed before status is returned.
1b	Status is returned after qualifying the CDB and Parameter Data structures as described in 4.28.4.2.2. Only valid if G_SENSE is 0b. Only valid if LIST ID USAGE is not 11b.

16 HEADER CSCD DESCRIPTOR TYPE CODE: FFh

17-19 Reserved

20-23 LIST IDENTIFIER: The copy operation originated by the XCOPY command to the copy manager as described in 4.28.3.2.2. The LIST ID USAGE field specifies the usage of the LIST IDENTIFIER field.

24-41 Reserved

42-43 CSCD DESCRIPTOR LIST LENGTH: The length in bytes of the CSCD descriptor list that follows the parameter list header. Each CSCD other than the copy manager is described by a CSCD descriptor (see 6.9.2.1.1).

44-45 SEGMENT DESCRIPTOR LIST LENGTH: The length in bytes of the segment descriptor list that follows the CSCD descriptors (see 6.9.2.1.1.1). See 6.9.2.1.2 for a detailed description of the segment descriptors.

The maximum number of segment descriptors permitted within a parameter list is indicated by the MAXIMUM SEGMENT DESCRIPTOR COUNT field in the Third-party Copy VPD page Parameter Data descriptor (see 6.3.6.4). If the number of segment descriptors exceeds the allowed number, the command is terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST, and the additional sense code set to TOO MANY SEGMENT DESCRIPTORS.

The maximum combined length of the CSCD descriptors and segment descriptors permitted within a parameter list is indicated by the MAXIMUM DESCRIPTOR LIST LENGTH field in the Third-party Copy VPD page Parameter Data descriptor (see 6.3.6.4). If the combined length of the CSCD descriptors and segment descriptors exceeds the allowed value, the command is terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST, and the additional sense code set to PARAMETER LIST LENGTH ERROR.

46-47 INLINE DATA LENGTH: 0000h

Inline data is not supported. If this field is not set to zero the XCOPY command will be terminated with an ILLEGAL REQUEST, INVALID FIELD IN PARAMETER LIST (5/2600).

48-
n(79) CSCD descriptors: See ECD E0h to FEh: CSCD descriptors (see 6.9.2.1.1 on page 498). The copy source or the copy destination is required to be the copy manager which has an implicit identifier not included in the CSCD descriptor list. Therefore, only the device that does not contain the copy manager has a CSCD descriptor listed.

n+1(80)
- m Segment descriptors: See ECD 00h to BFh: Segment descriptors (see 6.9.2.1.2 on page 502).

6.9.2.1 Descriptor type codes

Descriptor type codes are described in:

a) ECD E0h to FEh: CSCD descriptors (see 6.9.2.1.1 on page 498); and
b) ECD 00h to BFh: Segment descriptors (see 6.9.2.1.2 on page 502).

6.9.2.1.1 ECD E0h to FEh: CSCD descriptors

6.9.2.1.1.1 CSCD descriptors introduction

A CSCD descriptor contains information that identifies a copy source or a copy destination. The descriptor type codes for CSCD descriptors are described in:

a) ECD E0h: Fibre Channel N_Port_Name CSCD descriptor format (see 6.9.2.1.1.2 on page 499);
b) ECD E2h: Fibre Channel N_Port_ID With N_Port_Name Checking CSCD descriptor format (see 6.9.2.1.1.3 on page 500); and
c) ECD E4h: Identification Descriptor CSCD descriptor format (see 6.9.2.1.1.4 on page 501).

If a copy manager receives an unsupported descriptor type code in a CSCD descriptor, then the copy operation (see 4.28.4.2.1) originated by the command is terminated with ILLEGAL REQUEST, UNSUPPORTED TARGET DESCRIPTOR TYPE CODE (5/2607).

6.9.2.1.1.2 ECD E0h: Fibre Channel N_Port_Name CSCD descriptor format

The CSCD descriptor format shown in table 282 is used by an XCOPY command to specify an FCP CSCD using its Fibre Channel N_Port_Name.

Table 282 — Fibre Channel N_Port_Name CSCD descriptor format

Bit Byte	7	6	5	4	3	2	1	0
0	DESCRIPTOR TYPE CODE (E0h)							
1	LU ID TYPE (00B)		Obsolete	PERIPHERAL DEVICE TYPE (1H)				
2	(MSB)	RELATIVE INITIATOR PORT IDENTIFIER						
3								(LSB)
4	LU IDENTIFIER							
…								
11								
12	N_PORT_NAME							
…								
19								
20	Reserved							
…								
27								
28	Reserved					PAD (0B)	Reserved	FIXED(0b)
29	(MSB)	STREAM BLOCK LENGTH (000000H)						
30								
31								(LSB)

Byte Description

0 DESCRIPTOR TYPE CODE: E0h

1

Bit	**Description**
7-6	LU ID TYPE: 00b (i.e., the LU IDENTIFIER field, if any, contains a Logical Unit Number)
5	Obsolete
4-0	PERIPHERAL DEVICE TYPE: 01h (i.e., tape).

2-3 RELATIVE INITIATOR PORT IDENTIFIER: The relative port identifier of the initiator port the copy manager is instructed to use to access the CSCD.

Value	**Description**
0h	Any initiator port.
1h	Port 0
2h	Port 1

4-11 LU IDENTIFIER: The logical unit within the CSCD descriptor that is the copy source or copy destination.

12-19 N_PORT_NAME: The N_Port_Name defined by the port login (PLOGI) extended link service.

20-27 Reserved

28

Bit	**Description**
7-3	Reserved
2	PAD: 0b
1	Reserved
0	FIXED: 0b

29-31 STREAM BLOCK LENGTH: 000000h

6.9.2.1.1.3 ECD E2h: Fibre Channel N_Port_ID With N_Port_Name Checking CSCD descriptor format

The CSCD descriptor format shown in table 283 is used by an XCOPY command to specify an FCP CSCD using its Fibre Channel N_Port_ID and to require the copy manager to verify that the N_Port_Name of the specified N_Port matches the value in the CSCD descriptor.

Table 283 — Fibre Channel N_Port_ID With N_Port_Name Checking CSCD descriptor format

Bit Byte	7	6	5	4	3	2	1	0
0	DESCRIPTOR TYPE CODE (E2h)							
1	LU ID TYPE (00B)		Obsolete	PERIPHERAL DEVICE TYPE (1H)				
2	(MSB)	RELATIVE INITIATOR PORT IDENTIFIER						
3								(LSB)
4	LU IDENTIFIER							
…								
11								
12	N_PORT_NAME							
…								
19								
20	Reserved							
21	(MSB)	N_PORT_ID						
22								
23								(LSB)
24	Reserved							
…								
27								
28	Reserved					PAD (0B)	Reserved	FIXED(0b)
29	(MSB)	STREAM BLOCK LENGTH (000000H)						
30								
31								(LSB)

Byte	**Description**
0	DESCRIPTOR TYPE CODE: E2h
1	

Bit	**Description**
7-6	LU ID TYPE: 00b (i.e., the LU IDENTIFIER field, if any, contains a Logical Unit Number)
5	Obsolete
4-0	PERIPHERAL DEVICE TYPE: 01h (i.e., tape).

2-3	RELATIVE INITIATOR PORT IDENTIFIER: The relative port identifier of the initiator port the copy manager is instructed to use to access the CSCD.

Value	Description
0h	Any initiator port.
1h	Port 0
2h	Port 1

4-11	LU IDENTIFIER: The logical unit within the CSCD descriptor that is the copy source or copy destination.
12-19	N_PORT_NAME: The N_Port_Name defined by the port login (PLOGI) extended link service.
20	Reserved
21-23	N_PORT_ID: The port D_ID to use to communicate with the CSCD.
24-27	Reserved
28	

Bit	Description
7-3	Reserved
2	PAD: 0b
1	Reserved
0	FIXED: 0b

29-31	STREAM BLOCK LENGTH: 000000h

6.9.2.1.1.4 ECD E4h: Identification Descriptor CSCD descriptor format

The CSCD descriptor format shown in table 284 instructs the copy manager to locate a SCSI target device and logical unit that returns a Device Identification VPD page (see 6.3.4) containing an Identification descriptor having the CODE SET field set to 1h and the DESIGNATOR TYPE field set to 3h with matching values in the DESIGNATOR LENGTH field, and THE DESIGNATOR field (This designator field is the WWNN or WWPN descriptor (08h bytes)). The contents of bytes 4-27 are the NAA LUN or port designator from Inquiry page 83h or Inquiry page 88h of the device indicated by this CSCD.

Table 284 — Identification Descriptor CSCD descriptor format

Bit Byte	7	6	5	4	3	2	1	0
0	DESCRIPTOR TYPE CODE (E4h)							
1	LU ID TYPE		Obsolete	PERIPHERAL DEVICE TYPE (1H)				
2	(MSB)	RELATIVE INITIATOR PORT IDENTIFIER						
3								(LSB)
4	Reserved				CODE SET (1h)			
5	Reserved		ASSOCIATION		DESIGNATOR TYPE (3h)			
6	Reserved							
7	DESIGNATOR LENGTH (08h)							
8	DESIGNATOR							
…								
27								
28	Reserved					PAD (0B)	Reserved	FIXED(0b)
29	(MSB)	STREAM BLOCK LENGTH (000000H)						
30								
31								(LSB)

Byte	Description
0	DESCRIPTOR TYPE CODE: E4h

1

Bit	Description
7-6	LU ID TYPE: Ignored
5	Obsolete
4-0	PERIPHERAL DEVICE TYPE: 01h (i.e., tape).

2-3 RELATIVE INITIATOR PORT IDENTIFIER: The relative port identifier of the initiator port the copy manager is instructed to use to access the CSCD.

Value	Description
0h	Any initiator port.
1h	Port 0
2h	Port 1

4

Bit	Description
7-4	Reserved
3-0	CODE SET: 1h

5

Bit	Description
7-6	Reserved
5-4	ASSOCIATION:
3-0	DESIGNATOR TYPE: 3h (i.e., NAA)

ASSOCIATION:

Value	Description
00b	DESIGNATOR must be the 8 byte WWNN
01b	DESIGNATOR must be the 8 byte WWPN
10b	DESIGNATOR must be the 8 byte WWPN

6 Reserved

7 DESIGNATOR LENGTH: 8h

8-27 DESIGNATOR: Zero-padded DESIGNATOR of the CSCD as indicated by the ASSOCIATION field.

28

Bit	Description
7-3	Reserved
2	PAD: 0b
1	Reserved
0	FIXED: 0b

29-31 STREAM BLOCK LENGTH: 000000h

6.9.2.1.2 ECD 00h to BFh: Segment descriptors

6.9.2.1.2.1 Segment descriptors introduction

The descriptor type code values assigned to segment descriptors are shown in table 285.

Table 285 — XCOPY segment descriptor type codes (part 1 of 2)

Descriptor type code [a]	Reference	Description
07h	6.9.2.1.2.2	Verify CSCD
10h	6.9.2.1.2.3	Write filemarks to sequential-access device
11h	6.9.2.1.2.4	Space records or filemarks on sequential-access device
12h	6.9.2.1.2.5	Locate on sequential-access device
13h	6.9.2.1.2.6	Tape device image copy

[a] All supported segment descriptor types are listed in the Third-party Copy VPD page Supported Descriptors descriptor (see 6.3.6.5)

Table 285 — XCOPY segment descriptor type codes (part 2 of 2)

Descriptor type code_[a]	Reference	Description
14h	6.9.2.1.2.7	Register persistent reservation key
15h	6.9.2.1.2.8	Third party persistent reservations source I_T nexus
17h	6.9.2.1.2.9	Positioning on sequential-access device
18h	6.9.2.1.2.10	Tape device logical object copy
others		N/A

[a] All supported segment descriptor types are listed in the Third-party Copy VPD page Supported Descriptors descriptor (see 6.3.6.5)

Segment descriptors begin with an eight byte header, the format of which is shown in table 286. The header is repeated in each of the segment descriptors below showing the specific setting of each of the header fields.

Table 286 — Segment descriptor header

Bit Byte	7	6	5	4	3	2	1	0
0	DESCRIPTOR TYPE CODE							
1	Reserved						DC (0b)	CAT (0b)
2	(MSB)	DESCRIPTOR LENGTH (n-3)						
3								(LSB)
4	(MSB)	SOURCE CSCD DESCRIPTOR ID						
5								(LSB)
6	(MSB)	DESTINATION CSCD DESCRIPTOR ID						
7								(LSB)
8		Segment descriptor parameters						
…								
n								

Byte	Description
0	DESCRIPTOR TYPE CODE: If an unsupported descriptor type code is received in this field the copy operation (see 4.28.4.2.1) is terminated with ILLEGAL REQUEST, UNSUPPORTED SEGMENT DESCRIPTOR TYPE CODE (5/2609).
1	The two defined bits in this byte are treated as follows in all descriptors supported by this device.

Bit	Description
7-2	Reserved
1	DC: 0b - This bit should be set to 0b but is ignored.
0	CAT: 0b - This bit should be set to 0b but is ignored.

Byte	Description
2-3	DESCRIPTOR LENGTH: Length of descriptor that follows.

4-5 SOURCE CSCD DESCRIPTOR ID: The location of the CSCD descriptor that specifies the copy source. The list of CSCD descriptor ID values is the same for both this field, (i.e., SOURCE CSCD DESCRIPTOR ID) and the DESTINATION CSCD DESCRIPTOR ID field in bytes 6 and 7. These values are:

Code	**Description**
0000h	This value must be exclusively used for reserved CSCD descriptor ID fields.
0001h to 07FFh	The copy source or copy destination is specified by the contents of the CSCD descriptor whose location in the XCOPY command parameter list (see 4.28.4.1) is computed as follows: 16 + (code x 32)
FFFFh	The copy source or copy destination is the logical unit that contains the copy manager that is processing the XCOPY command (i.e., the logical unit to which the XCOPY command was sent).

Since this device supports at most one CSCD in the CSCD descriptor list, 0001h and FFFFh are the only values which may be used for the SOURCE CSCD DESCRIPTOR ID or the DESTINATION CSCD DESCRIPTOR ID.

If the CSCD specified in this field is not accessible, the XCOPY command is terminated with COPY ABORTED, UNREACHABLE COPY TARGET (A/0D02).

6-7 DESTINATION CSCD DESCRIPTOR ID: The location of the CSCD descriptor that specifies the copy destination. For a list of values, see the SOURCE CSCD DESCRIPTOR ID field.

If the CSCD specified in this field is accessible, the XCOPY command is terminated with COPY ABORTED, UNREACHABLE COPY TARGET (A/0D02).

8-n Segment descriptor parameters as shown in each specific segment descriptor description.

6.9.2.1.2.2 ECD 07h: Verify CSCD function

The segment descriptor format shown in table 287 instructs the copy manager to verify the accessibility of a CSCD.

Table 287 — Verify CSCD segment descriptor

Bit **Byte**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	DESCRIPTOR TYPE CODE (07h)							
1	Reserved							
2	(MSB)	DESCRIPTOR LENGTH (0008h)						
3								(LSB)
4	(MSB)	SOURCE CSCD DESCRIPTOR ID						
5								(LSB)
6	Reserved							
7								
8	Reserved							TUR
9	Reserved							
…								
11								

Byte	**Description**
0	DESCRIPTOR TYPE CODE: 07h
1	Reserved
2-3	DESCRIPTOR LENGTH: 0008h
4-5	SOURCE CSCD DESCRIPTOR ID: See 6.9.2.1.2.1 on page 502.
6-7	Reserved.

8

Bit	Description
7-1	Reserved
0	TUR

Value	Description
0b	Accessibility is verified using an INQUIRY command.
1b	Accessibility is verified using a TEST UNIT READY command.

9-11 Reserved.

6.9.2.1.2.3 ECD 10h: Write filemarks function

The segment descriptor format shown in table 288 instructs the copy manager to write filemarks on the destination tape device using a WRITE FILEMARKS command (see 5.2.52).

Table 288 — Write filemarks segment descriptor

Bit Byte	7	6	5	4	3	2	1	0
0	DESCRIPTOR TYPE CODE (10h)							
1	Reserved							
2	(MSB)	DESCRIPTOR LENGTH (0008h)						
3								(LSB)
4		Reserved						
5								
6	(MSB)	DESTINATION CSCD DESCRIPTOR ID						
7								(LSB)
8	Reserved						Obsolete	W_IMMED
9	(MSB)	FILEMARK COUNT						
10								
11								(LSB)

Byte Description

0 DESCRIPTOR TYPE CODE: 10h

1 Reserved

2-3 DESCRIPTOR LENGTH: 0008h

4-5 Reserved

6-7 DESTINATION CSCD DESCRIPTOR ID: See 6.9.2.1.2.1 on page 502.

8

Bit	Description
7-2	Reserved
1	Obsolete
0	W_IMMED (write immediate)

Value	Description
0b	Issue a non-immediate WRITE FILEMARKS command (i.e., IMMED = 0b)
1b	Issue an immediate WRITE FILEMARKS command (i.e., IMMED = 1b)

9-11 FILEMARK COUNT: The value to be set in the FILEMARK COUNT field of the WRITE FILEMARKS command.

6.9.2.1.2.4 ECD 11h: Space function

The segment descriptor format shown in table 289 instructs the copy manager to send a SPACE command (see 5.2.43) to the destination tape device.

Table 289 — Space segment descriptor

Bit Byte	7	6	5	4	3	2	1	0
0	DESCRIPTOR TYPE CODE (11h)							
1	Reserved							
2	(MSB)	DESCRIPTOR LENGTH (0008h)						
3								(LSB)
4	Reserved							
5								
6	(MSB)	DESTINATION CSCD DESCRIPTOR ID						
7								(LSB)
8	Reserved					CODE		
9	(MSB)	COUNT						
10								
11								(LSB)

Byte	Description
0	DESCRIPTOR TYPE CODE: 11h
1	Reserved
2-3	DESCRIPTOR LENGTH: 0008h
4-5	Reserved
6-7	DESTINATION CSCD DESCRIPTOR ID:See 6.9.2.1.2.1 on page 502.
8	

Bit	Description
7-3	Reserved
2-0	CODE: Value to set in the CODE field of the SPACE command.

Value	Description
000b	Blocks
001b	Filemarks
011b	End of Data
others	Not supported

9-11	COUNT: Value to set in the COUNT field of the SPACE command.

6.9.2.1.2.5 ECD 12h: Locate function

The segment descriptor format shown in table 290 instructs the copy manager to send a LOCATE command (see 5.2.11) to the destination tape device to position to a logical object identifier in the current partition.

Table 290 — Locate segment descriptor

Bit Byte	7	6	5	4	3	2	1	0
0	DESCRIPTOR TYPE CODE (12h)							
1	Reserved							
2	(MSB)	DESCRIPTOR LENGTH (0008h)						
3								(LSB)
4	Reserved							
5								
6	(MSB)	DESTINATION CSCD DESCRIPTOR ID						
7								(LSB)
8	(MSB)	LOGICAL OBJECT IDENTIFIER						
…								
11								(LSB)

Byte	Description
0	DESCRIPTOR TYPE CODE:
1	Reserved
2-3	DESCRIPTOR LENGTH: 0008h
4-5	Reserved
6-7	DESTINATION CSCD DESCRIPTOR ID: See 6.9.2.1.2.1 on page 502.
8-11	LOGICAL OBJECT IDENTIFIER: The value to set in the LOGICAL OBJECT IDENTIFIER field of the LOCATE command sent to the copy destination tape device

6.9.2.1.2.6 ECD 13h: Tape device image copy function

The segment descriptor format shown in table 291 instructs the copy manager to perform an image copy from the copy source tape device to the copy destination tape device.

When filemarks are copied they are written to the destination as buffered filemarks (i.e., IMMED=1).

The tape image copy operation (see 4.28.4.2.1) terminates when:

a) the copy source encounters an end-of-partition as defined by the copy source;
b) the copy source encounters an end-of-data as defined by the copy source (i.e., BLANK CHECK sense key); or
c) the copy manager has copied the number of consecutive filemarks specified in the COUNT field from the copy source to the copy destination.

Table 291 — Tape device image copy segment descriptor

Bit Byte	7	6	5	4	3	2	1	0
0	DESCRIPTOR TYPE CODE (13h)							
1	Reserved							
2	(MSB)	DESCRIPTOR LENGTH (0008h)						
3								(LSB)
4	(MSB)	SOURCE CSCD DESCRIPTOR ID						
5								(LSB)
6	(MSB)	DESTINATION CSCD DESCRIPTOR ID						
7								(LSB)
8	(MSB)	COUNT						
…								
11								(LSB)

Byte	Description
0	DESCRIPTOR TYPE CODE:
1	Reserved
2-3	DESCRIPTOR LENGTH: 0008h
4-5	SOURCE CSCD DESCRIPTOR ID: See 6.9.2.1.2.1 on page 502.
6-7	DESTINATION CSCD DESCRIPTOR ID: See 6.9.2.1.2.1 on page 502.
8-11	COUNT: The number of consecutive filemarks which, when encountered, specify that the copy function is to be terminated.

Value	Description
zero	The number of consecutive filemarks copied have no bearing on when the copy operation is terminated.
non-zero	The copy operation is terminated when this number of consecutive filemarks have been copied.

6.9.2.1.2.7 ECD 14h: Register persistent reservation key function

The segment descriptor format shown in table 292 instructs the copy manager to register an I_T nexus using the reservation key specified by the RESERVATION KEY field with the logical unit specified by the DESTINATION CSCD DESCRIPTOR ID field.

After processing of other segments stops regardless of the status, the registration performed as a result of this segment descriptor are unregistered. Errors encountered during the processing to unregister are not reported in the status of the copy operation.

Table 292 — Register persistent reservation key segment descriptor

Bit Byte	7	6	5	4	3	2	1	0
0	DESCRIPTOR TYPE CODE (14h)							
1	Reserved							
2	(MSB)	DESCRIPTOR LENGTH (0018h)						
3								(LSB)
4	Reserved							
5								
6	(MSB)	DESTINATION CSCD DESCRIPTOR ID						
7								(LSB)
8	(MSB)	RESERVATION KEY						
…								
15								(LSB)
16	(MSB)	SERVICE ACTION RESERVATION KEY						
…								
23								(LSB)
24	Reserved							
…								
27								

Byte	Description
0	DESCRIPTOR TYPE CODE: 14h
1	Reserved
2-3	DESCRIPTOR LENGTH: 0018h
4-5	Reserved
6-7	DESTINATION CSCD DESCRIPTOR ID: See 6.9.2.1.2.1 on page 502. Specifying the CSCD that contains the copy manager is not allowed.
8-15	RESERVATION KEY: The value the copy manager is to use in the PERSISTENT RESERVE OUT command used to register the copy destination.
16-23	SERVICE ACTION RESERVATION KEY: The value the copy manager is to use in the PERSISTENT RESERVE OUT command used to register the copy destination.

6.9.2.1.2.8 ECD 15h: Third party persistent reservations source I_T nexus function

The third party persistent reservations source I_T nexus function segment descriptor (see table 293) instructs the copy manager to send a PERSISTENT RESERVE OUT command with REGISTER AND MOVE service action (see 5.2.17) with the specified I_T nexus after all other segment descriptors have been processed. Multiple third party persistent reservations source I_T nexus function segment descriptors may be included in an XCOPY command. When this descriptor is encountered during a copy operation (see 4.28.4.2.1), no action is taken, and the segment is deferred for processing at the end of the copy operation. After processing of other segments stops regardless of the status, these deferred segments shall be processed in the order in which they were encountered. Errors encountered during the processing of these deferred segments are not reported in the status of the copy operation.

This segment descriptor should be placed at or near the beginning of the list of segment descriptors to assure the copy manager sends the PERSISTENT RESERVE OUT command with REGISTER AND MOVE service action in the event of an error that terminates the processing of segment descriptors. If an error is detected in a segment descriptor and the third party persistent reservations source I_T nexus segment descriptor has not been processed, then the copy manager does not send a PERSISTENT RESERVE OUT command with REGISTER AND MOVE service action.

Table 293 — Third party persistent reservations source I_T nexus segment descriptor

Bit Byte	7	6	5	4	3	2	1	0
0	DESCRIPTOR TYPE CODE (15h)							
1	Reserved							
2	(MSB)	DESCRIPTOR LENGTH (n-3)						
3								(LSB)
4	Reserved							
5								
6	(MSB)	DESTINATION CSCD DESCRIPTOR ID						
7								(LSB)
8	(MSB)	RESERVATION KEY						
…								
15								(LSB)
16	(MSB)	SERVICE ACTION RESERVATION KEY						
…								
23								(LSB)
24	Reserved							
25	Reserved						UNREG	APTPL
26	(MSB)	RELATIVE TARGET PORT IDENTIFIER						
27								(LSB)
28	(MSB)	TRANSPORTID LENGTH (n-31)						
…								
31								(LSB)
32	TRANSPORTID							
…								
n								

Byte	**Description**
0	DESCRIPTOR TYPE CODE: 15h
1	Reserved
2-3	DESCRIPTOR LENGTH: The length in bytes that follow in the segment descriptor. This is required to be a multiple of four.
4-5	Reserved
6-7	DESTINATION CSCD DESCRIPTOR ID: See 6.9.2.1.2.1 on page 502.
8-15	RESERVATION KEY: The value the copy manager is to use in the PERSISTENT RESERVE OUT command.
16-23	SERVICE ACTION RESERVATION KEY: The value the copy manager is to use in the PERSISTENT RESERVE OUT command.

24 Reserved

25

Bit	Description
7-2	Reserved
1	UNREG: The value the copy manager is to use in the PERSISTENT RESERVE OUT command.
0	APTPL: 0b. The value the copy manager is to use in the PERSISTENT RESERVE OUT command.

26-27 RELATIVE TARGET PORT IDENTIFIER: The value the copy manager is to use in the PERSISTENT RESERVE OUT command.

28-31 TRANSPORTID LENGTH (n-23): The value the copy manager is to use in the PERSISTENT RESERVE OUT command.

32-n TRANSPORTID: The value the copy manager is to use in the PERSISTENT RESERVE OUT command.

For a description of the RESERVATION KEY field, SERVICE ACTION RESERVATION KEY field, UNREG bit, APTPL bit, RELATIVE TARGET PORT IDENTIFIER field, TRANSPORTID LENGTH field, and TRANSPORTID, see 5.2.17.

6.9.2.1.2.9 ECD 17h: Positioning function

The segment descriptor format shown in table 294 instructs the copy manager to position the destination tape device by sending appropriate commands (e.g., SPACE(6), SPACE(16), LOCATE(10), LOCATE(16)). Some devices indicated by a CSCD may not support some of the locate mechanisms needed to perform certain actions that may be requested with these commands. For such conditions, the segment operation will fail with the status returned by such a device.

Table 294 — Positioning segment descriptor

Bit Byte	7	6	5	4	3	2	1	0
0	DESCRIPTOR TYPE CODE (17h)							
1	Reserved							
2	(MSB)	DESCRIPTOR LENGTH (0010h)						
3								(LSB)
4	Reserved							
5								
6	(MSB)	DESTINATION CSCD DESCRIPTOR ID						
7								(LSB)
8	CP	Reserved				POSITIONING TYPE		
9	PARTITION							
10	Reserved							
11								
12	(MSB)	LOGICAL IDENTIFIER						
…								
19								(LSB)

Byte	Description
0	DESCRIPTOR TYPE CODE: 17h
1	Reserved
2-3	DESCRIPTOR LENGTH: 10h
4-5	Reserved
6-7	DESTINATION CSCD DESCRIPTOR ID: See 6.9.2.1.2.1 on page 502.

8

Bit **Description**

7 CP (change partition):

Value	Description
0b	Do not change the partition and ignore the PARTITION field.
1b	Change to the partition specified in the PARTITION field prior to positioning to the logical object or logical file, as specified in the LOGICAL IDENTIFIER field. If the POSITIONING TYPE field is set to 000b or 001b, then the copy manager terminates the copy operation (see 4.28.4.2.1) originated by the command with ILLEGAL REQUEST, INVALID FIELD IN PARAMTER LIST (5/2600) during initial parameter checking.

6-3 Reserved

2-0 POSITIONING TYPE: Specifies how to interpret the LOGICAL IDENTIFIER field.

Value	Description
000b	Relative by logical block: Logical position upon successful completion is the EOP side of the number of logical blocks specified in the LOGICAL IDENTIFIER field. The LOGICAL IDENTIFIER field contains a value in two’s complement notation specifying the number of logical blocks to be spaced over and the direction of movement.
001b	Relative by logical file: Logical position upon successful completion is the EOP side of the number of logical files specified in the LOGICAL IDENTIFIER field. The LOGICAL IDENTIFIER field contains a value in two’s complement notation specifying the number of logical files to be spaced over and the direction of movement.
010b	Absolute logical object: Logical position upon successful completion is the BOP side of the logical object specified in the LOGICAL IDENTIFIER field. The LOGICAL IDENTIFIER field contains the absolute logical object identifier of the location at which to position the medium.
011b	Absolute logical file: Logical position upon successful completion is the BOP side of the logical file specified in the LOGICAL IDENTIFIER field. The LOGICAL IDENTIFIER field contains the absolute logical object identifier of the location at which to position the medium.
100b	End-of-data: Logical position upon successful completion is EOD. If the POSITIONING TYPE field is set to 100b (i.e., end-of-data), then the LOGICAL IDENTIFIER field is ignored.
101b to 111b	Reserved

9 PARTITION: The partition to select if the CP bit is set to 1b.

10-11 Reserved

12-19 LOGICAL IDENTIFIER: See the POSITIONING TYPE field for how to interpret the LOGICAL IDENTIFIER field.

6.9.2.1.2.10 ECD 18h: Tape device logical object copy function

The segment descriptor format shown in table 295 instructs the copy manager to copy logical objects from the current position of the source tape device to the current position of the destination tape device. When filemarks are copied they are written to the destination as buffered filemarks (i.e., IMMED=1).

Table 295 — Tape device logical object copy segment descriptor

Bit Byte	7	6	5	4	3	2	1	0
0	DESCRIPTOR TYPE CODE (18h)							
1	Reserved							
2	(MSB)	DESCRIPTOR LENGTH (0010h)						
3								(LSB)
4	(MSB)	SOURCE CSCD DESCRIPTOR ID						
5								(LSB)
6	(MSB)	DESTINATION CSCD DESCRIPTOR ID						
7								(LSB)
8	CTEOM	EOEOD	Reserved			CODE		
9	Reserved							
11								
12	(MSB)	NUMBER OF OBJECTS						
…								
19								(LSB)

Byte	Description
0	DESCRIPTOR TYPE CODE: 18h
1	Reserved
2-3	DESCRIPTOR LENGTH: 0010h
4-5	SOURCE CSCD DESCRIPTOR ID: See 6.9.2.1.2.1 on page 502.
6-7	DESTINATION CSCD DESCRIPTOR ID: See 6.9.2.1.2.1 on page 502.

8

Bit	**Description**
7	CTEOM (continue through end-of-medium)

Value	**Description**
0b	This function is terminated as a result of the copy destination device returning an EOM bit set to one.
1b	This function is not terminated as a result of the copy destination device returning an EOM bit set to one and the sense key set to NO SENSE or BLANK CHECK. Segment processing of this segment continues.

6	EOEOD (error on end-of-data)

Value	**Description**
0b	This function is not terminated with an error as a result of encountering EOD and segment processing continues with the next segment.
1b	This function terminates with an error if EOD is encountered while reading from the source device.

5-3	Reserved
2-0	CODE: Used in conjunction with the NUMBER OF OBJECTS field to indicate the objects to be copied.

Value	**Interpretation of the NUMBER OF OBJECTS field**
000b	Logical objects: The NUMBER OF OBJECTS field specifies the number of logical objects to copy. If the NUMBER OF OBJECTS field is set to zero, then no logical objects are copied.
001b	Logical blocks: The NUMBER OF OBJECTS field specifies the number of logical blocks to copy. If a filemark is encountered, then the filemark is copied and the function terminates with a filemark encountered indication. If the NUMBER OF OBJECTS field is set to zero, then no logical blocks are copied.
010b	Logical files: The NUMBER OF OBJECTS field specifies the number of filemarks through which to copy. All logical objects are copied through the number of filemarks specified in the NUMBER OF OBJECTS field. If the NUMBER OF OBJECTS field is set to zero, then no logical files are copied.
011b	All logical objects to EOD: All logical objects starting at the current position of the source device and ending at EOD are copied to the destination device. If the CODE field is set to 011b (i.e., all logical objects to EOD), then the NUMBER OF OBJECTS field is ignored.
100b to 111b	Reserved

9-11 Reserved

12-19 NUMBER OF OBJECTS: Used in conjunction with the CODE field to indicate the objects to be copied. See the description under the CODE field.

Annex A. Protocol Implementation Notes

This product is comprised of one or more SCSI-2 compliant devices with some important SCSI-3 extensions (deviations are noted). This appendix describes some of the specific implementation choices made within the SCSI architecture under the following headings:

- “Supported SCSI Status Codes” on page 515

Throughout this appendix, the drive is also called “target” and “device.”

A.1. Supported SCSI Status Codes

The SCSI status codes are defined in the SCSI-2 standard. The 3592 drive does not use all available status codes. However, the 3592 complies with the SCSI standard for all status codes that it supports. The list of status codes and their use in the drive follows:

Status	**Description**
00h	GOOD Used on the last command of any nexus when the last command finishes correctly. Since command linking is not supported, there may only be one commands per nexus.
02h	CHECK CONDITION Used to report any error condition that generates a contingent allegiance for the command. The device prepares sense data for the event and reports Auto-Sense information with the CHECK CONDITION status. A REQUEST SENSE command is not required, and a contingent allegiance does not exist.
18h	RESERVATION CONFLICT Used when an initiator not holding a current reservation attempts to execute an unauthorized command while a reservation is in effect.
08h	BUSY Used when required by SCSI-2 (for example, contingent allegiance). The dual port nature of the drive makes the SCSI-3 behavior extend the BUSY status to a second port. No unnecessary BUSY status is presented. However, some BUSY status reports are required and are presented. BUSY status may be presented during the power-up sequence until the diagnostics are complete. Under ordinary circumstances, this is the most common reason for encountering BUSY status.
28h	TASK SET FULL This is used when the maximum number of commands are currently queued for execution. This should not generally be presented, but may be used when a large number of initiators are present, or multiple commands are issued from single initiators using the simple queueing model.

A.2. Features of the Fibre Channel Interface

The 3592 J1A, E05, E06, E07, and 55F support Fibre Channel Arbitrated Loop (FC-AL) protocol, and uses Class 3 Service frames. The drive supports operating as a public (switch-attached) or private device (that is, L-Port to FL-Port; or L-Port to L-Port). The 3592 60F does not support FC-AL.

The 3592 drive can also attach using the point-to-point protocol (also known as an N-Port). When operating in the point-to-point protocol, the drive can attach in a Fabric topology (that is, N-Port to F-Port). The 60F supports attaching point-to-point protocol directly between N-ports (that is, N-port to N-port). Earlier models can be attached N-port to N-port when explicitly configured; however there is no support claimed.

The World Wide Node Name and Port Name that are used follow the format of the Institute of Electrical and Electronics Engineers (IEEE). The IBM 3592 Tape Drive is compliant with the FC-Tape Technical Report of the Accredited Standard Committee NCITS. IBM recommends that your server’s device driver and host bus adapter

(HBA) use the Class 3 Error Recovery procedures that are specified in the Fibre Channel Protocol for SCSI, Second Version (FCP-2).

Annex B. Error Sense Information

This annex lists all possible combinations of Sense Keys, Additional Sense Codes (ASC), and Additional Sense Code Qualifiers (ASCQ) that are reported by this device.

NOTE 149 - ASCs of EEh and EFh are used for encryption related features and are only supported by encryption capable devices

NOTE 150 - When using encryption enabled devices in an in-band configuration (System method or key path), ASCs of EFh are used to initiate a key management session via a system proxy.

B.1. Sense Key 0 (No Sense)

Table 296 — ASC, and ASCQ Summary for Sense Key 0 (No Sense)

ASC ASCQ	Description
00 00	NO ADDITIONAL SENSE INFORMATION - (unsolicited, no CA/CC)
00 00	NO ADDITIONAL SENSE INFORMATION - EOM=1b (Early Warning)
00 00	NO ADDITIONAL SENSE INFORMATION - ILI=1b
00 00	NO ADDITIONAL SENSE INFORMATION - FM=1b
00 01	FILEMARK DETECTED
00 02	END-OF-PARTITION/MEDIUM DETECTED, EARLY WARNING
00 04	BEGINNING-OF-PARTITION/MEDIUM DETECTED
00 07	PROGRAMMABLE EARLY WARNING DETECTED
00 16	OPERATION IN PROGRESS
14 01	RECORD NOT FOUND (STRING SEARCH)
2E 00	INSUFFICIENT TIME FOR OPERATION (STRING SEARCH)
EF 13	ENCRYPTION - KEY TRANSLATE

NOTE 151 - ASCs of EEh and EFh are used for encryption related features and are only supported by encryption capable devices

NOTE 152 - When using encryption enabled devices in an in-band configuration (System method or key path), ASCs of EFh are used to initiate a key management session via a system proxy.

B.2. Sense Key 1 (Recovered Error)

Table 297 — ASC, and ASCQ Summary for Sense Key 1 (Recovered Error)

ASC ASCQ	Description
00 00	NO ADDITIONAL SENSE INFORMATION
00 17	DRIVE NEEDS CLEANING
17 01	RECOVERED DATA WITH RETRIES
18 00	RECOVERED DATA WITH ERROR CORRECTION APPLIED
37 00	ROUNDED PARAMETER
5B 02	LOG COUNTER AT MAXIMUM
5D 00	FAILURE PREDICTION THRESHOLD EXCEEDED
5D FF	FAILURE PREDICTION THRESHOLD EXCEEDED (FALSE)
83 83	DRIVE HAS BEEN CLEANED
85 00	SEARCH MATCH LIST LIMIT (WARNING)
85 01	SEARCH SNOOP MATCH FOUND

NOTE 153 - Many additional ASC ASCQ combinations are possible if recovered error reporting is enabled via Mode Select. Recovered Error Reporting Enabled is the default option with some of the 3592 device drivers.

B.3. Sense Key 2 (Not Ready)

Table 298 — ASC, and ASCQ Summary for Sense Key 2 (Not Ready)

ASC ASCQ	Description
04 00	LOGICAL UNIT NOT READY, CAUSE NOT REPORTABLE
04 01	LOGICAL UNIT IS IN PROCESS OF BECOMING READY
04 03	LOGICAL UNIT NOT READY, MANUAL INTERVENTION REQUIRED
04 04	LOGICAL UNIT NOT READY, FORMAT IN PROGRESS
04 07	LOGICAL UNIT NOT READY, OPERATION IN PROGRESS
04 13	LOGICAL UNIT NOT READY, SA CREATION IN PROGRESS
0B 01	WARNING - SPECIFIED TEMPERATURE EXCEEDED
0B 0A	WARNING - HIGH CRITICAL TEMPERATURE LIMIT EXCEEDED
0B 0E	WARNING - HIGH CRITICAL HUMIDITY LIMIT EXCEEDED
30 03	CLEANING IN PROGRESS
3A 00	MEDIUM NOT PRESENT
3A 04	NOT READY - MEDIUM AUXILIARY MEMORY ACCESSIBLE
53 00	MEDIA LOAD OR EJECT FAILED
74 11	SA CREATION PARAMETER VALUE REJECTED

B.4. Sense Key 3 (Medium Error)

Table 299 — ASC, and ASCQ Summary for Sense Key 3 (Medium Error)

ASC ASCQ	Description
03 02	EXCESSIVE WRITE ERRORS
04 10	LOGICAL UNIT NOT READY, AUXILIARY MEMORY NOT ACCESSIBLE
09 00	TRACK FOLLOWING ERROR
0C 00	WRITE ERROR
11 00	UNRECOVERED READ ERROR
11 01	READ RETRIES EXHAUSTED
11 08	INCOMPLETE BLOCK READ
14 00	RECORDED ENTITY NOT FOUND
14 01	RECORD NOT FOUND
14 02	FILEMARK OR SETMARK NOT FOUND
14 03	END-OF-DATA NOT FOUND
14 04	BLOCK SEQUENCE ERROR
30 00	INCOMPATIBLE MEDIUM INSTALLED
30 01	CANNOT READ MEDIUM, UNKNOWN FORMAT
30 02	CANNOT READ MEDIUM, INCOMPATIBLE FORMAT
30 0D	WORM MEDIUM - TAMPERING DETECTED
31 00	MEDIUM FORMAT CORRUPTED
31 01	FORMAT COMMAND FAILED
33 00	TAPE LENGTH ERROR
51 00	ERASE FAILURE
52 00	MEDIA LOAD OR EJECT FAILED
53 04	MEDIUM THREAD OR UNTHREAD FAILURE
85 00	WRITE PROTECTED BECAUSE OF TAPE OR DRIVE FAILURE
85 01	WRITE PROTECTED BECAUSE OF TAPE FAILURE
85 02	WRITE PROTECTED BECAUSE OF DRIVE FAILURE
EE 60	ENCRYPTION - PROXY COMMAND ERROR
EE D0	ENCRYPTION - DATA READ DECRYPTION FAILURE
EE D1	ENCRYPTION - DATA READ AFTER WRITE DECRYPTION FAILURE
EE E0	ENCRYPTION - KEY TRANSLATION FAILURE
EE E1	ENCRYPTION - KEY TRANSLATION AMBIGUOUS
EE F0	ENCRYPTION - DECRYPTION FENCED (READ)
EE F1	ENCRYPTION - ENCRYPTION FENCED (WRITE)

NOTE 154 - ASCs of EEh and EFh are used for encryption related features and are only supported by encryption capable devices

NOTE 155 - When using encryption enabled devices in an in-band configuration (System method or key path), ASCs of EFh are used to initiate a key management session via a system proxy.

B.5. Sense Key 4 (Hardware Error)

Table 300 — ASC, and ASCQ Summary for Sense Key 4 (Hardware Error)

ASC ASCQ	Description
09 00	TRACK FOLLOWING ERROR
10 01	LOGICAL BLOCK GUARD CHECK FAILED
10 04	LOGICAL BLOCK PROTECTION ERROR ON RECOVER BUFFERED DATA
15 01	MECHANICAL POSITIONING ERROR
3B 00	SEQUENTIAL POSITIONING ERROR
3B 08	REPOSITION ERROR
40 00	DIAGNOSTIC FAILURE
44 00	INTERNAL TARGET FAILURE CLEANER FAILURE
47 80	READ INTERNAL CRC ERROR
47 81	WRITE INTERNAL CRC ERROR
4C 00	LOGICAL UNIT FAILED SELF-CONFIGURATION
51 00	HARDWARE ERROR - ERASE FAILURE
52 00	CARTRIDGE FAULT
53 00	MEDIA LOAD OR EJECT FAILED
53 01	UNLOAD TAPE FAILURE
EE 0E	ENCRYPTION - KEY SERVICE TIME-OUT [a]
EE 0F	ENCRYPTION - KEY SERVICE FAILURE [a]

NOTE 156 - ASCs of EEh and EFh are used for encryption related features and are only supported by encryption capable devices

NOTE 157 - When using encryption enabled devices in an in-band configuration (System method or key path), ASCs of EFh are used to initiate a key management session via a system proxy.

[a] Returned on device model 3592 E07 and earlier

B.6. Sense Key 5 (Illegal Request)

Table 301 — ASC, and ASCQ Summary for Sense Key 5 (Illegal Request) (part 1 of 3)

ASC ASCQ	Description
00 16	OPERATION IN PROGRESS
0E 03	INVALID FIELD IN COMMAND INFORMATION UNIT (e.g., FCP_DL error)
1A 00	PARAMETER LIST LENGTH ERROR
20 00	INVALID COMMAND OPERATION CODE
20 0C	ILLEGAL COMMAND WHEN NOT IN APPEND-ONLY MODE (see 4.18.3)
21 01	INVALID ELEMENT ADDRESS
24 00	INVALID FIELD IN CDB
25 00	LOGICAL UNIT NOT SUPPORTED
26 00	INVALID FIELD IN PARAMETER LIST
26 01	PARAMETER NOT SUPPORTED
26 02	PARAMETER VALUE INVALID
26 03	THRESHOLD PARAMETERS NOT SUPPORTED
26 04	INVALID RELEASE OF PERSISTENT RESERVATION
26 06	TOO MANY TARGET DESCRIPTORS
26 07	UNSUPPORTED TARGET DESCRIPTOR TYPE CODE
26 08	TOO MANY SEGMENT DESCRIPTORS
26 09	UNSUPPORTED SEGMENT DESCRIPTOR TYPE CODE
26 11	ENCRYPTION - INCOMPLETE KEY-ASSOCIATE DATA SET
26 0C	INVALID OPERATION FOR COPY SOURCE OR DESTINATION
2C 00	COMMAND SEQUENCE ERROR
2C 0B	NOT RESERVED - The OIR bit of the Sequential Access Device page is set and the I_T nexus attempting to communicate with the drive does not hold a reservation.
30 05	CANNOT WRITE MEDIUM - INCOMPATIBLE FORMAT
39 00	SAVING PARAMETERS NOT SUPPORTED
3B 00	SEQUENTIAL POSITIONING ERROR
3B 0D	MEDIUM DESTINATION ELEMENT FULL
3B 0E	MEDIUM SOURCE ELEMENT EMPTY
3B 11	MEDIUM MAGAZINE NOT ACCESSIBLE
3D 00	INVALID BITS IN IDENTIFY MESSAGE
40 80	RECOVERED DIAGNOSTIC FAILURE
49 00	INVALID MESSAGE ERROR (e.g., FCP CMD Fields Invalid)
53 02	MEDIUM REMOVAL PREVENTED
55 03	INSUFFICIENT RESOURCES
55 06	AUXILIARY MEMORY OUT OF SPACE
55 08	MAXIMUM NUMBER OF SUPPLEMENTAL DECRYPTION KEYS EXCEEDED

NOTE 158 - ASCs of EEh and EFh are used for encryption related features and are only supported by encryption capable devices

NOTE 159 - When using encryption enabled devices in an in-band configuration (System method or key path), ASCs of EFh are used to initiate a key management session via a system proxy.

Table 301 — ASC, and ASCQ Summary for Sense Key 5 (Illegal Request) (part 2 of 3)

ASC ASCQ	Description
5B 03	LOG LIST CODES EXHAUSTED
74 08	DIGITAL SIGNATURE VALIDATION FAILURE
74 0C	UNABLE TO DECRYPT PARAMETER LIST
74 10	SA CREATION PARAMETER VALUE INVALID
74 11	SA CREATION PARAMETER VALUE REJECTED
74 12	INVALID SA USAGE
74 21	DATA ENCRYPTION CONFIGURATION PREVENTED
74 30	SA CREATION PARAMETER NOT SUPPORTED
80 00	CU MODE, VENDOR-UNIQUE
85 03	WRITE PROTECTED BECAUSE OF CURRENT TAPE POSITION
EE 00	ENCRYPTION - KEY SERVICE NOT ENABLED
EE 01	ENCRYPTION - KEY SERVICE NOT CONFIGURED
EE 02	ENCRYPTION - KEY SERVICE NOT AVAILABLE
EE 0D	ENCRYPTION - MESSAGE CONTENT ERROR
EE 10	ENCRYPTION - KEY REQUIRED
EE 20	ENCRYPTION - KEY COUNT EXCEEDED
EE 21	ENCRYPTION - KEY ALIAS EXCEEDED
EE 22	ENCRYPTION - KEY RESERVED
EE 23	ENCRYPTION - KEY CONFLICT
EE 24	ENCRYPTION - KEY METHOD CHANGE
EE 25	ENCRYPTION - KEY FORMAT NOT SUPPORTED
EE 26	ENCRYPTION - UNAUTHORIZED REQUEST - dAK
EE 32	ENCRYPTION - UNAUTHORIZED REQUEST - dCERT
EE 27	ENCRYPTION - UNAUTHORIZED REQUEST - dSK
EE 28	ENCRYPTION - UNAUTHORIZED REQUEST - eAK
EE 29	ENCRYPTION - AUTHENTICATION FAILURE
EE 2A	ENCRYPTION - INVALID RDKi
EE 2B	ENCRYPTION - KEY INCORRECT
EE 2C	ENCRYPTION - KEY WRAPPING FAILURE
EE 2D	ENCRYPTION - SEQUENCING FAILURE
EE 2E	ENCRYPTION - UNSUPPORTED TYPE
EE 2F	ENCRYPTION - NEW KEY ENCRYPTED WRITE PENDING
EE 30	ENCRYPTION - PROHIBITED REQUEST
EE 31	ENCRYPTION - KEY UNKNOWN

NOTE 158 - ASCs of EEh and EFh are used for encryption related features and are only supported by encryption capable devices

NOTE 159 - When using encryption enabled devices in an in-band configuration (System method or key path), ASCs of EFh are used to initiate a key management session via a system proxy.

Table 301 — ASC, and ASCQ Summary for Sense Key 5 (Illegal Request) (part 3 of 3)

ASC ASCQ	Description
EE 42	ENCRYPTION - EKM CHALLENGE PENDING
EE E2	ENCRYPTION - KEY TRANSLATION DISALLOWED
EE FF	ENCRYPTION - SECURITY PROHIBITED FUNCTION
EF 01	ENCRYPTION - KEY SERVICE NOT CONFIGURED

NOTE 158 - ASCs of EEh and EFh are used for encryption related features and are only supported by encryption capable devices

NOTE 159 - When using encryption enabled devices in an in-band configuration (System method or key path), ASCs of EFh are used to initiate a key management session via a system proxy.

B.7. Sense Key 6 (Unit Attention)

Table 302 — ASC, and ASCQ Summary for Sense Key 6 (Unit Attention) (part 1 of 2)

ASC ASCQ	Description
00 02	END-OF-PARTITION/MEDIUM DETECTED, EARLY WARNING
28 00	NOT READY TO READY TRANSITION, MEDIUM MAY HAVE CHANGED
28 01	IMPORT OR EXPORT ELEMENT ACCESSED
29 00	POWER ON, RESET, OR BUS DEVICE RESET OCCURRED
2A 01	MODE PARAMETERS CHANGED
2A 02	LOG PARAMETERS CHANGED
2A 03	RESERVATIONS PREEMPTED
2A 04	RESERVATIONS RELEASED
2A 05	REGISTRATIONS PREEMPTED
2A 00	PARAMETERS CHANGED
2A 0D	DATA ENCRYPTION CAPABILITIES CHANGED
2A 10	TIMESTAMP CHANGED
2A 11	ENCRYPTION - DATA ENCRYPTION PARAMETERS CHANGED BY ANOTHER I_T NEXUS
2A 12	ENCRYPTION - DATA ENCRYPTION PARAMETERS CHANGED BY VENDOR SPECIFIC EVENT
2A 14	SA CREATION CAPABILITIES DATA HAS CHANGED
2F 00	COMMANDS CLEARED BY ANOTHER INITIATOR
30 00	INCOMPATIBLE MEDIUM INSTALLED
3B 12	MEDIUM MAGAZINE REMOVED
3B 13	MEDIUM MAGAZINE INSERTED
3B 14	MEDIUM MAGAZINE LOCKED
3B 15	MEDIUM MAGAZINE UNLOCKED
3B 1A	DATA TRANSFER DEVICE REMOVED
3B 1B	DATA TRANSFER DEVICE INSERTED
3F 01	MICROCODE HAS BEEN CHANGED
3F 02	CHANGED OPERATING DEFINITION
3F 03	INQUIRY DATA HAS CHANGED
3F 05	DEVICE IDENTIFIER CHANGED
3F 0E	REPORTED LUNS DATA HAS CHANGED
5A 01	OPERATOR MEDIUM REMOVAL REQUEST
82 83	DRIVE HAS BEEN CLEANED (older versions of microcode)
85 00	SEARCH MATCH LIST LIMIT (ALERT)
EE 12	ENCRYPTION - KEY CHANGE DETECTED
EE 18	ENCRYPTION - CHANGED (READ)
EE 19	ENCRYPTION - CHANGED (WRITE)

NOTE 160 - ASCs of EEh and EFh are used for encryption related features and are only supported by encryption capable devices

NOTE 161 - When using encryption enabled devices in an in-band configuration (System method or key path), ASCs of EFh are used to initiate a key management session via a system proxy.

Table 302 — ASC, and ASCQ Summary for Sense Key 6 (Unit Attention) (part 2 of 2)

ASC ASCQ	Description
EE 40	ENCRYPTION - EKM IDENTIFIER CHANGED
EE 41	ENCRYPTION - EKM CHALLENGE CHANGED
EE 50	ENCRYPTION - INITIATOR IDENTIFIER CHANGED
EE 51	ENCRYPTION - INITIATOR RESPONSE CHANGED

NOTE 160 - ASCs of EEh and EFh are used for encryption related features and are only supported by encryption capable devices

NOTE 161 - When using encryption enabled devices in an in-band configuration (System method or key path), ASCs of EFh are used to initiate a key management session via a system proxy.

B.8. Sense Key 7 (Data Protect)

Table 303 — ASC, and ASCQ Summary for Sense Key 7 (Data Protect)

ASC ASCQ	Description
26 10	ENCRYPTION - DATA DECRYPTION KEY FAIL LIMIT
27 00	WRITE PROTECTED
30 05	CANNOT WRITE MEDIUM - INCOMPATIBLE FORMAT
50 01	WRITE APPEND POSITION ERROR (WORM)
74 04	ENCRYPTION - CRYPTOGRAPHIC INTEGRITY VALIDATION FAILED
74 05	ENCRYPTION - ERROR DECRYPTING DATA
74 03	ENCRYPTION - INCORRECT DATA ENCRYPTION KEY
74 01	ENCRYPTION - UNABLE TO DECRYPT DATA
74 02	ENCRYPTION - UNENCRYPTED DATA ENCOUNTERED WHILE DECRYPTING
74 09	ENCRYPTION MODE MISMATCH ON READ
74 07	ENCRYPTION PARAMETERS NOT USEABLE
74 61	EXTERNAL DATA ENCRYPTION KEY MANAGER ACCESS ERROR
74 62	EXTERNAL DATA ENCRYPTION KEY MANAGER ERROR
74 63	EXTERNAL DATA ENCRYPTION KEY NOT FOUND
74 64	EXTERNAL DATA ENCRYPTION REQUEST NOT AUTHORIZED
74 00	SECURITY ERROR
74 06	UNKNOWN SIGNATURE VERIFICATION KEY
2A 13	ENCRYPTION - DATA ENCRYPTION KEY INSTANCE COUNTER HAS CHANGED
5A 02	OPERATOR SELECTED WRITE PROTECT
74 0A	ENCRYPTED BLOCK NOT RAW READ ENABLED
74 0B	INCORRECT ENCRYPTION PARAMETERS
EE 0E	ENCRYPTION - KEY SERVICE TIME-OUT [a]
EE 0F	ENCRYPTION - KEY SERVICE FAILURE [a]
EE 31	ENCRYPTION - KEY UNKNOWN
EF 10	ENCRYPTION - KEY REQUIRED
EF 11	ENCRYPTION - KEY GENERATION
EF 13	ENCRYPTION - KEY TRANSLATE
EF 1A	ENCRYPTION - KEY OPTIONAL
EF A0	ENCRYPTION - KEY REQUIRED
EF A1	ENCRYPTION - KEY GENERATION
EF C0	ENCRYPTION - NO OPERATION

NOTE 162 - ASCs of EEh and EFh are used for encryption related features and are only supported by encryption capable devices

NOTE 163 - When using encryption enabled devices in an in-band configuration (System method or key path), ASCs of EFh are used to initiate a key management session via a system proxy.

[a] Returned on device model 3592 E08 and later

B.9. Sense Key 8 (Blank Check)

Table 304 — ASC, and ASCQ Summary for Sense Key 8 (Blank Check)

ASC ASCQ	Description
00 05	END-OF-DATA DETECTED
14 01	RECORD NOT FOUND, VOID TAPE

B.10. Sense Key A (Copy Aborted)

The Copy Aborted sense key using a different format of sense data than the other sense keys (see 6.8.3).

Table 305 — ASC, and ASCQ Summary for Sense Key A (Copy Aborted)

ASC ASCQ	Description
0D 01	THIRD PARTY DEVICE FAILURE
0D 02	COPY TARGET DEVICE NOT REACHABLE
0D 03	INCORRECT COPY TARGET DEVICE TYPE
0D 04	COPY TARGET DEVICE DATA UNDERRUN
0D 05	COPY TARGET DEVICE DATA OVERRUN
26 0C	INVALID OPERATION FOR COPY SOURCE OR DESTINATION
2F 02	COMMANDS CLEARED BY DEVICE SERVER

B.11. Sense Key B (Aborted Command)

Table 306 — ASC, and ASCQ Summary for Sense Key B (Aborted Command)

ASC ASCQ	Description
0B 01	WARNING - SPECIFIED TEMPERATURE EXCEEDED
0B 0A	WARNING - HIGH CRITICAL TEMPERATURE LIMIT EXCEEDED
0B 0C	WARNING - HIGH OPERATING TEMPERATURE LIMIT EXCEEDED
0B 0E	WARNING - HIGH CRITICAL HUMIDITY LIMIT EXCEEDED
0B 10	WARNING - HIGH OPERATING HUMIDITY LIMIT EXCEEDED
14 00	RECORDED ENTITY NOT FOUND
14 01	RECORD NOT FOUND
14 02	FILEMARK OR SETMARK NOT FOUND
1B 00	SYNCHRONOUS DATA TRANSFER ERROR
43 00	MESSAGE ERROR
44 00	INTERNAL TARGET FAILURE
45 00	SELECT OR RESELECT FAILURE
47 00	SCSI PARITY ERROR
47 03	INFORMATION UNIT iuCRC ERROR DETECTED
48 00	INITIATOR DETECTED ERROR MESSAGE RECEIVED
49 00	INVALID MESSAGE ERROR
4A 00	COMMAND PHASE ERROR
4B 00	DATA PHASE ERROR
4E 00	OVERLAPPED COMMANDS ATTEMPTED
74 40	AUTHENTICATION FAILED

B.12. Sense Key D (Volume Overflow)

Table 307 — ASC, and ASCQ Summary for Sense Key D (Volume Overflow)

ASC ASCQ	Description
00 02	END-OF-PARTITION/MEDIUM DETECTED

B.13. Sense Key F (Completed)

Table 308 — ASC, and ASCQ Summary for Sense Key F (Completed)

ASC ASCQ	Description
00 20	EXTENDED COPY INFORMATION AVAILABLE

Annex C. Product Comparisons

This annex is provided to summarize key differences and features between this product and the IBM System Storage Tape System 3590 product, as well as the IBM System Storage Ultrium Tape Drive. As a summary, the following information is not intended to definitively detail all aspects and differences of each product family, but should be used in conjunction with this entire manual and various other documents for this and the other products.

> NOTE 164 - All comparisons are made from the perspective of this device (i.e., statements of support or non-support pertain to the 3592 device).
>
> NOTE 165 - All data rate values specified in MB/sec in this clause are in units of 10^6. Host interface attachment values expressed in MB/s for Fibre Channel interfaces are in the units defined in the Fibre Channel standards.

C.1. Summary of Best of Breed Enterprise Tape Drive Features

This product is designed to be the best of breed enterprise tape drive, and has many features to achieve this. These items include:

- Low nominal power consumption
- Small form factor
- Hot-pluggable CE service panel
- Eight (8) character alphanumeric bezel message display
- Rugged automation compatible 3590 form-factor cartridge
- Cartridge Memory
- Resettable loader for maximum load/unload reliability
- Large read/write buffer
- Speed matching
- Channel calibration
- Dual Port Fibre Channel interface
- Fibre Channel N-Port support (i.e., Fabric Topology)
- High linear density, for increased capacity
- High Resolution Tape Directory (HRTD) for optimal space/locate performance
- Virtual backhitch performance accelerator for small transactions
- SkipSync performance accelerator for large files
- WORM (Write Once, Read Many) media support
- Hardware String Search Function (not J1A)
- Support for media use of previous generation formats and media reuse at higher capacities. Supported densities (formats), medium types and capacities per generation are shown in 3592 capacities by density, cartridges, and products (see Table 7 — on page 20).

C.2. Differences between TS1160 (PRODUCT IDENTIFICATION 0359260F) and TS1155 (PRODUCT IDENTIFICATION 0359255F)

The seventh generation offering of the IBM System Storage Tape System 3592 is the TS1160 (which reports PRODUCT IDENTIFICATION 0359260F and is referred to as the 60F). This device maintains maximum compatibility with the sixth generation IBM System Storage Tape System 3592 which is the TS1155 (which reports PRODUCT IDENTIFICATION 0359255Fand is referred to as the 55F), while adding additional features and capabilities. New features have been added with care to minimize impact to existing device drivers and applications.

NOTE 166 - All comparisons are made from the perspective of the 60F (statements of new features or enhancements pertain to the 60F).

C.2.1. Physical Differences

There are no external physical differences beyond device labeling.

C.2.2. Functional Differences

- Cartridges can contain up to 60 000 GB (10^9) with 3:1 data compression.
- The Fibre Channel interface for host attachment is 16 GFC instead of 8 GFC.
- The Fibre Channel interface for host attachment no longer supports L-Port.
- The Fibre Channel interface for host attachment supports N-Port to N-port, whereas earlier generations are limited to explicit configuration only.
- Longitudinal Partitioning is not supported with the 60F format nor on the new media types (i.e., JE/JV/JM).
- 60F format or later and JE, JV, or JM media types or later do not support Performance Segment Scaling. Capacity Scaling is performed instead of Performance Segment Scaling.

C.2.3. Command Differences

There are no new commands.

C.2.4. Data Differences

- Log pages may contain additional counters
- Various density, capacity, and device changes (depending on mode and medium format)

C.3. Differences between TS1155 (PRODUCT IDENTIFICATION 0359255F) and TS1150 (PRODUCT IDENTIFICATION 03592E08)

The sixth generation offering of the IBM System Storage Tape System 3592 is the TS1155 (which reports PRODUCT IDENTIFICATION 0359255F and is referred to as the 55F). This device maintains maximum compatibility with the fifth generation IBM System Storage Tape System 3592 which is the TS1150 (which reports PRODUCT IDENTIFICATION 03592E08 and is referred to as the E08), while adding additional features and capabilities. New features have been added with care to minimize impact to existing device drivers and applications.

NOTE 167 - All comparisons are made from the perspective of the 55F (statements of new features or enhancements pertain to the 55F).

C.3.1. Physical Differences

There are no external physical differences beyond device labeling.

C.3.2. Functional Differences

- Cartridges can contain up to 45 000 GB (10^9) with 3:1 data compression.

C.3.3. Command Differences

There are no new commands.

C.3.4. Data Differences

- Log pages may contain additional counters
- Various density, capacity, and device changes (depending on mode and medium format)

C.4. Differences between 3592 Model E08 and Model E07

The fifth generation offering of the IBM System Storage Tape System 3592 is the E08 model. This model maintains maximum compatibility with the fifth generation E07 model, while adding additional features and capabilities. New features have been added with care to minimize impact to existing device drivers and applications.

> NOTE 168 - All comparisons are made from the perspective of the E08 model (statements of new features or enhancements pertain to the E08 model).

C.4.1. Physical Differences

There are no external physical differences beyond device labeling.

C.4.2. Functional Differences

- Improved resolution of speed matching to better match host data rates and reduce back-hitching (i.e., shoe-shining).
- Increased maximum native data rate of 360 MB/s in 3592A5 density
- Increased maximum transfer rate of 700 MB/sec with maximum compression.
- Improved virtual backhitch small file write synchronize handling
- Improved performance for large files with the new SkipSync feature
- Improved high-speed search to 18 meters/sec
- Larger main data buffer
- The compression core has been improved from previous generations providing an improved average compression ratio.
- Cartridges can contain up to 30 000 GB (10^9) with 3:1 data compression.
- BOP Caching - The initial set of tape blocks in a partition, once read, either by read-ahead function or explicit command, remain in a special place in the cache data buffer (until demount or partition change) such that subsequent locate operations to BOP or read operations of these blocks complete quickly without requiring completion of physical motion.
- Support for new JD, JZ and JL media types, which include increased cartridge memory allowing larger MAM and higher granularity tape directory.
- Improved Read Error Recovery Procedures

C.4.3. Command Differences

- Extended Copy support where TS1150 can perform tape to tape copy without server involvement.
- Supports VERIFY to EOD function to validate the data on the tape without requiring data transfer to the server.

C.4.4. Data Differences

- Log pages may contain additional counters
- Various density, capacity, and model changes (depending on mode and medium format)

C.5. Differences between 3592 Model E07 and Model E06

The fourth generation offering of the IBM System Storage Tape System 3592 is the E07 model. This model maintains maximum compatibility with the third generation E06 model, while adding additional features and capabilities. New features have been added with care to minimize impact to existing device drivers and applications.

NOTE 169 - All comparisons are made from the perspective of the E07 model (statements of new features or enhancements pertain to the E07 model).

C.5.1. Physical Differences

There are no external physical differences beyond device labeling.

C.5.2. Functional Differences

- No longer supports 3592A1 density.
- Improved resolution of speed matching to better match host data rates and reduce back-hitching (i.e., shoe-shining).
- Increased maximum native data rate of 250 MB/s in 3592A4 density
- Increased maximum transfer rate of 650 MB/sec with maximum compression.
- Improved virtual backhitch small file write synchronize handling
- Improved performance for large files with the new SkipSync feature
- Cartridges can contain up to 12 000 GB (10^9) with 3:1 data compression.
- Support for new JC, JY and JK media types.
- Dual port 800 MB/s Fibre Channel attachment for increased performance and connectivity. For specific host bus adapters (HBAs) supported see the IBM System Storage Interoperation Center (SSIC) at http://www-03.ibm.com/systems/support/storage/ssic/interoperability.wss.

C.5.3. Command Differences

- Support for Recommended Access Order

C.5.4. Data Differences

- Log pages may contain additional counters
- Various density, capacity, and model changes (depending on mode and medium format)

C.6. Differences between 3592 Model E06 and Model E05

The third generation offering of the IBM System Storage Tape System 3592 is the E06 model. This model maintains maximum compatibility with the second generation E05 model, while adding additional features and capabilities. New features have been added with care to minimize impact to existing device drivers and applications.

NOTE 170 - All comparisons are made from the perspective of the E06 model (statements of new features or enhancements pertain to the E06 model).

C.6.1. Physical Differences

There are no external physical differences beyond device labeling.

Internal differences follow:

- Giant Magneto Resistive (GMR) technology used for the head

C.6.2. Functional Differences

- Support for additional speeds for matching
- Larger read/write buffer of 1 GiB (2x larger than model E05)
- Increased high speed locate rate of 12 m/s
- Increased maximum native data rate of 160 MB/s in 3592A3 density
- Increased maximum transfer rate of 360 MB/sec with maximum compression.
- Improved virtual backhitch small file write synchronize handling
- Improved performance for large files with the new SkipSync feature
- Standby cooling management feature to reduce power and reduce the risk of unnecessary airborne debris contamination when idle
- Cartridges can contain up to 3000 GB (10^9) with 3:1 data compression.

C.6.3. Command Differences

- None

C.6.4. Data Differences

- Log pages may contain additional counters
- Various density, capacity, and model changes (depending on mode and medium format)

C.7. Differences between 3592 Model E05 and Model J1A

The second generation offering of the IBM System Storage Tape System 3592 is the E05 model. This model maintains maximum compatibility with the first generation J1A model, while adding additional features and capabilities. New features have been added with care to minimize impact to existing device drivers and applications.

> NOTE 171 - All comparisons are made from the perspective of the E05 model (statements of new features or enhancements pertain to the E05 model).

C.7.1. Physical Differences

There are no external physical differences beyond device labeling.

Internal differences follow:

- Improved tape mechanical tape guiding

C.7.2. Functional Differences

- Support for additional speeds for matching (speed 5 and 6) (2 more speeds than J1A)
- Larger read/write buffer of 512 MiB (4x larger than model J1A)
- Increased high speed locate rate of 10 m/s
- Increased maximum read/write speed
- Increased maximum native data rate of 100 MB/s in 3592A2 density
- Increased maximum native data rate of 50 MB/s in 3592A1 density (J1A model is 40 MB/s)
- Increased maximum transfer rate of 260 MB/sec with maximum compression.
- 4Gbps dual port fibre channel interfaces
- Support for Constant Capacity mode to provide high probability that tapes that are copied will fit on other tapes
- Default to automatic up formatting of existing medium when written from BOT
- Hardware String Search function at full device native data rates
- Virtual backhitch small file write synchronize handling
- Cartridges can contain up to 1500 GB (10^9) with 3:1 data compression.
- Support for new JB and JX media types.
- Dual port 400 MB/s Fibre Channel attachment for increased performance and connectivity. For specific host bus adapters (HBAs) supported see the web at

  http://www-03.ibm.com/systems/support/storage/config/ssic/displayesssearchwithoutjs.wss?start_over=yes.
- Some models of the 3592 E05 contain encryption hardware capable of encrypting data at tape speeds.
- The models of the 3592 E05 that contain encryption hardware support application transparent encryption

C.7.3. Command Differences

- New String Search command opcode E3h
- New String Search Send Diagnostic 3000h
- New Vendor-Unique String Search options for Read and Space
- New String Search Read Buffer ID 40h
- New String Search mode page 37h and subpage(s) support
- Reformat support using density code and/or Mode Page 25h
- New Log Page and Subpage support for Performance Characteristics (37h) for interfaces, device and medium

C.7.4. Data Differences

- Report Density supports medium reuse
- Log pages may contain additional counters
- Various density, capacity, and model changes (depending on mode and medium format)

C.8. Differences between 3592 Model J1A and IBM SystemStorage Tape System 3590

This product was designed to bring advanced technology features into the enterprise, while maintaining maximum compatibility with the proven high reliability of the IBM System Storage Tape System 3590. New features have been added with care to minimize impact to existing device drivers and applications.

NOTE 172 - All comparisons are made from the perspective of this device (i.e., statements of support or non-support pertain to the 3592 J1A model).

C.8.1. Physical Differences

- Maximum native data rate of 40 MB/sec (uncompressed)
- Maximum transfer rate of 110 MB/sec with maximum compression
- 8 meters/sec high-speed search provides rapid access to stored data.
- Cartridges can contain up to 900 GB (10^9) with 3:1 data compression.
- Up to a 100-fold increase in data integrity over the 3480.
- Smaller overall device form factor
- No pneumatic compressor
- Physically prevents loading of 3590 cartridge
- Loader is “hard” and requires code to move tray to mounting positions
- Leader pin cartridge design
- Cartridge memory
- Optimal non-expanding data compression
- 8 character alphanumeric bezel message display
- Dual port 200 MB/s Fibre Channel attachment (i.e., 2 Gbps) for increased performance and connectivity. For specific host bus adapters (HBAs) supported see the web at

  http://www-03.ibm.com/systems/support/storage/config/ssic/displayesssearchwithoutjs.wss?start_over=yes.
- No parallel SCSI interface

C.8.2. Command Differences

- Certain checks (Reservation conflicts, DCCs, UAs, Media format corrupt, After Check 1) are now checked and acted on when the command is queued (will immediately report these conditions). Checking is also repeated when the command is dequeued as condition(s) may have occurred.
- Command time-out changes
- Erase with Long 0b supported
- Host Early Warning on Mode Page 25h minimum is larger
- Load Unload with Load 1b supported in more environments
- Log Counters are not scaled as they are incremented (previously scaled counters are now more accurate and are only rounded when reported)
- Log Sense additional pages 2Eh, 30h, and 3Eh
- Read Buffer IDs 20h
- Read Buffer mode 2 supported
- Sense data for all ILI conditions may be only 18 bytes long (3590 only generated 18 bytes for underlength ILIs)
- Space and Locate at EOD will not attempt to cross into old data
- TapeAlert support (mode page 1Ch and log page 2Eh)
- Write Buffer ID 30h
- Write Buffer mode 2 and 4

C.8.3. Data Differences

- Inquiry page 83 reports port information
- Log pages may contain additional counters
- Mode page 21h TOD basis changed
- Sense data enhanced vendor-unique debug information
- Various density, capacity, and model changes
- Write sense Information field is command relative (SSC-2)

Annex D. Firmware Download

This annex describes how to compare a firmware level binary to information returned by a drive and determine which firmware is acceptable in which drive.

D.1. Identifying Level Hardware of Drive

The firmware that is loaded in the drive will report a LOAD ID and RU NAME in INQUIRY IP 03h: ASCII Information (see 6.3.2 on page 262). The LOAD ID and RU NAME are used to designate the Hardware (i.e., Product). The following table defines the LOAD ID and RU NAME values for each Hardware.

Table 309 — Load ID and RU Name Designation

Product (Hardware)	LOAD ID	RU NAME **"ASCII"** **Hex**	PRODUCT IDENTIFICATION **(Left-Aligned)**
3592	A170029E	"AJEFGP9E" 0x414A454647503945	03592J1A
TS1120	A1700D5C	"AJEFAX5C" 0x414A454641583543	03592E05
TS1130	A1700D6D	"AJEFAX6D" 0x414A454641583644	03592E06
TS1140	A1700D78	"AJEFAX78" 0x414A454641583738	03592E07
TS1150	A1700D86	"AJEFAX86" 0x414A454641583836	03592E08
TS1155 FCP			0359255F
TS1160 FCP	A1700D7D	"AJEFAX7D" 0x414A454641583744	0359260F
FCP - Fibre Channel			

D.2. Identifying the product for which the firmware image is intended

The Firmware Image is defined in table 310.

Table 310 — Firmware Image

Byte	Bit							
	7 msb	**6**	**5**	**4**	**3**	**2**	**1**	**0 lsb**
0		Not Specified						
3								
4	(MSB)	FIRMWARE LENGTH + HEADER LENGTH (m+1)						
7								(LSB)
8	(MSB)	LOAD ID (See “IP 03h: ASCII Information” on page 262.)						
11								
12	(MSB)	FIRMWARE REVISION LEVEL (See Inquiry Standard Data bytes 32 - 35)						
15								(LSB)
16		Reserved						
23								
24	(MSB)	RU NAME (See “IP 03h: ASCII Information” on page 262.)						
31								(LSB)
32		Not Specified						
m								

The LOAD ID and RU NAME fields in the Firmware Image are used to define the product (i.e., Level Hardware) for which the Firmware Image is intended.

D.3. Download Process

Confirm the Level Hardware of the Firmware Image (see D.2.) to be loaded matches the Level Hardware of the drive (see D.1.).

Download the Firmware Image using the WRITE BUFFER - 3Bh (see 5.2.50 on page 220) command.

Annex E. Protection Information CRC’s

E.1. Reed-Solomon CRC

E.1.1. Reed-Solomon CRC Algorithm

The Reed Solomon CRC algorithm defined in ECMA-319 is used in this drive:

a) as a format specific symbol written to tape with each logical block; and
b) is available for use as protection information associated with each logical block transferred between the drive and a host (see 4.7).

The CRC bytes are Reed-Solomon (N, N-4) codes over GF (256).

A calculation in GF (256) is defined by P(x) = $x^8 + x^4 + x^3 + x^2 + 1$

$\alpha = (0\ 0\ 0\ 0\ 0\ 0\ 1\ 0)$

$G(x) = x^4 + \alpha^{201}x^3 + \alpha^{246}x^2 + \alpha^{201}x + 1$

E.1.2. Sample C program to generate Reed-Solomon CRC

The following is sample C code for generating the Reed-Solomon CRC defined in ECMA-319

```
/*----------------------------------------------------------------------------
**  ABSTRACT: function to compute interim LBP CRC
**  INPUTS:   crc   - initial crc (0 for fresh) (i.e., seed)
**            cnt   - the number of data bytes to compute CRC for
**           start - the starting address of the data bytes (e.g., data buffer)
**  OUTPUTS:  UINT32 - crc in big endian (MSB is first byte)
*/
UINT32 GenerateRSCRC(UINT32 crc, UINT32 cnt, const void *start)
{
 static const UINT32 crcTable[256]=
 { 0x00000000,0x38CF3801,0x70837002,0x484C4803,0xE01BE004,0xD8D4D805,
   0x90989006,0xA857A807,0xDD36DD08,0xE5F9E509,0xADB5AD0A,0x957A950B,
   0x3D2D3D0C,0x05E2050D,0x4DAE4D0E,0x7561750F,0xA76CA710,0x9FA39F11,
   0xD7EFD712,0xEF20EF13,0x47774714,0x7FB87F15,0x37F43716,0x0F3B0F17,
   0x7A5A7A18,0x42954219,0x0AD90A1A,0x3216321B,0x9A419A1C,0xA28EA21D,
   0xEAC2EA1E,0xD20DD21F,0x53D85320,0x6B176B21,0x235B2322,0x1B941B23,
   0xB3C3B324,0x8B0C8B25,0xC340C326,0xFB8FFB27,0x8EEE8E28,0xB621B629,
   0xFE6DFE2A,0xC6A2C62B,0x6EF56E2C,0x563A562D,0x1E761E2E,0x26B9262F,
   0xF4B4F430,0xCC7BCC31,0x84378432,0xBCF8BC33,0x14AF1434,0x2C602C35,
   0x642C6436,0x5CE35C37,0x29822938,0x114D1139,0x5901593A,0x61CE613B,
   0xC999C93C,0xF156F13D,0xB91AB93E,0x81D5813F,0xA6ADA640,0x9E629E41,
   0xD62ED642,0xEEE1EE43,0x46B64644,0x7E797E45,0x36353646,0x0EFA0E47,
   0x7B9B7B48,0x43544349,0x0B180B4A,0x33D7334B,0x9B809B4C,0xA34FA34D,
   0xEB03EB4E,0xD3CCD34F,0x01C10150,0x390E3951,0x71427152,0x498D4953,
   0xE1DAE154,0xD915D955,0x91599156,0xA996A957,0xDCF7DC58,0xE438E459,
   0xAC74AC5A,0x94BB945B,0x3CEC3C5C,0x0423045D,0x4C6F4C5E,0x74A0745F,
   0xF575F560,0xCDBACD61,0x85F68562,0xBD39BD63,0x156E1564,0x2DA12D65,
   0x65ED6566,0x5D225D67,0x28432868,0x108C1069,0x58C0586A,0x600F606B,
   0xC858C86C,0xF097F06D,0xB8DBB86E,0x8014806F,0x52195270,0x6AD66A71,
   0x229A2272,0x1A551A73,0xB202B274,0x8ACD8A75,0xC281C276,0xFA4EFA77,
   0x8F2F8F78,0xB7E0B779,0xFFACFF7A,0xC763C77B,0x6F346F7C,0x57FB577D,
   0x1FB71F7E,0x2778277F,0x51475180,0x69886981,0x21C42182,0x190B1983,
   0xB15CB184,0x89938985,0xC1DFC186,0xF910F987,0x8C718C88,0xB4BEB489,
```

```
   0xFCF2FC8A,0xC43DC48B,0x6C6A6C8C,0x54A5548D,0x1CE91C8E,0x2426248F,
   0xF62BF690,0xCEE4CE91,0x86A88692,0xBE67BE93,0x16301694,0x2EFF2E95,
   0x66B36696,0x5E7C5E97,0x2B1D2B98,0x13D21399,0x5B9E5B9A,0x6351639B,
   0xCB06CB9C,0xF3C9F39D,0xBB85BB9E,0x834A839F,0x029F02A0,0x3A503AA1,
   0x721C72A2,0x4AD34AA3,0xE284E2A4,0xDA4BDAA5,0x920792A6,0xAAC8AAA7,
   0xDFA9DFA8,0xE766E7A9,0xAF2AAFAA,0x97E597AB,0x3FB23FAC,0x077D07AD,
   0x4F314FAE,0x77FE77AF,0xA5F3A5B0,0x9D3C9DB1,0xD570D5B2,0xEDBFEDB3,
   0x45E845B4,0x7D277DB5,0x356B35B6,0x0DA40DB7,0x78C578B8,0x400A40B9,
   0x084608BA,0x308930BB,0x98DE98BC,0xA011A0BD,0xE85DE8BE,0xD092D0BF,
   0xF7EAF7C0,0xCF25CFC1,0x876987C2,0xBFA6BFC3,0x17F117C4,0x2F3E2FC5,
   0x677267C6,0x5FBD5FC7,0x2ADC2AC8,0x121312C9,0x5A5F5ACA,0x629062CB,
   0xCAC7CACC,0xF208F2CD,0xBA44BACE,0x828B82CF,0x508650D0,0x684968D1,
   0x200520D2,0x18CA18D3,0xB09DB0D4,0x885288D5,0xC01EC0D6,0xF8D1F8D7,
   0x8DB08DD8,0xB57FB5D9,0xFD33FDDA,0xC5FCC5DB,0x6DAB6DDC,0x556455DD,
   0x1D281DDE,0x25E725DF,0xA432A4E0,0x9CFD9CE1,0xD4B1D4E2,0xEC7EECE3,
   0x442944E4,0x7CE67CE5,0x34AA34E6,0x0C650CE7,0x790479E8,0x41CB41E9,
   0x098709EA,0x314831EB,0x991F99EC,0xA1D0A1ED,0xE99CE9EE,0xD153D1EF,
   0x035E03F0,0x3B913BF1,0x73DD73F2,0x4B124BF3,0xE345E3F4,0xDB8ADBF5,
   0x93C693F6,0xAB09ABF7,0xDE68DEF8,0xE6A7E6F9,0xAEEBAEFA,0x962496FB,
   0x3E733EFC,0x06BC06FD,0x4EF04EFE,0x763F76FF};

       UINT32  i;
 const UINT8*  d = start;

  for ( i=0; i<cnt; i++ )
  {
    crc = (crc << 8) ^ crcTable[*d ^ (crc >> 24)];
    d++;
  }
return crc;
}
```

E.1.3. Sample C program to compute and append Reed-Solomon CRC to a data block

```
/*----------------------------------------------------------------------------
**  ABSTRACT: function to compute and append LBP CRC to a data block
**  INPUTS:   blkbuf  - starting address of the data block to protect
**            blklen  - length of block to protect (NOT including CRC)
**  OUTPUTS:  UINT32 - length of protected block (to write) including LBP CRC
*/
UINT32 BlockProtectRSCRC(UINT8 *blkbuf, UINT32 blklen)
{
 UINT32 crc = GenerateRSCRC(0x00000000, blklen, blkbuf);

  if (blklen == 0)
    return 0; //no such thing as a zero length block in SSC (write NOP)

 //append CRC in proper byte order (regardless of system endian-ness)
  blkbuf[blklen+0] = (crc >> 24) & 0xFF;
  blkbuf[blklen+1] = (crc >> 16) & 0xFF;
  blkbuf[blklen+2] = (crc >>  8) & 0xFF;
  blkbuf[blklen+3] = (crc >>  0) & 0xFF;

  return (blklen+4); //size of block to be written includes CRC
}
```

E.1.4. Sample C program to to verify block with Reed-Solomon CRC

```
/*---------------------------------------------------------------------------
**  ABSTRACT: function to verify block with LBP CRC
**  INPUTS:   blkbuf  - starting address of the data block to protect
**            blklen  - length of block to verify (INCLUDING CRC)
**  OUTPUTS:  UINT32 - length of block w/o CRC (0 if verify failed)
*/
UINT32 BlockVerifyRSCRC(const UINT8 *blkbuf, UINT32 blklen)
{
  if (blklen <= 4)
    return 0; //block is too small to be valid, cannot check CRC

  blklen -= 4; //user data portion does not include CRC

#if 1 //method 1: calculate CRC on data only and compare against CRC from block
  {
   UINT32 crccmp = GenerateRSCRC(0x00000000, blklen, blkbuf);
   UINT32 crcblk;

   //this matches the append method in the function above
    crcblk  = (blkbuf[blklen+0] << 24) |
              (blkbuf[blklen+1] << 16) |
              (blkbuf[blklen+2] <<  8) |
              (blkbuf[blklen+3] <<  0);

    if (crccmp != crcblk)
      return 0; //block CRC is incorrect
    return(blklen);
  }
#endif
#if 1 //method 2: calculate including CRC and check magic constant
  {
    if (GenerateRSCRC(0x00000000, blklen+4, blkbuf) != 0x00000000)
      return 0; //block CRC is incorrect (CRC did not neutralize)
    return(blklen);
  }
#endif
}
```

E.2. CRC32C (Castagnoli)

E.2.1. CRC32C Algorithm

The CRC32C CRC algorithm is available for use as protection information associated with each logical block transferred between the drive and a host (see 4.7). Some host system architectures may offer higher performance methods to calculate CRCs using this type of polynomial over the Reed-Solomon CRC polynomial.

The algorithm is defined by the equation $x^{32}+x^{28}+x^{27}+x^{26}+x^{25}+x^{23}+x^{23}+x^{22}++x^{20}+x^{19}+x^{18}+x^{14}+x^{13}+x^{11}+x^{10}+x^{9}+x^{8}+x^{6}+1$ which is sometimes expressed as 0x1EDC6F41.

NOTE 173 - The sample functions for CRC32C in this section use a bit swapped form, which is easier/faster for software implementation as it avoids the need to bit swap bytes of input and output.

NOTE 174 - Many other uses of CRC32C (e.g., iSCSI) pad the data to a 32 bit boundary before computing CRC32C. LBP use of CRC32C is byte oriented (as are SSC data blocks), and computes CRC32C on blocks to the actual byte size of the block and immediately appends the CRC32C without and pad in the calculation or storage of the CRC.

NOTE 175 - Some host hardware implementations may have standard form polynomial instruction(s) that can be used for high performance computation of CRC32C or other standard form polynomials (the Reed-Solomon CRC is not such an algorithm). These instructions may have particular input data alignment requirements (e.g., 32 or 64 bit). Since LBP has no padding and is byte (8 bit) aligned, special operations (e.g., a mix of hardware and software methods) may be required to properly calculate non-aligned block sizes. Additionally hardware implementations may use the normal form rather than the swapped form shown in the code examples in this section. This can affect the method for storage and compare (magic constant). In any event, take care to ensure the CRC is stored and checked properly in all cases. Some CRC32C test vectors can be found in the iSCSI RFC 3720 (B.4). Further information (e.g., hardware implementation(s)) may be found in RFC 3385 or from other sources.

E.2.2. Sample C program to generate CRC32C (Castagnoli)

```
/*----------------------------------------------------------------------------
**  ABSTRACT: function to compute interim LBP CRC (bit-swapped method)
**  INPUTS:   crc   - initial crc (0xFFFFFFFF for fresh)
**            cnt   - the number of data bytes to compute CRC for
**            start - the starting address of the data bytes
**  OUTPUTS:  UINT32 - [inverted] crc in big endian (LSB is first byte)
*/
UINT32 GenerateCRC32C(UINT32 crc, UINT32 cnt, const void *start)
{
 static const UINT32 crcTable[256]=
 { 0x00000000,0xF26B8303,0xE13B70F7,0x1350F3F4,0xC79A971F,0x35F1141C,
   0x26A1E7E8,0xD4CA64EB,0x8AD958CF,0x78B2DBCC,0x6BE22838,0x9989AB3B,
   0x4D43CFD0,0xBF284CD3,0xAC78BF27,0x5E133C24,0x105EC76F,0xE235446C,
   0xF165B798,0x030E349B,0xD7C45070,0x25AFD373,0x36FF2087,0xC494A384,
   0x9A879FA0,0x68EC1CA3,0x7BBCEF57,0x89D76C54,0x5D1D08BF,0xAF768BBC,
   0xBC267848,0x4E4DFB4B,0x20BD8EDE,0xD2D60DDD,0xC186FE29,0x33ED7D2A,
   0xE72719C1,0x154C9AC2,0x061C6936,0xF477EA35,0xAA64D611,0x580F5512,
   0x4B5FA6E6,0xB93425E5,0x6DFE410E,0x9F95C20D,0x8CC531F9,0x7EAEB2FA,
   0x30E349B1,0xC288CAB2,0xD1D83946,0x23B3BA45,0xF779DEAE,0x05125DAD,
   0x1642AE59,0xE4292D5A,0xBA3A117E,0x4851927D,0x5B016189,0xA96AE28A,
   0x7DA08661,0x8FCB0562,0x9C9BF696,0x6EF07595,0x417B1DBC,0xB3109EBF,
   0xA0406D4B,0x522BEE48,0x86E18AA3,0x748A09A0,0x67DAFA54,0x95B17957,
   0xCBA24573,0x39C9C670,0x2A993584,0xD8F2B687,0x0C38D26C,0xFE53516F,
   0xED03A29B,0x1F682198,0x5125DAD3,0xA34E59D0,0xB01EAA24,0x42752927,
   0x96BF4DCC,0x64D4CECF,0x77843D3B,0x85EFBE38,0xDBFC821C,0x2997011F,
   0x3AC7F2EB,0xC8AC71E8,0x1C661503,0xEE0D9600,0xFD5D65F4,0x0F36E6F7,
   0x61C69362,0x93AD1061,0x80FDE395,0x72966096,0xA65C047D,0x5437877E,
   0x4767748A,0xB50CF789,0xEB1FCBAD,0x197448AE,0x0A24BB5A,0xF84F3859,
   0x2C855CB2,0xDEEEDFB1,0xCDBE2C45,0x3FD5AF46,0x7198540D,0x83F3D70E,
   0x90A324FA,0x62C8A7F9,0xB602C312,0x44694011,0x5739B3E5,0xA55230E6,
   0xFB410CC2,0x092A8FC1,0x1A7A7C35,0xE811FF36,0x3CDB9BDD,0xCEB018DE,
   0xDDE0EB2A,0x2F8B6829,0x82F63B78,0x709DB87B,0x63CD4B8F,0x91A6C88C,
   0x456CAC67,0xB7072F64,0xA457DC90,0x563C5F93,0x082F63B7,0xFA44E0B4,
   0xE9141340,0x1B7F9043,0xCFB5F4A8,0x3DDE77AB,0x2E8E845F,0xDCE5075C,
   0x92A8FC17,0x60C37F14,0x73938CE0,0x81F80FE3,0x55326B08,0xA759E80B,
   0xB4091BFF,0x466298FC,0x1871A4D8,0xEA1A27DB,0xF94AD42F,0x0B21572C,
   0xDFEB33C7,0x2D80B0C4,0x3ED04330,0xCCBBC033,0xA24BB5A6,0x502036A5,
   0x4370C551,0xB11B4652,0x65D122B9,0x97BAA1BA,0x84EA524E,0x7681D14D,
   0x2892ED69,0xDAF96E6A,0xC9A99D9E,0x3BC21E9D,0xEF087A76,0x1D63F975,
   0x0E330A81,0xFC588982,0xB21572C9,0x407EF1CA,0x532E023E,0xA145813D,
   0x758FE5D6,0x87E466D5,0x94B49521,0x66DF1622,0x38CC2A06,0xCAA7A905,
   0xD9F75AF1,0x2B9CD9F2,0xFF56BD19,0x0D3D3E1A,0x1E6DCDEE,0xEC064EED,
   0xC38D26C4,0x31E6A5C7,0x22B65633,0xD0DDD530,0x0417B1DB,0xF67C32D8,
   0xE52CC12C,0x1747422F,0x49547E0B,0xBB3FFD08,0xA86F0EFC,0x5A048DFF,
   0x8ECEE914,0x7CA56A17,0x6FF599E3,0x9D9E1AE0,0xD3D3E1AB,0x21B862A8,
   0x32E8915C,0xC083125F,0x144976B4,0xE622F5B7,0xF5720643,0x07198540,
```

```
    0x590AB964,0xAB613A67,0xB831C993,0x4A5A4A90,0x9E902E7B,0x6CFBAD78,
    0x7FAB5E8C,0x8DC0DD8F,0xE330A81A,0x115B2B19,0x020BD8ED,0xF0605BEE,
    0x24AA3F05,0xD6C1BC06,0xC5914FF2,0x37FACCF1,0x69E9F0D5,0x9B8273D6,
    0x88D28022,0x7AB90321,0xAE7367CA,0x5C18E4C9,0x4F48173D,0xBD23943E,
    0xF36E6F75,0x0105EC76,0x12551F82,0xE03E9C81,0x34F4F86A,0xC69F7B69,
    0xD5CF889D,0x27A40B9E,0x79B737BA,0x8BDCB4B9,0x988C474D,0x6AE7C44E,
    0xBE2DA0A5,0x4C4623A6,0x5F16D052,0xAD7D5351 };

        UINT32  i;
  const UINT8*  d = start;

   for ( i=0; i<cnt; i++ )
   {
     crc = (crc >> 8) ^ crcTable[*d ^ (crc & 0xFF)];
     d++;
   }
   return crc;
 }
```

E.2.3. Sample C code to compute and append CRC32C to a data block

```
/*------------------------------------------------------------------------------
**  ABSTRACT: function to compute and append LBP CRC to a data block
**  INPUTS:   blkbuf  - starting address of the data block to protect
**            blklen  - length of block to protect (NOT including CRC)
**  OUTPUTS:  UINT32 - length of protected block (to write) including LBP CRC
*/
UINT32 BlockProtectCRC32C(UINT8 *blkbuf, UINT32 blklen)
{
 UINT32 crc = ~GenerateCRC32C(0xFFFFFFFF, blklen, blkbuf); //note bit inversion

  if (blklen == 0)
    return 0; //no such thing as a zero length block in SSC (write NOP)

 //append CRC in proper byte order (regardless of system endian-ness)
  blkbuf[blklen+0] = (crc >>  0) & 0xFF;
  blkbuf[blklen+1] = (crc >>  8) & 0xFF;
  blkbuf[blklen+2] = (crc >> 16) & 0xFF;
  blkbuf[blklen+3] = (crc >> 24) & 0xFF;

  return (blklen+4); //size of block to be written includes CRC
}
```

E.2.4. Sample C code to verify block with CRC32C CRC

```
/*------------------------------------------------------------------------------
**  ABSTRACT: function to verify block with LBP CRC
**  INPUTS:   blkbuf  - starting address of the data block to protect
**            blklen  - length of block to verify (INCLUDING CRC)
**  OUTPUTS:  UINT32 - length of block w/o CRC (0 if verify failed)
*/
UINT32 BlockVerifyCRC32C(const UINT8 *blkbuf, UINT32 blklen)
{
  if (blklen <= 4)
    return 0; //block is too small to be valid, cannot check CRC

  blklen -= 4; //user data portion does not include CRC

#if 1 //method 1: calculate CRC on data only and compare against CRC from block
  {
   UINT32 crccmp = ~GenerateRSCRC(0xFFFFFFFF, blklen, blkbuf); //note bit inversion
```

```
   UINT32 crcblk;

   //this matches the append method in the function above
    crcblk  = (blkbuf[blklen+0] <<  0) |
              (blkbuf[blklen+1] <<  8) |
              (blkbuf[blklen+2] << 16) |
              (blkbuf[blklen+3] << 24);

    if (crccmp != crcblk)
      return 0; //block CRC is incorrect
    return(blklen);
  }
#endif
#if 1 //method 2: calculate including CRC and check magic constant
  {
   //NOTE: bit swapped magic constant is also bit+byte swapped
   //          0x1C2D19ED //"nominal" result including [inverted] CRC
   //          0xB798B438 //"swapped" result including [inverted] CRC
   //NOTE: magic constant check below does NOT need bit inversion
    if (GenerateCRC32C(0xFFFFFFFF, blklen+4, blkbuf) != 0xB798B438)
      return 0; //block CRC is incorrect (CRC did not neutralize)
    return(blklen);
  }
#endif
}
```

E.2.5. Test vector for CRC32C CRC

An example of a 255 byte logical block containing incrementing data from 01h to FFh with CRC32C appended resulting in a 259 byte protected logical block:

Byte	0	1	2	3
0:	01h	02h	03h	04h
...				
(CRC begin) 252:	FDh	FEh	FFh	42h
(CRC end) 256:	B3h	F3h	31h	

E.3. CRC32-IEEE

E.3.1. CRC32-IEEE Algorithm

The CRC32-IEEE algorithm is not used for logical block protection but is used only in the transport layer (i.e., Fibre Channel CRC). This algorithm is listed as an unsupported algorithm in MP 24h: Initiator-Specific Extensions (see 6.6.20 on page 424). This algorithm is defined by the equation $x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^{8}+x^{7}+x^{5}+x^{4}+x^{2}+x+1$ which is sometimes expressed as 0x04C11DB7.

Index Of Counters (log parameters not in counter format are not included)

Notices

References in this publication to IBM programs or services do not imply that IBM intends to make these available in all countries (or regions) in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program, or service. Evaluation and verification of operation in conjunction with other products, except those expressly designed by IBM, are the user's responsibility.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to: Director, Intellectual Property and Licensing, North Castle Drive, Armonk, New York 10504, U.S.A.

Trademarks

The following are trademarks of International Business Machines Corporation in the United States, or other countries (or regions), or both:

AIX
AS/400
IBM
IBMLink
Magstar
NUMA-Q
RS/6000
S/390
Seascape
Sequent
Tivoli
System Storage

Intel is a registered trademark of Intel Corporation in the United States, or other countries (or regions), or both.

Microsoft, Windows, Windows NT, Windows 2000, and the Windows logo are registered trademarks of Microsoft Corporation in the United States, or other countries (or regions), or both.

Other company, product, or service names may be the trademarks or service marks of others.

IBM Agreement for Licensed Internal Code

IBM Agreement for Licensed Internal Code You accept the terms of this Agreement by your initial use of a machine that contains IBM Licensed Internal Code (called Code).

These terms apply to Code used by certain machines IBM or your reseller specifies (called Specific Machines). International Business Machines Corporation or one of its subsidiaries (IBM) owns copyrights in Code or has the right to license Code. IBM or a third party owns all copies of Code, including all copies made from them.

If you are the rightful possessor of a Specific Machine, IBM grants you a license to use the Code (or any replacement IBM provides) on, or in conjunction with, only the Specific Machine for which the Code is provided. IBM licenses the Code to only one rightful possessor at a time.

Under each license, IBM authorizes you to do only the following:

1) Execute the Code to enable the Specific Machine to function according to its Official Published Specifications (called Specifications);
2) Make a backup or archival copy of the Code (unless IBM makes one available for your use), provided you reproduce the copyright notice and any other legend of ownership on the copy. You may use the copy only to replace the original, when necessary; and
3) Execute and display the Code as necessary to maintain the Specific Machine.

You agree to acquire any replacement for, or additional copy of, Code directly from IBM in accordance with IBM's standard policies and practices. You also agree to use that Code under these terms.

You may transfer possession of the Code to another party only with the transfer of the Specific Machine. If you do so, you must 1) destroy all your copies of the Code that were not provided by IBM, 2) either give the other party all your IBM-provided copies of the Code or destroy them, and 3) notify the other party of these terms. IBM licenses the other party when it accepts these terms. These terms apply to all Code you acquire from any source.

Your license terminates when you no longer rightfully possess the Specific Machine.

Actions You May Not Take

You agree to use the Code only as authorized above. You may not do, for example, any of the following:

1) Otherwise copy, display, transfer, adapt, modify, or distribute the Code (electronically or otherwise), except as IBM may authorize in the Specific Machine's Specifications or in writing to you;
2) Reverse assemble, reverse compile, or otherwise translate the Code unless expressly permitted by applicable law without the possibility of contractual waiver;
3) Sublicense or assign the license for the Code; or
4) Lease the Code or any copy of it.

AIX License Information Additional Terms and Conditions

As pertaining to AIX code within the 3590 Model A00 and A50, the following applies.

This software and documentation is based in part on the Fourth Berkeley Software Distribution under license from the Regents of the University of California. We acknowledge the following institutions for their role in its development: the Electrical Engineering and Computer Sciences Department at the Berkeley campus.

The Fourth Berkeley Software Distribution is provided by the Regents of the University of California and other contributors on an AS-IS basis. NEITHER THE REGENTS OF THE UNIVERSITY OF CALIFORNIA NOR THE OTHER CONTRIBUTORS MAKE ANY WARRANTIES, EITHER EXPRESS OR IMPLIED, AS TO ANY MATTER WHATSOEVER, INCLUDING WITHOUT LIMITATION, THE CONDITION OF THE FOURTH BERKELEY SOFTWARE DISTRIBUTION, ITS MERCHANTABILITY OR ITS FITNESS FOR ANY PARTICULAR PURPOSE.

The Regents of the University of California and other contributors are under no obligation to provide either maintenance services, update services, notices of latent defects, or correction of defects for the Fourth Berkeley Software Distribution.

22 April 2019 **GA32-0968-05**

Printed in USA

GA32-0968-05

IBM 3592 SCSI Reference