ISO/IEC 14776-454

Edition 1.0 2018-04

INTERNATIONAL STANDARD

colour inside

Information technology – Small Computer System Interface (SCSI) – Part 454: SCSI Primary Commands – 4 (SPC-4)



ISO/IEC 14776-454:2018-04(en)



THIS PUBLICATION IS COPYRIGHT PROTECTED
Copyright © 2018 ISO/IEC, Geneva, Switzerland

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either IEC or IEC's member National Committee in the country of the requester. If you have any questions about ISO/IEC copyright or have an enquiry about obtaining additional rights to this publication, please contact the address below or your local IEC member National Committee for further information.

IEC Central Office
3, rue de Varembé
CH-1211 Geneva 20
Switzerland

Tel.: +41 22 919 02 11
info@iec.ch
www.iec.ch

About the IEC
The International Electrotechnical Commission (IEC) is the leading global organization that prepares and publishes International Standards for all electrical, electronic and related technologies.

About IEC publications
The technical content of IEC publications is kept under constant review by the IEC. Please make sure that you have the latest edition, a corrigenda or an amendment might have been published.

IEC Catalogue - webstore.iec.ch/catalogue
The stand-alone application for consulting the entire bibliographical information on IEC International Standards, Technical Specifications, Technical Reports and other documents. Available for PC, Mac OS, Android Tablets and iPad.

IEC publications search - webstore.iec.ch/advsearchform
The advanced search enables to find IEC publications by a variety of criteria (reference number, text, technical committee,…). It also gives information on projects, replaced and withdrawn publications.

IEC Just Published - webstore.iec.ch/justpublished
Stay up to date on all new IEC publications. Just Published details all new publications released. Available online and also once a month by email.

Electropedia - www.electropedia.org
The world's leading online dictionary of electronic and electrical terms containing 21 000 terms and definitions in English and French, with equivalent terms in 16 additional languages. Also known as the International Electrotechnical Vocabulary (IEV) online.

IEC Glossary - std.iec.ch/glossary
67 000 electrotechnical terminology entries in English and French extracted from the Terms and Definitions clause of IEC publications issued since 2002. Some entries have been collected from earlier publications of IEC TC 37, 77, 86 and CISPR.

IEC Customer Service Centre - webstore.iec.ch/csc
If you wish to give us your feedback on this publication or need further assistance, please contact the Customer Service Centre: sales@iec.ch.

ISO/IEC 14776-454

Edition 1.0 2018-04

INTERNATIONAL STANDARD

colour inside

Information technology – Small Computer System Interface (SCSI) – Part 454: SCSI Primary Commands – 4 (SPC-4)

INTERNATIONAL ELECTROTECHNICAL COMMISSION

ICS 35.200

ISBN 978-2-8322-5595-7

Warning! Make sure that you obtained this publication from an authorized distributor.

Contents

Page

1 Scope ... 40

2 Normative references ... 40

3 Terms, definitions, symbols, abbreviations, and conventions ... 45
3.1 Terms and definitions ... 45
3.2 Abbreviations and symbols ... 65
3.2.1 Abbreviations ... 65
3.2.2 Symbols ... 67
3.2.3 Mathematical operators ... 67
3.3 Keywords ... 68
3.4 Conventions ... 69
3.5 Numeric and character conventions ... 70
3.5.1 Numeric conventions ... 70
3.5.2 Units of measure ... 70
3.5.3 Byte encoded character strings conventions ... 71
3.6 Bit and byte ordering ... 71
3.7 Notation conventions ... 73
3.7.1 Notation for procedure calls ... 73
3.7.2 Notation for state diagrams ... 74
3.7.3 Notation for flowcharts ... 75
3.7.4 Notation for UML figures ... 75
3.7.4.1 Overview ... 75
3.7.4.2 Class notation ... 77
3.7.4.3 Class association relationships notation ... 77
3.7.4.4 Class aggregation relationships notation ... 78
3.7.4.5 Class generalization relationships notation ... 80
3.7.4.6 Class dependency relationships notation ... 81
3.7.4.7 Object notation ... 81
3.7.5 Notation for EXTENDED COPY command segment descriptors ... 81

4 General concepts ... 82
4.1 Introduction ... 82
4.2 Command Descriptor Block ... 82
4.2.1 CDB usage and structure ... 82
4.2.2 Fixed length CDB formats ... 83
4.2.2.1 Formats for 6-byte CDBs ... 83
4.2.2.1.1 Generic 6-byte CDB format ... 83
4.2.2.1.2 Typical 6-byte CDB format ... 83
4.2.2.2 Formats for 10-byte CDBs ... 84
4.2.2.2.1 Generic 10-byte CDB format ... 84
4.2.2.2.2 Typical 10-byte CDB format ... 85
4.2.2.3 Formats for 12-byte CDBs ... 86
4.2.2.3.1 Generic 12-byte CDB format ... 86
4.2.2.3.2 Typical 12-byte CDB format ... 86
4.2.2.3.3 MAINTENANCE IN CDB format ... 87
4.2.2.3.4 MAINTENANCE OUT CDB format ... 88
4.2.2.3.5 SERVICE ACTION IN(12) CDB format ... 88
4.2.2.3.6 SERVICE ACTION OUT(12) CDB format ... 89
4.2.2.4 Formats for 16-byte CDBs ... 89
4.2.2.4.1 Generic 16-byte CDB format ... 89
4.2.2.4.2 Typical 16-byte CDB format, if eight-byte LBAs not supported ... 90
4.2.2.4.3 Typical 16-byte CDB format with eight-byte LBAs supported ... 91
4.2.2.4.4 SERVICE ACTION IN(16) CDB format ... 92

4.2.2.4.5 SERVICE ACTION OUT(16) CDB format 92
4.2.2.4.6 SERVICE ACTION BIDIRECTIONAL CDB format 93
4.2.3 Variable length CDB formats 93
4.2.3.1 Generic variable length CDB format 93
4.2.3.2 Typical 32-byte variable length CDB format 95
4.2.4 Extended CDBs 96
4.2.4.1 XCDB model 96
4.2.4.2 The XCDB format 96
4.2.5 Common CDB fields 98
4.2.5.1 Operation code 98
4.2.5.2 Service action 98
4.2.5.3 Logical block address 98
4.2.5.4 Transfer length 99
4.2.5.5 Parameter list length 99
4.2.5.6 Allocation length 99
4.3 Data field requirements 99
4.3.1 ASCII data field requirements 99
4.3.2 Null data field termination and zero padding requirements 100
4.3.3 Variable type data field requirements 100
4.3.4 Port identifier field requirements 100
4.4 Secure random numbers 101
4.5 Sense data 101
4.5.1 Sense data introduction 101
4.5.2 Descriptor format sense data 102
4.5.2.1 Descriptor format sense data overview 102
4.5.2.2 Information sense data descriptor 104
4.5.2.3 Command-specific information sense data descriptor 105
4.5.2.4 Sense key specific sense data descriptor 106
4.5.2.4.1 Sense key specific sense data descriptor overview 106
4.5.2.4.2 Field pointer sense key specific information 107
4.5.2.4.3 Actual retry count sense key specific information 108
4.5.2.4.4 Progress indication sense key specific information 108
4.5.2.4.5 Segment pointer sense key specific information 109
4.5.2.4.6 Unit attention condition queue overflow sense key specific information 109
4.5.2.5 Field replaceable unit sense data descriptor 110
4.5.2.6 Another progress indication sense data descriptor 110
4.5.2.7 Forwarded sense data 111
4.5.2.8 Vendor specific sense data descriptors 112
4.5.3 Fixed format sense data 113
4.5.4 Returning a value in the INFORMATION field in the sense data 114
4.5.5 Returning a value in the COMMAND-SPECIFIC INFORMATION field in the sense data 115
4.5.6 Current information 115
4.5.7 Deferred errors 116
4.5.8 Sense key and additional sense code definitions 117

5 Model common to all device types 136
5.1 Introduction to the model common to all device types 136
5.1.1 Overview 136
5.1.2 Important commands for all SCSI device servers 136
5.1.2.1 Commands implemented by all SCSI device servers 136
5.1.2.2 Commands recommended for all SCSI device servers 136
5.1.2.3 Using the INQUIRY command 136
5.1.2.4 Using the REPORT LUNS command 136
5.1.2.5 Using the TEST UNIT READY command 137
5.1.2.6 Using the REQUEST SENSE command 137
5.1.3 Implicit head of queue 137
5.2 Device clocks and timestamps 137

5.3 Device specific background functions 138
5.3.1 Introduction 138
5.3.2 Suspending and resuming device specific background functions 139
5.4 Downloading and activating microcode 140
5.4.1 Downloading microcode 140
5.4.2 Activating microcode 144
5.5 Error history 145
5.5.1 Error history overview 145
5.5.2 Retrieving error history with the READ BUFFER command 145
5.5.3 Adding application client error history with the WRITE BUFFER command 148
5.5.4 Clearing error history with the WRITE BUFFER command 148
5.6 Identifying information 149
5.7 Medium auxiliary memory 149
5.8 Parameter rounding 150
5.9 Parsing variable length parameter lists and parameter data 151
5.10 Pollable condition information 152
5.10.1 Information that does not represent an exception condition 152
5.10.2 REQUEST SENSE pollable sense data 152
5.10.2.1 Making information available for the REQUEST SENSE command 152
5.10.2.2 Selecting pollable sense data to return 152
5.10.2.3 Returning one or more progress indications 152
5.10.3 Log parameter pollable device condition information 153
5.11 Power management 153
5.11.1 Power management overview 153
5.11.2 Power consumption management 153
5.11.3 Power conditions management 154
5.11.4 Active power condition 155
5.11.5 Idle power conditions 155
5.11.6 Standby power conditions 156
5.11.7 Power condition pollable sense data 156
5.11.8 Power condition state machine 157
5.11.8.1 Power condition state machine overview 157
5.11.8.2 PC0:Powered_On state 158
5.11.8.2.1 PC0:Powered_On state description 158
5.11.8.2.2 Transition PC0:Powered_On to PC4:Active_Wait 159
5.11.8.3 PC1:Active state 159
5.11.8.3.1 PC1:Active state description 159
5.11.8.3.2 Transition PC1:Active to PC5:Wait_Idle 159
5.11.8.3.3 Transition PC1:Active to PC6:Wait_Standby 159
5.11.8.4 PC2:Idle state 159
5.11.8.4.1 PC2:Idle state description 159
5.11.8.4.2 Transition PC2:Idle to PC4:Active_Wait 160
5.11.8.4.3 Transition PC2:Idle to PC5:Wait_Idle 160
5.11.8.4.4 Transition PC2:Idle to PC6:Wait_Standby 160
5.11.8.5 PC3:Standby state 160
5.11.8.5.1 PC3:Standby state description 160
5.11.8.5.2 Transition PC3:Standby to PC4:Active_Wait 161
5.11.8.5.3 Transition PC3:Standby to PC6:Wait_Standby 161
5.11.8.6 PC4:Active_Wait state 161
5.11.8.6.1 PC4:Active_Wait state description 161
5.11.8.6.2 Transition PC4:Active_Wait to PC1:Active 162
5.11.8.7 PC5:Wait_Idle state 162
5.11.8.7.1 PC5:Wait_Idle state description 162
5.11.8.7.2 Transition PC5:Wait_Idle to PC2:Idle 162
5.11.8.8 PC6:Wait_Standby state 163
5.11.8.8.1 PC6:Wait_Standby state description 163
5.11.8.8.2 Transition PC6:Wait_Standby to PC3:Standby 163

5.12 Reservations 163
5.12.1 Persistent Reservations overview 163
5.12.2 Third party persistent reservations 168
5.12.3 Exceptions to SPC-2 RESERVE and RELEASE behavior 168
5.12.4 Persistent reservations interactions with IKEv2-SCSI SA creation 169
5.12.5 Preserving persistent reservations and registrations 169
5.12.5.1 Requirements for preserving persistent reservations and registrations 169
5.12.5.2 Preserving persistent reservations and registrations through power loss 169
5.12.5.3 Nonvolatile memory considerations for preserving persistent reservations and registrations 170
5.12.5.4 Loss of persistent reservation information 170
5.12.5.4.1 Loss of persistent reservation information overview 170
5.12.5.4.2 Recoverable loss of persistent reservation information 170
5.12.5.4.3 Unrecoverable loss of persistent reservation information overview 171
5.12.6 Finding persistent reservations and reservation keys 171
5.12.6.1 Summary of commands for finding persistent reservations and reservation keys 171
5.12.6.2 Reporting reservation keys 171
5.12.6.3 Reporting the persistent reservation 171
5.12.6.4 Reporting full status 172
5.12.7 Registering 172
5.12.8 Registering and moving the reservation 176
5.12.9 Reserving 177
5.12.10 Persistent reservation holder 178
5.12.11 Releasing persistent reservations and removing registrations 178
5.12.11.1 Releasing persistent reservations, removing registrations, and lost reservation information 178
5.12.11.2 Service actions that release persistent reservations and remove registrations 179
5.12.11.2.1 Service actions that release persistent reservations and remove registrations overview 179
5.12.11.2.2 Releasing 180
5.12.11.2.3 Unregistering 180
5.12.11.2.4 Preempting 182
5.12.11.2.4.1 Commands that preempt reservations 182
5.12.11.2.4.2 Failed persistent reservation preempt 184
5.12.11.2.4.3 Preempting persistent reservations and registration handling 184
5.12.11.2.5 Removing registrations 185
5.12.11.2.6 Preempting and aborting 185
5.12.11.2.7 Clearing 186
5.12.11.3 Replacing lost reservations 187
5.13 Security features 188
5.13.1 Security goals and threat model 188
5.13.1.1 Introduction 188
5.13.1.2 Security goals 188
5.13.1.3 Threat model 189
5.13.1.4 Types of attacks 189
5.13.1.5 SCSI security considerations 190
5.13.2 Security associations 190
5.13.2.1 Principles of SAs 190
5.13.2.2 SA parameters 192
5.13.2.3 Creating an SA 194
5.13.3 Key derivation functions 195
5.13.3.1 KDFs overview 195
5.13.3.2 IKEv2-based iterative KDF 195
5.13.3.3 HMAC-based KDFs 196
5.13.3.4 AES-XCBC-PRF-128 IKEv2-based iterative KDF 197
5.13.4 Using IKEv2-SCSI to create an SA 198
5.13.4.1 Overview 198
5.13.4.2 IKEv2-SCSI Protocol summary 201
5.13.4.3 IKEv2-SCSI Authentication 204
5.13.4.3.1 Overview 204

5.13.4.3.2 Pre-shared key authentication 205
5.13.4.3.3 Digital signature authentication 205
5.13.4.3.3.1 Overview 205
5.13.4.3.3.2 Certificates and digital signature authentication 206
5.13.4.3.3.3 Example of certificate use for digital signature authentication 206
5.13.4.3.3.4 Handling of the Certificate Request payload and the Certificate payload 207
5.13.4.3.4 Constraints on skipping the Authentication step 207
5.13.4.4 Summary of IKEv2-SCSI shared keys nomenclature and shared key sizes 209
5.13.4.5 Device Server Capabilities step 210
5.13.4.6 IKEv2-SCSI Key Exchange step 212
5.13.4.6.1 Overview 212
5.13.4.6.2 Key Exchange step SECURITY PROTOCOL OUT command 212
5.13.4.6.3 Key Exchange step SECURITY PROTOCOL IN command 213
5.13.4.6.4 Key Exchange step completion 214
5.13.4.6.5 After the Key Exchange step 214
5.13.4.7 IKEv2-SCSI Authentication step 214
5.13.4.7.1 Overview 214
5.13.4.7.2 Authentication step SECURITY PROTOCOL OUT command 215
5.13.4.7.3 Authentication step SECURITY PROTOCOL IN command 216
5.13.4.8 Generating shared keys 217
5.13.4.8.1 Overview 217
5.13.4.8.2 Generating shared keys when the Authentication step is skipped 218
5.13.4.8.3 Generating shared keys when the Authentication step is processed 218
5.13.4.8.4 Initializing shared key generation 218
5.13.4.8.4.1 Initializing for SA creation shared key generation 218
5.13.4.8.4.2 Initializing for generation of shared keys used by the created SA 219
5.13.4.8.5 Generating shared keys used for SA management 219
5.13.4.8.6 Generating shared keys for use by the created SA 220
5.13.4.9 IKEv2-SCSI SA generation 221
5.13.4.10 Abandoning an IKEv2-SCSI CCS 222
5.13.4.11 Deleting an IKEv2-SCSI SA 223
5.13.5 Security progress indication 223
5.13.6 Command security 224
5.13.6.1 Overview 224
5.13.6.2 Secure CDB Originator class 224
5.13.6.3 Secure CDB Processor class 224
5.13.6.4 Enforcement Manager class 225
5.13.6.5 Security Manager class 225
5.13.6.6 The relationship between SAs and command security 226
5.13.6.7 Capability-based command security technique 226
5.13.6.7.1 Overview 226
5.13.6.7.2 Security Manager class 229
5.13.6.7.3 CbCS Management Device Server class 230
5.13.6.7.3.1 CbCS Management Device Server class overview 230
5.13.6.7.3.2 Decision Database attribute 230
5.13.6.7.4 CbCS Management Application Client class 230
5.13.6.7.5 Secure CDB Originator class 230
5.13.6.7.6 Secure CDB Processor class 230
5.13.6.7.7 Enforcement Manager class 231
5.13.6.7.8 CbCS methods 232
5.13.6.7.8.1 Overview 232
5.13.6.7.8.2 The BASIC CbCS method 232
5.13.6.7.8.3 The CAPKEY CbCS method 233
5.13.6.7.9 CbCS trust assumptions 233
5.13.6.7.10 CbCS security tokens 234
5.13.6.7.11 CbCS shared keys 235
5.13.6.7.11.1 Overview 235

5.13.6.7.11.2 CbCS shared key identifiers 236
5.13.6.7.11.3 Specifying which CbCS shared key to change 236
5.13.6.7.11.4 Updating a CbCS master key 237
5.13.6.7.11.5 Changing a CbCS working keys 237
5.13.6.7.12 CbCS credentials 237
5.13.6.7.12.1 Overview 237
5.13.6.7.12.2 CbCS capability key computations for the secure CDB originator 238
5.13.6.7.12.3 CbCS capability key computations for general use 238
5.13.6.7.13 CbCS capability descriptors 239
5.13.6.7.13.1 Overview 239
5.13.6.7.13.2 CbCS extension descriptor validation 239
5.13.6.7.13.3 CAPKEY CbCS method capability integrity validation 240
5.13.6.7.14 Association between commands and permission bits 241
5.13.6.7.15 CbCS parameters 244
5.13.6.7.15.1 Overview 244
5.13.6.7.16 CbCS extension descriptor format 246
5.13.7 ESP-SCSI encapsulations for parameter data 247
5.13.7.1 Overview 247
5.13.7.2 ESP-SCSI required inputs 247
5.13.7.3 ESP-SCSI data format before encryption and after decryption 248
5.13.7.4 ESP-SCSI outbound data descriptors 249
5.13.7.4.1 Overview 249
5.13.7.4.2 ESP-SCSI CDBs or Data-Out Buffer parameter lists including a descriptor length 250
5.13.7.4.2.1 Initialization vector absent 250
5.13.7.4.2.2 Initialization vector present 252
5.13.7.4.3 ESP-SCSI Data-Out Buffer parameter lists for externally specified descriptor length 253
5.13.7.4.3.1 Initialization vector absent 253
5.13.7.4.3.2 Initialization vector present 254
5.13.7.5 ESP-SCSI Data-In Buffer parameter data descriptors 254
5.13.7.5.1 Overview 254
5.13.7.5.2 ESP-SCSI Data-In Buffer parameter data including a descriptor length 255
5.13.7.5.2.1 Initialization vector absent 255
5.13.7.5.2.2 Initialization vector present 257
5.13.7.5.3 ESP-SCSI Data-In Buffer parameter data for externally specified descriptor length 258
5.13.7.5.3.1 Initialization vector absent 258
5.13.7.5.3.2 Initialization vector present 259
5.13.8 Security algorithm codes 260
5.14 Self-test operations 261
5.14.1 Self-test types 261
5.14.2 Default self-test 262
5.14.3 The short self-test and extended self-test 262
5.14.4 Self-test modes 263
5.14.4.1 Self-test modes overview 263
5.14.4.2 Foreground mode 263
5.14.4.3 Background mode 263
5.14.4.4 Features common to foreground and background self-test modes 264
5.15 Target port group asymmetric access states 267
5.15.1 Target port group access overview 267
5.15.2 Asymmetric logical unit access 267
5.15.2.1 Introduction to asymmetric logical unit access 267
5.15.2.2 Explicit and implicit asymmetric logical unit access 268
5.15.2.3 Discovery of asymmetric logical unit access behavior 268
5.15.2.4 Target port asymmetric access states 269
5.15.2.4.1 Target port asymmetric access states overview 269
5.15.2.4.2 Active/optimized state 269
5.15.2.4.3 Active/non-optimized state 269
5.15.2.4.4 Standby state 269

5.15.2.4.5 Unavailable state 270
5.15.2.4.6 Offline state 271
5.15.2.4.7 Logical block dependent state 271
5.15.2.5 Transitions between target port asymmetric access states 271
5.15.2.6 Preference indicator 272
5.15.2.7 Target port asymmetric access state reporting 272
5.15.2.8 Implicit asymmetric logical units access management 273
5.15.2.9 Explicit asymmetric logical units access management 273
5.15.2.10 Behavior after power on, hard reset, logical unit reset, and I_T nexus loss 274
5.15.2.11 Behavior of target ports that are not accessible from the service delivery subsystem 274
5.15.3 Symmetric logical unit access 274
5.16 Third-party copies 274
5.16.1 General considerations for third-party copies 274
5.16.2 Copy manager model 275
5.16.3 Third-party copy commands 278
5.16.4 Third-party copy command usage 279
5.16.4.1 Prior to sending a third-party copy command 279
5.16.4.2 List identifiers for third-party copy commands 280
5.16.4.3 Third-party copy commands and operations 280
5.16.4.4 Monitoring progress of and retrieving results from third-party copy commands 281
5.16.4.5 Held data 282
5.16.4.6 Aborting third-party copy commands and copy operations 283
5.16.4.7 The COPY OPERATION ABORT command 283
5.16.5 Responses to the conditions that result from SCSI events 284
5.16.6 RODs and ROD tokens 284
5.16.6.1 RODs and ROD related tokens overview 284
5.16.6.2 ROD types 285
5.16.6.2.1 ROD types overview 285
5.16.6.2.2 Access upon reference type RODs 285
5.16.6.2.3 Point in time copy RODs 286
5.16.6.2.3.1 Point in time copy RODs overview 286
5.16.6.2.3.2 Point in time copy – default type RODs 286
5.16.6.2.3.3 Point in time copy – change vulnerable type RODs 286
5.16.6.2.3.4 Point in time copy – persistent type RODs 286
5.16.6.2.3.5 Point in time copy – any type RODs 286
5.16.6.3 Populating a ROD or ROD token 287
5.16.6.4 ROD token format 288
5.16.6.5 Generic ROD tokens 290
5.16.6.5.1 Generic ROD token format 290
5.16.6.5.2 Validating generic ROD tokens 292
5.16.6.5.2.1 Overview of validating generic ROD tokens 292
5.16.6.5.2.2 Inexact validation of generic ROD tokens 293
5.16.6.5.2.3 Validation errors for generic ROD tokens 293
5.16.6.6 ROD token usage 295
5.16.6.7 ROD token lifetime 296
5.16.7 The EXTENDED COPY command 297
5.16.7.1 EXTENDED COPY parameter list 297
5.16.7.2 EXTENDED COPY command processing 297
5.16.7.3 EXTENDED COPY command errors detected before segment descriptor processing starts 302
5.16.7.4 EXTENDED COPY command errors detected during processing of segment descriptors 302
5.16.7.5 EXTENDED COPY considerations for RODs and ROD tokens 304
5.16.7.5.1 EXTENDED COPY command CSCD ROD identifiers 304
5.16.7.5.2 Populating an EXTENDED COPY command ROD 305
5.16.7.6 EXTENDED COPY command use of RODs when the peripheral device type is 00h (i.e., block device) 306
5.16.7.7 EXTENDED COPY command interactions with aliases 306

6 Commands for all device types 307
6.1 Summary of commands for all device types 307
6.2 CHANGE ALIASES command 309
6.2.1 CHANGE ALIASES command introduction 309
6.2.2 Alias entry format 311
6.2.3 Alias designation validation 312
6.2.4 Alias entry protocol independent designations 312
6.2.4.1 Alias entry protocol independent designations overview 312
6.2.4.2 NULL DESIGNATION alias format 312
6.3 COPY OPERATION ABORT command 313
6.4 EXTENDED COPY(LID4) command 313
6.4.1 EXTENDED COPY(LID4) command introduction 313
6.4.2 EXTENDED COPY(LID4) parameter data 315
6.4.3 Shared EXTENDED COPY parameter list fields 317
6.4.3.1 STR bit 317
6.4.3.2 LIST IDENTIFIER field and LIST ID USAGE field 318
6.4.3.3 PRIORITY field 320
6.4.3.4 CSCD DESCRIPTOR LIST LENGTH field and CSCD descriptor list 320
6.4.3.5 SEGMENT DESCRIPTOR LIST LENGTH field and segment descriptor list 320
6.4.3.6 INLINE DATA LENGTH field and inline data 321
6.4.4 Descriptor type codes 321
6.4.5 CSCD descriptors 322
6.4.5.1 CSCD descriptors introduction 322
6.4.5.2 The CSCD descriptor extension 324
6.4.5.3 Device type specific CSCD descriptor parameters for block device types 325
6.4.5.4 Device type specific CSCD descriptor parameters for sequential-access device types 326
6.4.5.5 Device type specific CSCD descriptor parameters for processor device types 327
6.4.5.6 Identification Descriptor CSCD descriptor format 328
6.4.5.7 Alias CSCD descriptor format 329
6.4.5.8 IP Copy Service CSCD descriptor 330
6.4.5.9 ROD CSCD descriptor 332
6.4.6 Segment descriptors 337
6.4.6.1 Segment descriptors introduction 337
6.4.6.2 Block device to stream device functions 340
6.4.6.3 Stream device to block device functions 342
6.4.6.4 Block device to block device functions 343
6.4.6.5 Stream device to stream device functions 345
6.4.6.6 Inline data to stream device function 346
6.4.6.7 Embedded data to stream device function 348
6.4.6.8 Stream device to discard functions 349
6.4.6.9 Verify CSCD function 350
6.4.6.10 Block device with offset to stream device function 352
6.4.6.11 Stream device to block device with offset function 353
6.4.6.12 Block device with offset to block device with offset function 355
6.4.6.13 Write filemarks function 356
6.4.6.14 Space function 357
6.4.6.15 Locate function 358
6.4.6.16 Tape device image copy function 359
6.4.6.17 Register persistent reservation key function 360
6.4.6.18 Third party persistent reservations source I_T nexus function 361
6.4.6.19 Block device image copy function 362
6.4.6.20 Populate a ROD from one or more block ranges function 364
6.4.6.21 Populate a ROD from one block range function 366
6.5 EXTENDED COPY(LID1) command 367
6.6 INQUIRY command 370
6.6.1 INQUIRY command introduction 370
6.6.2 Standard INQUIRY data 371

6.6.3 SCSI Parallel Interface specific INQUIRY data 387
6.7 LOG SELECT command 389
6.7.1 Introduction 389
6.7.2 Processing LOG SELECT when the parameter list length is zero 391
6.8 LOG SENSE command 394
6.9 MANAGEMENT PROTOCOL IN command 396
6.9.1 MANAGEMENT PROTOCOL IN command description 396
6.9.2 Management protocol information description 397
6.9.2.1 Overview 397
6.9.2.2 CDB description 397
6.9.2.3 Supported management protocols list description 398
6.10 MANAGEMENT PROTOCOL OUT command 399
6.11 MODE SELECT(6) command 400
6.12 MODE SELECT(10) command 402
6.13 MODE SENSE(6) command 403
6.13.1 MODE SENSE(6) command introduction 403
6.13.2 Current values 404
6.13.3 Changeable values 405
6.13.4 Default values 405
6.13.5 Saved values 405
6.13.6 Initial responses 405
6.14 MODE SENSE(10) command 406
6.15 PERSISTENT RESERVE IN command 407
6.15.1 PERSISTENT RESERVE IN command introduction 407
6.15.2 READ KEYS service action 408
6.15.3 READ RESERVATION service action 408
6.15.3.1 READ RESERVATION service action operation 408
6.15.3.2 Persistent reservations scope 410
6.15.3.3 Persistent reservations type 411
6.15.4 REPORT CAPABILITIES service action 412
6.15.5 READ FULL STATUS service action 416
6.16 PERSISTENT RESERVE OUT command 418
6.16.1 PERSISTENT RESERVE OUT command introduction 418
6.16.2 PERSISTENT RESERVE OUT service actions and parameter list formats 420
6.16.3 Basic PERSISTENT RESERVE OUT parameter list 423
6.16.4 Parameter list for the PERSISTENT RESERVE OUT command with REGISTER AND MOVE service action 426
6.17 READ ATTRIBUTE command 428
6.17.1 READ ATTRIBUTE command introduction 428
6.17.2 ATTRIBUTE VALUES service action 430
6.17.3 ATTRIBUTE LIST service action 431
6.17.4 LOGICAL VOLUME LIST service action 431
6.17.5 PARTITION LIST service action 432
6.17.6 SUPPORTED ATTRIBUTES service action 432
6.18 READ BUFFER command 433
6.18.1 READ BUFFER command introduction 433
6.18.2 Vendor specific mode (01h) 434
6.18.3 Data mode (02h) 434
6.18.4 Descriptor mode (03h) 435
6.18.5 Read data from echo buffer mode (0Ah) 436
6.18.6 Echo buffer descriptor mode (0Bh) 436
6.18.7 Error history mode (1Ch) 437
6.18.7.1 Error history overview 437
6.18.7.2 Error history directory 438
6.18.7.3 Error history data buffer 441
6.18.7.4 Clear error history I_T nexus 441
6.18.7.5 Clear error history I_T nexus and release snapshot 441

6.19 READ MEDIA SERIAL NUMBER command 442
6.20 RECEIVE COPY DATA(LID4) command 443
6.21 RECEIVE COPY DATA(LID1) command 445
6.22 RECEIVE COPY OPERATING PARAMETERS command 446
6.23 RECEIVE COPY FAILURE DETAILS(LID1) command 450
6.24 RECEIVE COPY STATUS(LID4) command 452
6.25 RECEIVE COPY STATUS(LID1) command 456
6.26 RECEIVE CREDENTIAL command 458
6.26.1 RECEIVE CREDENTIAL command description 458
6.26.1.1 Overview 458
6.26.1.2 CbCS logical unit credential request descriptor 460
6.26.1.3 CbCS logical unit and volume credential request descriptor 460
6.26.2 RECEIVE CREDENTIAL parameter data 461
6.26.2.1 RECEIVE CREDENTIAL parameter data encryption 461
6.26.2.2 RECEIVE CREDENTIAL decrypted parameter data 461
6.26.2.3 CbCS capability descriptor 462
6.26.2.3.1 Overview 462
6.26.2.3.2 Logical unit designation descriptor format 465
6.26.2.3.3 Volume designation descriptors 465
6.27 RECEIVE DIAGNOSTIC RESULTS command 466
6.28 RECEIVE ROD TOKEN INFORMATION command 467
6.29 REMOVE I_T NEXUS command 470
6.30 REPORT ALIASES command 472
6.31 REPORT ALL ROD TOKENS command 474
6.32 REPORT IDENTIFYING INFORMATION command 475
6.32.1 REPORT IDENTIFYING INFORMATION command overview 475
6.32.2 Peripheral device identifying information parameter data 477
6.32.3 Identifying information supported parameter data 477
6.33 REPORT LUNS command 478
6.34 REPORT PRIORITY command 481
6.35 REPORT SUPPORTED OPERATION CODES command 483
6.35.1 REPORT SUPPORTED OPERATION CODES command introduction 483
6.35.2 All_commands parameter data format 485
6.35.3 One_command parameter data format 486
6.35.4 Command timeouts descriptor 487
6.35.4.1 Overview 487
6.35.4.2 WRITE BUFFER command timeouts descriptor COMMAND SPECIFIC field usage 488
6.36 REPORT SUPPORTED TASK MANAGEMENT FUNCTIONS command 489
6.37 REPORT TARGET PORT GROUPS command 493
6.38 REPORT TIMESTAMP command 498
6.39 REQUEST SENSE command 499
6.40 SECURITY PROTOCOL IN command 501
6.41 SECURITY PROTOCOL OUT command 503
6.42 SEND DIAGNOSTIC command 505
6.43 SET IDENTIFYING INFORMATION command 511
6.44 SET PRIORITY command 513
6.45 SET TARGET PORT GROUPS command 515
6.46 SET TIMESTAMP command 519
6.47 TEST UNIT READY command 520
6.48 WRITE ATTRIBUTE command 522
6.49 WRITE BUFFER command 524
6.49.1 WRITE BUFFER command introduction 524
6.49.2 Vendor specific mode (01h) 525
6.49.3 Data mode (02h) 526
6.49.4 Download microcode and activate mode (04h) 526
6.49.5 Download microcode, save, and activate mode (05h) 526
6.49.6 Download microcode with offsets and activate mode (06h) 526

6.49.7 Download microcode with offsets, save, and activate mode (07h) 527
6.49.8 Write data to echo buffer mode (0Ah) 527
6.49.9 Download microcode with offsets, select activation, save, and defer activate mode (0Dh) 527
6.49.10 Download microcode with offsets, save, and defer activate mode (0Eh) 528
6.49.11 Activate deferred microcode mode (0Fh) 528
6.49.12 Download application client error history mode (1Ch) 529

7 Parameters for all device types 533
7.1 Overview 533
7.2 Diagnostic parameters 533
7.2.1 Summary of diagnostic page codes 533
7.2.2 Diagnostic page format for all device types 533
7.2.3 Protocol Specific diagnostic page 535
7.2.4 Supported Diagnostic Pages diagnostic page 535
7.3 Log parameters 537
7.3.1 Summary of log page codes 537
7.3.2 Log page structure and log parameter structure for all device types 538
7.3.2.1 Log page structure 538
7.3.2.2 Log parameter structure 540
7.3.2.2.1 Introduction 540
7.3.2.2.2 Parameter control byte 541
7.3.2.2.2.1 Introduction 541
7.3.2.2.2.2 Parameter control byte values for bounded data counter parameters 543
7.3.2.2.2.3 Parameter control byte values for unbounded data counter parameters 544
7.3.2.2.2.4 Parameter control byte values for ASCII format list log parameters 545
7.3.2.2.2.5 Parameter control byte values for binary format list log parameters 546
7.3.3 Resetting and setting log parameters 547
7.3.4 Application Client log page 547
7.3.4.1 Overview 547
7.3.4.2 General Usage Application Client log parameter 548
7.3.5 Buffer Over-Run/Under-Run log page 549
7.3.5.1 Overview 549
7.3.5.2 Buffer Over-run/Under-run log parameter 551
7.3.6 Cache Memory Statistics log page 552
7.3.6.1 Overview 552
7.3.6.2 Read Cache Memory Hits log parameter 553
7.3.6.3 Reads To Cache Memory log parameter 554
7.3.6.4 Write Cache Memory Hits log parameter 555
7.3.6.5 Writes From Cache Memory log parameter 556
7.3.6.6 Time From Last Hard Reset log parameter 557
7.3.6.7 Time Interval log parameter 558
7.3.7 General Statistics and Performance log pages 559
7.3.7.1 Overview 559
7.3.7.2 General Access Statistics and Performance log parameter 561
7.3.7.3 Idle Time log parameter 563
7.3.7.4 Force Unit Access Statistics and Performance log parameter 564
7.3.8 Group Statistics and Performance (n) log pages 565
7.3.8.1 Overview 565
7.3.8.2 Group n Statistics and Performance log parameter 568
7.3.8.3 Group n Force Unit Access Statistics and Performance log parameter 569
7.3.9 Informational Exceptions log page 571
7.3.9.1 Overview 571
7.3.9.2 Informational Exceptions General log parameter 572
7.3.10 Last n Deferred Errors or Asynchronous Events log page 573
7.3.10.1 Overview 573
7.3.10.2 Deferred Error or Asynchronous Event log parameters 574
7.3.11 Last n Error Events log page 575

7.3.11.1 Overview ... 575
7.3.11.2 Error Event log parameters ... 576
7.3.12 Non-Medium Error log page ... 576
7.3.12.1 Overview ... 576
7.3.12.2 Non-Medium Error Count log parameter ... 577
7.3.13 Power Condition Transitions log page ... 578
7.3.13.1 Overview ... 578
7.3.13.2 Accumulated Transitions log parameter ... 579
7.3.14 Protocol Specific Port log page ... 580
7.3.14.1 Overview ... 580
7.3.14.2 Generic protocol specific port log parameter ... 581
7.3.15 Read Error Counters log page ... 582
7.3.15.1 Overview ... 582
7.3.15.2 Read Error Counter log parameter ... 583
7.3.16 Read Reverse Error Counters log page ... 584
7.3.16.1 Overview ... 584
7.3.16.2 Read Reverse Error Counter log parameter ... 585
7.3.17 Self-Test Results log page ... 586
7.3.17.1 Overview ... 586
7.3.17.2 Self-Test Results log parameters ... 588
7.3.18 Start-Stop Cycle Counter log page ... 590
7.3.18.1 Overview ... 590
7.3.18.2 Date of Manufacture log parameter ... 591
7.3.18.3 Accounting Date log parameter ... 592
7.3.18.4 Specified Cycle Count Over Device Lifetime log parameter ... 593
7.3.18.5 Accumulated Start-Stop Cycles log parameter ... 593
7.3.18.6 Specified Load-Unload Count Over Device Lifetime log parameter ... 594
7.3.18.7 Accumulated Load-Unload Cycles log parameter ... 595
7.3.19 Supported Log Pages log page ... 596
7.3.20 Supported Log Pages and Subpages log page ... 597
7.3.21 Supported Subpages log page ... 598
7.3.22 Temperature log page ... 599
7.3.22.1 Overview ... 599
7.3.22.2 Temperature log parameter ... 600
7.3.22.3 Reference Temperature log parameter ... 600
7.3.23 Verify Error Counters log page ... 601
7.3.23.1 Overview ... 601
7.3.23.2 Verify Error Counter log parameter ... 602
7.3.24 Write Error Counters log page ... 603
7.3.24.1 Overview ... 603
7.3.24.2 Write Error Counter log parameter ... 604
7.4 Medium auxiliary memory attributes ... 606
7.4.1 Attribute format ... 606
7.4.2 Attribute identifier values ... 607
7.4.2.1 Attribute identifier values overview ... 607
7.4.2.2 Device type attributes ... 607
7.4.2.2.1 REMAINING CAPACITY IN PARTITION and MAXIMUM CAPACITY IN PARTITION ... 608
7.4.2.2.2 LOAD COUNT ... 608
7.4.2.2.3 MAM SPACE REMAINING ... 608
7.4.2.2.4 INITIALIZATION COUNT ... 609
7.4.2.2.5 VOLUME IDENTIFIER ... 609
7.4.2.2.6 DEVICE VENDOR/SERIAL NUMBER AT LAST LOAD, DEVICE VENDOR/SERIAL NUMBER AT LOAD –1, DEVICE VENDOR/SERIAL NUMBER AT LOAD –2 and DEVICE VENDOR/SERIAL NUMBER AT LOAD –3 ... 609
7.4.2.2.7 TOTAL MEBIBYTES WRITTEN IN MEDIUM LIFE and TOTAL MEBIBYTES READ IN MEDIUM LIFE ... 609
7.4.2.2.8 TOTAL MEBIBYTES WRITTEN IN CURRENT/LAST LOAD and TOTAL MEBIBYTES

READ IN CURRENT/LAST LOAD 609
7.4.2.2.9 LOGICAL POSITION OF FIRST ENCRYPTED BLOCK 610
7.4.2.2.10 LOGICAL POSITION OF FIRST UNENCRYPTED BLOCK AFTER THE FIRST ENCRYPTED BLOCK 610
7.4.2.2.11 MEDIUM USAGE HISTORY 610
7.4.2.2.12 PARTITION USAGE HISTORY 613
7.4.2.3 Medium type attributes 615
7.4.2.3.1 MEDIUM MANUFACTURER 615
7.4.2.3.2 MEDIUM SERIAL NUMBER 615
7.4.2.3.3 MEDIUM MANUFACTURE DATE 615
7.4.2.3.4 MAM CAPACITY 616
7.4.2.3.5 MEDIUM TYPE and MEDIUM TYPE INFORMATION 616
7.4.2.3.6 NUMERIC MEDIUM SERIAL NUMBER 616
7.4.2.4 Host type attributes 617
7.4.2.4.1 APPLICATION VENDOR 617
7.4.2.4.2 APPLICATION NAME 617
7.4.2.4.3 APPLICATION VERSION 617
7.4.2.4.4 USER MEDIUM TEXT LABEL 617
7.4.2.4.5 DATE & TIME LAST WRITTEN 618
7.4.2.4.6 TEXT LOCALIZATION IDENTIFIER 618
7.4.2.4.7 BARCODE 618
7.4.2.4.8 OWNING HOST TEXTUAL NAME 618
7.4.2.4.9 MEDIA POOL 618
7.4.2.4.10 PARTITION USER TEXT LABEL 618
7.4.2.4.11 LOAD/UNLOAD AT PARTITION 618
7.4.2.4.12 APPLICATION FORMAT VERSION 619
7.5 Mode parameters 620
7.5.1 Summary of mode page codes 620
7.5.2 Mode page policies 620
7.5.3 Mode parameters overview 621
7.5.4 Mode parameter list format 621
7.5.5 Mode parameter header formats 621
7.5.6 Mode parameter block descriptor formats 623
7.5.6.1 General block descriptor format 623
7.5.7 Mode page and subpage formats and page codes 624
7.5.8 Control mode page 625
7.5.9 Control Extension mode page 630
7.5.10 Disconnect-Reconnect mode page 631
7.5.11 Extended mode page 634
7.5.12 Extended Device-Type Specific mode page 634
7.5.13 Power Condition mode page 635
7.5.14 Power Consumption mode page 640
7.5.15 Protocol Specific Logical Unit mode page 640
7.5.16 Protocol Specific Port mode page 641
7.6 Protocol specific parameters 643
7.6.1 Protocol specific parameters introduction 643
7.6.2 Alias entry protocol specific designations 643
7.6.2.1 Introduction to alias entry protocol specific designations 643
7.6.2.2 Fibre Channel specific alias entry formats 643
7.6.2.2.1 Summary of Fibre Channel specific alias entry formats 643
7.6.2.2.2 Fibre Channel world wide port name alias entry format 644
7.6.2.2.3 Fibre Channel world wide port name with N_Port checking alias entry format 645
7.6.2.3 RDMA specific alias entry formats 646
7.6.2.3.1 Summary of RDMA specific alias entry formats 646
7.6.2.3.2 RDMA target port identifier alias entry format 646
7.6.2.3.3 InfiniBand global identifier with target port identifier checking alias entry format 647
7.6.2.4 Internet SCSI specific alias entry formats 648

7.6.2.4.1 Summary of Internet SCSI specific alias entry formats 648
7.6.2.4.2 iSCSI name alias entry format 649
7.6.2.4.3 iSCSI name with binary IPv4 address alias entry format 650
7.6.2.4.4 iSCSI name with IPname alias entry format 652
7.6.2.4.5 iSCSI name with binary IPv6 address alias entry format 654
7.6.3 EXTENDED COPY protocol specific CSCD descriptors 655
7.6.3.1 Introduction to EXTENDED COPY protocol specific CSCD descriptors 655
7.6.3.2 Fibre Channel N_Port_Name CSCD descriptor format 655
7.6.3.3 Fibre Channel N_Port_ID CSCD descriptor format 656
7.6.3.4 Fibre Channel N_Port_ID With N_Port_Name Checking CSCD descriptor format 657
7.6.3.5 SCSI Parallel T_L CSCD descriptor format 658
7.6.3.6 IEEE 1394 EUI-64 CSCD descriptor format 659
7.6.3.7 RDMA CSCD descriptor format 660
7.6.3.8 iSCSI IPv4 CSCD descriptor format 661
7.6.3.9 iSCSI IPv6 CSCD descriptor format 662
7.6.3.10 SAS Serial SCSI Protocol CSCD descriptor format 663
7.6.4 TransportID identifiers 664
7.6.4.1 Overview of TransportID identifiers 664
7.6.4.2 TransportID for initiator ports using SCSI over Fibre Channel 665
7.6.4.3 TransportID for initiator ports using a parallel SCSI bus 665
7.6.4.4 TransportID for initiator ports using SCSI over IEEE 1394 666
7.6.4.5 TransportID for initiator ports using SCSI over an RDMA interface 666
7.6.4.6 TransportID for initiator ports using SCSI over iSCSI 667
7.6.4.7 TransportID for initiator ports using SCSI over SAS Serial SCSI Protocol 669
7.6.4.8 TransportID for initiator ports using SCSI over PCI Express 669
7.7 Security protocol parameters 670
7.7.1 Security protocol information description 670
7.7.1.1 Overview 670
7.7.1.2 CDB description 670
7.7.1.3 Supported security protocols list description 671
7.7.1.4 Certificate data description 672
7.7.1.4.1 Certificate overview 672
7.7.1.4.2 Public Key certificate description 672
7.7.1.4.3 Attribute certificate description 672
7.7.1.5 Security compliance information description 673
7.7.1.5.1 Security compliance information overview 673
7.7.1.5.2 Compliance descriptor overview 674
7.7.1.5.3 FIPS 140 compliance descriptor 675
7.7.2 SA creation capabilities 676
7.7.2.1 Overview 676
7.7.2.2 SA creation capabilities CDB description 676
7.7.2.3 SA creation capabilities parameter data formats 677
7.7.2.3.1 Supported device server capabilities formats parameter data format 677
7.7.2.3.2 IKEv2-SCSI device server capabilities parameter data format 677
7.7.3 IKEv2-SCSI 678
7.7.3.1 Overview 678
7.7.3.2 IKEv2-SCSI SECURITY PROTOCOL IN CDB description 678
7.7.3.3 IKEv2-SCSI SECURITY PROTOCOL OUT CDB description 679
7.7.3.4 IKEv2-SCSI parameter data format 680
7.7.3.5 IKEv2-SCSI payloads 687
7.7.3.5.1 IKEv2-SCSI payload format 687
7.7.3.5.2 No Next payload 688
7.7.3.5.3 Key Exchange payload 689
7.7.3.5.4 Identification – Application Client payload and Identification – Device Server payload 690
7.7.3.5.5 Certificate payload 691
7.7.3.5.6 Certificate Request payload 692
7.7.3.5.7 Authentication payload 693

7.7.3.5.8 Nonce payload 695
7.7.3.5.9 Notify payload 696
7.7.3.5.10 Delete payload 697
7.7.3.5.11 Encrypted payload 698
7.7.3.5.11.1 Combined mode encryption 698
7.7.3.5.11.2 Encrypted payload introduction 699
7.7.3.5.11.3 IKEv2-SCSI AAD 701
7.7.3.5.11.4 Processing a received Encrypted payload 702
7.7.3.5.12 IKEv2-SCSI SA Creation Capabilities payload 704
7.7.3.5.13 IKEv2-SCSI SA Cryptographic Algorithms payload 705
7.7.3.5.14 IKEv2-SCSI SAUT Cryptographic Algorithms payload 707
7.7.3.5.15 IKEv2-SCSI Timeout Values payload 708
7.7.3.6 IKEv2-SCSI cryptographic algorithm descriptors 709
7.7.3.6.1 Overview 709
7.7.3.6.2 ENCR IKEv2-SCSI cryptographic algorithm descriptor 711
7.7.3.6.3 PRF IKEv2-SCSI cryptographic algorithm descriptor 713
7.7.3.6.4 INTEG IKEv2-SCSI cryptographic algorithm descriptor 715
7.7.3.6.5 D-H IKEv2-SCSI cryptographic algorithm descriptor 716
7.7.3.6.6 IKEv2-SCSI authentication algorithm IKEv2-SCSI cryptographic algorithm descriptor 718
7.7.3.7 Errors in IKEv2-SCSI security protocol commands 721
7.7.3.8 Errors in IKEv2-SCSI security protocol parameter data 723
7.7.3.8.1 Overview 723
7.7.3.8.2 Errors with high denial of service attack potential 723
7.7.3.8.3 Errors with low denial of service attack potential 724
7.7.3.9 Translating IKEv2 errors 724
7.7.4 CbCS security protocol 725
7.7.4.1 Overview 725
7.7.4.2 CbCS SECURITY PROTOCOL IN CDB description 726
7.7.4.3 CbCS SECURITY PROTOCOL IN parameter data 727
7.7.4.3.1 Supported CbCS SECURITY PROTOCOL IN Pages CbCS page 727
7.7.4.3.2 Supported CbCS SECURITY PROTOCOL OUT pages CbCS page 728
7.7.4.3.3 Unchangeable CbCS Parameters CbCS page 728
7.7.4.3.4 Security Token CbCS page 731
7.7.4.3.5 Current CbCS Parameters CbCS page 731
7.7.4.3.6 Set Master Key – Seed Exchange CbCS page 734
7.7.4.4 CbCS SECURITY PROTOCOL OUT CDB description 736
7.7.4.5 CbCS SECURITY PROTOCOL OUT parameter list 737
7.7.4.5.1 Set Policy Access Tag CbCS page 737
7.7.4.5.2 Set Minimum CbCS Method CbCS page 737
7.7.4.5.3 Invalidate Key CbCS page 738
7.7.4.5.4 Set Key CbCS page 740
7.7.4.5.5 Set Master Key – Seed Exchange CbCS page 741
7.7.4.5.6 Set Master Key – Change Master Key CbCS page 742
7.8 Vital product data parameters 744
7.8.1 Vital product data parameters overview and page codes 744
7.8.2 VPD page format for all device types 745
7.8.3 ASCII Information VPD page 746
7.8.4 CFA Profile Information VPD page 747
7.8.5 Device Constituents VPD page 748
7.8.6 Device Identification VPD page 751
7.8.6.1 Device Identification VPD page overview 751
7.8.6.2 Device designation descriptor requirements 753
7.8.6.2.1 Designation descriptors for logical units other than well known logical units 753
7.8.6.2.2 Designation descriptors for well known logical units 754
7.8.6.2.3 Designation descriptors for SCSI target ports 754
7.8.6.2.3.1 Relative target port identifiers 754
7.8.6.2.3.2 Target port names or identifiers 754

7.8.6.2.4 Designation descriptors for SCSI target devices 754
7.8.6.3 Vendor specific designator format 755
7.8.6.4 T10 vendor ID based designator format 755
7.8.6.5 EUI-64 based designator format 756
7.8.6.5.1 EUI-64 based designator format overview 756
7.8.6.5.2 EUI-64 designator format 756
7.8.6.5.3 EUI-64 based 12-byte designator format 757
7.8.6.5.4 EUI-64 based 16-byte designator format 757
7.8.6.6 NAA designator format 758
7.8.6.6.1 NAA identifier basic format 758
7.8.6.6.2 NAA IEEE Extended designator format 758
7.8.6.6.3 NAA Locally Assigned designator format 759
7.8.6.6.4 NAA IEEE Registered designator format 759
7.8.6.6.5 NAA IEEE Registered Extended designator format 760
7.8.6.7 Relative target port designator format 760
7.8.6.8 Target port group designator format 761
7.8.6.9 Logical unit group designator format 761
7.8.6.10 MD5 logical unit designator format 761
7.8.6.11 SCSI name string designator format 763
7.8.6.12 Protocol specific port identifier designator format 763
7.8.6.12.1 Protocol specific port identifier designator format overview 763
7.8.6.12.2 USB target port identifier designator format 764
7.8.6.12.3 PCI Express routing ID designator format 764
7.8.7 Extended INQUIRY Data VPD page 765
7.8.8 Management Network Addresses VPD page 769
7.8.9 Mode Page Policy VPD page 770
7.8.10 Power Condition VPD page 772
7.8.11 Power Consumption VPD page 773
7.8.12 Protocol Specific Logical Unit Information VPD page 775
7.8.13 Protocol Specific Port Information VPD page 777
7.8.14 SCSI Ports VPD page 778
7.8.15 Software Interface Identification VPD page 781
7.8.16 Supported VPD Pages VPD page 782
7.8.17 Third-party Copy VPD page 782
7.8.17.1 Third-party Copy VPD page overview 782
7.8.17.2 Third-party copy descriptor format 783
7.8.17.3 Third-party copy descriptor type codes 784
7.8.17.4 Supported Commands third-party copy descriptor 785
7.8.17.4.1 Supported Commands third-party copy descriptor overview 785
7.8.17.4.2 Command support descriptor format 786
7.8.17.5 Parameter Data third-party copy descriptor 787
7.8.17.6 Supported Descriptors third-party copy descriptor 788
7.8.17.7 Supported CSCD Descriptor IDs third-party copy descriptor 789
7.8.17.8 ROD Token Features third-party copy descriptor 790
7.8.17.8.1 ROD Token Features third-party copy descriptor overview 790
7.8.17.8.2 Block ROD device type specific features descriptor 792
7.8.17.8.3 Stream ROD token device type features descriptor 794
7.8.17.8.4 Copy manager ROD token device type features descriptor 795
7.8.17.9 Supported ROD Types third-party copy descriptor 796
7.8.17.10 General Copy Operations third-party copy descriptor 798
7.8.17.11 Stream Copy Operations third-party copy descriptor 800
7.8.17.12 Held Data third-party copy descriptor 800
7.8.18 Unit Serial Number VPD page 801

8 Well known logical units 802
8.1 Model for well known logical units 802
8.2 REPORT LUNS well known logical unit 802

8.3 ACCESS CONTROLS well known logical unit 803
8.3.1 Access controls model 803
8.3.1.1 Access controls commands 803
8.3.1.2 Access controls overview 803
8.3.1.3 Access Control List 804
8.3.1.3.1 ACL overview 804
8.3.1.3.2 Access identifiers 805
8.3.1.3.3 Logical Unit Access Control Descriptor 805
8.3.1.4 Managing the ACL 806
8.3.1.4.1 ACL management overview 806
8.3.1.4.2 Authorizing ACL management 806
8.3.1.4.3 Identifying logical units during ACL management 807
8.3.1.4.4 Tracking changes in logical unit identification 807
8.3.1.5 Enrolling AccessIDs 807
8.3.1.5.1 Enrollment states 807
8.3.1.5.1.1 Summary of enrollment states 807
8.3.1.5.1.2 Not-enrolled state 808
8.3.1.5.1.3 Enrolled state 809
8.3.1.5.1.4 Pending-enrolled state 809
8.3.1.5.2 ACL LUN conflict resolution 809
8.3.1.6 Granting and revoking access rights 810
8.3.1.6.1 Non-proxy access rights 810
8.3.1.6.2 Proxy access 810
8.3.1.6.2.1 Proxy tokens 810
8.3.1.6.2.2 Proxy LUNs 811
8.3.1.7 Verifying access rights 811
8.3.1.8 The management identifier key 812
8.3.1.8.1 Management identifier key usage 812
8.3.1.8.2 Overriding the management identifier key 813
8.3.1.8.2.1 The OVERRIDE MGMT ID KEY service action 813
8.3.1.8.2.2 The override lockout timer 813
8.3.1.9 Reporting access control information 814
8.3.1.10 Access controls log 814
8.3.1.11 Interactions of access controls and other features 815
8.3.1.11.1 Task set management and access controls 815
8.3.1.11.2 Existing reservations and ACL changes 816
8.3.1.12 Access controls information persistence and memory usage requirements 816
8.3.1.13 Access identifier formats 818
8.3.1.13.1 Access identifier type 818
8.3.1.13.2 AccessID access identifiers 818
8.3.2 ACCESS CONTROL IN command 818
8.3.2.1 ACCESS CONTROL IN introduction 818
8.3.2.2 REPORT ACL service action 819
8.3.2.2.1 REPORT ACL introduction 819
8.3.2.2.2 REPORT ACL parameter data format 820
8.3.2.2.2.1 REPORT ACL parameter data introduction 820
8.3.2.2.2.2 Granted ACL data page format 821
8.3.2.2.2.3 Granted All ACL data page format 822
8.3.2.2.2.4 Proxy Tokens ACL data page format 824
8.3.2.3 REPORT LU DESCRIPTORS service action 825
8.3.2.3.1 REPORT LU DESCRIPTORS introduction 825
8.3.2.3.2 REPORT LU DESCRIPTORS parameter data format 827
8.3.2.4 REPORT ACCESS CONTROLS LOG service action 831
8.3.2.4.1 REPORT ACCESS CONTROLS LOG introduction 831
8.3.2.4.2 REPORT ACCESS CONTROLS LOG parameter data format 832
8.3.2.4.2.1 REPORT ACCESS CONTROLS LOG parameter data introduction 832
8.3.2.4.2.2 Key Overrides access controls log portion page format 833

8.3.2.4.2.3 Invalid Keys access controls log portion page format 834
8.3.2.4.2.4 ACL LUN Conflicts access controls log portion page format 835
8.3.2.5 REPORT OVERRIDE LOCKOUT TIMER service action 836
8.3.2.6 REQUEST PROXY TOKEN service action 837
8.3.3 ACCESS CONTROL OUT command 839
8.3.3.1 ACCESS CONTROL OUT introduction 839
8.3.3.2 MANAGE ACL service action 840
8.3.3.2.1 MANAGE ACL introduction 840
8.3.3.2.2 The Grant/Revoke ACE page 843
8.3.3.2.3 The Grant All ACE page 846
8.3.3.2.4 The Revoke Proxy Token ACE page 847
8.3.3.2.5 The Revoke All Proxy Tokens ACE page 848
8.3.3.3 DISABLE ACCESS CONTROLS service action 848
8.3.3.4 ACCESS ID ENROLL service action 849
8.3.3.5 CANCEL ENROLLMENT service action 850
8.3.3.6 CLEAR ACCESS CONTROLS LOG service action 850
8.3.3.7 MANAGE OVERRIDE LOCKOUT TIMER service action 851
8.3.3.8 OVERRIDE MGMT ID KEY service action 852
8.3.3.9 REVOKE PROXY TOKEN service action 853
8.3.3.10 REVOKE ALL PROXY TOKENS service action 854
8.3.3.11 ASSIGN PROXY LUN service action 855
8.3.3.12 RELEASE PROXY LUN service action 856
8.4 TARGET LOG PAGES well known logical unit 857
8.5 SECURITY PROTOCOL well known logical unit 857
8.6 MANAGEMENT PROTOCOL well known logical unit 858

9 Security manager command set 859

Annex A (informative) Terminology mapping 860

Annex B (informative) REPORT LUNS command examples 861

Annex C (informative) Replacing RESERVE/RELEASE functionality with PERSISTENT RESERVE IN/OUT equivalents 865
C.1 Introduction 865
C.2 Replacing the reserve/release method with the PERSISTENT RESERVE OUT COMMAND 865
C.3 Third party reservations 866

Annex D (informative) Third-party copy implementation and usage 867
D.1 Embedded and dedicated copy manager implementations 867
D.1.1 Overview 867
D.1.2 Embedded copy manager implementations 867
D.1.3 Dedicated copy manager implementations 867
D.2 Tracking copy operation progress 868
D.2.1 Overview 868
D.2.2 Detecting lack of progress in active copy operations 868

Annex E (informative) Variations between this standard and equivalent security protocols 869
E.1 IKEv2 protocol details and variations for IKEv2-SCSI 869
E.2 ESP protocol details and variations for ESP-SCSI 872

Annex F (informative) Numeric order codes 873
F.1 Numeric order codes introduction 873
F.2 Additional sense codes 873
F.3 Operation codes 891
F.3.1 Operation codes 891
F.3.2 Additional operation codes for devices with the ENCSERV bit set to one 896

F.3.3 MAINTENANCE IN service actions and MAINTENANCE OUT service actions 897
F.3.4 SERVICE ACTION IN service actions and SERVICE ACTION OUT service actions 898
F.3.5 SERVICE ACTION BIDIRECTIONAL service actions 898
F.3.6 Variable length CDB service action codes 899
F.4 Diagnostic page codes 900
F.5 Log page codes 901
F.6 Mode page codes 904
F.7 VPD page codes 907
F.8 ROD type codes 909
F.9 Version descriptor values 910
F.10 T10 IEEE binary identifiers 930

Annex G (informative) T10 vendor identification 931

Bibliography 948

Tables

Page
Table 1 — Numbering conventions examples ... 70
Table 2 — Comparison of decimal prefixes and binary prefixes ... 71
Table 3 — Class diagram constraints and notes notation ... 76
Table 4 — Class diagram multiplicity notation ... 76
Table 5 — Class diagram notation for classes ... 77
Table 6 — Class diagram notation for associations ... 78
Table 7 — Class diagram notation for aggregations ... 79
Table 8 — Class diagram notation for generalizations ... 80
Table 9 — Class diagram notation for dependencies ... 81
Table 10 — Notation for objects ... 81
Table 11 — Generic CDB format for 6-byte commands ... 83
Table 12 — Typical CDB format for 6-byte commands ... 83
Table 13 — Generic CDB format for 10-byte commands ... 84
Table 14 — Typical CDB format for 10-byte commands ... 85
Table 15 — Generic CDB format for 12-byte commands ... 86
Table 16 — Typical CDB format for 12-byte commands ... 86
Table 17 — Generic CDB format for MAINTENANCE IN commands ... 87
Table 18 — Generic CDB format for MAINTENANCE OUT commands ... 88
Table 19 — Generic CDB format for SERVICE ACTION IN(12) commands ... 88
Table 20 — Generic CDB format for SERVICE ACTION OUT(12) commands ... 89
Table 21 — Generic CDB format for 16-byte commands ... 89
Table 22 — Typical CDB format for 16-byte commands, if eight-byte LBAs not supported ... 90
Table 23 — Typical CDB format for 16-byte commands with eight-byte LBAs supported ... 91
Table 24 — Generic CDB format for SERVICE ACTION IN(16) commands ... 92
Table 25 — Generic CDB format for SERVICE ACTION OUT(16) commands ... 92
Table 26 — Generic CDB format for SERVICE ACTION BIDIRECTIONAL commands ... 93
Table 27 — Generic variable length CDB ... 93
Table 28 — Typical variable length CDB format for 32-byte commands ... 95
Table 29 — XCDB format ... 96
Table 30 — XCDB descriptor format ... 97
Table 31 — EXTENSION TYPE field ... 97
Table 32 — OPERATION CODE field ... 98
Table 33 — Code set enumeration ... 100
Table 34 — Relative port identifier values ... 100
Table 35 — Sense data response codes ... 101
Table 36 — Descriptor format sense data ... 102
Table 37 — Sense data descriptor format ... 103
Table 38 — DESCRIPTOR TYPE field ... 103
Table 39 — Information sense data descriptor format ... 104
Table 40 — Command-specific information sense data descriptor format ... 105
Table 41 — Sense key specific sense data descriptor format ... 106
Table 42 — Sense key specific information definitions ... 107
Table 43 — Field pointer sense key specific information ... 107
Table 44 — Actual retry count sense key specific information ... 108
Table 45 — Progress indication sense key specific information ... 108
Table 46 — Segment pointer sense key specific information ... 109
Table 47 — Unit attention condition queue overflow sense key specific information ... 109
Table 48 — Field replaceable unit sense data descriptor format ... 110
Table 49 — Another progress indication sense data descriptor format ... 110
Table 50 — Forwarded sense data descriptor format ... 111
Table 51 — SENSE DATA SOURCE field ... 112
Table 52 — Vendor specific sense data descriptor format ... 112
Table 53 — Fixed format sense data ... 113
Table 54 — Sense key descriptions ... 117
Table 55 — ASC and ASCQ assignments ... 118

Table 56 — Device clock value format ... 138
Table 57 — Timestamp origin value ... 138
Table 58 — WRITE BUFFER download microcode modes ... 141
Table 59 — WRITE BUFFER download microcode field processing ... 142
Table 60 — MULTI I_T NEXUS MICROCODE DOWNLOAD field ... 143
Table 61 — Identifying information types ... 149
Table 62 — Types of MAM attributes ... 150
Table 63 — MAM attribute states ... 150
Table 64 — Power condition state machine states ... 157
Table 65 — Power condition state machine timers ... 158
Table 66 — SPC-4 commands that are allowed in the presence of various reservations ... 165
Table 67 — PERSISTENT RESERVE OUT service actions that are allowed in the presence of various reservations ... 168
Table 68 — Register behaviors for a REGISTER service action ... 173
Table 69 — Register behaviors for a REGISTER AND IGNORE EXISTING KEY service action ... 174
Table 70 — I_T Nexuses being registered ... 175
Table 71 — Register behaviors for a REGISTER AND MOVE service action ... 176
Table 72 — Processing for a released or preempted persistent reservation ... 179
Table 73 — Preempting actions ... 182
Table 74 — Minimum SA parameters ... 192
Table 75 — USAGE_TYPE SA parameter ... 194
Table 76 — Security protocols that create SAs ... 194
Table 77 — KDFs summary ... 195
Table 78 — HMAC-based KDFs ... 196
Table 79 — Hash functions used by HMAC based on KDF_ID ... 197
Table 80 — RFC 3566 parameter translations for the KDF based on AES-XCBC-PRF-128 ... 197
Table 81 — IKEv2-SCSI shared key names and SA shared key names ... 209
Table 82 — Shared key size determination ... 210
Table 83 — Device Server Capabilities step parameter data requirements ... 211
Table 84 — IKEv2-SCSI command terminations that do not abandon the CCS ... 222
Table 85 — Security Manager class relationships ... 225
Table 86 — CbCS methods ... 232
Table 87 — CbCS communications trust requirement ... 234
Table 88 — Summary of CbCS shared keys ... 236
Table 89 — CbCS shared key identifier values ... 236
Table 90 — Associations between commands and permissions ... 241
Table 91 — Associations between security protocol commands and permissions ... 243
Table 92 — Summary of changeable CbCS parameters ... 244
Table 93 — CbCS extension descriptor format ... 246
Table 94 — ESP-SCSI data format before encryption and after decryption ... 248
Table 95 — ESP-SCSI outbound data descriptors ... 249
Table 96 — ESP-SCSI CDBs or Data-Out Buffer parameter list descriptor without initialization vector ... 250
Table 97 — ESP-SCSI CDBs or Data-Out Buffer full parameter list descriptor ... 252
Table 98 — ESP-SCSI Data-Out Buffer parameter list descriptor without length and initialization vector ... 253
Table 99 — ESP-SCSI Data-Out Buffer parameter list descriptor without length ... 254
Table 100 — ESP-SCSI Data-In Buffer parameter data descriptors ... 255
Table 101 — ESP-SCSI Data-In Buffer parameter data descriptor without initialization vector ... 255
Table 102 — ESP-SCSI Data-In Buffer full parameter data descriptor ... 257
Table 103 — ESP-SCSI Data-In Buffer parameter data descriptor without length and initialization vector ... 258
Table 104 — ESP-SCSI Data-In Buffer parameter data descriptor without length ... 259
Table 105 — Security algorithm codes ... 260
Table 106 — Exception commands for background self-tests ... 264
Table 107 — Self-test mode summary ... 266
Table 108 — Third-party copy commands ... 278
Table 109 — Mandatory copy manager command support requirements ... 279
Table 110 — Responses to the conditions that result from SCSI events ... 284
Table 111 — ROD types ... 285

Table 112 — ROD token format 288
Table 113 — Generic ROD token format 290
Table 114 — ROD TYPE field in generic ROD 291
Table 115 — Generic ROD token errors sorted by reporting importance 294
Table 116 — Copy manager relationships for processing ROD tokens 295
Table 117 — Segment descriptor type specific copy manager processing requirements 299
Table 118 — PAD and CAT bit definitions 301
Table 119 — PAD bit processing if there is no copy source or copy destination 302
Table 120 — Commands for all device types 307
Table 121 — CHANGE ALIASES command 309
Table 122 — CHANGE ALIASES parameter list 310
Table 123 — Alias entry format 311
Table 124 — Alias entry PROTOCOL IDENTIFIER field 311
Table 125 — Protocol independent alias entry FORMAT CODE field 312
Table 126 — COPY OPERATION ABORT command 313
Table 127 — EXTENDED COPY(LID4) command 314
Table 128 — EXTENDED COPY(LID4) parameter list 315
Table 129 — PARAMETER LIST FORMAT field 316
Table 130 — LIST ID USAGE field for the EXTENDED COPY(LID4) command 318
Table 131 — LIST ID USAGE field for the EXTENDED COPY(LID1) command 319
Table 132 — EXTENDED COPY descriptor type codes 321
Table 133 — EXTENDED COPY CSCD descriptor type codes 322
Table 134 — CSCD descriptor format 323
Table 135 — LU ID TYPE field 323
Table 136 — Device type specific parameters in CSCD descriptors 324
Table 137 — CSCD descriptor extension format 324
Table 138 — Device type specific CSCD descriptor parameters for block device types 325
Table 139 — Device type specific CSCD descriptor parameters for sequential-access device types 326
Table 140 — Stream device transfer lengths 326
Table 141 — Device type specific CSCD descriptor parameters for processor device types 327
Table 142 — Identification Descriptor CSCD descriptor format 328
Table 143 — Alias CSCD descriptor format 329
Table 144 — IP Copy Service CSCD descriptor format 330
Table 145 — ROD CSCD descriptor format 332
Table 146 — Inputs that affect the processing of the ROD PRODUCER CSCD DESCRIPTOR ID field 334
Table 147 — DEL_TKN bit processing 336
Table 148 — EXTENDED COPY segment descriptor type codes 337
Table 149 — Segment descriptor header 338
Table 150 — CSCD descriptor ID values 339
Table 151 — Block device to stream device segment descriptor 340
Table 152 — Stream device to block device segment descriptor 342
Table 153 — Block device to block device segment descriptor 343
Table 154 — Stream device to stream device segment descriptor 345
Table 155 — Inline data to stream device segment descriptor 346
Table 156 — Embedded data to stream device segment descriptor 348
Table 157 — Stream device to discard segment descriptor 349
Table 158 — Verify CSCD segment descriptor 350
Table 159 — Block device with offset to stream device segment descriptor 352
Table 160 — Stream device with offset to block device segment descriptor 353
Table 161 — Block device with offset to block device with offset segment descriptor 355
Table 162 — Write filemarks segment descriptor 356
Table 163 — Space segment descriptor 357
Table 164 — Locate segment descriptor 358
Table 165 — Tape device image copy segment descriptor 359
Table 166 — Register persistent reservation key segment descriptor 360
Table 167 — Third party persistent reservations source I_T nexus segment descriptor 361
Table 168 — Block device image copy segment descriptor 362

Table 169 — Populate a ROD from one or more block ranges segment descriptor 364
Table 170 — RANGE DESCRIPTOR TYPE field 365
Table 171 — Populate a ROD four gibi-block range descriptor format 365
Table 172 — Populate a ROD from one block range segment descriptor 366
Table 173 — EXTENDED COPY(LID1) command 367
Table 174 — EXTENDED COPY(LID1) parameter list 368
Table 175 — INQUIRY command 370
Table 176 — Standard INQUIRY data format 371
Table 177 — PERIPHERAL QUALIFIER field 372
Table 178 — Peripheral device type 373
Table 179 — VERSION field 374
Table 180 — RESPONSE DATA FORMAT field 374
Table 181 — TPGS field 375
Table 182 — Version descriptor values 376
Table 183 — SPI-specific standard INQUIRY bits 387
Table 184 — Maximum logical device configuration table 388
Table 185 — CLOCKING field 388
Table 186 — LOG SELECT command 389
Table 187 — Page control (PC) field 390
Table 188 — PAGE CODE field and SUBPAGE CODE field 391
Table 189 — PCR bit, SP bit, and PC field meanings when parameter list length is zero 391
Table 190 — LOG SENSE command 394
Table 191 — MANAGEMENT PROTOCOL IN command 396
Table 192 — MANAGEMENT PROTOCOL field in MANAGEMENT PROTOCOL IN command 396
Table 193 — MANAGEMENT PROTOCOL SPECIFIC2 field for MANAGEMENT PROTOCOL IN protocol 00h 397
Table 194 — Supported management protocols MANAGEMENT PROTOCOL IN parameter data 398
Table 195 — MANAGEMENT PROTOCOL OUT command 399
Table 196 — MANAGEMENT PROTOCOL field in MANAGEMENT PROTOCOL OUT command 399
Table 197 — MODE SELECT(6) command 400
Table 198 — MODE SELECT(10) command 402
Table 199 — MODE SENSE(6) command 403
Table 200 — Page control (PC) field 403
Table 201 — Mode page code usage in MODE SENSE commands for all devices 404
Table 202 — MODE SENSE(10) command 406
Table 203 — PERSISTENT RESERVE IN command 407
Table 204 — PERSISTENT RESERVE IN service action codes 407
Table 205 — PERSISTENT RESERVE IN parameter data for READ KEYS 408
Table 206 — Format of PERSISTENT RESERVE IN parameter data for READ RESERVATION with no reservation held 409
Table 207 — Format of PERSISTENT RESERVE IN parameter data for READ RESERVATION with a reservation held 409
Table 208 — Persistent reservation SCOPE field 410
Table 209 — Persistent reservation TYPE field 411
Table 210 — PERSISTENT RESERVE IN parameter data for REPORT CAPABILITIES 412
Table 211 — ALLOW COMMANDS field 413
Table 212 — Persistent Reservation Type Mask format 415
Table 213 — PERSISTENT RESERVE IN parameter data for READ FULL STATUS 416
Table 214 — PERSISTENT RESERVE IN full status descriptor format 417
Table 215 — PERSISTENT RESERVE OUT command 418
Table 216 — PERSISTENT RESERVE OUT service action codes 420
Table 217 — PERSISTENT RESERVE OUT service actions and valid parameters (part 1 of 2) 421
Table 218 — PERSISTENT RESERVE OUT parameter list 423
Table 219 — PERSISTENT RESERVE OUT specify initiator ports additional parameter data 424
Table 220 — PERSISTENT RESERVE OUT with REGISTER AND MOVE service action parameter list 426
Table 221 — READ ATTRIBUTE command 428
Table 222 — READ ATTRIBUTE service action codes 428
Table 223 — Status to be returned if medium auxiliary memory is not accessible 429

Table 224 — READ ATTRIBUTE with ATTRIBUTE VALUES service action parameter data format 430
Table 225 — READ ATTRIBUTE with ATTRIBUTE LIST service action parameter data format 431
Table 226 — READ ATTRIBUTE with LOGICAL VOLUME LIST service action parameter data format 431
Table 227 — READ ATTRIBUTE with PARTITION LIST service action parameter data format 432
Table 228 — READ ATTRIBUTE with SUPPORTED ATTRIBUTES service action parameter data format 432
Table 229 — READ BUFFER command 433
Table 230 — READ BUFFER MODE field 434
Table 231 — READ BUFFER descriptor 435
Table 232 — OFFSET BOUNDARY field 435
Table 233 — Echo buffer descriptor 436
Table 234 — Error history BUFFER ID field 437
Table 235 — Summary of error history directory device server actions 438
Table 236 — Error history directory 439
Table 237 — EHS_RETRIEVED field 440
Table 238 — EHS_SOURCE field 440
Table 239 — Error history directory entry 440
Table 240 — READ MEDIA SERIAL NUMBER command 442
Table 241 — READ MEDIA SERIAL NUMBER parameter data format 442
Table 242 — RECEIVE COPY DATA(LID4) command 443
Table 243 — Parameter data for the RECEIVE COPY DATA(LID4) command 444
Table 244 — RECEIVE COPY DATA(LID1) command 445
Table 245 — Parameter data for the RECEIVE COPY DATA(LID1) command 446
Table 246 — RECEIVE COPY OPERATING PARAMETERS command 446
Table 247 — Parameter data for the RECEIVE COPY OPERATING PARAMETERS command 447
Table 248 — RECEIVE COPY FAILURE DETAILS(LID1) command 450
Table 249 — Parameter data for the RECEIVE COPY FAILURE DETAILS(LID1) command 451
Table 250 — RECEIVE COPY STATUS(LID4) command 452
Table 251 — Parameter data for the RECEIVE COPY STATUS(LID4) command 453
Table 252 — COPY OPERATION STATUS field 454
Table 253 — EXTENDED COPY COMPLETION STATUS field contents based on COPY OPERATION STATUS field .. 455
Table 254 — COPY STATUS TRANSFER COUNT UNITS field 455
Table 255 — RECEIVE COPY STATUS(LID1) command 456
Table 256 — Parameter data for the RECEIVE COPY STATUS(LID1) command 457
Table 257 — COPY COMMAND STATUS field 457
Table 258 — RECEIVE CREDENTIAL command 458
Table 259 — RECEIVE CREDENTIAL command unencrypted bytes format 459
Table 260 — CREDENTIAL REQUEST TYPE field 459
Table 261 — CbCS logical unit credential request descriptor format 460
Table 262 — CbCS logical unit and volume credential request descriptor format 460
Table 263 — CbCS credential format 461
Table 264 — Credential format values 461
Table 265 — CbCS capability descriptor format 462
Table 266 — DESIGNATION TYPE field 462
Table 267 — CBCS METHOD field 463
Table 268 — PERMISSIONS BIT MASK field format 463
Table 269 — Logical unit designation descriptor format 465
Table 270 — Volume designation descriptor format 465
Table 271 — RECEIVE DIAGNOSTIC RESULTS command 466
Table 272 — RECEIVE ROD TOKEN INFORMATION command 467
Table 273 — Parameter data for the RECEIVE ROD TOKEN INFORMATION command 468
Table 274 — ROD token descriptor format 469
Table 275 — REMOVE I_T NEXUS command 470
Table 276 — REMOVE I_T NEXUS parameter list format 471
Table 277 — I_T nexus descriptor 471
Table 278 — REPORT ALIASES command 472
Table 279 — REPORT ALIASES parameter data 473
Table 280 — REPORT ALL ROD TOKENS command 474

Table 281 — Parameter data for the REPORT ALL ROD TOKENS command ... 475
Table 282 — REPORT IDENTIFYING INFORMATION command ... 476
Table 283 — IDENTIFYING INFORMATION TYPE field ... 476
Table 284 — Peripheral device identifying information parameter data ... 477
Table 285 — Identifying information supported parameter data ... 477
Table 286 — Identifying information descriptor ... 478
Table 287 — REPORT LUNS command ... 478
Table 288 — SELECT REPORT field ... 479
Table 289 — REPORT LUNS parameter data format ... 480
Table 290 — REPORT PRIORITY command ... 481
Table 291 — PRIORITY REPORTED field ... 481
Table 292 — REPORT PRIORITY parameter data format ... 482
Table 293 — Priority descriptor format ... 482
Table 294 — REPORT SUPPORTED OPERATION CODES command ... 483
Table 295 — REPORT SUPPORTED OPERATION CODES REPORTING OPTIONS field ... 484
Table 296 — All_commands parameter data ... 485
Table 297 — Command descriptor format ... 485
Table 298 — One_command parameter data ... 486
Table 299 — SUPPORT values ... 487
Table 300 — Command timeouts descriptor format ... 488
Table 301 — REPORT SUPPORTED TASK MANAGEMENT FUNCTIONS command ... 489
Table 302 — REPORT SUPPORTED TASK MANAGEMENT FUNCTIONS basic parameter data ... 490
Table 303 — REPORT SUPPORTED TASK MANAGEMENT FUNCTIONS extended parameter data ... 490
Table 304 — REPORT TARGET PORT GROUPS command ... 493
Table 305 — PARAMETER DATA FORMAT field ... 493
Table 306 — Length only header parameter data format ... 494
Table 307 — Extended header parameter data format ... 494
Table 308 — Target port group descriptor format ... 495
Table 309 — ASYMMETRIC ACCESS STATE field ... 496
Table 310 — STATUS CODE field ... 497
Table 311 — Target port descriptor format ... 497
Table 312 — REPORT TIMESTAMP command ... 498
Table 313 — REPORT TIMESTAMP parameter data format ... 498
Table 314 — REQUEST SENSE command ... 499
Table 315 — DESC bit ... 499
Table 316 — SECURITY PROTOCOL IN command ... 501
Table 317 — SECURITY PROTOCOL field in SECURITY PROTOCOL IN command ... 502
Table 318 — SECURITY PROTOCOL OUT command ... 503
Table 319 — SECURITY PROTOCOL field in SECURITY PROTOCOL OUT command ... 504
Table 320 — SEND DIAGNOSTIC command ... 505
Table 321 — SELF-TEST CODE field ... 505
Table 322 — The meanings of the SELF-TEST CODE field, the PF bit, the SELFTEST bit, and the PARAMETER LIST LENGTH field ... 507
Table 323 — SET IDENTIFYING INFORMATION command ... 511
Table 324 — IDENTIFYING INFORMATION TYPE field ... 512
Table 325 — SET IDENTIFYING INFORMATION parameter list ... 512
Table 326 — SET PRIORITY command ... 513
Table 327 — I_T_L NEXUS TO SET field ... 514
Table 328 — SET PRIORITY parameter list format ... 515
Table 329 — SET TARGET PORT GROUPS command ... 516
Table 330 — SET TARGET PORT GROUPS parameter list format ... 517
Table 331 — Set target port group descriptor parameter list ... 517
Table 332 — ASYMMETRIC ACCESS STATE field ... 518
Table 333 — SET TIMESTAMP command ... 519
Table 334 — SET TIMESTAMP parameter list format ... 519
Table 335 — TEST UNIT READY command ... 520
Table 336 — Preferred TEST UNIT READY responses ... 521

Table 337 — WRITE ATTRIBUTE command 522
Table 338 — WRITE ATTRIBUTE parameter list format 523
Table 339 — WRITE BUFFER command 524
Table 340 — WRITE BUFFER MODE field 525
Table 341 — MODE SPECIFIC field 528
Table 342 — Application client error history parameter list format 530
Table 343 — ERROR TYPE field 531
Table 344 — ERROR LOCATION FORMAT field 531
Table 345 — Summary of diagnostic page codes 533
Table 346 — Diagnostic page format 534
Table 347 — Protocol Specific diagnostic page 535
Table 348 — Supported Diagnostic Pages diagnostic page 535
Table 349 — Summary of log page codes 537
Table 350 — Log page format 538
Table 351 — LOG SELECT PCR bit, SP bit, and DS bit meanings when parameter list length is not zero 539
Table 352 — Log parameter 540
Table 353 — Threshold met criteria (TMC) field 542
Table 354 — FORMAT AND LINKING field 542
Table 355 — Parameter control byte values for bounded data counter parameters 543
Table 356 — Parameter control byte values for unbounded data counter parameters 544
Table 357 — Parameter control byte values for ASCII format list log parameters 545
Table 358 — Parameter control byte values for binary format list log parameters 546
Table 359 — Keywords for resetting or changing log parameter cumulative values 547
Table 360 — Application Client log page parameter codes 547
Table 361 — Application Client log page 548
Table 362 — General Usage Application Client log parameter 548
Table 363 — Buffer Over-Run/Under-Run log page parameter codes 549
Table 364 — Buffer Over-Run/Under-Run log page 550
Table 365 — Buffer Over-run/Under-run log parameter 551
Table 366 — Cache Memory Statistics log page parameter codes 552
Table 367 — Cache Memory Statistics log page commands 552
Table 368 — Cache Memory Statistics log page 553
Table 369 — Read Cache Memory Hits log parameter 553
Table 370 — Reads To Cache Memory log parameter 554
Table 371 — Write Cache Memory Hits log parameter 555
Table 372 — Writes From Cache Memory log parameter 556
Table 373 — Time From Last Hard Reset log parameter 557
Table 374 — Time Interval log parameter 558
Table 375 — Time interval descriptor 558
Table 376 — General Statistics and Performance log page parameter codes 559
Table 377 — Statistics and Performance log pages commands 560
Table 378 — General Statistics and Performance log page 560
Table 379 — General Access Statistics and Performance log parameter 561
Table 380 — Idle Time log parameter 563
Table 381 — Force Unit Access Statistics and Performance log parameter 564
Table 382 — Group Statistics and Performance log page parameter codes 565
Table 383 — Group Statistics and Performance (n) log page 566
Table 384 — Group Statistics and Performance (n) subpage codes 567
Table 385 — Group n Statistics and Performance log parameter 568
Table 386 — Group n Force Unit Access Statistics and Performance log parameter 569
Table 387 — Informational Exceptions log page parameter codes 571
Table 388 — Informational Exceptions log page 571
Table 389 — Informational Exceptions General log parameter 572
Table 390 — Last n Deferred Errors or Asynchronous Events log page parameter codes 573
Table 391 — Last n Deferred Errors or Asynchronous Events log page 573
Table 392 — Deferred Error or Asynchronous Event log parameter 574
Table 393 — Last n Error Events log page parameter codes 575

Table 394 — Last n Error Events log page 575
Table 395 — Error Event log parameter 576
Table 396 — Non-Medium Error log page parameter codes 576
Table 397 — Non-Medium Error log page 577
Table 398 — Non-Medium Error Count log parameter 577
Table 399 — Power Conditions Transitions log page parameter codes 578
Table 400 — Power Condition Transitions log page 578
Table 401 — Accumulated Transitions log parameter 579
Table 402 — Accumulated Transitions parameter codes and saturating counters 579
Table 403 — Protocol Specific Port log page 580
Table 404 — Generic protocol specific port log parameter 581
Table 405 — Read Error Counters log page parameter codes 582
Table 406 — Read Error Counters log page 583
Table 407 — Read Error Counter log parameter 583
Table 408 — Read Reverse Error Counters log page parameter codes 584
Table 409 — Read Reverse Error Counters log page 585
Table 410 — Read Reverse Error Counter log parameter 585
Table 411 — Self-Test Results log page parameter codes 586
Table 412 — Self-Test Results log page 586
Table 413 — Unused Self-Test Results log parameter 587
Table 414 — Self-Test Results log parameter 588
Table 415 — SELF-TEST RESULTS field 589
Table 416 — Start-Stop Cycle Counter log page parameter codes 590
Table 417 — Start-Stop Cycle Counter log page 590
Table 418 — Date of Manufacture log parameter 591
Table 419 — Accounting Date log parameter 592
Table 420 — Specified Cycle Count Over Device Lifetime log parameter 593
Table 421 — Accumulated Start-Stop Cycles log parameter 593
Table 422 — Specified Load-Unload Count Over Device Lifetime log parameter 594
Table 423 — Accumulated Load-Unload Cycles log parameter 595
Table 424 — Supported Log Pages log page 596
Table 425 — Supported page descriptor 596
Table 426 — Supported Log Pages and Subpages log page 597
Table 427 — Supported page/subpage descriptor 597
Table 428 — Supported Subpages log page 598
Table 429 — Supported subpage descriptor 598
Table 430 — Temperature log page parameter codes 599
Table 431 — Temperature log page 599
Table 432 — Temperature log parameter 600
Table 433 — Reference Temperature log parameter 600
Table 434 — Verify Error Counters log page parameter codes 601
Table 435 — Verify Error Counters log page 602
Table 436 — Verify Error Counter log parameter 602
Table 437 — Write Error Counters log page parameter codes 603
Table 438 — Write Error Counters log page 604
Table 439 — Write Error Counter log parameter 604
Table 440 — MAM ATTRIBUTE format 606
Table 441 — MAM attribute FORMAT field 606
Table 442 — MAM attribute identifier range assignments 607
Table 443 — Device type attributes 607
Table 444 — DEVICE VENDOR/SERIAL NUMBER attribute format 609
Table 445 — MEDIUM USAGE HISTORY attribute format 610
Table 446 — PARTITION USAGE HISTORY attribute format 613
Table 447 — Medium type attributes 615
Table 448 — MEDIUM TYPE and MEDIUM TYPE INFORMATION attributes 616
Table 449 — Host type attributes 617
Table 450 — TEXT LOCALIZATION IDENTIFIER attribute values 618

Table 451 — Summary of mode page codes 620
Table 452 — Mode page policies 621
Table 453 — Mode parameter list 621
Table 454 — Mode parameter header(6) 621
Table 455 — Mode parameter header(10) 622
Table 456 — General mode parameter block descriptor 623
Table 457 — Page_0 mode page format 624
Table 458 — Sub_page mode page format 624
Table 459 — Control mode page 625
Table 460 — Task set type (TST) field 626
Table 461 — QUEUE ALGORITHM MODIFIER field 627
Table 462 — Queue error management (QERR) field 627
Table 463 — Unit attention interlocks control (UA_INTLCK_CTRL) field 628
Table 464 — AUTOLOAD MODE field 629
Table 465 — Control Extension mode page 630
Table 466 — Disconnect-Reconnect mode page 632
Table 467 — Data transfer disconnect control (DTDC) field 633
Table 468 — Extended mode page 634
Table 469 — Extended Device-Type Specific mode page 635
Table 470 — Power Condition mode page 636
Table 471 — PM_BG_PRECEDENCE field 637
Table 472 — CCF IDLE field 638
Table 473 — CCF STANDBY field 639
Table 474 — CCF STOPPED field 639
Table 475 — Power Consumption mode page 640
Table 476 — Protocol Specific Logical Unit mode page 641
Table 477 — Page_0 mode page format Protocol Specific Port mode page 642
Table 478 — Sub_page mode page format Protocol Specific Port mode page 642
Table 479 — PROTOCOL IDENTIFIER field values 643
Table 480 — Fibre Channel alias entry format codes 643
Table 481 — Fibre Channel world wide port name alias entry format 644
Table 482 — Fibre Channel world wide port name with N_Port checking alias entry format 645
Table 483 — RDMA alias entry format codes 646
Table 484 — RDMA target port identifier alias entry format 646
Table 485 — InfiniBand global identifier with target port identifier checking alias entry format 647
Table 486 — iSCSI alias entry format codes 648
Table 487 — iSCSI name alias entry format 649
Table 488 — iSCSI name with binary IPv4 address alias entry format 650
Table 489 — iSCSI name with IPname alias entry format 652
Table 490 — iSCSI name with binary IPv6 address alias entry format 654
Table 491 — Fibre Channel N_Port_Name CSCD descriptor format 655
Table 492 — Fibre Channel N_Port_ID CSCD descriptor format 656
Table 493 — Fibre Channel N_Port_ID With N_Port_Name Checking CSCD descriptor format 657
Table 494 — SCSI Parallel T_L CSCD descriptor format 658
Table 495 — IEEE 1394 EUI-64 CSCD descriptor format 659
Table 496 — RDMA CSCD descriptor format 660
Table 497 — iSCSI IPv4 CSCD descriptor format 661
Table 498 — iSCSI IPv6 CSCD descriptor format 662
Table 499 — SAS Serial SCSI Protocol CSCD descriptor format 663
Table 500 — TransportID format 664
Table 501 — TransportID formats for specific SCSI transport protocols 664
Table 502 — Fibre Channel TransportID format 665
Table 503 — Parallel SCSI bus TransportID format 665
Table 504 — IEEE 1394 TransportID format 666
Table 505 — RDMA TransportID format 666
Table 506 — iSCSI TPID FORMAT field codes 667
Table 507 — iSCSI initiator device TransportID format 667

Table 508 — iSCSI initiator port TransportID format 668
Table 509 — SAS Serial SCSI Protocol TransportID format 669
Table 510 — SOP TransportID format 669
Table 511 — SECURITY PROTOCOL SPECIFIC field for SECURITY PROTOCOL IN protocol 00h 670
Table 512 — Supported security protocols SECURITY PROTOCOL IN parameter data 671
Table 513 — Certificate data SECURITY PROTOCOL IN parameter data 672
Table 514 — Security compliance information SECURITY PROTOCOL IN parameter data 673
Table 515 — Compliance descriptor format 674
Table 516 — COMPLIANCE DESCRIPTOR TYPE field 674
Table 517 — FIPS 140 compliance descriptor 675
Table 518 — RELATED STANDARD field 675
Table 519 — SECURITY PROTOCOL SPECIFIC field for the SA creation capabilities 676
Table 520 — Supported device server capabilities formats parameter data 677
Table 521 — IKEv2-SCSI device server capabilities parameter data 677
Table 522 — SECURITY PROTOCOL SPECIFIC field as defined by the IKEv2-SCSI SECURITY PROTOCOL IN command 678
Table 523 — SECURITY PROTOCOL SPECIFIC field as defined by the IKEv2-SCSI SECURITY PROTOCOL OUT command 679
Table 524 — IKEv2-SCSI SECURITY PROTOCOL OUT command and SECURITY PROTOCOL IN command parameter data 680
Table 525 — IKEv2-SCSI header checking of SAIs 682
Table 526 — NEXT PAYLOAD field 683
Table 527 — MESSAGE ID field 684
Table 528 — Next payload values in SECURITY PROTOCOL OUT/IN parameter data 685
Table 529 — IKEv2-SCSI payload format 687
Table 530 — Key Exchange payload format 689
Table 531 — Identification payload format 690
Table 532 — ID TYPE field 690
Table 533 — Certificate payload format 691
Table 534 — CERTIFICATE ENCODING field 691
Table 535 — Certificate Request payload format 692
Table 536 — Authentication payload format 693
Table 537 — Nonce payload format 695
Table 538 — Notify payload format 696
Table 539 — Delete payload format 697
Table 540 — Encrypted payload format 699
Table 541 — Plaintext format for Encrypted payload CIPHERTEXT field 701
Table 542 — IKEv2-SCSI SA Creation Capabilities payload format 704
Table 543 — IKEv2-SCSI SA Cryptographic Algorithms payload format 705
Table 544 — IKEv2-SCSI SAUT Cryptographic Algorithms payload format 707
Table 545 — IKEv2-SCSI Timeout Values payload format 708
Table 546 — IKEv2-SCSI cryptographic algorithm descriptor format 709
Table 547 — ALGORITHM TYPE field 710
Table 548 — ENCR IKEv2-SCSI cryptographic algorithm descriptor format 711
Table 549 — ENCR ALGORITHM IDENTIFIER field 712
Table 550 — PRF IKEv2-SCSI cryptographic algorithm descriptor format 713
Table 551 — PRF ALGORITHM IDENTIFIER field 714
Table 552 — INTEG IKEv2-SCSI cryptographic algorithm descriptor format 715
Table 553 — INTEG ALGORITHM IDENTIFIER field 715
Table 554 — D-H IKEv2-SCSI cryptographic algorithm descriptor format 716
Table 555 — D-H ALGORITHM IDENTIFIER field 717
Table 556 — SA_AUTH_OUT and SA_AUTH_IN IKEv2-SCSI cryptographic algorithm descriptor format 718
Table 557 — SA_AUTH_OUT and SA_AUTH_IN ALGORITHM IDENTIFIER field 719
Table 558 — IKEv2-SCSI command ordering processing requirements on a single I_T_L nexus 721
Table 559 — IKEv2-SCSI parameter error categories 723
Table 560 — IKEv2 Notify payload error translations for IKEv2-SCSI 725
Table 561 — SECURITY PROTOCOL SPECIFIC field for the CbCS SECURITY PROTOCOL IN command 726

Table 562 — Supported CbCS SECURITY PROTOCOL IN Pages CbCS page format 727
Table 563 — Supported CbCS SECURITY PROTOCOL OUT Pages CbCS page format 728
Table 564 — Unchangeable CbCS Parameters CbCS page format 728
Table 565 — KEYS SUPPORT field 730
Table 566 — MIN CBCS METHOD SUP field 730
Table 567 — Security Token CbCS page format 731
Table 568 — Current CbCS Parameters CbCS page format 732
Table 569 — Set Master Key – Seed Exchange CbCS page format 734
Table 570 — SECURITY PROTOCOL SPECIFIC field for the CbCS SECURITY PROTOCOL OUT command 736
Table 571 — Set Policy Access Tag CbCS page format 737
Table 572 — Set Minimum CbCS Method CbCS page format 737
Table 573 — Minimum CbCS Method CbCS Parameter set 738
Table 574 — Invalidate Key CbCS page format 739
Table 575 — Set Key CbCS page format 740
Table 576 — Set Master Key – Seed Exchange CbCS page format 741
Table 577 — Set Master Key – Change Master Key CbCS page format 742
Table 578 — Vital product data page codes 744
Table 579 — VPD page format 745
Table 580 — ASCII Information VPD page 746
Table 581 — CFA Profile Information VPD page 747
Table 582 — CFA profile descriptor 747
Table 583 — Device Constituents VPD page 748
Table 584 — Constituent descriptor 749
Table 585 — CONSTITUENT TYPE field 749
Table 586 — CONSTITUENT DEVICE TYPE field 750
Table 587 — Constituent specific descriptor format 750
Table 588 — CONSTITUENT SPECIFIC TYPE field 750
Table 589 — Device Identification VPD page 751
Table 590 — Designation descriptor 752
Table 591 — ASSOCIATION field 752
Table 592 — DESIGNATOR TYPE field 753
Table 593 — Vendor specific DESIGNATOR field format 755
Table 594 — T10 vendor ID based DESIGNATOR field format 755
Table 595 — EUI-64 based designator lengths 756
Table 596 — EUI-64 DESIGNATOR field format 756
Table 597 — EUI-64 based 12-byte DESIGNATOR field format 757
Table 598 — EUI-64 based 16-byte DESIGNATOR field format 757
Table 599 — NAA DESIGNATOR field format 758
Table 600 — Network Address Authority (NAA) field 758
Table 601 — NAA IEEE Extended DESIGNATOR field format 758
Table 602 — NAA Locally Assigned DESIGNATOR field format 759
Table 603 — NAA IEEE Registered DESIGNATOR field format 759
Table 604 — NAA IEEE Registered Extended DESIGNATOR field format 760
Table 605 — Relative target port DESIGNATOR field format 760
Table 606 — Target port group DESIGNATOR field format 761
Table 607 — Logical unit group DESIGNATOR field format 761
Table 608 — MD5 logical unit DESIGNATOR field format 762
Table 609 — MD5 logical unit identifier example available data 762
Table 610 — Example MD5 input for computation of a logical unit identifier 762
Table 611 — SCSI name string DESIGNATOR field format 763
Table 612 — Protocol specific port designator formats 764
Table 613 — USB target port identifier DESIGNATOR field format 764
Table 614 — PCI Express routing ID DESIGNATOR field format 764
Table 615 — Extended INQUIRY Data VPD page 765
Table 616 — ACTIVATE MICROCODE field 766
Table 617 — SPT field for peripheral device type 00h 766
Table 618 — SPT field for peripheral device type 01h 766

Table 619 — Management Network Addresses VPD page 769
Table 620 — Network service descriptor format 769
Table 621 — SERVICE TYPE field 770
Table 622 — Mode Page Policy VPD page 770
Table 623 — Mode page policy descriptor 771
Table 624 — MODE PAGE POLICY field 771
Table 625 — Power Condition VPD page 772
Table 626 — Power Consumption VPD page 774
Table 627 — Power consumption descriptor format 774
Table 628 — POWER CONSUMPTION UNITS field 774
Table 629 — Protocol Specific Logical Unit Information VPD page 775
Table 630 — Logical unit information descriptor 776
Table 631 — Protocol Specific Port Information VPD page 777
Table 632 — Port information descriptor 777
Table 633 — SCSI Ports VPD page 778
Table 634 — SCSI port designation descriptor 779
Table 635 — Target port descriptor 780
Table 636 — Software Interface Identification VPD page 781
Table 637 — Software interface identifier format 781
Table 638 — Supported VPD Pages VPD page 782
Table 639 — Third-party Copy VPD page 782
Table 640 — Third-party copy descriptor format 783
Table 641 — Third-party copy descriptor type codes 784
Table 642 — Supported Commands third-party copy descriptor format 785
Table 643 — Command support descriptor format 786
Table 644 — Parameter Data third-party copy descriptor format 787
Table 645 — Supported Descriptors third-party copy descriptor format 788
Table 646 — Supported CSCD Descriptor IDs third-party copy descriptor format 789
Table 647 — ROD Token Features third-party copy descriptor format 790
Table 648 — REMOTE TOKENS field 791
Table 649 — Block ROD device type specific features descriptor format 792
Table 650 — Stream ROD device type specific features descriptor format 794
Table 651 — Copy manager ROD device type specific features descriptor format 795
Table 652 — Supported ROD Types third-party copy descriptor format 796
Table 653 — ROD type descriptor format 797
Table 654 — General Copy Operations third-party copy descriptor format 798
Table 655 — Stream Copy Operations third-party copy descriptor format 800
Table 656 — Held Data third-party copy descriptor format 800
Table 657 — Unit Serial Number VPD page 801
Table 658 — Well known logical unit numbers 802
Table 659 — Commands for the REPORT LUNS well known logical unit 802
Table 660 — Commands for the ACCESS CONTROLS well known logical unit 803
Table 661 — ACCESS CONTROL OUT management identifier key requirements 806
Table 662 — ACCESS CONTROL IN management identifier key requirements 806
Table 663 — Mandatory access controls resources 817
Table 664 — Optional access controls resources 817
Table 665 — ACCESS IDENTIFIER TYPE field 818
Table 666 — AccessID access identifier format 818
Table 667 — ACCESS CONTROL IN service actions 819
Table 668 — ACCESS CONTROL IN command with REPORT ACL service action 819
Table 669 — ACCESS CONTROL IN with REPORT ACL parameter data format 820
Table 670 — ACL data page codes 821
Table 671 — Granted ACL data page format 821
Table 672 — Granted ACL data page LUACD descriptor format 822
Table 673 — ACCESS MODE field 822
Table 674 — Granted All ACL data page format 823
Table 675 — Proxy Tokens ACL data page format 824

Table 676 — Proxy token descriptor format 825
Table 677 — ACCESS CONTROL IN command with REPORT LU DESCRIPTORS service action 825
Table 678 — ACCESS CONTROL IN with REPORT LU DESCRIPTORS parameter data format 827
Table 679 — SUPPORTED LUN MASK FORMAT field format 828
Table 680 — Logical Unit descriptor format 829
Table 681 — ACCESS CONTROL IN command REPORT ACCESS CONTROLS LOG service action 831
Table 682 — CDB LOG PORTION field 832
Table 683 — ACCESS CONTROL IN with REPORT ACCESS CONTROLS LOG parameter data format . 832
Table 684 — Parameter data LOG PORTION field 833
Table 685 — Key Overrides access controls log portion page format 833
Table 686 — Invalid Keys access controls log portion page format 834
Table 687 — ACL LUN Conflicts access controls log portion page format 835
Table 688 — ACCESS CONTROL IN command REPORT OVERRIDE LOCKOUT TIMER service action 836
Table 689 — ACCESS CONTROL IN with REPORT OVERRIDE LOCKOUT TIMER parameter data 837
Table 690 — ACCESS CONTROL IN command with REQUEST PROXY TOKEN service action 838
Table 691 — ACCESS CONTROL IN with REQUEST PROXY TOKEN parameter data 839
Table 692 — ACCESS CONTROL OUT service actions 839
Table 693 — ACCESS CONTROL OUT command format 840
Table 694 — ACCESS CONTROL OUT with MANAGE ACL parameter data format 841
Table 695 — ACE page codes 842
Table 696 — Grant/Revoke ACE page format 843
Table 697 — ACE page LUACD descriptor format 844
Table 698 — Access Coordinator Grant/Revoke ACE page actions 845
Table 699 — Grant All ACE page format 846
Table 700 — Revoke Proxy Token ACE page format 847
Table 701 — Revoke All Proxy Tokens ACE page format 848
Table 702 — ACCESS CONTROL OUT with DISABLE ACCESS CONTROLS parameter data format 848
Table 703 — ACCESS CONTROL OUT with ACCESS ID ENROLL parameter data format 849
Table 704 — ACCESS CONTROL OUT with CLEAR ACCESS CONTROLS LOG parameter data format 851
Table 705 — CLEAR ACCESS CONTROLS LOG LOG PORTION field 851
Table 706 — ACCESS CONTROL OUT command MANAGE OVERRIDE LOCKOUT TIMER service action parameter data format 852
Table 707 — ACCESS CONTROL OUT with OVERRIDE MGMT ID KEY parameter data format 853
Table 708 — ACCESS CONTROL OUT with REVOKE PROXY TOKEN parameter data format 854
Table 709 — ACCESS CONTROL OUT with REVOKE ALL PROXY TOKENS parameter data format 854
Table 710 — ACCESS CONTROL OUT with ASSIGN PROXY LUN parameter data format 855
Table 711 — ACCESS CONTROL OUT with RELEASE PROXY LUN parameter data format 856
Table 712 — Commands for the TARGET LOG PAGES well known logical unit 857
Table 713 — Commands for the SECURITY PROTOCOL well known logical unit 858
Table 714 — Commands for the MANAGEMENT PROTOCOL well known logical unit 858
Table 715 — Commands for a security manager 859
Table A.1 — This standard to SPC-2 terminology mapping 860
Table B.1 — Example logical unit inventory 861
Table B.2 — REPORT LUNS command returned LUN list 862
Table C.1 — PERSISTENT RESERVE OUT command features 865
Table E.1 — IKE payload names shorthand 871
Table F.1 — ASC and ASCQ assignments 873
Table F.2 — Operation codes 891
Table F.3 — Additional operation codes for devices with the ENCSERV bit set to one 896
Table F.4 — MAINTENANCE IN service actions and MAINTENANCE OUT service actions 897
Table F.5 — SERVICE ACTION IN(12) service actions and SERVICE ACTION OUT(12) service actions . 898
Table F.6 — SERVICE ACTION IN(16) service actions and SERVICE ACTION OUT(16) service actions . 898
Table F.7 — SERVICE ACTION BIDIRECTIONAL service actions 898
Table F.8 — Variable Length CDB Service Action Code Ranges 899
Table F.9 — Variable Length CDB Service Action Codes Used by All Device Types 899
Table F.10 — Diagnostic page codes 900
Table F.11 — Log page codes 901

Table F.12 — Transport protocol specific log page codes 903
Table F.13 — Mode page codes 904
Table F.14 — Transport protocol specific mode page codes 906
Table F.15 — VPD page codes 907
Table F.16 — Transport protocol specific VPD page codes 908
Table F.17 — ROD type codes 909
Table F.18 — Version descriptor assignments 910
Table F.19 — Standard code value guidelines 925
Table F.20 — IEEE binary identifiers assigned by T10 930
Table G.1 — T10 vendor identification list 931

Figures

Page
Figure 1 — SCSI document structure 38
Figure 2 — Example state diagram 74
Figure 3 — Example flowchart 75
Figure 4 — Example class association relationships 78
Figure 5 — Example class aggregation relationships 79
Figure 6 — Example class generalization relationships 80
Figure 7 — Example class dependency relationships 81
Figure 8 — Power condition state machine 158
Figure 9 — Device server interpretation of PREEMPT service action 183
Figure 10 — SA relationships 191
Figure 11 — IKEv2-SCSI Device Server Capabilities step 201
Figure 12 — IKEv2-SCSI Key Exchange step 202
Figure 13 — IKEv2-SCSI Authentication step 203
Figure 14 — IKEv2-SCSI Delete operation 204
Figure 15 — CbCS overview class diagram 228
Figure 16 — Primary target port group example 268
Figure 17 — Examples of copy manager configurations 277
Figure 18 — EXTENDED COPY parameter list structure diagram 297
Figure 19 — ACL Structure 805
Figure B.1 — Example logical unit representation 862

INFORMATION TECHNOLOGY – SMALL COMPUTER SYSTEM INTERFACE (SCSI) –

Part 454: SCSI Primary Commands - 4 (SPC-4)

FOREWORD

1) ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.
2) The formal decisions or agreements of IEC and ISO on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC National Committees and ISO member bodies.
3) IEC, ISO and ISO/IEC publications have the form of recommendations for international use and are accepted by IEC National Committees and ISO member bodies in that sense. While all reasonable efforts are made to ensure that the technical content of IEC, ISO and ISO/IEC publications is accurate, IEC or ISO cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.
4) In order to promote international uniformity, IEC National Committees and ISO member bodies undertake to apply IEC, ISO and ISO/IEC publications transparently to the maximum extent possible in their national and regional publications. Any divergence between any ISO, IEC or ISO/IEC publication and the corresponding national or regional publication should be clearly indicated in the latter.
5) ISO and IEC do not provide any attestation of conformity. Independent certification bodies provide conformity assessment services and, in some areas, access to IEC marks of conformity. ISO or IEC are not responsible for any services carried out by independent certification bodies.
6) All users should ensure that they have the latest edition of this publication.
7) No liability shall attach to IEC or ISO or its directors, employees, servants or agents including individual experts and members of their technical committees and IEC National Committees or ISO member bodies for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication of, use of, or reliance upon, this ISO/IEC publication or any other IEC, ISO or ISO/IEC publications.
8) Attention is drawn to the normative references cited in this publication. Use of the referenced publications is indispensable for the correct application of this publication.
9) Attention is drawn to the possibility that some of the elements of this ISO/IEC publication may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

International Standard ISO/IEC 14776-454 was prepared by subcommittee 25: Interconnection of information technology equipment, of ISO/IEC joint technical committee 1: Information technology.

The list of all currently available parts of the ISO/IEC 14776 series, under the general title *Information technology – Small computer system interface (SCSI)*, can be found on the IEC web site.

This International Standard has been approved by vote of the member bodies and the voting results may be obtained from the address given on the second title page.

This publication has been drafted in accordance with the ISO/IEC Directives, Part 2 except as described in 3.4 and 3.5.

A bilingual version of this publication may be issued at a later date.

IMPORTANT - The 'colour inside' logo on the cover page of this publication indicates that it contains colours which are considered to be useful for the correct understanding of its contents. Users should therefore print this document using a colour printer.

INTRODUCTION

The SCSI command set is designed to provide efficient peer-to-peer operation of SCSI devices (e.g., disks, tapes, media changers) by an operating system. The SCSI command set assumes an underlying command-response protocol.

The SCSI command set provides multiple operating systems concurrent control over one or more SCSI devices. However, proper coordination of activities between the multiple operating systems is critical to avoid data corruption. Commands that assist with coordination between multiple operating systems are described in this standard. However, details of the coordination are beyond the scope of the SCSI command set.

This standard defines the device model for all SCSI devices. This standard defines the SCSI commands that are basic to every device model and the SCSI commands that may apply to any device model.

With any technical document there may arise questions of interpretation as new products are implemented. INCITS has established procedures to issue technical opinions concerning the standards developed by INCITS. These procedures may result in SCSI Technical Information Bulletins being published by INCITS.

Figure 1 shows the relationship of this standard to the other standards and related projects in the SCSI family of standards as of the publication of this standard.

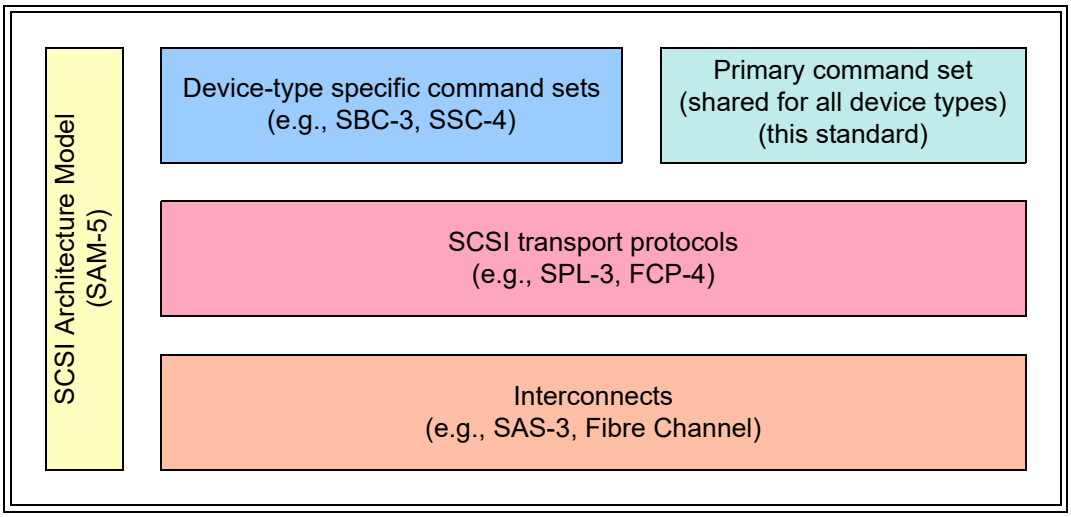


Figure 1 — SCSI document structure

The SCSI document structure in figure 1 is intended to show the general applicability of the documents to one another. Figure 1 is not intended to imply a relationship such as a hierarchy, protocol stack, or system architecture.

SCSI Architecture Model: Defines the SCSI systems model, the functional partitioning of the SCSI standard set and requirements applicable to all SCSI implementations and implementation standards.

Device-Type Specific Command Sets: Implementation standards that define specific device types including a device model for each device type. These standards specify the required commands and behaviors that are specific to a given device type and prescribe the requirements to be followed by a SCSI initiator device when sending commands to a SCSI target device having the specific device type. The commands and behaviors for a specific device type may include by reference commands and behaviors that are shared by all SCSI devices.

Shared Command Set: An implementation standard that defines a model for all SCSI device types. This standard specifies the required commands and behavior that is common to all SCSI devices, regardless of

device type, and prescribes the requirements to be followed by a SCSI initiator device when sending commands to any SCSI target device.

SCSI Transport Protocols: Implementation standards that define the requirements for exchanging information so that different SCSI devices are capable of communicating.

Interconnects: Implementation standards that define the communications mechanism employed by the SCSI transport protocols. These standards may describe the electrical and signaling requirements essential for SCSI devices to interoperate over a given interconnect. Interconnect standards may allow the interconnection of devices other than SCSI devices in ways that are outside the scope of this standard.

The term SCSI is used to refer to ISO/IEC 14776 (all parts).

These standards specify the interfaces, functions and operations necessary to ensure interoperability between conforming implementations. This standard is a functional description. Conforming implementations may employ any design technique that does not violate interoperability.

INFORMATION TECHNOLOGY – SMALL COMPUTER SYSTEM INTERFACE (SCSI) –

Part 454: SCSI Primary Commands - 4 (SPC-4)

1 Scope

ISO/IEC 14776 (all parts) provides for many different types of SCSI devices (e.g., disks, tapes, media changers). This standard defines a device model that is applicable to all SCSI devices. Other command standards expand on the general SCSI device model in ways appropriate to specific types of SCSI devices.

ISO/IEC 14776 (all parts) specifies the interfaces, functions, and operations necessary to ensure interoperability between conforming SCSI implementations. This standard is a functional description. Conforming implementations employ any design technique that does not violate interoperability.

This standard defines the SCSI commands that are mandatory and optional for all SCSI devices. Support for any feature defined in this standard is optional unless otherwise stated. This standard also defines the SCSI commands that may apply to any device model.

The following commands, parameter data, and features defined in previous versions of the SPC standard are made obsolete by this standard:

a) the TARGET RESET supported (TRS) bit and the WAKEUP supported (WAKES) bit in the REPORT SUPPORTED TASK MANAGEMENT FUNCTIONS parameter data;
b) code value 10b (i.e., Per initiator port) in the MODE PAGE POLICY field in the mode page policy descriptor in the Mode Page Policy VPD page;
c) the removable medium devices with an attached medium changer model, MCHNGR bit in the standard INQUIRY data, the MOVE MEDIUM ATTACHED command in disks and tapes, and the READ ELEMENT STATUS ATTACHED command in disks and tapes;
d) linked commands;
e) the PPC bit in the LOG SENSE command;
f) the NUL bit in EXTENDED COPY command target descriptors (i.e., CSCD descriptors in this standard);
g) EXTENDED COPY support for the processing of setmarks by sequential-access devices;
h) READ BUFFER commands with the MODE field set to 00h and 1Ah; and
i) WRITE BUFFER commands with the MODE field set to 00h, 1Ah, and 1Bh.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 14776-115, *Information technology - Small Computer System Interface (SCSI) - Part 115: Parallel Interface - 5 (SPI-5)*

ISO/IEC 14776-153, *Information technology - Small Computer System Interface (SCSI) - Part 153: Serial Attached SCSI - 2.1 (SAS-2.1)*

ISO/IEC 14776-322, *Information technology - Small Computer System Interface (SCSI) - Part 322: SCSI Block Commands - 2 (SBC-2)*

ISO/IEC 14776-323, *Information technology - Small Computer System Interface (SCSI) - Part 323: SCSI Block Commands - 3 (SBC-3)*

ISO/IEC 14776-331, *Information technology - Small Computer System Interface (SCSI) - Part 331: Stream Commands (SSC)*

ISO/IEC 14776-414, *Information technology - Small Computer System Interface (SCSI) - Part 414: SCSI Architecture Model - 4 (SAM-4)*

ISO/IEC 14776-452, *Information technology - Small Computer System Interface (SCSI) - Part 452: SCSI Primary Commands - 2 (SPC-2)*

ISO/IEC 14776-453, *Information technology - Small Computer System Interface (SCSI) - Part 453: SCSI Primary Commands - 3 (SPC-3)*

ISO 5807:1985, *Information processing - Documentation symbols and conventions for data, program and system flowcharts, program network charts and system resources charts*

INCITS 318-1998, *SCSI Controller Commands - 2 (SCC-2)*

INCITS 365, *Information technology SCSI RDMA Protocol (SRP)*

INCITS 375-2004, *Information technology - Serial Bus Protocol 3 (SBP-3)*

INCITS 457, *Information technology - Serial Attached SCSI - 2 (SAS-2)* [1)]

INCITS 468-2010, *Information technology - Multi-Media Command Set - 6 (MMC-6)*

INCITS 468-2010/AM 1-2012, *Information technology - Multi-Media Command Set 6 - Amendment 1 (MMC 6 AM1)*

INCITS 470-2011, *Information technology - Fibre Channel - Framing and Signaling Interface - 3 (FC-FS-3)*

INCITS 477-2011, *Information technology - Fibre Channel - Link Services - 2 (FC-LS-2)*

INCITS 481-2011, *Information technology - Fibre Channel Protocol for SCSI - 4 (FCP-4)* (planned as ISO/IEC 14776-224)

INCITS 484-2012, *Information technology SCSI Media Changer Commands 3 (SMC-3)*

INCITS 489, *Information technology - SCSI Over PCI Express® (SOP)*

INCITS 492, *Information technology - SAS Protocol Layer - 3 (SPL-3)* (planned as ISO/IEC 14776-263)

INCITS 496-2012, *Information technology - Fibre Channel - Security Protocols - 2 (FC-SP-2)*

INCITS 496-2012/AM 1-2015, *Information technology - Fibre Channel - Security Protocols - 2/Amendment 1 - (FC-SP-2/AM1)*

INCITS 515, *Information technology - Small Computer System Interface (SCSI) - SCSI Architecture Model - 5 (SAM-5)* (planned as ISO/IEC 14776-415)

INCITS 516, *Information technology - SCSI Stream Commands - 4 (SSC-4)*

---

1) SAS-2.1 supersedes SAS-2 but does not contain the information that this standard references normatively in 7.5.13.

INCITS 517, *Information technology - SCSI / ATA Translation - 3 (SAT-3)*

ANSI/IEEE 1394a-2000, *High Performance Serial Bus (supplement to ANSI/IEEE 1394-1995)*

ANSI/IEEE 1619.1-2007, *Standard for Authenticated Encryption with Length Expansion for Storage Devices*

ANSI/IEEE 1667:2009, *Standard Protocol for Authentication in Host Attachments of Transient Storage Devices*

INCITS 4-1986 (R2002), *Information Systems - Coded Character Sets - 7-Bit American National Standard Code for Information Interchange (7-Bit ASCII)*

ISO/IEC 13213, *Information technology - Microprocessor systems - Control and Status Registers (CSR) Architecture for microcomputer buses*

ISO/IEC 646:1991, *Information technology - ISO 7-bit coded character set for information interchange*

ISO/IEC 8859-1:1998, *Information technology - 8-bit single-byte coded graphic character sets - Part 1: Latin alphabet No. 1*

ISO/IEC 8859-2:1999, *Information technology - 8-bit single-byte coded graphic character sets - Part 2: Latin alphabet No. 2*

ISO/IEC 8859-3:1999, *Information technology - 8-bit single-byte coded graphic character sets - Part 3: Latin alphabet No. 3*

ISO/IEC 8859-4:1998, *Information technology - 8-bit single-byte coded graphic character sets - Part 4: Latin alphabet No. 4*

ISO/IEC 8859-5:1999, *Information technology - 8-bit single-byte coded graphic character sets - Part 5: Latin/Cyrillic alphabet*

ISO/IEC 8859-6:1999, *Information technology - 8-bit single-byte coded graphic character sets - Part 6: Latin/Arabic alphabet*

ISO/IEC 8859-7:2003, *Information technology - 8-bit single-byte coded graphic character sets - Part 7: Latin/Greek alphabet*

ISO/IEC 8859-8:1999, *Information technology - 8-bit single-byte coded graphic character sets - Part 8: Latin/Hebrew alphabet*

ISO/IEC 8859-9:1999, *Information technology - 8-bit single-byte coded graphic character sets - Part 9: Latin alphabet No. 5*

ISO/IEC 8859-10:1998, *Information technology - 8-bit single-byte coded graphic character sets - Part 10: Latin alphabet* No. 6

ISO/IEC 10646:2017, *Information technology - Universal Coded Character Set (UCS)*

Video Performance Guarantee Profile 1 Revision 1.0 (VPG1), May 24, 2011

NOTE 1 - For more information about documents published by the CFA, see http://www.compactflash.org/.

RFC 791, *Internet Protocol - DARPA Internet Program - Protocol Specification*

RFC 1321, *The MD5 Message-Digest Algorithm*

RFC 2104, *HMAC: Keyed-Hashing for Message Authentication*

RFC 2279, *UTF-8, a transformation format of ISO 10646*

RFC 2410, *The NULL Encryption Algorithm and Its Use With IPsec*

RFC 2437, *PKCS #1: RSA Cryptography Specifications Version 2.0*

RFC 2616, *Hypertext Transfer Protocol - HTTP/1.1*

RFC 3526, *More Modular Exponential (MODP) Diffie-Hellman groups for Internet Key Exchange (IKE)*

RFC 3566, *The AES-XCBC-MAC-96 Algorithm and Its Use With IPsec*

RFC 3602, *The AES-CBC Cipher Algorithm and Its Use with IPsec*

RFC 3766, *Determining Strengths For Public Keys Used For Exchanging Symmetric Keys*

RFC 4086, *Randomness Requirements for Security*

RFC 4106, *The Use of Galois/Counter Mode (GCM) in IPsec Encapsulating Security Payload*

RFC 4291, *IP Version 6 Addressing Architecture*

RFC 4303, *IP Encapsulating Security Payload (ESP)*

RFC 4306, *Internet Key Exchange (IKEv2) Protocol*

RFC 4309, *Using Advanced Encryption Standard (AES) CCM Mode with IPsec Encapsulating Security Payload*

RFC 4434, *The AES-XCBC-PRF-128 Algorithm for the Internet Key Exchange Protocol (IKE)*

RFC 4595, *Use of IKEv2 in the Fibre Channel Security Association Management Protocol*

RFC 4718, *IKEv2 Clarifications and Implementation Guidelines*

RFC 4753, *ECP Groups for IKE and IKEv2*

RFC Errata for RFC 4753, *http://rfc-editor.org/errata_search.php?rfc=4753*

RFC 4754, *IKE and IKEv2 Authentication Using the Elliptic Curve Digital Signature Algorithm (ECDSA)*

RFC 4868, *Using HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512 with IPsec*

RFC 5280, *Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*

RFC 5282, *Using Authenticated Encryption Algorithms with the Encrypted Payload of the Internet Key Exchange version 2 (IKEv2) Protocol*

RFC 5755, *An Internet Attribute Certificate Profile for Authorization*

RFC 6818, *Updates to the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*

RFC 7143, *iSCSI Protocol (Consolidated)*

RFC 7144, *Internet Small Computer Systems Interface (iSCSI) SCSI Features Update*

NOTE 2 - For more information on the RFCs and standards published by the Internet Engineering Task Force (IETF), see http://www.ietf.org/.

NIST SP (Special Publication) 800-38D, *Recommendation for Block Cipher Modes of Operation: Galois/Counter (GCM) Mode for Confidentiality and Authentication and GMAC*

NIST SP 800-90 A, *Recommendation for Random Number Generation Using Deterministic Random Bit Generators*

FIPS 140-2, *Security Requirements for Cryptographic Modules*

FIPS 140-2, *Annex C: Approved Random Number Generators*

FIPS 140-3, *Security Requirements for Cryptographic Modules (in development)*

FIPS 180-4, *Secure Hash Standard*

FIPS 198-1, *The Keyed-Hash Message Authentication Code (HMAC)*

NOTE 3 - For more information on NIST and FIPS standards published by the National Institute of Standards and Technology, see http://csrc.nist.gov/publications/.

Universal Serial Bus 3.0 Specification Revision 1.0 (USB-3), November 12, 2008

NOTE 4 - For more information on publications by the USB Implementers Forum, see http://www.usb.org/.

3 Terms, definitions, symbols, abbreviations, and conventions

3.1 Terms and definitions

3.1.1
access control list
ACL
data used by a SCSI target device to configure access rights for initiator ports according to the access controls state of the SCSI target device

Note 1 to entry: See 8.3.1.3.

3.1.2
access control list entry
ACE
one entry in the access control list (see 3.1.1)

3.1.3
access controls
SCSI target device feature that restricts initiator port access to specific logical units

Note 1 to entry: Information is modified about logical units in the parameter data of the INQUIRY command (see 6.6) and REPORT LUNS command (see 6.33).

Note 2 to entry: See 8.3.1.

3.1.4
access controls coordinator
entity within a SCSI target device that coordinates the management and enforcement of access controls (see 8.3.1) for all logical units within the SCSI target device

Note 1 to entry: The access controls coordinator is always addressable through the ACCESS CONTROLS well known logical unit (see 8.3) and LUN 0.

3.1.5
active power condition
power condition in which a device server is capable of completing the processing of all its supported commands

Note 1 to entry: See 5.11.4.

3.1.6
additional authenticated data
AAD
bytes of data input to a combined mode encryption algorithm (e.g., AES-GCM) as part of integrity checking computations

EXAMPLE: The IKEv2-SCSI Encrypted payload (see 7.7.3.5.11) uses AAD.

3.1.7
additional sense code
combination of the ADDITIONAL SENSE CODE field and the ADDITIONAL SENSE CODE QUALIFIER field in sense data (see 4.5)

3.1.8
alias list
list of alias values (see 3.1.9) and their associated designations (see 3.1.41) maintained by the device server and managed by the CHANGE ALIASES command (see 6.2) and REPORT ALIASES command (see 6.30)

3.1.9
alias value
numeric value associated to a designation (see 3.1.41) in the alias list (see 3.1.8)

Note 1 to entry: Used in command or parameter data to reference a SCSI target device or SCSI target port.

Note 2 to entry: See 6.2.2.

3.1.10
application client
object that is the source of SCSI commands

Note 1 to entry: See SAM-5.

3.1.11
ASCII format list log parameter
log parameter that contains ASCII data in a list log parameter format

Note 1 to entry: See 7.3.2.2.2.4.

3.1.12
attribute
single unit of MAM (see 3.1.88) information

3.1.13
auto contingent allegiance
ACA
task set condition established following the return of CHECK CONDITION status if the NACA bit is set to one in the CONTROL byte

Note 1 to entry: See SAM-5.

3.1.14
background function
background scan operation (see SBC-3) or device specific background function (see 5.3)

3.1.15
binary format list log parameter
log parameter that contains binary data in a list log parameter format

Note 1 to entry: See 7.3.2.2.2.5.

3.1.16
blocked command
command that is in the blocked state (see SAM-5)

Note 1 to entry: Commands become blocked if an ACA condition occurs and the blocked state ends when the ACA condition is cleared.

3.1.17
bounded data counter log parameter
log parameter that contains one saturating counter (see 3.1.134) value for which the establishment of thresholds is allowed

Note 1 to entry: Effects on other log parameters in the same log page are determined by the format and linking field in the parameter control byte.

Note 2 to entry: See 7.3.2.2.2.2.

3.1.18
byte
B
sequence of eight contiguous bits considered as a unit

3.1.19
cache memory
temporary and often volatile data storage area outside the area accessible by application clients

Note 1 to entry: Cache memory may contain a subset of the data stored in the non-volatile data storage area.

3.1.20
CbCS capability
descriptor that specifies access to a logical unit or volume (see SSC-4) for specific commands

Note 1 to entry: A CbCS capability is a component of a CbCS credential (see 3.1.20) and a CbCS extension descriptor (see 5.13.6.7.16).

Note 2 to entry: See 5.13.6.7.13.

3.1.21
CbCS capability key
shared key (see 3.1.159) that is cryptographically associated with a specific CbCS capability (see 3.1.20)

3.1.22
CbCS credential
combination of a CbCS capability (see 3.1.20) and a CbCS capability key (see 3.1.21)

Note 1 to entry: A CbCS credential is returned by a RECEIVE CREDENTIAL command (see 6.26).

Note 2 to entry: See 5.13.6.7.12.

3.1.23
CbCS Management Application Client class
CbCS (see 5.13.6.7) class whose objects manage, or an object that manages CbCS shared keys (see 5.13.6.7.11)

Note 1 to entry: See 5.13.6.7.4.

3.1.24
CbCS Management Device Server class
CbCS (see 5.13.6.7) class whose objects prepare, or an object that prepares CbCS credentials (see 5.13.6.7.12)

Note 1 to entry: A CbCS credential is prepared in response to a RECEIVE CREDENTIAL command (see 6.26) request from a secure CDB originator (see 5.13.6.7.5).

Note 2 to entry: See 5.13.6.7.3.

3.1.25
CbCS master key
shared key (see 3.1.159) from which CbCS working keys (see 3.1.28) are derived that is composed of an authentication key component and a generation key component

Note 1 to entry: See 5.13.6.7.11.

3.1.26
CbCS security token
random nonce (see 3.1.119) that is chosen by the enforcement manager (see 5.13.6.7.7) for a given I_T nexus and returned to a secure CDB originator (see 5.13.6.7.5)

Note 1 to entry: See 5.13.6.7.10.

3.1.27
CbCS working keys
shared keys (see 3.1.159) that are used to cryptographically associate a CbCS capability (see 3.1.20) with a CbCS capability key (see 3.1.21)

Note 1 to entry: See 5.13.6.7.11.

3.1.28
code set enumeration
coded value used in one field to indicate the format of data in other fields or descriptors

Note 1 to entry: See 4.3.3.

3.1.29
command
request describing a unit of work to be performed by a device server

Note 1 to entry: See SAM-5.

3.1.30
command descriptor block
CDB
structure used to communicate commands from an application client to a device server (see 4.2)

Note 1 to entry: A CDB may have a fixed length of up to 16 bytes or a variable length of between 12 and 260 bytes.

3.1.31
command standard
SCSI standard that defines the model, commands, and parameter data for a device type

EXAMPLE: SBC-3, SSC-4, SMC-3, MMC-6, and SES-2 are examples of command standards.

3.1.32
company_id
synonym for OUI (see 3.1.99)

3.1.33
copy manager
object (see SAM-5) that processes third-party copy commands (see 5.16.3) and manages copy operations (see 5.16.4.3)

3.1.34
copy operation
foreground or background operation that results from the processing of a third-party copy command

Note 1 to entry: See 5.16.4.3.

3.1.35
copy source or copy destination
CSCD
source or destination of a copy operation (see 6.4.6) performed by an EXTENDED COPY(LID4) command (see 6.4) or an EXTENDED COPY(LID1) command (see 6.5)

3.1.36
cryptographic command sequence
CCS
defined sequence of SECURITY PROTOCOL IN commands (see 6.40) and SECURITY PROTOCOL OUT commands (see 6.41) that realize the cryptographic protocol of a specified security operation

EXAMPLE: The SECURITY PROTOCOL IN commands and SECURITY PROTOCOL OUT commands in the Key Exchange step and Authentication step, if any, of an IKEv2-SCSI (see 3.1.67) SA creation transaction (see 3.1.128).

3.1.37
data counter log parameter
bounded data counter log parameter (see 7.3.2.2.2.2) or unbounded data counter log parameter (see 7.3.2.2.2.3)

3.1.38
Data-In Buffer
buffer specified by the application client to receive data from the device server during the processing of a command

Note 1 to entry: See SAM-5.

3.1.39
Data-Out Buffer
buffer specified by the application client to supply data that is sent from the application client to the device server during the processing of a command

Note 1 to entry: See SAM-5.

3.1.40
deferred error
CHECK CONDITION status and sense data that is returned as the result of an error or exception condition that occurred during processing of a previous command for which GOOD status or CONDITION MET status has already been returned

Note 1 to entry: See 4.5.7.

3.1.41
designation
distinguishing name, identifier, or title

3.1.42
designation
<access controls> name and optional identifier information that specifies a SCSI target device or SCSI target port for association with an alias value (see 3.1.9) in the alias list (see 3.1.8) as described in 6.2.2

3.1.43
device server
object within a logical unit that processes SCSI commands according to the rules of command management

Note 1 to entry: See SAM-5.

3.1.44
device service request
request, submitted by an application client, conveying a SCSI command to a device server

Note 1 to entry: See SAM-5.

3.1.45
device service response
response returned to an application client by a device server on completion of a SCSI command

Note 1 to entry: See SAM-5.

3.1.46
device type
type of peripheral device (i.e., device model) implemented by the device server and indicated to the application client by the contents of the PERIPHERAL DEVICE TYPE field in the standard INQUIRY data (see 6.6.2)

3.1.47
element
addressable physical component of a medium changer SCSI device that may serve as the location of a removable unit of data storage medium

Note 1 to entry: See SMC-3.

3.1.48
enabled command
command that is being processed by the device server and is progressing towards completion

Note 1 to entry: See SAM-4.

3.1.49
Encapsulating Security Payload for SCSI
ESP-SCSI
method for transferring encrypted and/or integrity checked parameter data in Data-In Buffers and/or Data-Out Buffers based on Encapsulating Security Payload (see RFC 4303)

Note 1 to entry: See 5.13.7.

3.1.50
Enforcement Manager class
command security (see 5.13.6) class whose objects are, or an object that is consulted by the Secure CDB Processor class (see 3.1.147) or Secure CDB Processor class objects to determine if the processing of a command is permitted or prohibited

Note 1 to entry: The Enforcement Manager class is equivalent to the Policy Enforcement Point in the policy model defined by RFC 2753 and similar policy-based authorization standards.

Note 2 to entry: See 5.13.6.4.

3.1.51
Error history I_T nexus
I_T nexus for which the device server has reserved access to the error history snapshot (see 3.1.52)

3.1.52
Error history snapshot
contents of the error history at a specific point in time

Note 1 to entry: See 5.5.2.

3.1.53
Extended CDB
XCDB
CDB that contains another CDB and additional information to support extra processing (e.g., security features)

Note 1 to entry: See 4.2.4.

3.1.54
EUI-48
48-bit identifier that is globally unique

Note 1 to entry: The IEEE maintains a tutorial describing EUI-48 at http://standards.ieee.org/develop/regauth/tut/eui.pdf.

3.1.55
EUI-64
64-bit identifier that is globally unique

Note 1 to entry: The IEEE maintains a tutorial describing EUI-64 at http://standards.ieee.org/develop/regauth/tut/eui.pdf.

3.1.56
faulted I_T nexus
I_T nexus on which a CHECK CONDITION status was returned that resulted in the establishment of an ACA

Note 1 to entry: The faulted I_T nexus condition is cleared when the ACA condition is cleared.

Note 2 to entry: See SAM-5.

3.1.57
field
group of one or more contiguous bits, a part of a larger structure such as a CDB (see 3.1.30) or sense data (see 3.1.155)

3.1.58
hard reset
condition resulting from the events defined by SAM-5 in which the SCSI device performs the hard reset operations described in SAM-5, this standard, and the applicable command standards

3.1.59
hashed message authentication code
HMAC
type of message authentication code that is calculated using the cryptographic hash function as defined in FIPS 198-1 in combination with a secret key

3.1.60
host
SCSI device with the characteristics of a primary computing device

Note 1 to entry: A host is typically a personal computer, workstation, server, minicomputer, mainframe computer, or auxiliary computing device.

Note 2 to entry: A host includes one or more SCSI initiator devices.

3.1.61
IEEE company_id
synonym for OUI (see 3.1.99)

3.1.62
I_T nexus
nexus between a SCSI initiator port and a SCSI target port

Note 1 to entry: See SAM-5.

3.1.63
I_T nexus loss
condition resulting from the events defined by SAM-5

Note 1 to entry: In response to this condition, the SCSI device performs the I_T nexus loss operations described in SAM-5, this standard, and other applicable standards.

3.1.64
I_T_L nexus
nexus between a SCSI initiator port, a SCSI target port, and a logical unit

Note 1 to entry: See SAM-5.

3.1.65
I_T_L_Q nexus transaction
information transferred between SCSI ports in a single data structure with defined boundaries (e.g., an information unit)

3.1.66
idle power condition
power condition in which a device server is capable of completing the processing of all its supported commands, except those that require the logical unit be in the active power condition (see 3.1.5) to be capable of completing the command with GOOD status

Note 1 to entry: See 5.11.5.

3.1.67
IKEv2-SCSI
Internet Key Exchange protocol version 2 for SCSI

Note 1 to entry: See 5.13.4.

3.1.68
IKEv2-SCSI keys
shared keys (see 3.1.159) used to provide security for IKEv2-SCSI operations (e.g., creation and deletion of the SA)

Note 1 to entry: IKEv2-SCSI keys used after SA creation is complete are maintained in the MGMT_DATA SA parameter (see 3.1.131).

Note 2 to entry: The shared key names used by IKEv2-SCSI are listed in 5.13.4.4.

3.1.69
IKEv2-SCSI CCS
CCS (see 3.1.36) that is the Key Exchange step and Authentication step, if any, of an IKEv2-SCSI (see 3.1.67) SA creation transaction (see 3.1.128)

3.1.70
information unit
IU
delimited and sequenced set of information in a format appropriate for transport by the service delivery subsystem (e.g., a CDB for a specific SCSI command)

3.1.71
initiator port
synonymous with SCSI initiator port (see 3.1.139)

3.1.72
initiator port identifier
value by which a SCSI initiator port (see 3.1.139) is referenced within a SCSI domain (see SAM-5)

3.1.73
initiator port name
name (see 3.1.91) of a SCSI initiator port

3.1.74
integrity check value
value used to cryptographically validate the integrity of a specified set of bytes that contain specified data based on a shared key (see 3.1.159)

3.1.75
internet assigned numbers authority
IANA
service that assigns and manages registries of code values (i.e., code points) for the IETF (see http://www.ietf.org) and other users of the public internet (see http://www.iana.org)

EXAMPLE: The IANA maintains code values used by Internet Key Exchange version 2 (IKEv2) at https://www.iana.org/assignments/ikev2-parameters/ikev2-parameters.xhtml.

3.1.76
Internet protocol domain name
name of a computer or hierarchy of computers within the domain name system defined by the IETF (see RFC 1035 and RFC 1591)

3.1.77
Internet protocol number
coded value assigned to identify protocols that layer on the Internet protocol (see RFC 791)

EXAMPLE: The Internet protocol number assigned to the transmission control protocol (TCP, see RFC 793) is six.

Note 1 to entry: The IANA maintains a list of Internet protocol number assignments at https://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml.

3.1.78
key derivation function
KDF
algorithm that is used to derive cryptographic keying material from a shared secret and other information

3.1.79
least significant bit
LSB
<in a binary code> bit or bit position with the smallest numerical weighting in a group of bits that, when taken as a whole, represent a numerical value

EXAMPLE – In the number 0001b, the bit that is set to one.

3.1.80
left-aligned
type of field containing ASCII data in which unused bytes are placed at the end of the field (i.e., highest offset) and are filled with ASCII space (20h) characters

Note 1 to entry: See 4.3.1.

3.1.81
logical unit
externally addressable entity within a SCSI target device that implements a SCSI device model and contains a device server

Note 1 to entry: See SAM-5.

3.1.82
logical unit access control descriptor
LUACD
structure within an ACE (see 3.1.2) that identifies a logical unit to which access is allowed and specifies the LUN by which the logical unit is to be accessed

Note 1 to entry: See 8.3.1.3.3.

3.1.83
logical unit inventory
list of the logical unit numbers reported by a REPORT LUNS command (see 6.33)

3.1.84
logical unit number
LUN
encoded 64-bit identifier for a logical unit

Note 1 to entry: See SAM-5.

3.1.85
logical unit reset
condition resulting from the events defined by SAM-5

Note 1 to entry: In response to this condition, the logical unit performs the logical unit reset operations described in SAM-5, this standard, and other applicable standards.

3.1.86
low power condition
any power condition other than the active power condition (see 5.11.4)

3.1.87
medium
physical entity that stores data in a nonvolatile manner (i.e., retained through a power cycle) in accordance with commands processed by the device server

3.1.88
medium auxiliary memory
MAM
auxiliary memory residing in a volume (e.g., a tape cartridge) that is accessible to the device server

Note 1 to entry: See 7.4.

3.1.89
media changer
device that mechanizes the movement of volumes to and from the SCSI device that records on or reads from the media

Note 1 to entry: See SMC-3.

3.1.90
most significant bit
MSB
<in a binary code> bit or bit position with the largest numerical weighting in a group of bits that, when taken as a whole, represent a numerical value

EXAMPLE: In the number 1000b, the bit that is set to one.

3.1.91
name
label of an object that is unique within a specified context and should never change

EXAMPLE: The term name and world wide identifier (WWID) may be interchangeable.

3.1.92
network address authority
NAA
field within a name (see 3.1.91) that specifies the format and length of that name

Note 1 to entry: See 7.8.6.6 and FC-FS-3.

3.1.93
nexus
relationship between two SCSI devices, and the SCSI initiator port and SCSI target port objects within those SCSI devices

Note 1 to entry: See SAM-5.

3.1.94
non-volatile cache memory
cache memory (see 3.1.19) that retains data through any power cycle

3.1.95
nonce
value that is used one and only one time to provide uniqueness to a value (e.g., a secure cryptographic key) in whose derivation it participates or to uniquely identify a single instance of something (e.g., a timestamp) exchanged between an application client and a device server

3.1.96
null-padded
type of field in which unused bytes are placed at the end of the field (i.e., highest offset) and are filled with ASCII null (00h) characters

Note 1 to entry: See 4.3.2.

3.1.97
null-terminated
type of field in which the last used byte (i.e., highest offset) is required to contain an ASCII null (00h) character

Note 1 to entry: See 4.3.2.

3.1.98
one
logical true condition of a variable

3.1.99
organizationally unique identifier
OUI
numeric identifier that is assigned by the IEEE such that no assigned identifiers are identical

Note 1 to entry: OUI is equivalent to company_id or IEEE company_id.

Note 2 to entry: The IEEE prefers OUI for EUI-48 identifiers (see 3.1.54) and company_id for EUI-64 identifiers (see 3.1.55). However, the numeric identifier is called an OUI when that identifier is assigned by the IEEE.

Note 3 to entry: The IEEE maintains a tutorial describing the OUI at http://standards.ieee.org/develop/regauth/tut/eui.pdf.

3.1.100
page
regular parameter structure or format used by several commands and identified with a value known as a page code

3.1.101
peripheral device
object that connects to a logical unit

3.1.102
peripheral device identifying information
value associated with a peripheral device accessed via the REPORT IDENTIFYING INFORMATION command (see 6.32) and SET IDENTIFYING INFORMATION command (see 6.43)

Note 1 to entry: See 5.6.

3.1.103
peripheral device text identifying information
UTF-8 (see 3.1.186) format string associated with a peripheral device accessed via the REPORT IDENTIFYING INFORMATION command (see 6.32) and SET IDENTIFYING INFORMATION command (see 6.43)

Note 1 to entry: See 5.6.

3.1.104
peripheral device type
synonym for device type (see 3.1.46)

3.1.105
persist through power loss
capability associated with some features that allows an application client to request that a device server maintain information regarding that feature across power failures

3.1.106
persistent reservation holder
I_T nexus(es) that are allowed to release or change a persistent reservation without preempting it

Note 1 to entry: See 5.12.10.

3.1.107
plaintext
unencrypted bytes

3.1.108
power cycle
power being removed from and later applied to a SCSI device

3.1.109
power on
condition resulting from the events defined by SAM-5 in which the SCSI device performs the power on operations described in SAM-5, this standard, and the applicable command standards

3.1.110
pre-shared key
shared key (see 3.1.159) that is established by a method outside the scope of this standard prior to the initiation of the function that requires its use

Note 1 to entry: The requirements on pre-shared keys that are particular to this standard are described in 5.13.4.3.2.

Note 2 to entry: Some of the RFCs referenced by this standard use the name pre-shared secret for the same information that this standard calls a pre-shared key.

3.1.111
primary target port asymmetric access state
characteristic that defines the behavior of a target port and the allowable command set for a logical unit when commands and task management functions are routed through the target port maintaining that state

Note 1 to entry: See 5.15.2.1.

3.1.112
primary target port group
set of target ports that are in the same primary target port asymmetric access state (see 3.1.111) at all times

Note 1 to entry: See 5.15.2.1.

3.1.113
primary target port group asymmetric access state
primary target port asymmetric access state (see 3.1.111) common to the set of target ports in a primary target port group (see 3.1.112)

Note 1 to entry: See 5.15.2.1.

3.1.114
protection information
one or more fields appended to each logical block that contain a CRC and device type specific information

Note 1 to entry: See SBC-3 and SSC-4.

3.1.115
protocol identifier
coded value used in various fields to identify the SCSI transport protocol to which other fields apply

Note 1 to entry: See 7.6.1.

3.1.116
protocol specific
requirement that is defined by a SCSI transport protocol standard (see 3.1.143)

Note 1 to entry: See SAM-5.

3.1.117
protocol standard
synonymous with SCSI transport protocol standard (see 3.1.143)

3.1.118
proxy token
identifier for a logical unit that may be generated to share access to that logical unit in the presence of access controls

Note 1 to entry: See 8.3.1.6.2.

3.1.119
random nonce
secure random number (see 4.4) that has a negligible chance of repeating and whose uses include providing significant uniqueness and randomness in cryptographic calculations

3.1.120
request for comment
RFC
name given to standards developed by the Internet Engineering Task Force

3.1.121
registered
condition that exists for an I_T nexus following the successful completion of a PERSISTENT RESERVE OUT command with a REGISTER service action, REGISTER AND IGNORE EXISTING KEY service action (see 5.12.7), or REGISTER AND MOVE service action (see 5.12.8) and lasting until the registration is removed (see 5.12.11)

3.1.122
registrant
I_T nexus that is registered (see 3.1.121)

3.1.123
right-aligned
type of field containing ASCII data in which unused bytes are placed at the start of the field (i.e., lowest offset) and are filled with ASCII space (20h) characters

Note 1 to entry: See 4.3.1.

3.1.124
relative port identifier
identifier for a SCSI port (see 3.1.140) that is unique within a SCSI device (see 3.1.135)

Note 1 to entry: Application clients may use the SCSI Ports VPD page (see 7.8.14) to determine relative port identifier values.

Note 2 to entry: See SAM-5.

3.1.125
relative initiator port identifier
relative port identifier (see 3.1.124) for a SCSI initiator port (see 3.1.139)

3.1.126
relative target port identifier
relative port identifier (see 3.1.124) for a SCSI target port (see 3.1.142)

3.1.127
ROD token
kind of token (see 3.1.178) that is a representation of data

Note 1 to entry: See 5.16.6.

3.1.128
SA creation transaction
any sequence of SECURITY PROTOCOL IN commands (see 6.40) and SECURITY PROTOCOL OUT commands (see 6.41) that is used to create an SA between an application client and device server

Note 1 to entry: A CCS (see 3.1.36) is a kind of SA creation transaction.

3.1.129
SA generation
computation and initialization of the SA parameter values (see 3.1.131) required to create an SA (see 3.1.152)

Note 1 to entry: SA generation is performed separately by the application client and device server after all SCSI commands required to create an SA have been performed without error

Note 2 to entry: See 5.13.4.9.

3.1.130
SA keys
shared keys (see 3.1.159) that are maintained in the USAGE_DATA SA parameter (see 3.1.131)

Note 1 to entry: SA keys are used to provide security for the operations that use the SA (e.g., encryption of command parameter data).

Note 2 to entry: The shared key names used by IKEv2-SCSI are listed in 5.13.4.4.

3.1.131
SA parameters
parameters stored by both an application client and a device server

Note 1 to entry: SA parameters are associated with one SA (see 3.1.152) and identified by a pair of SAIs (see 3.1.153).

Note 2 to entry: See 5.13.2.2..

3.1.132
SA participant
application client or device server that participates in the creation or use of an SA (see 3.1.152)

3.1.133
salt bytes
sequence of bytes whose contents are unpredictable in advance of their use

Note 1 to entry: Salt bytes are kept secret prior to their use, and disclosed during their use (e.g., bytes added to a cryptographic operation to increase the entropy in the result (see RFC 4106)).

3.1.134
saturating counter
counter that remains at its maximum value after reaching its maximum value

3.1.135
SCSI device
device that contains one or more SCSI ports that are each connected to a service delivery subsystem and supports a SCSI application protocol

Note 1 to entry: See SAM-5.

3.1.136
SCSI device name
name (see 3.1.91) of a SCSI device that is world wide unique within the protocol of a SCSI domain (see 3.1.137) in which the SCSI device has SCSI ports (see 3.1.140)

Note 1 to entry: The SCSI device name may be made available to other SCSI devices or SCSI ports in protocol specific ways.

Note 2 to entry: See SAM-5.

3.1.137
SCSI domain
interconnection of two or more SCSI devices and a service delivery subsystem

3.1.138
SCSI initiator device
SCSI device (see 3.1.135) containing application clients (see 3.1.10) and SCSI initiator ports (see 3.1.139)

Note 1 to entry: SCSI initiator devices originate device service and task management requests (see SAM-5) to be processed by a SCSI target device (see 3.1.141) and receive device service and task management responses from SCSI target devices.

Note 2 to entry: See SAM-5.

3.1.139
SCSI initiator port
SCSI initiator device (see SAM-5) object that acts as the connection between application clients and a service delivery subsystem through which requests and responses are routed

Note 1 to entry: See SAM-5.

3.1.140
SCSI port
SCSI initiator port (see 3.1.139) or SCSI target port (see 3.1.142)

3.1.141
SCSI target device
SCSI device (see 3.1.135) containing logical units (see 3.1.81) and SCSI target ports (see 3.1.142)

Note 1 to entry: SCSI target devices receive device service and task management requests (see SAM-5) for processing and send device service and task management responses to SCSI initiator devices.

Note 2 to entry: See SAM-5.

3.1.142
SCSI target port
target port
SCSI target device (see SAM-5) object that acts as the connection between device servers and task managers and a service delivery subsystem through which requests and responses are routed

Note 1 to entry: See SAM-5.

3.1.143
SCSI transport protocol standard
SCSI standard that defines a SCSI transport protocol (e.g., FCP-4, SPL-3, SRP, or SBP-3)

3.1.144
secondary target port asymmetric access state
characteristic indicating a condition that affects the way in which an individual target port participates in its assigned primary target port group (see 3.1.112)

Note 1 to entry: See 5.15.2.1.

3.1.145
secondary target port group
one or more target ports that are in the same secondary target port asymmetric access state (see 3.1.144)

Note 1 to entry: See 5.15.2.1.

3.1.146
Secure CDB Originator class
command security (see 5.13.6) class whose objects are, or an object that is an application client that originates secure CDBs

Note 1 to entry: See 5.13.6.2.

3.1.147
Secure CDB Processor class
command security (see 5.13.6) class whose objects are, or an object that is a device server that processes secure CDBs

Note 1 to entry: See 5.13.6.3.

3.1.148
Secure Content Storage Association
SCSA
organization that develops specifications for protecting digital media content

Note 1 to entry: See http://www.vidity.com.

3.1.149
Secure Digital Card Association
SD Card
organization that establishes and maintains standards for secure digital memory card applications

Note 1 to entry: See http://www.sdcard.org/.

3.1.150
secure hash algorithm
SHA
secure hash algorithm (e.g., SHA-1) defined in FIPS 180-4, Secure Hash Standard (SHS)

3.1.151
secure random number
random number that is generated in ways that protect it from security attacks

Note 1 to entry: See 4.4.

3.1.152
security association
SA
relationship and associated security processing between an application client and a device server that is used to apply security functions (e.g., integrity checking, encryption) to data that is transferred in either direction

Note 1 to entry: See 5.13.2.

3.1.153
security association index
SAI
number representing the parameters for an SA as stored internally by the application client or device server

Note 1 to entry: In other security models, this value is called the security parameters index (SPI).

Note 2 to entry: See 5.13.2.

3.1.154
Security Manager class
command security (see 5.13.6) class whose objects maintain, or an object that maintains information about which secure CDBs may be originated by which secure CDB originator (see 3.1.146) application clients to which secure CDB processor (see 3.1.147) device servers

Note 1 to entry: The Security Manager responds to requests to allow the origination of secure CDBs from secure CDB originator application clients, and updates secure CDB permissions and prohibitions in appropriate enforcement manager (see 3.1.50) members.

Note 2 to entry: The Security Manager is equivalent to the Policy Decision Point in the policy model defined by RFC 2753 and similar policy-based authorization standards.

Note 3 to entry: See 5.13.6.5.

3.1.155
sense data
data describing an error, exceptional condition, or completion information

Note 1 to entry: A device server delivers sense data to an application client in the same I_T_L_Q nexus transaction (see 3.1.65) as the status or as parameter data in response to a REQUEST SENSE command (see 6.39).

Note 2 to entry: See 4.5.

3.1.156
sense key
contents of the SENSE KEY field in sense data (see 4.5)

3.1.157
service action
extension of the operation code (see 4.2.5.1) used to define a command

Note 1 to entry: See 4.2.5.2.

3.1.158
service delivery subsystem
that part of a SCSI domain that transmits service requests to a logical unit or SCSI target device and returns logical unit or SCSI target device responses to a SCSI initiator device

Note 1 to entry: See SAM-5.

3.1.159
shared key
SK
cryptographically protected secret (e.g., with more randomness (see RFC 4089) than a password) that is known only to a defined and limited set of entities (e.g., one application client and one device server)

Note 1 to entry: The shared key names used by IKEv2-SCSI are listed in 5.13.4.4.

3.1.160
SK_ai
shared key (see 3.1.159) used for integrity checking data in a Data-Out Buffer (e.g., integrity checking the Encrypted payload in an IKEv2-SCSI Authentication step SECURITY PROTOCOL OUT command (see 5.13.4.7.2))

3.1.161
SK_ar
shared key (see 3.1.159) used for integrity checking data in a Data-In Buffer (e.g., integrity checking the Encrypted payload in an IKEv2-SCSI Authentication step SECURITY PROTOCOL IN command (see 5.13.4.7.3))

3.1.162
SK_d
shared key (see 3.1.159) KDF input material use to generate the shared keys stored in the KEYMAT SA Parameter (see 3.1.131)

3.1.163
SK_ei
shared key (see 3.1.159) used for encrypting data in a Data-Out Buffer (e.g., encrypting the Encrypted payload in an IKEv2-SCSI Authentication step SECURITY PROTOCOL OUT command (see 5.13.4.7.2))

3.1.164
SK_er
shared key (see 3.1.159) used for encrypting data in a Data-In Buffer (e.g., encrypting the Encrypted payload in an IKEv2-SCSI Authentication step SECURITY PROTOCOL IN command (see 5.13.4.7.3))

3.1.165
SK_pi
shared key (see 3.1.159) used to construct the IKEv2-SCSI Data-Out Buffer Authentication payload (see 5.13.4.7.2)

3.1.166
SK_pr
shared key (see 3.1.159) used to construct the IKEv2-SCSI Data-In Buffer Authentication payload (see 5.13.4.7.3)

3.1.167
standby power condition
power condition in which a device server is capable of completing the processing of all its supported commands, except those that require the logical unit be in the idle power condition (see 3.1.66) or active power condition (see 3.1.5) to be capable of completing the command with GOOD status

Note 1 to entry: See 5.11.6.

3.1.168
status
one byte of response information that contains a coded value defined in SAM-5, transferred from a device server to an application client upon completion of each command

Note 1 to entry: See SAM-5.

3.1.169
storage networking industry association
SNIA
organization that develops and promotes standards for storage management and other storage-related functions

Note 1 to entry: See http://www.snia.org/.

3.1.170
system
one or more SCSI domains operating as a single configuration

3.1.171
target port asymmetric access state
primary target port asymmetric access state (see 3.1.111) or secondary target port asymmetric access state (see 3.1.144)

3.1.172
target port group
primary target port group (see 3.1.112) or secondary target port group (see 3.1.145)

3.1.173
target port identifier
value by which a SCSI target port (see 3.1.142) is referenced within a SCSI domain (see SAM-5)

3.1.174
target port name
name (see 3.1.91) of a SCSI target port

3.1.175
task set
group of commands within a logical unit, whose interaction is dependent on the task management (i.e., queuing) and ACA rules

Note 1 to entry: The Control mode page (see 7.5.8) defines management capabilities for task sets.

Note 2 to entry: See SAM-5.

3.1.176
TCP port numbers
one of the data necessary to establish a TCP connection

Note 1 to entry: TCP port numbers may be assigned to protocols that layer on TCP by IANA. IANA maintains a list of TCP port number assignments at http://www.iana.org/assignments/port-numbers.

3.1.177
third-party command
command sent to one SCSI device requesting that an operation be performed involving two other SCSI devices

EXAMPLE: The EXTENDED COPY command may perform copy operations between two or more SCSI devices none of which are the SCSI device to which the EXTENDED COPY command was sent.

3.1.178
token
representation of a collection of data (e.g., management data, security data)

3.1.179
Trusted Computing Group
TCG
organization that develops and promotes open standards for hardware-enabled trusted computing and security technologies

Note 1 to entry: See https://www.trustedcomputinggroup.org.

3.1.180
unbounded data counter log parameter
log parameter that contains one or more saturating counter values (see 3.1.134) or wrapping counter values (see 3.1.190) that do not affect other log parameters in the same log page and for which the establishment of thresholds is not allowed

Note 1 to entry: See 7.3.2.2.2.3.

3.1.181
unit attention condition
asynchronous status information that a logical unit establishes to report to the initiator ports associated with one or more I_T nexuses

Note 1 to entry: See SAM-5.

3.1.182
universal time
UT
time at longitude zero, colloquially known as Greenwich Mean Time

Note 1 to entry: See http://aa.usno.navy.mil/faq/docs/UT.php.

3.1.183
URI Schemes
syntax specifications for UTF-8 (see 3.1.186) format strings that identify resources (see RFC 3986)

Note 1 to entry: IANA maintains a list of schemes for URI and URL names at: https://www.iana.org/assignments/uri-schemes/uri-schemes.xhtml.

3.1.184
user data
data contained in logical blocks (e.g., see SBC-3) that is not protection information (see 3.1.114)

Note 1 to entry: Each command standard (see 3.1.31) may provide its own definition of user data.

3.1.185
user data segment
contiguous sequence of logical blocks

Note 1 to entry: See SBC-3.

3.1.186
UTF-8
character set that is a transformation format of the character set defined by ISO 10646

Note 1 to entry: See RFC 2279.

3.1.187
volatile cache memory
cache memory (see 3.1.19) that does not retain data through power cycles

3.1.188
well known logical unit
logical unit that only does specific functions in a SCSI target device

Note 1 to entry: Well known logical units allow an application client to issue requests to receive and manage specific information usually relating to a SCSI target device.

Note 2 to entry: See clause 8 and SAM-5.

3.1.189
word
sequence of 16 contiguous bits considered as a unit

3.1.190
wrapping counter
counter that wraps back to zero after reaching its maximum value

3.1.191
zero
logical false condition of a variable

3.1.192
zero-padded
type of field in which unused bytes are placed at the end of the field (i.e., highest offset) and are filled with zeros

Note 1 to entry: See 4.3.2.

3.2 Abbreviations and symbols

3.2.1 Abbreviations

Abbreviations used in this standard:

Abbreviation	Meaning
AAD	Additional Authenticated Data (see 3.1.6)
ACE	Access Control list Entry (see 3.1.2)
ACL	Access Control List (see 3.1.1)
ACA	Auto Contingent Allegiance (see 3.1.13)
ADC-3	Automation/Drive Interface - Commands - 3 (see bibliography)
ADT-2	Automation/Drive Interface - Transport Protocol - 2 (see bibliography)
AES	Advanced Encryption Standards
AES-GCM	Advanced Encryption Standard - Galois Counter Mode (see RFC 4106)
ASC	Additional Sense Code (see 4.5)
ASCII	American Standard Code for Information Interchange
ASCQ	Additional Sense Code Qualifier (see 4.5)
ASN.1	Abstract Syntax Notation One
ATA	AT Attachment (see www.t13.org)
ATAPI	AT Attachment with Packet Interface (see www.t13.org)
CbCS	Capability-based Command Security (see 5.13.6.7)
CCS	Cryptographic Command Sequence (see 3.1.36)
CDB	Command Descriptor Block (see 3.1.30)
CFA	CompactFlash Association
CRC	Cyclic Redundancy Check
CSCD	Copy Source or Copy Destination (see 3.1.35)
CSEC	Communications Security Establishment Canada
D_ID	Destination Identifier (defined in FC-FS-3)
ECP	Elliptic Curve group modulo Prime (see RFC 4753)
ECDSA	Elliptic Curve Digital Signature Algorithm (see RFC 4754)
ESP-SCSI	Encapsulating Security Payload for SCSI (see 3.1.49)
EUI-48	Extended Unique Identifier, a 48-bit globally unique identifier (see 3.1.54)
EUI-64	Extended Unique Identifier, a 64-bit globally unique identifier (see 3.1.55)
FC-FS-3	Fibre Channel Framing and Signaling Interface - 3 (see clause 2)
FC-LS-2	Fibre Channel Link Services - 2 (see clause 2)
FC-SP-2	Fibre Channel Security Protocols - 2 (see clause 2)
FCP-4	Fibre Channel Protocol for SCSI - 4 (see clause 2)
FIPS	Federal Information Processing Standard
HMAC	Hashed Message Authentication Code (see 3.1.59)
HTTP	Hypertext Transfer Protocol (see RFC 2616)
IANA	Internet Assigned Numbers Authority (see 3.1.75)
ID	Identifier or Identification
IEEE	Institute of Electrical and Electronics Engineers
IKEv2	Internet Key Exchange version 2
IP	Internet Protocol
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
iSCSI	Internet SCSI
IU	Information Unit
KDF	Key Derivation Function (see 3.1.78)
LBA	Logical Block Address

Abbreviation	Meaning
LID1	List Identifier is 1 byte (i.e., a third-party copy command with a one-byte LIST IDENTIFIER field)
LID4	List Identifier is 4 bytes (i.e., a third-party copy command with a four-byte LIST IDENTIFIER field)
LSB	Least Significant Bit (see 3.1.79)
LUACD	Logical Unit Access Control Descriptor (see 3.1.82)
LUN	Logical Unit Number (see 3.1.84)
MAC	Message Authentication Code
MAM	Medium Auxiliary Memory (see 3.1.88)
MMC-6	SCSI Multi-Media Commands - 6 and MMC - 6 Amendment 1 (see clause 2)
MODP	Modular exponential (see RFC 3526)
MSB	Most Significant Bit (see 3.1.90)
NAA	Network Address Authority (see 3.1.92)
OCRW	SCSI Specification for Optical Card Reader/Writer
OSD	Object-based Storage Devices Commands
OUI	Organizationally Unique Identifier (see 3.1.99)
PKI	Public Key Infrastructure (see RFC 5280 and RFC 6818)
PRF	Pseudo-Random Function (see RFC 4306)
RAID	Redundant Array of Independent Disks
RBC	SCSI Reduced Block Commands (see bibliography)
RDMA	Remote Direct Memory Access (see SRP)
RFC	Request For Comments (see 3.1.120)
ROD	Representation Of Data (see 5.16.6)
SA	Security Association (see 3.1.152)
SAI	Security Association Index (see 3.1.153)
SAM-2	SCSI Architecture Model - 2 (see bibliography)
SAM-4	SCSI Architecture Model - 4 (see clause 2)
SAM-5	SCSI Architecture Model - 5 (see clause 2)
SAS-2	Serial Attached SCSI - 2 (see clause 2)
SAT-3	SCSI / ATA Translation -3 (see clause 2)
SAUT	SA Usage Type (see 7.7.3.5.14)
SBC-2	SCSI Block Commands - 2 (see clause 2)
SBC-3	SCSI Block Commands - 3 (see clause 2)
SBC-4	SCSI Block Commands - 4
SBP-3	Serial Bus Protocol - 3 (see clause 2)
SCC-2	SCSI Controller Commands - 2 (see clause 2)
SCSA	Secure Content Storage Association (see 3.1.148)
SCSI	Small Computer System Interface
SES-2	SCSI Enclosure Services - 2 (see bibliography)
SES-3	SCSI Enclosure Services - 3 (see bibliography)
SHA-1	Secure Hash Algorithm, 160 bits (see 3.1.150)
SHA-256	Secure Hash Algorithm, 256 bits (see 3.1.150)
SHA-384	Secure Hash Algorithm, 384 bits (see 3.1.150)
SHA-512	Secure Hash Algorithm, 512 bits (see 3.1.150)
SK	Shared Key (see 3.1.159)
SMC-3	SCSI Media Changer Commands -3
SNIA	Storage Networking Industry Association (see 3.1.169)
SPC-2	SCSI Primary Commands - 2 (see clause 2)
SPC-3	SCSI Primary Commands - 3 (see clause 2)
SPC-4	SCSI Primary Commands - 4 (this standard)
SPI-5	SCSI Parallel Interface - 5 (see clause 2)

Abbreviation	Meaning
SPL-2	SAS Protocol Layer - 2 (see bibliography)
SPL-3	SAS Protocol Layer - 3 (see clause 2)
SRP	SCSI RDMA Protocol (see clause 2)
SSC-4	SCSI Stream Commands - 4 (see clause 2)
TCG	Trusted Computing Group (see 3.1.179)
TCP	Transmission Control Protocol (see RFC 793)
UFS	Universal Flash Storage (developed by JEDEC)
UML	Unified Modeling Language (developed by OMG)
URI	Uniform Resource Identifier (see RFC 3986 and 3.1.183)
URL	Uniform Resource Locator (see RFC 3986 and 3.1.183)
UT	Universal time (see 3.1.182)
USB	Universal Serial Bus
USB-3	Universal Serial Bus - 3
VPD	Vital Product Data (see 7.8)
VPG1	Video Performance Guarantee Profile 1
VS	Vendor Specific (see 3.3.12)
XCDB	Extended CDB (see 3.1.53)
ZBC	Zoned Block Commands (see clause 2)

3.2.2 Symbols

Symbols used in this standard:

Symbol	Meaning
||	concatenation
n/a	not applicable
x, xx	any valid value for a bit or field
&	grammatical and
→	segment descriptor transfer data (see 3.7.5)

3.2.3 Mathematical operators

Mathematical operators used in this standard:

Mathematical operator	Meaning
+	plus
–	minus
/	divided by
×	multiplied by
<	less than
≤	less than or equal to
>	greater than
≥	greater than or equal to
≠	not equal to
||	concatenation

3.3 Keywords

3.3.1
invalid
keyword used to describe an illegal or unsupported bit, byte, word, field or code value

Note 1 to entry: Receipt by the device server of an invalid bit, byte, word, field or code value shall be reported as an error.

3.3.2
mandatory
keyword indicating an item that is required to be implemented as defined in this standard

3.3.3
may
keyword indicating flexibility of choice with no implied preference

3.3.4
may not
keyword indicating flexibility of choice with no implied preference

3.3.5
obsolete
keyword indicating that an item was defined in prior SCSI standards but has been removed from this standard

3.3.6
option, optional
keywords that describe features that are not required to be implemented by this standard

Note 1 to entry: If any optional feature defined by this standard is implemented, then it shall be implemented as defined in this standard.

3.3.7
prohibited
keyword used to describe a feature, function, or coded value that is defined in a non-SCSI standard (i.e., a standard that is not a member of the SCSI family of standards) to which this standard makes a normative reference where the use of said feature, function, or coded value is not allowed for implementations of this standard

3.3.8
reserved
keyword referring to bits, bytes, words, fields and code values that are set aside for future standardization

Note 1 to entry: A reserved bit, byte, word or field shall be set to zero, or in accordance with a future extension to this standard.

Note 2 to entry: Recipients are not required to check reserved bits, bytes, words or fields for zero values.

Note 3 to entry: Receipt of reserved code values in defined fields shall be reported as an error.

3.3.9
restricted
keyword referring to bits, bytes, words, and fields that are set aside for other identified standardization purposes

Note 1 to entry: A restricted bit, byte, word, or field shall be treated as a reserved bit, byte, word or field in the context where the restricted designation appears.

3.3.10
shall
keyword indicating a mandatory requirement

Note 1 to entry: Designers are required to implement all such mandatory requirements to ensure interoperability with other products that conform to this standard.

3.3.11
should
keyword indicating flexibility of choice with a strongly preferred alternative

3.3.12
vendor specific (VS)
something (e.g., a bit, field, code value) that is not defined by this standard

Note 1 to entry: Specification of the referenced item is determined by the SCSI device vendor and may be used differently in various implementations.

3.4 Conventions

Certain words and terms used in this standard have a specific meaning beyond the normal English meaning. These words and terms are defined either in 3.1 or in the text where they first appear. Names of commands, statuses, sense keys, and additional sense codes are in all uppercase (e.g., REQUEST SENSE). Lowercase is used for words having the normal English meaning.

If there is more than one CDB length for a particular command (e.g., MODE SENSE(6) and MODE SENSE(10)) and the name of the command is used in a sentence without any CDB length descriptor (e.g., MODE SENSE), then the condition described in the sentence applies to all CDB lengths for that command.

The names of fields are in small uppercase (e.g., ALLOCATION LENGTH). When a field name is a concatenation of acronyms, uppercase letters may be used for readability (e.g., NORMACA). Normal case is used when the contents of a field are being discussed. Fields containing only one bit are usually referred to as the name bit instead of the name field.

When the value of the bit or field is not relevant, x or xx appears in place of a specific value.

Lists sequenced by lowercase or uppercase letters show no ordering relationship between the listed items.

EXAMPLE 1 – The following list shows no relationship between the colors named:

a) red, specificity one of the following colors:
   A) crimson; or
   B) amber;
b) blue; or
c) green.

Lists sequenced by numbers show an ordering relationship between the listed items.

EXAMPLE 2 – The following list shows the order in which a page is meant to be read:

1) top;
2) middle; and
3) bottom.

Lists are associated with an introductory paragraph or phrase, and are numbered relative to that paragraph or phrase (i.e., all lists begin with an a) or 1) entry).

If a conflict arises between text, tables, or figures, the order of precedence to resolve the conflicts is text; then tables; and finally figures. Not all tables or figures are fully described in the text. Tables show data format and values. Notes do not constitute any requirements for implementors.

3.5 Numeric and character conventions

3.5.1 Numeric conventions

A binary number is represented in this standard by any sequence of digits comprised of only the Arabic numerals 0 and 1 immediately followed by a lower-case b (e.g., 0101b). Underscores or spaces may be included in binary number representations to increase readability or delineate field boundaries (e.g., 0 0101 1010b or 0_0101_1010b).

A hexadecimal number is represented in this standard by any sequence of digits comprised of only the Arabic numerals 0 through 9 and/or the upper-case English letters A through F immediately followed by a lower-case h (e.g., FA23h). Underscores or spaces may be included in hexadecimal number representations to increase readability or delineate field boundaries (e.g., B FD8C FA23h or B_FD8C_FA23h).

A decimal number is represented in this standard by any sequence of digits comprised of only the Arabic numerals 0 through 9 not immediately followed by a lower-case b or lower-case h (e.g., 25).

A range of numeric values is represented in this standard in the form “a to z”, where a is the first value included in the range, all values between a and z are included in the range, and z is the last value included in the range (e.g., the representation “0h to 3h” includes the values 0h, 1h, 2h, and 3h).

This standard uses the following conventions for representing decimal numbers:

a) the decimal separator (i.e., separating the integer and fractional portions of the number) is a period;
b) the thousands separator (i.e., separating groups of three digits in a portion of the number) is a space;
c) the thousands separator is used in both the integer portion and the fraction portion of a number; and
d) the decimal representation for a year is 1999 not 1 999.

Table 1 shows some examples of decimal numbers represented using various conventions.

Table 1 — Numbering conventions examples

French	English	This standard
0,6	0.6	0.6
3,141 592 65	3.14159265	3.141 592 65
1 000	1,000	1 000
1 323 462,95	1,323,462.95	1 323 462.95

A decimal number represented in this standard with an overline over one or more digits following the decimal point is a number where the overlined digits are infinitely repeating (e.g., 666.$\overline{6}$ means 666.666 666… or 666 2/3 and 12.$\overline{142\ 857}$ means 12.142 857 142 857… or 12 1/7).

3.5.2 Units of measure

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

a) for values based on decimal units of measure:
    1) numerical value (e.g., 100);
    2) space;
    3) prefix symbol and unit:
        1) decimal prefix symbol (e.g., M) (see table 2); and
        2) unit abbreviation (e.g., B);

and

b) for values based on binary units of measure:
   1) numerical value (e.g., 1 024);
   2) space;
   3) prefix symbol and unit:
      1) binary prefix symbol (e.g., Gi) (see table 2); and
      2) unit abbreviation (e.g., b).

Table 2 compares the prefix, symbols, and power of the binary and decimal units.

Table 2 — Comparison of decimal prefixes and binary prefixes

Decimal			Binary		
Prefix name	**Prefix symbol**	**Power (base-10)**	**Prefix name**	**Prefix symbol**	**Power (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}

3.5.3 Byte encoded character strings conventions

When this standard requires one or more bytes to contain specific encoded characters, the specific characters are enclosed in single quotation marks. The single quotation marks identify the start and end of the characters that are required to be encoded but are not themselves to be encoded. The characters that are to be encoded are shown in the case that is to be encoded.

An ASCII space character (i.e., 20h) may be represented in a string by the character '¬' (e.g., 'SCSI¬device').

The encoded characters and the single quotation marks that enclose them are preceded by text that specifies the character encoding methodology and the number of characters required to be encoded.

EXAMPLE - Using the notation described in this subclause, stating that the eleven ASCII characters 'SCSI device' represent encoded characters is the same as writing out the following sequence of byte values: 53h 43h 53h 49h 20h 64h 65h 76h 69h 63h 65h.

3.6 Bit and byte ordering

This subclause describes the representation of fields in a table that defines the format of a SCSI structure (e.g., the format of a CDB).

If a field consists of more than one bit and contains a single value (e.g., a number), the least significant bit (LSB) is shown on the right and the most significant bit (MSB) is shown on the left (e.g., in a byte, bit 7 is the MSB and is shown on the left, and bit 0 is the LSB and is shown on the right). The MSB and LSB are not labeled if the field consists of 8 or fewer bits.

If a field consists of more than one byte and contains a single value, the byte containing the MSB is stored at the lowest address and the byte containing the LSB is stored at the highest address (i.e., big-endian byte ordering). The MSB and LSB are labeled.

If a field consists of more than one byte and always contains multiple fields each with their own values (e.g., a descriptor), then there is no MSB and LSB of the field itself and thus there are no MSB and LSB labels. Each individual field has an MSB and LSB that are labeled as appropriate in the table, if any, that describes the format of the sub-structure having multiple fields.

If a field that consists of more than one byte contains either multiple fields each with their own values or a single value, then:

a) the MSB and LSB are labeled and they apply as described in this subclause whenever the field contains a single value; and
b) the labels on constituent fields supersede the single value labels whenever the field contains multiple fields.

If a field contains a text string (e.g., ASCII or UTF-8), the MSB label is the MSB of the first character and the LSB label is the LSB of the last character.

When required for clarity, multiple byte fields may be represented with only two rows in a table. This condition is represented by values in the byte number column not increasing by one in each subsequent table row, thus indicating the presence of additional bytes.

3.7 Notation conventions

3.7.1 Notation for procedure calls

In this standard, the model for functional interfaces between objects is a procedure call. Such interfaces are specified using the following notation:

[Result =] Procedure Name (IN ([input-1] [,input-2] …]), OUT ([output-1] [,output-2] …))

Where:

Result: A single value representing the outcome of the procedure call.

Procedure Name: A descriptive name for the function modeled by the procedure call. When the procedure call model is used to describe a SCSI transport protocol service, the procedure name is the same as the service name.

Input-1, Input-2, …: A comma-separated list of names identifying caller-supplied input arguments.

Output-1, Output-2, …: A comma-separated list of names identifying output arguments to be returned by the procedure call.

"[…]": Brackets enclosing optional or conditional arguments.

This notation allows arguments to be specified as inputs and outputs. An interface between entities may require only inputs. If a procedure call has no output arguments, the word OUT, preceding comma, and associated pair of balanced parentheses are omitted.

The following is an example of a procedure call specification:

Found = Search (IN (Pattern, Item List), OUT ([Item Found]))

Where:

Found = Flag
Flag: if set to one, indicates that a matching item was located.

Input Arguments:

Pattern = … /* Definition of Pattern argument */
Argument containing the search pattern.

Item List = Item<NN> /* Definition of Item List as an array of NN Item arguments*/
Contains the items to be searched for a match.

Output Arguments:

Item Found = Item … /* Item located by the search procedure call */
This argument is only returned if the search succeeds.

3.7.2 Notation for state diagrams

All state diagrams use the notation shown in figure 2.

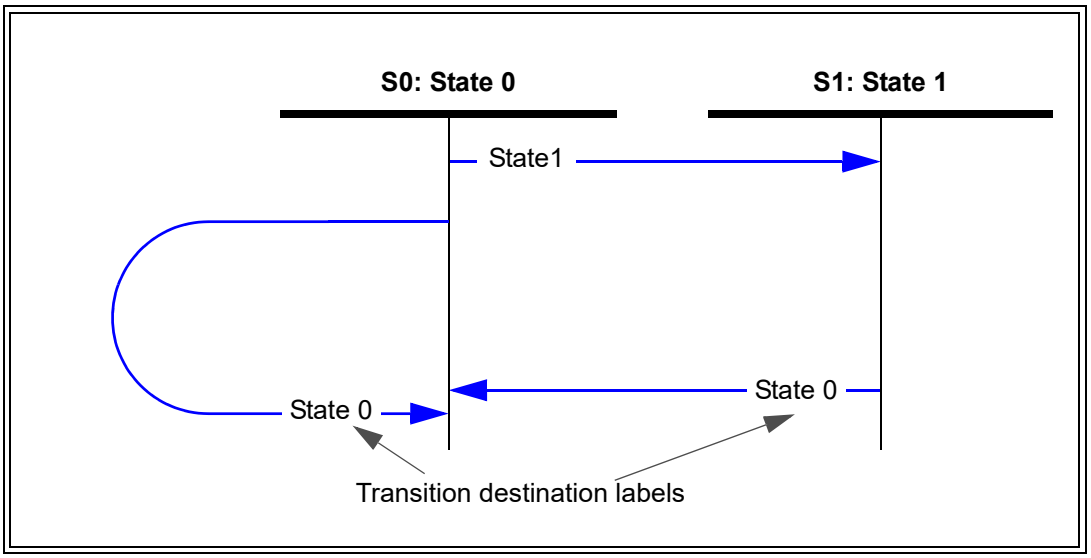


Figure 2 — Example state diagram

The state diagram is followed by subclauses describing the states and state transitions.

Each state and state transition is described in the list with particular attention to the conditions that cause the transition to occur and special conditions related to the transition.

A system described in this manner has the following properties:

a) time elapses only within discrete states; and
b) state transitions are logically instantaneous.

3.7.3 Notation for flowcharts

This standard uses flowcharts that ISO 5807:1985 defines as program flowcharts. Figure 3 shows an example flowchart.

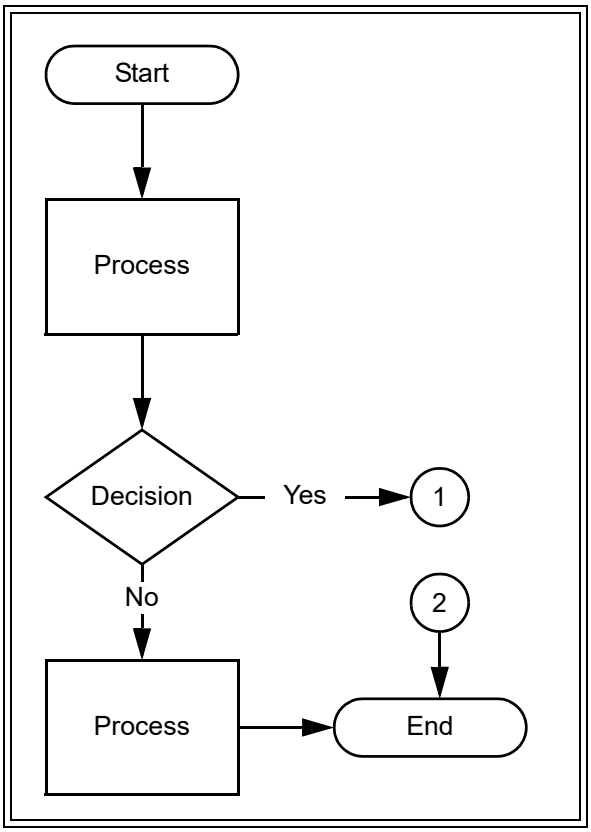


Figure 3 — Example flowchart

The following types of symbols are shown in figure 3:

a) a termination (e.g., start and end) symbol;
b) a process symbol;
c) a decision symbol; and
d) a reference (e.g., 1 and 2) symbol.

A termination symbol shows the starting point for the flowchart or the ending point for the flowchart.

A process symbol shows any kind of processing function that occurs as a result of entering this condition from a previous symbol.

A decision symbol shows a point in the progression of the flowchart from which there is more than one exit possibility of which only one is satisfied by the condition described within the decision symbol.

A reference symbol shows a connection to or from another flowchart and has the same number in both the source flowchart and destination flowchart.

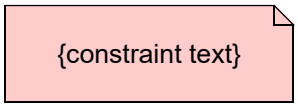
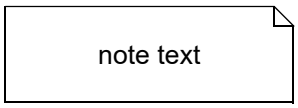
3.7.4 Notation for UML figures

3.7.4.1 Overview

This standard contains class diagram figures that use notation that is based on UML.

Some class diagrams contain constraints or notes that use the notion shown in table 3.

Table 3 — Class diagram constraints and notes notation

Notation	Description
	The presence of the curly brackets defines constraint that is a normative requirement. An example of a constraint is shown in figure 4.
	The absence of curly brackets defines a note that is informative. An example of a note is shown in figure 6.

The notation used to denote multiplicity of instances of classes and attributes in class diagrams is shown in table 4.

Table 4 — Class diagram multiplicity notation

Notation	Description
not specified	The number of instances of a class or an attribute is not specified.
1	One instance of the class or attribute exists.
0..*	Zero or more instances of the class or attribute exist.
1..*	One or more instances of the class or attribute exist.
0..1	Zero or one instance of the class or attribute exists.
n..m	n to m instances of the class or attribute exist (e.g., 2..8).
x,n..m	Multiple disjoint instances of the class or attribute exist (e.g., 2, 8..15).

Class diagrams show:

a) two or more classes (see 3.7.4.2); and
b) one or more of the following relationships between them:
   A) association (see 3.7.4.3);
   B) aggregation (see 3.7.4.4);
   C) generalization (see 3.7.4.5); and
   D) dependency (see 3.7.4.6).

3.7.4.2 Class notation

The notation used for classes is shown in table 5.

Table 5 — Class diagram notation for classes

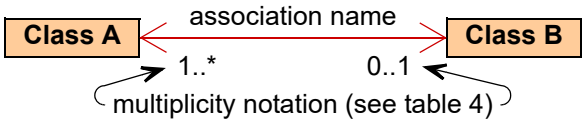
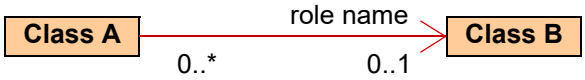
Notation	Description
Class Name **Class Name** **Class Name**	A class with no attributes or operations
Class Name Attribute01[1] Attribute02[1] **Class Name** Attribute01[1] Attribute02[1]	A class with attributes and no operations
Class Name Operation01() Operation02()	A class with operations and no attributes
Class Name Attribute01[1] Attribute02[1] Operation01() Operation02()	A class with attributes and operations
Class Name Attribute01[1..*] Attribute02[1] Operation01() Operation02()	A class with attributes that have a specified multiplicity (see table 4 in 3.7.4.1) and operations

3.7.4.3 Class association relationships notation

The notation used to denote association (i.e., "knows about") relationships between classes is shown in table 6. Unless the two classes in an association relationship also have an aggregation relationship (see 3.7.4.4),

association relationships have multiplicity notation (see table 4 in 3.7.4.1) at each end of the relationship line.

Table 6 — Class diagram notation for associations

Notation	Description
	Class A knows about Class B (i.e., read as "Class A association name Class B") and Class B knows about Class A (i.e., read as "Class B association name Class A")
	Class B knows about Class A (i.e., read as "Class B knows about Class A") but Class A does not know about Class B
	Class A knows about Class B (i.e., read as "Class A uses the role name attribute of Class B") but Class B does not know about Class A
Note: The use of role names and association names are optional.	

Several example association relationships between classes are shown in figure 4.

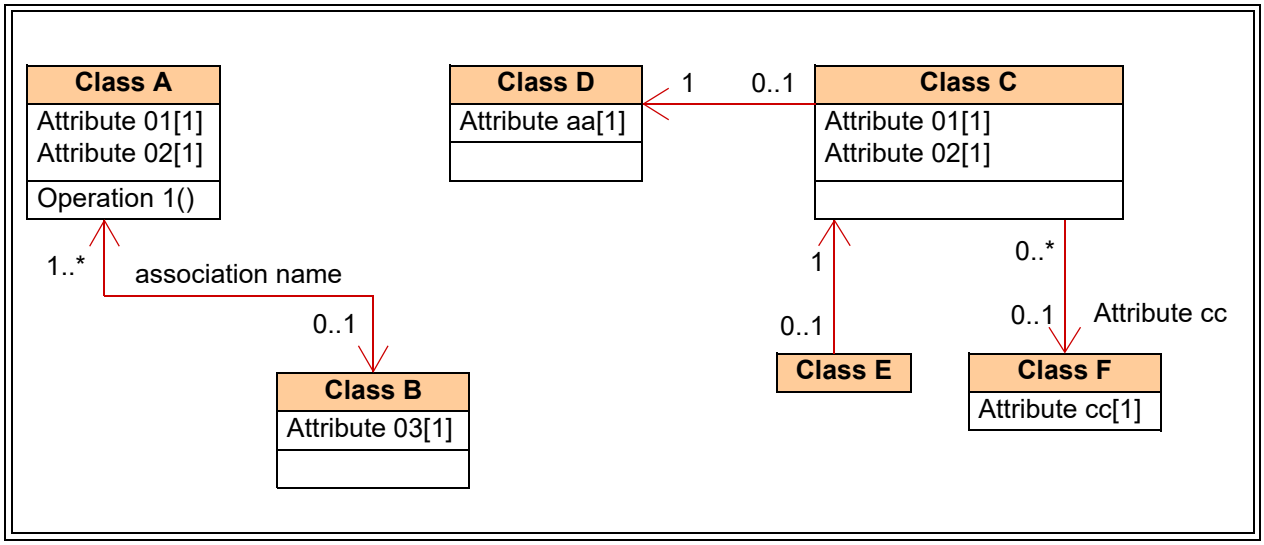


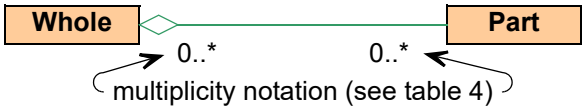
Figure 4 — Example class association relationships

3.7.4.4 Class aggregation relationships notation

The aggregation relationship is a specific type of association (see table 6). The notation used to denote aggregation (i.e., "is a part of" or "contains") relationships between classes is shown in table 7. Aggregation

relationships always include multiplicity notation (see table 4 in 3.7.4.1) at each end of the relationship line.

Table 7 — Class diagram notation for aggregations

Notation	Description
	The Part class is part of the Whole class and may continue to exist even if the Whole class is removed (i.e., read as "the Whole contains the Part.")
	The Part class is part of the Whole class, shall only belong to one Whole class, and shall not continue to exist if the Whole class is removed (i.e., read as "the Whole contains the Part.")

Several example aggregation relationships between classes are shown in figure 5.

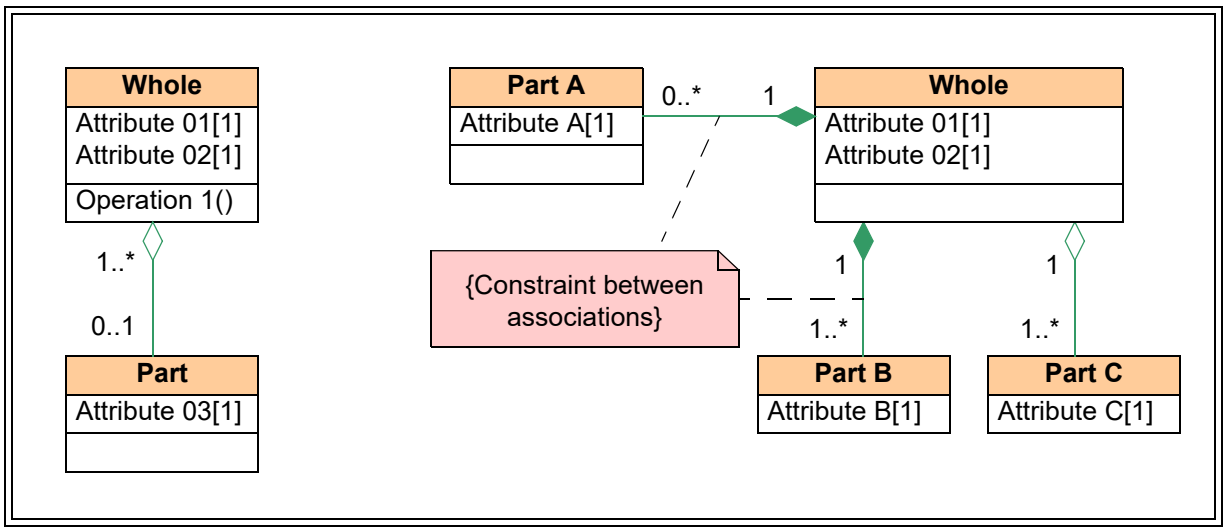


Figure 5 — Example class aggregation relationships

3.7.4.5 Class generalization relationships notation

The notation used to denote generalization (i.e., "is a kind of") relationships between classes is shown in table 8.

Table 8 — Class diagram notation for generalizations

Notation	Description
	Subclass is a kind of superclass. A subclass shares all the attributes and operations of the superclass (i.e., the subclass inherits from the superclass).

Several example generalization relationships between classes are shown in figure 6.

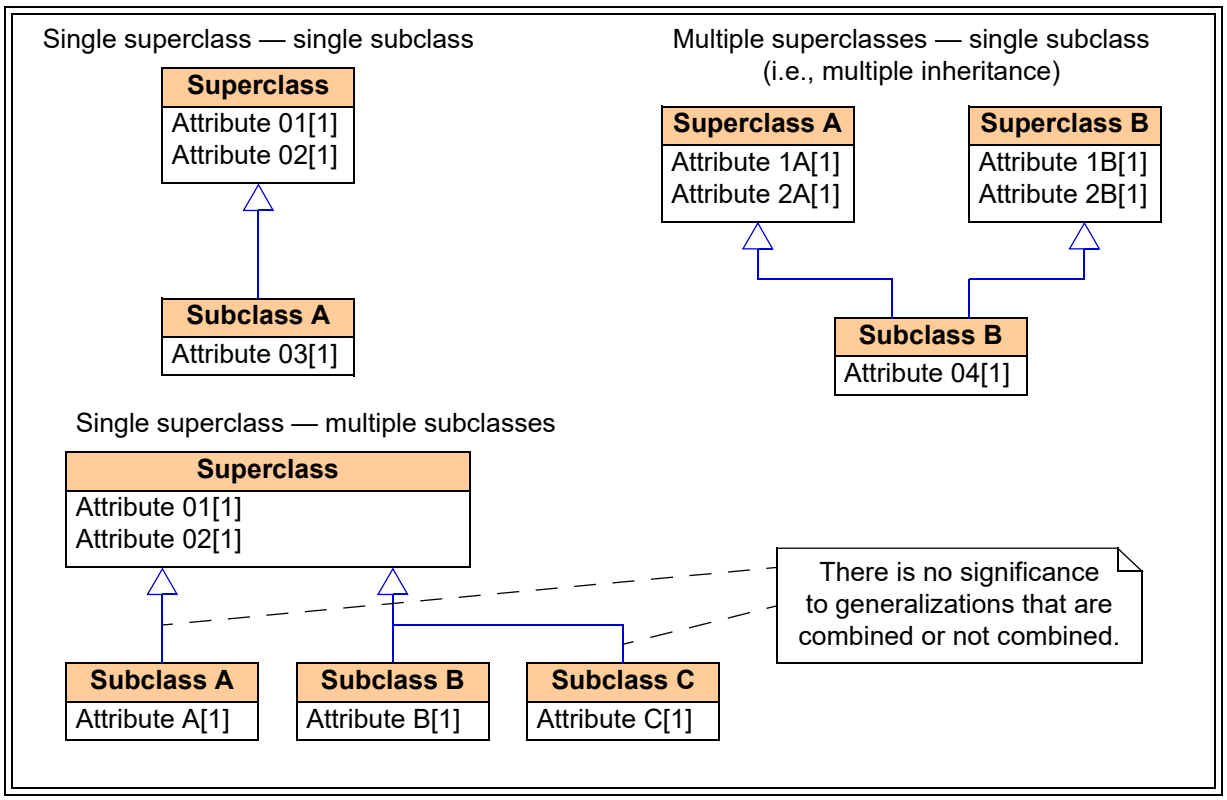


Figure 6 — Example class generalization relationships

3.7.4.6 Class dependency relationships notation

The notation used to denote dependency (i.e., "depends on") relationships between classes is shown in table 9.

Table 9 — Class diagram notation for dependencies

Notation	Description
	Class A depends on class B. A change in class B may cause a change in class A.

An example dependency relationship between classes is shown in figure 7.

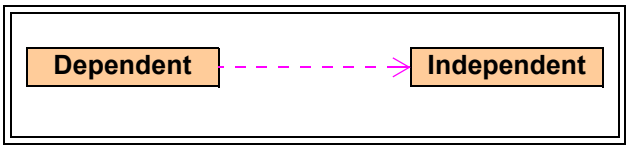


Figure 7 — Example class dependency relationships

3.7.4.7 Object notation

The notation used for objects is shown in table 10.

Table 10 — Notation for objects

Notation	Description
label : Class Name	Notation for a named object with no attributes
label : Class Name Attribute01 = x Attribute02 = y	Notation for a named object with attributes
: Class Name	Notation for an anonymous object with no attributes
: Class Name Attribute01 = x Attribute02 = y	Notation for an anonymous object with attributes

3.7.5 Notation for EXTENDED COPY command segment descriptors

The segment descriptors (see 6.4.6) contained in the parameter list for an EXTENDED COPY command (see 6.4) request the copy manager to perform operations on copy sources and copy destinations. The names of segment descriptors indicate this by inserting a right pointing arrow (i.e., →) between a brief description of the copy source and a brief description of the copy destination (e.g., block→stream).

If the segment descriptor requests the transfer of data to the copy destination and another location, an ampersand (i.e., &) is appended to the copy destination with a description of the additional location following the ampersand (e.g., block→stream&application client\).