Working Draft

T10.1/1051D

Revision 4a
February 17, 1997

Information Technology - Serial Storage Architecture - SCSI-3 Protocol (SSA-S3P)

Preliminary draft proposed American National Standard

This is an internal working document of T10, a Technical Committee of Accredited Standards Committee X3. As such, this is not a completed standard, but is considered to be technically complete by the T10.1 Task Group. The contents of this document may be modified as a result of comments received during the review process.

ABSTRACT

This document describes the SCSI-3 protocol (SSA-S3P) to be used on the Serial Storage Architecture - Transport Layer 2 (SSA-TL2).

Project Leader
John P. Scheible
IBM Corporation.
Bldg 902 Mailstop 9263
11400 Burnet Road
Austin, TX 78758
Voice (512) 823-8208
FAX (512) 838-3822
EMail scheible@vnet.ibm.com

Technical Editor
John P. Scheible
IBM Corporation.
Bldg 902 Mailstop 9263
11400 Burnet Road
Austin, TX 78758
Voice (512) 823-8208
FAX (512) 838-3822
EMail scheible@vnet.ibm.com

Reference number
ISO/IEC ***** : 199x
ANSI NCITS.*** - 199x
Printed 02/17/97

Other Points of Contact:

	T10.1 Chairman	T10.1 Vice-Chairman
	Lawrence Lamers	John P. Scheible
	Adaptec	IBM Corporation.
	MS 293	Bldg 902 Mailstop 9263
	691 South Milpitas Blvd	11400 Burnet Road
	Milpitas CA, 95035	Austin, TX 78758
Voice:	408-957-7817	(512) 823-8208
Fax:	408-957-7193	(512) 838-3822
Email:	ljlamers@aol.com	scheible@vnet.ibm.com

NCITS Secretariat
Administrator Standards Processing
NCITS Secretariat
1250 Eye Street, NW Suite 200
Washington, DC 20005

Voice: 202-737-8888
FAX: 202-638-4922
Email: x3sec@itic.nw.dc.us

SSA Reflector

Internet address for subscription to the T10.1 reflector:	majordomo@symbios.com
Note should contain a line stating...	subscribe X3T10-SSA *your email address*
Internet address for distribution via T10.1 reflector:	x3t10-ssa@symbios.com
T10 Bulletin Board	719-533-7950
FTP Site:	ftp.symbios.com /pub/standards/io/x3t10.1
Web sites:	http://www.x3.org/x3t10 or http://www.symbios.com/x3t10 http://www.ssaia.org

Document Distribution
Global Engineering
15 Inverness Way East
Englewood, CO 80112-5704

Voice: 303-792-2181
or: 800-854-7179
FAX: 303-792-2192

PATENT STATEMENT

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

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

Revision History

Significant technical changes from SSA-S3P rev 4 are shown with revision marks.

T10.1/1051D revision 4a (February 16, 1997)

During the February 5, 1997 T10.1 working group, I was instructed to generate revision 4a of SSA-S3P with the following modifications:

a) 97a103r3 - Letter ballot comment resolution on SSA-S3P rev 4

X3T10.1/1051D revision 3 (September 11, 1996)

During the August 28 1996 X3T10.1 plenary, I was instructed to generate revision 3 of SSA-S3P with the following modifications:

a) 95a150r1 - SSA-S3P support of CA and ACA proposal
b) 96a155r0 - Comments on TL2 and S3P (selected parts as modified)
c) 96a159r0 - SSA-S3P changes for SMS validation.
d) 96a161r1 - AER reporting between ULP and TL2 layers.

X3T10.1/1051D revision 2 (July 25, 1996)

During the June 26th 1996 X3T10.1 plenary, I was instructed to generate revision 2 of SSA-S3P with the following modifications:

a) Add a section for SCSI-3 family as is common in the other SCSI-3 documents.
b) 96a136r2 - Update Annexes for SAM Services model changes.
c) 96a140r2 - Third party support for SSA-S3P.
d) 96a141r1 - Extend Buffer Full Condition to cover BUSY.
e) Modify the format to match the ongoing work to meet ANSI standards.

X3T10.1/1051D revision 1 (May 21, 1996)

During the May 1st 1996 X3T10.1 plenary, I was instructed to generate revision 1 of SSA-S3P based off of SSA-S2P rev 7 rather than the out of date SSA-SSP rev 0 with the following modifications:

a) Update the boilerplate to SSA-S3P.
b) Change terminology to match SCSI-3
c) Change SCA to ACA.
d) 95a133r1 - SCSI STATUS SMS Enhancement proposal, including AER support.
e) 96a131r0 - Confirmed delivery service
f) 96a136r0 - Add SAM Services to front Model section.
g) Update TAG interaction.
h) Add a new SMS entitled LOGICAL UNIT RESET SMS to match the similar message in SCSI-3.
i) Add new disconnect/reconnect fields to be ignored in Control Mode page (new to SCSI-3) and remove the Control Mode parameters section since AER is now supported.

ANSI®
NCITS.*-199x**

American National Standard
for Information Systems -

Serial Storage Architecture - SCSI-3 Protocol (SSA-S3P)

Secretariat
Information Technology Industry Council

Approved Month dd, yy

American National Standards Institute, Inc.

Abstract

This document describes the SCSI-3 mapping protocol (SSA-S3P) to be used on the Serial Storage Architecture - Transport Layer 2 (SSA-TL2).

American National Standard

Approval of an American National Standard requires verification by ANSI that the requirements for due process, consensus, and other criteria for approval have been met by the standards developer. Consensus is established when, in the judgment of the ANSI Board of Standards Review, substantial agreement has been reached by directly and materially affected interests. Substantial agreement means much more than a simple majority, but not necessarily unanimity. Consensus requires that all views and objections be considered, and that a concerted effort be made towards their resolution.

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

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

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

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

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

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

Contents Page

Annexes

Tables

Figures

Foreword (This foreword is not part of American National Standard NCITS.nnn-199n.)

This standard was developed by Task Group T10.1 of Accredited Standards Committee NCITS during 1993-96. The standards approval process started in 1995. This document includes informative annexes which are not considered part of the standard.

Requests for interpretation, suggestions for improvement and addenda, or defect reports are welcome. They should be sent to the NCITS Secretariat, Information Technology Industry Council, 1250 Eye Street, NW, Suite 200, Washington, DC 20005-3922.

This standard was processed and approved for submittal to ANSI by Accredited Standards Committee on Information Processing Systems, NCITS. Committee approval of the standard does not necessarily imply that all committee members voted for approval. At the time it approved this standard, the NCITS Committee had the following members:

James D. Converse, Chairman

Donald C. Loughry, Vice-Chairman

Joanne M. Flanagan, Secretary

Organization Represented	*Name of Representative*
American Nuclear Society	Geraldine C. Main
	Sally Hartzell (Alt.)
AMP, Inc	Edward Kelly
	Charles Brill (Alt.)
Apple Computer	Karen Higginbottom
Association of the Institute for Certification of Professionals (AICCP)	Kenneth Zemrowski
AT&T/NCR	Thomas W. Kern
	Thomas F. Frost (Alt.)
Boeing Company	Catherine Howells
	Andrea Vanosdoll (Alt.)
Bull HN Information Systems, Inc.	William George
Compaq Computer Corporation	James Barnes
Digital Equipment Corporation	Delbert Shoemaker
	Kevin Lewis (Alt.)
Eastman Kodak	James D. Converse
	Michael Nier (Alt.)
GUIDE International	Frank Kirshenbaum
	Harold Kuneke (Alt.)
Hewlett-Packard	Donald C. Loughry
Hitachi America, Ltd.	John Neumann
	Kei Yamashita (Alt.)
Hughes Aircraft Company	Harold L. Zebrack
IBM Corporation	Joel Urman
	Mary Anne Lawler (Alt.)
National Communication Systems	Dennis Bodson
National Institute of Standards and Technology	Robert E. Roundtree
	Michael Hogan (Alt.)
Northern Telecom, Inc	Mel Woinsky
	Subhash Patel (Alt.)
Neville & Associates	Carlton Neville
Recognition Technology Users Association	Herbert P. Schantz
	G. Edwin Hale (Alt.)
SHARE, Inc	Gary Ainsworth
	David Thewlis (Alt.)
Sony Corporation	Michael Deese
Storage Technology Corporation	Joseph S. Zajaczkowski
	Samuel D. Cheatham (Alt.)
Sun Microsystems	Scott Jameson
	Gary Robinson (Alt.)

3M Company	Eddie T. Morioka
	Paul D. Jahnke (Alt.)
Unisys Corporation	John L. Hill
	Stephen P. Oksala (Alt.)
U.S. Department of Defense	William C. Rinehuls
	C. J. Pasquariello (Alt.)
U.S. Department of Energy	Alton Cox
	Lawrence A. Wasson (Alt.)
U.S. General Services Administration	Douglas Arai
	Larry L. Jackson (Alt.)
Wintergreen Information Services	Joun Wheeler
Xerox Corporation	Dwight McBain
	Roy Peirce (Alt.)

Technical Committee T10 on I/O Interfaces, which reviewed this standard, had the following members:

John B. Lohmeyer, Chairman

Lawrence J. Lamers, Vice-Chairman

Ralph Weber, Secretary

Mr. I. Dal Allan
Mr. Tak Asami
Mr. Geoffrey Barton
Mr. Paul Boulay
Mr. John Cannon
Mr. Joe Chen
Ms. Nancy Cheng
Mr. Roger Cummings
Mr. William Dallas
Mr. Jan V. Dedek
Mr. Mark Evans
Mr. Edward Fong
Mr. Raymond Gilson
Mr. Dave Guss
Mr. Kenneth J. Hallam
Mr. Norm Harris
Mr. Mike Hetzel
Mr. Peter Johansson
Mr. Skip Jones
Mr. Thomas J. Kulesza
Mr. Edward Lappin
Mr. Robert Liu
Mr. John Masiewicz
Mr. Pete McLean
Dr. Akira James Miura
Mr. Ian Morrell
Mr. Jay Neer
Mr. Vit Novak
Mr. Duncan Penman
Mr. Kevin R. Pokorney
Mr. Ron Roberts
Mr. Steve D. Schueler
Mr. Robert N. Snively
Mr. Arlan P. Stone
Mr. Matthew Thomas
Mr. Dennis Van Dalsen
Mr. Roger Wang
Mr. Bob Whiteman
Mr. Devon Worrell
Mr. Ruben Yomtoubian

Mr. Paul D. Aloisi
Mr. Akram Atallah
Mr. Robert Bellino
Mr. Charles Brill
Mr. Ting Li Chan
Mr. Andy Chen
Mr. Mike Chennery
Mr. Chris D'Iorio
Mr. Joe Dambach
Mr. Dhiru N. Desai
Mr. Timothy Feldman
Mr. Edward A. Gardner
Mr. Chuck Grant
Mr. Peter Haas
Dr. William Ham
Mr. Edward Haske
Mr. Stephen Holmstead
Mr. Gerry Johnsen
Mr. Richard Kalish
Mr. Lawrence J. Lamers
Mr. Pat LaVarre
Mr. John Lohmeyer
Mr. David McFadden
Mr. Patrick Mercer
Mr. E.J. Mondor
Mr. John Moy
Mr. Marc A. Noblitt
Mr. Erich Oetting
Mr. George Penokie
Mr. Gary Porter
Mr. Frank Samela
Mr. J. R. Sims
Mr. Jeff Stai
Mr. Clifford E. Strang Jr.
Mr. Pete Tobias
Mr. Joseph Wach
Mr. Gary M. Watson
Mr. Jeffrey L. Williams
Mr. Anthony Yang

Mr. David Andreatta
Mr. Wayne Baldwin
Mr. Rick Bohn
Mr. Michael Bryan
Mr. Shufan Chan
Mr. Jack Chen
Mr. Dan Colegrove
Mr. Zane Daggett
Mr. Brian N. Davis
Mr. Mike Eneboe
Mr. Stephen G. Finch
Mr. John Geldman
Mr. Louis Grantham
Mr. Douglas Hagerman
Mr. Tom Hanan
Mr. Stephen F. Heil
Mr. Gerald Houlder
Mr. Brian Johnson
Mr. Greg Kapraun
Mr. Dennis Lang
Mr. Jaff Lin
Mr. Bill Mable
Mr. James McGrath
Mr. Gene Milligan
Mr. Charles Monia
Mr. S. Nadershahi
Mr. Tim Norman
Mr. Dennis Pak
Mr. Doug Piper
Mr. Robert Reisch
Mr. John P. Scheible
Mr. Michael Smith
Mr. Gary R. Stephens
Mr. Nicos Syrimis
Mr. Tokuyuki Totani
Mr. Dean Wallace
Mr. Ralph O. Weber
Mr. Michael Wingard
Mr. Danny Yeung

Task Group T10.1 on Serial Storage Architecture, which developed this standard, had the following members:

Lawrence J. Lamers, Chairman

John Scheible, Vice-Chairman

Charles Brill	David Deming	Mark DeWilde
Wolfgang Drichelt	Neil Edmunds	Chuck Grant
Bill Ham	Brad Kitson	Lawrence Lamers
John Scheible	Kelly Tenuta	Michael Wingard
Greg Alvey	Bob Atkinson	Robert Bellino
Adge Hawes	Bill Mable	Gary Manchester
Charles Monia	Brian Morissette	Said Rahmani
Herb Silverman	Daniel Tsai	

Introduction

This standard is divided into the following clauses and annexes.

Clause 1 defines the scope of the Serial Storage Architecture - SCSI-3 Protocol (SSA-S3P).

Clause 2 specifies the normative references.

Clause 3 defines the definitions, symbols and abbreviations.

Clause 4 contains the general SSA-S3P information.

Clause 5 contains the frame information.

Clause 6 contains the SSA-S3P SMS (SSA Message Structure).

Clause 7 contains transport related items.

Clause 8 contains SSA changes from parallel SCSI-2.

Annex A is informative and contains information on conversion from parallel SCSI-2 to SSA-S3P.

Annex B is informative and contains information on differences between parallel SCSI-2 to SSA-S3P.

Annex C is informative and contains the protocol services model.

Annex D is informative and contains the transport services model.

American National Standard
for Information Systems -

Information Technology -
Serial Storage Architecture (SSA-S3P)

1 Scope

This document defines a protocol layer of the Serial Storage Architecture (SSA) that runs on SSA-TL2 while running on SSA-PH2. The goals of SSA-S3P are:

a) map the SAM services and terminology to SSA;
b) define the data field format of the SSA-S3P SMSs;
c) support for dual port and alternate pathing;
d) provide support for auto-sense;
e) provide support for third-party operations;
f) other capabilities that fit within the scope of SSA-S3P that may be proposed during the development phase by the participants in the project.

This document defines the SCSI-3 protocol layer of the Serial Storage Architecture (SSA). SSA defines a serial interface hierarchy to be used for purposes within its distance and performance characteristics, including but not limited to storage subsystems. This standard is intended to be used with a transport layer protocol [e.g., SSA Transport Layer 2 (SSA-TL2)] and a physical layer [e.g., SSA Physical Layer 2 (SSA-PH2)].

1.1 SSA family of standards

This document describes an upper-level protocol of Serial Storage Architecture. SSA-S3P is a mapping of the existing SCSI-2 protocol, X3.131-1994, with extensions to map SCSI-2 to the SSA serial link.

Figure 1 shows the relationships of the SSA standards

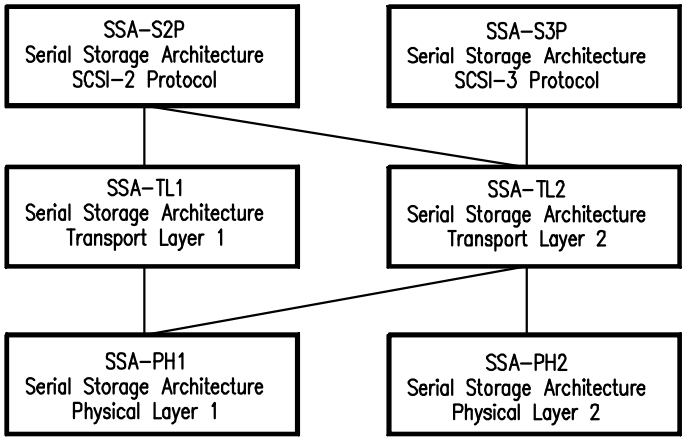

Figure 1 - Relationship of the SSA standards

1.1.1 SSA-S2P

Serial Storage Architecture - SCSI-2 Protocol (SSA-S2P) defines the SCSI-2 Protocol used to runs with the SSA transport layers 1 or 2. SSA-S2P is intended to operate on the following transport and physical layers: SSA-TL1 with SSA-PH1, SSA-TL2 with SSA-PH1, and SSA-TL2 with SSA-PH2.

1.1.2 SSA-TL1

Serial Storage Architecture - (SSA-TL1) Transport Layer 1 defines the Transport layer that supports SSA-S2P and requires SSA-PH1.

1.1.3 SSA-PH1

Serial Storage Architecture - (SSA-PH1) Physical Layer 1 defines the Physical layer that supports SSA-TL1 and SSA-TL2, and consists of the electrical characteristics of the interface and the connectors.

1.1.4 SSA-S3P

Serial Storage Architecture - (SSA-S3P) SCSI-3 Protocol, this standard, defines the SCSI-3 Protocol used with the SSA transport layer 2. SSA-S3P is intended to operate on the following transport and physical layers: SSA-TL2 with SSA-PH1, and SSA-TL2 with SSA-PH2.

1.1.5 SSA-TL2

Serial Storage Architecture - (SSA-TL1) Transport Layer 2 defines the Transport layer that supports SSA-S2P and SSA-S3P and requires SSA-PH1 or SSA-PH2

1.1.6 SSA-PH2

Serial Storage Architecture - (SSA-PH2) Physical Layer 2 (Project X3T10.1/1146) defines the Physical layer that supports SSA-TL2, and consists of the electrical characteristics of the interface and the connectors.

1.2 SCSI-3 family of standards

Figure 2 is intended to show the relationship of this document to other SCSI-3 standards. The figure is not intended to imply a relationship such as hierarchy, protocol stack, or system architecture. It indicates the applicability of a standard to the implementation of a given transport.

SCSI-3 Architecture Roadmap

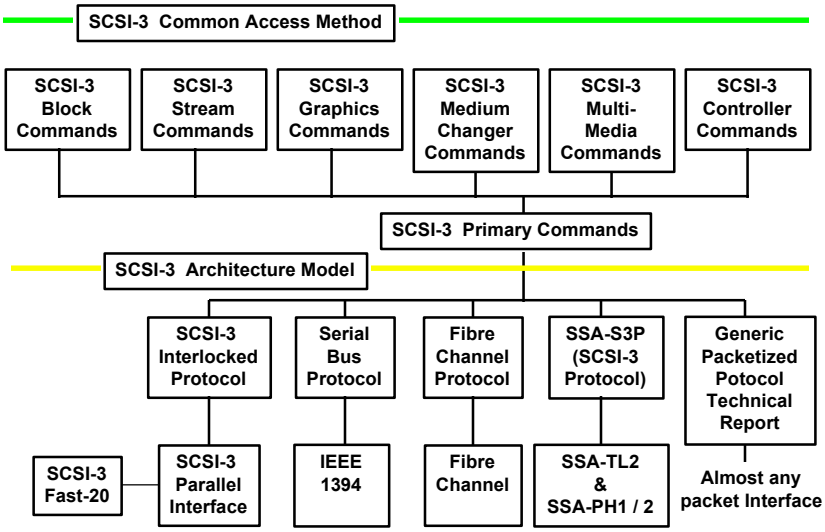

Figure 2 - SCSI-3 Family

2 References

2.1 Normative references

The following standards contain provisions which, through reference in SSA-S3P, constitute provisions of this standard. At the time of publication, the edition indicated was valid. All standards are subject to revision, and parties to agreements based on this standard are encouraged to investigate the possibility of applying the most recent editions of the standard listed below. Members of IEC and ISO maintain registers of currently valid International Standards, and ANSI maintains registers for American National Standards.

X3T10 / 994D - *Information technology - SCSI-3 Architectural model*

X3T10 / 995D - *American National Standard for information systems - SCSI-3 Primary Commands*

X3T10.1 / 1147D - *Information Technology, Serial Storage Architecture - Transport Layer 2 (SSA-TL2)*

Editor s note: This section will be adjusted at time of publication to reference appropriate standards.

2.2 Informative references

The following related standards are for informational purposes only, and do not contain provisions of this standard. At the time of publication, the edition indicated was valid. All standards are subject to revision, and parties to agreements based on this standard are encouraged to investigate the possibility of applying the most recent editions of the standard listed below. Members of IEC and ISO maintain registers of currently valid International Standards, and ANSI maintains registers for American National Standards.

X3.293:1996 - *Information Technology - Serial Storage Architecture - Physical Layer 1 (SSA-PH1)*

X3.295:1996 - *Information Technology - Serial Storage Architecture - Transport Layer 1 (SSA-TL1)*

X3.294:1996 - *Information Technology - Serial Storage Architecture - SCSI-2 Protocol (SSA-S2P)*

X3T10.1 / 1146D - *Information Technology, Serial Storage Architecture - Physical Layer 2 (SSA-PH2)*

Editor s note: This clause will be adjusted at time of publications to reference appropriate standards.

3 Definitions, symbols, abbreviations and conventions

3.1 Definitions

3.1.1 Channel: The facilities in a port to receive an SMS or a single data transfer.

3.1.2 character: A sequence of 10 encoded bits that represents a data byte or a protocol character.

3.1.3 Contingent Allegiance: An SCSI-2 concept of an error condition that is cleared following the execution of the next command. It does not work well in serial interfaces, and has been replaced by the concept of Auto Contingent Allegiance.

3.1.4 Data frame: An Application frame with a non-zero channel component.

3.1.5 destination node: The node that receives a particular frame.

3.1.6 field: A group of related data characters in a frame, (e.g. the CRC field).

3.1.7 frame: A sequence of 6 or more data characters surrounded by FLAG characters.

3.1.8 I_T nexus: The specific initiator and target participating in an SCSI command.

3.1.9 I_T_L nexus: The specific initiator, target and logical unit for an SCSI command.

3.1.10 link: A serial connection between two ports.

3.1.11 logical path: A unidirectional conduit for the ordered delivery of SMS and data frames between an initiator-target pair.

3.1.12 nexus: A relationship identifying the addressable SCSI entities associated with an SSA SMS transfer.

3.1.13 node: A system, controller or device with one or more ports.

3.1.14 port: The hardware and firmware that implements one end of a link.

3.1.15 Auto Contingent Allegiance: An error condition that holds all queued tasks until cleared explicitly by a reset or a CLEAR ACA CONDITION SMS and allows error recovery while the queued tasks are held.

3.1.16 SMS: The data field portion of a frame with a Channel component of 00h. The FRAME TYPE field information shall be considered part of the SMS.

3.1.17 source: The node that originates a particular frame.

3.1.18 Web: A collection of SSA nodes that may address each other, interconnected into a dedicated connection, a loop, a string, or a complex configuration (i.e., includes a switch).

3.2 Symbols and abbreviations

CDB	Command Descriptor Block
CRC	cyclic redundancy check.
DDRM	Disable DATA READY SMSs
DMA	direct memory access
ERP	error recovery procedure
LUN	Logical Unit Number
N/A	Not Applicable
OOT	Out of Order Transfer
POR	power-on reset
POST	power-on self-test
Qerr	A bit in the SCSI mode select pages that controls the clearing of the Queue following a ACA.
RAS	reliability, availability and serviceability
RPL	Rotational Position Locking
ACA	Auto Contingent Allegiance
SCSI	Small Computer Systems Interface
SSA	Serial Storage Architecture
SYNC	A signal (8B/10B character) that allows separate spindle motors to be synchronized.
SMS	SSA Message Structure
&	Logical AND
=	Assignment or comparison for EQUAL
<	Comparison for LESS THAN
+	ADD
-	SUBTRACT

3.3 Conventions

Certain words and terms used in this standard have a specific meaning beyond the normal English meaning. These words and terms are defined in the text with the first letter capitalized. Lower case is used for words having the normal English meaning.

Fields containing only one bit are usually referred to as the named bit instead of the named field. When a bit is set, it s value is 1. When a bit is cleared, it s value is 0.

Numbers that are not immediately followed by lower-case "b" or "h" are decimal values.

Numbers immediately followed by lower-case "b" (xxb) are binary values.

Numbers immediately followed by lower-case "h" (xxh) are hexadecimal values.

Decimal fractions are indicated with a comma(e.g., two and one half is represented as "2,5). Decimal numbers having a value exceeding 999 are represented with a space (e.g., 24 255).

Table 1 illustrates the bit ordering used within a byte in SSA-S3P.

Table 1 - Bit ordering in a byte

Bit 7	6	5	4	3	2	1	Bit 0
msb							lsb

Reserved bits, fields, bytes, and code values are set aside for future standardization. Their use and interpretation may be specified by future extensions to this standard. A reserved bit, field, or byte shall be set to zero, or in accordance with a future extension of this standard. Reserved bits, fields, bytes, or reserved field values shall be ignored when cut-through routing a frame. A destination node that receives a reserved bit, field, or byte that is not zero, or receives a reserved code value shall terminate the SMS as described in 6.2.

Ignored bits, fields, or bytes shall be ignored by the receiving node. Any value is considered valid.

SMS names are shown as all capital letters, such as SCSI STATUS SMS. Field names are shown as small capital letters, such as the STATUS field. Field values are shown as all capital letters, such as the STATUS field TASK SET FULL value. Variables are shown in italics.

The byte ordering convention is Big Endian (i.e., the most significant byte of a number is sent first).

In case of conflict, figures take precedence over tables, and both figures and tables take precedence over text.

4 General

4.1 Overview

It is intended that SSA-S3P should conform as closely as possible to the existing SCSI-3 logical model. This minimizes the programming changes required to convert existing systems and devices from the parallel bus to an SSA interface. Therefore the following functions of the SSA-S3P are identical to SCSI:

a) Tagged queuing;
b) Command descriptor blocks;
c) Status byte;
d) Sense bytes.

Except where necessary for clarity the above functions are not described in this document (Please refer to the ANSI SCSI-3 standards for information). SSA-S3P concentrates on mapping the following aspects of parallel SCSI-3:

a) Bus functions;
b) Addressing;
c) Messages.

SSA-S3P supports Webs containing strings, loops and switches. The concepts of initiator, target and Logical Unit are retained although SSA-S3P supports larger configurations than parallel SCSI-3. Initiators and targets may be freely mixed throughout the Web. Each node may have from 1 to 126 physical ports.

SSA-S3P offers the following benefits compared to the parallel SCSI bus:

a) Open-ended Webs with alternative paths for availability and performance;
b) Full-duplex communication with spatial reuse on strings and loops;
c) Frame multiplexing;
d) No overhead for arbitration, disconnection or reselection;
e) Integrated spindle synchronization for array applications;
f) Fewer initiator-target exchanges;
g) Concurrent tasks on the same device or different devices;
h) Out-of-order data transfers.

Untagged queuing may be simply emulated with Tagged Queuing. In SSA, the target effectively disconnects after each frame because of frame multiplexing. Hence each command shall have a tag for identification.

Untagged command queuing may be simply emulated by having the initiator have only one outstanding command, re-use the same tag, and use the simple queue type.

4.2 Model

4.2.1 SAM services

The SSA-S3P protocol layer maps the SAM Services model with some additional enhancements onto the SSA-TL2 transport layer services.

4.2.2 Logical paths

The SSA-S3P protocol layer provides a logical path between the Application Client and the node identified by a Unique ID. This outgoing logical path guarantees ordered delivery of tasks. A feature in addition to the SAM Services is the ability of the Application Client to request multiple logical paths and control what tasks use which logical path. If the task uses the Confirmed Status feature, the SSA-S3P layer will automatically redirect all outstanding tasks to use an alternate path (if any exist) in the event of a path failure.

4.2.2.1 Use of multiple outgoing logical paths

If an initiator uses multiple outgoing logical paths, then ordering across logical paths is not guaranteed (although ordering is guaranteed within a logical path).

4.2.2.2 Use of multiple return logical paths

If an initiator uses multiple return logical paths, then receipt of SCSI STATUS SMSs received may not indicate the order the commands were executed within the target (although ordering is guaranteed within a logical path).

4.2.3 Path control

The SSA-S3P protocol layer controls the use of paths to guarantee a logical path when needed.

4.2.4 Tag control

The SSA-TL2 transport layer controls the allocation of TAG field values. This is required to meet the requirement that TAG field values be unique across all outstanding tasks that use data transfer SMSs for all simultaneous protocol layers being used by the I_T nexus.

4.2.5 Data channels

The SSA-TL2 transport layer controls the allocation of data channel for data transfer.

4.2.6 Outstanding Commands table and Outstanding Tasks table

The SSA-S3P protocol layer maintains a list of outstanding commands and tasks for the following reasons:

a) map the appropriate completion SMSs to the task;
b) transparently handle link failure by redirecting tasks;
c) provide an interlock for TASK SET FULL, BUSY and ACA ACTIVE;
d) provide a means to emulate Contingent Allegiance.

5 Frames

All SSA-S3P SMSs and data are transmitted in Application frames whose format is described in SSA-TL2. A Data frame is an Application frame with a non-zero Channel component.

To improve system bandwidth, individual frames are delivered without end-to-end acknowledgment (this mode of delivery is referred to as datagram service). Such acknowledgment, when required, is more efficient if performed by the higher level protocol on protocol-specific transactions, such as an S3P SCSI command or task management function.

6 SSA-S3P message structure

SMSs are used to communicate control information between the target and the initiator. They are typically used for commands, status and controlling tasks. Some SSA-S3P SMSs are similar in function to SCSI messages but there is not a one-to-one mapping and they should not be confused.

The ADDRESS field in an SMS contains the Path component (address to the destination node) followed by the byte 00h to select the SMS Channel component. The length of the DATA field depends on the particular SMS. Each SMS shall be fully contained in a single frame.

SSA-S3P SMS frames are identified by a Channel component value of 00h, and the first byte (byte 0) of the DATA field containing SSA-S3P SMS CODE value of 83h. The second byte (byte 1) of the DATA field of SSA-S3P SMSs contains the S3P CODE field identifying the function of SSA-S3P SMS.

The SCSI COMMAND SMS optionally contains a 2-byte CHANNEL field. This contains the Channel component for the ADDRESS field of data frames. If the Channel component is a single byte, the CHANNEL field contains the Channel component left aligned, with the second byte padded with any value by the source node and ignored by the destination node.

All SMSs contain a TAG field that is used to relate replies to the original request (see 6.5).

Any SMS received with a valid SSA-S3P SMS code whose DATA field meets any of the error conditions in sub-clause 6.2 shall cause the specified error response. An SMS has a maximum length of 32 bytes. Only the bytes defined in the appropriate SMS are required , however the SMS may be padded up to a total length of 32 bytes. The source node shall pad (if any) with zeros, and the destination node shall ignore any pad bytes.

Table 2 defines the generic structure of an SSA-S3P SMS generated by the initiator.

Table 2 - Generic SMS format for initiator generated SSA-S3P SMSs

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE (83h)							
1	S3P CODE							
2	TAG							
3	TAG							
4	RETURN PATH ID							
5	RETURN PATH ID							
6	RETURN PATH ID							
7	RETURN PATH ID							
8	LUN							
9	(Depends on the SMS)							
up to 31	(Depends on the SMS)							

All SMSs from the initiator to the target contain a RETURN PATH ID field, a value supplied by the target during configuration to identify the logical path to the appropriate initiator (see SSA-TL).

The LUN field need not be present, but if present shall reside in byte 8.

Table 3 defines the generic structure of an SSA-S3P SMS generated by the initiator.

Table 3 - Generic SMS format for target generated SSA-S3P SMSs

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE (83h)							
1	S3P CODE							
2	TAG							
3	TAG							
4	(Depends on the SMS)							
up to 31	(Depends on the SMS)							

The following sub-clauses define the SMSs that are used by SSA-S3P.

6.1 Summary of SSA-S3P SMSs

Table 4 shows a summary of all SSA-S3P SMSs.

Table 4 - Summary of SSA-S3P SMSs

SMS name	Byte 0 SMS CODE	Byte 1 S3P CODE	SMS FRAME TYPE	Node type support	
				Sent by	Received by
SCSI RESPONSE	83h	03h	APPLICATION	target	initiator
SCSI COMMAND	83h	10h	APPLICATION	initiator	target
SCSI STATUS	83h	11h	APPLICATION	target	initiator
CONFIRM STATUS	83h	12h	APPLICATION	initiator	target
ENABLE AER	83h	20h	APPLICATION	initiator	target
AER	83h	21h	APPLICATION	target	initiator
ABORT TASK	83h	30h	APPLICATION	initiator	target
ABORT TASK SET	83h	31h	APPLICATION	initiator	target
CLEAR TASK SET	83h	32h	APPLICATION	initiator	target
TARGET RESET	83h	33h	APPLICATION	initiator	target
CLEAR ACA	83h	34h	APPLICATION	initiator	target
LOGICAL UNIT RESET	83h	35h	APPLICATION	initiator	target

6.2 SMS validation

Prior to the SSA-S3P protocol layer receiving an SMS, the transport layer has validated the SMS CODE field to be 83h. Each SMS then undergoes three levels of validation as described in the following three clauses.

6.2.1 SMS code validation

After the transport layer checks the SMS for validity, the SSA-S3P layer shall perform validity checks on byte 1, the S3P CODE, in the following order. If the Asynchronous Alert process is invoked then the SMS is terminated without any other response. The SMS code validity check process is as follows in order:

a) If the S3P CODE is not a value supported in Table 4, then invoke the Asynchronous Alert process with an UNKNOWN SMS ALERT CODE field (see SSA-TL2 for a description of ASYNC ALERT SMS format).

b) If the S3P CODE is supported in Table 4, but the node does not match the corresponding type (initiator or target), then invoke the Asynchronous Alert process with an UNKNOWN SMS ALERT CODE field.

c) If Table 4 matches the S3P CODE of the SMS, but the FRAME TYPE field does not have a value of APPLICATION then invoke the Asynchronous Alert process with an SMS UNKNOWN ALERT CODE field.

d) If the RETURN PATH ID field is unknown, then invoke the Asynchronous Alert process with an UNKNOWN RETURN PATH OR RETURN PATH ID ALERT CODE field value.

e) If the S3P CODE corresponds to the ABORT TASK SET, ABORT TASK, CLEAR TASK SET, TARGET RESET, or CLEAR ACA CONDITION SMS, and one of these SMSs is already outstanding in the destination node from this Initiator, then generate an SCSI RESPONSE SMS with an OVERLAPPED SMSs ATTEMPTED RETURN CODE value.

f) If the SMS is an SCSI COMMAND SMS, then destination node shall perform the following functions:

   1) If the destination node has no room to store the SMS and the destination node has not discarded the previous SCSI COMMAND SMS from this Initiator, then the destination node shall discard the SMS and generate an SCSI STATUS SMS indicating TASK SET FULL status.

   2) If the destination node has discarded the previous SCSI COMMAND SMS from this Initiator and the SMS has the RESUME bit cleared, the destination node shall discard the SCSI COMMAND SMS even if room to store the SMS has become available. The destination node shall not generate an SCSI STATUS or SCSI RESPONSE SMS to the discarded SCSI COMMAND SMS.

   3) If the destination node has no room to store the SMS, the destination node has discarded the previous SCSI COMMAND SMS from this Initiator, and the SMS has the RESUME bit set, the

destination node shall discard the SMS and generate an SCSI STATUS SMS indicating a STATUS value of TASK SET FULL.

4) If the destination node has not discarded the previous SCSI COMMAND SMS from this Initiator and the SCSI COMMAND SMS has the RESUME bit set, the destination node shall discard the SMS and generate an SCSI RESPONSE SMS with a RETURN CODE value of INVALID FIELD.
5) If the destination node has room to store the SMS, the destination node has discarded the previous SCSI COMMAND SMS from this Initiator, and the SMS has the RESUME bit set, the destination node shall continue SMS validation processing as defined in sub-clauses 6.2.2 and 6.2.3.

6.2.2 SMS length validation

If the SMS length is shorter than that specified, then invoke the Asynchronous Alert process with an SMS TOO SHORT ALERT CODE field. If the Asynchronous Alert process is invoked the original SMS is terminated, without generating an SCSI STATUS or SCSI RESPONSE SMS.

6.2.3 SMS field validation

Each SMS listed in Table 4 shall perform the following SMS field validations in the order shown. If the Asynchronous Alert process is invoked then the SMS is then terminated, without any other response.

a) If the initiator receives an SCSI RESPONSE or SCSI STATUS SMS with an invalid TAG field, then generate an ASYNC ALERT with an SMS UNEXPECTED ALERT CODE field value.
b) If any reserved field is non-zero or valid field contains a reserved code value, then generate an SCSI RESPONSE SMS with a RETURN CODE value of INVALID FIELD. There is one exception to this case, where any improper field values within the CDB or LUN fields of the SCSI COMMAND SMS shall reply with an SCSI STATUS SMS with STATUS field value of CHECK CONDITION STATUS and sense data as defined in parallel SCSI-3.

6.3 SMS Buffer Full condition

The S3P protocol layer of the destination node shall enter an SMS Buffer Full condition for a given initiator when an SCSI COMMAND SMS is received from that initiator and the command cannot be accepted due to a TASK SET FULL or BUSY. Upon entering an SMS Buffer Full condition, the destination node shall terminate the SCSI COMMAND SMS and generate an SCSI STATUS SMS indicating a TASK SET FULL or BUSY status. When in an SMS Buffer Full condition, the destination node shall discard SCSI COMMAND SMSs with a RESUME bit cleared from the same Initiator even if the condition which caused the SMS Buffer Full condition no longer exists.

The destination node shall exit the SMS Buffer Full condition after receiving an SCSI COMMAND SMS with a RESUME bit set from the Initiator with the SMS Buffer Full condition. If the destination node still cannot accept the command due to a TASK SET FULL or BUSY, the destination node shall once again enter the SMS Buffer Full condition.

When an Initiator S3P protocol layer receives an SCSI STATUS SMS indicating TASK SET FULL or BUSY status, the Initiator shall mark the command identified by the SMS and any subsequent commands already issued to the destination node as having completed with a TASK SET FULL or BUSY status. The S3P protocol layer then has the option of reporting the TASK SET FULL or BUSY status to the Application client or may reissue the commands itself after a delay. The Initiator S3P protocol layer shall send the next SCSI COMMAND SMS with the RESUME bit set.

6.4 Limitation on task management SMSs

An Initiator shall only have one ABORT TASK SET, ABORT TASK, CLEAR TASK SET, TARGET RESET, or CLEAR ACA CONDITION SMS outstanding to a given target. A target shall provide storage to process one of these SMSs for each Initiator and therefore, need never discard one, provided the initiator follows the SSA-S3P requirements.

6.5 TAG field values

6.5.1 Tag interaction within SSA-S3P

The TAG field value is assigned by the initiator and it shall be unique among all the SSA-S3P TAG field values that are currently active from that initiator on any target or Logical Unit throughout the Web. Effectively the TAG field identifies the target, LUN and Queue-tag components of the SCSI-2 nexus (the initiator and logical path are identified by the RETURN PATH ID field in the SMS). The initiator uses the TAG field value of the SCSI STATUS SMS or SCSI RESPONSE SMS to determine both the source (target, LUN) and the SMS associated with the SCSI STATUS SMS or SCSI RESPONSE SMS. Since the target identifies processes with both a TAG and RETURN PATH ID field, TAG field values outstanding at a target need not be unique between initiators,.

6.5.2 Tag interaction between ULPs

Since the data transfer SMSs do not have a ULP CODE field, the TAG values shall be unique for all outstanding commands using data transfer SMSs (see SSA-TL) from a given initiator for all active protocols. An initiator using only one simultaneous protocol need not worry about this restriction.

6.5.3 Tag interaction between SSA-S3P and SSA-TL2

Since there is no overlap between the responses to SSA-S3P SMS, and SSA-TL2 SMSs (including ASYNC ALERT SMSs), there is no requirement that the TAG field values be unique across outstanding SSA-S3P and SSA-TL2 SMSs.

6.6 SCSI COMMAND SMS

The SCSI COMMAND SMS is sent from the initiator to a target to initiate an task or to send the next command in a series of linked commands. If the SCSI COMMAND SMS may be successfully parsed, the target may respond by initiating a data transfer and sending an SCSI STATUS SMS. See 6.2 for invalid SMS processing that may result in either an ASYNC ALERT SMS with various ALERT CODE fields or an SCSI RESPONSE SMS with a RETURN CODE of INVALID FIELD.

Table 5 defines the contents of the DATA field in an SCSI COMMAND SMS.

Table 5 - SCSI COMMAND SMS

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE (83h)							
1	S3P CODE (10h)							
2	TAG							
3	TAG							
4	RETURN PATH ID							
5	RETURN PATH ID							
6	RETURN PATH ID							
7	RETURN PATH ID							
8	LUN							
9	reserved							
10	DDRM	OOT	RESUME	CONFIRM	reserved		QUEUE CNTL	
11	reserved=00h							
12	CHANNEL							
13	CHANNEL							
14	reserved							
15	reserved							
16	COMMAND DESCRIPTOR BLOCK							
...	COMMAND DESCRIPTOR BLOCK							
m	COMMAND DESCRIPTOR BLOCK							

The TAG field is assigned by the initiator and used to relate subsequent SMSs to this task. The value used in the TAG field becomes available for re-assignment when the task completes or is terminated. All SCSI COMMAND SMSs in a linked list shall specify the same value for the TAG field.

NOTE 1 - Parallel SCSI-3 requires all TAG field values to be unique for a given I_T_L nexus, in addition SSA requires that the tag field values be unique for an initiator, regardless of the I_T_L nexus. SSA uses the TAG field value to identify the target/LUN and the SCSI COMMAND SMS the SCSI STATUS SMS is associated with. Therefore, an easy conversion from a parallel SCSI-3 device driver to SSA-S3P is to include the parallel SCSI concepts of TAG field (1 byte), target address (4 bits), and LUN (4 bits) into the 2 byte SSA-S3P TAG field.

The RETURN PATH ID field is used by the target to locate the initiator table entry containing the return path, port and Unique ID of the initiator that issued the SCSI COMMAND SMS. Each SCSI COMMAND SMS in a series of linked commands for a given task shall use the same value for the RETURN PATH ID field.

The LUN field refers to a Logical Unit. All SCSI COMMAND SMSs in a linked list shall specify the same values for the LUN field.

If the DDRM (Disable DATA READY SMS) bit is set, DATA READY SMSs (see SSA-TL2) are disabled for data transfers from the target to the initiator. If the DDRM bit is set, the initiator shall specify the channel to be used in the CHANNEL field. If the DDRM bit is cleared, then the target shall issue a DATA READY or DATA REQUEST SMS to initiate data transfer (see SSA-TL2). The DDRM bit is ignored for any commands that do not transfer data to the initiator.

If the OOT (Out-of-Order Transfer) bit is set the target or Logical Unit is permitted to transfer read data to or write data from the initiator out-of-order. This enables a device to get the maximum advantage from an out-of-order read or out-of-order write. It is also useful for array controllers, e.g. RAID-5. If the OOT bit is cleared then data shall be transferred sequentially from the beginning. For control commands the OOT bit is ignored and data is always transferred in order from the beginning. If both the OOT and DDRM bits are set then the initiator shall be able to identify each logical block from a header within the block.

The RESUME bit controls the processing of QUEUE FULL and BUSY conditions within the target protocol layer. The initiator normally clears the RESUME bit when processing SCSI COMMAND SMSs. After receiving an SCSI STATUS SMS from a destination node indicating TASK SET FULL or BUSY status, the initiator protocol layer shall set the RESUME bit for the next SCSI COMMAND SMS issued to the destination node in an attempt to resume SCSI COMMAND SMS processing. Receiving an SCSI COMMAND SMS with the RESUME bit set informs the destination node to stop discarding SCSI COMMAND SMSs (see 6.3).

If the CONFIRM bit is set the target does not consider the task complete until the appropriate CONFIRM STATUS SMS is received with the COMPLETE bit set. If the CONFIRM bit is cleared, the target considers the task complete after sending the SCSI STATUS SMS, and the receipt of an associated CONFIRM STATUS SMS creates an Asynchronous Alert (see 6.8). If both the OOT and CONFIRM bits are set, then generate a SCSI STATUS SMS with a RETURN CODE value of INVALID FIELD.

The QUEUE CNTL field controls how the task is to be queued at the specified Logical Unit, as defined in Table 6. For a linked list of commands the QUEUE CNTL field is ignored in all SCSI COMMAND SMSs except the first.

Table 6 - QUEUE CNTL field values

Value	Description
00b	AUTO CONTINGENT ALLEGIANCE
01b	HEAD OF QUEUE
10b	ORDERED QUEUE
11b	SIMPLE QUEUE

The target only executes tasks with the AUTO CONTINGENT ALLEGIANCE QUEUE CNTL field value while a Auto Contingent Allegiance condition exists for the same nexus (see 8.2). For example, a Request Sense command specifying a QUEUE CNTL field value of AUTO CONTINGENT ALLEGIANCE is used to retrieve sense information after a command terminates with STATUS field of CHECK CONDITION.

The target places an task with the QUEUE CNTL field value of HEAD OF QUEUE first in the queue, to be executed next when the currently active task finishes.

The target shall execute a sequence of tasks with the QUEUE CNTL field value of ORDERED QUEUE in the strict order that they were received.

The target is permitted to reorder a sequence of tasks with the QUEUE CNTL field value SIMPLE QUEUE. This allows a device to optimize performance (e.g. by using an elevator seeking algorithm). All previously issued ORDERED tasks shall be executed prior to executing any of the SIMPLE tasks. Any subsequently issued ORDERED task shall be executed after all the SIMPLE tasks have completed.

The CHANNEL field specifies the Channel component of the ADDRESS field for data frames that the target sends to the initiator (the channel to be used within the initiator). CHANNEL is ignored for any commands that do not transfer data to the initiator, or when the DDRM bit is cleared.

The COMMAND DESCRIPTOR BLOCK field is the Command Descriptor Block (CDB) as defined by SCSI-3. For SCSI-3 commands, the CDB is 6, 10, 12 or 16 bytes long. The last byte of the CDB is always a Control byte containing the NACA, FLAG and LINK bits. The NACA bit controls the use of Contingent Allegiance or Auto Contingent Allegiance. The FLAG bit is returned to the initiator in the SCSI STATUS SMS corresponding to the command. When set, the FLAG bit typically causes the initiator to interrupt the application to indicate that a linked command has completed. When set, the LINK bit indicates that the task is to be continued with a further command. In this case the initiator sends a further SCSI COMMAND SMS in response to the SCSI STATUS SMS for the current command, provided the FLAG bit is set and the STATUS field has a value of INTERMEDIATE or INTERMEDIATE-CONDITION MET.

6.7 SCSI STATUS SMS

The SCSI STATUS SMS is sent from a target to an initiator to indicate that a command and any associated data transfer have been terminated or completed. The SCSI STATUS SMS is returned using the RETURN PATH ID field specified in the SCSI COMMAND SMS.

The SCSI STATUS SMS is returned for each SCSI COMMAND SMS unless the command is rejected (with an ASYNC ALERT SMS or an SCSI RESPONSE SMS) or the command is cleared by any of the following:

a) ABORT TASK SMS,
b) ABORT TASK SET SMS,
c) CLEAR TASK SET SMS,
d) TARGET RESET SMS,
e) A Hard Reset condition (including a transport layer Total Reset or Absolute Reset frame).

Table 7 defines the contents of the DATA field in the SCSI STATUS SMS.

Table 7 - SCSI STATUS SMS

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE (83h)							
1	S3P CODE (11h)							
2	TAG							
3	TAG							
4	STATUS							
5	reserved						FLAG	LINK
6	RETURN CODE							
7	reserved							
8	SENSE DATA							
...	SENSE DATA							
31	SENSE DATA							

The TAG field is a copy of the TAG field in the corresponding SCSI COMMAND SMS. It allows the initiator to associate the status with the correct task.

The STATUS field contains status as defined by SCSI-3

SSA-S3P uses a single S3P CODE value with imbedded FLAG and LINK bits, rather than create three different completion messages as parallel SCSI-3 does. Table 8 defines which completion messages are associated with the FLAG and LINK bits.

Table 8 - Meaning of flag and link bits

FLAG	LINK	Corresponding parallel SCSI-2 completion message
0	0	Command Complete
0	1	Linked Command Complete
1	0	reserved
1	1	Linked Command Complete with Flag

Table 21 defines the RETURN CODE field indicating the result of the SCSI COMMAND SMS.

Table 9 - RETURN CODE values for the SCSI STATUS SMS

Value	Description
00h	THE SCSI COMMAND WAS PARSED SUCCESSFULLY.
01h-FEh	reserved
FFh	INVALID FIELD

Table 10 defines the contents of the SENSE DATA field. If the STATUS field value is not CHECK CONDITION STATUS, then the SCSI STATUS SMS may be truncated to 8 bytes. If all of the Sense Data could be reported via the SCSI STATUS SMS (i.e. ADDITIONAL SENSE LENGTH field < 16), then the target may clear the sense data.

Table 10 - SENSE DATA field

Byte	Bit 7	6	5	4	3	2	1	Bit 0
8	VALID	ERROR CODE (70h or 71h)						
9	SEGMENT NUMBER							
10	FILEMARK	OEM	ILI	reserved	SENSE KEY			
11	INFORMATION							
12	INFORMATION							
13	INFORMATION							
14	INFORMATION							
15	ADDITIONAL SENSE LENGTH							
16	COMMAND SPECIFIC INFORMATION							
17	COMMAND SPECIFIC INFORMATION							
18	COMMAND SPECIFIC INFORMATION							
19	COMMAND SPECIFIC INFORMATION							
20	ADDITIONAL SENSE CODE							
21	ADDITIONAL SENSE QUALIFIER							
22	FIELD REPLACEABLE UNIT							
23	SKSV	SENSE KEY SPECIFIC						
24	SENSE KEY SPECIFIC							
25	SENSE KEY SPECIFIC							
26	ADDITIONAL SENSE BYTES							
27	ADDITIONAL SENSE BYTES							
28	ADDITIONAL SENSE BYTES							
29	ADDITIONAL SENSE BYTES							
30	ADDITIONAL SENSE BYTES							
31	ADDITIONAL SENSE BYTES							

6.8 CONFIRM STATUS SMS

The CONFIRM STATUS SMS is sent from an initiator to a target to either confirm receipt of a SCSI STATUS or to instruct the target to restart the task from a specified point (see 8.3).

The CONFIRM STATUS SMS is returned for each SCSI COMMAND SMS with the CONFIRM bit set unless the command is rejected (with an ASYNC ALERT SMS or an SCSI RESPONSE SMS) or the command is cleared by any of the following:

a) ABORT TASK SMS,
b) ABORT TASK SET SMS,
c) CLEAR TASK SET SMS,
d) TARGET RESET SMS,
e) A Hard Reset condition (including a transport layer Total Reset or Absolute Reset frame).

Table 11 defines the contents of the DATA field in the CONFIRM STATUS SMS.

Table 11 - CONFIRM STATUS SMS

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE (83h)							
1	S3P CODE (12h)							
2	TAG							
3	TAG							
4	OLD RETURN PATH ID							
5	OLD RETURN PATH ID							
6	OLD RETURN PATH ID							
7	OLD RETURN PATH ID							
8	NEW RETURN PATH ID							
9	NEW RETURN PATH ID							
10	NEW RETURN PATH ID							
11	NEW RETURN PATH ID							
12	BYTE OFFSET							
13	BYTE OFFSET							
14	BYTE OFFSET							
15	BYTE OFFSET							
16	COMPLETE	reserved						

The TAG field is a copy of the TAG field in the corresponding SCSI COMMAND SMS.

The OLD RETURN PATH ID field is a copy of the RETURN PATH ID of the SCSI COMMAND SMS. The combination of TAG and OLD RETURN PATH ID fields identifies the correct task.

The NEW RETURN PATH ID field replaces the RETURN PATH ID field of the SCSI COMMAND SMS. If the task is active, any active data transfer is terminated, and all future activity uses the NEW RETURN PATH ID field to route data and SMSs.

The BYTE OFFSET field indicates the number of bytes of read data that has been received by the initiator, or write data that has been sent from the initiator. An BYTE OFFSET field value of zero indicates the target shall execute the entire task. An BYTE OFFSET field value equal to the task transfer count indicates that the initiator received all the data, but has not seen a SCSI STATUS SMS.

If the COMPLETE bit is set, the CONFIRM STATUS SMS is used to confirm the receipt of the SCSI STATUS SMS. If the COMPLETE bit is cleared, the CONFIRM STATUS SMS is used to potentially change the RETURN PATH ID for an task and to potentially restart the task at a given point (BYTE OFFSET field).

If a CONFIRM STATUS SMS is received for an task that had the CONFIRM bit cleared in the SCSI COMMAND SMS, then generate an Asynchronous Alert (see SSA-TL2) with an ALERT CODE field value of SMS UNEXPECTED. If a CONFIRM STATUS SMS is received for an task which is unknown to the target (i.e. TAG field and OLD RETURN PATH ID field does not match an existing task) then generate an Asynchronous Alert (see SSA-TL2) with an ALERT CODE field value of SMS UNEXPECTED. No check is performed to verify that the NEW RETURN PATH ID field is associated with the same initiator pointed to by the OLD RETURN PATH ID field. If the BYTE OFFSET field is larger than the transfer count of the task, then generate an Asynchronous Alert (see SSA-TL2) with an ALERT CODE field value of INVALID FIELD. If the target does not receive an associated CONFIRM STATUS SMS within 5 seconds of sending a SCSI STATUS SMS, then generate an Asynchronous Alert with an ALERT CODE value of UNCONFIRMED STATUS.

6.9 ENABLE AER SMS

The ENABLE AER SMS is sent from an initiator to a target to enable Asynchronous Event Reporting by supplying a TAG and RETURN PATH ID field to be used by the target for any future AER REPORT SMS between the target and the particular initiator that sent the ENABLE AER SMS. The ENABLE AER SMS is also used to disable Asynchronous Event Reporting. The target shall respond with a SCSI RESPONSE SMS using the TAG value specified in the ENABLE AER SMS.

Table 12 defines the contents of the DATA field in the ENABLE AER SMS.

Table 12 - ENABLE AER SMS

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE (83h)							
1	S3P CODE (20h)							
2	TAG							
3	TAG							
4	RETURN PATH ID							
5	RETURN PATH ID							
6	RETURN PATH ID							
7	RETURN PATH ID							
8	LUN							
9	reserved							AEROFF
10	AER TAG							
11	AER TAG							

The TAG field is used to associate the AER SMS with a particular target.

The RETURN PATH ID field identifies the logical path that shall be used by the target to return the associated AER SMS.

If the AEROFF bit is set, then asynchronous event reporting is disabled for the appropriate initiator, and any asynchronous event shall be reported as a deferred error to that initiator. If the AEROFF bit is cleared, then asynchronous event reporting is enabled for the appropriate initiator, and any pending deferred error shall be reported as an asynchronous event to that initiator. A power on condition causes Asynchronous event reporting to be disabled. A QUIESCE SMS (see SSA-TL2) disables Asynchronous Event Reporting for the initiator being quiesced.

The AER TAG field is shall be placed in the tag field of any AER SMS sent to this initiator.

6.10 AER SMS

The AER SMS is sent from a target to an initiator to indicate an Asynchronous Event Report. The initiator responds with a SCSI RESPONSE SMS using the TAG value specified in the AER SMS. If all of the Sense Data could be reported via the AER SMS (i.e. ADDITIONAL SENSE LENGTH field < 16), then the receipt of the associated SCSI RESPONSE SMS clears the AER sense data in the target.

Asynchronous events are only sent to the appropriate initiators that have enabled them via the ENABLE AER SMS with the AEROFF bit cleared (see 6.9). If an initiator has disabled asynchronous event reporting, then asynchronous events are reported as deferred errors using the SCSI STATUS SMS.

Table 13 defines the contents of the DATA field in the AER SMS.

Table 13 - AER SMS

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE (83h)							
1	S3P CODE (21h)							
2	TAG							
3	TAG							
4	reserved							
5	reserved							
6	reserved							
7	reserved							
8	SENSE DATA							
...	SENSE DATA							
31	SENSE DATA							

The TAG field is a copy of the AER TAG field in the corresponding ENABLE AER SMS, and identifies the target.

Table 10 defines the contents of the SENSE DATA field.

6.11 ABORT TASK SMS

The ABORT TASK SMS is sent from an initiator to a target to abort a particular task. Other tasks are not affected. The ABORT TASK SMS shall be sent over the logical path used to create the task (see 8.5).

Previously established conditions including MODE SELECT parameters, reservations and the Auto Contingent Allegiance condition shall not be changed by the ABORT TASK SMS.

Before issuing the ABORT TASK SMS the initiator s S3P layer should terminate any related outbound data transfer to ensure that data is not sent to a non-existent nexus. The logical unit terminates execution of the task if it has already begun or removes the task from the queue if execution has not begun. The logical unit shall not send a SCSI STATUS SMS for the aborted command after the SCSI RESPONSE SMS has been sent for the ABORT TASK SMS. Upon receiving the SCSI RESPONSE SMS (see D.3.1), the initiator s S3P layer removes the aborted command if present in its Outstanding Commands Table.

Table 14 defines the contents of the DATA field in the ABORT TASK SMS.

Table 14 - ABORT TASK SMS

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE (83h)							
1	S3P CODE (30h)							
2	TAG							
3	TAG							
4	RETURN PATH ID							
5	RETURN PATH ID							
6	RETURN PATH ID							
7	RETURN PATH ID							
8	TAG 2							
9	TAG 2							

The TAG field is used to relate the SCSI RESPONSE SMS to the ABORT TASK SMS.

The RETURN PATH ID field identifies the logical path, that shall be used by the target to send the associated SCSI RESPONSE SMS.

The TAG 2 field contains the TAG of the task to be aborted. It is not an error if TAG 2 is unknown to the target since the execution of the task may have already completed.

In the SCSI RESPONSE SMS, the RETURN CODE field shall have a value of REQUESTED FUNCTION COMPLETED SUCCESSFULLY, or TASK NOT FOUND. If the task has already completed then the SCSI STATUS SMS may have been sent prior to the SCSI RESPONSE SMS.

6.12 ABORT TASK SET SMS

The ABORT TASK SET SMS is sent from an initiator to a target to abort all tasks from that initiator for a selected Logical Unit. Tasks from other initiators are not affected. If the initiator is using multiple logical paths to the target, the initiator shall issue an ABORT TASK SET SMS over each different logical path used by tasks for the I_T_L nexus (see 8.5).

Previously established conditions including MODE SELECT parameters, reservations and the Auto Contingent Allegiance condition shall not be changed by the ABORT TASK SET SMS.

Before issuing the ABORT TASK SET SMS the initiator should terminate any related outbound data transfers to ensure that data is not sent to a non-existent nexus. A SCSI STATUS SMS shall be sent for each task that completed prior to sending the SCSI RESPONSE SMS. After sending the SCSI RESPONSE SMS, the target shall not send an SCSI STATUS SMS for any aborted command. Upon receiving the SCSI RESPONSE SMS, the initiator s S3P layer removes all aborted commands from its Outstanding Commands Table (see D.3.1).

Table 15 defines the contents of the DATA field in the ABORT TASK SET SMS.

Table 15 - ABORT TASK SET SMS

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE (83h)							
1	S3P CODE (31h)							
2	TAG							
3	TAG							
4	RETURN PATH ID							
5	RETURN PATH ID							
6	RETURN PATH ID							
7	RETURN PATH ID							
8	LUN							

The TAG field is used by the initiator to associate the SCSI RESPONSE SMS to the ABORT TASK SET SMS.

The RETURN PATH ID field specifies the logical path, that shall be used by the target to return the associated SCSI RESPONSE SMS.

The LUN field the Logical Unit. If an unsupported or unattached LUN is specified, an SCSI RESPONSE SMS is generated with a RETURN CODE of INVALID FIELD.

In the SCSI RESPONSE SMS, the RETURN CODE field shall have a value of REQUESTED FUNCTION COMPLETED SUCCESSFULLY or INVALID FIELD.

6.13 CLEAR TASK SET SMS

The CLEAR TASK SET SMS is sent from an initiator to a target to abort all tasks from all initiators for a specified Logical Unit. If the initiator is using multiple logical paths to the target, the initiator shall issue a CLEAR TASK SET SMS over each different logical path used by tasks for the I_T_L nexus (see 8.5).

Before issuing the CLEAR TASK SET SMS the initiator should terminate any related outbound data transfers to ensure that data is not sent to a non-existent nexus. If the target has issued any DATA REQUEST SMSs to other initiators on behalf of the selected Logical Unit, then the target shall receive and flush all of the requested data before aborting the corresponding tasks.

All pending status and data for that Logical Unit for all initiators shall be cleared. A SCSI STATUS SMS shall be sent for each task that completed prior to the SCSI RESPONSE SMS. Once the target has sent the SCSI RESPONSE SMS for the CLEAR TASK SET request, no status shall be sent for any aborted task. The target sets Unit Attention for all initiators that had an task was aborted. The sense code indicates COMMANDS CLEARED BY ANOTHER INITIATOR. Upon receiving the SCSI RESPONSE SMS, the initiator s S3P layer removes all aborted commands from its Outstanding Commands Table (see D.3.1).

Table 16 defines the contents of the DATA field in the CLEAR TASK SET SMS.

Table 16 - CLEAR TASK SET SMS

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE (83h)							
1	S3P CODE (32h)							
2	TAG							
3	TAG							
4	RETURN PATH ID							
5	RETURN PATH ID							
6	RETURN PATH ID							
7	RETURN PATH ID							
8	LUN							

The TAG field is used to relate the SCSI RESPONSE SMS to the CLEAR TASK SET SMS.

The RETURN PATH ID field identifies the logical path that shall be used by the target to return the associated SCSI RESPONSE SMS.

The LUN field specifies the Logical Unit. If an unsupported or unattached LUN is specified, an SCSI RESPONSE SMS is generated with a RETURN CODE of INVALID FIELD.

In the SCSI RESPONSE SMS, the RETURN CODE field shall have a value of REQUESTED FUNCTION COMPLETED SUCCESSFULLY or INVALID FIELD.

6.14 TARGET RESET SMS

The TARGET RESET SMS is sent from an initiator to a target to abort all tasks for all initiators on all Logical Units. If the initiator is using multiple logical paths to the target, the initiator shall issue a TARGET RESET SMS over each different logical path used by tasks for the I_T nexus (see 8.5). The target executes a hard reset that shall perform the following:

a) Abort all tasks.
b) Clear all Auto Contingent Allegiance conditions.
c) Release all SCSI device reservations.
d) Returns any device operating modes to their appropriate initial conditions similar to those that may be found following a device power-on. The MODE SELECT conditions shall be restored to their last saved values if saved values have been established. Any MODE SELECT conditions that have no saved values established shall be returned to their default values.

The SSA-TL2 constructs of Initiator Tables, pending asynchronous alerts, the state of asynchronous event reporting and unprocessed SMSs shall be unaffected. There shall be no effect on other upper level protocols running in the node.

Before issuing the TARGET RESET SMS the initiator should terminate any related outbound data transfers to ensure that data is not sent to a non-existent nexus. If the target has issued any DATA REQUEST SMSs to other initiators then the target shall receive and flush all the requested data before aborting the corresponding tasks.

A SCSI STATUS SMS shall be sent for each task that completed before the SCSI RESPONSE SMS for the TARGET RESET was sent. Once the target has sent the SCSI RESPONSE SMS, no status shall be sent for any aborted task. However the target shall set the Unit Attention condition for all initiators. The ASC/ASCQ indicates POWER-ON OR RESET OCCURRED. The target always replies to the TARGET RESET SMS with an SCSI RESPONSE SMS. Upon receiving the SCSI RESPONSE SMS, the initiator s S3P layer shall remove all aborted commands from its Outstanding Commands Table (see D.3.1).

Table 17 defines the contents of the DATA field in a TARGET RESET SMS.

Table 17 - TARGET RESET SMS

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE (83h)							
1	S3P CODE (33h)							
2	TAG							
3	TAG							
4	RETURN PATH ID							
5	RETURN PATH ID							
6	RETURN PATH ID							
7	RETURN PATH ID							

The TAG field is used to relate the SCSI RESPONSE SMS to the TARGET RESET SMS.

The RETURN PATH ID field identifies the logical path that shall be used by the target to return the associated SCSI RESPONSE SMS.

In the SCSI RESPONSE SMS, the RETURN CODE field shall be REQUESTED FUNCTION COMPLETED SUCCESSFULLY.

6.15 CLEAR ACA SMS

The CLEAR ACA SMS is sent from an initiator to a target to clear a Auto Contingent Allegiance condition for that initiator and a selected Logical Unit. After the Auto Contingent Allegiance condition is cleared, any suspended queued command for that initiator may become an active task subject to the SCSI-3 ordering rules. Auto Contingent Allegiance conditions for other initiators, Logical Units are not affected. The target shall reply to the CLEAR ACA SMS with an SCSI RESPONSE SMS.

Table 18 defines the contents of the DATA field in the CLEAR ACA SMS.

Table 18 - CLEAR ACA SMS

Byte	**Bit 7**	**6**	**5**	**4**	**3**	**2**	**1**	**Bit 0**
0	SMS CODE (83h)							
1	S3P CODE (34h)							
2	TAG							
3	TAG							
4	RETURN PATH ID							
5	RETURN PATH ID							
6	RETURN PATH ID							
7	RETURN PATH ID							
8	LUN							

The TAG field is used to relate the SCSI RESPONSE SMS to the CLEAR ACA SMS.

The RETURN PATH ID field identifies the logical path, that shall be used by the target to return the associated SCSI RESPONSE SMS.

The LUN field specifies the Logical Unit. If an unsupported or unattached LUN is specified, an SCSI RESPONSE SMS is generated with a RETURN CODE of INVALID FIELD.

In the SCSI RESPONSE SMS, the RETURN CODE field shall be one of REQUESTED FUNCTION COMPLETED SUCCESSFULLY, NO ACA CONDITION EXISTED FOR THE ADDRESSED LOGICAL UNIT, or INVALID FIELD.

6.16 LOGICAL UNIT RESET SMS

The LOGICAL UNIT RESET SMS is sent from an initiator to a target to abort all tasks for all initiators on the specified Logical Unit. If the initiator is using multiple logical paths to the target, the initiator shall issue a LOGICAL UNIT RESET SMS over each different logical path used by tasks for the I_T nexus (see 8.5). The target executes a hard reset that shall perform the following:

a) abort all tasks on the specified Logical Unit;

b) clear all Auto Contingent Allegiance conditions on the specified Logical Unit;

c) release all SCSI device reservations on the specified Logical Unit;

d) returns any device operating modes to their appropriate initial conditions similar to those that may be found following a device power-on for the specified Logical Unit. The MODE SELECT conditions shall be restored to their last saved values if saved values have been established. Any MODE SELECT conditions that have no saved values established shall be returned to their default values.

The SSA-TL2 constructs of Initiator Tables, pending asynchronous alerts, the state of asynchronous event reporting and unprocessed SMSs shall be unaffected. There shall be no effect on other upper level protocols running in the node.

Before issuing the LOGICAL UNIT RESET SMS the initiator should terminate any related outbound data transfers to ensure that data is not sent to a non-existent nexus. If the target has issued any DATA REQUEST SMSs to other initiators then the target shall receive and flush all the requested data before aborting the corresponding tasks.

A SCSI STATUS SMS shall be sent for each task that completed before the SCSI RESPONSE SMS for the LOGICAL UNIT RESET was sent. Once the target has sent the SCSI RESPONSE SMS, no status shall be sent for any aborted task. However the target shall set the Unit Attention condition for all initiators. The ASC/ASCQ indicates POWER-ON OR RESET OCCURRED. The target always replies to the LOGICAL UNIT RESET SMS with an SCSI RESPONSE SMS. Upon receiving the SCSI RESPONSE SMS, the initiator s S3P layer shall remove all aborted commands from its Outstanding Commands Table (see D.3.1).

Table 17 defines the contents of the DATA field in a LOGICAL UNIT RESET SMS.

Table 19 - LOGICAL UNIT RESET SMS

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE (83h)							
1	S3P CODE (35h)							
2	TAG							
3	TAG							
4	RETURN PATH ID							
5	RETURN PATH ID							
6	RETURN PATH ID							
7	RETURN PATH ID							
8	LUN							

The TAG field is used to relate the SCSI RESPONSE SMS to the LOGICAL UNIT RESET SMS.

The RETURN PATH ID field identifies the logical path that shall be used by the target to return the associated SCSI RESPONSE SMS.

The LUN field specifies the Logical Unit. If an unsupported or unattached LUN is specified, an SCSI RESPONSE SMS is generated with a RETURN CODE of INVALID FIELD.

In the SCSI RESPONSE SMS, the RETURN CODE field shall be REQUESTED FUNCTION COMPLETED SUCCESSFULLY or INVALID FIELD.

6.17 SCSI RESPONSE SMS

The SCSI RESPONSE SMS is used to acknowledge the following SSA-S3P SMSs:

a) ABORT TASK,

b) ABORT TASK SET,

c) CLEAR TASK SET,

d) TARGET RESET,

e) CLEAR ACA CONDITION,

f) LOGICAL UNIT RESET,

g) SCSI COMMAND (only if certain fields are invalid).

Table 20 defines the contents of the DATA field in an SCSI RESPONSE SMS.

Table 20 - SCSI RESPONSE

Byte	Bit 7	6	5	4	3	2	1	Bit 0
0	SMS CODE (83h)							
1	S3P CODE (03h)							
2	TAG							
3	TAG							
4	RETURN CODE							

The TAG field is copied from the original SMS, and identifies the SMS that is being acknowledged.

Table 21 defines the RETURN CODE field indicates the result of the original SMS, and the values.

Table 21 - RETURN CODE values for the SCSI RESPONSE SMS

Value	Description
00h	REQUESTED FUNCTION WAS COMPLETED SUCCESSFULLY.
01h	TASK NOT FOUND
02h-03h	reserved
04h	OVERLAPPED SMSS ATTEMPTED
05h-1Fh	reserved
20h	NO ACA CONDITION EXISTED FOR THE ADDRESSED LOGICAL UNIT
21h-FEh	reserved
FFh	INVALID FIELD

7 Transport related items

7.1 Spindle synchronization

7.1.1 SYNC character

SSA defines the 8B/10B character K28.0 as a SYNC character for rotating media as spindle sync, that may be interleaved within frames on the link subject to the rules for User Defined characters in SSA-TL2.

One node in the Web should be nominated to originate the SYNC characters at the nominal rotation rate of the devices. For example, this node may be an array controller or a disk drive. The other devices decode the SYNC characters and may use the event to synchronize their spindle servos. The event replaces the Master sync pulse that was provided on a separate cable for the parallel SCSI bus. If the originating node fails, a backup node may be nominated to replace it.

Propagation of User Defined characters is controlled by the EUDC bit of the CONFIGURE PORT SMS (see SSA-TL2). Dual-port and switch nodes should normally be programmed to propagate User Defined characters received on one port to the other port(s). However, to avoid indefinite circulation, one node in each Cyclic Path should be programmed to block propagation of User Defined characters.

The origination and processing of SYNC characters is controlled by the RPL field in mode page 04h of the SCSI MODE SELECT command.

7.1.2 Option block usage of the SYNC pin

SSA-PH defines an options connector that includes a SYNC pin. This SYNC pin provides the same function as the SYNC character over the SSA links, but allows synchronization across SSA Webs.

NOTE 2 - The device should provide a way to block generation and receipt of any signal on the SYNC pin for customers who use the SYNC character on the SSA links and use a backplane or cable structure that connects the SYNC pin.

7.2 Unit attention flag in the Initiator Table

SSA-S3P adds an additional flag, UNIT ATTENTION flag, to each Initiator Table entry (see 8.6).

7.3 Effects of reset conditions

SSA-TL2 defines the following types of reset: Link Reset, Total Reset, Absolute Reset, and Power On Reset.

Link Reset has no affect on the SCSI constructs.

In response to a Absolute Reset or Power On Reset, the S3P layer is initialized and shall clear SCSI constructs (equivalent to an SCSI-3 Hard reset condition). In response to the notification of a Total Reset or Local Reset , the S3P layer is initialized and shall clear SCSI constructs (equivalent to an SCSI-3 Hard reset condition). In addition to the normal clearing of all tasks, reservations and operating conditions shall be reset to their last saved states.

Neither SSA-TL2 nor SSA-S3P provide a global reset equivalent to the RST signal in the parallel SCSI bus.

8 SSA-S3P changes from parallel SCSI-3

8.1 Untagged queuing is emulated

Since SSA is frame multiplexed, a tag or channel shall be associated with each frame to indicate the proper context. For this reason, Untagged commands are not supported by SSA-S3P. Untagged command queuing may be emulated by the initiator having only one outstanding command per LUN, re-use the same tag, and use the simple queue type. This is enforced by not allowing the Untagged queue type in the QUEUE CNTL field in the SCSI COMMAND SMS of SSA-S3P.

8.2 Contingent Allegiance vs. Auto Contingent Allegiance.

The concept of Contingent Allegiance requires an interlocked interface to work. In Contingent Allegiance, a CHECK CONDITION Status causes the queue to be held. The next command will execute, free the hold on the queue, and clear the sense data. In an interlocked interface like parallel SCSI, the next command arrives after the initiator sees the CHECK CONDITION Status. However, in a non-interlocked serial interface like SSA, the next command may be in the pipeline and received at the target before the initiator even receives the CHECK CONDITION Status. This causes the Contingent Allegiance condition to be reset, the queue freed and sense data lost before the initiator even knew a problem existed.

The answer to this problem is the SCSI-3 concept of Auto Contingent Allegiance. In Auto Contingent Allegiance, the queue is held following a CHECK CONDITION STATUS until specifically cleared by a CLEAR ACA task. Pipelined commands are rejected with an ACA ACTIVE status. More complex error recovery may be accomplished by the use of as many commands as needed using the ACA queue type.

Since Contingent Allegiance will not work properly on the SSA interface, the SSA target always uses Auto Contingent Allegiance. However, SSA-S3P needs to support Contingent Allegiance as per SAM. For this reason the initiator SSA-S3P protocol layer uses Auto Contingent Allegiance over the transport layer and emulates Contingent Allegiance to the Application Client as shown in the following sub clauses without the pitfalls of Contingent Allegiance.

8.2.1 Target supports Auto Contingent Allegiance only

If the Application Client running above the SSA-S3P initiator protocol layer requests Contingent Allegiance operation (as defined by the NACA bit being cleared), then the SSA-S3P initiator protocol layer will emulate Contingent Allegiance (see 8.2.2).

The target ignores the NACA bit. The target always assumes the NACA bit is set, regardless of the value of bit 2 of the CONTROL byte in the CDB field. The target will handle Auto Contingent Allegiance as it is described in SAM.

8.2.2 Initiator support of Contingent Allegiance

Contingent Allegiance is supported by making the following changes in the initiator SSA-S3P protocol layer.

a) Maintain a Contingent Allegiance Table (see Table 22).

b) When an SMS RECEIVED transport service is received with a valid SCSI STATUS SMS and a STATUS value of CHECK CONDITION STATUS, the DESTINATION ID and LUN field values from the Outstanding Commands Table entry associated with the SCSI STATUS SMS TAG field is entered in the table. The queue within the SSA-S3P initiator is held.

c) When an SMS RECEIVED transport service is received with a valid SCSI STATUS SMS and a STATUS field value of ACA ACTIVE, then the command is returned to the initiator s queue as if it had not be issued, in such a way that the command will be issued in the order originally intended.

d) When a Send SCSI Command protocol service request is received and the Contingent Allegiance Table has an entry corresponding the corresponding Destination ID and LUN, then the following actions occur in order.

e) Send the SCSI command normally except the QUEUE CNTL field shall have a value of AUTO CONTINGENT ALLEGIANCE. The Tag parameter of the SCSI Command is entered in the Contingent Allegiance Table in the entry appropriate for that Destination ID.

f) Hold any additional Send SCSI Command protocol service requests when an entry exists in the Contingent Allegiance table until the table entry associated with that Destination ID is empty.

g) When the SCSI STATUS SMS returns with the TAG value in the Contingent Allegiance Table, generate a CLEAR ACA SMS as described in 7.15, and when the associated SCSI RESPONSE SMS returns, clear the entry in the Contingent Allegiance Table. The hold on the SSA-S3P initiator queue shall be released, and any queued commands sent.

Table 22 - Contingent Allegiance Table entry format

Field	Description
TAG	The Tag value of the SCSI command from the Task Identifier parameter.
DESTINATION ID	The 8 byte unique ID of the target
LUN	The Logical Unit number from the Task Identifier parameter.

8.2.3 Initiator support of Auto Contingent Allegiance

The initiator support of Auto Contingent Allegiance shall follow the requirements of the SAM specification.

8.3 Confirmed status

The Confirmed Status feature of SSA-S3P is enabled with the CONFIRM bit in the SCSI COMMAND SMS, and is therefore settable on a command by command basis. When the confirmed Status feature is enabled, it allows three additional features of SSA-S3P:

a) Confirm Status process;

b) Restart Command process;

c) Redirect Command process.

8.3.1 Confirm Status process

The Confirm Status process provides an interlock between the target and the initiator, and is enabled for a command when a SCSI COMMAND SMS is sent with the CONFIRM bit set. When the initiator receives the SCSI STATUS SMS associated with a command using the Confirm Status process, the initiator will generate a CONFIRM STATUS SMS with the COMPLETE bit set. The receipt of the CONFIRM STATUS SMS by the target with the COMPLETE bit set confirms that the initiator has received the SCSI STATUS SMS sent by the target. The target is now free to discard the status information and perform the next task. If no CONFIRM STATUS SMS is received within 5 seconds of sending the SCSI STATUS SMS, the target shall generate an ASYNC ALERT SMS with an ALERT CODE of UNCONFIRMED STATUS. The initiators in the web shall not consider the command complete until 10 seconds after the most recent CONFIRM STATUS SMS for the command has been sent.

This allows for an opportunity for the target to notify the initiators of a lost CONFIRM STATUS SMS. If a initiator receives a MASTER ALERT indicating UNCONFIRMED STATUS, it will use the FRAME DATA field to discover the TAG and RETURN PATH ID fields of the unconfirmed command. If the initiator indicated has already sent a CONFIRM STATUS SMS, then the initiator re-sends the CONFIRM STATUS SMS. If the Initiator has not completed the command, then the Restart Command process is invoked.

8.3.2 Restart Command process

The Restart Command process provides a method of the initiator instructing the active task to be restarted from a particular point. The receipt of the CONFIRM STATUS SMS by the target with the COMPLETE bit cleared instructs the target to restart the active task from the point specified in the BYTE OFFSET field. Typically, this is used when a physical path has been lost and the logical path is being associated to a new physical path. In the case of a write data transfer from the initiator to the target, the target shall send a DATA REQUEST SMS to the initiator using the BYTE OFFSET and NEW RETURN PATH ID fields provided in the CONFIRM STATUS SMS. The initiator will send the data and confirm the status when received from the target. In the case of a read data transfer from the target to the initiator, the target shall send a DATA READY SMS to the initiator using the BYTE OFFSET and NEW RETURN PATH ID fields from the CONFIRM STATUS SMS. The Initiator shall send a DATA REPLY SMS to begin the data transfer. Following the receipt of the status, the initiator shall send the CONFIRM STATUS SMS with the COMPLETE bit set to indicate the receipt of the status. Since the old path used by the command may not be flushed of old data and SMSs, data and SMSs for the command prior to it being restarted shall be discarded.

8.3.3 Redirect Command process

The Redirect Command process provides a method of the initiator instructing the active or queued task to use another path. Typically, redirection may occur for the following reasons:

a) a path is lost and queued commands (not yet begun data transfer) need to be redirected to an alternate path,

b) a physical path has not been lost but, for load balancing and efficiency reasons, the logical path is being associated to a new physical path.

The receipt of the CONFIRM STATUS SMS by the target with the COMPLETE bit cleared instructs the target to redirect the active or queued task using the path specified in the NEW RETURN PATH ID field. The protocol follows that of the restart command.

8.4 Concurrent data transfers from a single target

SSA allows a logical unit to have several data transfers in progress at one time. For example, a media read may be transferring over one SSA port, while a cached read (or write prefetch) is transferring over another.

When a logical unit generates a Auto Contingent Allegiance condition, any tasks within the Auto Contingent Allegiance domain shall be suspended. When suspending an task that involves a concurrent data transfer, the data transfer burst may complete, but the target shall not initiate a new data burst (DATA READY or DATA REQUEST), nor complete the task (SCSI STATUS SMS) until the Auto Contingent Allegiance has been cleared.

8.5 Multiple logical paths

As seen by S3P, a logical path is a conduit for the ordered delivery of SMS and data frames between an initiator-target pair (an I_T nexus or I_T_L nexus). For a specific I_T nexus or I_T_L nexus, a logical path is uniquely identified by the RETURN PATH ID field in the SCSI COMMAND and task management SMSs. Ordered delivery means that for a given logical path, data frames and SMSs are received in the same order they were sent. SSA allows an I_T nexus or I_T_L nexus to be connected through multiple logical paths but does not guarantee that the order of frame and SMS delivery is preserved between them. As a result, the following conditions apply to the use of multiple logical paths between an I_T nexus or I_T_L nexus:

a) for a particular task, all SMSs and data frames shall use the same logical path;

b) different tasks for the same I_T nexus or I_T_L nexus may use different logical paths. Since delivery order is not guaranteed between logical paths, it is possible for the arrival order of frames sent along different logical paths to vary from the order that such frames were sent;

c) to insure that all in-flight commands are aborted, the TARGET RESET, CLEAR TASK SET, ABORT TASK SET, ABORT TASK, and LOGICAL UNIT RESET SMSs shall be sent over each different logical path that was used to start tasks.

8.6 Unit attention

An SSA Web may contain multiple initiators. During configuration every node receives a QUERY NODE SMS from each initiator. If an initiator intends to use alternate logical paths to the same target then it issues a QUERY NODE SMS over each logical path. A target node uses the information in QUERY NODE SMS to build an entry in its Initiator Table. The entry contains the Return Path and port for that initiator (see SSA-TL2).

When a target generates a Unit Attention it sets an SSA-S3P specific UNIT ATTENTION flag in all appropriate entries of the Initiator Table (only one flag for all logical paths to a given initiator). If the Unit Attention was generated by an SMS, then the target does not set the UNIT ATTENTION flag for the associated initiator.

When a target receives an SCSI COMMAND SMS it locates the UNIT ATTENTION flag for the initiator by accessing the Initiator Table entry containing the RETURN PATH ID field value. If the UNIT ATTENTION flag is set, then the target clears the flag and terminates the SCSI COMMAND SMS by generating an SCSI STATUS SMS with a STATUS field value of CHECK CONDITION, and generates sense data. Thus the Unit Attention is presented once to each initiator.

8.7 Third party command support

Third party commands are supported in SSA-S3P by using the devices Unique ID as the 8 byte identifier.

RESERVE (10) and RELEASE(10) are supported by setting the LONGID bit and using the Unique ID as the identifiers in parameter list.

COPY, COPY AND VERIFY and COMPARE are supported by using the Unique ID as the identifier in the Segment Descriptor.

Only Configutors may manage third party commands.

8.8 Optional SCSI-3 features not supported in SSA-S3P

The following optional SCSI-3 features are not supported in SSA-S3P.

8.8.1 Commands with small device addresses

The third party options of the following commands are not supported due to the 3 bit address space, although the 10 byte CDB versions are supported:

a) RESERVE (6)

b) RELEASE (6)

NOTE 3 - The initiator S3P protocol layer may map third party RESERVE(6) and RELEASE(6) commands into RESERVE(10) and RELEASE(10) commands respectively to avoid the need to modify the Application Client.

8.8.2 Terminate Task message

SSA-S3P does not support the Terminate Task message, due to lack of implementation in today's parallel SCSI products and the testing difficulties.

8.9 Mode Page 02h parameter differences

Several parameters within the SCSI-3 mode page 02h Disconnect/Reconnect parameters page are not required by SSA due to the frame multiplexing nature of the link. The following parameters shall be ignored:

a) BUS INACTIVITY LIMIT;

b) DISCONNECT TIME LIMIT;

c) CONNECT TIME LIMIT;

d) MAXIMUM BURST SIZE;

e) DTDC;

f) FARD;

g) FARWRT;

h) FASTAT;

i) DIMM.

The target shall return the Page 02h parameters in the Mode Sense command that were issued with the Mode Select command, according to the rules regarding current and saved values. SSA-S3P does not specify what values the parameters shall be initialized to, nor does SSA-S3P specify the parameters that are changeable.

Annex A
(informative)
Parallel SCSI-3 message to SSA-S3P SMS conversion

A.1 SCSI message summary

Table A.1 defines the mapping (if any) between the functions of the SCSI-3 messages and SSA-S3P.

Table A.1 - Mapping from parallel SCSI-3 messages to SSA-S3P SMSs

SCSI-3 message	SSA-S3P SMS
No Operation	N/A
Simple Queue Tag	SCSI COMMAND(QUEUE CNTL=11b, TAG)
Ordered Queue Tag	SCSI COMMAND(QUEUE CNTL=10b, TAG)
Head of Queue Tag	SCSI COMMAND(QUEUE CNTL=01b, TAG)
ACA	SCSI COMMAND (QUEUE CNTL=00b, TAG)
Identify (Out)	SCSI COMMAND (LUN)
Task Complete	SCSI STATUS
Linked Command Complete	SCSI STATUS(LINK set, FLAG cleared)
Linked Command Complete with Flag	SCSI STATUS(LINK set, FLAG set)
Identify (In)	SCSI STATUS(TAG) DATA READY (TAG) DATA REQUEST(TAG) SCSI RESPONSE
Modify Data Pointer	DATA READY(BYTE OFFSET) DATA REQUEST(BYTE OFFSET)
Abort Task Set	ABORT TASK SET
Abort Task	ABORT TASK
Clear Task Set	CLEAR TASKS SET
Clear ACA	CLEAR ACA
Target Reset	TARGET RESET
Logical Unit Reset	LOGICAL UNIT RESET
Message Reject	SCSI RESPONSE ASYNC ALERT & MASTER ALERT
Disconnect	N/A
Save Data Pointer	N/A
Continue Task	N/A
Target Transfer Disable	N/A
Restore Pointers	N/A
Terminate Task	N/A
Initiator Detected Error	N/A
Message Parity Error	N/A
Synchronous Transfer Request	N/A
Wide Data Transfer Request	N/A
Ignore Wide Residue	N/A
N/A	DATA REPLY

A.2 Parallel SCSI-3 message to SSA-S3P SMS conversion discussion

A brief description of how to map each SCSI-3 message into an SSA-S3P SMS is as follows:

(00h) Task Complete maps to the SCSI STATUS SMS with the status byte residing in the STATUS field, the LINK bit cleared, and the FLAG bit cleared.

(01h, 00h) Modify Data Pointer message maps to the SSA DATA READY and DATA REQUEST SMSs.

(01h, 01h) Synchronous Data Transfer Request message is N/A since SSA is a frame interleaved serial bus that always transfers at a fixed speed.

(01h, 03h) Wide Data Transfer Request message is N/A since SSA is a serial interface, and does not operate in a wide mode.

(02h) Save Data Pointer message is N/A since a frame multiplexed protocol does an implicit disconnect after every frame.

(03h) Restore Data Pointer message is N/A since a frame multiplexed protocol does an implicit disconnect after every frame.

(04h) Disconnect message is N/A since a frame multiplexed protocol does an implicit disconnect after every frame.

(05h) Initiator Detected Error message is N/A as frame rejects are handled in the transport layer of SSA. The level of reset depends on the initiator's error recovery procedure.

(06h) Abort Task Set message maps directly to the SSA ABORT TASK SET SMS.

(07h) Message Reject is N/A as frame rejects are handled in the transport layer of SSA.

(08h) No Operation is N/A for SSA. When a node does not need to send any information, FLAG characters as sent over the SSA cable.

(09h) Message Parity Error message is N/A as frame CRC checks are handled in the transport layer of SSA.

(0Ah) Linked Command Complete message maps into the SCSI STATUS SMS with the status byte residing in the STATUS FIELD, the LINK bit set, and the FLAG bit cleared.

(0Bh) Linked Command Complete with Flag message maps into the SCSI STATUS SMS with the status byte residing in the STATUS field, the LINK bit set, and the FLAG bit set.

(0Ch) Target Reset message maps to the SSA TARGET RESET SMS.

(0Dh) Abort Task message maps to the SSA ABORT TASK SMS.

(0Eh) Clear Task Set message maps to the SSA CLEAR TASK SET SMS.

(11h) Terminate Task message is N/A since SSA does not support the Terminate Task message.

(12h) Continue Task message is N/A since SSA does not support the Continue Task or Target Transfer Disable messages.

(13h) Target Transfer Disable message is N/A since SSA does not support the Continue Task or Target Transfer Disable messages.

(16h) Clear ACA message maps to the SSA CLEAR ACA SMS.

(17h) Logical Unit Reset message maps to the SSA LOGICAL UNIT RESET SMS.

(20h, xxh) Simple Queue Tag message maps to the SCSI COMMAND SMS with a QUEUE TYPE field = 11b (Simple) and the queue tag value in the TAG field.

(21h, xxh) Head of Queue Tag message maps to the SCSI COMMAND SMS with a QUEUE TYPE field = 01b (Head) and the queue tag value in the TAG field.

(22h, xxh) Ordered Queue Tag message maps to the SCSI COMMAND SMS with a QUEUE TYPE field = 10b (Ordered) and the queue tag value in the TAG field.

(23h, xxh) Ignore Wide Residue message is N/A as a set of data frames may be any number of bytes, and the concept of ignoring data with a grosser granularity of one byte does not apply.

(24h, xxh) ACA Queue Tag (SCSI-3) message maps to the SCSI COMMAND SMS with a QUEUE TYPE field = 00b (ACA) and the queue tag value in the TAG field.

(80h-FFh) Identify message maps to the SCSI COMMAND with the LUN value indicated in the LUN field. The DISCPRIV bit is not applicable in SSA where each frame has an implicit disconnect.

Annex B
(informative)
Some differences between SSA-S3P and parallel SCSI-3

This annex explores some of the differences between SSA-S3P and parallel SCSI-3.

B.1 Non-interlocked

Since SSA is a frame multiplexed interface rather than an interlocked interface, it does not depend on the bus phases to indicate acceptance or completion of a message. The ACK character pair in SSA-TL2 only indicates that the remote port has correctly received the frame. This implies that some form of response is needed for each SMS to indicate completion (see 6.16 for more details).

B.1.1 Message reject handling

Since SSA-S3P is not an interlocked protocol, the Message Reject condition does not exist as it does today in the parallel environment. By sending an ACK character pair, the remote port indicates acceptance of the SMS. Acceptance of the frame, does not indicate successful parsing of the frame.

However, SSA-S3P does respond to some of the SMS reject cases with an ASYNC ALERT SMS. Refer to 6.2 for the proper handling of invalid SMSs such as UNKNOWN RETURN PATH OR RETURN PATH ID, or UNSUPPORTED SMS.

B.1.2 Bus phases do not exist

Since SSA is frame multiplexed, parallel SCSI-3 bus phases do not exist. Each frame may be considered to do an arbitration, selection/reselection, and disconnect phase (without the parallel SCSI-3 bus overhead). A data frame or SCSI COMMAND SMS frame is considered to have done a data transfer phase.

B.2 Performance features

B.2.1 Less overhead for arbitration, selection, disconnection, reselection

Since SSA is frame multiplexed, each frame does implicit arbitration, selection/reselection, and disconnection with 8-12 bytes of overhead per frame (400 ns to 600 ns when running on SSA-PH1).

B.2.2 Out of order data transfers

Out of order data transfers (zero latency reads or writes) reduce latency by allowing the target to begin transfer from the current rotational position. For a full track read, this may save an average of 1/2 a revolution or 33% over a target that does not support out of order transfers. Without out of order, on average 1/2 of a rotation is used to find the beginning, and one revolution to read or write the track. With out-of-order transfers, only one revolution is used (R/W last *N* %, then R/W first 100- *N* %).

In parallel SCSI, out of order data transfers were optional by using the Modify Data Pointers message. SSA-S3P makes the support mandatory for the target, and selectable by the initiator (OOT bit). The SCSI COMMAND SMS is used along with the data transfer SMSs to activate out of order data transfers.

B.2.3 Concurrent data transfers within a web

Since SSA is frame multiplexed with a low frame overhead and multiple channel capability, there is no penalty for transferring multiple data transfers simultaneously. This may be an advantage in a RAID application using a single SSA Web, because the data from each of the RAID components may be processed (i.e. reconstruction) as the data arrives. In other words, a parallel bus may transfer all the data from one target and then disconnect, allowing the next target to transfer a chunk of data, etc., before the data is verified. But with SSA, all RAID components may be transferring simultaneously in an interleaved fashion, allowing the data to be verified as it arrives, thereby reducing latency.

B.2.4 Minimal initiator/target exchanges

Initiator/target exchanges may be minimized by grouping the parallel SCSI-3 functions of arbitration, message out (i.e. Identify message, Queue Tag message), and command phase. into SSA-S3P SCSI COMMAND SMS. The SCSI COMMAND SMS has additional function of being able to define options not available in SCSI-3 such as the DDRM and OOT bits on a command by command basis.

B.3 Availability features

SSA has high availability by nature of its rich topologies and multiple logical paths. See 8.5 for information on multiple logical paths.

B.4 Addressing

SSA greatly enhances the maximum number of addresses on an SSA Web. This requires some changes to parallel SCSI-3.

B.4.1 Additional addressing

SCSI-3 allows addressing of 8 (8 bit SCSI) or 16 (16 bit SCSI) devices on an SCSI parallel bus. The four byte address field of SSA-TL2 supports approximately 250 million devices with the use of switches. However, the two byte tag value limits the number of nodes to less than 64K, since a tag is needed for a possible ASYNC ALERT SMS from each node.

B.4.2 Larger LUN field

SSA-S3P defines a 256 Logical Units (LUN) within a target, as opposed to parallel SCSI-3, that allows 32 LUNs.

Annex C
(informative)
Protocol services model

This annex demonstrates how the SAM Services model may map onto the SSA-S3P protocol. This annex is for informational purposes only, and does not place any requirements on the implementation, other than those requirements already specified in this standard. Since the SAM Service model is a logical model, certain assumptions were made to map it onto a more detailed implementation (see D.1.2 and D.1.3).

C.1 Introduction

C.1.2 The SAM/protocol/transport layers

A partial list of the major responsibilities of the SAM, Protocol, and Transport layers is as follows.

The SAM layer is responsible for SCSI Command generation, queuing, and execution. SAM also handles sense data building, Contingent Allegiance or Auto Contingent Allegiance conditions, and command termination.

The Protocol layer provides the function to map the SAM Protocol services onto the SSA-TL2 transport layer. In addition, the Protocol layer maintains a data structure of all outstanding commands to allow for deadlock avoidance and error handling in case of illegal operations resulting in a deadlock situation. The protocol layer maintains control of path selection to maintain logical paths.

The Transport layer provides the function to map the Transport Services onto the SSA-PH1 physical environment. It provides procedures for configuration, error reporting, error handling, and configuration management. The Transport layer also converts the Protocol Unique ID into an RETURN PATH ID used by some SMSs by the use of the Initiator Table. The transport layer is also responsible for TAG field value generation for the protocol layer.

C.1.3 Guidance for implementation of the SAM services model

The SAM Services model is a logical model. In an attempt to show a more detailed implementation of the SAM Services model on SSA-S3P, certain assumptions were made as shown as follows:

a) Any implementation shall have a known parameter list rather than a set of optional parameters. Therefore, a test for null is used to determine if a parameter is included.

b) The SAM Services model (a logical model) assumes data is queued and sent with the service request or confirmation. In an actual implementation, pointers to host or device memory are sent and data is written to or read from memory below the SAM layer. Therefore the data structures are replaced with memory address pointers, and the names changed to add Address to the name.

C.1.4 Deadlock avoidance

A potential deadlock situation may occur since resources are limited in nodes, and SMSs are unsolicited. This situation does not occur with data transfers, since they are solicited with the use of data transfer SMSs.

Deadlock avoidance consists of two parts described as follows:

a) The number of pending non-SCSI COMMAND SMSs are limited and sufficient resources exist for them. Both the Transport and Protocol layers are responsible for this. The Protocol layer simply holds any service that may send a non-SCSI COMMAND SMS while any non-SCSI COMMANDS SMSs are outstanding.

b) The SCSI COMMAND SMS resource limitations are controlled by the use of Queue Full and the protocol layer s Outstanding Commands Table. When the initiator S3P protocol layer receives an SCSI STATUS SMS with STATUS field of either TASK SET FULL or BUSY, it generates a Command Complete Received protocol service response with the same status for all commands in its Outstanding Commands Table that were issued after the command responding with TASK SET FULL or BUSY status. Since the initiator generates the TASK SET FULL or BUSY status for those commands, the

target shall discard all SCSI Commands with the RESUME bit cleared, without issuing a SCSI STATUS SMSs or generating a Command Complete Received protocol service indication. The next SCSI Command to be sent has the RESUME bit set, causing the QUEUE FULL flag in the device to be cleared and the resumption of SCSI command processing.

C.1.5 Implementation specific options

The SSA protocol and transport services have several implementation specific features as described below.

a) The transport layer is responsible for controlling hardware and allocating data transfer resources, including CHANNEL values. If sufficient resources exist to allocate the data channel when the command is received (Send Command protocol service) rather than Data Transfer time, then overhead may be reduced during command overhead (Refer to the DDRM and CHANNEL fields of the SCSI Command SMS). If this is implemented, then the TARGET FAST READ and FAST READ DATA COMPLETED Transport services are used in place of the TARGET READ, INITIATOR READ, and READ DATA COMPLETED Transport services.

b) Accepting data transfers that are out of order (non-contiguous) for a command may improve performance significantly (Refer to the OOT bit of the SCSI COMMAND SMS in 6.6).

c) If a single initiator node uses two different protocols, then Tag assignment needs to be at the transport layer to avoid confusion in the data transfer SMSs. This involves the use of the GET TAG and FREE TAG transport services at the beginning and end respectively of the SCSI Command protocol services (see Table C.3). The Outstanding Commands Table entry (see Table C.1) also needs an TAG field (from the GET TAG transport service) to associate the TAG value received by the SCSI STATUS SMS to the proper Tag parameter value for the application client. The same requirements exist for the Task Management functions (see Table C. 7) and Outstanding Tasks table entry (see Table C.2).

C.2 Conventions

This sub-clause describes how to map the various SAM objects into SSA-S3P fields, and describes some of the notations used.

C.2.2 Service naming conventions.

The term Protocol Services refers to those services between the SSA-S3P layer and the SAM layer, and in following SAM the name contains first letter capitalized words. The term Transport Services refers to those services between the SSA-TL2 transport layer and the SSA-S3P protocol layer, and in following SSA-TL2 and SSA-S3P conventions the name is all letters capitalized.

C.2.3 Notations

The services are indicated with indented text as illustrated below.

Service-name (Input parm1, [Optional input parm2] || Output parm1, [Optional Output parm2])

The service name precedes the parentheses. The parentheses enclose the parameter lists. Any parameter enclosed in square brackets ([...]) is considered optional. The double bar (||) separates the input parameters from the output parameters.

C.3 Protocol layer data structures

C.3.1 Outstanding Commands table

The initiator s protocol layer shall maintain an Outstanding Commands Table consisting of entries for all outstanding commands as shown in Table C.1. The target does not maintain this table. The sequence of entries corresponds to the order in that the commands were sent to the target. As seen by the initiator s S3P layer, a command is outstanding after the Send SCSI Command protocol service request is received from the application client until the Command Complete Received protocol service confirmation is sent to the application client or the command is aborted. All fields are obtained from the Send SCSI Command protocol service.

Table C.1 - Outstanding Commands Table entry format

Field	Description
DESTINATION NODE ID	The 8 byte unique ID of the target
SOURCE ID	The 8 byte unique ID of the initiator from the Task Identifier parameter.
TAG	The TAG value of the SCSI command from the Task Identifier parameter.
LUN	The Logical Unit number from the Task Identifier parameter.
RETURN PATH ID	The RETURN PATH ID for all initiator generated SMSs for this command
DATA-OUT BUFFER ADDRESS	The optional Buffer-Out address parameter
DATA-IN BUFFER ADDRESS	The optional Buffer In address parameter
COMMAND BYTE COUNT	The optional Command Byte Count parameter
CDB	A copy of the Command Descriptor Block

C.3.2 Outstanding Tasks table

The initiator s protocol layer shall maintain an Outstanding Tasks Table consisting of entries for all outstanding tasks as shown in Table C.2. The target does not maintain this table. The sequence of entries corresponds to the order in that the tasks were sent to the target. As seen by the initiator s S3P layer, a task is outstanding after the Send Task Management Request protocol service is received from the application client until the Received Function-Executed protocol service is sent to the application client. All fields are obtained from the Send Task Management Request protocol service.

Table C.2 - Outstanding tasks Table entry format

Field	Description
DESTINATION NODE ID	The 8 byte unique ID of the target
SOURCE ID	The 8 byte unique ID of the initiator from the Task Identifier parameter.
TAG	The TAG value of the SCSI command from the Task Identifier parameter.
LUN	The Logical Unit number from the Task Identifier parameter.
RETURN PATH ID	The RETURN PATH ID for all initiator generated SMSs for this command
TASK FUNCTION	The S3P CODE field of SMS invoking the Task Management request

C.3.2 Queue Full flag

A single bit QUEUE FULL flag for the entire data structure is maintained for deadlock avoidance.

For the target protocol layer, the QUEUE FULL flag is processed as follows:

a) The QUEUE FULL flag in the target is set when it was previously cleared and a Send Command Complete protocol service response is received with a Status parameter value of TASK SET FULL. The Send Command Complete protocol service response is processed normally.

b) Any Send Command Complete protocol service response received with a Status parameter value of TASK SET FULL while the QUEUE FULL flag is set shall be ignored.

c) Any SMS RECEIVED transport service is invoked with a valid SCSI COMMAND SMS with a RESUME bit cleared while the QUEUE FULL flag is set shall be ignored (the Return Code parameter value is FUNCTION SUCCESSFUL, but no action is taken).

d) The QUEUE FULL flag is cleared and the command is processed when an SMS RECEIVED transport service is invoked with a valid SCSI COMMAND SMS with a RESUME bit set. The SMS RECEIVED transport service executed normally.

For the initiator protocol layer, the QUEUE FULL flag is processed as follows:

a) The QUEUE FULL flag in the initiator is set regardless of its previous state, and an SMS RECEIVED transport service is invoked with a valid SCSI STATUS SMS with a STATUS field value of TASK SET FULL. A Command Complete Received protocol service confirmation is then generated. The Outstanding Commands Table is scanned for any commands issued after the command returning TASK SET FULL status, and a Command Complete Received protocol service confirmation is generated and the entry cleared as if an SCSI STATUS was received with a TASK SET FULL status.

b) The QUEUE FULL flag is cleared, when it was previously set and a Send SCSI Command protocol service request is received. The resulting SCSI COMMAND SMS STRUCTURE has the RESUME bit set (see 6.6).

C.3.3 Object definitions

The Destination ID is the 64 bit Unique ID of the Destination.

The initiator Identifier is the device s 64 bit Unique ID.

The Logical Unit Number is a 8 bit value that resides in the LUN field (byte 8) of those SMSs that contain a Logical Unit Number (SCSI COMMAND, ABORT, CLEAR QUEUE, and CLEAR ACA).

The Source ID is the 64 bit Unique ID of the Source.

The Tag is a two byte value that resides in the TAG field (bytes 2 and 3) of all SSA-S3P SMSs.

The Target Identifier is the device s 64 bit Unique ID. The Unique ID is globally unique, and is the value is the same to each device in the SSA Web.

The Task Attribute is a two bit value that resides in the QUEUE CNTL field of the SCSI COMMAND SMS.

C.4 SCSI command protocol services

Table C.3 shows the typical services activity to execute a command that does not involve data transfer, where ==> and <== indicate direction.

Table C.3 - Command execution services activity

Initiator			Cable	Target		
SAM	Protocol	Transport	Cable	Transport	Protocol	SAM
Protocol Services	Transport Services		Cable	Transport Services	Protocol Services	
==>Send SCSI Command	==>GET LOGICAL PATH ==>SEND SMS		==> SCSI COMMAND SMS	==> SMS RECEIVED <== GET UNIQUE ID	==> SCSI Command Received	
					(execute command)	
<== Command Complete Received	<==SMS RECEIVED		<== SCSI STATUS SMS	<== SEND SMS	<== Send Command Complete	

C.4.1 Send SCSI Command protocol service request

Send SCSI Command (Destination ID, Source ID, Tag, LUN, CDB, [Task Attribute], [Data-Out Buffer Address], [Data-In Buffer Address], [Command Byte Count], [Autosense Request] ||)

Upon receipt of the Send SCSI Command protocol Service request from SAM, the protocol layer builds an SCSI COMMAND SMS and instructs the transport layer to transmit it.

The SSA-S3P protocol builds an SCSI COMMAND SMS into a data structure entitled SMS STRUCTURE as follows:

a) The SMS CODE and S3P CODE fields are set to 83h and 10h respectively.

b) The Tag and LUN parameters go in the TAG and LUN fields respectively.

c) The RETURN PATH ID field is the Return path id parameter from the invocation of the GET LOGICAL PATH transport service (see E.2.1).

d) The CDB parameter goes in the CDB field.

e) The Task Attribute parameter goes in the QUEUE CNTL field (the SIMPLE QUEUE value is used if the Task Attribute parameter is null).

f) The use of the DDRM, CHANNEL, and OOT fields is implementation dependent.

g) The use of the RESUME bit depends on the status of the QUEUE FULL flag condition (see D.3.2).

The Autosense parameter is ignored in the SSA-S3P protocol.

Editor s note: Keep the Autosense parameter available to decide if the sense data should be held for a potential REQUEST SENSE command (the target always has Autosense active).

The SSA-S3P protocol then invokes the transport layer service of SEND SMS as follows to send the SCSI COMMAND SMS.

SEND SMS (Control, Destination ID, Buffer Address, Length || Return Code)

a) The control parameter is always 00h.

b) The Destination ID parameter is the Destination ID from the Send SCSI Command protocol service.

c) The Buffer Address parameter is the location of the SMS STRUCTURE data structure.

d) The Length parameter is the length of the SMS STRUCTURE data structure.

If the Return Code parameter indicates FUNCTION FAILED or INVALID PARAMETER, then the protocol layer generates a Command Complete Received protocol service confirmation with a Service Response parameter value of SERVICE DELIVERY OR TARGET FAILURE.

C.4.2 SCSI Command Received protocol service indication

SCSI Command Received (Source ID, LUN, Tag, Task Attribute, CDB ||)

When the target s transport layer receives an SMS, it generates an SMS RECEIVED transport service indication. If the Protocol layer validates it as an SCSI Command SMS, then an SCSI Command Received Protocol service indication is generated as follows:

a) The Source ID parameter is obtained from the Unique ID parameter returned by the invocation of the GET UNIQUE ID (Return path id || Unique ID, Return Code) transport service. The Return path id parameter is taken from the RETURN PATH ID field of the SCSI COMMAND SMS

b) The TAG field of the SCSI COMMAND SMS is returned as the Tag parameter.

c) The LUN field of the SCSI COMMAND SMS is returned as the LUN parameter.

d) The QUEUE CNTL field of the SCSI COMMAND SMS is returned as the Task Attribute parameter.

e) The CDB field of the SCSI COMMAND SMS is returned as the CDB parameter.

C.4.3 Send Command Complete protocol service response

Send Command Complete (Destination ID, Source ID, Tag, LUN, Status, Service Response ||)

Upon receipt of the Send Command Compete protocol service from the device server, the protocol performs the following actions.

The SSA-S3P protocol builds an SCSI STATUS SMS into a data structure entitled SMS STRUCTURE as follows:

a) The SMS CODE and S3P CODE fields are set to 83h and 10h respectively.

b) The Tag parameter goes into the TAG field.

c) The Status parameter goes into the STATUS field.

d) The Service Response parameter causes the LINK and FLAG bits to be set as defined in Table C.4.

Table C.4 - Service Response parameter and its impact on flag and link bits

Service Response parameter	FLAG bit	LINK bit
Task Complete	0b	0b
Linked Command Complete	0b	1b
Linked Command Complete (with flag)	1b	1b
Service Delivery of Target Failure	0b	0b

The SSA-S3P protocol then invokes the transport layer service of SEND SMS as follows to send the SCSI STATUS SMS.

SEND SMS (Control, Destination ID, Buffer Address, Length || Return Code)

a) The control field is always 00h.
b) The Destination ID parameter of the Send Command Complete protocol service passed as the Destination ID parameter.
c) The Buffer Address parameter is the location of the SMS STRUCTURE data structure.
d) The Length parameter is the length of the SMS STRUCTURE data structure.

C.4.4 Command Complete Received protocol service confirmation

Command Complete Received (Destination ID, Source ID, Tag, LUN, [Data-In Buffer Address], Status, Service Response ||)

When the initiator s transport layer generates an SMS RECEIVED transport service indication, and the SMS is validated by the protocol layer as an SCSI STATUS SMS, then a Command Complete Received protocol service indication is generated as follows:

a) The TAG field of the SCSI STATUS SMS is returned as the Tag parameter.
b) The Destination ID parameter is returned from the DESTINATION NODE ID field of the Outstanding Commands Table entry identified by the Tag parameter.
c) The Source ID parameter is the initiators Unique ID.
d) The LUN parameter is returned from the LUN field of the Outstanding Commands Table entry identified by the Tag parameter.
e) The Data-In Buffer Address parameter is returned from the DATA-IN BUFFER ADDRESS field of the Outstanding Commands Table entry identified by the Tag parameter.
f) The STATUS field is returned as the Status parameter.
g) The Service Response parameter shall be returned with TASK COMPLETE.

If any of the following transport services returns a FUNCTION FAILED or INVALID PARAMETER Return Code parameter, then the protocol generates a Command Complete Received protocol service indication.

a) SEND SMS associated with the Send SCSI Command protocol service request.
b) INITIATOR READ parameters are invalid.
c) INITIATOR WRITE parameters are invalid.

In any of the previous cases, then a Command Complete Received protocol service indication is generated as follows:

a) The Tag parameter of the associated protocol service request (Send SCSI Command Protocol) or transport service (INITIATOR READ or INITIATOR WRITE) is returned as the Tag parameter.
b) The Destination ID parameter is returned from the DESTINATION NODE ID field of the Outstanding Commands Table entry identified by the Tag parameter.
c) The Source ID parameter is the initiators Unique ID.
d) The LUN parameter is returned from the LUN field of the Outstanding Commands Table entry identified by the Tag parameter.
e) The Data-In Buffer parameter is not used.
f) The Status parameter is not used.

g) The Service Response parameter shall be returned with DELIVERY OR TARGET FAILURE.

C.5 Data transfer protocol services

Table C.5 shows the typical services activity to transmit Data-In, where ==> or <== indicates direction.

Table C.5 - Data-In services activity

Initiator			Cable	Target		
SAM	Protocol	Transport	Cable	Transport	Protocol	SAM
Protocol Services	Transport Services		Cable	Transport Services	Protocol Services	
	Initiator Read <==		DATA READY <== ==> DATA REPLY data<==	Target Read <== (send data)	Send Data-In <==	
				Read Data Completed ==>	Data Sent ===>	

Table C.6 shows the typical services activity to transmit Data-Out, where ==> or <== indicates direction.

Table C.6 - Data-Out services activity

Initiator			Cable	Target		
SAM	Protocol	Transport	Cable	Transport	Protocol	SAM
Protocol Services	Transport Services		Cable	Transport Services	Protocol Services	
	Initiator Write <== (send data)		DATA REQUEST<== ==>data	Target Write <== (receive data)	Receive Data Out <==	
				Write Data Completed ==>	Data Out Received ==>	

C.5.1 Send Data-In protocol service request

Send Data-In (Destination ID, Source ID, Tag, LUN, Device Server Buffer, Application Client Buffer Offset, Request Byte Count ||)

Upon receipt of the Send Data-In protocol service request, the protocol invokes the following transport service.

TARGET READ (Destination ID, Tag, Buffer Address, Byte Offset, Byte Count, Start Count, Threshold Count || Return Code)

a) The Destination ID parameter of the Send Data In protocol service request becomes the Destination ID parameter of the TARGET READ protocol service.

b) The Tag parameter of the Send Data In protocol service request becomes the Tag parameter of the TARGET READ protocol service.

c) The Device Server Buffer parameter of the Send Data In protocol service request becomes the Buffer Address parameter of the TARGET READ protocol service.

d) The Application Client Buffer Offset parameter of the Send Data In protocol service request becomes the Byte Offset parameter of the TARGET READ protocol service.

e) The Request Byte Count parameter of the Send Data In protocol service request becomes the Byte Count parameter of the TARGET READ protocol service.

f) The use of the Start Length and Threshold Length parameters are implementation dependent. These parameters may be used when the target does not have Byte Count worth of data in its buffer at the time of invocation (but it does have Start Count), and intends to use the transport layer to throttle the data transfer. Refer to the Target Read service in the SSA-TL2 standard for more information on this function. Normal operation may set the Start Length parameter to the Byte Count parameter, and the Threshold Length parameter to zero.

If the Return Code parameter indicates FUNCTION FAILED, then the protocol layer generates a Command Complete Received protocol service confirmation with a Service Response parameter value of SERVICE DELIVERY OR TARGET FAILURE.

If the Return Code parameter indicates FUNCTION SUCCESSFUL, then the invocation of the TARGET READ transport service causes the Initiator protocol to receive an INITIATOR READ transport service request, by sending one or more DATA READY SMS(s).

C.5.2 Data Sent protocol service confirmation

Data Sent (Destination ID, Source ID, Tag, LUN ||)

When the data transfer initiated by the Send Data-In protocol service request completes, the transport layer informs the protocol layer with a READ DATA COMPLETE transport service confirmation.

READ DATA COMPLETED (|| Tag, LUN, Destination ID, Return Code)

Upon receipt of the READ DATA COMPLETED transport service, the protocol layer generates a Data Sent protocol service confirmation as follows:

Data Sent (Destination ID, Source ID, Tag, LUN ||)

a) The READ DATA COMPLETED Destination ID parameter is copied into the Destination ID parameter.

b) The target s Unique ID is returned as the Source ID parameter.

c) The READ DATA COMPLETED Tag parameter is copied into the Tag parameter.

d) The READ DATA COMPLETED LUN parameter is copied into the LUN parameter.

C.5.3 Receive Data-Out protocol service request

Receive Data-Out (Destination ID, Source ID, Tag, LUN, Application Client Buffer Offset, Request Byte Count, Device Server Buffer ||)

Upon receipt of the Receive Data-Out protocol service request, the protocol invokes the following transport service.

Target Write (Destination ID, Tag, Buffer Address, Byte Offset, Byte Count || Return Code)

a) The Receive Data-Out Destination ID parameter becomes the Destination ID parameter.

b) The Receive Data-Out Tag parameter becomes the Tag parameter.

c) The Receive Data-Out Device Server Buffer parameter becomes the Buffer Address parameter,

d) The Receive Data-Out Application Client Buffer Offset parameter becomes the Byte Offset parameter,

e) The Receive Data-Out Request Byte Count parameter becomes the Byte Count parameter,

If the Return Code parameter indicates FUNCTION FAILED, then the protocol layer generates a Send Command Complete protocol service request with a Service Response parameter value of SERVICE DELIVERY OR TARGET FAILURE.

The invocation of the Target Write transport service causes the Initiator protocol to receive an Initiator Write transport service request.

C.5.4 Data-Out Received protocol service confirmation

Data-Out Received (Destination ID, Source ID, Tag, LUN ||)

When the data transfer initiated by the Send Data-Out protocol service request completes, the transport layer informs the protocol layer with a WRITE DATA COMPLETE transport service confirmation.

WRITE DATA COMPLETED (|| Tag, LUN, Destination ID, Return Code)

Upon receipt of the READ DATA COMPLETED transport service, the protocol layer generates a Data-Out Received protocol service confirmation as follows:

Data-Out Received (Destination ID, Source ID, Tag, LUN ||)

a) The WRITE DATA COMPLETED Destination ID parameter is copied into the Destination ID parameter.
b) The target s Unique ID is returned as the Source ID parameter.
c) The WRITE DATA COMPLETED Tag parameter is copied into the Tag parameter.
d) The WRITE DATA COMPLETED LUN parameter is copied into the LUN parameter.

C.6 Task management functions

This sub-clause describes the common actions of all Task Management functions except for the Terminate Task function, that always returns a Service Response of FUNCTION REJECTED.

Upon receipt of the Task Management Function protocol Service request, the protocol layer builds an SMS structure and instructs the transport layer to transmit it. The type of SMS depends on the Task Management function, and is detailed in each Task Management sub-clause.

In all cases, the SSA-S3P protocol then invokes the transport layer service of SEND SMS as follows:

SEND SMS (Control, Destination ID, Buffer Address, Length || Return Code)

a) The Control parameter is always 00h
b) The Destination ID parameter is taken from the Send Task Management Request Destination ID parameter.
c) The Buffer Address parameter is the location of the SMS STRUCTURE data structure.
d) The Length parameter is the length of the SMS STRUCTURE data structure.

If the Return Code parameter indicates FUNCTION FAILED or INVALID FIELD, then the protocol layer Service Response value is SERVICE DELIVERY OR TARGET FAILURE. If the Return Code parameter indicates FUNCTION SUCCESSFUL, then the protocol layer Service Response returns a value of FUNCTION COMPLETE.

It is the responsibility of the initiator s S3P protocol layer to make sure that only one Task Management request for a given target is outstanding. Additional Task Management requests received from an application client while another Task Management request is active for an I_T nexus are held until the outstanding Task Management function completes.

C.6.1 Abort Task

Service Response = ABORT TASK (Destination ID, LUN, Tag ||)

The SSA-S3P protocol builds an ABORT TASK SMS into a data structure entitled SMS STRUCTURE as follows:

a) The SMS CODE and S3P CODE fields are set to 83h and 30h respectively.
b) The TAG field is set to 00h.
c) The RETURN PATH ID field is set to the Return path id parameter returned by GET LOGICAL PATH (see E.2.1).
d) The TAG2 field is set to the Tag parameter.

C.6.2 Abort Task Set

Service Response = ABORT TASK SET (Destination ID, LUN ||)

The SSA-S3P protocol builds an ABORT SMS into a data structure entitled SMS STRUCTURE as follows:

a) The SMS CODE and S3P CODE fields are set to 83h and 31h respectively.

b) The TAG field is set to 00h.

c) The RETURN PATH ID field is set to the Return path id parameter from the invocation of the GET LOGICAL PATH (see E.2.1).

d) The LUN field is set to the LUN parameter.

C.6.3 Clear ACA

Service Response = CLEAR ACA (Destination ID, LUN ||)

The SSA-S3P protocol builds a CLEAR ACA SMS into a data structure entitled SMS STRUCTURE as follows:

a) The SMS CODE and S3P CODE fields are set to 83h and 34h respectively.

b) The TAG field is set to 00h.

c) The Logical Unit Identifier parameter is passed in the LUN field.

C.6.4 Clear Task Set

Service Response = CLEAR TASK SET (Destination ID, LUN ||)

The SSA-S3P protocol builds a CLEAR QUEUE SMS into a data structure entitled SMS STRUCTURE as follows:

a) The SMS CODE and S3P CODE fields are set to 83h and 32h respectively.

b) The TAG field is set to 00h.

c) The RETURN PATH ID field is set to the Return path id parameter from the invocation of the GET LOGICAL PATH (see E.2.1).

d) The LUN field is set to the LUN parameter.

C.6.5 Target Reset

Service Response = TARGET RESET (Destination ID ||)

The SSA-S3P protocol builds a TARGET RESET SMS into a data structure entitled SMS STRUCTURE as follows:

a) The SMS CODE and S3P CODE fields are set to 83h and 33h respectively.

b) The TAG field is set to 00h.

c) The RETURN PATH ID field is set to the Return path id parameter from the invocation of the GET LOGICAL PATH (see E.2.1).

C.6.6 Logical Unit Reset

Service Response = LOGICAL UNIT RESET (Destination ID, LUN ||)

The SSA-S3P protocol builds a LOGICAL UNIT RESET SMS into a data structure entitled SMS STRUCTURE as follows:

a) The SMS CODE and S3P CODE fields are set to 83h and 35h respectively.

b) The TAG field is set to 00h.

c) The LUN field is set to the LUN parameter.

d) The RETURN PATH ID field is set to the Return path id parameter from the invocation of the GET LOGICAL PATH (see E.2.1).

C.6.7 Terminate Task

Service Response = TERMINATE TASK (Destination ID, LUN, Tag ||)

Implementation of the terminate Task protocol service is optional for a logical unit in SAM and is not supported in SSA-S3P. The Service Response is always unconditionally returned as FUNCTION REJECTED.

C.7 Task management protocol services

Table C. 7 - Task management execution services activity

SAM	Protocol	Transport	Cable	Transport	Protocol	SAM
Protocol Services		**Transport Services**	**Cable**	**Transport Services**		**Protocol Services**
==>Send Task Management Request		==>GET LOGICAL PATH ==>SEND SMS	==>Task Management SMS	==> SMS RECEIVED <== GET UNIQUE ID		==> Task Management Request Received
						(execute task)
<== Command Complete Received		<==SMS RECEIVED	<==SCSI RESPONSE SMS	<==SEND SMS		<== Task Management Function Executed

C.7.1 Send Task Management Request

Send Task Management Request (Object Address, Function Identifier ||)

Upon receipt of the Send Task Management Request protocol service, the protocol layer performs the actions defined in the appropriate Task Management Function sub-clause.

C.7.2 Task Management Request Received

Task Management Request Received (Object Address, Function Identifier ||)

Upon receipt of an SMS RECEIVED transport service indication and validation of SMS contents, the target s S3P layer generates a Task Management Request Received protocol service indication with the appropriate parameters.

C.7.3 Task Management Function Executed

Task Management Function Executed (Object Address, Service Response ||)

Upon receipt of a Task Management Function Executed protocol service, the protocol layer builds a partial SCSI RESPONSE SMS and instructs the transport layer to transmit it.

The SSA-S3P protocol builds an SCSI Response SMS into a data structure entitled SMS STRUCTURE as follows:

a) The SMS CODE and S3P CODE fields are set to 83h and 03h respectively.

b) The TAG field is set to 00h.

c) Table C.8 defines the return code field value to be used based on the value of Service Response.

Table C.8 - RETURN CODE field value based on service response

Service Response parameter value	RETURN CODE field value
FUNCTION REJECTED	INVALID PARAMETER
FUNCTION COMPLETE	FUNCTION SUCCESSFUL
SERVICE DELIVERY OR TARGET FAILURE	FUNCTION FAILED

The SSA-S3P protocol then invokes the transport layer service of SEND SMS as follows:

SEND SMS (Control, Destination ID, Buffer Address, Length || Return Code)

a) The control field is always 00h.
b) The Destination ID parameter is the Destination ID parameter of the Task Function.
c) The Buffer Address parameter is the location of the SMS STRUCTURE data structure.
d) The Length parameter is the length of the SMS STRUCTURE data structure.

C.7.4 Received Function-Executed

Received Function-Executed (Object Address, Service Response ||)

Upon receipt of an SMS RECEIVED transport service indication and validation of SMS contents, the initiator s S3P protocol layer generates a Received Function Executed protocol service confirmation with the appropriate parameters. The return code field maps into the Service Response parameter as defined in Table C.8.

Annex D
(informative)
Transport Service handling

This annex describes how the protocol layer handles Transport Services. Refer to the SSA-TL2 standard for more information on the transport layer services.

D.1 Transport services invoked by the transport layer

The following sub-clauses describe:

a) Transport Service requests directed to the Transport Layer from the S3P layer;
b) Transport Service indications received by the S3P layer from the Transport Layer;
c) Transport Service confirmations received by the S3P layer from the transport layer,
d) The responsibilities of the S3P layer.

D.1.1 Object definitions

The parameters for the Transport services are as follows (see the SSA-TL2 standard for a detailed description):

a) The Buffer Address parameter is a memory location where data is to be stored to or retrieved from.
b) The Byte Offset parameter is a 4 byte value indicating the offset of this data transfer based on the Buffer Address parameter.
c) The Byte Count parameter is a 4 byte value indicating the number of bytes for a data transfer. In the case of a read or write that transfers only a part of the data for an SCSI command, the byte count field only indicates the amount of data to be transferred by the request.
d) The Control parameter indicates the CONTROL field for or from the SMS frame as described in the SSA-TL2 standard.
e) The Length parameter is a 1 byte count of the number of bytes in the DATA field of the frame to be sent or received.
f) The Channel parameter is a 2 byte value indicating the initiator channel for the target to use as the channel component in the ADDRESS field of the read Data frame(s).
g) The Return Path ID parameter identifies a logical path and is used to set the value of the RETURN PATH ID field when required in the SMS.
h) The In Order parameter is a 1 bit flag that indicates that the data is written into the buffer space as received (In Order set to 0b), or written into the buffer space in order with the Buffer Address location containing the byte of data with an offset of zero (In order set to 1b). The In Order parameter allows the transport layer to use split data transfers even if the Protocol layer does not support splits.
i) The Replace4-7 parameter is a flag to indicate whether the transport layer replaces bytes 4 through 7 of the SMS with the Return Path ID associated with the Unique ID parameter.
j) The Return Code parameter is an indicator of the success or failure of the service to execute properly. Valid Return Code values are listed in SSA-TL2.

D.1.2 SMS RECEIVED

SMS RECEIVED (Control, Buffer Address, Length ||)

The SMS RECEIVED transport service input parameters are Control, Buffer Address and Length. There are no output parameters. It is initiated by the Transport layer when an SMS is received. The contents of the SMS are stored in the location designated by the Buffer Address parameter. The Length parameter specifies the length of the SMS payload.

In response to an SMS RECEIVED transport service indication, the protocol shall perform the SMS Validation process defined in 6.2. If the SMS is valid, it is processed as defined in the sub-clauses shown in Table D.1.

Table D.1 - SMS processing sub-clause references

SMS Received	sub-clause
SCSI COMMAND	D.4.2 SCSI Command received protocol service indication
SCSI STATUS	C.4.4 Command Complete Received protocol service confirmation
ABORT TASK	C.6.1 Abort Task
ABORT	C.6.2 Abort Task Set
CLEAR ACA	C.6.3 Clear ACA
CLEAR QUEUE	C.6.4 Clear Task Set
TARGET RESET	C.6.5 Target Reset
SCSI RESPONSE	C.7.4 Received Function-Executed

D.1.3 INITIATOR READ

INITIATOR READ (Tag, Byte Offset, Byte Count || Return path id, Buffer Address, Start Count, Threshold Count, Return Code)

When the initiator transport layer receives a DATA READY SMS, it generates the INITIATOR READ transport service indication. This causes the Initiator protocol layer to perform the following actions.

a) If the Command Byte Count or Buffer Address entries referenced by the Tag parameter in the OUTSTANDING COMMANDS data structure are null, then return FUNCTION FAILED in the Return Code parameter.

b) If the Byte Offset plus the Byte Count exceeds the Command Byte Count entry referenced by the Tag parameter in the OUTSTANDING COMMANDS data structure, then return FUNCTION FAILED in the Return Code parameter.

c) If neither of the conditions in a) nor b) occurred, then set up the data transfer to begin at the location pointed to by the Buffer Address entry in the OUTSTANDING COMMANDS data structure plus the Byte Offset parameter. The length of the transfer shall be the Byte Count parameter.

d) The Protocol layer returns the Initiator Read Buffer Address parameter from the Data-In Buffer Address parameter of the appropriate entry in the OUTSTANDING COMMANDS data structure.

e) The use of the Start Length and Threshold Length parameters are implementation dependent. These parameters may be used when the initiator does not have Byte Count worth of buffer space at the time of receipt of this service (but it does have Start Count), and intends to use the transport layer to throttle the data transfer. Refer to the Initiator Read service in the SSA-TL2 standard for more information on this function.

D.1.4 READ DATA COMPLETED

READ DATA COMPLETED (|| Tag, Unique ID, LUN, Return Code)

The receipt of a READ DATA COMPLETED transport service confirmation by the target s S3P layer results in the generation of a Data Sent protocol service confirmation (see D.5.2).

D.1.5 INITIATOR WRITE

INITIATOR WRITE (Tag, Byte Count, Byte Offset || Unique ID, Buffer Address, Return Code)

The invocation of the Target Write transport service causes the Initiator protocol to receive an Initiator Write transport service indication. This causes the Initiator protocol to perform the following functions.

a) If the Command Byte Count or Buffer Address entries referenced by the Tag parameter in the OUTSTANDING COMMANDS data structure are null, then return FUNCTION FAILED in the Return Code parameter.

b) If the Byte Offset plus the Byte Count exceeds the Command Byte Count entry referenced by the Tag parameter in the OUTSTANDING COMMANDS data structure, then return FUNCTION FAILED in the Return Code parameter.

c) The Protocol layer returns the Initiator Write Buffer Address parameter from the Data-Out Buffer Address parameter of the appropriate entry in the OUTSTANDING COMMANDS data structure.

d) If neither of the conditions in a) nor b) occurred, then set up the data transfer to begin at the location pointed to by the Initiator Write Buffer Address plus the Byte Offset parameter. The length of the transfer shall be the Byte Count parameter.

D.1.6 WRITE DATA COMPLETED

WRITE DATA COMPLETED (|| Tag, Unique ID, LUN, Return Code)

The receipt of a WRITE DATA COMPLETED transport service confirmation by the target s S3P layer results in the generation of a Data-Out Received protocol service confirmation (see D.5.4).

D.1.7 QUIESCE

QUIESCE (Source ID || Return Code)

The receipt of a Quiesce transport service indication causes the following actions to occur.

a) Generate a Abort Task Set protocol service indication for all Logical Units supported by the Target.

b) If any Service Response was SERVICE DELIVERY OR TARGET FAILURE, then return a FUNCTION FAILED Return Code value, otherwise return a FUNCTION SUCCESSFUL Return Code value.

D.2 Transport services invoked by the protocol layer

D.2.1 GET LOGICAL PATH

GET LOGICAL PATH (Destination ID, Logical Path Number ||Return Path ID, Return Code)

The GET LOGICAL PATH initiator transport service request is used during the construction of a protocol layer SMS to assign a Return Path ID for a given Logical Path Number. The input parameters are the Destination ID and Logical Path Number. The Output parameters are Return Path ID and Return Code.

The Destination ID parameter is the identifier for the Destination node.

The Logical Path Number is a number for a logical path. The same Return Path ID will always be returned for a given Logical Path Number.

The Return Path ID parameter is the Return Path ID to be used for that Logical Path. The same Return Path ID may be used for multiple Logical Path Numbers, but a given Logical Path Number will always return the same Return Path ID.

The Return Code parameter indicates the status of the GET LOGICAL PATH service request. A value of INVALID PARAMETER indicates that the Destination ID is not known.

D.2.2 GET TAG

GET TAG (Destination ID || Tag, Return Code)

The GET TAG initiator transport service request is used during the construction of a protocol layer SMS to assign a Tag value. The input parameter is the Destination ID. The Output parameters are Tag and Return Code. This service request does not need to exist and Tag assignment may be controlled within the protocol layer if only one protocol layer is running in the initiator at a time. The GET TAG service request needs to be implemented only in initiators that have multiple protocols running at any given time.

The Destination ID parameter is the identifier for the Destination node.

The Tag parameter is the value to be used for the TAG field, and is unique for all outstanding Tag values to that node. The Tag value is freed with a FREE TAG transport layer service request, and is only valid if the Return Code parameter indicates FUNCTION SUCCESSFUL.

The Return Code parameter indicates the status of the GET TAG service request. A value of FUNCTION FAILED indicates that no unique Tag value is available at this time.

D.2.3 FREE TAG

FREE TAG (Destination ID, Tag || Return Code)

The FREE TAG initiator transport service request is used during the construction of a protocol layer SMS to free a Tag value. The input parameters are the Destination ID and Tag. The Output parameter is Return Code. This service request does not need to exist and Tag assignment is controlled within the protocol layer if only one protocol layer is running in the initiator at a time. The FREE TAG service request needs to be implemented only in initiators that have multiple protocols running at any given time.

The Destination ID parameter is the identifier for the Destination node.

The Tag parameter is the value to be freed. The Tag value may be considered freed if the Return Code parameter indicates FUNCTION SUCCESSFUL.

The Return Code parameter indicates the status of the FREE TAG service request. A value of INVALID PARAMETER indicates that the Destination ID is not known, or the Tag value was not assigned.

D.2.4 GET UNIQUE ID

GET UNIQUE ID (Return Path ID || Unique ID, Return Code)

The GET UNIQUE ID transport service request is used to map the Return path id in a received SMS to a Source ID.

D.2.5 SEND SMS

SEND SMS (Control, Unique ID, Buffer Address, Length, Replace4-7 || Return Code)

The SEND SMS request is used to send SMS data structures.

D.2.6 TARGET READ

TARGET READ (Unique ID, Tag, Buffer Address, Byte Offset, Byte Count, Start Count, Threshold Count || Return Code)

The TARGET READ transport service request is used to initiate read data transfers initiated by the Send Data-In protocol service request.

D.2.7 TARGET FAST READ

TARGET FAST READ (Unique ID, Byte Count, Channel || Return Code)

The TARGET FAST READ request is only used by those implementations that support it (see D.1.5).

D.2.8 TARGET WRITE

TARGET WRITE (Unique ID, Tag, Buffer Address, Byte Offset, Byte Count || Return Code)

The TARGET WRITE transport service request is used to request write data transfers by the Receive Data-Out protocol service request.

D.2.9 REGISTER ULP

REGISTER ULP (Vendor Code, AER Mask || ULP Code, Return Code)

When the protocol layer is initialized, it invokes the REGISTER ULP service to the transport layer specifying its Vendor Code and AER Mask (see SSA-TL2). The ULP Code value returned is used in the ULP CODE field of the protocol SMSs. For SSA-S3P, the Vendor Code parameter is 83h, and the returned ULP Code parameter is 83h.

D.2.10 AER

AER (AER Code, Device ID, Alert Code, Frame Data ||)

When the protocol layer receives the AER service (see SSA-TL2), the protocol layer will perform the recovery specified in the following paragraphs.

An AER Code parameter value of LOCAL RESET PROCESSED causes the protocol layer to restart or re-issue any outstanding I/O processes. If the CONFIRM bit was set in the SCSI COMMAND SMS, the protocol may issue a CONFIRM STATUS SMS with a COMPLETE bit cleared, to verify that the device received the command, and to restart it if necessary. If the CONFIRM STATUS SMS generates an Asynchronous Alert with an alert code value of UNEXPECTED SMS, then the protocol layer re-issues the command. If the confirm bit was cleared then the protocol should issue an ABORT TASK SMS and reissue the command.

An AER Code parameter value of TOTAL RESET PROCESSED causes the protocol layer to restart or re-issue any outstanding I/O processes. If the CONFIRM bit was set in the SCSI COMMAND SMS, the protocol may issue a CONFIRM STATUS SMS with a COMPLETE bit cleared, to verify that the device received the command, and to restart it if necessary. If the CONFIRM STATUS SMS generates an Asynchronous Alert with an alert code value of UNEXPECTED SMS, then the protocol layer re-issues the command. If the confirm bit was cleared then the protocol should issue an ABORT TASK SMS and reissue the command.

An AER Code parameter value of NEW DEVICE ADDED clauses the protocol layer to add the device indicated by the Device ID parameter to its internal device data structures and alert the Application client as to the existence of the device.

An AER Code parameter value of DEVICE ACCESS LOST causes the protocol layer to remove the device ID indicated by the Device ID parameter from its internal device data structures and alert the Application client as to the removal of the device.

An AER Code parameter value of DEVICE PATH CHANGED causes the protocol layer to restart or re-issue any outstanding I/O processes. If the CONFIRM bit was set in the SCSI COMMAND SMS, the protocol may issue a CONFIRM STATUS SMS with a COMPLETE bit cleared, to verify that the device received the command, and to restart it if necessary. If the CONFIRM STATUS SMS generates an Asynchronous Alert with an alert code value of UNEXPECTED SMS, then the protocol layer re-issues the command. If the confirm bit was cleared then the protocol should issue an ABORT TASK SMS and reissue the command.

An AER Code value of INVALID SMS causes the protocol layer to perform implementation specific recovery procedures, if any, to reissue the SMS specified in the Frame Data parameter if sent by the protocol layer.

An AER Code value of ENVIRONMENTAL ERROR causes the protocol layer to perform implementation specific recovery procedures, if any, to handle the environmental error specified by the Alert Code parameter.

An AER Code value of WARNING causes the protocol layer to perform implementation specific recovery procedures, if any, to handle the warning specified by the Alert Code parameter.