To: T10 Technical Committee
From: Rob Elliott, HP (elliott@hp.com)
Date: 12 March 2005
Subject: 05-107r1 SAS-1.1 SAM-4 Overlapped tag handling

Revision history
Revision 0 (4 March 2005) First revision
Revision 1 (12 March 2005) Incorporated comments from March SAS WG and CAP WG.

Related documents
04-137r2 ST_T state machines (Mark Evans, Maxtor)(incorporated into sas1r05)
sas1r08 - Serial Attached SCSI 1.1 revision 8
sam4r01 - SCSI Architecture Model - 4 revision 1

Overview
If overlapped tag checking is supported by the target, the required behavior changed from SAS-1 to SAS-1.1 based on 04-137r2.

In both SAS-1 and SAS-1.1, COMMAND/COMMAND tag conflicts result in a a RESPONSE frame for the overlapped tag with CHECK CONDITION/ABORTED COMMAND/OVERLAPPED COMMANDS ATTEMPTED. The target then aborts all tasks for the I_T nexus which delivered the overlapped tag.

In SAS-1.0, the target replied to TASK frames with tag conflicts with a RESPONSE frame indicating INVALID FRAME. It was unclear on exactly how to handle all the TASK/COMMAND, COMMAND/TASK, and TASK/TASK combinations.

Currently in SAS-1.1, the target does not report the error immediately. It acts as if an I_T nexus loss had occurred, which includes aborting all tasks, and establishes a unit attention condition. The next command that arrives ends up getting CHECK CONDITION/UNIT ATTENTION/I_T NEXUS LOSS OCCURRED.

This is different than COMMAND handling, although real designs are likely to treat them the same. TASK conflicts should be handled more similarly to COMMAND conflicts: they should return a RESPONSE frame immediately indicating the error, abort all commands (and task management functions) from the I_T nexus, but not establish a unit attention condition.

Proposed handling:

a) COMMAND/COMMAND: the device server shall send a RESPONSE frame with SENSE_DATA containing CHECK CONDITION/ABORTED COMMAND/OVERLAPPED COMMANDS ATTEMPTED.
b) For COMMAND/TASK, TASK/COMMAND, or TASK/TASK: task manager (peer of the device server) shall send a RESPONSE frame with RESPONSE_DATA containing OVERLAPPED TAG ATTEMPTED.

Both cases result in all commands and task management functions being aborted.

2. The ST_T state machine lacks a means to get the task management function result from the application layer into the RESPONSE frame - the current text refers to status and sense data, which only apply to commands not TMFs.

3. SAM-3 defines an INCORRECT LOGICAL UNIT NUMBER service response, which SAS should map that error to rather than FUNCTION REJECTED.

Suggested changes to SAM-4

3.1.124 task tag: An object containing up to 64 bits that uniquely identifies each task for a given I_T_L nexus (see 3.1.45) in a task set (see 3.1.130). See 4.11.

4.7.6 SCSI task router

The task router routes commands and task management functions as follows:

a) Commands addressed to a valid logical unit number are routed to the task manager in the specified logical unit;
b) Commands addressed to an incorrect logical unit number are handled as described in 5.9.4;
c) Task management functions with I_T_L nexus scope (e.g., ABORT TASK SET, CLEAR TASK SET, CLEAR ACA, and LOGICAL UNIT RESET) or I_T_L_Q nexus scope (e.g., ABORT TASK and QUERY TASK) addressed to a valid logical unit number are routed to the task manager in the specified logical unit;
d) Task management functions with an I_T nexus scope (e.g., I_T NEXUS RESET) are routed to the task manager in each logical unit about which the task router knows; and
e) Task management functions with I_T_L nexus scope or I_T_L_Q nexus scope addressed to an incorrect logical unit number are handled as described in 7.9.

In some transport protocols, the task router may check for overlapped task tags on commands (see 5.9.3).

4.11 Tasks and task tags

A task is represented by an I_T_L_Q nexus (see 4.12) and is composed of:

a) A definition of the work to be performed by the logical unit in the form of a command or a group of linked commands;
b) A task attribute (see 8.6) that allows the application client to specify processing relationships between various tasks in the task set; and
c) Optionally, a task priority (see 8.7).

The I_T_L_Q nexus representing a task includes a task tag, allowing many uniquely identified tagged tasks to be present in a single task set. A task tag is a value that is composed of up to 64 bits.

A SCSI initiator device assigns task tag values for each I_T_L_Q nexus in a way that ensures that the nexus uniqueness requirements stated in this subclause are met. Transport protocols may define additional restrictions on task tag assignment (e.g., restricting task tag length, requiring task tags to be unique per I_T nexus or per I_T_L nexus, or sharing task tag values with other uses such as task management functions).

An I_T_L_Q nexus that is in use (i.e., during the interval bounded by the events specified in 5.5) shall be unique as seen by the SCSI initiator port originating the command and the logical unit to which the command was addressed, otherwise an overlapped command condition exists (see 5.9.3). An I_T_L_Q nexus is unique if one or more of its components is unique within the specified time interval.

A SCSI initiator device shall not create more than one task from a specific SCSI initiator port having identical values for the target port identifier, logical unit number, and task tag.

5.9.3 Overlapped commands

An overlapped command occurs when a task router or task manager detects the use of a duplicate I_T_L_Q nexus (see 4.11) in a command before a task holding that I_T_L_Q nexus completes its task lifetime (see 5.5). Each SCSI transport protocol standard shall specify whether or not a task router or task manager is required to detect overlapped commands.

A task router or task manager that detects an overlapped command shall abort all tasks received on the I_T nexus on which the overlapped command was received and the device server shall return CHECK CONDITION status for the overlapped command. The sense key shall be set to ABORTED COMMAND and the additional sense code shall be set to OVERLAPPED COMMANDS ATTEMPTED.

NOTE 9 - An overlapped command may be indicative of a serious error and, if not detected, may result in corrupted data. This is considered a catastrophic failure on the part of the SCSI initiator device. Therefore, vendor specific error recovery procedures may be required to guarantee the data integrity on the medium. The SCSI target device logical unit may return additional sense data to aid in this error recovery procedure (e.g., sequential-access devices may return the residue of blocks remaining to be written or read at the time the second command was received).

Suggested changes to SAS-1.1

4.5 I_T nexus loss

When a SAS port receives OPEN_REJECT (NO DESTINATION), OPEN_REJECT (PATHWAY BLOCKED), or an open connection timeout occurs in response to a connection request, it shall retry the connection request until:

a) the connection is established;
b) for SSP target ports, the time indicated by the i_t nexus loss field in the Protocol-Specific Port mode page (see 10.2.6.2) expires; or
c) for STP or SMP connection requests, a vendor-specific I_T nexus loss time expires.

An SSP initiator port should retry the connection request for the time indicated by the i_t nexus loss field in the Protocol-Specific Port mode page (see 10.2.6.2) for the SSP target port to which it is trying to establish a connection.

If the time expires, then the port shall send a Nexus Loss event notification to the SCSI application layer (see 10.2.4); the SCSI device shall perform the actions defined for I_T nexus loss in SAM-3.

An I_T nexus loss based on the aforementioned conditions handled by the port layer state machine (see 8.2.2.3).

~~If an SSP target port checks for duplicate tags and receives a COMMAND frame with a tag that is already in use for a task management function or a TASK frame with a tag that is already in use for a command or another task management function, then the SSP target port shall send a Nexus Loss event notification to the SCSI application layer (see 10.2.4). The logical unit shall perform the actions defined for I_T nexus loss in SAM-3. An I_T nexus loss event based on this condition is handled by the SSP transport layer state machine (see 9.2.6.3.2).~~

9.2.2.5.3 RESPONSE information unit RESPONSE_DATA format

If the DATAPRES field is set to RESPONSE_DATA, then:

a) the SSP target port shall set the STATUS field to zero and the SENSE DATA LENGTH field to zero;
b) the SSP initiator port shall ignore the STATUS field and the SENSE DATA LENGTH field;
c) the SSP target port shall not include the SENSE DATA field;
d) the SSP target port shall set the RESPONSE DATA LENGTH field to 00000004h. Other lengths are reserved for future standardization; and
e) the SSP target port shall include the RESPONSE DATA field.

Table 1 defines the RESPONSE DATA field, which contains information describing protocol failures detected during processing of a request received by the SSP target port. The RESPONSE DATA field shall be present if the SSP target port detects any of the conditions described by a non-zero value in the RESPONSE CODE field and shall be present for a RESPONSE frame sent in response to a TASK frame.

Table 1 — RESPONSE DATA **field**

Byte\Bit	7	6	5	4	3	2	1	0
0	Reserved							
2								
3	RESPONSE CODE							

Table 2 defines the RESPONSE CODE field, which specifies the error condition or the completion status of a task management function. See 10.2.1.5 and 10.2.1.15 for the mapping of these response codes to SCSI service responses.

Table 2 — RESPONSE CODE field

Code	Description
00h	TASK MANAGEMENT FUNCTION COMPLETE [a]
02h	INVALID FRAME
04h	TASK MANAGEMENT FUNCTION NOT SUPPORTED [a]
05h	TASK MANAGEMENT FUNCTION FAILED [a]
08h	TASK MANAGEMENT FUNCTION SUCCEEDED [a]
09h	~~INVALID~~ INCORRECT LOGICAL UNIT NUMBER [a]
0Ah	OVERLAPPED TAG ATTEMPTED
All others	Reserved
[a] Only valid when responding to a TASK frame	

9.2.5.6 ST_T (transport layer for SSP target ports) state machines

9.2.5.6.1 ST_T state machines overview

The ST_T state machines are as follows:

a) ST_TFR (target frame router) state machine (see 9.2.5.6.2); and
b) ST_TTS (target transport server) state machine (see 9.2.5.6.3).

This state machine may maintain the timers listed in table 3.

Table 3 — ST_T state machine timers

Timer	Initial value
Initiator Response Timeout	The value in the INITIATOR RESPONSE TIMEOUT field in the Protocol-Specific Port mode page (see 10.2.6.2).

Figure 1 shows the ST_T state machines.

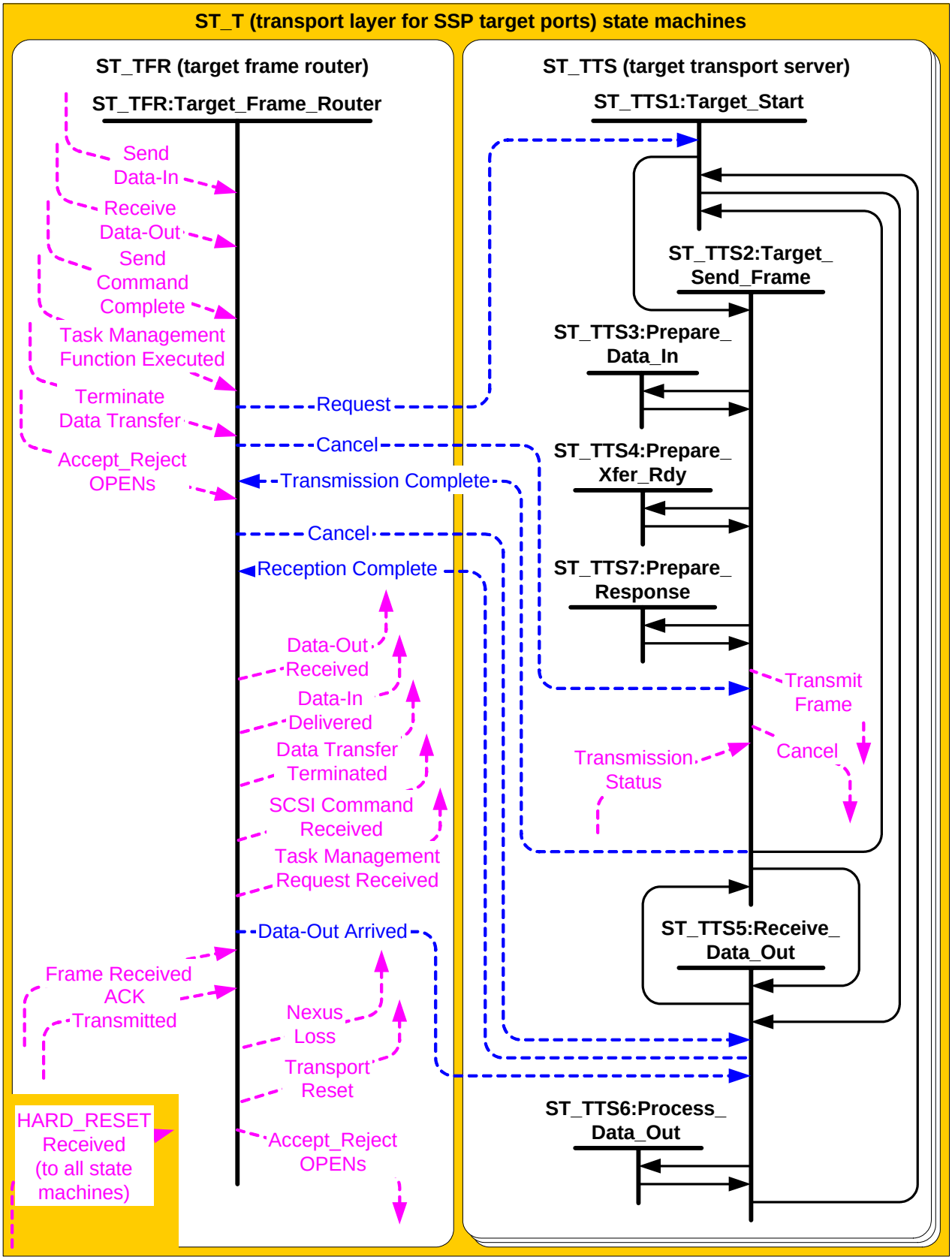

Figure 1 — ST_T (transport layer for SSP target ports) state machines

9.2.5.6.2 ST_TFR (target frame router) state machine

The ST_TFR state machine performs the following functions:

a) receives confirmations from the port layer;
b) receives transport protocol service requests from the SCSI application layer;
c) sends transport protocol service indications to the SCSI application layer;
d) sends messages to the ST_TTS state machine;
e) receives messages from the ST_TTS state machine;
f) receives Accept_Reject OPENs requests from the SCSI application layer;
g) sends Accept_Reject OPENs requests to the port layer;
h) sends Nexus Loss event notifications to the SCSI application layer; and
i) sends Transport Reset event notifications to the SCSI application layer.

This state machine consists of one state.

This state machine shall be started after power on.

If this state machine receives a Frame Received (ACK/NAK Balanced) or Frame Received (ACK/NAK Not Balanced) confirmation, then this state machine shall check the frame type in the received frame (see table 104 in 9.2.1). If the frame type is not COMMAND, TASK, or DATA, then this state machine shall discard the frame. If the confirmation was Frame Received (ACK/NAK Not Balanced) and the frame type is not DATA, then this state machine shall discard the frame.

This state machine may check that reserved fields in the received frame are zero. If any reserved fields are checked and they are not set to zero, then this state machine shall send a Request (Send Transport Response) message with an argument of Invalid Frame to the ST_TTS1:Target_Start state in an ST_TTS state machine that does not have an active task. The check of reserved fields described above shall not apply to the reserved fields within the CDB in a COMMAND frame. Checking of reserved fields in a CDB is described in SPC-3.

If the frame type is correct relative to the confirmation, then this state machine may check that the hashed source SAS address matches the SAS address of the SAS port that transmitted the frame and that the hashed destination SAS address matches the SAS address of the SAS port that received the frame based on the connection information. If this state machine checks these SAS addresses, and they do not match, then this state machine shall discard the frame.

If the frame type is COMMAND or TASK then this state machine shall check the length of the information unit. If the length of the information unit is not correct, then this state machine shall send a Request (Send Transport Response) message with an argument of Invalid Frame to the ST_TTS1:Target_Start state in an ST_TTS state machine that does not have an active task.

~~If the frame type is COMMAND or TASK, then this state machine may check if the tag conflicts with an existing tag (i.e., an existing command or task management function). If this state machine checks the tag, then this state machine shall, based on the condition, take the action specified in table 4.~~

~~Table 4 — Overlapped tag conditions and actions~~

~~Condition~~	~~Action~~
~~The tag for a new COMMAND frame is the same as the tag for a previous COMMAND frame where the task for the previous COMMAND frame is not complete.~~	~~This state machine shall ignore the tag for the new COMMAND frame (i.e., the SCSI application layer detects the overlapped condition).~~
~~The tag for a new COMMAND frame is the same as the tag for a previous TASK frame where the task management function for the previous TASK frame is not complete.~~	~~This state machine shall send a Nexus Loss event notification to the SCSI application layer.~~
~~The tag for a new TASK frame is the same as the tag for a previous COMMAND frame where the task for the previous COMMAND frame is not complete.~~	~~This state machine shall send a Nexus Loss event notification to the SCSI application layer.~~
~~The tag for a new TASK frame is the same as the tag for a previous TASK frame where the task management function for the previous TASK frame is not complete, and the RETRANSMIT bit in the new TASK frame is set to zero.~~	~~This state machine shall send a Nexus Loss event notification to the SCSI application layer.~~
~~The tag for a new TASK frame is the same as the tag for a previous TASK frame where the task management function for the previous TASK frame is not complete, but the RETRANSMIT bit in the new TASK frame is set to one.~~	~~This state machine shall discard the new TASK frame.~~

If the frame type is TASK, this state machine checks tags, the RETRANSMIT bit in the new TASK frame is set to one, and the tag for the new TASK frame is the same as the tag for a previous TASK frame where the task management function for the previous TASK frame is not complete, this state machine shall discard the new TASK frame and not send a Task Management Request Received confirmation to the port layer.

If the frame type is TASK and this state machine does not check tags, then this state machine shall ignore the RETRANSMIT bit.

Editor's Note 1: ST_TFR simply sends up SCSI Command Received or Task Management Request Received and lets the application determine if there is a tag conflict and return a RESPONSE frame indicating the problem (and do whatever else is required). Send (Request Transport Response) will not imply a RESPONSE frame carrying RESPONSE_DATA type only requested by the transport layer; it can also be requested by the application layer. This helps the overlap case and also is needed for normal TMF completion, which was not supported.

If the frame type is COMMAND or TASK, then this state machine may check the target port transfer tag. If this state checks the target port transfer tag and the tag is set to a value other than FFFFh, then this state machine shall send a Request (Send Transport Response) message with an argument of Invalid Frame to the ST_TTS1:Target_Start state in an ST_TTS state machine that does not have an active task.

If the frame type is TASK, then this state machine shall check the logical unit number. If the logical unit number is unknown, then this state machine shall send a Request (Send Transport Response) message with an argument of ~~Invalid~~ Incorrect Logical Unit Number to the ST_TTS1:Target_Start state in an ST_TTS state machine that does not have an active task.

If the frame type is DATA and this frame is for first burst data or this state machine did not assign a target port transfer tag for the data transfer, then this state machine may check the target port transfer tag. If target port transfer tag is set to a value other than FFFFh, then this state machine may send a Request (Send Transport Response) message with an argument of Invalid Frame to the ST_TTS1:Target_Start state in an ST_TTS state machine that does not have an active task.

If this state machine sends a Request (Send Transport Response) message with an argument of Invalid Frame to an ST_TTS state machine ~~as the result of this state machine receiving an invalid frame~~, then this state machine shall discard the frame and shall include the following arguments in the message:

a) connection rate;
b) initiator connection tag;
c) destination SAS address set to the SAS address from which the invalid frame was received;
d) source SAS address set to the SAS address of the SAS port that received the invalid frame; and
e) tag.

If the frame type is COMMAND or TASK and the items checked in the frame are correct, then this state machine shall wait to receive an ACK Transmitted confirmation.

If the frame type is COMMAND, the items checked in the frame are correct, and this state machine receives an ACK Transmitted confirmation, then this state machine shall send a SCSI Command Received transport protocol service indication with the following arguments to the SCSI application layer:

a) source SAS address (i.e., the SAS address that transmitted the COMMAND frame);
b) tag;
c) logical unit number;
d) task attribute;
e) task priority;
f) CDB; and
g) additional CDB bytes, if any.

If the frame type is TASK, the items checked in the frame are correct, and this state machine receives an ACK Transmitted confirmation, then this state machine shall send a Task Management Request Received transport protocol service indication to the SCSI application layer. The indication shall include:

a) source SAS address (i.e., the SAS address that transmitted the TASK frame);
b) tag;
c) logical unit number;
d) task management function; and
e) tag of the task to be managed.

If the frame type is DATA, and the tag does not match a tag for an outstanding command performing write operations, then this state machine shall discard the frame.

If the frame type is DATA, and the tag matches a tag for an outstanding command performing write operations without first burst data enabled or for which no Transmission Complete (Xfer_Rdy Delivered) message has been received from an ST_TTS state machine, then this state machine shall discard the frame.

If the frame type is DATA and a target port transfer tag was assigned in an XFER_RDY frame for the request, then this state machine shall check the target port transfer tag. If the target port transfer tag does not specify a valid state machine, then this state machine shall discard the frame.

If the frame type is DATA and the items checked in the frame are correct, and there is first burst data enabled or this state machine has received a Transmission Complete (Xfer_Rdy Delivered) from the ST_TTS state machine for the request, then this state machine shall send a Data-Out Arrived message to the ST_TTS5:Receive_Data_Out state in the ST_TTS state machine specified by the tag in the frame. The message shall include the content of the SSP frame.

If this state machine receives a Send Data-In transport protocol service request from the SCSI application layer, then this state machine shall send a Request (Send Data-In) message to the ST_TTS1:Target_Start state in an ST_TTS state machine that does not have an active task.

If this state machine receives a Receive Data-Out transport protocol service request from the SCSI application layer, then this state machine shall send a Request (Receive Data-Out) message to the ST_TTS1:Target_Start state in an ST_TTS state machine that does not have an active task.

A Request (Send Data-In) message or a Request (Receive Data-Out) message shall include the following arguments:

a) connection rate;
b) initiator connection tag;
c) destination SAS address (i.e., the SAS address to which the XFER_RDY frame or read DATA frame is to be transmitted);
d) tag;
e) device server buffer; and
f) request byte count.

A Request (Receive Data-Out) message shall also include the target port transfer tag.

If this state machine receives a Send Command Complete transport protocol service response from the SCSI application layer with the Service Response argument set to TASK COMPLETE or LINKED COMMAND COMPLETE, then this state machine shall send a Request (Send Application Response) message to the ST_TTS1:Target_Start state in the ST_TTS state machine specified by the tag. The message shall include the following arguments:

a) connection rate;
b) initiator connection tag;
c) destination SAS address (i.e., the SAS address to which the RESPONSE frame is to be transmitted);
d) tag;
e) the status; and
f) the sense data.

If this state machine receives a Task Management Function Executed transport protocol service response from the SCSI application layer, then this state machine shall send a Request (Send ~~Application~~Transport Response) message to the ST_TTS state machine specified by the tag with the argument described in table 5 and ~~with~~ the following arguments ~~to the ST_TTS1:Target_Start state in an ST_TTS state machine that does not have an active task~~:

a) connection rate;
b) initiator connection tag;
c) destination SAS address (i.e., the SAS address to which the RESPONSE frame is to be transmitted); and
d) tag;
~~e) the status; and~~
f) ~~the sense data~~..

Table 5 — Task Management Function Executed Service Response argument mapping

Service Response argument received	Request (Send Transport Response) argument to send
FUNCTION COMPLETE	Task Management Function Complete
FUNCTION SUCCEEDED	Task Management Function Succeeded
FUNCTION REJECTED	Task Management Function Not Supported
INCORRECT LOGICAL UNIT NUMBER	Incorrect Logical Unit Number
SERVICE DELIVERY OR TARGET FAILURE - Overlapped Tag Attempted	Overlapped Tag Attempted

If this state machine receives a Terminate Data Transfer protocol service request from the SCSI application layer and this state machine has not sent a Request message to a ST_TTS state machine for the Send Data-In or Receive Data-Out protocol service request to which the Terminate Data Transfer request applies,

then this state machine shall discard the Terminate Data Transfer request and any corresponding Send Data-In or Receive Data-Out request. This state shall then send a Data Transfer Terminated protocol service confirmation to the SCSI application layer.

If this state machine receives a Terminate Data Transfer protocol service request from the SCSI application layer and this state machine has sent a Request message to a ST_TTS state machine for the Send Data-In protocol service request to which the Terminate Data Transfer request applies, then this state machine shall send a Cancel message to the ST_TTS2:Target_Send_Frame state in the ST_TTS state machine specified by the tag and the Send Data-In protocol service request.

If this state machine receives a Terminate Data Transfer protocol service request from the SCSI application layer and this state machine has sent a Request message to a ST_TTS state machine for the Receive Data-Out protocol service request to which the Terminate Data Transfer request applies, then this state machine shall send a Cancel message to the ST_TTS5:Receive_Data_Out state in the ST_TTS state machine specified by the tag and the Receive Data-Out protocol service request.

This state machine receives Transmission Complete and Reception Complete messages from the ST_TTS state machines that may result in this state machine sending a Nexus Loss event notification or a protocol service confirmation to the SCSI application layer. If this state machine receives a Transmission Complete (I_T Nexus Loss) message, then this state machine shall send a Nexus Loss event notification to the SCSI

application layer. Table 6 defines the received messages that require a protocol service confirmation and the corresponding service confirmations that shall be sent upon receipt of the message.

Table 6 — Confirmations sent to the SCSI application layer

Message received from ST_TTS state machine	Protocol service confirmation sent to SCSI application layer
Transmission Complete (Data-In Delivered)	Data-In Delivered with the Delivery Result argument set to DELIVERY SUCCESSFUL
Transmission Complete (Xfer_Rdy Failed, NAK Received)	Data-Out Received with the Delivery Result argument set to DELIVERY FAILURE - NAK RECEIVED
Transmission Complete (Xfer_Rdy Failed, Connection Failed)	Data-Out Received with the Delivery Result argument set to DELIVERY FAILURE - CONNECTION FAILED
Transmission Complete (Data Failed, NAK Received)	Data-In Delivered with the Delivery Result argument set to DELIVERY FAILURE - NAK RECEIVED
Transmission Complete (Data Failed, Connection Failed)	Data-In Delivered with the Delivery Result argument set to DELIVERY FAILURE - CONNECTION FAILED
Reception Complete (Data-Out Received)	Data-Out Received with the Delivery Result argument set to DELIVERY SUCCESSFUL
Reception Complete (Data Offset Error)	Data-Out Received with the Delivery Result argument set to DELIVERY FAILURE - DATA OFFSET ERROR
Reception Complete (Too Much Write Data)	Data-Out Received with the Delivery Result argument set to DELIVERY FAILURE - TOO MUCH WRITE DATA
Reception Complete (Information Unit Too Short)	Data-Out Received with the Delivery Result argument set to DELIVERY FAILURE - INFORMATION UNIT TOO SHORT.
Reception Complete (Initiator Response Timeout)	Data-Out Received with the Delivery Result argument set to DELIVERY FAILURE - INITIATOR RESPONSE TIMEOUT
Transmission Complete (Data Transfer Terminated)	Data Transfer Terminated
Reception Complete (Data Transfer Terminated)	Data Transfer Terminated

Each protocol service confirmation shall include the tag as an argument.

If this state machine receives an Accept_Reject OPENs (Accept SSP) or Accept_Reject OPENs (Reject SSP) request, then this state machine shall send a corresponding Accept_Reject OPENs request to the port layer.

If this state machine receives a HARD_RESET Received confirmation, then this state shall send a Transport Reset event notification to the SCSI application layer.

9.2.5.6.3 ST_TTS (target transport server) state machine

9.2.5.6.3.1 ST_TTS state machine overview

The ST_TTS state machine performs the following functions:

a) receives and processes messages from the ST_TFR state machine;
b) sends messages to the ST_TFR state machine;
c) communicates with the port layer using requests and confirmations regarding frame transmission; and
d) receives HARD_RESET Received confirmations from the port layer.

This state machine consists of the following states:

a) ST_TTS1:Target_Start (see 9.2.5.6.3.2) (initial state);
b) ST_TTS2:Target_Send_Frame (see 9.2.5.6.3.3);
c) ST_TTS3:Prepare_Data_In (see 9.2.5.6.3.4);
d) ST_TTS4:Prepare_Xfer_Rdy (see 9.2.5.6.3.5);
e) ST_TTS5:Receive_Data_Out (see 9.2.5.6.3.6);
f) ST_TTS6:Process_Data_Out (see 9.2.5.6.3.7); and
g) ST_TTS7:Prepare_Response (see 9.2.5.6.3.8).

Each ST_TTS state machines shall be started in the ST_TTS1:Target_Start state after power on. There shall be one ST_TTS state machine for each possible task that may be accepted by the SAS target port.

If transport layer retries are enabled, this state machine:

a) shall retain the data offset for a read DATA frame transmitted for which ACK/NAK Balance was achieved (i.e., when the number of read DATA frames sent matches the number of ACK Received confirmations received) for use as the restart point in case of a retry for read DATA frames; and
b) shall retain the data offset for the last XFER_RDY frame transmitted, for use in these cases:
   A) if this state machine retries the XFER_RDY frame, to retransmit the XFER_RDY frame; or
   B) if the SSP initiator port retries write DATA frames, to help reestablish the write data context.

If this state machine receives a HARD_RESET Received confirmation, then this state machine shall transition to the ST_TTS1:Target_Start state.

9.2.5.6.3.2 ST_TTS1:Target_Start state

9.2.5.6.3.2.1 State description

This state is the initial state of an ST_TTS state machine.

9.2.5.6.3.2.2 Transition ST_TTS1:Target_Start to ST_TTS2:Target_Send_Frame

This transition shall occur after this state receives a Request message from the ST_TFR state machine other than a Request (Receive Data-Out) message if first burst is enabled.

9.2.5.6.3.2.3 Transition ST_TTS1:Target_Start to ST_TTS5:Receive_Data_Out

This transition shall occur after this state receives a Request (Receive Data-Out) message from the ST_TFR state machine and first burst is enabled.

9.2.5.6.3.3 ST_TTS2:Target_Send_Frame state

9.2.5.6.3.3.1 State description

If this state is entered from the ST_TTS3:Prepare_Data_In state for transmission of a read DATA frame, then this state shall send a Transmit Frame (Non-Interlocked) request to the port layer.

If this state is entered from the ST_TTS4:Prepare_Xfer_Rdy state for transmission of an XFER_RDY frame, then this state shall send a Transmit Frame (Interlocked) request to the port layer.

If this state is entered from the ST_TTS7:Prepare_Response state for transmission of a RESPONSE frame, then this state shall send a Transmit Frame (Interlocked) request to the port layer.

NOTE 10 - The XFER_RDY and RESPONSE frame rules ensure that wide ports do not send an XFER_RDY or RESPONSE frame on a phy until all the ACKs have been transmitted for write DATA frames on a different phy. In a narrow port, the link layer ensures that ACK/NAKs are balanced before transmitting an interlocked frame.

A Transmit Frame request from this state shall include the SSP frame and the following arguments to be used for any OPEN address frame:

a) initiator port bit set to zero;
b) protocol set to SSP;
c) connection rate;
d) initiator connection tag;
e) destination SAS address; and
f) source SAS address set to the SAS address of the SSP target port.

After sending a Transmit Frame request this state shall wait to receive a Transmission Status confirmation.

If the confirmation is Transmission Status (I_T Nexus Loss), then this state shall send a Transmission Complete (I_T Nexus Loss) message to the ST_TFR state machine. The Transmission Complete message shall include the tag as an argument.

If the confirmation is not Transmission Status (Frame Transmitted) or Transmission Status (I_T Nexus Loss), and the Transmit Frame request was for an XFER_RDY, DATA, or RESPONSE frame, then this state shall send a Transmission Complete (Connection Failed) message to the ST_TFR state machine. The message shall include the following arguments:

a) the tag; and
b) the arguments received with the Transmission Status confirmation.

If the confirmation is Transmission Status (Frame Transmitted) and the Transmit Frame request was for an XFER_RDY frame, a RESPONSE frame, or a read DATA frame where the number of bytes that have been transmitted equal the request byte count (i.e., all data has been transferred for the request), then this state shall wait to receive one of the following confirmations:

a) Transmission Status (ACK Received);
b) Transmission Status (NAK Received);
c) Transmission Status (ACK/NAK Timeout); or
d) Transmission Status (Connection Lost Without ACK/NAK).

NOTE 11 - If the number of data bytes that have been transmitted for the Request (Send Data-In) message are fewer than the request byte count, then this state transitions to the ST_TTS3:Prepare_Data_In state to construct the additional read DATA frames for the request before receiving a Transmission Status (ACK Received), Transmission Status (NAK Received), Transmission Status (ACK/NAK Timeout), or Transmission Status (Connection Lost Without ACK/NAK) confirmation for Transmit Frame requests for previous read DATA frames sent for the I_T_L_Q nexus (see 9.2.5.6.3.3.4).

Table 7 defines the confirmations to be received from the port layer after a Transmission Status (Frame Transmitted) and the message that shall be sent by this state to the ST_TFR state machine upon receipt of the confirmation based on the conditions under which the confirmation was received.

Table 7 — Messages sent to the ST_TFR state machine based on port layer confirmations

Confirmation received from the port layer	Conditions under which confirmation was received	Message sent to the ST_TFR state machine
Transmission Status (ACK Received)	the Transmit Frame request was for an XFER_RDY frame	Transmission Complete (Xfer_Rdy Delivered) with the target port transfer tag argument
Transmission Status (ACK Received)	Transmit Frame request was for a RESPONSE frame	Transmission Complete (Response Delivered)
Transmission Status (ACK Received)	a) the Transmit Frame request was for a read DATA frame; b) the number of data bytes transmitted equal the request byte count; and c) this state has received a Transmission Status (ACK Received) confirmation for each read DATA frame transmitted for the request	Transmission Complete (Data-In Delivered)
Transmission Status (NAK Received), Transmission Status (ACK/NAK Timeout), or Transmission Status (Connection Lost Without ACK/NAK)	a) the Transmit Frame request was for a RESPONSE frame; and b) the vendor-specific number of retries has been reached	Transmission Complete (Response Delivery Failed)
Transmission Status (NAK Received)	a) the Transmit Frame request was for an XFER_RDY frame; and b) the vendor-specific number of retries has been reached	Transmission Complete (Xfer_Rdy Failed, NAK Received)
Transmission Status (ACK/NAK Timeout) or Transmission Status (Connection Lost Without ACK/NAK)	a) the Transmit Frame request was for an XFER_RDY frame; and b) the vendor-specific number of retries has been reached	Transmission Complete (Xfer_Rdy Failed, Connection Failed)
Transmission Status (NAK Received)	a) the Transmit Frame request was for a read DATA frame; and b) the vendor-specific number of retries has been reached	Transmission Complete (Data Failed, NAK Received)
Transmission Status (ACK/NAK Timeout) or Transmission Status (Connection Lost Without ACK/NAK)	a) the Transmit Frame request was for a read DATA frame; and b) the vendor-specific number of retries has been reached	Transmission Complete (Data Failed, Connection Failed)

If this state receives a Cancel message from the ST_TFR state machine and this state has received confirmations for all Transmit Frame requests sent to the port layer, then this state shall send a Transmission Complete (Cancel Acknowledged) message to the ST_TFR state machine.

If this state receives a Cancel message from the ST_TFR state machine and this state has not received confirmations for all Transmit Frame requests sent to the port layer, then this state shall send a Cancel

request to the port layer to cancel previous Transmit Frame requests. The Cancel request shall include the following arguments:

a) the destination SAS address; and
b) the tag.

Upon receipt of a Transmission Status (Cancel Acknowledged) confirmation this state shall send a Transmission Complete (Cancel Acknowledged) message to the ST_TFR state machine.

A Transmission Complete message to the ST_TFR state machine shall include the following arguments:

a) the destination SAS address; and
b) the tag.

9.2.5.6.3.3.3 Transition ST_TTS2:Target_Send_Frame to ST_TTS1:Target_Start

This transition shall occur after this state has sent a Transmission Complete message other than Transmission Complete (Xfer_Rdy Delivered) to the ST_TFR state machine.

9.2.5.6.3.3.4 Transition ST_TTS2:Target_Send_Frame to ST_TTS3:Prepare_Data_In

This transition shall occur after this state:

a) receives a Transmission Status (ACK Received) confirmation for an XFER_RDY frame;
b) receives a Transmission Status (Frame Transmitted) confirmation for a read DATA frame and the number of bytes moved for a Request (Send Data-In) message is less than the requested byte count; or
c) receives a Transmission Status (Frame Transmitted) confirmation and a confirmation other than Transmission Status (ACK Received) for a read DATA frame for which a Transmission Complete message was not sent to the ST_TFR state machine (i.e., in order to retry transmitting the frame when the vendor-specific number of retries has not been reached).

9.2.5.6.3.3.5 Transition ST_TTS2:Target_Send_Frame to ST_TTS4:Prepare_Xfer_Rdy

This transition shall occur after this state machine has received a Request (Receive Data-Out) message and:

a) first burst is enabled, all first burst data has been received, and there is more data to transfer for the message;
b) first burst is not enabled and no XFER_RDY has been transmitted for the message;
c) all data for a previous XFER_RDY has been received and there is more data to transfer for the message; or
d) this state receives a Transmission Status (Frame Transmitted) confirmation for an XFER_RDY frame followed by a Transmission Status (NAK Received), Transmission Status (ACK/NAK Timeout), or Transmission Status (Connection Lost Without ACK/NAK) confirmation for the frame, and the vendor-specific number of retries has not been reached.

9.2.5.6.3.3.6 Transition ST_TTS2:Target_Send_Frame to ST_TTS5:Receive_Data_Out

This transition shall occur after this state receives a Transmission Status (Frame Transmitted) confirmation for an XFER_RDY frame followed by a Transmission Status (ACK Received) confirmation for the frame.

9.2.5.6.3.3.7 Transition ST_TTS2:Target_Send_Frame to ST_TTS7:Prepare_Response

This transition shall occur after:

a) this state machine receives a Request (Send Transport Response) message from the ST_TFR state machine;
b) this state machine receives a Request (Send Application Response) message from the ST_TFR state machine; or
c) this state receives a Transmission Status (Frame Transmitted) confirmation for a RESPONSE frame followed by a Transmission Status (NAK Received), Transmission Status (ACK/NAK Timeout), or Transmission Status (Connection Lost Without ACK/NAK) confirmation for the frame, and the

vendor-specific number of retries has not been reached. The number of retries for a RESPONSE frame shall be greater than or equal to one.

9.2.5.6.3.4 ST_TTS3:Prepare_Data_In state

9.2.5.6.3.4.1 State description

This state retrieves the data from the Device Server Buffer and constructs a read DATA frame. This state shall use arguments received in the Request (Send Data-In) message to construct the frame (see 9.2.5.6.2).

This state shall generate the following values when constructing the frame:

a) FRAME TYPE field set to 01h (i.e., DATA frame);
b) HASHED DESTINATION SAS ADDRESS field set to the hashed value of the specified destination SAS address;
c) HASHED SOURCE SAS ADDRESS field set to the hashed value of the SSP target port's SAS address;
d) RETRY DATA FRAMES bit set to zero;
e) RETRANSMIT bit set to zero;
f) CHANGING DATA POINTER bit set to zero, unless otherwise specified in this subclause;
g) NUMBER OF FILL BYTES field set to the number of fill bytes needed for the specified data;
h) TAG field set to the specified tag;
i) TARGET PORT TRANSFER TAG field set to zero;
j) DATA OFFSET field set as specified in this subclause; and
k) in the information unit, DATA field set to the specified data; and
l) fill bytes, if any.

If this is the first read DATA frame constructed by this state, then this state shall set the CHANGING DATA POINTER bit and the DATA OFFSET field in the read DATA frame to zero.

If this state is entered after the ST_TTS2:Target_Send_Frame state received a Transmission Status (Frame Transmitted) confirmation and ST_TTS2:Target_Send_Frame state has only received confirmations of Transmission Status (Frame Transmitted) and Transmission Status (ACK Received), then this state shall generate the following values when constructing the frame:

a) set the CHANGING DATA POINTER bit set to zero; and
b) set the DATA OFFSET field set to the value in the DATA OFFSET field in the previous read DATA frame plus the number of bytes in the previous DATA information unit.

If this state is entered after the ST_TTS2:Target_Send_Frame state received a Transmission Status (Frame Transmitted) confirmation and a confirmation other than Transmission Status (ACK Received) for which a Transmission Complete message was not sent to the ST_TFR state machine (i.e., to retry transmitting a frame), then this state shall generate the following values when constructing the frame:

a) set the CHANGING DATA POINTER bit in the frame to one; and
b) set the DATA OFFSET FIELD to a data offset value associated with a previous ACK/NAK balance.

9.2.5.6.3.4.2 Transition ST_TTS3:Prepare_Data_In to ST_TTS2:Target_Send_Frame

This transition shall occur after this state constructs a read DATA frame.

9.2.5.6.3.5 ST_TTS4:Prepare_Xfer_Rdy state

9.2.5.6.3.5.1 State description

This state shall construct an XFER_RDY frame using arguments received in the Request (Receive Data-Out) message (see 9.2.5.6.2):

a) FRAME TYPE field set to 05h (i.e., XFER_RDY frame);
b) HASHED DESTINATION SAS ADDRESS field set to the hashed value of the specified destination SAS address;
c) HASHED SOURCE SAS ADDRESS field set to the hashed value of the SSP target port's SAS address;

d) RETRY DATA FRAMES bit set to the value of the TRANSPORT LAYER RETRIES bit in the Protocol-Specific Logical Unit mode page (see 10.2.6.3);
e) RETRANSMIT bit set to zero, unless otherwise specified in this subclause;
f) CHANGING DATA POINTER bit set to zero;
g) NUMBER OF FILL BYTES field set to zero;
h) TAG field set to the specified tag;
i) TARGET PORT TRANSFER TAG field set to a vendor-specific value, unless otherwise specified in this subclause;
j) DATA OFFSET field set to zero; and
k) in the information unit, REQUESTED OFFSET field set to the specified data offset; and
l) in the information unit, WRITE DATA LENGTH field set to the specified write data length.

If first burst is enabled, this state shall adjust the write data length to reflect the amount of first burst data.

If this state is entered after the ST_TTS2:Target_Send_Frame state received a Transmission Status (Frame Transmitted) confirmation and a confirmation other than Transmission Status (ACK Received) for which a Transmission Complete message was not sent to the ST_TFR state machine (i.e., to retry transmitting a frame), then this state shall construct a new XFER_RDY frame using the values from the previous XFER_RDY frame except:

a) the RETRANSMIT bit shall be set to one; and
b) the value in the TARGET PORT TRANSFER TAG field shall be set to a different value than the value in the previous XFER_RDY frame. The new target port transfer tag value shall not conflict with any other target port transfer tag currently in use. If write data is received for a subsequent XFER_RDY frame for a command, then all target port transfer tags used for previous XFER_RDY frames for the command are no longer in use.

9.2.5.6.3.5.2 Transition ST_TTS4:Prepare_Xfer_Rdy to ST_TTS2:Target_Send_Frame

This transition shall occur after this state constructs an XFER_RDY frame.

9.2.5.6.3.6 ST_TTS5:Receive_Data_Out state

9.2.5.6.3.6.1 State description

If this state receives a Data-Out Arrived message from the ST_TFR state machine, then this state shall verify the write DATA frame received with the message as follows:

1) If the data offset was not expected (i.e., the CHANGING DATA POINTER bit is set to one and the value in the DATA OFFSET field is not set to the data offset associated with the XFER_RDY frame, or the CHANGING DATA POINTER bit is set to zero and the value in the DATA OFFSET field is not set to the value in the DATA OFFSET FIELD in the previous write DATA information unit plus the number of bytes in that information unit), then this state shall send a Reception Complete (Data Offset Error) message to the ST_TFR state machine;
2) If first burst is enabled and the length of the information unit DATA field is greater than the amount indicated by the FIRST BURST SIZE field in the Disconnect-Reconnect mode page (see 10.2.6.1.5), or if an XFER_RDY frame was sent requesting the write data and the length of the information unit DATA field plus the length of the write data previously received is greater than that specified by the XFER_RDY frame, then this state shall send a Reception Complete (Too Much Write Data) message to the ST_TFR state machine;
3) If the length of the information unit DATA field is zero, then this state shall send a Reception Complete (Information Unit Too Short) message to the ST_TFR-state machine.

The Reception Complete message, if any, shall include the tag as an argument.

If the Initiator Response Timeout timer is implemented, this state shall initialize and start the Initiator Response Timeout timer after any of the following occur:

a) this state is entered from the ST_TTS1:Target_Start state (i.e., a Request (Receive Data-Out) message is received and first burst is enabled);

b) this state is entered from the ST_TTS2:Target_Send_Frame state (i.e., an XFER_RDY was successfully transmitted); or
c) this state is entered from the ST_TTS6:Process_Data_Out state (i.e., Data-Out data was received and processed).

If the Initiator Response Timeout timer is running, this state shall stop the timer before transitioning from this state.

If the Initiator Response Timeout timer expires this state shall send a Reception Complete (Initiator Response Timeout) message to the ST_TFR state machine.

If this state is entered from the ST_TTS6:Process_Data_Out state and number of bytes moved for the Request (Receive Data-Out) message equals the request byte count, then this state shall send a Reception Complete (Data-Out Received) message to the ST_TFR state machine.

If this state receives a Cancel message from the ST_TFR state machine, then this state shall send a Reception Complete (Cancel Acknowledged) message to the ST_TFR state machine.

9.2.5.6.3.6.2 Transition ST_TTS5:Receive_Data_Out to ST_TTS6:Process_Data_Out

This transition shall occur after this state receives and verifies a Data-Out Arrived message.

9.2.5.6.3.6.3 Transition ST_TTS5:Receive_Data_Out to ST_TTS1:Target_Start

This transition shall occur after this state sends a Reception Complete message to the ST_TFR state machine.

9.2.5.6.3.7 ST_TTS6:Process_Data_Out state

9.2.5.6.3.7.1 State description

This state shall process the data received in the Data-Out Arrived message using the Device Server Buffer (e.g., logical block address) to which the data is to be transferred.

9.2.5.6.3.7.2 Transition ST_TTS6:Process_Data_Out to ST_TTS5:Receive_Data_Out

This transition shall occur after this state has processed the data received in a Data-Out Arrived message.

9.2.5.6.3.8 ST_TTS7:Prepare_Response state

9.2.5.6.3.8.1 State description

If this state was entered as a result of this state machine receiving a Request (Send Transport Response) or a Request (Send Application Response) message, then this state shall construct a RESPONSE frame using arguments received with the Request message (see 9.2.5.6.2):

a) FRAME TYPE field set to 07h (i.e., RESPONSE frame);
b) HASHED DESTINATION SAS ADDRESS field set to the hashed value of the specified destination SAS address;
c) HASHED SOURCE SAS ADDRESS field set to the hashed value of the SSP target port's SAS address;
d) RETRY DATA FRAMES bit set to zero;
e) RETRANSMIT bit set to zero;
f) CHANGING DATA POINTER bit set to zero;
g) TAG field set to the specified tag;
h) TARGET PORT TRANSFER TAG field set to zero;
i) DATA OFFSET field set to zero.

If this state was entered as a result of this state machine receiving a Request (Send Transport Response), this state shall set the fields relating to the information unit as follows:

a) NUMBER OF FILL BYTES field set to 0h;
b) in the information unit, DATAPRES field set to RESPONSE_DATA;
c) in the information unit, STATUS field set to zero;

d) in the information unit, SENSE DATA LENGTH field set to zero;
e) in the information unit, RESPONSE DATA LENGTH field set to 00000004h;
f) in the information unit, RESPONSE DATA field set as specified in table 121;
g) in the information unit, SENSE DATA field not included; and
h) fill bytes, if any.

Table 121 specifies how the information unit fields shall be set based on the arguments received with the Request (Send Transport Response) message.

Table 8 — Request argument to RESPONSE frame RESPONSE DATA field mapping

Request argument	~~RESPONSE frame~~ RESPONSE DATA field
~~Invalid Frame~~	~~The information unit fields shall be set as follows:~~ ~~a) DATAPRES field shall be set to RESPONSE_DATA;~~ ~~b) STATUS field shall be set to 00h;~~ ~~c) SENSE DATA LENGTH field shall be set to 00000000h;~~ ~~d) RESPONSE DATA LENGTH field shall be set to 00000004h;~~ ~~e) RESPONSE DATA field shall be set to INVALID FRAME; and~~ ~~f) SENSE DATA field shall not be included.~~
~~Invalid Logical Unit Number~~	~~The information unit fields shall be set as follows:~~ ~~a) DATAPRES field shall be set to RESPONSE_DATA;~~ ~~b) STATUS field shall be set to 00h;~~ ~~c) SENSE DATA LENGTH field shall be set to 00000000h;~~ ~~d) RESPONSE DATA LENGTH field shall be set to 00000004h;~~ ~~e) RESPONSE DATA field shall be set to INVALID LOGICAL UNIT NUMBER; and~~ ~~f) SENSE DATA field shall not be included.~~
Invalid Frame	INVALID FRAME
Function Complete	TASK MANAGEMENT FUNCTION COMPLETE
Function Succeeded	TASK MANAGEMENT FUNCTION SUCCEEDED
Function Not Supported	TASK MANAGEMENT FUNCTION NOT SUPPORTED
Function Failed	TASK MANAGEMENT FUNCTION FAILED
Incorrect Logical Unit Number	INCORRECT LOGICAL UNIT NUMBER
Overlapped Tag Attempted	OVERLAPPED TAG ATTEMPTED

If this state was entered as a result of this state machine receiving a Request (Send Application Response) message, this state shall shall set the fields relating to the information unit as follows:

a) NUMBER OF FILL BYTES field set to the number of fill bytes, based on the length of the specified sense data, if any;
b) in the information unit, DATAPRES field set to SENSE_DATA if sense data is specified or NO_DATA if sense data is not specified;
c) in the information unit, STATUS field set to the specified status;
d) in the information unit, SENSE DATA LENGTH field set to the length of the specified sense data, if any;
e) in the information unit, RESPONSE DATA LENGTH field set to zero;
f) in the information unit, RESPONSE DATA field not included;
g) in the information unit, SENSE DATA field set to the specified sense data, if any; and
h) fill bytes, if any.

Editor's Note 2: above Send Application Response paragraph and its a)b) list moved from above

the RESPONSE frame table to below the table.

If this state was entered as the result of the ST_TTS2:Target_Send_Frame state receiving something other than a Transmission Status (Frame Transmitted) confirmation followed by a Transmission Status (ACK Received) confirmation for a RESPONSE frame (i.e., the frame transmission was unsuccessful and the vendor-specific number of retries has not been reached), then this state shall construct a new RESPONSE frame using all of the values for the previous RESPONSE frame except that the RETRANSMIT bit shall be set to one.

9.2.5.6.3.8.2 Transition ST_TTS7:Prepare_Response to ST_TTS2:Target_Send_Frame

This transition shall occur after this state constructs a RESPONSE frame.

10.2.1.3 SCSI Command Received transport protocol service

An SSP target port uses the SCSI Command Received transport protocol service indication to notify a device server that it has received a COMMAND frame.

SCSI Command Received (IN (I_T_L_Q Nexus, CDB, Task Attribute, [Task Priority], [Command Reference Number]))

Table 9 shows how the arguments to the SCSI Command Received transport protocol service are determined.

Table 9 — SCSI Command Received transport protocol service arguments

Argument	SAS SSP implementation
I_T_L_Q nexus	I_T_L_Q nexus, where: a) I indicates the initiator port that sent the COMMAND frame; b) T indicates the target port that received the COMMAND frame; c) L indicates the value of the LOGICAL UNIT NUMBER field in the COMMAND frame header; and d) Q indicates the value of the TAG field in the COMMAND frame header.
CDB	Indicates the value of the CDB field in the COMMAND frame.
Task Attribute	Indicates the value of the TASK ATTRIBUTE field in the COMMAND frame.
[Task Priority]	Indicates the value of the TASK PRIORITY field in the COMMAND frame.
[Command Reference Number]	Ignored
[First Burst Enabled]	Indicates that first burst data is being delivered based on the ENABLE FIRST BURST field in the COMMAND frame and the FIRST BURST SIZE field in the Disconnect-Reconnect mode page (see 10.2.6.1.5).

~~If an SSP target port calls SCSI Command Received () with a TAG already in use by another SCSI command (i.e., an overlapped command) or by a SCSI task management function, the device server responses are defined in SAM-3.~~

Editor's Note 3: above paragraph moved down to the device server error handling section and reworded to add aborting task management functions using the I_T nexus

10.2.1.4 Send Command Complete transport protocol service

A device server uses the Send Command Complete transport protocol service response to request that an SSP target port transmit a RESPONSE frame.

Send Command Complete (IN (I_T_L_Q Nexus, [Sense Data], [Sense Data Length], Status, Service Response))

A device server shall only call Send Command Complete () after receiving SCSI Command Received ().

A device server shall not call Send Command Complete () for a given I_T_L_Q nexus until all its outstanding Receive Data-Out () calls have been responded to with Data-Out Received () and all its outstanding Send Data-In () calls have been responded to with Data-In Delivered ().

Table 10 shows how the arguments to the Send Command Complete transport protocol service are used.

Table 10 — Send Command Complete transport protocol service arguments

Argument	SAS SSP implementation
I_T_L_Q nexus	I_T_L_Q nexus, where: a) I specifies the initiator port to which the RESPONSE frame is to be sent; b) T specifies the target port to send the RESPONSE frame; c) L specifies the LOGICAL UNIT NUMBER field in the RESPONSE frame header; and d) Q specifies the TAG field in the RESPONSE frame header.
[Sense Data]	Specifies the SENSE DATA field in the RESPONSE frame.
[Sense Data Length]	Specifies the SENSE DATA LENGTH field in the RESPONSE frame.
Status	Specifies the STATUS field in the RESPONSE frame.
Service Response	Specifies the DATAPRES field and STATUS field in the RESPONSE frame: a) TASK COMPLETE: The DATAPRES field is set to NO_DATA or SENSE_DATA and the STATUS field is set to a value other than INTERMEDIATE or INTERMEDIATE-CONDITION MET; b) LINKED COMMAND COMPLETE: The DATAPRES field is set to NO_DATA or SENSE_DATA and the STATUS field is set to INTERMEDIATE or INTERMEDIATE-CONDITION MET; or c) SERVICE DELIVERY OR TARGET FAILURE: The DATAPRES field is set to RESPONSE_DATA and the RESPONSE CODE field is set to INVALID FRAME or OVERLAPPED TAG ATTEMPTED.

10.2.1.5 Command Complete Received transport protocol service

An SSP initiator port uses the Command Complete Received transport protocol service confirmation to notify an application client that it has received a response for its COMMAND frame (e.g., a RESPONSE frame or a NAK).

Command Complete Received (IN (I_T_L_Q Nexus, [Data-In Buffer], [Sense Data], Status, Service Response))

Table 11 shows how the arguments to the Command Complete Received transport protocol service are determined.

Table 11 — Command Complete Received transport protocol service arguments

Argument	SAS SSP implementation
I_T_L_Q nexus	I_T_L_Q nexus, where: a) I indicates the initiator port that received the RESPONSE frame; b) T indicates the target port that sent the RESPONSE frame; c) L indicates the value of the LOGICAL UNIT NUMBER field in the RESPONSE frame header or COMMAND frame header; and d) Q indicates the value of the TAG field in the RESPONSE frame header or COMMAND frame header.
[Data-In Buffer]	Internal to the SSP initiator port.
[Sense Data]	Indicates the value of the SENSE DATA field in the RESPONSE frame.
[Sense Data Length]	The smaller of the value of the SENSE DATA LENGTH field in the RESPONSE frame and the actual number of sense data bytes received by the SSP initiator port.
Status	Indicates the value of the STATUS field in the RESPONSE frame.
Service Response	From the DATAPRES field and STATUS field in the RESPONSE frame, or from a NAK on the COMMAND frame: a) TASK COMPLETE: The RESPONSE frame contains a DATAPRES field set to NO_DATA or SENSE_DATA and a STATUS field set to a value other than INTERMEDIATE or INTERMEDIATE-CONDITION MET; b) LINKED COMMAND COMPLETE: The RESPONSE frame contains a DATAPRES field set to NO_DATA or SENSE_DATA and a STATUS field set to INTERMEDIATE or INTERMEDIATE-CONDITION MET; or c) SERVICE DELIVERY OR TARGET FAILURE: The RESPONSE frame contains a DATAPRES field set to RESPONSE_DATA and a RESPONSE CODE field set to INVALID FRAME or OVERLAPPED TAG ATTEMPTED, or a NAK was received for the COMMAND frame, or the length of the RESPONSE frame is incorrect.

10.2.1.13 Task Management Request Received transport protocol service

An SSP target port uses the Task Management Request Received transport protocol service indication to notify a ~~device server~~task manager that it has received a TASK frame.

Task Management Request Received (IN (Nexus, Function Identifier, [Association]))

Table 12 shows how the arguments to the Task Management Request Received transport protocol service are determined.

Table 12 — Task Management Request Received transport protocol service arguments

Argument	SAS SSP implementation
Nexus	I_T_L nexus or I_T_L_Q nexus (depending on the Function Identifier), where: a) I indicates the initiator port that sent the TASK frame; b) T indicates the target port that received the TASK frame; c) L indicated by the LOGICAL UNIT NUMBER field in the TASK frame header; and d) Q (for an I_T_L_Q nexus) indicated by the TAG OF TASK TO BE MANAGED field in the TASK frame header.
Function Identifier	Indicates the TASK MANAGEMENT FUNCTION field in the TASK frame. Only these task management functions are supported: a) ABORT TASK (Nexus argument specifies an I_T_L_Q Nexus); b) ABORT TASK SET (Nexus argument specifies an I_T_L Nexus); c) CLEAR ACA (Nexus argument specifies an I_T_L Nexus); d) CLEAR TASK SET (Nexus argument specifies an I_T_L Nexus); e) LOGICAL UNIT RESET (Nexus argument specifies an I_T_L Nexus); and f) QUERY TASK (Nexus argument specifies an I_T_L_Q Nexus).
[Association]	Indicates the TAG field in the TASK frame header.

10.2.1.14 Task Management Function Executed transport protocol service

A ~~device server~~task manager uses the Task Management Function Executed transport protocol service response to request that an SSP target port transmit a RESPONSE frame.

Task Management Function Executed (IN (Nexus, Service Response, [Association]))

A ~~device server~~task manager shall only call Task Management Function Executed () after receiving Task Management Request Received ().

Table 13 shows how the arguments to the Task Management Function Executed transport protocol service are used.

Table 13 — Task Management Function Executed transport protocol service arguments

Argument	SAS SSP implementation
Nexus	I_T_L nexus or I_T_L_Q nexus (depending on the function), where: a) I specifies the initiator port to which the RESPONSE frame is sent; b) T specifies the target port to send the RESPONSE frame; c) L specifies the LOGICAL UNIT NUMBER field in the RESPONSE frame header; and d) Q (for an I_T_L_Q nexus) indirectly specifies the TAG field in the RESPONSE frame header.
Service Response	Specifies the DATAPRES field and RESPONSE CODE field in the RESPONSE frame: a) FUNCTION COMPLETE: The RESPONSE frame DATAPRES field is set to RESPONSE_DATA and the RESPONSE CODE field is set to TASK MANAGEMENT FUNCTION COMPLETE; b) FUNCTION SUCCEEDED: The RESPONSE frame DATAPRES field is set to RESPONSE_DATA and the RESPONSE CODE field is set to TASK MANAGEMENT FUNCTION SUCCEEDED; c) FUNCTION REJECTED: The DATAPRES field is set to RESPONSE_DATA and the RESPONSE CODE field is set to TASK MANAGEMENT FUNCTION NOT SUPPORTED ~~or INVALID LOGICAL UNIT NUMBER.;~~ d) INCORRECT LOGICAL UNIT NUMBER: The DATAPRES field is set to RESPONSE_DATA and the RESPONSE CODE field is set to INCORRECT LOGICAL UNIT NUMBER; or e) SERVICE DELIVERY OR ~~SUBSYSTEM~~ TARGET FAILURE: The RESPONSE frame DATAPRES field is set to RESPONSE_DATA and the RESPONSE CODE field is set to INVALID FRAME ~~or~~, TASK MANAGEMENT FUNCTION FAILED, or OVERLAPPED TAG ATTEMPTED..
[Association]	Specifies the TAG field in the RESPONSE frame header.

Editor's Note 4: resorted the Service Response a)b)c) list

10.2.1.15 Received Task Management Function-Executed transport protocol service

An SSP initiator port uses the Received Task Management Function-Executed transport protocol service confirmation to notify an application client that it has received a response to a TASK frame (e.g., received a RESPONSE frame or a NAK).

Received Task Management Function-Executed (IN (Nexus, Service Response, [Association]))

Table 14 shows how the arguments to the Received Task Management Function-Executed transport protocol service are determined.

Table 14 — Received Task Management Function-Executed transport protocol service arguments

Argument	SAS SSP implementation
Nexus	I_T_L nexus or I_T_L_Q nexus (depending on the function), where: a) I indicates the initiator port that received the RESPONSE frame; b) T indicates the target port that sent the RESPONSE frame; c) L indicates the LOGICAL UNIT NUMBER field in the RESPONSE frame header or TASK frame header; and d) Q (for an I_T_L_Q nexus) indirectly indicates the TAG field in the RESPONSE frame header, or indicates the TAG OF TASK TO BE MANAGED field TASK frame header.
Service Response	Indicates the response to the TASK frame: a) FUNCTION COMPLETE: The RESPONSE frame contains a DATAPRES field set to RESPONSE_DATA and a RESPONSE CODE field set to TASK MANAGEMENT FUNCTION COMPLETE; b) FUNCTION SUCCEEDED: The RESPONSE frame contains a DATAPRES field set to RESPONSE_DATA and a RESPONSE CODE field set to TASK MANAGEMENT FUNCTION SUCCEEDED; c) FUNCTION REJECTED: The RESPONSE frame contains a DATAPRES field set to RESPONSE_DATA and a RESPONSE CODE field set to TASK MANAGEMENT FUNCTION NOT SUPPORTED~~or INVALID LOGICAL UNIT NUMBER~~; d) INCORRECT LOGICAL UNIT NUMBER: The RESPONSE frame contains a DATAPRES field set to RESPONSE_DATA and a RESPONSE CODE field set to INCORRECT LOGICAL UNIT NUMBER; or e) SERVICE DELIVERY OR TARGET FAILURE: The RESPONSE frame contains a DATAPRES field set to RESPONSE_DATA and a RESPONSE CODE field set to INVALID FRAME ~~or~~, TASK MANAGEMENT FUNCTION FAILED, or OVERLAPPED TAG ATTEMPTED, or a NAK was received for the TASK frame, or the length of the RESPONSE frame is incorrect.
[Association]	Indicates the TAG field in the RESPONSE frame header or the TASK frame header.

Editor's Note 5: resorted the Service Response a)b)c) list above

10.2.2 Application client error handling

If an SSP initiator port calls Command Complete Received () and delivers a Service Response of:

a) Service Delivery or Target Failure - XFER_RDY Information Unit Too Short;
b) Service Delivery or Target Failure - XFER_RDY Information Unit Too Long;
c) Service Delivery or Target Failure - XFER_RDY Incorrect Write Data Length;
d) Service Delivery or Target Failure - XFER_RDY Requested Offset Error;
e) Service Delivery or Target Failure - XFER_RDY Not Expected;
f) Service Delivery or Target Failure - DATA Incorrect Data Length;
g) Service Delivery or Target Failure - DATA Too Much Read Data;
h) Service Delivery or Target Failure - DATA Data Offset Error;
i) Service Delivery or Target Failure - NAK Received; or
j) Service Delivery or Target Failure - Connection Failed,

it shall abort the command (e.g., by sending an ABORT TASK task management function).

After an application client calls Send SCSI Command (), if Command Complete Received () returns a Service Response of Service Delivery or Target Failure - ACK/NAK Timeout, the application client shall send a QUERY TASK task management function with Send Task Management Request () to determine whether the command was received successfully. If Received Task Management Function Executed () returns a Service Response of FUNCTION SUCCEEDED, the application client shall assume the command was delivered successfully. If Received Task Management Function Executed () returns a Service Response of FUNCTION COMPLETE, and Command Complete Received () has not yet been invoked a second time for the command in question (e.g., indicating a RESPONSE frame arrived for the command before the QUERY TASK was processed), the application client shall assume the command was not delivered successfully and may reuse the tag. The application client should call Send SCSI Command () again with identical arguments.

After a Command Complete Received () or Received Task Management Function Executed () call, an application client shall not reuse the tag until it determines the tag is no longer in use by the logical unit (e.g., the ACK for the RESPONSE frame was seen by the SSP target port). Examples of ways the application client may determine that a tag may be used are:

a) receiving another frame in the same connection;
b) receiving a DONE (NORMAL) or DONE (CREDIT TIMEOUT) in the same connection; or
c) receiving a DONE (ACK/NAK TIMEOUT) in the same connection, then running a QUERY TASK task management function to confirm that the tag is no longer active in the logical unit.

10.2.3 Device server error handling

If the SCSI target device performs tag checking and an SSP target port calls SCSI Command Received () with a tag already in use by another SCSI command (i.e., an overlapped command) in any logical unit, the task router and device server(s) shall abort all task management functions received on that I_T nexus and shall respond to the overlapped command as defined in SAM-3.

If a device server calls Receive Data-Out () and receives a Delivery Result set to a value in table 15, it shall terminate the command with CHECK CONDITION status with the sense key set to ABORTED COMMAND and the additional sense code set as indicated in table 15.

Table 15 — Delivery Result to additional sense code mapping

Delivery Result	Additional sense code
DELIVERY FAILURE - DATA OFFSET ERROR	DATA OFFSET ERROR
DELIVERY FAILURE - TOO MUCH WRITE DATA	TOO MUCH WRITE DATA
DELIVERY FAILURE - INFORMATION UNIT TOO SHORT	INFORMATION UNIT TOO SHORT
DELIVERY FAILURE - ACK/NAK TIMEOUT	ACK/NAK TIMEOUT
DELIVERY FAILURE - NAK RECEIVED	NAK RECEIVED
DELIVERY FAILURE - INITIATOR RESPONSE TIMEOUT	INITIATOR RESPONSE TIMEOUT

10.2.4 Task router and task manager error handling

If the SCSI target device performs tag checking and:

a) an SSP target port calls SCSI Command Received () with a tag already in use by a SCSI task management function in any logical unit; or
b) an SSP target port calls calls Task Management Request Received () with a tag already in use by a SCSI command or SCSI task management function in any logical unit,

the task router and task manager(s) shall:

a) abort all commands received on that I_T nexus;
b) abort all task management functions received on that I_T nexus; and
c) call Task Management Function Executed () with the Service Response set to FUNCTION REJECTED - Overlapped Tag Attempted (i.e., requesting that the target port set the DATAPRES field to RESPONSE_DATA and the RESPONSE CODE field to OVERLAPPED TAG ATTEMPTED).

10.2.4 SCSI transport protocol event notifications

Table 16 lists the SCSI transport protocol event notifications supported by this standard.

Table 16 — SCSI transport protocol events

Event notification	SAS SSP implementation
Transport Reset	Receipt of a hard reset sequence (see 4.4.2)
Nexus Loss	Receipt of specific OPEN_REJECTs for a specific time period~~, or duplicate tag condition causing an I_T nexus loss event~~ (see 4.5).