

Date: April 15, 2003

To: T10 Committee (SCSI)

From: George Penokie (IBM/Tivoli)

Subject: SAS: Logical Unit Communication timer

## **1 Overview**

This proposal adds in a timer to assure there is no hang in the ST\_TTS4: Receive\_Data\_Out state in the ST\_TTS state machine.

### **1.0.0.1 ST\_T (transport layer for SSP target ports) state machines**

#### **1.0.0.1.1 ST\_T state machines overview**

The ST\_T state machines are as follows:

- a) ST\_TFR (target frame router) state machine (see 1.0.0.1.2); and
- b) ST\_TTS (target transport server) state machine (see 1.0.0.1.3).

Figure 1 shows the ST\_T state machines.

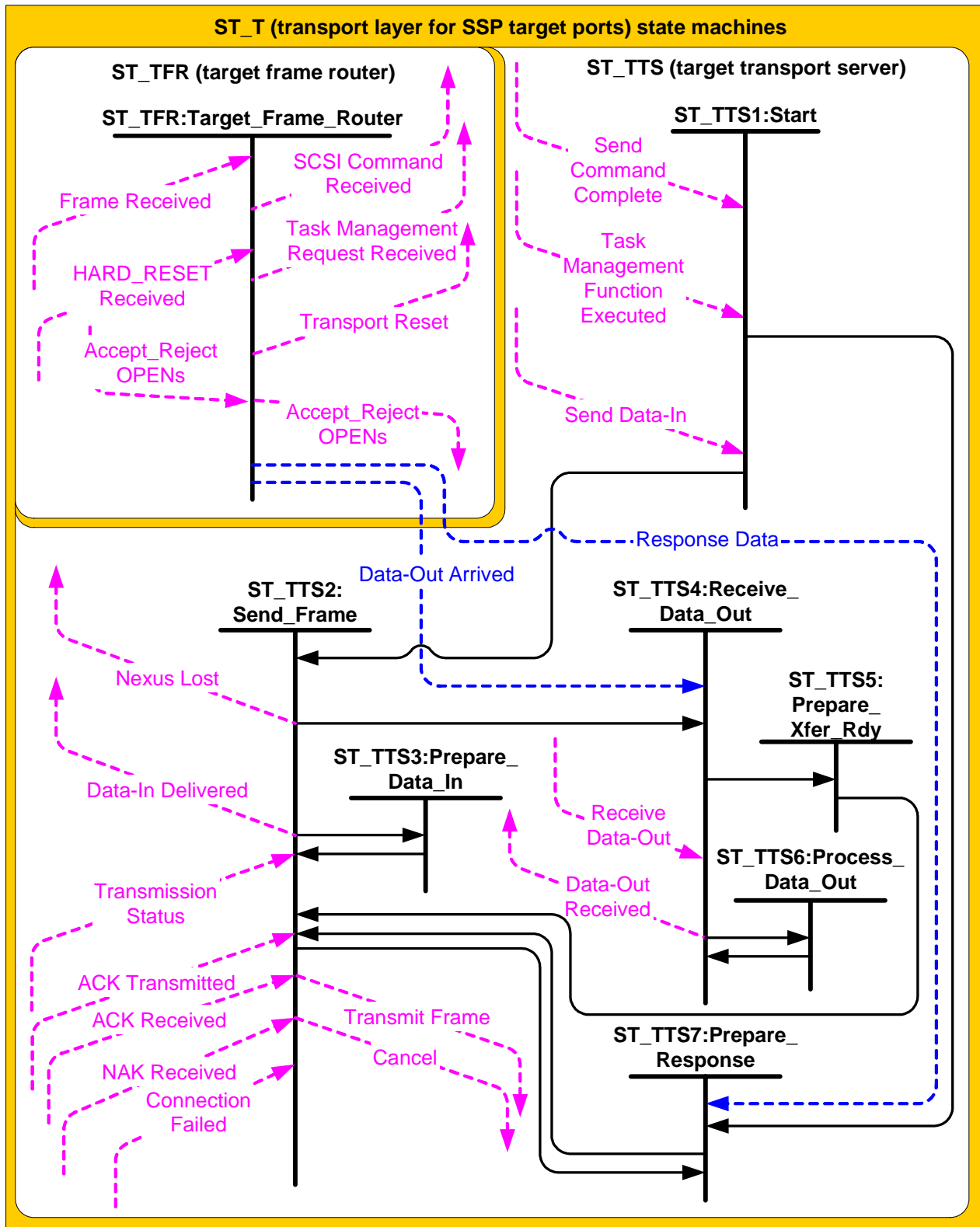


Figure 1 — ST\_T (transport layer for SSP target ports) state machines

#### **1.0.0.1.2 ST\_TFR (target frame router) state machine**

The ST\_TFR state machine receives confirmations from the port layer and sends a transport protocol service indication to the SCSI application layer or a message to the ST\_TTS state machine. This state machine also receives Accept\_Reject OPENs requests from the application layer and sends corresponding requests to the port layer.

This state machine consists of one state.

This state machine shall be started when:

- a) an Accept\_Reject OPENs request is received;
- b) a Frame Received confirmation is received; or
- c) a HARD\_RESET Received confirmation is received.

If this state machine was started as the result of receiving 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. This state machine shall terminate after sending an Accept\_Reject OPENs request to the port layer.

If this state machine was started as the result of a Frame Received (Unsuccessful) confirmation, then this state machine shall terminate.

If this state machine was started as the result of receiving 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 108). If the frame type is not COMMAND, TASK, or DATA, then this state machine shall discard the frame and terminate.

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 and terminate.

This state machine may check that reserved fields in the frame are zero. If any reserved fields are not zero, then this state machine may send a Response Data (Invalid Frame) message to the ST\_TTS7:Prepare\_Response state including the logical unit number and tag.

NOTE 1 - This check only applies to reserved fields defined in the SSP frame formats (e.g. formats defined in this clause), not reserved fields within the CDB in a COMMAND frame. Handling checking of reserved fields in a CDB is described in SAM-3.

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

If the frame type is DATA, and the tag does not match a tag for an outstanding data-out command, then this state machine shall discard the frame and terminate.

If the frame type is COMMAND, then this state machine shall check the length of the information unit. If the length of the information unit is not correct (see 9.2.2.2), then this state machine shall send a Response Data (Invalid Frame) message to the ST\_TTS7:Prepare\_Response state including the logical unit number and tag.

If the frame type is TASK, then this state machine shall check the length of the information unit. If the length of the information unit is not correct (see 9.2.2.2), then this state machine shall send a Response Data (Invalid Frame) message to the ST\_TTS7:Prepare\_Response state including the logical unit number and tag.

If the frame type is COMMAND and the length of the information unit is correct, then this state machine shall send a SCSI Command Received transport protocol service indication to the SCSI application layer.

If the frame type is TASK, then this state machine shall check the logical unit number. If there is no logical unit at the specified logical unit number, then this state machine shall send a Response Data (Invalid Logical Unit Number) message to the ST\_TTS7:Prepare\_Response state including the logical unit number and tag.

If the frame type is COMMAND or TASK, then this state machine may check the target port transfer tag. If target port transfer tag is invalid, then this state machine may send a Response Data (Invalid Frame) message to the ST\_TTS7:Prepare\_Response state including the logical unit number and tag.

If the frame type is TASK and the length of the information unit is correct, then this state may check if the tag conflicts with an existing tag (i.e., an existing command or task management function). If the tag is checked and it conflicts, this state shall send a Response Data (Invalid Frame) message to the ST\_TTS7:Prepare\_Response state including the logical unit number and tag. If it does not check the tag or the tag does not conflict, this state machine shall send a Task Management Request Received transport protocol service indication to the SCSI application layer. If the frame type is DATA, then this state machine shall send a Data-Out Arrived message to the ST\_TTS4:Receive\_Data\_Out state. Each indication or message shall contain the content of the SAS frame.

If this state machine was started as the result of receiving a HARD\_RESET Received confirmation, then this state machine shall send a Transport Reset event notification to the SCSI application layer and terminate.

This state machine shall terminate after sending a message, transport protocol service indication, or event notification.

### 1.0.0.1.3 ST\_TTS (target transport server) state machine

#### 1.0.0.1.3.1 ST\_TTS state machine overview

The ST\_TTS state machine performs the following functions:

- a) processes and sends transport protocol service confirmations to the SCSI application layer;
- b) receives and processes transport protocol service requests and responses from the SCSI application layer; and
- c) communicates with the port layer via requests and confirmations regarding frame transmission.

This state machine consists of the following states:

- a) ST\_TTS1:Start (see 1.0.0.1.3.2);
- b) ST\_TTS2:Send\_Frame (see 1.0.0.1.3.3);
- c) ST\_TTS3:Prepare\_Data\_In (see 1.0.0.1.3.4);
- d) ST\_TTS4:Receive\_Data\_Out (see 1.0.0.1.3.5);
- e) ST\_TTS5:Prepare\_Xfer\_Rdy (see 9.2.6.3.3.6);
- f) ST\_TTS6:Process\_Data\_Out (see 9.2.6.3.3.7); and
- g) ST\_TTS7:Prepare\_Response (see 9.2.6.3.3.8).

This state machine shall be started in the ST\_TTS1:Start state ~~when~~ if one of the following is received:

- a) a Send Data-In transport protocol service request;
- b) a ~~Receive Data-Out~~ Task Management Function Executed transport protocol service ~~request; response; or~~
- c) a ~~Task Management Function Executed~~ Send Command Complete transport protocol service ~~response; or~~
- ~~d) a Send Command Complete transport protocol service response.~~

This state machine shall be started in the ST\_TTS4:Receive\_Data\_Out state if:

- a) a Receive Data-Out transport protocol service request is received; or
- b) ~~This state machine shall be started in the ST\_TTS4:Receive\_Data\_Out state when~~ a Data-Out Arrived message is received, first burst is enabled, and this state machine is not already running.

This state machine shall be started in the ST\_TTS7:Prepare\_Response state when a Response Data message is received and this state machine is not already running.

#### 1.0.0.1.3.2 ST\_TTS1:Start state

##### 1.0.0.1.3.2.1 State description

The request or response that caused this state machine to be started includes the following to be used in any OPEN address frames required to service the request or response:

- a) connection rate;
- b) initiator connection tag; and
- c) destination SAS address.

A Send Data-In transport protocol service request also includes the following:

- a) logical unit number;
- b) tag;
- c) device server buffer (e.g., starting logical block address); and
- d) request byte count (e.g., transfer length).

~~A Receive Data-Out transport protocol service request also includes the following:~~

- ~~a) logical unit number;~~
- ~~b) tag;~~
- ~~c) device server buffer (e.g., starting logical block address); and~~
- ~~d) request byte count (e.g., transfer length).~~

A Task Management Function Executed transport protocol service response or Send Command Complete transport protocol service response also includes the following:

- a) logical unit number;
- b) tag;
- c) task management function; and
- d) tag of task to be managed.

##### 1.0.0.1.3.2.2 Transition ST\_TTS1:Start to ST\_TTS2:Send\_Frame

~~This transition shall occur after receiving a Send Data-In transport protocol service request.~~

This transition shall occur after receiving a Send Data-In transport protocol service request and shall include all the information received in the Send Data-In transport protocol service request as arguments.

##### 1.0.0.1.3.2.3 Transition ST\_TTS1:Start to ~~ST\_TTS4~~ST\_TTS7:Receive\_Data\_OutPrepare\_Response

~~This transition shall occur after receiving a Receive Data-Out transport protocol service request.~~

##### ~~1.0.0.1.3.2.4 Transition ST\_TTS1:Start to ST\_TTS7:Prepare\_Response~~

~~This transition shall occur after receiving a Task Management Function Executed transport protocol service response or a Send Command Complete transport protocol service response.~~

This transition shall occur after receiving a Task Management Function Executed transport protocol service response or a Send Command Complete transport protocol service response and shall include all the information received in the Task Management Function Executed transport protocol service response or Send Command Complete transport protocol service response as arguments.

#### 1.0.0.1.3.3 ST\_TTS2:Send\_Frame state

##### 1.0.0.1.3.3.1 State description

If this state is entered from the ST\_TTS3:Prepare\_Data\_In state for transmission of a 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\_TTS5:Prepare\_Xfer\_Rdy state for transmission of an XFER\_RDY frame and this state has received an ACK Transmitted confirmation for each DATA frame previously received (i.e.,

received with a Data-Out Arrived message), 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 and this state has received an ACK Transmitted confirmation for each DATA frame previously received (i.e., received with a Data-Out Arrived message), then this state shall send a Transmit Frame (Interlocked) request to the port layer.

NOTE 2 - 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 to be used for any OPEN address frame:

- a) the 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), this state shall send a Nexus Lost confirmation to the SCSI application layer.

If the confirmation is Transmission Status (Frame Transmitted) confirmation, then this state machine shall expect to receive one of the following confirmations for the frame:

- a) ACK Received;
- b) NAK Received; or
- c) Connection Failed.

If the frame transmitted was an XFER\_RDY frame or a RESPONSE frame, then the state machine shall wait to receive an ACK Received, NAK Received, or Connection Failed confirmation before transitioning from this state. If the frame transmitted was a DATA frame, then the state machine may transition to ST\_TTS3:Prepare\_Data\_In as described in 1.0.0.1.3.3.3.

This state shall send a Data-In Delivered (Delivery Result = DELIVERY SUCCESSFUL) transport protocol service confirmation to the SCSI application layer if:

- a) for a DATA frame, this state receives a Transmission Status (Frame Transmitted) confirmation followed by an ACK Received confirmation for each of the DATA frames transmitted and the number of bytes moved for the Send Data-In transport protocol service request equals the Request Byte Count; or
- b) for a RESPONSE frame, this state receives a Transmission Status (Frame Transmitted) confirmation followed by an ACK Received confirmation.

This state shall send a Data-In Delivered (Delivery Result = DELIVERY FAILURE - NAK RECEIVED) transport protocol service confirmation to the SCSI application layer if the received transmission status confirmation message for a DATA or XFER\_RDY frame was Transmission Status (Frame Transmitted) followed by a confirmation of NAK Received.

This state shall send a Data-In Delivered (Delivery Result = DELIVERY FAILURE - ACK/NAK TIMEOUT) transport protocol service confirmation to the SCSI application layer if the received transmission status confirmation message for a DATA or XFER\_RDY frame was Transmission Status (Frame Transmitted) followed by a confirmation of Connection Failed (ACK/NAK Timeout) or Connection Failed (Connection Lost Without ACK/NAK).

A Data-In Delivered transport protocol service confirmation to the SCSI application layer confirmation shall include the following:

- a) any argument received from the port layer (e.g., Transmission Status (Frame Transmitted) or Service Delivery Subsystem Failure; and
- b) I\_T\_L\_x nexus information (i.e., destination SAS address and tag).

This state machine shall terminate after sending the Data-In Delivered confirmation.

This state may also send a Cancel request to the port layer to cancel a previous Transmit Frame request. A Cancel request shall include the following arguments:

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

This state machine terminates upon receipt of a Transmission Status (Cancel Acknowledge) confirmation.

#### **1.0.0.1.3.3.3 Transition ST\_TTS2:Send\_Frame to ST\_TTS3:Prepare\_Data\_In**

If this state machine was started as the result of receiving a Send Data-In transport protocol service request, the specified values are included with the request, and this state has received an ACK Transmitted confirmation for the COMMAND frame, then this state shall transition to the ST\_TTS3:Prepare\_Data\_In state.

NOTE 3 - The COMMAND frame rule ensures that ports do not send a read DATA frame until the ACK has been transmitted for the COMMAND frame.

If this state receives a Transmission Status (Frame Transmitted) confirmation for a DATA frame and the number of bytes moved for the Send Data-In transport protocol service request is less than the Request Byte Count, then this state shall transition to the ST\_TTS3:Prepare\_Data\_In state.

#### **1.0.0.1.3.3.4 Transition ST\_TTS2:Send\_Frame to ST\_TTS4:Receive\_Data\_Out**

This transition shall occur after receiving a Transmission Status (Frame Transmitted) confirmation and an ACK Received confirmation for an XFER\_RDY frame.

#### **1.0.0.1.3.3.5 Transition ST\_TTS2:Send\_Frame to ST\_TTS7:Prepare\_Response**

This transition shall occur after receiving one of the following confirmations for a RESPONSE frame:

- a) Transmission Status with an argument other than (Frame Transmitted);
- b) NAK Received; or
- c) Connection Failed.

#### **1.0.0.1.3.4 ST\_TTS3:Prepare\_Data\_In state**

##### **1.0.0.1.3.4.1 State description**

This state fetches the data from the Device Server Buffer and constructs a DATA frame. This state shall use the logical unit number and tag received from the SCSI application layer to construct the frame.

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

- a) frame type;
- b) hashed destination SAS address;
- c) hashed source SAS address;
- d) retransmit bit set to zero;
- e) number of fill bytes;
- f) fill bytes;
- g) data offset;
- h) data.

##### **1.0.0.1.3.4.2 Transition ST\_TTS3:Prepare\_Data\_In to ST\_TTS2:Send\_Frame**

This transition shall occur after constructing a DATA frame.

### 1.0.0.1.3.5 ST\_TTS4:Receive\_Data\_Out state

#### 1.0.0.1.3.5.1 State description

If a Receive Data-Out transport protocol service request caused this state machine to be started it includes the following to be used in any OPEN address frames required to service the request:

- a) connection rate;
- b) initiator connection tag;
- c) destination SAS address;
- d) logical unit number;
- e) tag;
- f) device server buffer (e.g., starting logical block address); and
- g) request byte count (e.g., transfer length).

If a Data-Out Arrived message caused this state machine to be started (i.e., first burst data occurred) and a Receive Data-Out transport protocol service request has not been received this state shall wait until the Receive Data-Out transport protocol service request occurs.

If this state was ~~entered~~ started as the result of receiving a Data-Out Arrived message, then this state shall after receiving the Receive Data-Out transport protocol service request:

- a) check the target transport tag value in the DATA frame. If the value is incorrect, then this state shall send a Data-Out Received (Delivery Result = DELIVERY FAILURE - INVALID TARGET PORT TRANSFER TAG RECEIVED) transport protocol service confirmation to the SCSI application layer. This confirmation shall include the tag. This state machine shall terminate after sending the confirmation;
- b) check the length of the data. If an XFER\_RDY frame was sent for the data (i.e., it is not first burst data) and the length of the data exceeds that specified by the XFER\_RDY frame that requested the data, then this state shall send a Data-Out Received (Delivery Result = DELIVERY FAILURE - TOO MUCH WRITE DATA) transport protocol service confirmation to the SCSI application layer. This confirmation shall include the tag. This state machine shall terminate after sending the confirmation;
- c) check the length of the data. If the length of the data is zero, then this state shall send a Data-Out Received (Delivery Result = DELIVERY FAILURE - INFORMATION UNIT TOO SHORT) transport protocol service confirmation to the SCSI application layer. This confirmation shall include the tag. This state machine shall terminate after sending the confirmation; and
- d) check the data offset. If the data offset was not expected, then this state shall send a Data-Out Received (Delivery Result = DELIVERY FAILURE - DATA OFFSET ERROR) transport protocol service confirmation to the SCSI application layer. This confirmation shall include the tag. This state machine shall terminate after sending the confirmation.

If this state is entered from the ST\_TTS2:Send\_Frame state, then this state ~~shall wait for a Data-Out Arrived message from the ST\_TFR state machine.~~ shall:

- a) wait for a Data-Out Arrived message;
- b) initializes the Logical Unit Communication timer to the value in the I T NEXUS LOSS TIME field in the Protocol-Specific Port Control mode page for the SSP target port (see 10.2.6.2); and
- c) start the Logical Unit Communication timer.

On receipt of a Data-Out Arrived message this state shall:

- a) initialize the Logical Unit Communication timer to the value in the I T NEXUS LOSS TIME field; and
- b) start the Logical Unit Communication timer.

If the Logical Unit Communication timer expires this state shall send a Data-Out Received (Delivery Result = DELIVERY FAILURE - LOGICAL UNIT COMMUNICATION TIME-OUT) transport protocol service confirmation to the SCSI application layer. This confirmation shall include the tag. This state machine shall terminate after sending the confirmation.

If this state is entered from the ST\_TTS6:Process\_Data\_Out state and the number of bytes moved for the Receive Data-Out transport protocol service request is less than the Request Byte Count, then this state shall wait for a Data-Out Arrived message.



If this state is entered from the ST\_TTS6:Process\_Data\_Out state and number of bytes moved for the Receive Data-Out transport protocol service request equals the Request Byte Count, then this state shall send a Data-Out Received (Delivery Result = DELIVERY SUCCESSFUL) transport protocol service confirmation to the SCSI application layer. This confirmation shall include the tag. This state machine shall terminate after sending the confirmation.

#### 1.0.0.1.3.5.2 Transition ST\_TTS4:Receive\_Data\_Out to ST\_TTS5:Prepare\_Xfer\_Rdy

This transition shall occur if:

- a) this state was started as a result of receiving a Receive Data-Out transport protocol service request:  
or
- b) ~~This transition shall occur if~~ this state was ~~entered~~ started as a result of receiving a ~~Receive~~ Data-Out transport protocol service request ~~if the data requested is not entirely contained within the first burst data range for the command~~ Arrived message, ~~or if all the first burst data has been consumed~~ consumed, and the ~~last Receive Data-Out transport protocol service~~ request byte count has not been satisfied.

#### 1.0.0.1.3.5.3 Transition ST\_TTS4:Receive\_Data\_Out to ST\_TTS6:Process\_Data\_Out

After receiving a Data-Out Arrived message, if the target transport tag value matches the value sent with the corresponding XFER\_RDY frame, and the length of the data does not exceed that specified by the XFER\_RDY frame that requested the data, then this state shall transition to the ST\_TTS6:Process\_Data\_Out state.

#### 1.0.1 Device server error handling

If a device server calls Receive Data-Out () and receives a Delivery Result set to a value in table 1, it shall terminate the command with a CHECK CONDITION status with a sense key of ABORTED COMMAND and an additional sense code as indicated by table 1.

**Table 1 — Delivery Result to additional sense code mapping**

Delivery Result	Additional sense code
DELIVERY FAILURE - INVALID TARGET PORT TRANSFER TAG	INVALID TARGET PORT TRANSFER TAG RECEIVED
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
<u>DELIVERY FAILURE - LOGICAL UNIT COMMUNICATION TIME-OUT</u>	<u>LOGICAL UNIT COMMUNICATION TIME-OUT</u>