Date: March 07, 2007

To: T10 Committee (SCSI)

From: George Penokie (IBM)

Subject: SAS-2: Transport level read fixes

1 Overview

During the comparison between SAS-2 rev 8 and 06-470r5 there were some differences that need to be addressed. There are also some omissions in 06-470r5 that were discovered during incorporation into SAS-2 that are addressed in this proposal. These were pointed out in SAS-2 rev 8 editor notes 46, 47, and 48. Note that this proposal only addresses the read data path. The write data path is not addressed in this proposal.

2 ST_ITS6:Receive_Data_In state

9.2.6.2.2.2 Processing transport protocol service requests

If this state machine receives a Send SCSI Command transport protocol service request then this state machine shall send a Request (Send Command) message with Command arguments and Buffer arguments to the ST_ITS state machine for the specified tag.

The following is the list of Command arguments:

a) connection rate;
b) initiator connection tag;
c) destination SAS address;
d) source SAS address set to the SAS address of the SSP initiator port;
e) tag;
f) logical unit number;
g) task priority;
h) task attribute;
i) additional CDB length;
j) CDB; and
k) additional CDB bytes, if any.

The following is the list of Application Client Buffer arguments:

a) data-in buffer size;
b) data-out buffer; and
c) data-out buffer size.

...

9.2.6.2.3.1 ST_ITS state machine overview

The ST_ITS state machine performs the following functions:

a) receives and processes messages from the ST_IFR state machine;
b) sends messages to the ST_IFR state machine;
c) sends request to the port layer regarding frame transmission;
d) receives confirmations from the port layer regarding frame transmission; and
e) receives HARD_RESET Received confirmations from the port layer.

This state machine consists of the following states:

a) ST_ITS1:Initiator_Start state (see 9.2.6.2.3.2) (initial state);
b) ST_ITS2:Initiator_Send_Frame state (see 9.2.6.2.3.3);
c) ST_ITS3:Prepare_Command state (see 9.2.6.2.3.4);
d) ST_ITS4:Prepare_Task state (see 9.2.6.2.3.5);
e) ST_ITS5:Prepare_Data_Out state (see 9.2.6.2.3.6); and

f) ST_ITS6:Receive_Data_In state (see 9.2.6.2.3.7).

This state machine shall start in the ST_ITS1:Initiator_Start state after power on.

If this state machine receives a HARD_RESET Received confirmation, then this state machine shall transition to the ST_ITS1:Initiator_Start state.

This state machine shall maintain the state machine variables defined in table 1.

Table 1 — ST_ITS state machine variables

State machine variable	Description
Data-In Buffer Offset	Current offset in the data-in application client buffer for read data
Data-Out Buffer Offset	Current offset in the data-out application client buffer for write data
Previous Requested Offset	Application client data offset from the last XFER_RDY frame received
Previous Write Data Length	Write data length from the last XFER_RDY frame received

This state machine shall maintain the state machine arguments defined in table 2.

Table 2 — ST_ITS state machine arguments

State machine argument	Description
Command	Consists of the Command arguments received in the Request (Send Command) message
Task	Consists of the arguments received in the Request (Send Task) message
Xfer_Rdy	Consists of the arguments received in the XFER_RDY Arrived message
Data-Out Buffer	The location of the write data application client buffer
Data-Out Buffer Size	The size in bytes of the write data application client buffer
Data-In Buffer Size	The size in bytes of the read data application client buffer

The below editors note suggests that the Data-in Buffer Size needs to be adjusted. But that value should never be adjusted as it is an argument and is only used to make sure none of the read data frames that are received will put data beyond the buffer allocated by the application.

Editor's Note 46: if Buffer Offset is an app client buffer based value, then Data-In Buffer Size needs to be adjusted.

The flow chart also needs to be adjusted as there are some errors in it (see figure 1).

10 ST_T state machines

9.2.6.3.3 ST_TTS (target transport server) state machine

9.2.6.3.3.1 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 9.2.6.3.3.2) (initial state);
b) ST_TTS2:Target_Send_Frame (see 9.2.6.3.3.3);
c) ST_TTS3:Prepare_Data_In (see 9.2.6.3.3.3);
d) ST_TTS4:Prepare_Xfer_Rdy (see 9.2.6.3.3.5);
e) ST_TTS5:Receive_Data_Out (see 9.2.6.3.3.6); and
f) ST_TTS6:Prepare_Response (see 9.2.6.3.3.7).

This state machine shall start in the ST_TTS1:Target_Start state after power on.

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

The state machine shall maintain the state machine variables defined in table 3.

Table 3 — ST_TTS state machine variables

State machine variable	Description
Read Data Offset	Offset into the application client buffer and the device server buffer for read data
Balance Point Read Data Offset	Offset into the application client buffer and the device server buffer for read data of last point at which the number of Transmission Status (ACK Received) confirmations or arguments was equal to the number of transmitted read DATA frames
Read Data Frames Transmitted	The number of Transmission Status (Frame Transmitted) confirmations received for read DATA frames
Read Data Frames ACKed	The number of Transmission Status (ACK Received) confirmations received for read DATA frames
Requested Write Data Offset	Device server requested offset in the application client buffer and the device server buffer for write data
Requested Write Data Length	Amount of write data requested by the device server from the application client buffer

This state machine shall maintain the state machine arguments defined in table 4.

Table 4 — ST_TTS state machine arguments

State machine argument	Description
Data-In	The Data-In arguments received in the Request (Send Data-In) message (see 9.2.6.3.2.3)
Data-Out	The Data-Out arguments received in the Request (Receive Data-Out) message (see 9.2.6.3.2.3)

9.2.6.3.3.2 ST_TTS1:Target_Start state

9.2.6.3.3.2.1 State description

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

Upon entry into this state, this state shall:

a) set the Read Data Offset state machine variable to the Data-In Application Client Buffer Offset argument;
b) set the Balance Point Read Data Offset state machine variable to the Data-In Application Client Buffer Offset argument;
c) set the Read Data Frames Transmitted ~~Count~~ state machine variable to zero;
d) set the Read Data Frames ACKed state machine variable to zero; and
e) set the Requested Write Data Offset state machine variable to zero.

If this state was entered without an Enable First Burst argument, then the Requested Write Data Length state machine variable shall be set to the Request Byte Count Data-Out state machine argument.

If this state was entered with an Enable First Burst argument, then the Requested Write Data Length state machine variable shall be set to the First Burst Size argument.

...

9.2.6.3.3.3 ST_TTS3:Prepare_Data_In state

9.2.6.3.3.3.1 State description

This state retrieves the data from the Data-In Device Server Buffer argument and constructs a read DATA frame.

This state shall construct a read DATA frame using the Data-In arguments as follows:

a) FRAME TYPE field set to 01h (i.e., DATA frame);
b) HASHED DESTINATION SAS ADDRESS field set to the hashed value of the Data-In Destination SAS Address argument;
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 set as specified in this subclause;
g) NUMBER OF FILL BYTES field set to the number of fill bytes needed for the specified read data;
h) TAG field set to the Data-In Tag argument;
i) TARGET PORT TRANSFER TAG field set to zero;
j) DATA OFFSET field set as specified in this subclause;
k) in the information unit, DATA field set as specified in this subclause; and
l) fill bytes, if required.

If this state is entered without a Retry argument then this state shall:

a) set the CHANGING DATA POINTER bit set to zero;
b) set the DATA OFFSET field to the Read Data Offset state machine variable;
c) set the Read Data Frames Transmitted state machine variable to zero;
d) set the Read Date Frames ACKed state machine variable to zero; and
e) in the information unit, set the DATA field to the information that starts at the location in the Data-In Device Server Buffer argument pointed to by the Read Data Offset state machine variable ~~minus the Data-In Application Client Buffer Offset argument,~~ containing the amount of data that is the lesser of:
   A) the Data-In Request Byte Count argument minus the Read Data Offset state machine variable; and
   B) the maximum size of the read Data information unit for this Data-In request.

If this state is entered with a Retry argument then this state shall either:

a) set the CHANGING DATA POINTER bit in the frame to one;

b) set the DATA OFFSET field to the Balance Point Read Data Offset state machine variable;
c) set the Read Data Offset state machine variable to the Balance Point Read Data Offset state machine variable;
d) set the Read Data Frames Transmitted state machine variable to zero;
e) set the Read Date Frames ACKed state machine variable to zero; and
f) in the information unit, set the DATA field to the information that starts at the location in the Data-In Device Server Buffer argument pointed to by the Balance Point Read Data Offset state machine variable ~~minus the Data-In Application Client Buffer Offset argument,~~ containing the amount of data that is the lesser of:
   A) the Data-In Request Byte Count argument minus the Balance Point Read Data Offset state machine variable; and
   B) the maximum size of the read Data information unit for this Data-In request;

or:

a) set the CHANGING DATA POINTER bit in the frame to one;
b) set the DATA OFFSET field to the Data-In Application Client Buffer Offset argument;
c) set the Read Data Offset state machine variable to the Data-In Application Client Buffer Offset argument; and
d) in the information unit, set the DATA field to the information that starts at the location in the specified device server buffer pointed to by the Data-In Application Client Buffer Offset argument ~~Balance Point Read Data Offset state machine variable~~ and containing the amount of data that is the lesser of:
   A) the Data-In Request Byte Count argument; and
   B) the maximum size of the read Data information unit for this Data-In request.

3 ST_ITS transport layer read data flowcharts

DATA OFFSET field = Contains the first offset location into read data buffer for the current DATA information unit
Data-In Buffer Offset = Offset into read data buffer for the last received data frame.
Data-In Buffer Size = The number of bytes to be read as requested by the application client.

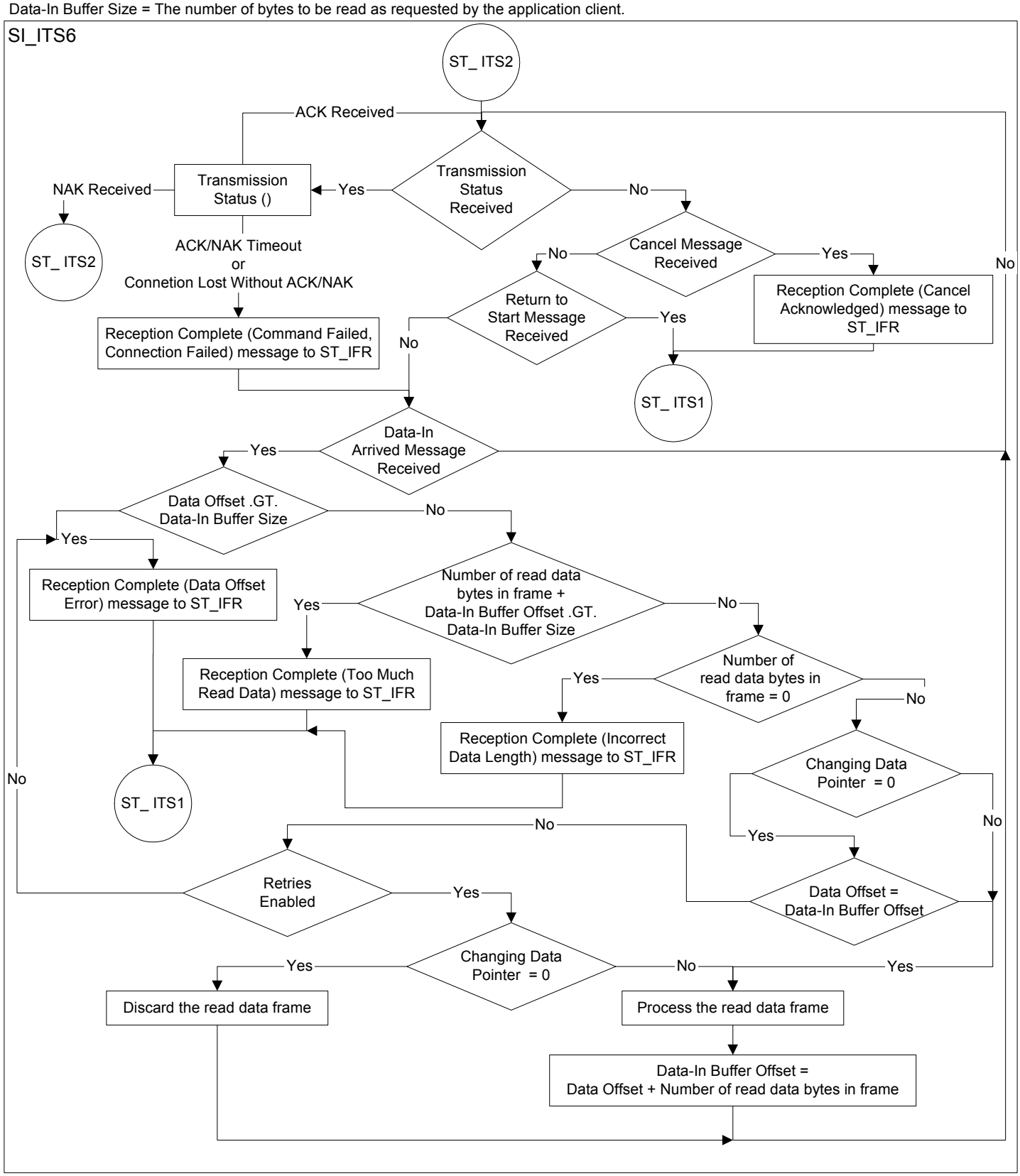

Figure 1 — Representation of transport layer (i.e., ST_ITS6) read data operation

4 ST_TTS transport layer read data flowcharts

Read Data Offset = Offset into application client read data buffer and the device server read data buffer
Read Data Frames Transmitted = The number of Transmission Status (Frame Transferred) confirmations received
Read Data Frames ACKed = The number of Transmission Status (ACK Received) confirmation received.
Balance Point Read Data Offset = Offset into the appliction client read data buffer and the device server read data buffer for last data frame that the number of frames transmitted = number ACKs received
Data-In Request Byte Count = The number of bytes requested to be transferred. Set by the device server.

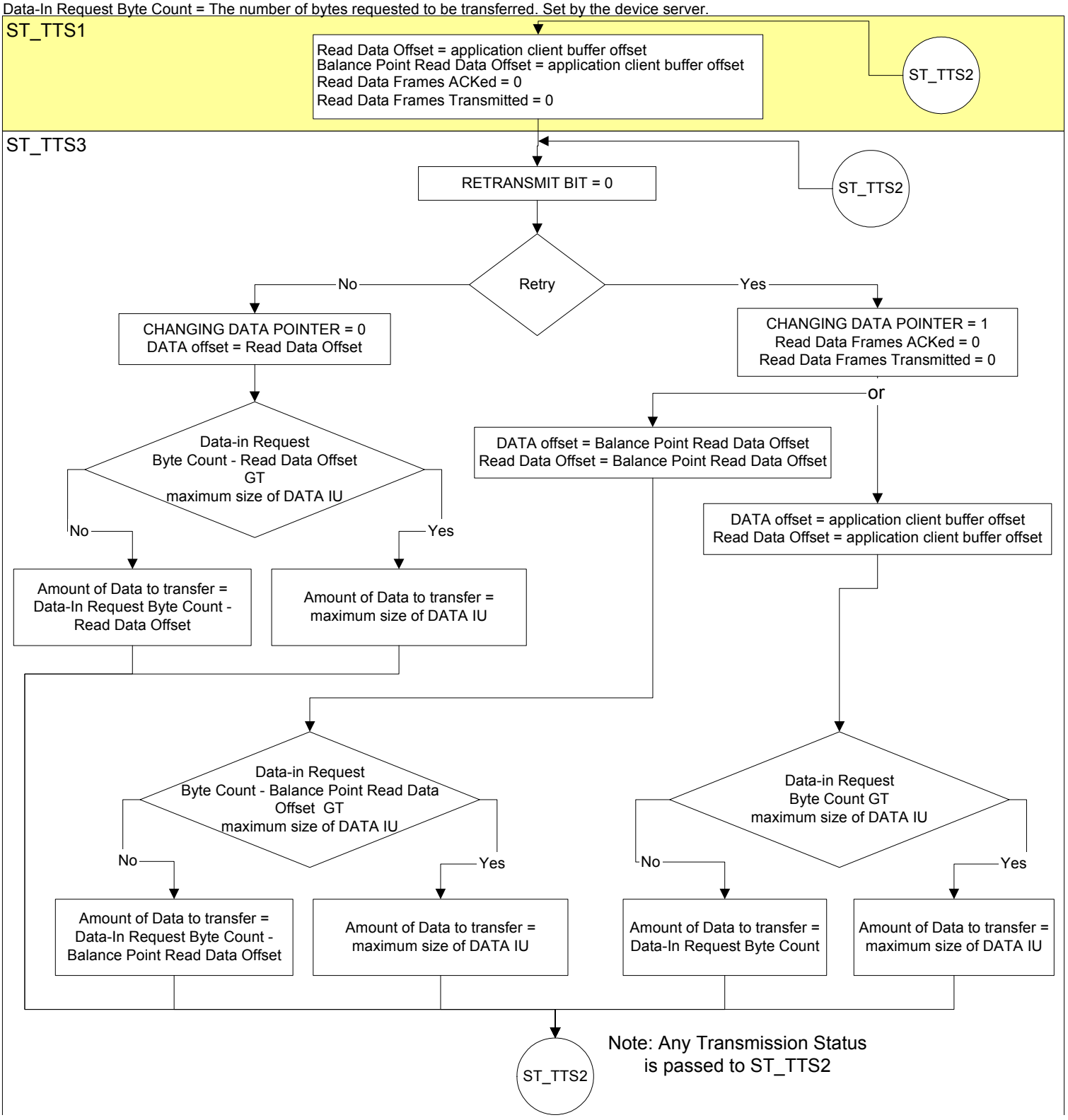

Figure 2 — Representation of transport layer (i.e., ST_TTS1 and ST_TTS3) read data operation

Read Data Frames Transmitted = The number of Transmission Status (Frame Transferred) confirmations received
Read Data Frames ACKed = The number of Transmission Status (ACK Received) confirmation received.
Balance Point Read Data Offset = Offset into the appliction client read data buffer and the device server read data buffer for last data frame that the number of frames transmitted = number ACKs received
Data-In Request Byte Count = The number of bytes requested to be transferred. Set by the device server.

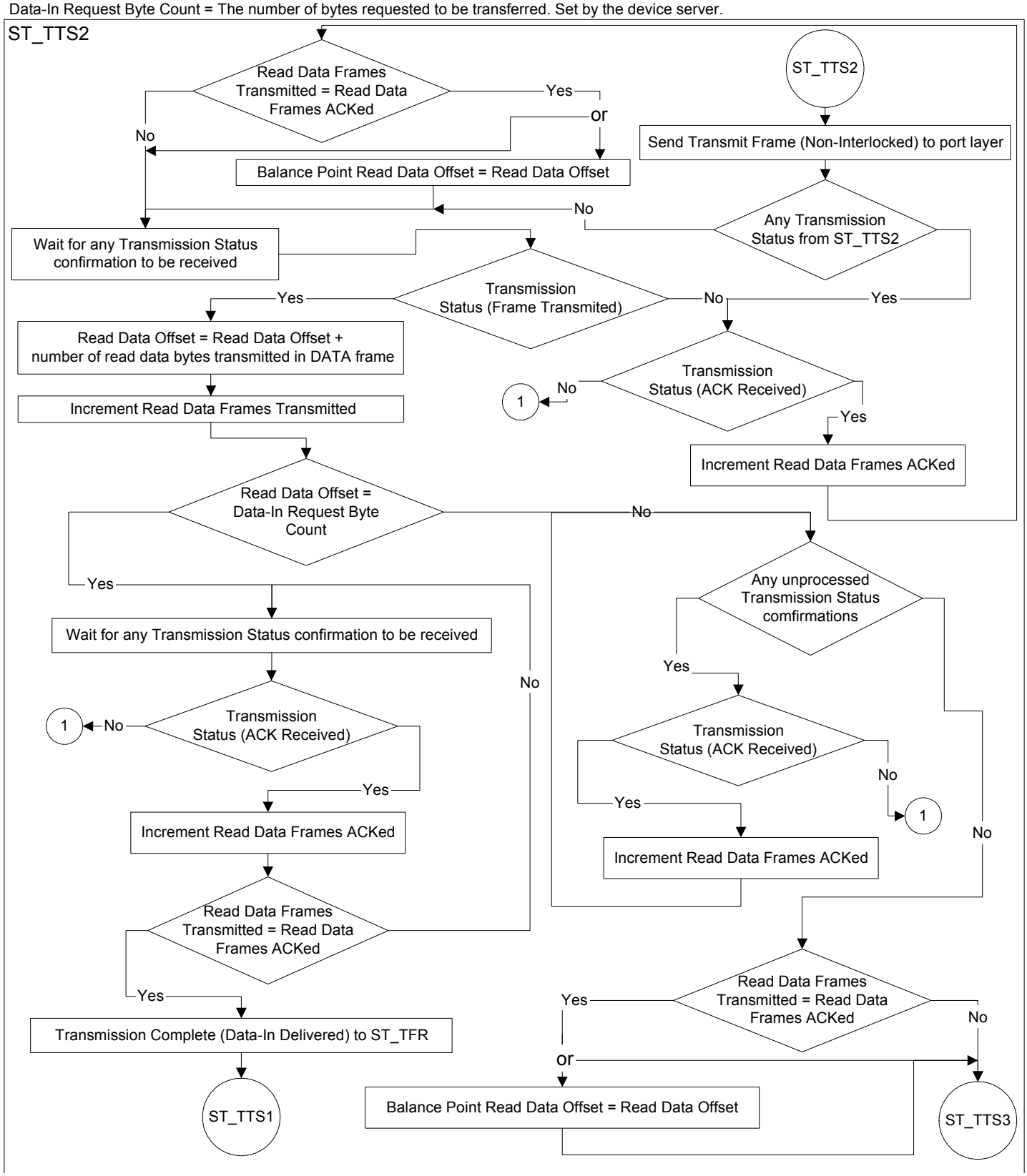

Figure 3 — Representation of transport layer (i.e., ST_TTS2) read data operation (part 1 or 2)

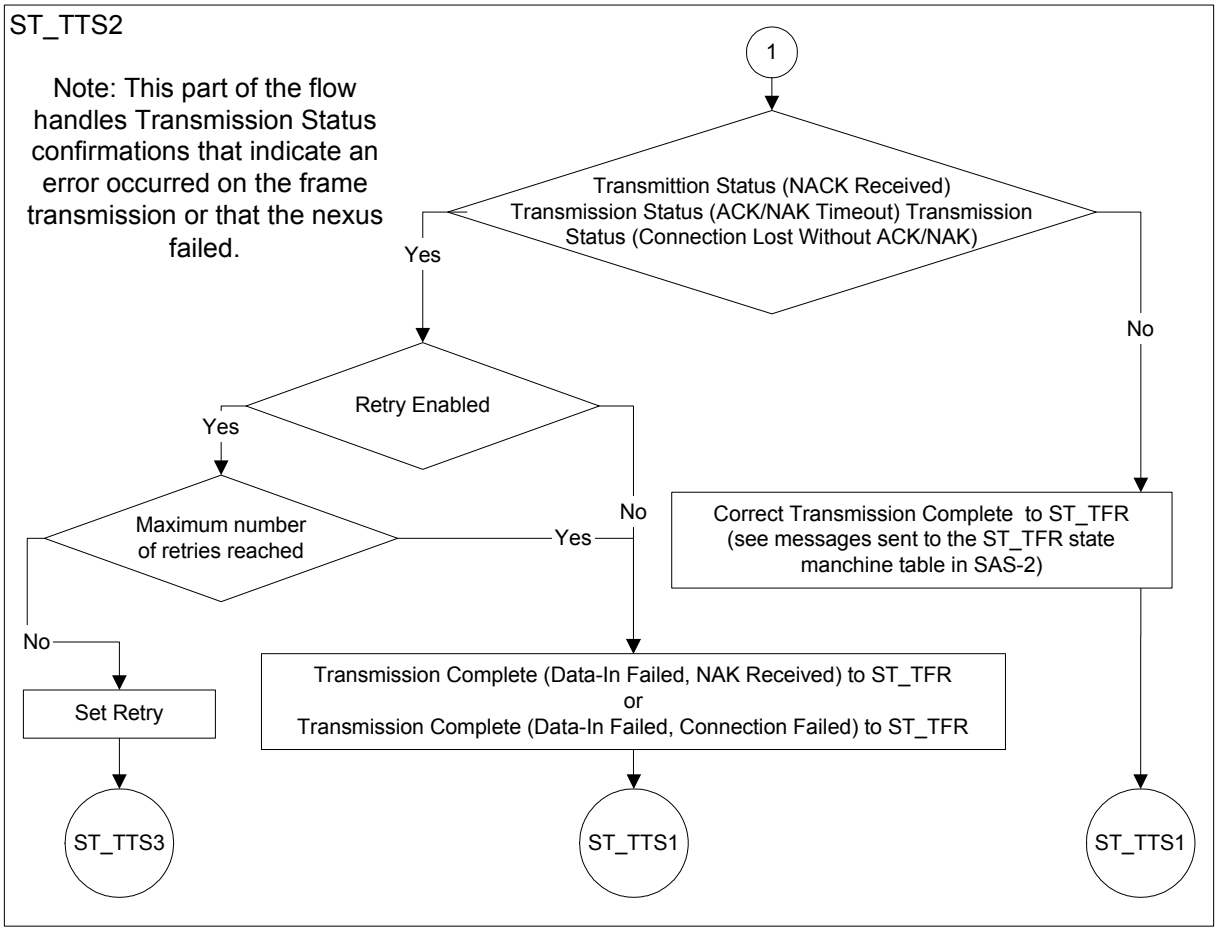

Figure 4 — Representation of transport layer (i.e., ST_TTS2) read data operation (part 2 or 2)