

To: T10 Technical Committee
From: John Worden (john.worden@seagate.com)
Date: 28 October 2002
Subject: 02-424r2 SAS Additional Port Layer Tweaks

Revision History

Revision 0 (18 October 2002) first revision
Revision 1 (22 October 2001) second revision
Revision 2 (28 October 2002) third revision

Related Documents

sas-r02 - Serial Attached SCSI revision 02
02-202r4 SAS Port Control State Machine updates (John Worden)
02-393r1 SAS Port layer rewrite

Overview

This revision adds clarification and minor corrections to the SAS Port layer rewrite (02-393r1), [corrects the header in revision 0, and adds changes discussed in the 10/24/02 SAS Protocol Meeting in Las Vegas, NV.](#) Changes include:

- a) Clarifies usage of the I_T nexus loss timer
- b) Adds the I_T nexus loss argument to the Transmit Frame request
- c) Adds Figure 3. Transmit Arguments plus explanatory text
- d) Minor technical corrections and other clarifications

Port Layer

8.1 Overview

The port layer (PL) state machines interface with one or more SAS link layer state machines and one or more SSP, SMP, and STP transport layer state machines to establish port connections and disconnections. The port layer state machines also interpret or pass transmit data, receive data, commands, and confirmations between the link and transport layers.

[Figure 1](#) shows the relationship of the port layer to the transport and link layers.

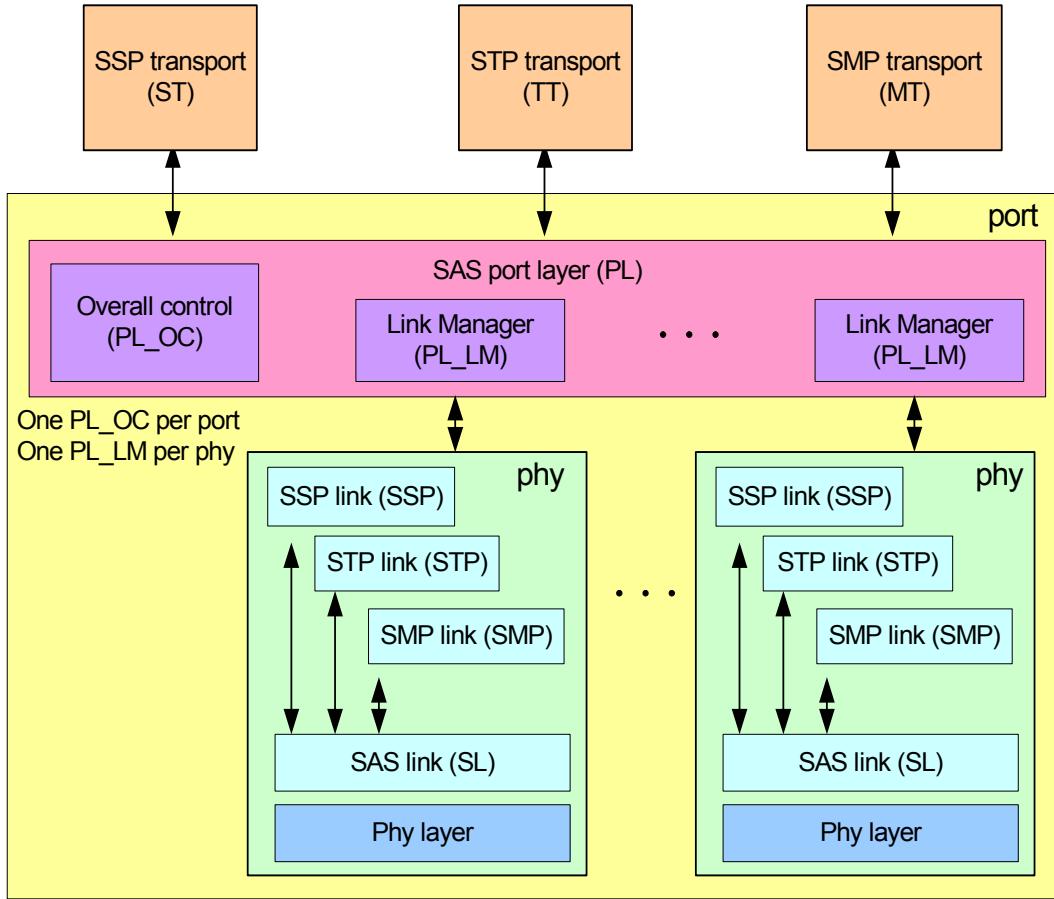


Figure 1. SAS port layer position

Two state machines running in parallel plus several commonly accessible counter/timers form the port layer.

The PL state machines are:

- Overall Control (PL_OC state machine); and
- Link Layer Management (PL_LM state machine).

There is one PC_OC state machine per port. There is one PL_LM state machine for each phy that is part of the port.

The port layer maintains the following timers/counters for each phy ([i.e., each PL_LM state machine](#)):

- bus inactivity time limit timer;
- maximum connect timer;
- I_T nexus loss timer;
- arbitration wait time (AWT) timer; and
- pathway blocked count (PBC) counter.

[The port layer maintains the following timers in the PL_OC state machine:](#)

- a AWT timer for each connection attempt to be retried (see 8.3.3.1), and
- a I_T Nexus Loss timer for each destination for which a connection is requested.

8.2 Port layer timers/counters descriptions

8.2.1 Bus inactivity time limit timer

The bus inactivity time limit timer (see 10.1.1.2) is initialized and monitored by the PL_LM state machine. [The bus inactivity time limit timer is initialized to the BUS INACTIVITY TIME LIMIT defined in bytes 4 and 5 of the](#)

Disconnect-Reconnect mode page for SSP. The timer shall count down. When the timer count is decremented to zero, the timer shall stop counting. A count of zero signifies that the timer has expired.

8.2.2 Maximum connect time limit timer

The maximum connect time limit timer (see 10.1.1.3) is initialized and monitored by the PL_LM state machine. The maximum connect time limit timer is initialized to the MAXIMUM CONNECT TIME LIMIT defined in bytes 8 and 9 of the Disconnect-Reconnect mode page for SSP. The timer shall count down. When the timer count is decremented to zero, the timer shall stop counting. A count of zero signifies that the timer has expired.

8.2.4 I_T nexus loss timer

The I_T nexus loss timer (see 10.1.2) is ~~initialized and monitored by the PL_LM state machine while trying to open a connection. There shall be one timer maintained for each destination for which a connection request has resulted in an Open Failed (No Destination) or Open Failed (Link Rate Not Supported) confirmation from the link layer. either:~~

- a) initialized to the I_T NEXUS LOSS TIME defined in bytes 4 and 5 of the Protocol-Specific Port Control mode page for SSP, started counting, and assigned a running status. When running, the timer shall count down. When the timer count is decremented to zero, the timer shall stop counting and assigned an expired status. ;
- b) initialized to a count of zero, stopped counting, and assigned a stopped status ; or
- c) initialized with received I_T nexus loss time and status arguments (i.e., stopped, running, or expired). If a I_T nexus loss timer is initialized with a stopped status, the I_T nexus loss timer shall remain stopped. If a I_T nexus loss timer is initialized with a running status, the I_T nexus loss timer shall start counting from the initialized I_T nexus loss time. If a I_T nexus loss timer is initialized with an expired status, the I_T nexus loss timer shall remain stopped and shall retain the expired status.

The I_T nexus loss timer provides the transport and application layers the information to enforce I_T nexus loss timeouts for:

- a) connection attempts which fail for either
 - a. Open Failed (Link Rate Not Supported);
 - b. Open Failed (No Destination);
 - c. Open Failed (Open Timeout Occurred) confirmations; or
- b) Open Failed (Pathway Blocked) confirmations received from the link layer if either a
 - a. Open Failed (Link Rate Not Supported);
 - b. Open Failed (No Destination); or
 - c. Open Failed (Open Timeout Occurred) confirmation

is received from the link layer prior to receiving Open Failed (Pathway Blocked) confirmations. If only Open Failed (Pathway Blocked) confirmations are received, the I_T nexus loss timer does not start running and the I_T nexus loss timeout limit is not applicable.

8.2.5 Arbitration wait time (AWT) timer

The AWT timer value (see 7.5.3) is passed to the link layer by the PL_LM state machine as an argument in its Open Connection request. ~~The timer shall be initialized as described in 7.11.3. The AWT timer is initialized to zero. The AWT timer shall count up.~~

8.2.6 Pathway blocked count (PBC) counter

The PBC counter (see 7.5.3) is passed to the link layer by the PL_LM state machine as an argument in its Open Connection request, initialized and incremented by the PL_LM state machine. The timer shall be initialized as described in 7.11.3. The PBC counter is initialized to zero. The PBC counter shall count up.

8.3 PL_OC state machine

8.3.1 Overview

A single PL_OC state machine exists for the port layer. The PL_OC state machine's purpose is to:

- a) Select a Transmit Frame request from the SSP, SMP and STP transport layers or a Retry Frame parameter from the PL_LM state machines and select a phy on which to transmit the frame;
- b) Close connections if needed;
- c) Direct the PL_LM state machine to open if needed and transmit the frame;
- d) Forward the response back to the transport layer;
- e) Process the Cancel, SMP Transmit Break, and Accept_Reject Opens requests from the transport layer; **and**
- f) Receive the HARD_RESET Received confirmation from the link layer and pass this confirmation to the transport layer; and
- g) Initialize and operate the PL_OC's I_T nexus loss timers, the AWT timers, and the PBC.

The PL_OC state machine contains these states:

- a) PL_OC1:Idle; and
- b) PL_OC2:Overall_Control.

The state machine shall start in the PL_OC1:Idle state. The state machine shall transition to the PL_LM1:Idle state from any other state after receiving an Enable Disable SAS Link (Disable) parameter from each of the link layers (see 7.8.4).

Figure 2 shows the PL_OC state machine.

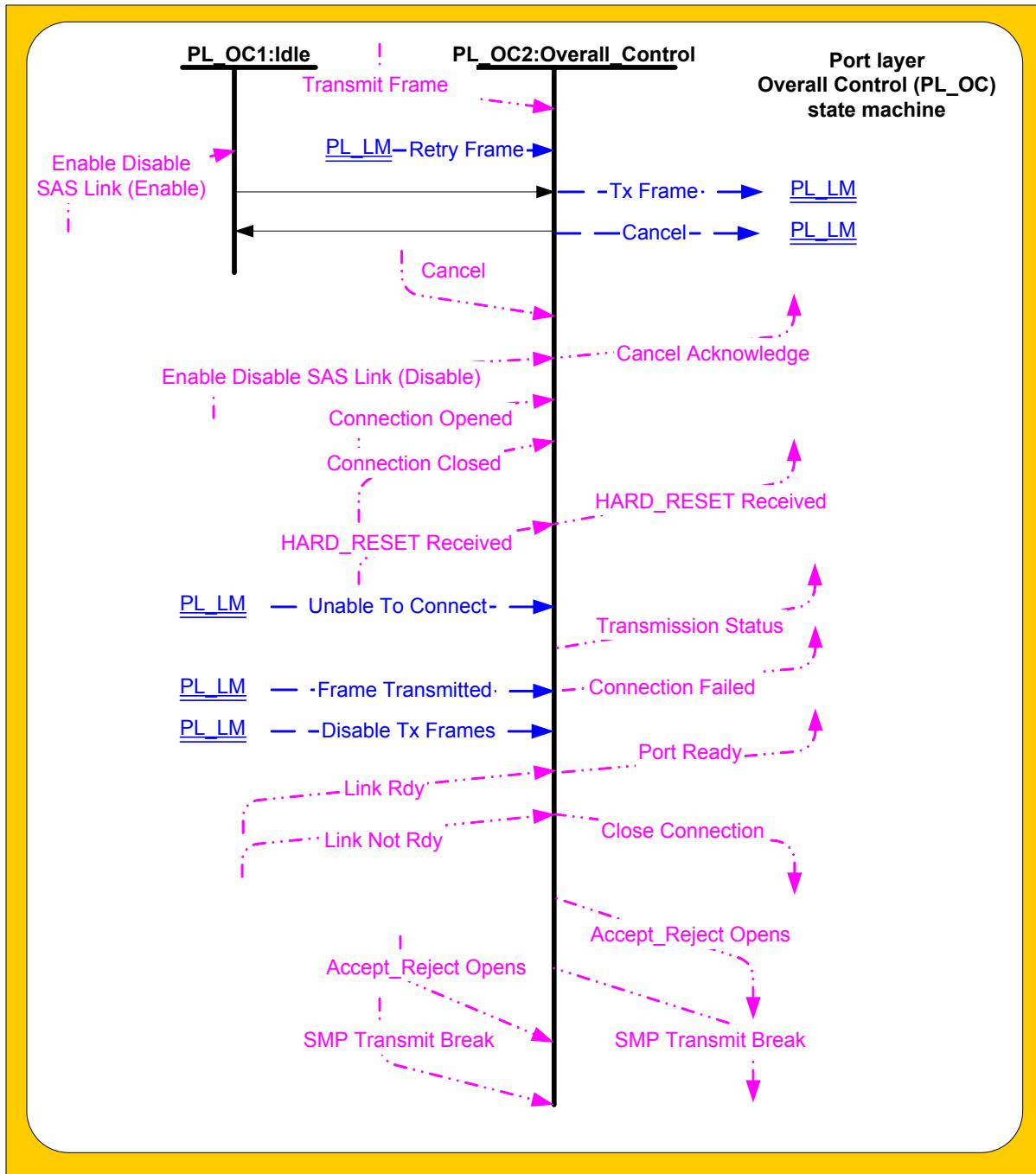


Figure 2. Port layer Overall Control (PL_OC) state machine

(Editors Note: Delete "Link Rdy" and "Link Not Rdy" from figure 2 ???)

8.3.2 PL_OC1:Idle state

8.3.2.1 State description

This state waits for at least one link layer to send confirmation that a phy is enabled, meaning that the port is enabled.

8.3.2.2 Transition PL_OC1:Idle to PC_OC2:Overall_Control

This transition shall occur when at least one Enable Disable SAS Link (Enable) is received from a link layer in the port.

8.3.3 PL_OC2:Overall_Control state

8.3.3.1 State description

8.3.3.1.1 State description overview

This state receives Transmit Frame requests from one or more transport layers (e.g., SSP, SMP, and STP) and Retry Frame parameters from one or more PL_LM state machines.

- a) each transport layer may have multiple Transmit Frame requests pending at once.
- b) and each PM_LM state machine may generate multiple Retry Frame parameters over time.
- c) This state chooses the order in which the requests are processed.
- d) This state machine shall service one Transmit Frame request or Retry Frame parameter at a time.

If multiple phys exist in the port, this state may establish multiple connections to multiple destination SAS addresses simultaneously (one per phy) and may transmit and receive via the different connections simultaneously.

The Transmit Frame request from the transport layer includes the following arguments:

- a) Interlocked or Not Interlocked (for SSP only);
- b) destination SAS address;
- c) source SAS address;
- d) link rate;
- e) initiator bit;
- f) initiator connection tag; and
- g) frame.

The Retry Frame parameter from the PL_LM state machine includes the following arguments:

- a) Phy identifier - the phy on which the frame was last attempted;
- b) Interlocked or Non Interlocked Balance Required (for SSP only);
- c) arbitration wait time;
- d) pathway blocked count;
- e) destination SAS address;
- f) source SAS address;
- g) link rate;
- h) initiator bit;
- i) initiator connection tag; and
- j) frame.

The Tx Frame parameter sent to the PL_LM state machine includes the following arguments:

- a) Balance Required or Balance Not Required (for SSP only);
- b) arbitration wait time;
- c) pathway blocked count;
- d) I T nexus loss time and status
- e) destination SAS address;
- f) source SAS address;
- g) link rate;
- h) initiator bit;
- i) initiator connection tag; and
- j) frame.

[Editor's note: really, there is a data structure holding each new request from the transport layer. If it gets retried, it is tagged as such in the structure.]

8.3.3.1.2 Keep track of connections/frame requests

This state machine shall keep track of which phys (represented by corresponding PL_LM state machines) are available. A phy is available if it is not processing a Tx Frame parameter.

This state shall consider a phy as having an active connection after receiving a Connection Opened confirmation from the link layer until receiving a Connection Closed confirmation from the link layer.

This state shall consider a phy as processing a Tx Frame parameter after sending the parameter to the PL_LM state machine until receiving a Transmission Status confirmation or an Unable to Connect confirmation from the link layer.

If this state has no pending Transmit Frame requests or Retry Frame parameters pending for a phy with an open connection, this state should send a Close Connection request to that phy (see 7.14.7).

8.3.3.1.3 Select a request to process and the phy on which to process it

This state may process a Transmit Frame request or Retry Frame parameter whenever a phy is available. The phy should be chosen following this order of precedence:

1. a phy that has a connection open to the same destination and is not processing a Tx Frame request;
2. a phy that does not have a connection open; or
3. a phy that has a connection open to a different destination and is not processing a Tx Frame request.

A destination is considered the same if it has the same SAS address and protocol.

Additional rules for wide SSP port selection of phys are described in 9.2.3.

[Editor's note: consider moving the rules here. They're frame type dependent, though.]

If a phy is chosen with a connection open to a different destination, this state shall send a Close Connection to the phy. This state may then process the request on the selected phy when the connection is closed, or it may choose another request to process on the selected phy when the connection is closed, or it may choose another phy on which to process the request if one becomes available. This state shall send no more than one Close Connection on behalf of the same request.

When Retry Frame parameters are pending with non-zero arbitration wait times, this state shall increment those arbitration wait times until they are processed.

This state shall not process a Retry Frame parameter until a retry delay (see 7.x) has expired from when the parameter was received.

[Editor's note: Define "Retry Delay" in 7.2 as 15 us.]

8.3.3.1.4 Filling in the Transmit Frame/Retry Frame/Tx Frame arguments

This state shall send a Tx Frame parameter to the PL_LM state machine corresponding to the selected phy carrying all the arguments from the selected Transmit Frame request or Retry Frame parameter unless otherwise noted in this subclause.

The AWT and PBC arguments for the PL_OC Tx Frame parameter are transferred to the selected PL_LM's AWT timer and PBC counter respectively. The AWT and PBC arguments for Transmit Frame requests shall be zero and are not received as arguments with the Transmit Frame request from the transport layer. The AWT and PBC arguments for the PL_OC Retry Frame parameters are received as arguments (which are read from the appropriate PL_LM AWT timer or PBC counter) in the Unable to Connect parameter received from the PL_LM state machine. . The PL_OC Retry Frame AWT argument is stored in a corresponding PL_OC Retry Frame AWT timer which shall start incrementing if the AWT argument received from the PL_LM state machine is non zero. If the AWT argument received from the PL_LM state machine is zero, the PL_OC AWT timer shall not start incrementing.

The PL_OC Retry Frame parameter shall include a Balanced Required argument.

The I_T nexus loss arguments for the PL_OC Tx Frame parameter are transferred to the selected PL_LM's I_T nexus loss timer. . The selected PL_LM timer shall be set to the time and state (i.e, either stopped, running, or expired) of these arguments received in the TX Frame parameter. The I_T nexus loss time and status arguments for the Tx Frame parameter are read from the PL_OC I_T nexus loss timer for this destination. For each destination, the PL_OC I_T nexus loss timer is updated with the time and status arguments of the PL_LM I_T nexus loss timer received with an Unable To Connect parameter from the PL_LM state machine.

Some of the arguments are modified as follows:

A) For SSP requests, the Tx Frame parameter should include a Balance Not Required argument when:

- a) a Transmit Frame (Non Interlocked) request is being processed;
- b) the phy selected has a connection open to the same destination;
- c) the previous request processed for this phy was a Transmit Frame(Non Interlocked) request to the same destination;
- d) the connection has not been lost since the previous frame was transmitted; and
- e) the frame has the same tag value as the previous frame.

Otherwise, it shall include a Balance Required argument.

B) If this state is processing either:

- a) a Transmit Frame request or Retry Frame parameter with the same destination as the last Retry Frame parameter received; or
- b) the last Retry Frame parameter received;

and:

- a) the phy selected does not have a connection opened or has a connection open to a different destination; and
- b) this is the next request sent to any phy after the last Retry Frame parameter was received;

then the pathway blocked count shall be set to the value of the pathway blocked count argument received with the last Retry Frame parameter. ~~Otherwise, the pathway blocked count argument shall be set to zero.~~

~~This ensures that As a result~~, only the next request from the port uses that pathway blocked count value. If the port chooses a request to a different destination, it ~~is not allowed to shall not~~ transfer the pathway blocked count value to the different request. and the pathway blocked count shall be set to zero.

C) The arbitration wait time argument shall be set to:

- a) the value of the arbitration wait time argument received with the last Retry Frame parameter; or
- b) zero or a vendor-specific value less than 7FFFh (see 7.7.3). The argument should be set to zero.

If the arbitration wait time from the last Retry Frame parameter is used for a different frame, the arbitration wait time for the last Retry Frame parameter shall be set to zero.

~~This ensures that As a result~~ only the next request from the port uses that arbitration wait time value. If the port chooses a request to a different destination, it is allowed to transfer the arbitration wait time value to the different request.

[Editor's note: you get to use a nonzero AWT one time on the next available link for any request. If you were trying to open A and B opened you instead, you can put OPEN(A) out on another link with an incremented AWT, or you can put OPEN(C) out instead. Once you use the value, however, the OPEN (B) loses ownership of it; when it is issued later, it gets zero (unless it borrows from a later AWT).]

8.3.3.1.5 Confirmations

This state shall send a Transmission Status confirmation or a ~~n~~ Unable To Connect Connection Failed confirmation to the transport layer to finish servicing each Transmit Frame request.

This state shall send a Transmission Status (Frame Transmitted) confirmation to the transport layer after receiving a Frame Transmitted parameter from the PL_LM state machine. Since the transport layer responses for any sequence of frame transmissions (e.g. multiple non-interlocked frame transmissions in SSP) are instantaneous, the next Transmit Request from the transport layer for the same I_T_L_Q should be available when the Transmission Status (Frame Transmitted) confirmation is sent. This ensures that this state is able to choose the same phy and connection (e.g. and continue a non-interlocked sequence).

This state shall send a Transmission Status (Connection Lost) confirmation to the transport layer if a Connection Closed confirmation is received from the link layer after a Tx Frame parameter is sent but before a Frame Transmitted parameter is received.

For SSP only, t~~This~~ state shall send a Connection Failed (Connection Lost Without ACK/NAK) confirmation to the transport layer if a Connection Closed confirmation is received from the link layer after a Frame Transmitted

parameter is received but before an ACK Received, NAK Received or ACK/NAK Timeout confirmation is received.

[Editor's note: if Unable To Connect (Retry or Pathway Blocked, I_T_Nexus Loss) arrives, pass that up as Transmission Status]

After a Connection Closed confirmation is received from the link layer, the phy is available for a new connection.

This state shall send a Transmission Status confirmation or a Cancel Acknowledgement confirmation to the transport layer after receiving an Unable to Connect parameter from the PL_LM state machine based on the mapping in Table 60.

Table 60 – Confirmations to transport layer from the PL_OC state machine

| Unable To Connect parameter | Confirmation sent to transport layer |
|---|---|
| Unable To Connect (Wrong Destination) | Transmission Status (Open Failed – Wrong Destination) |
| Unable To Connect (Link Rate Not Supported) | Transmission Status (Open Failed – Link Rate Not Supported) |
| Unable To Connect (Protocol Not Supported) | Transmission Status (Open Failed – Protocol Not Supported) |
| Unable To Connect (Bad Destination) | Transmission Status (Open Failed – Bad Destination) |
| Unable To Connect (STP Resources Busy) | Transmission Status (Open Failed – STP Resources Busy) |
| Unable To Connect (Break Received) | Transmission Status (Open Failed – Break Received) |
| Unable To Connect (No Destination) | Transmission Status (Open Failed – No Destination) |
| Unable To Connect (Open Timeout Occurred) | Transmission Status (Open Failed – Open Timeout Occurred) |
| Unable To Connect (Pathway Blocked) | Transmission Status (Open Failed – Pathway Blocked) |
| Unable To Connect (Cancel Acknowledgement) | Cancel Acknowledgement |

The stopped, running or expired I_T nexus loss timer status received with the Unable To Connect parameter is sent to the transport layer as an argument with the Open Failed Transmission Status confirmation.

8.3.3.1.6 Handling Cancel requests

If this state receives a Cancel request for a specific Transmit Frame request from the transport layer and this frame is not currently being processed (either from the Transmit Frame request itself or from a Retry Frame parameter), this state shall send a Cancel Acknowledge confirmation to the transport layer and terminate processing of the Transmit Frame request.

If this state receives a Cancel request for a specific Transmit Frame request from the transport layer, this frame is currently being processed, and a connection has been opened, this state shall send a Close Connection request to the link layer. When the connection has been closed, this state shall send a Cancel Acknowledge confirmation to the transport layer and terminate processing of the Transmit Frame request.

If this state receives a Cancel request for a specific Transmit Frame request from the transport layer, this frame is currently being processed, and a connection has not yet been opened, this state shall send a Cancel parameter to the PL_LM state machine. When the PL_LM state machine has terminated the connection and returned an Unable To Connect (Cancel Acknowledge) parameter, this state shall send a Cancel Acknowledge confirmation to the transport layer and terminate processing of the Transmit Frame request.

8.3.3.1.7 Handling other requests

If a Disable Tx Frames parameter is received from the PL_LM state machine, this state shall not send another Tx Frame parameter to the phy until after a Connection Closed confirmation has been received from the phy.

If this state receives an Accept_Reject Opens request from the transport layer, this state shall send an Accept_Reject Opens request to the link layer.

If this state receives an SMP Transmit Break request from the transport layer, this state shall send an SMP Transmit Break request to the link layer.

If this state receives a HARD_RESET Received confirmation from the link layer, this state shall send a HARD_RESET Received confirmation to the transport layer.

8.3.3.2 Transition PL_OC2:Overall_Control to PL_OC1:Idle

This transition shall occur when an Enable Disable SAS Link (Disable) confirmation is received from each of the link layers in the port.

8.4 PL_LM state machine

8.4.1 Overview

A PL_LM state machine exists for each phy contained in the port. The PL_LM state machine's purpose is to:

- a) Request or retry a connection sequence via the link layer and handle exception conditions;
- b) Send Retry Frame parameters to the PL_OC state machine if a connection attempt fails and is to be retried;
- b)c) Maintain a connected state for the phy;
- e)d) Forward frame transmission requests from the PL_OC state machine to the link layer;
- e)e) Pass confirmations from the link layer to the PL_OC state machine;
- e)f) Update the PL_LM1_T nexus loss timer, the PL_LM arbitration wait time timer, and the PL_LM pathway blocked count counter; and
- f)g) For target ports only, implement the bus inactivity time limit timer and the maximum connect time timer and close the connection if they have been exceeded.

The PL_LM state machine contains these states:

- a) PL_LM1:Idle;
- b) PL_LM2:ReqWait;
- c) PL_LM3:Connected; and
- d) PL_LM4:Wait_For_Close.

The state machine shall start in the PL_LM1:Idle state. The state machine shall transition to the PL_LM1 state from any other state after receiving an Enable Disable SAS Link (Disable) parameter from the link layer (see 7.8.4).

Figures 3 and 4 shows the PL_LM state machine.

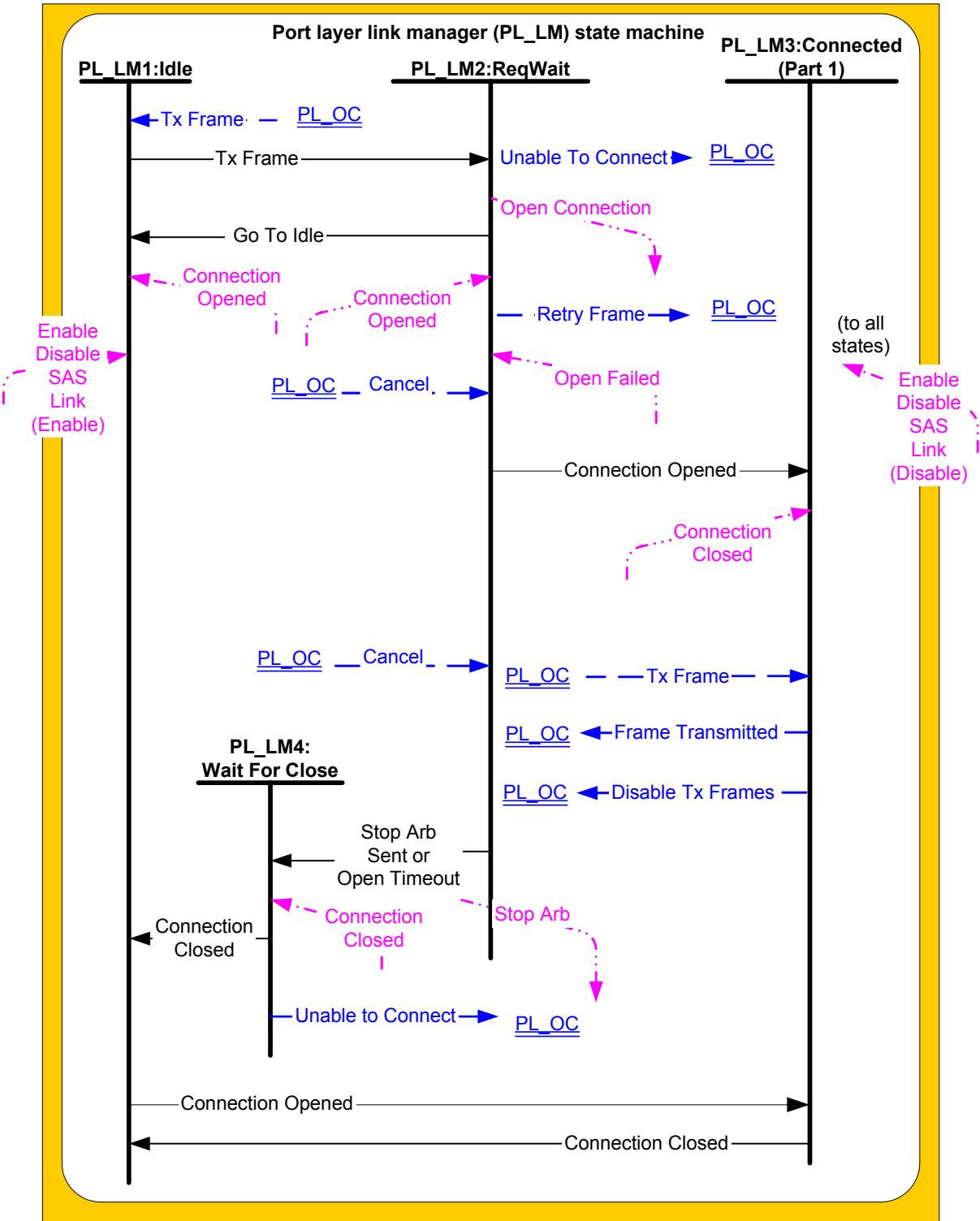


Figure 3. Port layer Link Manager (PL_LM) state machine – part 1

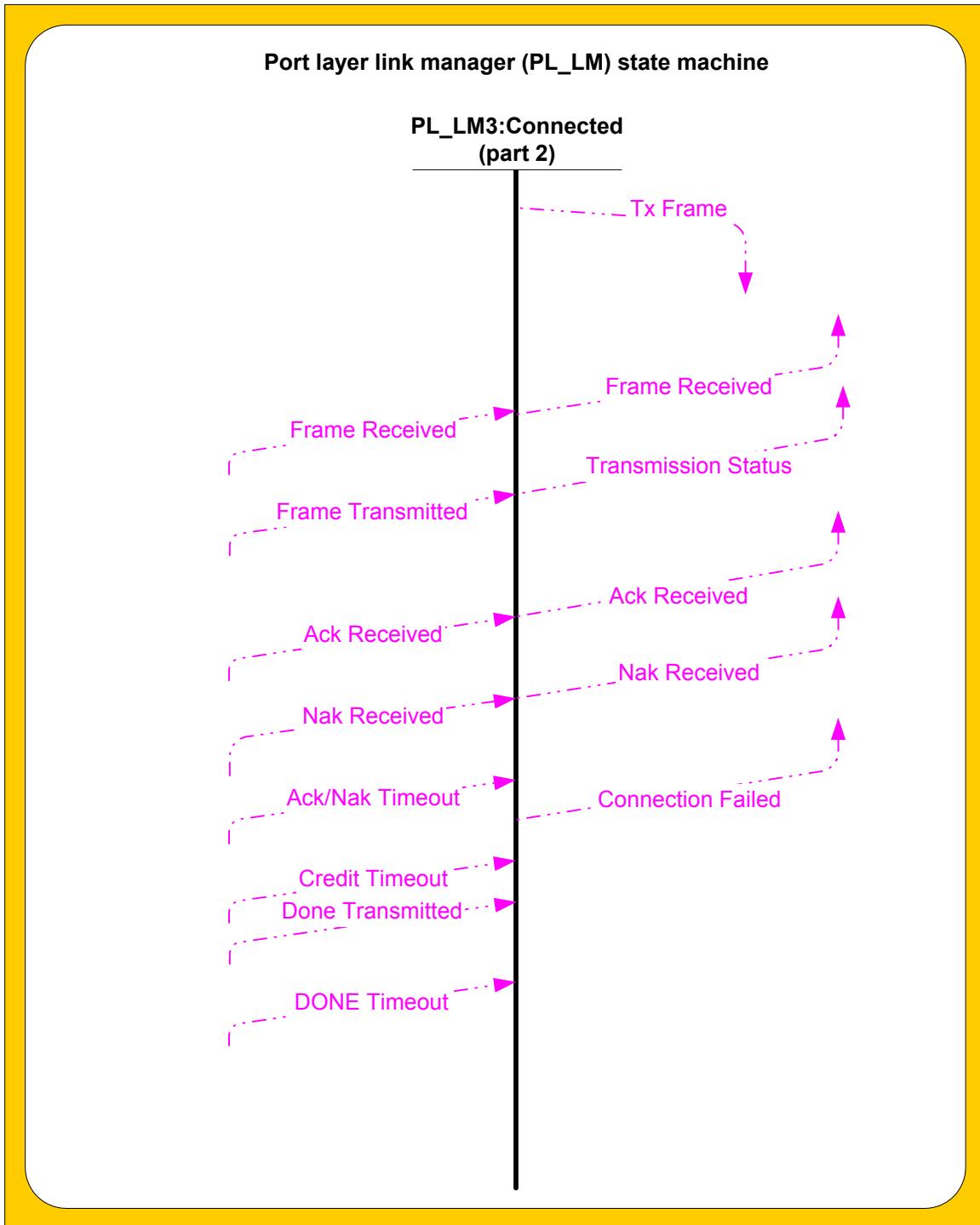


Figure 4. Port layer Link Manager (PL_LM) state machine – part 2

8.4.2 PL_LM1:Idle state

8.4.2.1 State description

This is the initial state of the PL_LM state machine. This state is also entered when a connection is closed or a connection attempt is terminated.

8.4.2.2 Transition PL_LM1:Idle to PL_LM2:ReqWait

This transition shall occur after receiving a Tx Frame parameter from the PL_OC state machine.

8.4.2.3 Transition PL_LM1:Idle to PL_LM3:Connected

This transition shall occur after receiving a Connection Opened parameter from the link layer.

8.4.3 PL_LM2:ReqWait state

8.4.3.1 State description

8.4.3.1.1 State description overview

This state sends an Open Connection request to the selected phy and waits for a response.

The Open Connection request includes these arguments, filled in with values from the Tx Frame parameter:

- a) destination SAS address;
- b) protocol;
- c) arbitration wait time;
- d) pathway blocked count;
- e) link rate;
- f) initiator bit; and
- g) initiator connection tag.

This state shall start incrementing the arbitration wait time after sending the Open Connection request.

If a Cancel parameter is received from the PL_OC state machine, this state shall send a Stop Arb request to the link layer to cancel the connection request.

8.4.3.1.2 I_T nexus loss timer

For SSP ports, the [PL_LM](#) [I_T](#) nexus loss timer ~~for a destination~~ shall be stopped, set to zero, and assigned a stopped status when one of the following confirmations is received:

- a) Connection Opened;
- b) Open Failed (Protocol Not Supported);
- c) Open Failed (Retry); or
- d) Open Failed (STP Resources Busy).

[Editor's note: requests that successfully reach the destination stop the I_T nexus loss timer]

For SSP ports, if the [PL_LM](#) [I_T](#) nexus loss timer is not already running, the [PL_LM](#) [I_T](#) nexus loss timer ~~for a destination~~ shall be initialized, assigned a running status, and started decrementing when one of the following confirmations is received:

- a) Open Failed (Link Rate Not Supported);
- b) Open Failed (No Destination); or
- c) Open Failed (Open Time Out).

If the I_T nexus loss timer is already running, the I_T nexus loss timer shall continue running for those confirmations.

[Editor's note: requests that cannot find the destination, but have the possibility of finding it when retried, start the I_T nexus loss timer]

[Editor's note: all other responses continue running the timer if already started, but don't start it themselves - Wrong Destination, Bad Destination, Pathway Blocked.]

[Editor's note: how should OPEN_REJECT(RESERVED) be handled? This is a global issue for the standard. It needs to have certain behaviors predefined. Perhaps a set of OPEN_REJECT(RESERVED)s are needed.]

For SSP ports, the I_T Nexus Loss argument shall be included in any Open Failed confirmation if the I_T nexus loss timer expires while in this state.

[Editor's note: make sure the Transport Layer looks for I_T Nexus Loss in its Open Failed confirmations and passes it to the Application Layer as the "Nexus Loss" signal described in SAM-3. The SAS state machines will shall only check the timer after trying another request.]

8.4.3.1.3 Connection Opened handling

If a Connection Opened (Destination Opened) confirmation is received from the link layer and the destination does not match, this state shall send a Retry Frame parameter to the PL_OC state machine with arguments from the Tx Frame, including the updated arbitration wait time. The pathway blocked count shall be maintained. The arbitration wait time timer shall be stopped.

[if Stop Arb was sent but Connection Opened is still received, the Stop Arb was unsuccessful and the PL_OC state machine has to send a Close Connection to the link layer itself]

8.4.3.1.4 Open Failed handling

If a Cancel parameter has been received from the PL_OC state machine and an Open Failed confirmation is received, this state shall send an Unable To Connect (Cancel Acknowledgement) parameter to the PC_OC state machine. Otherwise, this state shall send either an Unable To Connect parameter with the same argument as that received with the Open Failure confirmation or a Retry Frame parameter. The Unable To Connect parameter shall include as arguments:

- a) the counts of the AWT timer, PBC counter, and I_T Nexus loss timer; and
- b) the status of the I_T Nexus loss timer.

The arbitration wait time timer shall be stopped if an Open Failed (Protocol Not Supported), Open Failed (Retry), or Open Failed (Wrong Destination) confirmation is received from the link layer.

~~In addition to the Unable to Connect parameter,~~ This state shall send a Retry Frame parameter under the conditions shown in Table 61. The pathway blocked argument shall be modified as specified; the other arguments shall be copied from the Tx Frame being processed.

Table 61– Retry Frame conditions

| Open Failed confirmation argument received from the link layer | Has the I_T nexus loss timer has expired? | Pathway blocked count argument for Retry Frame |
|--|---|--|
| Open Failed (Pathway Blocked) | no | increment by one |
| Open Failed (Retry) | yes or no | set to zero |

8.4.3.2 Transition PL_LM2:ReqWait to PL_LM1:Idle

This transition shall occur after an Open Failed parameter has been sent to the PC_OC state machine.

8.4.3.3 Transition PL_LM2:ReqWait to PL_LM3:Connected

This transition shall occur after a Connection Opened confirmation has been received from the link layer.

8.4.3.4 Transition PL_LM2:ReqWait to PL_LM4:Wait_For_Close

This transition shall occur after receiving an Open Failed (Port Layer Request) confirmation from the link layer.

If a Cancel parameter is received from the PL_OC state machine, this transition shall occur after receiving an Open Failed (Open Timeout Occurred) confirmation from the link layer.

8.4.4 PL_LM3:Connected state

8.4.4.1 State description

When in this state, frame transmissions and receptions may occur.

When in this state and a frame is to be transmitted, the PL_OC state machine may send the Tx Frame parameter to this state. For STP connections, the port layer connects the transport layer to the link layer and is transparent; Tx Frame is not used. For SSP connections, arguments for the to-Tx Frame parameter shall include Balance Required or Balance Not Required. For SMP connections , Tx Frame shall include THE the SMP argument.

This state shall generate a Tx Frame request to the link layer when a Tx Frame parameter is received from the PL_OC state machine.

After a Tx Frame request has been sent to the link layer and a Frame Transmitted confirmation is received from the link layer, this state shall send a Frame Transmitted parameter to the PL_OC state machine to notify the PL_OC state that another frame may be transmitted.

This state shall also send the following confirmations regarding the previous frame transmission confirmations from the link layer to the transport layer:

1. ACK Received;
2. NAK Received;
3. Connection Failed (ACK/NAK Timeout): Mapped from the ACK/NAK Timeout confirmation from the link layer; or
4. Transmission Status (Credit Timeout): Mapped from the Credit Timeout confirmation from the link layer.

For SMP ports, this state shall send the Frame Received (SMP) or Frame Received (SMP Failure) confirmations from the link layer to the transport layer

For SSP ports, this state shall send the Frame Received (ACK/NAK Balanced) or Frame Received (ACK/NAK Not Balanced) confirmations from the link layer to the transport layer.

For SSP ports, this state shall transfer a Disable Tx Frames parameter to the PL_OC state machine if a DONE Received confirmation is received from the link layer.

For SSP target ports, upon entry into this state, the bus inactivity time limit timer shall be initialized and started. It shall be re-initialized every time a Tx Frame request is processed. If the timer expires, this state shall send a Close Connection request to the link layer after the current Tx Frame is finished and shall ignore new Tx Frame requests. The timer stops running upon exit from this state.

For SSP target ports, upon entry into this state, the maximum connect time timer shall be initialized and started. If the timer expires, this state shall send a Close Connection request to the link layer after the current Tx Frame is finished and shall ignore new Tx Frame requests. The timer stops running upon exit from this state.

8.4.4.2 Transition PL_LM3:Connected to PL_LM1:Idle

This transition shall occur after a Connection Closed confirmation is received from the link layer.

8.4.5 PL_LM4:Wait_For_Close state

8.4.5.1 State description

This state shall wait for a Connection Closed confirmation to be received from the link layer and then send an Unable to Connect (Cancel Acknowledge) parameter to the PL_OC state machine.

8.4.5.2 Transition PL_LM4:Wait_For_Close to PL_LM1:Idle

This transition shall occur after a Connection Closed confirmation is received from the link layer.