Date: November 1, 2002

To: T10 Committee (SCSI)

From: George Penokie (IBM/Tivoli)

Subject: SSP Link Layer State Manchine Rewrite

6.10.1 SSP link layer transmission rules

When the link is request to transmit a dword, it shall transmit the that dword. If there are multiple requests to transmit primitives, the following priority should be followed when selecting the primitives:

1) ALIGN;
2) ACK or NAK;
3) RRDY;
4) CREDIT_BLOCKED;
5) DONE; then
6) SOF/frame/EOF.

When the link has been requested to transmit a frame (i.e., SOF, the data dwords of the frame, and an EOF) and a primitive is requested to be transmitted while the frame is being transmitted, the link may transmit the indicated primitive by inserting the primitive between the frames' dwords.

6.11 SAS link layer state machine for initiator phys and target phys (SL)

6.11.1 Overview

The SAS link layer (SL) state machine controls connections, handling both connection requests (OPEN address frames), CLOSE primitive sequences, and BREAK primitive sequences. The state machine contains the following states:

a) SL0:Idle (see 6.11.2);
b) SL1:ArbSel (see 6.11.3);
c) SL2:Selected (see 6.11.4);
d) SL3:Connected (see 6.11.5);
e) SL4:DisconnectWait (see 6.11.6);
f) SL5:BreakWait (see 6.11.7); and
g) SL6:Break (see 6.11.8).

The state machine shall start in the SL0:Idle state. The state machine shall transition to the SL0:Idle state from any other state after receiving an Enable Disable SAS Link (Disable) parameter from the IR state machines (see 7.8.5).

The SL state machine receives the following requests from the port layer:

a) Open Connection;
b) Stop Arb;
c) Accept_Reject Opens (Accept); and
d) Accept_Reject Opens (Reject).

The SL state machine receives the following parameters from the SSP, STP, and SMP link layer state machines:

a) Request Break; and
b) Request Close.

The SL state machine sends the following parameters to the SSP, STP, and SMP link layer state machines:

a) Enable Disable SSP;
b) Enable Disable STP; and
c) Enable Disable SMP.

The SL state machine sends the following confirmations to the port layer:

a) Open Failed (Port Layer Request);
b) Open Failed (Open Timeout Occurred);
c) Open Failed (Break Received);
d) Open Failed (No Destination);
e) Open Failed (Bad Destination);
f) Open Failed (Wrong Destination);
g) Open Failed (Link Rate Not Supported);
h) Open Failed (Protocol Not Supported);
i) Open Failed (Retry);
j) Open Failed (STP Resources Busy);
k) Open Failed (Pathway Blocked);
l) Connection Opened (SSP, Source Opened);
m) Connection Opened (SSP, Destination Opened);
n) Connection Opened (STP, Source Opened);
o) Connection Opened (STP, Destination Opened);
p) Connection Opened (SMP, Source Opened);
q) Connection Opened (SMP, Destination Opened);
r) Connection Closed (Break Received);
s) Connection Closed (Link Broken);
t) Connection Closed (Close Timeout); and
u) Connection Closed (Normal).

The SL state machine sends no requests to the phy layer.

The SL state machine receives no confirmations from the phy layer.

The SL state machine receives the following parameters from the IR state machines (see 7.8.5):

a) Enable Disable SAS Link (Enable); and
b) Enable Disable SAS Link (Disable).

Unless otherwise stated within the state description, all disparity errors, illegal characters, and unexpected primitives (i.e., any primitive not described in the description of the SL state) received within any SL state shall be ignored and idle dwords shall be transmitted.

Any detection of an internal error shall cause the SL state machine to transition to the SL5:BreakWait state.

Figure 56 describes the SAS link layer state machine.

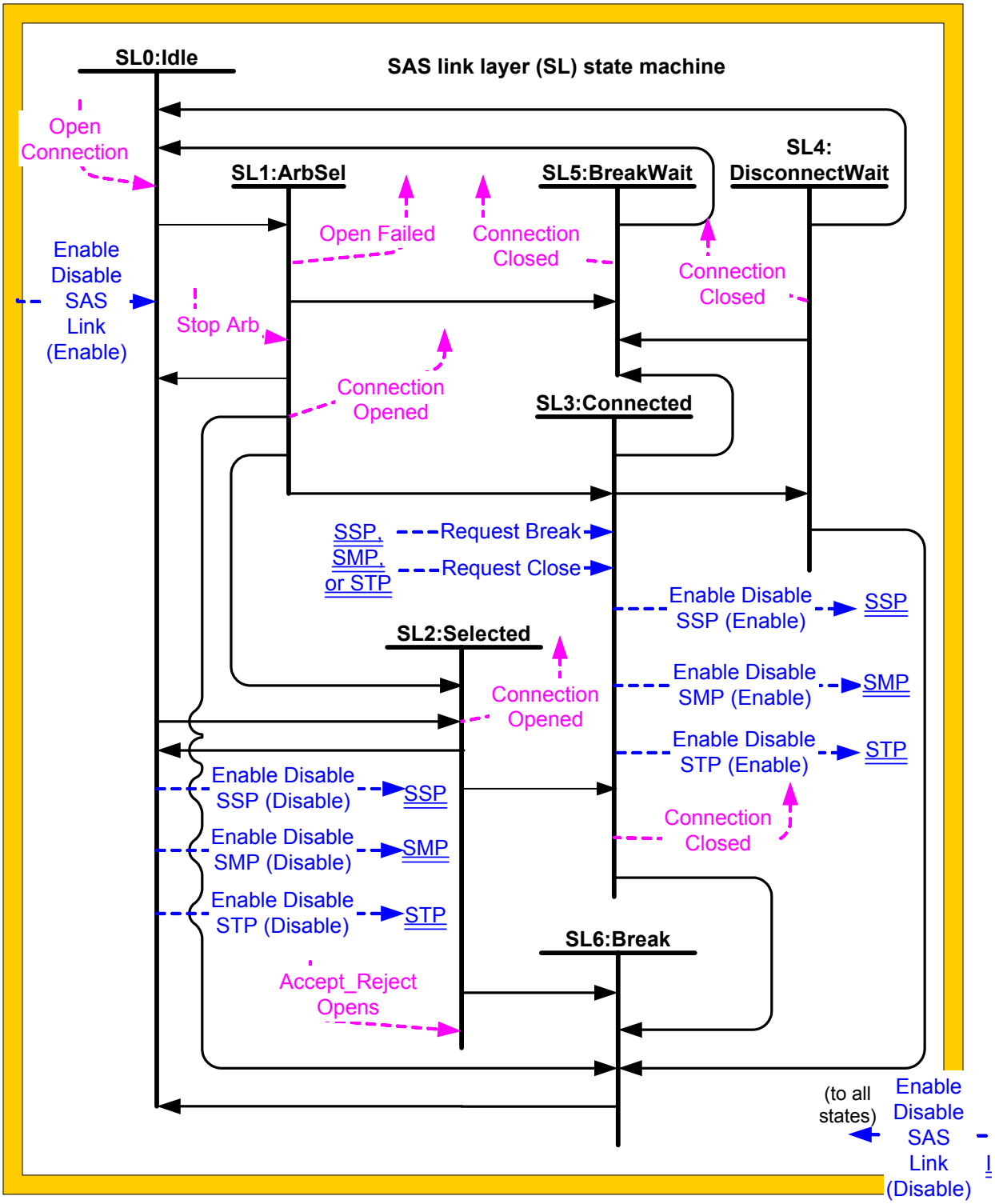

Figure 56 — SAS link layer (SL) state machine

6.11.2 SL0:Idle state

6.11.2.1 State description

The SL0:Idle state is the initial state and is the state that is used when the SL state machine is activated and there is no active connection.

Upon entry to this state, this state shall send Enable Disable SSP (Disable), Enable Disable SMP (Disable), and Enable Disable STP (Disable) parameters to the SSP, SMP, and STP link layer state machines.

This state shall transmit idle dwords (see 7.3).

6.11.2.2 Transition SL0:Idle to SL1:ArbSel

This transition shall occur after both an Enable Disable SAS Link (Enable) parameter is received from the IR state machines and an Open Connection request is received from the port layer. The Open Connection request includes these arguments:

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

6.11.2.3 Transition SL0:Idle to SL2:Selected

This transition shall occur after both an Enable Disable SAS Link (Enable) parameter is received from the IR state machines and a valid OPEN address frame is received.

A valid OPEN address frame is 32 bytes long and has no CRC error (see 7.7.3).

6.11.3 SL1:ArbSel state

6.11.3.1 State description

This state is used to make a connection request.

This state shall:

1) transmit an OPEN address frame with the address frame fields set to the arguments received with the Open Connection request;
2) initialize an open timeout timer to 1 ms, start the open timeout timer; and
3) start transmitting idle dwords.

This state shall ignore incoming BREAKs, OPEN_REJECTs, and OPEN_ACCEPTs until the OPEN address frame has been transmitted.

If an AIP is received after transmitting the OPEN address frame, the open timeout timer shall be set to its initial value of 1 ms and resume counting. The state machine shall not enforce a limit on the number of AIPs received.

The SL1_ArbSel state shall be exited after:

a) receiving an OPEN_REJECT;
b) receiving a BREAK primitive sequence;
c) receiving an OPEN_ACCEPT;
d) receiving an OPEN address frame if the arbitration fairness rules indicate the received OPEN address frame overrides the transmitted OPEN address frame;
e) the open timeout timer is exceeded; or
f) a Stop Arb request is received from the port layer.

6.11.3.2 Transition SL1:ArbSel to SL0:Idle

If an OPEN_REJECT (NO DESTINATION) is received this state shall send an Open Failed (No Destination) confirmation to the port layer.

If an OPEN_REJECT (BAD DESTINATION) is received this state shall send an Open Failed (Bad Destination) confirmation to the port layer.

If an OPEN_REJECT (WRONG DESTINATION) is received this state shall send an Open Failed (Wrong Destination) confirmation to the port layer.

If an OPEN_REJECT (LINK RATE NOT SUPPORTED) is received this state shall send an Open Failed (Link Rate Not Supported) confirmation to the port layer.

If an OPEN_REJECT (PROTOCOL NOT SUPPORTED) is received this state shall send an Open Failed (Protocol Not Supported) confirmation to the port layer.

If an OPEN_REJECT (RETRY) is received this state shall send an Open Failed (Retry) confirmation to the port layer.

If an OPEN_REJECT (STP RESOURCES BUSY) is received this state shall send an Open Failed (STP Resources Busy) confirmation to the port layer.

If an OPEN_REJECT (PATHWAY BLOCKED) is received this state shall send an Open Failed (Pathway Blocked) confirmation to the port layer.

This transition shall occur after sending the Open Failed confirmation.

6.11.3.3 Transition SL1:ArbSel to SL2:Selected

If one or more AIPs have been received, then a valid OPEN address frame is received, this transition shall occur. The incoming OPEN address frame overrides the outgoing OPEN address frame after AIP.

If no AIPs have been received when a valid OPEN address frame is received, and the arbitration fairness rules (see 7.12.3) indicate the received OPEN address frame overrides the outgoing OPEN address frame, this transition shall occur.

The arbitration fairness comparison shall use the value of the arbitration wait time argument to the Open Connection request for the outgoing OPEN address frame and the value of the ARBITRATION WAIT TIME field received in the incoming OPEN address frame.

6.11.3.4 Transition SL1:ArbSel to SL3:Connected

This transition shall occur if an OPEN_ACCEPT is received.

If the PROTOCOL field in the transmitted OPEN address frame was set to STP, then this state shall send a Connection Opened (STP, Source Opened) confirmation to the port layer before the transition.This transition shall pass a parameter indicating an STP connection has been opened to the SL3:Connected state. At this point an STP connection has been opened between the source ATA phy and the destination ATA phy.

If the PROTOCOL field in the transmitted OPEN address frame was set to SSP, then this state shall send a Connection Opened (SSP, Source Opened) confirmation to the port layer before the transition. This transition shall pass a parameter indicating an SSP connection has been opened to the SL3:Connected state along with the hashed value of the source port identifier and the destination port identifier. At this point an SSP connection has been opened between the source SCSI phy and the destination SCSI phy.

If the PROTOCOL field in the transmitted OPEN address frame was set to SMP, then this state shall send a Connection Opened (SMP, Source Opened) confirmation to the port layer before the transition. This transition shall pass a parameter indicating an SMP connection has been opened to the SL3:Connected state along with the hashed value of the source SAS address and the destination SAS address. At this point an SMP connection has been opened between the source SAS phy and the destination SAS phy.

6.11.3.5 Transition SL1:ArbSel to SL5:BreakWait

This transition shall occur after:

a) A Stop Arb request is received from the port layer and an Open Failed (Port Layer Request) confirmation has been sent to the port layer; or
b) there is no response to the OPEN address frame within the open timeout and an Open Failed (Open Timeout Occurred) confirmation has been sent to the port layer.

6.11.3.6 Transition SL1:ArbSel to SL6:Break

This transition shall occur after:

a) a BREAK primitive sequence is received; and
b) an Open Failed (Break Received) confirmation has been sent to the port layer.

6.11.4 SL2:Selected state

6.11.4.1 State description

This state completes the establishment of an SSP, SMP, or STP connection when an incoming connection request has won arbitration by transmitting OPEN_ACCEPT or OPEN_REJECT.

This state shall be exited after:

a) transmitting an OPEN_ACCEPT;
b) transmitting an OPEN_REJECT; or
c) receiving a BREAK primitive sequence.

While in this state, the SAS port accepts opening a connection between it and the destination SAS port by transmitting an OPEN_ACCEPT.

This state may reject opening a connection by transmitting an OPEN_REJECT.

6.11.4.2 Transition SL2:Selected to SL0:Idle

This state shall parse the OPEN address frame in this order:

1) If the OPEN address frame DESTINATION SAS ADDRESS field does not match the SAS address of this device, this state shall transmit OPEN_REJECT (WRONG DESTINATION) and transition to SL0:Idle;
2) If the OPEN address frame PROTOCOL field indicates a protocol which is not supported, this state shall transmit OPEN_REJECT (PROTOCOL NOT SUPPORTED), and transition to SL0:Idle; and
3) If an Accept_Reject Opens (Reject) request is received from the port layer, this state shall transmit OPEN_REJECT (RETRY) and transition to SL0:Idle.

6.11.4.3 Transition SL2:Selected to SL3:Connected

If the requested protocol is SSP and the port layer is not sending an Accept_Reject Opens (Reject) request then this transition shall occur after:

a) transmitting an OPEN_ACCEPT; and
b) sending a Connection Opened (SSP, Destination Opened) confirmation to the port layer.

At this point an SSP connection has been opened between the source SCSI phy and the destination SCSI phy.

If the requested protocol is SMP and the port layer is not sending an Accept_Reject Opens (Reject) request then this transition shall occur after:

a) transmitting an OPEN_ACCEPT; and
b) sending a Connection Opened (SMP, Destination Opened) confirmation to the port layer.

At this point an SMP connection has been opened between the source SAS phy and the destination SAS phy.

If the requested protocol is STP and the port layer is not sending an Accept_Reject Opens (Reject) request then this transition shall occur after:

a) transmitting an OPEN_ACCEPT; and
b) sending a Connection Opened (STP, Destination Opened) confirmation to the port layer.

At this point an STP connection has been opened between the source SATA phy and the destination SATA phy.

6.11.4.4 Transition SL2:Selected to SL6:Break

This transition shall occur after a BREAK primitive sequence is received.

6.11.5 SL3:Connected state

6.11.5.1 State description

This state enables the SSP, STP, or SMP link layer state machine to transmit dwords during a connection.

If the connection is SMP then this state shall send an Enable Disable SMP (Enable) parameter to the SMP link layer state machines (see 6.12.4).

If the connection is SSP then this state shall send an Enable Disable SSP (Enable) parameter to the SSP link layer state machines (see 6.11.9).

If the connection is STP then this state shall send an Enable Disable STP (Enable) parameter to the STP link layer state machines.

This state shall be exited after receiving a Request Break or Request Close parameter from the enabled link layer state machine, or after receiving a BREAK primitive sequence.

This state transmits idle dwords until the SSP, SMP, or STP link layer state machine starts transmitting.

6.11.5.2 Transition SL3:Connected to SL4:DisconnectWait

This transition shall occur after receipt of a Request Close parameter.

For an SSP connection, the Request Close parameter is received from the SSP link layer state machine.

For an SMP connection, the Request Close parameter is received from the SMP link layer state machine.

For an STP connection, the Request Close parameter is received from the STP link layer state machine.

6.11.5.3 Transition SL3:Connected to SL5:BreakWait

This transition shall occur after receipt of a Request Break parameter.

For an SSP connection, the Request Break parameter is received from the SSP link layer state machine.

For an SMP connection, the Request Break parameter is received from the SMP link layer state machine.

For an STP connection, the Request Break parameter is received from the STP link layer state machine.

6.11.5.4 Transition SL3:Connected to SL6:Break

This transition shall occur after a BREAK primitive sequence is received and after this state has sent a Connection Closed (Break Received) confirmation to the port layer.

6.11.6 SL4:DisconnectWait state

6.11.6.1 State description

This state closes the connection and releases all resources associated with the connection.

This state:

1) shall transmit a CLOSE primitive sequence;
2) transmit at least three idle dwords; and
3) initialize a close timeout timer to 1 ms and start the timer.

This state shall be exited after a CLOSE primitive sequence is received, a BREAK primitive sequence is received, or after the close timeout timer is exceeded. The CLOSE primitive sequence may be received at any time while in this state.

6.11.6.2 Transition SL4:DisconnectWait to SL0:Idle

This transition shall occur after:

a) transmitting a CLOSE primitive sequence;
b) receiving a CLOSE primitive sequence; and
c) sending a Connection Closed (Normal) confirmation to the port layer.

6.11.6.3 Transition SL4:DisconnectWait to SL5:BreakWait

This transition shall occur when there is no response to a CLOSE primitive sequence transmitted within a close timeout and after the state machine has sent a Connection Closed (Close Timeout) confirmation to the port layer.

6.11.6.4 Transition SL4:DisconnectWait to SL6:Break

This transition shall occur after a BREAK primitive sequence is received and after this state has sent a Connection Closed (Break Received) confirmation to the port layer.

6.11.7 SL5:BreakWait state

6.11.7.1 State description

This state closes the connection if one is established and releases all resources associated with the connection.

This state shall be exited after a BREAK primitive sequence is received or the break timeout is exceeded. The BREAK primitive sequence may be received at any time while in this state.

This state shall:

1) transmit a BREAK primitive sequence;
2) transmit at least six idle dwords; and
3) initialize a break timeout timer to 1 ms and start the timer.

6.11.7.2 Transition SL5:BreakWait to SL0:Idle

This transition shall occur after receiving a BREAK primitive sequence or if the break timeout is exceeded. If a BREAK primitive sequence is not received before the timer is exceeded, this state shall send a Connection Closed (Link Broken) confirmation to the port layer before making this transition.

6.11.8 SL6:Break state

6.11.8.1 State description

This state closes any connection and releases all resources associated with this connection.

This state shall transmit a BREAK primitive sequence.

While in this state all primitives received shall be ignored.

Editor's Note 42: Consider BREAKs crossing on the wire

6.11.8.2 Transition SL6:Break to SL0:Idle

This transition shall occur after transmitting a BREAK primitive sequence.

6.11.9 SSP link layer (SSP) state machines

6.11.9.1 Overview

The SSP link layer contains several state machines that run in parallel to control the flow of dwords on the link during an SSP connection. The SSP link state machines are as follows:

~~a) SAS Phy Receiver (SSP_R state machine);~~
~~b) SAS Phy Transmitter (SSP_T state machine);~~
a) Transmit Interlocked Frame Monitor (SSP_TIM state machine);
b) Transmit Frame Credit Monitor (SSP_TCM state machine);
c) DONE Control (SSP_D state machine);
d) Transmit Frame Control (SSP_TF state machine);
e) Receive Frame Control (SSP_RF state machine);
f) Receive Frame Credit Monitor (SSP_RCM state machine);
g) Receive Interlocked Frame Monitor (SSP_RIM state machine);
h) Transmit Credit Control (SSP_TC state machine); and
i) Transmit ACK/NAK Control (SSP_TAN state machine).

All the SSP state machines shall begin after receiving an Enable Disable SSP parameter with an argument of Enable from the SL state machine (see 6.11).

All the SSP state machines shall stop after:

a) receiving an Enable Disable SSP parameter with an argument of Disable from the SL state machine;
b) receiving a Request Close parameter from the SSP_D1:DONE_Wait state indicating that the connection shall be closed; or
c) receiving a Request Break parameter from the SSP_D1:DONE_Wait state indicating that a BREAK primitive sequence shall be transmitted.

If a state machine consists of multiple states the initial state is as indicated in the state machine description in this subclause.

~~The SSP_R state machine's function is to receive primitives and frames and indicate to other SSP state machines the receipt of those dwords. The SSP_R state machine contains the SSP_R1:Receive state (see 6.11.9.2).~~

The SSP_TIM state machine's function is to ensure that ACKs or NAKs are received for each transmitted frame before the ACK/NAK timeout. The SSP_TIM state machine contains the SSP_TIM1:Tx_Interlock_Monitor state (see 6.11.9.4).

The SSP_TCM state machine's function is to ensure that credit is available from the originator before a frame is transmitted. The SSP_TCM state machine contains the SSP_TCM1:Tx_credit_monitor state (see 6.11.9.5).

The SSP_D state machine's function is to ensure a DONE has been received and transmitted before the SL state machine disables the SSP state machines. The SSP_D state machine contains the SSP_D1:Done_Wait state (see 6.11.9.6).

The SSP_TF state machine's function it to control when the SSP_T state machine transmits an SOF, frame dwords, EOF, and a DONE. The SSP_TF state machine contains the following states:

a) Initial state: SSP_TF1:Connected_Idle state (see 6.11.9.7);
b) SSP_TF2:Tx_Wait state (see 6.11.9.8);
c) SSP_TF3:Indicate_Frame_Tx state (see 6.11.9.9); and
d) SSP_TF4:Indicate_DONE_Tx state (see 6.11.9.10).

The SSP_RF state machine's function is to receive frames and to indicate the successful or unsuccessful receipt of those frames. The SSP_RF state machine contains the SSP_RF1:Rcv_Frame state (see 6.11.9.11).

The SSP_RCM state machine's function is to ensure that there was credit given to the originator for every frame that is received. The SSP_RCM state machine contains the SSP_RCM1:Rcv_Credit_Monitor state (see 6.11.9.12).

The SSP_RIM state machine's function is to indicate to the SSP_RF1:Rcv_Frame state when the number of ACKs and NAKs transmitted equals the number of the EOFs received. The SSP_RIM state machine contains the SSP_RIM1:Rcv_Interlock_Monitor state (see 6.11.9.13).

The SSP_TC state machine's function it to control when the SSP_T state machine transmits an RRDY or CREDIT_BLOCKED. The SSP_TC state machine contains the following states:

a) Initial state: SSP_TC1:Idle state (see 6.11.9.14); and
b) SSP_TC2:Indicate_Credit_Tx state (see 6.11.9.15).

The SSP_TAN state machine's function is to control when the SSP_T state machine transmits an ACK or NAK. The SSP_TAN state machine contains the following states:

a) Initial state: SSP_TAN1:Idle state (see 6.11.9.16); and
b) SSP_TAN2:Indicate_ACK/NAK_Tx state (see 6.11.9.17).

Figure 57 shows the SSP states related to frame transmission.

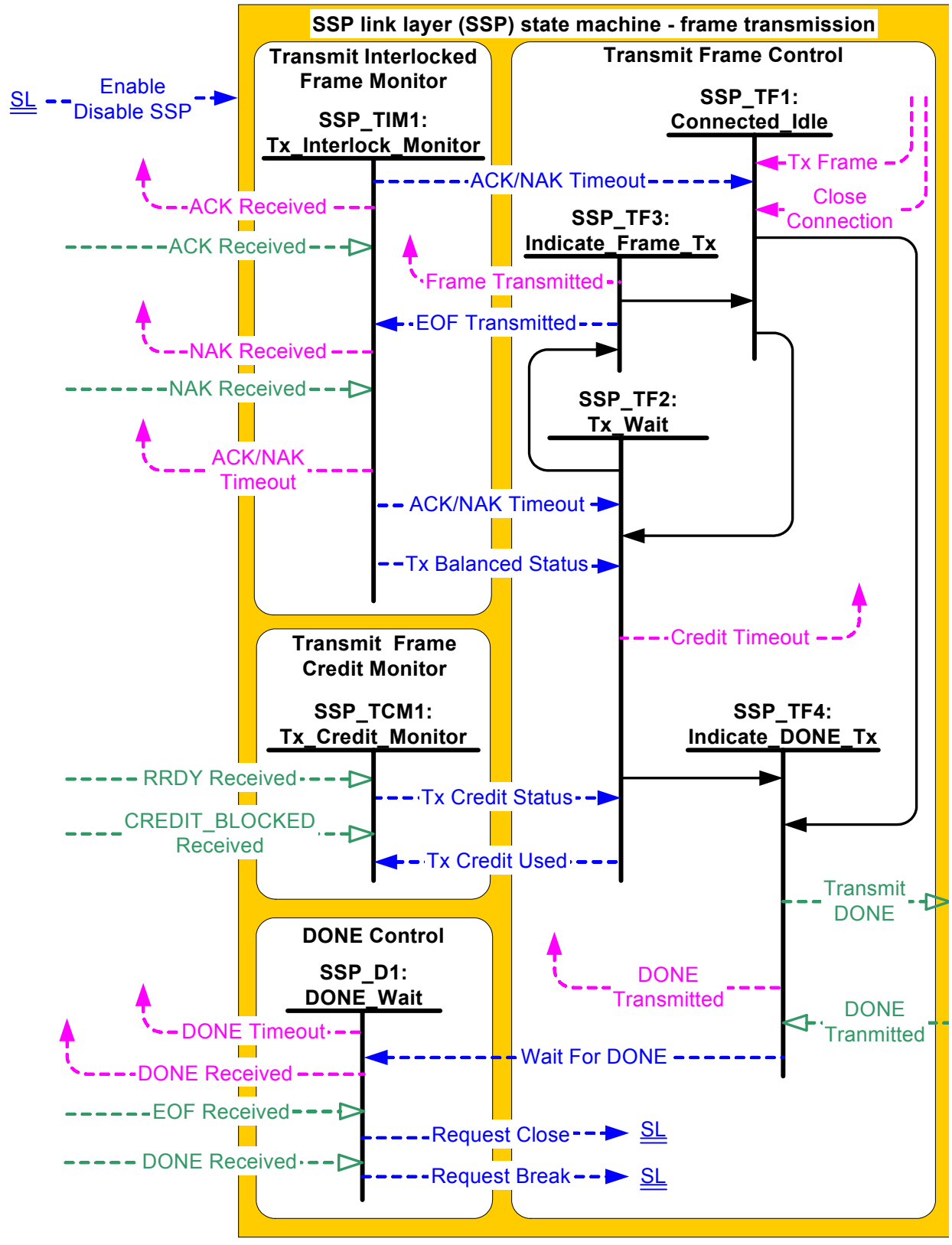

Figure 57 — SSP link layer (SSP) state machines (part 1 - frame transmission)

Figure shows the SSP states related to frame reception.SSP link layer (SSP) state machines (part 2 - frame reception)

Figure shows the SSP states related to primitive transmission.SSP link layer (SSP) state machines (part 3 - primitive transmission)

~~6.11.9.2 SSP_R1:Receive state~~

~~6.11.9.2.1 State description~~

~~This state receives frames and primitives.~~

~~As a result of receiving an ACK or a NAK this state shall send an ACK Received parameter or a NAK Received parameter to SSP_TIM1:Tx_Interlock_Monitor state.~~

~~As a result of receiving a RRDY or a CREDIT_BLOCKED this state shall send an RRDY Received parameter or a CREDIT_BLOCKED Received parameter to SSP_TCM1:Tx_Credit_Monitor state.~~

~~As a result of receiving a DONE (CLOSE CONNECTION) this state shall send a DONE Received parameter to the following states:~~

~~a) SSP_D1:DONE_Wait state; and~~
~~b) SSP_RF1:Frame_Rcv state.~~

~~As a result of receiving a DONE (CREDIT TIMEOUT) this state shall send a DONE Received parameter to the following states:~~

~~a) SSP_D1:DONE_Wait state; and~~
~~b) SSP_RF1:Frame_Rcv state.~~

~~As a result of receiving a DONE (ACK/NAK TIMEOUT) this state shall send a DONE Received parameter to the following states:~~

~~a) SSP_D1:DONE_Wait state; and~~
~~b) SSP_RF1:Frame_Rcv state.~~

~~This state shall also send a DONE Received confirmation to the port layer. This confirmation is a warning to the port layer that the transmitter is going to close the connection within 1 ms.~~

~~As a result of receiving a SOF this state shall send a SOF Received parameter to the SSP_RF1:Frame_Rcv state.~~

~~As a result of receiving an EOF this state shall send an EOF Received parameter to the following states:~~

~~a) SSP_D1:DONE_Wait state; and~~
~~b) SSP_RF1:Frame_Rcv state.~~

~~6.11.9.2.2 Transition SSP_R1:Receive to SSP_R1:Receive~~

~~This transition shall occur:~~

~~a) every time a data dword associated with a frame is received;~~
~~b) every time an ACK or a NAK is received;~~
~~c) every time a RRDY or a CREDIT_BLOCKED is received;~~
~~d) every time a DONE is received;~~
~~e) every time a SOF is received; and~~
~~f) every time an EOF is received.~~

~~6.11.9.3 SSP_T1:Transmit state~~

~~6.11.9.3.1 State description~~

~~This state transmits frames and primitives.~~

~~On a request to transmit, this state shall transmit the indicated dword. If there are multiple requests to transmit primitives, the following priority should be followed when selecting the primitives:~~

~~1) ALIGN;~~
~~2) ACK or NAK;~~

3) RRDY;
4) CREDIT_BLOCKED;
5) DONE; then
6) SOF/frame/EOF.

In the absence of any transmit requests this state shall transmit idle dwords and ALIGNs as necessary.

When the Transmit ACK parameter is sent by the SSP_TAN2:Indicate_ACK/NAK_Tx state, this state shall transmit an ACK and shall send the ACK Transmitted parameter to the SSP_TAN2:Indicate_ACK/NAK_Tx state.

When the Transmit NAK parameter is sent by the SSP_TAN2:Indicate_ACK/NAK_Tx state, this state shall transmit a NAK and shall send the NAK Transmitted parameter to the SSP_TAN2:Indicate_ACK/NAK_Tx state.

When the Transmit RRDY parameter is sent by the SSP_TC2:Indicate_Credit_Tx state, this state shall transmit a RRDY and shall send the RRDY Transmitted parameter to the SSP_TC2:Indicate_Credit_Tx state.

When the Transmit CREDIT_BLOCKED parameter is sent by the SSP_TC2:Indicate_Credit_Tx state, this state shall transmit a CREDIT_BLOCKED and shall send the CREDIT_BLOCKED Transmitted parameter to the SSP_TC2:Indicate_Credit_Tx state.

When the Transmit DONE parameter is sent by the SSP_TF4:Indicate_DONE_Tx state, this state shall transmit a DONE and shall send the DONE Transmitted parameter to the SSP_TF4:Indicate_DONE_Tx state.

When the Transmit SOF/frame/EOF parameter is sent by the SSP_TF3:Indicate_Frame_Tx state, this state shall transmit an SOF, the data dwords of the frame, and an EOF. If during the transmission of a frame an indication that a primitive is to be transmitted occurs, this state may transmit the indicated primitive by inserting the primitive between the frames' dwords. This state shall send a SOF/frame/EOF Transmitted parameter to the SSP_TF3:Indicate_Frame_Tx state each time an EOF is transmitted.

6.11.9.3.2 Transition SSP_T1:Transmit to SSP_T1:Transmit

This transition shall occur:

a) every time an ALIGN is transmitted;
b) every time an ACK or a NAK is transmitted;
c) every time an RRDY is transmitted;
d) every time an CREDIT_BLOCKED is transmitted;
e) every time a DONE is transmitted; and
f) every time an SOF, frame dword, or EOF is transmitted;
g) every time an idle is transmitted.

6.11.9.4 SSP_TIM1:Tx_Interlock_Monitor state

This state monitors the number of frames transmitted and monitors the number of ACKs and NAKs received. This state ensures that an ACK or NAK is received for each frame transmitted and indicates a timeout if they are not.

When the number of frames transmitted equals the number of ACKs and NAKs received then there is balance and this state is balanced and shall send the Tx Balanced Status parameter with the Balanced argument to the SSP_TF2:Tx_Wait state. When the number of frames transmitted does not equal the number of ACKs and NAKs received then this state there is not balanced an unbalance and this state shall send the Tx Balanced Status parameter with the Not Balanced argument to the SSP_TF2:Tx_Wait state.

The EOF Transmitted parameter received from the SSP_TF3:Indicate_Frame_Tx state shall be used by this state to count the number of frames transmitted.

If there is an unbalance and an ACK Received parameter is received this state shall:

a) ~~If this state is not balanced,~~ use the ACK Received parameter ~~from the SSP_R1:Receive state shall be used by this state~~ to count the number of ACKs and NAKs ~~received~~ received; and ~~shall be used to send an ACK Received confirmation to the port layer.~~
b) send an ACK Received confirmation to the port layer each time the ACK Received parameter is received.

If there is an unbalance and an NAK Received parameter is received this state shall:

a) ~~If this state is not balanced,~~ use the NAK Received parameter ~~from the SSP_R1:Receive state shall be used by this state~~ to count the number of ACKs and NAKs ~~received~~ received; and ~~shall be used to send a NAK Received confirmation to the port layer.~~
b) send an NAK Received confirmation to the port layer each time the NAK Received parameter is received.

If ~~this state~~ there is ~~balanced~~balance, the ACK Received parameter and NAK Received parameter ~~from the SSP_R1:Receive state~~ shall be ignored.

~~When this state~~ If there is ~~balanced~~ balance the ACK/NAK timeout timer shall be disabled.

Each time ~~this state becomes not balanced~~an unbalance occurs, the ACK/NAK timeout timer shall be initialized to 1 ms and shall start timing. The timer shall be re-initialized to 1 ms each time an ACK or NAK is counted. If the timer expires, this state shall send the ACK/NAK Timeout confirmation to the port layer and to the following states:

a) SSP_TF1:Connected_Idle; and
b) SSP_TF2:Tx_Wait state.

When the SL state machine sends the Enable Disable SSP parameter with an argument of Enable, the number of frames transmitted shall be set to the number of ACKs and NAKs received.

6.11.9.5 SSP_TCM1:Tx_credit_monitor state

This state shall keep track of the number of transmit frame credits received versus the number of transmit frame credits used. This state adds transmit frame credit for each RRDY Received parameter ~~indication from the SSP_R1:Receive state~~ received and subtracts transmit frame credit for each Tx Credit Used parameter ~~indication~~ received from the SSP_TF2:Tx_Wait state. This state shall remember ~~the~~ any CREDIT_BLOCKED Received parameter ~~indication from the SSP_R1:Receive state~~that is receives.

When transmit frame credit is available, this state shall send the Tx Credit Status parameter with the Credit Available argument to the SSP_TF2:Tx_Wait state.

When transmit frame credit is not available and credit is not blocked, this state shall send the Tx Credit Status parameter with the Credit Not Available argument to the SSP_TF2:Tx_Wait state.

When transmit frame credit is not available and credit is blocked, this state shall send the Tx Credit Status parameter with the Credit Blocked argument to the SSP_TF2:Tx_Wait state.

When the SL state machine sends the Enable Disable SSP parameter with an argument of Enable, transmit frame credit shall be set to not available and credit shall not be blocked.

6.11.9.6 SSP_D1:DONE_Wait state

6.11.9.6.1 State description

This state ensures that a DONE is received and transmitted before the connection is closed. The DONE may be transmitted and received in any order.

If the DONE Received parameter has been received ~~from the SSP_R1:Receive state~~ when the Wait For DONE Parameter is received from the SSP_TF4:Indicate_DONE_Tx state, this state shall send the Request Close parameter to the SL state machine (see 6.11) to indicate that a CLOSE shall be transmitted.

If the DONE Received parameter has not been received ~~from the SSP_R1:Receive state~~ when the Wait For DONE Parameter is received from the SSP_TF4:Indicate_DONE_Tx state, this state shall initialize the DONE timeout timer to 1 ms. If the Wait For DONE parameter was received with the Close Connection argument or the Credit Timeout argument, the DONE timeout timer shall be re-initialized to 1 ms each time the EOF Received parameter is received from the SSP_R1:Receive state.

If the DONE Received parameter is received ~~from the SSP_R1:Receive state~~ before the DONE timeout timer expires, this state shall send the Request Close parameter to the SL state machine.

If the DONE Received parameter is not received ~~from the SSP_R1:Receive state~~ before the DONE timeout timer expires, this state shall:

a) send a DONE Timeout confirmation to the port layer; and
b) send a Request Break parameter to the SL state machine indicating that a BREAK shall be transmitted.

Any time a DONE received parameter is received this state shall send a DONE Received confirmation to the port layer. This confirmation is a warning to the port layer that the transmitter is going to close the connection within 1 ms.

Editor's Note 43: Is the DONE received confirmation to the port layer really needed?

6.11.9.7 SSP_TF1:Connected_idle state

6.11.9.7.1 State description

This state waits for a request from the port layer to transmit a frame or to close the connection.

6.11.9.7.2 Transition SSP_TF1:Connected_Idle to SSP_TF2:Tx_Wait

This transition shall occur when the Tx Frame request is received from the port layer indicating that a frame be transmitted or a Close Connection request is received from the port layer indicating the connection to be closed.

~~The Balanced or Not Balanced arguments to the Tx Frame request are sent to the SSP_TF2:Tx_Wait state.~~

If the request from the port layer was a Tx Frame request that passed a Balanced argument this transition shall pass a Transmit Balanced Frame argument.

If the request from the port layer was a Tx Frame request that passed a Nonbalanced argument this transition shall pass a Transmit Unbalanced Frame argument.

If the request from the port layer was a Close Connection request this transition shall pass a Close Connection argument.

6.11.9.7.3 Transition SSP_TF1:Connected_Idle to SSP_TF4:Indicate_Done_Tx

This transition shall occur when the ACK/NAK Timeout parameter is sent by the SSP_TIM1:Tx_Interlock_Monitor state requesting the connection ~~to~~ be closed. This transition shall indicate ACK/NAK Timeout.

6.11.9.8 SSP_TF2:Tx_Wait state

6.11.9.8.1 State description

This state monitors the Tx Balanced Status parameter from the SSP_TIM1:Tx_Interlock_Monitor state and the Tx Credit Status from the SSP_TCM1:Tx_Credit_Monitor state to ensure that frames are transmitted and connections are closed at the proper time.

If the transition from SSP_TF1:Connected_Idle state ~~indicated that~~ passes a Transmit Balanced Frame argurment or a ~~frame is to be transmitted~~ Transmit Nonbalanced Frame argument and the last Tx Credit Status parameter received has an argument of Not Available, this state shall initialize the Credit timeout timer to 1 ms. The Credit timeout timer shall be disabled if the Tx Credit Status parameter has an argument other than Not Available.

6.11.9.8.2 Transition SSP_TF2:Tx_Wait to SSP_TF3:Indicate_Frame_Tx

If the SSP_TF1:Connected_Idle state ~~indicated that the frame is to be transmitted Balanced~~passes a Transmit Balanced Frame argument, this transition shall occur if the last Tx Balanced Status parameter ~~has~~ received had an argument of Balanced and the last Tx Credit Status parameter ~~has~~ received had an argument of Credit Available.

If the SSP_TF1:Connected_Idle state ~~indicated that the frame is to be transmitted Not Balanced~~passes a Transmit Nonbalanced Frame argument, this transition shall occur if the last Tx Credit Status parameter ~~has~~ received had an argument of Credit Available.

This transition shall cause this state to send a Tx Credit Used parameter to the SSP_TCM1:Tx_Credit_Monitor state.

6.11.9.8.3 Transition SSP_TF2:Tx_Wait to SSP_TF4:Indicate_Done_Tx

This transition shall ~~always~~ occur if an ACK/NAK Timeout parameter is received from the SSP_TIM1:Tx_Interlock_Monitor state. This transition shall ~~indicate~~ pass an ACK/NAK ~~Timeout~~Timeout argument to the Indicate_Done_Tx state.

If the SSP_TF1:Connected_Idle state ~~indicated that the connection is to be closed~~passes a Close Connection argument, this transition shall occur if the last Tx Balanced Status parameter ~~has~~ received had an argument of Balanced. This transition shall ~~indicate~~ pass an Connection ~~Closed~~Closed argument to the Indicate_Done_Tx state.

If the SSP_TF1:Connected_Idle state ~~indicated that~~ passes a Transmit Balanced Frame argument or a ~~frame is to be transmitted~~Transmit Nonbalanced Frame argument, this transition shall occur if the last Tx Credit Status parameter ~~has~~ received had an argument of Blocked. This transition shall ~~indicate~~ pass a Credit ~~Timeout~~Timeout argument to the Indicate_Done_Tx state.

If the SSP_TF1:Connected_Idle state ~~indicated that~~ passes a Transmit Balanced Frame argument or a ~~frame is to be transmitted~~Transmit Nonbalanced Frame argument, this transition shall occur if the Credit timeout timer expires before the Tx Credit Status parameter ~~has~~ is received with an argument of Available. This transition shall ~~indicate~~ pass a Credit Timeout Credit ~~Timeout~~Timeout argument to the Indicate_Done_Tx state.

6.11.9.9 SSP_TF3:Indicate_Frame_Tx state

6.11.9.9.1 State description

This state requests a frame transmission by ~~indicating~~ sending a Transmit SOF/frame/EOF parameter to ~~the SSP_T1:Transmit state~~a transmitter. Each time a Transmit SOF/frame/EOF parameter is ~~indicated~~ sent to ~~the SSP_T1:Transmit state~~a transmitter, one SOF/frame/EOF is transmitted.

6.11.9.9.2 Transition SSP_TF3:Indicate_Frame_Tx to SSP_TF1:Connected_idle

This transition shall occur after:

a) an SOF/frame/EOF Transmitted parameter has been received from the SSP_T1:Transmit state indicating that the frame has been transmitted; ~~and~~
b) an EOF Transmitted parameter has been sent to the SSP_TIM1:Tx_Interlock_Monitor state indicating that a frame has been transmitted~~.~~; and
c) a Frame Transmitted confirmation is sent to the port layer.

6.11.9.10 SSP_TF4:Indicate_Done_Tx state

This state requests a DONE be transmitted by sending a Transmit DONE parameter ~~to the SSP_T1:Transmit state~~ with one of the following ~~arguments~~arguments to a transmitter:

a) Close Connection if the transition into this state indicated a port layer has requested the connection be closed;
b) ACK/NAK Timeout if the transition into this state indicated an ACK/NAK timeout occurred; or
c) Credit Timeout if the transition into this state indicated a credit timeout occurred.

After a DONE transmitted parameter is received ~~from the SSP_T1:Transmit state~~ this state shall:

a) send the Wait for DONE parameter to the SSP_D1:DONE_wait state to indicate that no more frames are going to be transmitted during the current connection with a argument of Close Connection, ACK/NAK Timeout, or Credit Timeout; and
b) send the DONE Transmitted confirmation to the port layer indicating that a DONE was transmitted.

6.11.9.11 SSP_RF1:Rcv_Frame state

This state:

a) checks the frame to determine if the frame should be accepted or discarded by the link:
b) ~~This state~~ checks the frame ~~to determine if the frame should be accepted or discarded by the link,~~ to determine if an ACK or NAK should be transmitted~~, and indicates the Received Frame parameter.~~; and
c) sends a Received Frame parameter.

The frame (i.e., all the dwords between an SOF and EOF) shall be discarded if any of the following conditions are true:

a) the number of bytes between the SOF and EOF is less than 28;
b) the number of bytes after the SOF is greater than 1 052 bytes;
c) the Rx Credit Status parameter is received from the SSP_RCM1:Rcv_Credit_Monitor state has an argument of Credit Exhausted; or
d) the ~~SSP_R1:Receive state sent a~~ DONE Received ~~parameter~~parameter is received.

If consecutive SOF Received parameters are ~~sent by the SSP_R1:Receive state~~ received without an intervening EOF Received parameter (i.e., SOF, data dwords, SOF, data dwords, and EOF instead of SOF, data dwords, EOF, SOF, data dwords, and EOF) then this state shall discard all dwords between ~~the~~ those SOFs

If the frame is discarded then no further action is taken by this state. If the frame is not discarded then this state shall:

a) send the Frame Received parameter to the SSP_RCM1:Rcv_Credit_Monitor state; and
b) send the Frame Received parameter to the SSP_RIM1:Rcv_Interlock_Monitor state;

If the frame CRC is good and the frame contained no invalid dwords, this state shall:

a) send the Frame Received parameter to the SSP_TAN1:Idle state state with an argument of Successful; and
b) send the Frame Received confirmation to the port layer with an argument of ACK/NAK Balanced or ACK/NAK Not Balanced as determined by the last Rx Balanced Status parameter received from the SSP_RIM1:Rcv_Interlock_Monitor state~~;~~.

Editor's Note 44: This state should not send Frame Received confirmation. The state machine should wait until it sends the ACK for the frame (from TAN2 after ACK Received or NAK Received arrives from the Transmitter state machine). Otherwise, if the connection is lost before the ACK is sent (CLOSE or BREAK), the upper level won't know anything is wrong with the frame. Also, the ACK/NAK Balanced count is off by 1 if you don't wait.

If the frame CRC is bad or the frame contained invalid dwords, this state shall send the Frame Received parameter to the SSP_TAN1:Idle state state with an argument of Unsuccessful.

6.11.9.12 SSP_RCM1:Rcv_Credit_Monitor state

This state monitors the receiver's resources and keeps track of the number of RRDYs transmitted versus the number of frames received.

Any time resources are released or become available this state shall send the Rx Credit Status parameter with an Available argument to the SSP_TC1:Idle state. This state shall only send the Rx Credit Status parameter with an Available argument to the SSP_TC1:Idle state after frame receive resources become available. The specifications for when or how resources become available is outside the scope of this standard.

This state may send the Rx Credit Status parameter with a Blocked argument to the SSP_TC1:Idle state to indicate that no more credit is going to be sent during this connection. After sending the Rx Credit Status parameter with a Blocked argument to the SSP_TC1:Idle state, this state shall not send the Rx Credit Status parameter with an Available argument to the SSP_TC1:Idle state for the duration of the current connection. The Rx Credit Status parameter with a Blocked argument should be sent to the SSP_TC1:Idle state when no further credit is going to become available within a Credit Timeout (i.e. less than one millisecond).

This state shall ~~only~~ indicate through the Rx Credit Status parameter only the amount of resources available to handle received frames (e.g., if this state has resources for 5 frames the maximum number of Rx Credit Status requests with the Available argument outstanding is 5).

This state shall use the RRDY Transmitted parameter from the SSP_TC1:Idle state to keep track of the number of RRDYs transmitted. This state shall use the Frame Received parameter from the SSP_RF1:Rcv_Frame state to keep a track of the number of frames received.

Any time the number of RRDYs exceeds the number of frames received this state shall send a Rx Credit Status parameter with an Credit Extended argument to the SSP_RF1:Rcv_Frame state to indicate that credit has been given to the transmitter.

Any time the number of RRDYs equals the number of frames received this state shall send a Rx Credit Status parameter with an Credit Exhausted argument to the SSP_RF1:Rcv_Frame state to indicate that no credit has been given to the transmitter.

When the SL state machine sends the Enable Disable SSP parameter with an argument of Enable, there is no frame receiver resource credit sahll be initialized to the no credit value for the current connection.

6.11.9.13 SSP_RIM1:Rcv_Interlock_Monitor state

This state monitors the number of frames received versus the number of ACKs and NAKs transmitted.

This state shall use the ACK Transmitted parameter and the NAK Transmitted parameter from the SSP_TAN1:Idle state state to keep track of the number of ACKs and NAKs transmitted. This state shall use the Received Frame parameter from the SSP_RF1:Rcv_Frame state to keep a track of the number of frames received.

Any time the number of the ACKs and NAKs transmitted equals the number of frames received this state shall send the Rx Balanced Status parameter with an argument of Balanced to the SSP_RF1:Rcv_Frame state.

Any time the number of the ACKs and NAKs transmitted does not equal the number of frames received this state shall send the Rx Balanced Status parameter with an argument of Not Balanced to the SSP_RF1:Rcv_Frame state.

When the SL state machine sends the Enable Disable SSP parameter with an argument of Enable, the number of the ACKs and NAKs transmitted shall be set to the number of frames received.

6.11.9.14 SSP_TC1:Idle state

6.11.9.14.1 State description

This state waits for a Rx Credit Status parameter ~~indication~~ to be received from the SSP_RCM1:Rcv_Credit_Monitor state.

When this state is transitioned into from the SSP_TC2:Indicate_Credit_Tx state with an ~~indication that a~~ argument of RRDY ~~was transmitted~~ Transmitted it shall ~~indicate~~ send the RRDY Transmitted parameter to the SSP_RCM1:Rcv_Credit_Monitor.

~~When this state is transitioned into from the SSP_TC2:Indicate_Credit_Tx state with an indication that a CREDIT_BLOCKED was transmitted it shall indicate the CREDIT_BLOCKED Transmitted parameter to the SSP_RCM1:Rcv_Credit_Monitor.~~

6.11.9.14.2 Transition SSP_TC1:Idle to SSP_TC2:Indicate_Credit_Tx

This transition shall occur if the SSP_RCM1:Rcv_Credit_Monitor state ~~indicates~~ sends the Rx Credit Status ~~parameter with the Available argument. This transition shall indicate Transmit RRDY~~parameter.

~~This transition shall occur if the SSP_RCM1:Rcv_Credit_Monitor state indicates the Rx Credit Status parameter with the Blocked argument. This transition shall indicate Transmit CREDIT_BLOCKED.~~

If the Rx Credit Status parameter has an arguement of Available this transition shall pass a Transmit RRDY argument to the Indicate_Credit_Tx state.

If the Rx Credit Status parameter has an arguement of Blocked argument. This transition shall this transition shall pass a Transmit CREDIT_BLOCKED argument to the Indicate_Credit_Tx state.

6.11.9.15 SSP_TC2:Indicate_Credit_Tx state

6.11.9.15.1 State description

When the transition to this state ~~indicated~~ passes a Transmit ~~RRDY~~RRDY argument, this state requests a single RRDY be transmitted by ~~sending a Transmit RRDY parameter to~~ the ~~SSP_T1:Transmit state~~transmitter.

When the transition to this state ~~indicated~~ passes a Transmit CREDIT_BLOCKED, this state requests a single CREDIT_BLOCKED be transmitted by ~~sending a Transmit CREDIT_BLOCKED parameter to~~ the ~~SSP_T1:Transmit state~~transmitter.

6.11.9.15.2 Transition SSP_TC2:Indicate_Credit_Tx to SSP_TC1:Idle

This transition shall occur after receiving an RRDY Transmitted parameter or the CREDIT_BLOCKED Transmitted ~~parameter from the SSP_T1:Transmit state~~parameter.

6.11.9.16 SSP_TAN1:Idle state

6.11.9.16.1 State description

This state waits for a Frame Received parameter from the SSP_RF1:Rcv_Frame state.

When the transitioned into this state ~~is transitioned into~~ from the SSP_TAN2:Indicate_ACK/NAK_Tx state ~~with~~ passes an ~~indication that an~~ ACK ~~was transmitted~~ Transmitted arguement it shall send the ACK Transmitted parameter to the SSP_RIM1:Rcv_Interlock_Monitor state.

When the transitioned into from this state ~~is transitioned into~~ from the SSP_TAN2:Indicate_ACK/NAK_Tx state ~~with an indication that~~ passes an NAK ~~was transmitted~~ Transmitted arguement it shall send the NAK Transmitted parameter to the SSP_RIM1:Rcv_Interlock_Monitor state.

6.11.9.16.2 Transition SSP_TAN1:Idle to SSP_TAN2:Indicate_ACK/NAK_Tx

~~This transition shall occur if the SSP_RF1:Rcv_Frame state indicates the Frame Received parameter.~~

This transition shall occur when a Frame Received parameter is received from the SSP_RF1:Rcv_Frame state.

If the Frame Received parameter has an argument of Successful, this transition shall ~~indicate~~ send a Transmit ~~ACK~~ACK argument to the Indicate_ACK/NAK_Tx state. If the Frame Received parameter has an argument of Unsuccessful, this transition shall ~~indicate~~ send a Transmit ~~NAK~~NAK argument to the Indicate_ACK/NAK_Tx state.

If multiple requests for transmitting ACKs and or NAKs occur, then the order in which the ACK and NAK transmissions are requested shall be the same order as the indications for transmitting the ACKs and NAKs were received.

6.11.9.17 SSP_TAN2:Indicate_ACK/NAK_Tx state

6.11.9.17.1 State description

When the transition to this state ~~indicated~~ passes a Transmit ~~ACK~~ACK argument, this state requests a single ACK be transmitted by ~~sending a Transmit ACK parameter to~~ the ~~SSP_T1:Transmit state~~transmitter.

When the transition to this state ~~indicated~~ passes a Transmit ~~NAK~~NAK argument, this state requests a single NAK be transmitted by ~~sending a Transmit NAK parameter to~~ the ~~SSP_T1:Transmit state~~transmitter.

6.11.9.17.2 Transition SSP_TAN2:Indicate_ACK/NAK_tx to SSP_TAN1:Idle

This transition shall occur after receiving an ACK Transmitted parameter or the NAK Transmitted ~~parameter from the SSP_T1:Transmit state~~parameter.

6.12 SMP link layer

6.12.1 SMP frame transmission

Inside an SMP connection, the source device transmits a single SMP_REQUEST frame and the destination device responds with a single SMP_RESPONSE frame (see 9.4).

Frames are surrounded by SOF and EOF as shown in figure 58. There is no acknowledgement of SMP frames with ACK and NAK. There is no credit exchange with RRDY

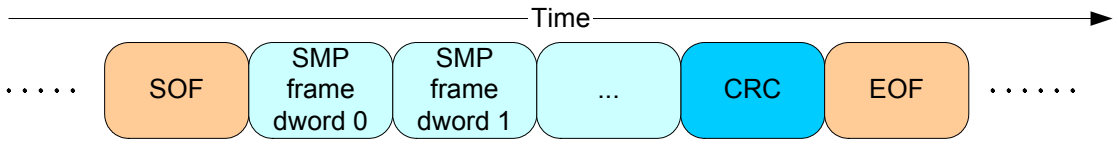

Figure 58 — SMP frame transmission

The last data dword after the SOF prior to the EOF always contains a CRC (see 7.4). The link layer shall check that the number of data dwords between the SOF and the EOF is at least 8 bytes and that the CRC is valid.

6.12.2 SMP flow control

By accepting an SMP connection, the destination device indicates it is ready to receive one SMP_REQUEST frame.

When the source device transmits one SMP_REQUEST frame, it shall be ready to receive one SMP_RESPONSE frame.

6.12.3 Preparing to close an SMP connection

After receiving the SMP_RESPONSE frame, the source device shall transmit a CLOSE primitive sequence to close the connection. The source device may leave the connection open to run loopback tests (see 7.10).

After receiving a CLOSE, the destination device shall reply with a CLOSE primitive sequence.

See 7.12.7 for details on closing connections.

6.12.4 SMP link layer (SMP) state machines

6.12.4.1 Overview

The SMP link layer state machines run in parallel to control the flow of dwords during an SMP connection. The SMP link state machines are as follows:

a) SMP Initiator Link (SMP_IL) - in SMP initiator devices;
b) SMP Target Link (SMP_TL) - in SMP target devices;
a) SAS phy receiver (SMP_R); and
b) SAS phy transmitter (SMP_T).

The SMP state machines shall begin on receipt of an Enable SMP parameter from the SL state machine (see 6.11).

All the state machines within SMP shall stop after:

a) the SMP_IL state machine sends a Request Close or Request Break parameter to the SL state machine;
b) the SMP_TL state machine sends a Request Close or Request Break parameter to the SL state machine; or
c) receiving a BREAK primitive sequence.

If a state machine consists of multiple states the initial state is as indicated in state machine description in this subclause.

The SMP_IL state machine's function is to transmit an SMP request frame and then receive the corresponding response frame. The SMP_IL state machine contains the following states:

a) Initial state: SMP_IL1:Command_idle (see 6.12.4.2);
b) SMP_IL2:Indicate_frame_tx (see 6.12.4.2.2);
c) SMP_IL3:Rcv_response_frame (see 6.12.4.2.3).

The SMP_TL state machine's function is to receive an SMP request frame and then transmit the corresponding SMP response frame. The SMP_TL state machine contains the following states:

a) Initial state:SMP_TL1:Wait_orginate_frame state (see 6.12.4.3.1); and
b) SMP_TL2:Wait_transmit_frame state (see 6.12.4.3.2).

The SMP_R state machine's function is to receive primitives and frames for the SMP_IL or SMP_TL state machine. The SMP_R state machine contains the SMP_R1:Receive state (see 6.12.4.4).

The SMP_T state machine's function is to transmit primitives and frames for the SMP_IL or SMP_TL state machine. The SMP_T state machine contains the SMP_T1:Transmit state (see 6.12.4.5).

Figure 59 shows the SMP state machines implemented by initiator devices.

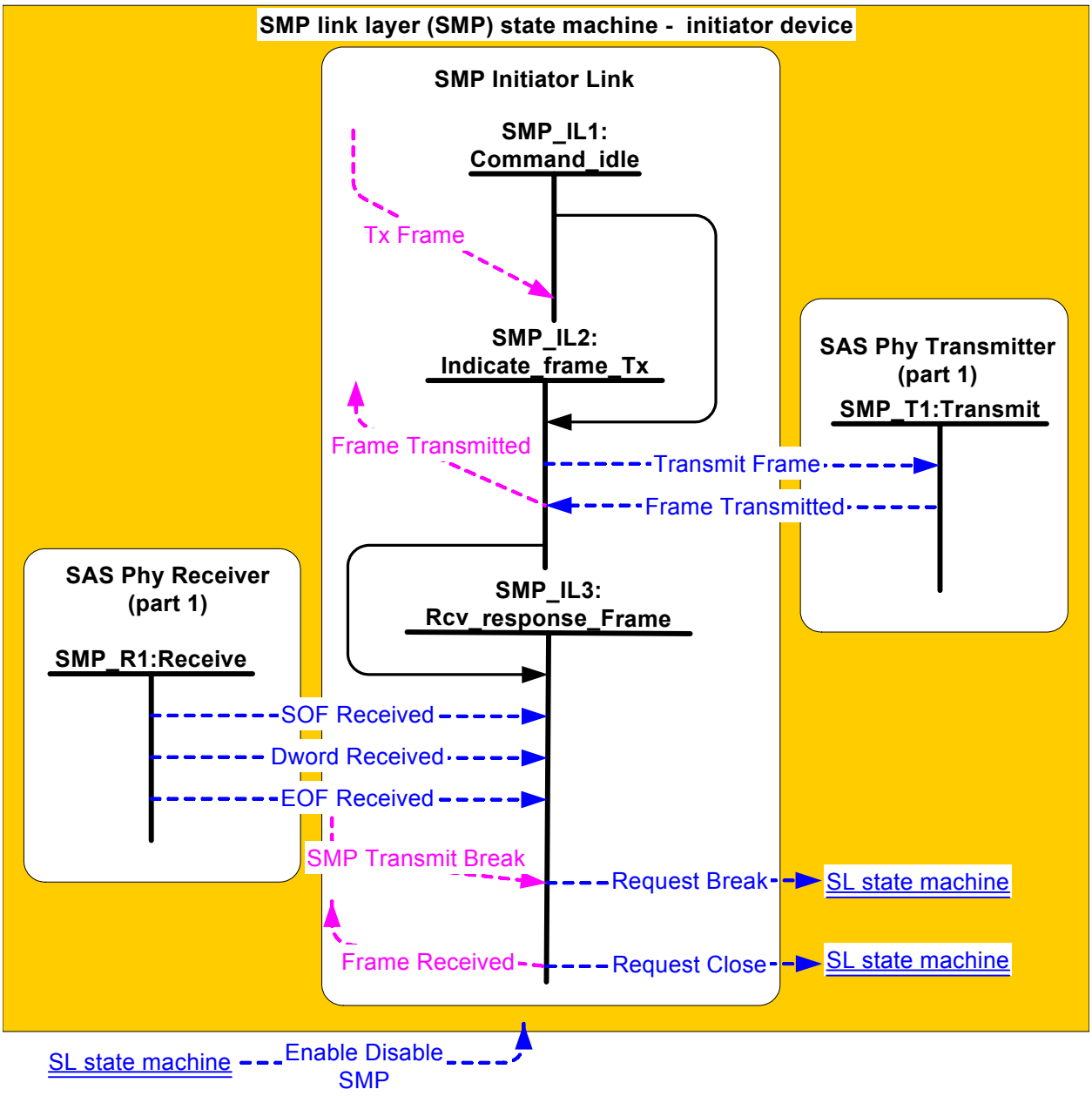

Figure 59 — SMP link layer (SMP) state machines – initiator device

Figure 60 shows the SMP state machines implemented by target devices.

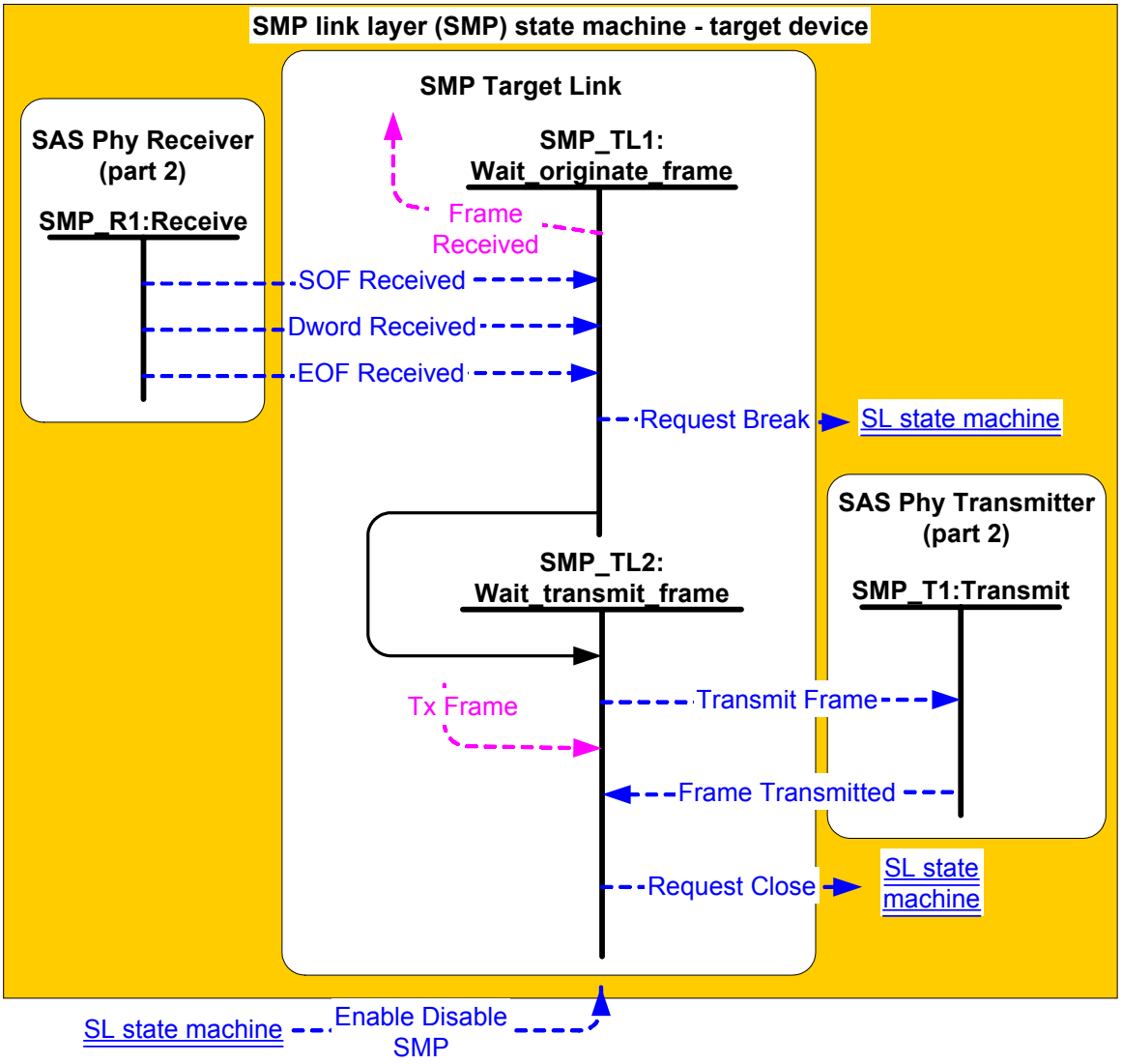

Figure 60 — SMP link layer (SMP) state machines – target device

6.12.4.2 SMP Initiator Link state machine

6.12.4.2.1 SMP_IL1:Command_idle state

6.12.4.2.1.1 State description

This state is the default state entered when an SMP connection is established.

6.12.4.2.1.2 Transition SMP_IL1:Command_idle to SMP_IL2:Indicate_frame_tx

This transition shall occur when a Tx Frame (SMP) request is received from the port layer.

6.12.4.2.2 SMP_IL2:Indicate_frame_tx state

6.12.4.2.2.1 State description

This state sends a Transmit Frame parameter to the SMP_T state machine to indicate that an SMP frame be transmitted.

After the Frame Transmitted parameter is received from the SMP_T state machine, this state shall send a Frame Transmitted confirmation to the port layer.

6.12.4.2.2.2 Transition SMP_IL2:Indicate_frame_tx to SMP_IL3:Rcv_response_frame

This transition shall occur after sending a Frame Transmitted confirmation to the port layer.

6.12.4.2.3 SMP_IL3:Rcv_response_frame state

6.12.4.2.3.1 State description

This state checks the SMP response frame and determines if the SMP response frame was successfully received (e.g., no CRC error).

If the SMP response frame is received with a CRC error, this state shall send a Frame Received (SMP Failure) confirmation to the port layer.

If the number of bytes between the SOF and EOF of the SMP response frame is less than 8, or the number of bytes after an SOF is greater than 1 052 bytes, this state shall send a Frame Received (SMP Failure) confirmation to the port layer. If the SMP response frame is received with no CRC error and the SMP response frame is valid, the rcv_response_frame state shall:

a) send a Frame Received (SMP) confirmation to the port layer; and
b) send a Request Close parameter to the SL state machine (see 6.11) to indicate that a CLOSE shall be transmitted.

If a SMP Transmit Break request is received from the port layer the rcv_response_frame state shall send a Request Break parameter to the SL state machine to indicate that a BREAK primitive sequence shall be transmitted.

6.12.4.3 SMP Target Link state machine

6.12.4.3.1 SMP_TL1:Wait_originate_frame state

6.12.4.3.1.1 State description

This state checks the originate SMP frame and determines if the SMP frame was successfully received (e.g., no CRC error).

a) If the SMP originate frame is received with a CRC error the wait_originate_frame state shall
b) send a Request Break parameter to the SL state machine (see 6.11) to indicate that a BREAK primitive sequence shall be transmitted.

If the number of data dwords between the SOF and EOF is less than 8, or the number of bytes after the SOF is greater than 1 052, the wait_originate_frame state shall send a Request Break parameter to the SL state machine to indicate that a BREAK primitive sequence shall be transmitted.

6.12.4.3.1.2 Transition SMP_TL1:Wait_originate_frame to SMP_TL2:Wait_transmit_frame

This transition shall occur after the wait_originate_frame state determines the received SMP originate frame is accepted and after the rcv_response_frame state sends a Frame Received (SMP) confirmation to the port layer.

An SMP originate frame shall be accepted if the number of bytes between the SOF and EOF is greater than or equal to 8 and less than or equal to 1 052; and the CRC is valid.

6.12.4.3.2 SMP_TL2:Wait_transmit_frame state

6.12.4.3.2.1 State description

This state, after receiving a Transmit Frame (SMP) request from the port layer, sends a Transmit Frame parameter to the transmit state indicating that an SMP response frame is to be transmitted.

After the Frame Transmitted parameter is received from the transmit state, the indicate_frame_tx state shall send a Request Close parameter to the SL state machine (see 6.11) to indicate that a CLOSE shall be transmitted.

6.12.4.4 SMP_R state machine

This state receives frames and primitives.

As a result of receiving an SOF this state shall send an SOF Received parameter to the following states:

a) SMP_TL1:Wait_originate_frame state; and
b) SMP_IL3:Rcv_response_frame state.

As a result of receiving an EOF this state shall send an EOF Received parameter to the following states:

a) SMP_TL1:Wait_originate_frame state; and
b) SMP_IL3:Rcv_response_frame state.

As a result of receiving a dword after an SOF before an EOF, this state shall send a Dword Received parameter to the following states:

a) SMP_TL1:Wait_originate_frame state; and
b) SMP_IL3:Rcv_response_frame state.

6.12.4.5 SMP_T state machine

This state transmits an SMP frame when it receive a Transmit Frame parameter from the Originate SMP Frame state machine or the SMP Frame Response state machine.

In the absence of any transmit requests, this state shall transmit idle dwords and ALIGNs as necessary.

On reception of Transmit Frame parameter, this state shall transmit:

1) SOF;
2) the dwords of the frame; and
3) EOF.

The transmit state shall send an Frame Transmitted parameter to the Originate SMP Frame state machine and the SMP Frame Response state machine each time an EOF is transmitted.