To: T10 Technical Committee
From: Rob Elliott, HP (elliott@hp.com)
Date: 25 May 2004
Subject: 04-167r0 SAS-1.1 Invalid dword handling

Revision history
Revision 0 (25 May 2004) First revision, split off from 04-115r1.

Related documents
sas1r04 - Serial Attached SCSI 1.1 revision 4
04-172 SAS-1.1 More error counters

Overview

If ALIGNs or NOTIFYs are corrupted, but they are replaced in the data stream by expander(s) with valid ERROR primitives, the elasticity buffer of an expander or the frame recipient could overflow at a later time. If the expander determines that its elasticity buffer is getting full and it is due an ALIGN, it should be allowed to choose to delete the invalid dword (assuming it might have been an ALIGN) rather than replace it with an ERROR.

If an expander is taking data into a HOLD/HOLDA buffer during an STP connection, it should not be required to remember each error. It should be allowed to delete any number of them.

Editor's Note 1: May status: WG wanted to drop this, and perhaps define a counter for elasticity buffer overflows in SMP & log page. Response: added HOLD/HOLDA case as another reason this is needed. Started a new proposal 04-172 with additional error counters.

SAS requires that frames received with invalid dwords generate a NAK, but allows incoming ERRORs to be either ignored and not generate a NAK, or generate a NAK. Invalid dwords and ERRORs should be allowed to be treated the same. There should be no functional difference if the error happened on the immediate physical link, resulting in an invalid dword being received, rather than on a remote physical link, resulting in an ERROR being received (from an expander).

Discussion of "invalid dwords and unexpected primitives" in the SL_CC state machine is inappropriate, since it only acts on messages. The SL receiver should send an Invalid Dword Received message to to be handled. Unexpected messages are ignored by the convention of the state machines. Similar changes are proposed for the XL and SSP state machines.

The "default" crutch in the SL state machine overview about sending idle dwords when there is nothing else to send belongs in the SL transmitter section. Similar changes are proposed for the XL and SSP state machines. XL shouldn't handwave on how ERRORs get forwarded through the ECR.

The SMP state machines forgot to mention invalid dwords at all. The same handling as in the other state machines is proposed.

Notes from May WG meeting

Define an SP receiver (part of SP_DWS, which already does most of this?) which sends up Dwords to the link layer. It must decide if the dword is valid or not. Pass up:

a) Valid Data Dword Received (32 bits) four valid Dxx.y
b) Valid Primitive Received (~ 32 bits) K28.5 or K28.3 + 3 valid Dxx.y
c) Invalid Dword Received (1 bit) - anything else

or

a) Valid Dword (33 bits; Z+32) and Invalid Dword (1 bit)

Usage varies per state machine. Have them behave thusly:

SL:

a) *Valid Data Dword, Valid Primitive => handled per SL_RAF*
b) *Invalid Dword => SSP_RAF may treat as a CRC error or ignore (discard)*

SSP, and SMP:

a) *Valid Data Dword, Valid Primitive => handled per SSP_RF and corresponding SMP state.*
b) *Invalid Dword => SSP_RF may generate a NAK (CRC ERROR) or may ignore (discard) and hope for a CRC error. SMP may treat as a CRC error or ignore (discard)*

XL:

a) *Valid Data Dword, Valid Primitive => forward*
b) *Invalid Dword => forward as ERROR or discard*

Suggested changes

3.1 Definitions

3.1.1 8b10b coding: A coding scheme that represents an 8-bit data byte as a 10-bit character. See 6.2.

3.1.11 byte: A sequence of eight contiguous bits considered as a unit.

3.1.12 character: A sequence of ten contiguous bits considered as a unit. A byte is encoded as a character using 8b10b coding (see 6.2).

3.1.19 control character (Kxx.y): A character that does not represent a byte of data. See 6.2.

3.1.22 data character (Dxx.y): A character representing a byte of data. See 6.2.

3.1.xx invalid character: A character that is not a valid control character (see 3.1.19) or a valid data character (see 3.1.22).

3.1.33 dword: A sequence of four contiguous bytes or four contiguous characters considered as a unit.

3.1.23 data dword: A dword that ~~starts with a Dxx.y (data character) and~~ contains four valid data characters.

3.1.93 primitive: A dword starting with a valid K28.5 or K28.3 followed by three valid data characters. See 7.2.

3.1.66 invalid dword: A dword that is not a data dword or a primitive (e.g., that contains ~~with~~ an ~~illegal~~invalid character, ~~with~~ a control character in other than the first character position, ~~with~~ a control character other than K28.5 or K28.3 in the first character position, or with ~~a~~one or more characters with a running disparity error).

3.1.184 valid dword: A dword that is not an invalid dword.

3.1.34 dword synchronization: Detection of an incoming stream of dwords from a physical link by a phy. See 6.8.

6.2 Encoding (8b10b)

6.2.1 Encoding overview

All data bytes transferred in SAS are encoded into 10-bit data characters using 8b10b coding. Additional characters not related to data bytes are called control characters.

All characters transferred in SAS are grouped into four-character sequences called dwords. [new para]

A primitive is a dword whose first character is a control character and remaining three characters are data characters.

...

A data dword is a dword ~~starting with a data character~~that contains four valid data characters.

...

6.3 Character encoding and decoding

6.3.3 Valid and invalid transmission characters

6.3.3.1 Definitions

Table 35 and table 36 define the valid data characters (Dxx.y characters) and valid control characters (Kxx.y characters), respectively, and shall be used for both generating valid transmission characters (encoding) and checking the validity of received transmission characters (decoding). Each data character and control character entry has two columns that represent two (not necessarily different) transmission characters, corresponding to the current value of the running disparity (current RD - or current RD +). Running disparity is a binary parameter with either the value negative (-) or the value positive (+). The running disparity at the beginning of a primitive is the beginning running disparity (beginning RD).

After powering on, the transmitter may initialize the current RD to positive or negative. Upon transmission of any transmission character, the transmitter shall calculate a new value for its running disparity based on the contents of the transmitted character.

After powering on or exiting diagnostic mode (the definition of diagnostic mode is beyond the scope of this standard), the receiver should assume either the positive or negative value for its initial running disparity. Upon reception of any transmission character, the receiver shall determine whether the transmission character is valid or invalid according to the following rules and shall calculate a new value for its running disparity based on the contents of the received character.

...

Table 35 defines the valid data characters (Dxx.y characters).

...

Table 36 defines the valid control characters (Kxx.y characters). Comma patterns, two bits of one polarity followed by five bits of the opposite polarity, are underlined.

...

6.3.3.3 Validity of received transmission characters

The columns in table 35 and table 36 corresponding to the current value of the receiver's running disparity shall be searched for each received transmission character. If the received transmission character is found in the proper column, then the transmission character shall be considered valid and the associated data byte or control byte determined (decoded). If the received transmission character is not found in the proper column, then the transmission character shall be considered invalid and the dword containing the character shall be considered an invalid dword. Independent of the transmission character's validity, the received transmission character shall be used to calculate a new value of running disparity.

This new value shall be used as the receiver's current running disparity for the next received transmission character.

Detection of a code violation does not necessarily indicate that the transmission character in which the code violation was detected is in error. Code violations may result from a prior error that altered the running disparity of the bit stream but did not result in a detectable error at the transmission character in which the error occurred. The example shown in table 37 exhibits this behavior.

...

6.8 SP_DWS (phy layer dword synchronization) state machine

6.8.1 SP_DWS state machine overview

Each phy includes an SP_DWS state machine.

This state machine establishes the same dword boundaries at the receiver as at the attached transmitter by searching for control characters. A receiver in the phy monitors and decodes the incoming data stream and forces K28.5 characters into the first character position to effectively perform dword alignment when requested by the SP_DWS state machine. The receiver continues to reestablish dword alignment by forcing received K28.5 characters into the first character position until a valid primitive is detected. The resultant primitives, dwords and valid dword indicators (e.g., encoding error indicators) are sent to this state machine to enable it to determine the dword synchronization policy.

After dword synchronization has been achieved, this state machine monitors invalid dwords that are received. When an invalid dword is detected, it requires two valid dwords to nullify its effect. When four invalid dwords are detected without nullification, dword synchronization is considered lost.

While dword synchronization is lost, the data stream received is invalid and dwords shall not be passed to the link layer.

...

6.8.2 SP_DWS receiver

The SP_DWS receiver receives the following messages from the SP_DWS state machine:

a) Find Dword.

The SP_DWS receiver sends the following messages to the SP_DWS state machine:

a) Dword Received (Valid Primitive);
b) Dword Received (Valid Data Dword); and
c) Dword Received (Invalid).

The SP_DWS receiver sends Dword Received confirmations to the link layer state machine receivers (e.g., SL_IR, SL, SSP, SMP, and XL).

Upon receiving a Find Dword message, the SP_DWS receiver shall monitor the input data stream and force each K28.5 character detected into the first character position as a possible dword. If the next three characters are data characters, it shall send the dword as a Dword Received (Valid Primitive) message to the SP_DWS state machine. Until it receives another Find Dword message, for every four characters it receives it shall:

a) send a Dword Received (Invalid) message to the SP_DWS state machine if the dword is an invalid dword (see 3.1.66);
b) send a Dword Received (Valid Primitive) message to the SP_DWS state machine if the dword is a primitive (i.e., the dword contains a K28.5 character in the first character position followed by three data characters); or
c) send a Dword Received (Valid Data Dword) message to the SP_DWS state machine if the dword is not an invalid dword or a primitive.

6.8.3 SP_DWS0:AcquireSync state

6.8.3.1 State description

This is the initial state of this state machine.

After receiving a Start DWS message, this state shall:

a) send a Find Dword message to the SP_DWS receiver; and
b) initialize and start the DWS Reset Timeout timer;

If this state is entered from SP_DWS1:Valid1 or SP_DWS2:Valid2, this state shall send a Find Dword message to the SP_DWS receiver. and the DWS Reset Timeout timer shall continue running.

If this state is entered from SP_DWS1:Valid1 or SP_DWS2:Valid2 and the DWS Reset Timeout timer has expired, this state may send a DWS Reset message to the SP state machine (e.g., if the phy chooses to initiate a new link reset sequence because dword synchronization has been lost for too long).

This state shall not send a DWS Reset message to the SP until the DWS Reset Timeout timer expires. If the DWS Reset Timeout timer expires, this state may send a DWS Reset message to the SP state machine.

6.8.3.2 Transition SP_DWS0:AcquireSync to SP_DWS1:Valid1

This transition shall occur after sending a Find Dword message and receiving a Dword Received (Valid Primitive) message.

6.8.4 SP_DWS1:Valid1 state

6.8.4.1 State description

This state is reached after one valid primitive has been received. This state waits for a second valid primitive or an invalid dword.

The DWS Reset Timeout timer shall continue running.

6.8.4.2 Transition SP_DWS1:Valid1 to SP_DWS0:AcquireSync

This transition shall occur after receiving a Dword Received (Invalid) message or after the DWS Reset Timeout timer expires.

6.8.4.3 Transition SP_DWS1:Valid1 to SP_DWS2:Valid2

This transition shall occur after receiving a Dword Received (Valid Primitive) message.

6.8.5 SP_DWS2:Valid2 state

6.8.5.1 State description

This state is reached after two valid primitives have been received without adjusting the dword synchronization. This state waits for a third valid primitive or an invalid dword.

The DWS Reset Timeout timer shall continue running.

6.8.5.2 Transition SP_DWS2:Valid2 to SP_DWS0:AcquireSync

This transition shall occur after receiving a Dword Received (Invalid) message or after the DWS Reset Timeout timer expires.

6.8.5.3 Transition SP_DWS2:Valid2 to SP_DWS3:SyncAcquired

This transition shall occur after receiving a Dword Received (Valid Primitive) message.

6.8.6 SP_DWS3:SyncAcquired state

6.8.6.1 State description

This state is reached after three valid primitives have been received without adjusting the dword synchronization.

The most recently received primitive and all subsequent dwords shall be forwarded for processing by the link layer.

This state waits for a Dword Received (Invalid) message, which indicates that dword synchronization might be lost.

6.8.6.2 Transition SP_DWS3:SyncAcquired to SP_DWS4:Lost1

This transition shall occur after receiving a Dword Received (Invalid) message.

6.8.7 SP_DWS4:Lost1 state

6.8.7.1 State description

This state is reached when one invalid dword has been received and not nullified. This state waits for a Dword Received message.

6.8.7.2 Transition SP_DWS4:Lost1 to SP_DWS5:Lost1Recovered

This transition shall occur after receiving a Dword Received (Valid Data Dword) message or a Dword Received (Valid Primitive) message.

6.8.7.3 Transition SP_DWS4:Lost1 to SP_DWS6:Lost2

This transition shall occur after receiving a Dword Received (Invalid) message.

6.8.8 SP_DWS5:Lost1Recovered state

6.8.8.1 State description

This state is reached when a valid dword has been received after one invalid dword had been received. This state waits for a Dword Received message.

6.8.8.2 Transition SP_DWS5:Lost1Recovered to SP_DWS3:SyncAcquired

This transition shall occur after receiving a Dword Received (Valid Data Dword) message or a Dword Received (Valid Primitive) message.

6.8.8.3 Transition SP_DWS5:Lost1Recovered to SP_DWS6:Lost2

This transition shall occur after receiving a Dword Received (Invalid) message.

6.8.9 SP_DWS6:Lost2 state

6.8.9.1 State description

This state is reached when two invalid dwords have been received and not nullified. This state waits for a Dword Received message.

6.8.9.2 Transition SP_DWS6:Lost2 to SP_DWS7:Lost2Recovered

This transition shall occur after receiving a Dword Received (Valid Data Dword) message or a Dword Received (Valid Primitive) message.

6.8.9.3 Transition SP_DWS6:Lost2 to SP_DWS8:Lost3

This transition shall occur after receiving a Dword Received (Invalid) message.

6.8.10 SP_DWS7:Lost2Recovered state

6.8.10.1 State description

This state is reached when a valid dword has been received after two invalid dwords had been received. This state waits for a Dword Received message.

6.8.10.2 Transition SP_DWS7:Lost2Recovered to SP_DWS4:Lost1

This transition shall occur after receiving a Dword Received (Valid Data Dword) message or a Dword Received (Valid Primitive) message.

6.8.10.3 Transition SP_DWS7:Lost2Recovered to SP_DWS8:Lost3

This transition shall occur after receiving a Dword Received (Invalid) message.

6.8.11 SP_DWS8:Lost3 state

6.8.11.1 State description

This state is reached when three invalid dwords have been received and not nullified. This state waits for aDword Received message.

If a Dword Received (Invalid) message is received (i.e., the fourth non-nullified invalid dword is received), this state shall send a DWS Lost message to the SP state machine.

6.8.11.2 Transition SP_DWS8:Lost3 to SP_DWS9:Lost3Recovered

This transition shall occur after receiving a Dword Received (Valid Data Dword) message or a Dword Received (Valid Primitive) message.

6.8.11.3 Transition SP_DWS8:Lost3 to SP_DWS0:AcquireSync

This transition shall occur after sending a DWS Lost message.

6.8.12 SP_DWS9:Lost3Recovered state

6.8.12.1 State description

This state is reached when a valid dword has been received after three invalid dwords had been received.

This state waits for a Dword Received message.

If a Dword Received (Invalid) message is received (i.e., the fourth non-nullified invalid dword is received), this state shall send a DWS Lost message to the SP state machine.

6.8.12.2 Transition SP_DWS9:Lost3Recovered to SP_DWS6:Lost2

This transition shall occur after receiving a Dword Received (Valid Data Dword) message or a Dword Received (Valid Primitive) message.

6.8.12.3 Transition SP_DWS9:Lost3Recovered to SP_DWS0:AcquireSync

This transition shall occur after sending a DWS Lost message.

7.2.3 Primitive encodings

...

Table 55 — Primitive encoding for primitives used only inside STP connections and on SATA physical links

...

b SATA_ERROR does not appear inside STP connections. It is an invalid dword, used by expander devices forwarding an error onto a SATA physical link (see 7.2.7.1).

7.2.5.7 ERROR

ERROR ~~is~~ may be sent by an expander device when it is forwarding dwords from a SAS physical link or SATA physical link to a SAS physical link and it receives an invalid dword or an ERROR.

See 7.15 for details on error handling by expander devices.

~~SAS phys may ignore ERROR or treat it as an invalid dword.~~

7.2.5.8 HARD_RESET

HARD_RESET is used to force a phy to generate a hard reset to its port. This primitive is only valid after the phy reset sequence without an intervening identification sequence (see 4.4) and shall be ignored at other times.

7.2.6.5 NAK (Negative acknowledgement)

NAK indicates the negative acknowledgement of an SSP frame and the reason for doing so.

The versions of NAK representing different reasons are defined in table 65.

Table 1 — NAK primitives

Primitive	Description
NAK (CRC ERROR)	The frame had a bad CRC, an invalid dword, or an ERROR primitive.
NAK (RESERVED 0)	Reserved. Processed the same as NAK (CRC ERROR).
NAK (RESERVED 1)	Reserved. Processed the same as NAK (CRC ERROR).
NAK (RESERVED 2)	Reserved. Processed the same as NAK (CRC ERROR).

7.2.7.1 SATA_ERROR

SATA_ERROR ~~is~~ may be sent by an expander device when it is forwarding dwords from a SAS physical link to a SATA physical link and it receives an invalid dword or an ERROR. SATA_ERROR is an invalid dword.

See 6.8 for details on error handling by expander devices.

7.9.5.4 SL_IR_RIF (receive IDENTIFY address frame) state machine

7.9.5.2 SL_IR transmitter and receiver

The SL_IR transmitter receives the following messages from the SL_IR state machines indicating primitive sequences, frames, and dwords to transmit:

a) Transmit IDENTIFY Address Frame;
b) Transmit HARD_RESET; and

c) Transmit Idle Dword.

The SL_IR transmitter sends the following messages to the SL_IR state machines:

a) HARD_RESET Transmitted; and
b) IDENTIFY Address Frame Transmitted.

The SL_IR receiver sends the following messages to the SL_IR state machines indicating primitive sequences and dwords received from the SP_DWS receiver (see 6.8.2):

a) SOAF Received;
b) EOAF Received;
c) Data Dword Received;
d) ERROR Received;
e) Invalid Dword Received; and
f) HARD_RESET Received.

The SL_IR receiver shall ignore all other dwords.

7.9.5.4 SL_IR_RIF (receive IDENTIFY address frame) state machine

7.9.5.4.1 SL_IR_RIF state machine overview

The SL_IR_RIF state machine receives an IDENTIFY address frame and checks the IDENTIFY address frame to determine if the frame should be accepted or discarded by the link layer.

This state machine consists of the following states:

a) SL_IR_RIF1:Idle (see 7.9.5.4.2)(initial state);
b) SL_IR_RIF2:Receive_Identify_Frame (see 7.9.5.4.3); and
c) SL_IR_RIF3:Completed (see 7.9.5.4.4).

This state machine shall start in the SL_IR_RIF1:Idle state. This state machine shall transition to the SL_IR_RIF1:Idle state from any other state after receiving a Phy Layer Not Ready confirmation.

7.9.5.4.2 SL_IR_RIF1:Idle state

7.9.5.4.2.1 State description

This state waits for an SOAF to be received from the physical link, indicating an address frame is arriving.

7.9.5.4.2.2 Transition SL_IR_RIF1:Idle to SL_IR_RIF2:Receive_Identify_Frame

This transition shall occur after both:

a) a Start SL_IR Receiver confirmation is received; and
b) an SOAF Received message is received.

7.9.5.4.3 SL_IR_RIF2:Receive_Identify_Frame state

7.9.5.4.3.1 State description

This state receives the dwords of an address frame and the EOAF.

If this state receives an SOAF Received message, then this state shall discard the address frame (i.e., the subsequent Data Dword Received and EOAF Received messages) and send an Address Frame Failed confirmation to the management application layer to indicate that an invalid IDENTIFY address frame was received.

If this state receives more than eight Data Dword Received messages after an SOAF Received message and before an EOAF Received message, then this state shall discard the address frame and send an Address Frame Failed confirmation to the management application layer to indicate that an invalid IDENTIFY address frame was received.

If this state receives an Invalid Dword Received message or an ERROR Received message after an SOAF Received message and before an EOAF Received message, then this state may ignore the invalid dword or ERROR or may discard the address frame and send an Address Frame Failed confirmation to the management application layer to indicate that an invalid IDENTIFY address frame was received.

After receiving an EOAF Received message, this state shall check if it the IDENTIFY address frame is valid.

This state shall accept an IDENTIFY address frame and send an Identify Received message to the SL_IR_IRC state machine if:

a) the ADDRESS FRAME TYPE field is set to Identify;
b) the number of bytes between the SOAF and EOAF is 32; and
c) the CRC field contains a valid CRC.

Otherwise, this state shall discard the IDENTIFY address frame and send an Address Frame Failed confirmation to the management application layer to indicate that an invalid IDENTIFY address frame was received.

7.9.5.4.3.2 Transition SL_IR_RIF2:Receive_Identify_Frame to SL_IR_RIF3:Completed

This transition shall occur after sending an Identify Received message or Address Frame Failed confirmation.

7.9.5.4.4 SL_IR_RIF3:Completed state

This state waits for a Phy Layer Not Ready confirmation.

7.14 SL (link layer for SAS phys) state machines

7.14.2 SL transmitter and receiver

The SL transmitter receives the following messages from the SL state machines specifying primitive sequences, frames, and dwords to transmit:

a) Transmit Idle Dword;
b) Transmit SOAF/Data Dwords/EOAF;
c) Transmit OPEN_ACCEPT;
d) Transmit OPEN_REJECT with an argument indicating the specific type (e.g., Transmit OPEN_REJECT (Retry));
e) Transmit BREAK;
f) Transmit BROADCAST; and
g) Transmit CLOSE with an argument indicating the specific type (e.g., Transmit CLOSE (Normal)).

When there is no outstanding message specifying a dword to transmit, the SL transmitter shall transmit idle dwords.

The SL transmitter sends the following messages ~~from~~ to the SL state machines based on dwords that have been transmitted:

a) SOAF/Data Dwords/EOAF Transmitted.

The SL receiver sends the following messages to the SL state machines indicating primitive sequences and dwords received from the SP_DWS receiver (see 6.8.2):

a) SOAF Received;
b) EOAF Received;
c) BROADCAST Received with an argument indicating the specific type (e.g., BROADCAST Received (Change));
d) BREAK Received;
e) OPEN_ACCEPT Received;
f) OPEN_REJECT Received with an argument indicating the specific type (e.g., OPEN_REJECT Received (No Destination));
g) AIP Received;
h) CLOSE Received with an argument indicating the specific type (e.g., CLOSE Received (Normal));
i) ERROR Received;
j) Data Dword Received with an argument indicating the valid data dword received; and
k) Invalid Dword Received;

The SL receiver shall ignore all other dwords.

...

7.14.3 SL_RA (receive OPEN address frame) state machine

The SL_RA state machine's function is to receive address frames and determine if the received address frame is an OPEN address frame and whether or not it was received successfully. This state machine consists of one state.

This state machine receives SOAFs, dwords of an OPEN address frames, and EOAFs.

This state machine shall ignore all messages except SOAF Received, Data Dword Received, and EOAF Received.

If this state machine receives a subsequent SOAF Received message after receiving an SOAF Received message but before receiving an EOAF Received message, then this state machine shall discard the Data Dword Received messages received before the subsequent SOAF Received message.

If this state machine receives more than eight Data Dword Received messages after an SOAF Received message and before an EOAF Received message, then this state machine shall discard the address frame. If this state machine receives an Invalid Dword Received message or an ERROR Received message after an SOAF Received message and before an EOAF Received message, then this state machine may either ignore the invalid dword or ERROR or may discard the address frame. After receiving an EOAF Received message, this state machine shall check if the address frame is a valid OPEN address frame.

This state machine shall accept an address frame ~~a valid OPEN address frame~~ if:

a) the ADDRESS FRAME TYPE field is set to Open;
b) the number of data dwords between the SOAF and EOAF is 8; and
c) the CRC field contains a valid CRC.

Otherwise, this state machine shall discard the address frame. If the frame is not discarded then this state machine shall send a OPEN Address Frame Received message to the SL_CC0:Idle state and the SL_CC1:ArbSel state with an argument that contains all the data dwords received in the OPEN address frame.

7.14.4 SL_CC (connection control) state machine

7.14.4.1 SL_CC state machine overview

...

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

7.15 XL (link layer for expander phys) state machine

7.15.1 XL state machine overview

~~...~~

~~Unless otherwise stated within a state description, all invalid dwords and unexpected primitives received within any XL state shall be ignored.~~

7.15.2 XL transmitter and receiver

The XL transmitter receives the following messages from the XL state machine ~~indicating~~ specifying primitive sequences, frames, and dwords to transmit:

a) Transmit Idle Dword;
b) Transmit AIP with an argument indicating the specific type (e.g., Transmit AIP (Normal));
c) Transmit BREAK;
d) Transmit BROADCAST with an argument indicating the specific type (e.g., Transmit BROADCAST (Change));
e) Transmit CLOSE with an argument indicating the specific type (e.g., Transmit CLOSE (Normal));
f) Transmit OPEN_ACCEPT;
g) Transmit OPEN_REJECT, with an argument indicating the specific type (e.g., Transmit OPEN_REJECT (No Destination));

h) Transmit OPEN Address Frame; and
i) Transmit Dword.

The XL transmitter sends the following messages to the XL state machine based on dwords that have been transmitted:

a) a) OPEN Address Frame Transmitted.

...

When there is no outstanding message specifying a dword to transmit, the XL transmitter shall transmit idle dwords.

The XL receiver sends the following messages to the XL state machine indicating primitive sequences, frames, and dwords received from the SP_DWS receiver (see 6.8.2):

a) AIP Received with an argument indicating the specific type (e.g., AIP Received (Normal));
b) BREAK Received;
c) BROADCAST Received;
d) CLOSE Received;
e) OPEN_ACCEPT Received;
f) OPEN_REJECT Received;
g) OPEN Address Frame Received;
h) Dword Received with an argument indicating the valid data dword or valid primitive received; and
i) Invalid Dword Received.

The XL receiver shall ignore all other dwords.

While receiving an address frame, if the XL receiver receives an invalid dword or ERROR primitive after an SOAF and before an EOAF, then the XL receiver may either ignore the invalid dword or ERROR primitive or may discard the address frame.

7.15.10 XL7:Connected state

7.15.10.1 State description

...

If:

a) an ~~invalid dword is received with the~~ Invalid Dword Received message is received; and
b) the expander phy is forwarding to an expander phy attached to a SAS physical link,

the expander phy shall:

a) send an ERROR primitive with the Transmit Dword request instead of the invalid dword; or
b) delete the invalid dword.

If:

a) an ~~invalid dword or an~~ ERROR primitive is received with the Dword Received message or an Invalid Dword Received message is received; and
b) the expander phy is forwarding to an expander phy attached to a SATA physical link,

the expander phy shall:

a) send a SATA_ERROR primitive with the Transmit Dword request instead of the invalid dword or ERROR primitive; or
b) delete the ERROR primitive or invalid dword.

7.15.11 XL8:Close_Wait state

7.15.11.1 State description

If:

a) an ~~invalid dword is received with the~~ Invalid Dword Received message is received;
b) and the expander phy is forwarding to an expander phy attached to a SAS physical link,

the expander phy shall:

a) send an ERROR primitive with the Transmit Dword request instead of the invalid dword; or
b) delete the invalid dword.

If:

a) an ~~invalid dword or an~~ ERROR primitive is received with the Dword Received message or an Invalid Dword Received message is received; and
b) the expander phy is forwarding to an expander phy attached to a SATA physical link,

the expander phy shall:

a) send a SATA_ERROR primitive with the Transmit Dword request instead of the invalid dword or ERROR primitive; or
b) delete the invalid dword or ERROR primitive.

...

7.16.3 SSP frame transmission and reception

...

Receiving SSP phys shall acknowledge SSP frames within 1 ms if not discarded as described in 7.16.7.7 with either a positive acknowledgement (ACK) or a negative acknowledgement (NAK). ACK means the SSP frame was received into a frame buffer without errors. NAK (CRC ERROR) means the SSP frame was received with a CRC error, an invalid dword, or an ERROR primitive.

NOTE 1 It is not required that frame recipients generate NAK (CRC ERROR) from invalid dwords and ERRORs (see 7.16.7.2).

...

7.16.7 SSP (link layer for SSP phys) state machines

7.16.7.1 SSP state machines overview

In Figure 99, add Invalid Dword message into SSP_RF.

7.16.7.2 SSP transmitter and receiver

The SSP transmitter receives the following messages from the SSP state machines ~~indicating~~ specifying primitive sequences and frames to transmit:

a) Transmit RRDY with an argument indicating the specific type (e.g., Transmit RRDY (Normal));
b) Transmit CREDIT_BLOCKED;
c) Transmit ACK;
d) Transmit NAK with an argument indicating the specific type (e.g., Transmit NAK (CRC Error));
e) Transmit Frame (i.e., SOF/data dwords/EOF); and
f) Transmit DONE with an argument indicating the specific type (e.g., Transmit DONE (Normal)).

The SSP transmitter sends the following messages to the SSP state machines based on dwords that have been transmitted:

a) DONE Transmitted;
b) RRDY Transmitted;
c) CREDIT_BLOCKED Transmitted;
d) ACK Transmitted;
e) NAK Transmitted; and
f) Frame Transmitted.

~~When the SSP transmitter is not processing a message to transmit, it shall transmit idle dwords.~~

When there is no outstanding message specifying a dword to transmit, the SSP transmitter shall transmit idle dwords.

The SSP receiver sends the following messages to the SSP state machines indicating primitive sequences and dwords received from the SP_DWS receiver (see 6.8.2):

a) ACK Received;
b) NAK Received;
c) RRDY Received;
d) CREDIT_BLOCKED Received;
e) DONE Received with an argument indicating the specific type (e.g., DONE Received (Normal));
f) SOF Received;
g) EOF Received;
h) ERROR Received;
i) Data Dword Received with an argument indicating the valid data dword received; and
j) Invalid Dword Received.
~~k) EOF Received.~~

The SSP receiver shall ignore all other dwords.

7.16.7.7 SSP_RF (receive frame control) state machine

...

The SSP_RF state machine's function is to receive frames and determine whether or not those frames were received successfully. This state machine consists of one state.

This state machine:

a) checks the frame to determine if the frame should be accepted or discarded;
b) checks the frame to determine if an ACK or NAK should be transmitted; and
c) sends a Frame Received confirmation to the port layer.

~~If consecutive SOF Received messages are received without an intervening EOF Received message (i.e., SOF, data dwords, SOF, data dwords, and EOF instead of SOF, data dwords, EOF, SOF, data dwords, and EOF) then this state machine shall discard all dwords between those SOFs.~~

If this state receives a subsequent SOF Received message after receiving an SOF Received message but before receiving an EOF Received message, then this state shall discard the Data Dword Received messages received before the subsequent SOF Received message.

~~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 data dwords between the SOF and EOF is less than 7;~~
~~b) the number of data dwords after the SOF is greater than 263 data dwords;~~
~~c) the Rx Credit Status (Credit Exhausted) message is received; or~~
~~d) the DONE Received message is received.~~

If:

a) this state receives more than 263 Data Dword Received messages after an SOF Received message and before an EOF Received message;
b) this state receives less than 7 Data Dword Received messages after an SOF Received message and before an EOF Received message,
c) this state receives an Rx Credit Status (Credit Exhausted) message; or
d) this state receives a DONE Received message,

then this state shall discard the frame.

If this state receives an Invalid Dword Received message or an ERROR Received message after receiving an SOF Received message and before receiving an EOF Received message, then this state machine may either:

a) ignore the invalid dword or ERROR; or
b) discard the frame, send a Frame Received message to the SSP_RCM state machine, send a Frame Received message to the SSP_RIM state machine, and send a Frame Received (Unsuccessful) message to the SSP_TAN1:Idle state.

~~If the frame is discarded then no further action is taken by this state machine. If the frame is not discarded then~~ If the frame CRC is bad, this state machine shall:

a) send a Frame Received message to the SSP_RCM state machine~~; and~~
b) send a Frame Received message to the SSP_RIM state machine; and
c) send a Frame Received (Unsuccessful) message to the SSP_TAN1:Idle state.

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

a) send a Frame Received message to the SSP_RCM state machine;
b) send a Frame Received message to the SSP_RIM state machine;
c) send a Frame Received (Successful) message to the SSP_TAN1:Idle state; ~~and:~~
a) if the last Rx Balance Status message received had an argument of Balanced, send a Frame Received (ACK/NAK Balanced) confirmation to the port layer; ~~or~~and
b) if the last Rx Balance Status message received had an argument of Not Balanced, send a Frame Received (ACK/NAK Not Balanced) confirmation to the port layer.

~~If the frame CRC is bad or the frame contained invalid data dwords, this state machine shall send a Frame Received (Unsuccessful) message to the SSP_TAN1:Idle state.~~

7.16.7.11 SSP_TAN (transmit ACK/NAK control) state machine

...

Any time this state machine receives a Frame Received (Unsuccessful) message it shall send a Transmit NAK (CRC Error) message to the SSP transmitter.

7.18 SMP link layer

7.18.1 SMP frame transmission and reception

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 104. ~~There is no acknowledgement of SMP frames with ACK and NAK. There is no credit exchange with RRDY.~~

NOTE 2 Unlike SSP, there is no acknowledgement of SMP frames with ACK and NAK.; see 7.18.4 for error handling details. There is no credit exchange with RRDY.

...

The last data dword after the SOF prior to the EOF always contains a CRC (see 7.5). The SMP link layer state machine checks that the frame is not too short and that the CRC is valid (see 7.18.4).

7.18.4.2 SMP transmitter and receiver

The SMP transmitter receives the following messages from the SMP state machines ~~indicating~~ specifying dwords and frames to transmit:

a) Transmit Idle Dword; and
b) Transmit Frame.

The SMP transmitter sends the following messages to the SMP state machines:

a) Frame Transmitted.

When there is no outstanding message specifying a dword to transmit, the SMP transmitter shall transmit idle dwords.

The SMP receiver sends the following messages to the SMP state machines indicating primitive sequences and dwords received from the SP_DWS receiver (see 6.8.2):

a) SOF Received;
b) EOF Received;
c) Data Dword Received with an argument indicating the valid data dword or valid primitive received; and

d) Invalid Dword Received.

The SMP receiver shall ignore all other dwords.

7.18.4.3.4 SMP_IP3:Receive_Frame state

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

If this state receives a subsequent SOF Received message after receiving an SOF Received message but before receiving an EOF Received message, then this state shall discard the Data Dword Received messages received before the subsequent SOF Received message.

If:

a) this state receives more than 258 Data Dword Received messages after an SOF Received message and before an EOF Received message; or
b) this state receives less than 2 Data Dword Received messages after an SOF Received message and before an EOF Received message,

then this state shall discard the frame and send a Frame Received (SMP Failure) confirmation to the port layer.

If this state receives an Invalid Dword Received message or an ERROR Received message after an SOF Received message and before an EOF Received message, then this state machine may either:

a) ignore the invalid dword or ERROR; or
b) discard the frame and send a Frame Received (SMP Failure) confirmation to the port layer.

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

~~If the number of dwords between the SOF and EOF of the SMP response frame is less than 2, or the number of dwords after an SOF is greater than 258, 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, this state shall:

a) send a Frame Received confirmation to the port layer; and
b) send a Request Close message to the SL state machines (see 7.14).

If an SMP Transmit Break request is received, this state shall send a Request Break message to the SL state machines and terminate.

This state shall request idle dwords be transmitted by repeatedly sending Transmit Idle Dword messages to the SMP transmitter.

7.18.4.4.2 SMP_TP1:Receive_Frame state

7.18.4.4.2.1 State description

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

If this state receives an Invalid Dword Received message or an ERROR Received message after receiving an SOF Received message and before receiving an EOF Received message, then this state shall discard the Data Dword Received messages received before the subsequent SOF Received message.

If:

a) this state receives more than 258 Data Dword Received messages after an SOF Received message and before an EOF Received message; or
b) this state receives less than 2 Data Dword Received messages after an SOF Received message and before an EOF Received message,

then this state shall discard the frame, send a Request Break message to the SL state machines (see 7.14) and shall terminate the state machine.

If this state receives an Invalid Dword Received message or an ERROR Received message after an SOF Received message and before an EOF Received message, then this state machine may either:

a) ignore the invalid dword or ERROR; or
b) discard the frame, send a Request Break message to the SL state machines (see 7.14) and shall terminate the state machine.

If the SMP request frame is received with a CRC error, this state shall discard the frame, send a Request Break message to the SL state machines (see 7.14) and shall terminate the state machine.

~~If an SMP frame is received, this state shall send a Request Break message to the SL state machines (see 7.14) and terminate if:~~

~~a) the SMP frame has a CRC error;~~
~~b) the number of data dwords between the SOF and EOF is less than 2; or~~
~~c) the number of data dwords after the SOF is greater than 258.~~

Otherwise, this state shall send a Frame Received confirmation to the port layer.

This state shall request idle dwords be transmitted by repeatedly sending Transmit Idle Dword messages to the SMP transmitter.