

**From:** Gerry Houlder, Seagate Technology <gerry.houlder@seagate.com>  
**Subj:** SAS: Add low power transceiver options  
**Date:** Apr. 25, 2008

---

## Overview

This proposal adds lower power transceiver modes to SAS. The intent is to add low power modes that are similar to or compatible with the low power modes available in SATA. This should allow an expander design that can work with either SATA or SAS devices in low power mode.

This proposal focuses on target to initiator and target to expander links. These links should provide the greatest opportunity for power savings because they are likely to have more idle time and will not have performance impact on access to other targets. For these cases, only the target would need to initiate low power modes and could reasonably predict idle times based on commands in its queue and timers that allow for idle and standby modes.

For compatibility with SATA, there needs to be 4 primitives. If SAS chooses the same primitive encoding that SATA has, expander design might be easier. The primitives are:

- (a) PMREQ\_P to request partial power mode (transceiver must recover within 10 us);
- (b) PMREQ\_S to request slumber power mode (transceiver must recover within 10 msec);
- (c) PMACK to accept change into the requested power mode;
- (d) PMNAK to reject the power mode request.

When a power mode request is accepted, both ends of the link stop transmitting. Either end may revive the link by sending COMWAKE. Both ends are required to remember whether the link is SSP or STP protocol and previously negotiated settings (speed, SSC, multiplexing, etc.). The intent is that both ends can restart and obtain synchronization without needing to change receiver settings or redo training. Repeating training may increase the recovery time so much that partial power mode may not be feasible.

When an initiator requests an expander to open a link that is in partial power mode, recovery time should be fast enough so any delay can be covered by returning AIP(WAITING ON DEVICE) one time until the link is ready. When an initiator requests to open a link that is in slumber power mode, the expander should be allowed to use OPEN\_REJECT(RETRY) (the recovery time will be several milliseconds and we don't want to tie up expander pathways for that long by using AIP). I think it is important to NOT pick new primitives for this to maintain compatibility with older initiators that wouldn't recognize the new primitive.

The partial mode has fast enough recovery so that it may be used by a target-expander link without explicit approval by an initiator. The slumber mode causes enough delay so that an initiator should explicitly allow/ disallow this option based on its performance requirements. There are several possibilities for this:

- (a) Use SNW-3 to negotiate capability/ permission similar to speed negotiation. The expander would base its setting on something from an initiator. Could it be SNW-3 setting from each initiator? Could it be a setting in an SMP function?
- (b) Use something in IDENTIFY frame?
- (c) Use a SAS specific mode page. The initiator grants permission directly to a target without needing expander participation, but using the application layer to control link layer behavior has been frowned upon in the past.
- (d) Only invoke slumber when idle, standby, or sleep is either commanded by initiator or initiated by timer expiration in the target. Recovery from slumber could also command the drive to change to active state (from idle or standby).

There may be a desire to add information to SMP functions to indicate links that are in a low power mode, especially slumber mode. The DISCOVER function and the DISCOVER LIST function seem like obvious candidates to me. Are there others?

If a drive is removed while in partial or slumber and another drive is inserted, the drive will send a COMRESET at power up and alert the initiator/expander that something has changed since it did not receive a COMWAKE. Similarly, if a drive is pulled and not replaced, it would become a loss of signal situation, so I think that both of these are covered.

Questions have also been raised about whether slumber mode actually saves a significant amount of power compared to partial mode. If the additional savings are not significant enough then perhaps this mode is not worth defining? Of course, if it is used in conjunction with idle or other longer recovery low power modes then perhaps it is still useful. This will be a discussion point. Anyone willing to provide comparative data on this is invited to do so.

## **SAS-2++ Changes (based on SAS-2 rev.14):**

### **6.6 Out of band (OOB) signals**

*[clauses 6.6.1 through 6.6.4 are unchanged]*

#### **6.6.5 Idle bus condition**

During power management states (Partial and Slumber), the transmitter device and receiver device are allowed to float. The maximum transmitter and receiver device transients specified in Table 54 and Table 56, respectively apply.

### **6.7 Phy reset sequences**

*[clauses 6.7.1 through 6.7.5 are unchanged]*

#### **6.7.6 SAS transceiver low power sequences**

##### **6.7.6.1 Transition from active to power management state**

[Need a new figure showing accepted SAS\_PMREQx to PMACK to not transmitting sequence]

##### **6.7.6.2 COMWAKE sequence to recover from power management state**

The sequence to recover from either partial or slumber power management state is shown in figure new1.

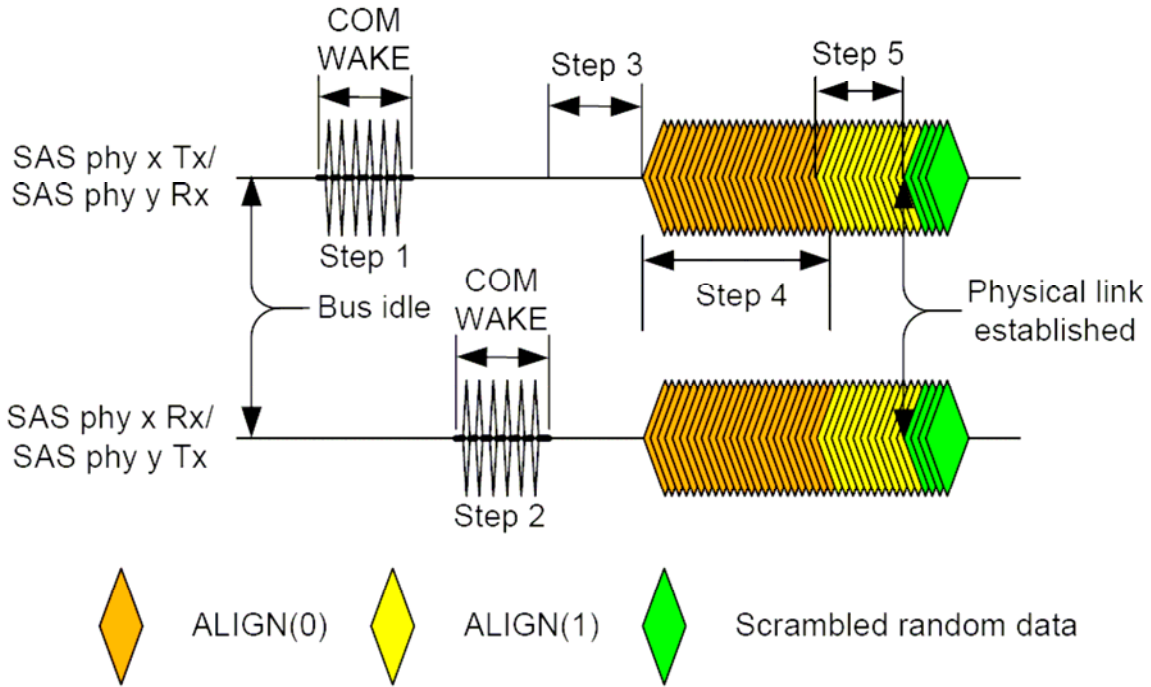


Figure new1 – SAS COMWAKE recovery sequence  
 [editors note: Original Visio file for this figure available from Mark Evans.]

The COMWAKE recovery sequence is as follows:

- 1) port x decides to wake up the link and initiates power management action to transition phy x to active state;
- 2) phy x transmits COMWAKE;
- 3) phy y detects COMWAKE, initiates power management action to transition phy y to active state, and transmits COMWAKE back to phy x;
- 4) both phys transmit ALIGN(0) primitives at previously negotiated settings;
- 5) when both receivers synchronize on ALIGN(0)s, the transmitters change to sending ALIGN(1)s; and
- 6) when both receivers synchronize on ALIGN(1)s, the transmitters change to idle dwords.

The link is now re-established with the same transfer rate and SNW3 settings (e.g., SSC mode and multiplexing mode) that were negotiated before the link was placed in power management mode.

### 6.7.6.3 Recovery from hot plug events during power management

[Need a new figure, similar to figure 150, showing detection and recovery from hot plug while in low power mode. Existing figures 141 or 142 may already be sufficient for this purpose.]

## 6.8 SP (phy layer) state machine

### 6.8.1 SP state machine overview

The SP state machine controls the phy reset sequence. This state machine consists of **three** **these** sets of states:

- a) OOB sequence (OOB) states;
- b) **SAS power management states**;
- c) SAS speed negotiation (SAS) states; and
- d) SATA host emulation (SATA) states.

This state machine consists of the following states:

- a) SP0:OOB\_COMINIT (see 6.8.3.2)(initial state);
- b) SP1:OOB\_AwaitCOMX (see 6.8.3.3);
- c) SP2:OOB\_NoCOMSASTimeout (see 6.8.3.4);
- d) SP3:OOB\_AwaitCOMINIT\_Sent (see 6.8.3.5);
- e) SP4:OOB\_COMSAS (see 6.8.3.6);
- f) SP5:OOB\_AwaitCOMSAS\_Sent (see 6.8.3.7);
- g) SP6:OOB\_AwaitNoCOMSAS (see 6.8.3.8);
- h) SP7:OOB\_AwaitCOMSAS (see 6.8.3.9);
- i) SP8:SAS\_Start (see 6.8.4.2);
- j) SP9:SAS\_WindowNotSupported (see 6.8.4.3);
- k) SP10:SAS\_AwaitALIGN (see 6.8.4.4);
- l) SP11:SAS\_AwaitALIGN1 (see 6.8.4.5);
- m) SP12:SAS\_AwaitSNW (see 6.8.4.6);
- n) SP13:SAS\_Pass (see 6.8.4.7);
- o) SP14 SAS\_Fail (see 6.8.4.8);
- p) SP15:SAS\_PHY\_Ready (see 6.8.4.9);
- q) SP16:SATA\_COMWAKE (see 6.8.5.2);
- r) SP17:SATA\_AwaitCOMWAKE (see 6.8.5.3);
- s) SP18:SATA\_AwaitNoCOMWAKE (see 6.8.5.4);
- t) SP19:SATA\_AwaitALIGN (see 6.8.5.5);
- u) SP20:SATA\_AdjustSpeed (see 6.8.5.6);
- v) SP21:SATA\_Transmit\_ALIGN (see 6.8.5.7);
- w) SP22:SATA\_PHY\_Ready (see 6.8.5.8);
- x) SP23:SATA\_PM\_Partial (see 6.8.5.9);
- y) SP24:SATA\_PM\_Slumber (see 6.8.5.10);
- z) SP25:SATA\_PortSel (see 6.8.6.2);
- aa) SP26:SATA\_SpinupHold (see 6.8.7.2);
- bb) SP27:SAS\_Settings (see 6.8.4.10);
- cc) SP28:SAS\_TrainSetup (see 6.8.4.11);
- dd) SP29:SAS\_Train (see 6.8.4.12); ~~and~~
- ee) SP30:SAS\_TrainingDone (see 6.8.4.13);
- ff) SP31:SAS\_PM\_Partial (see 6.8.x.x);
- gg) SP32:SAS\_PM\_Slumber (see 6.8.x.x).
- hh) SP33:SAS\_COMWAKE (see 6.8.x.x);
- ii) SP34:SAS\_AwaitCOMWAKE\_Sent (see 6.8.x.x);
- jj) SP35:SAS\_AwaitNoCOMWAKE (see 6.8.x.x);
- kk) SP36:SAS\_AwaitCOMWAKE (see 6.8.x.x);
- ll) SP37:SAS\_PM\_AwaitALIGN (see 6.8.x.x);
- m) SP38:SAS\_ReceivedALIGN0 (see 6.8.x.x); and
- n) SP39:SAS\_TransmitALIGN1 (see 6.8.x.x).

The SP state machine shall start in the SP0:OOB\_COMINIT state after:

- a) a power on;
- b) a hard reset;

- c) receiving a Management Reset request from the management layer (e.g., from the SMP PHY CONTROL function requesting a phy operation of LINK RESET or HARD RESET in an expander device); or
- d) receiving a Disable Phy request from the management layer (e.g., from the SMP PHY CONTROL function requesting a phy operation of DISABLE in an expander device).

The SP state machine shall maintain a MgmtReset state machine variable to determine whether a Management Reset request has been received. Any SP state that receives a Management Reset request shall set the MgmtReset state machine variable to one before making a transition to the SP0:OOB\_COMINIT state (see 6.8.3.2). Any SP state that receives a power on, or a hard reset shall set the MgmtReset state machine variable to zero before making a transition to the SP0:OOB\_COMINIT state.

The SP state machine shall maintain a Current SNW state machine variable to determine the current SNW (e.g., SNW-1, SNW-2, SNW-3, Final-SNW, Train-SNW, or Unsupported Phy Attached).

If the phy status is available through any of the following:

- a) the SMP DISCOVER response (see 10.4.3.10);
- b) the SMP DISCOVER LIST response (see 10.4.3.15);
- c) the Phy Control And Discover mode page (see 10.2.7.5); or
- d) the Protocol-Specific Port log page (see 10.2.8.1),

then the SP state machine shall maintain a ResetStatus state machine variable to determine the NEGOTIATED PHYSICAL LINK RATE field and/or the NEGOTIATED LOGICAL LINK RATE field.

If the phy supports SNW-3, then the SP state machine shall maintain a Commonly Supported Settings state machine variable that contains supported settings common between the phy and the attached phy.

If the phy supports SATA port selectors, the SP state machine shall maintain a COMWAKE\_Received state machine variable to indicate whether a COMWAKE detected message was received in the SP0:OOB\_COMINIT state or the SP1:OOB\_AwaitCOMX state since the last time the SP0:OOB\_COMINIT state was entered, and the SP state machine shall transition to the SP25:SATA\_PortSel state whenever it receives a Transmit SATA Port Selection Signal request.

The SP state machine sends the following messages to the SP\_DWS state machine (see 6.9):

- a) Start DWS; and
- b) Stop DWS.

The SP state machine receives the following messages from the SP\_DWS state machine:

- a) DWS Lost; and
- b) DWS Reset.

The SP state machine shall maintain the timers listed in table 104.

Table 104 – SP state machine timers

Timer	Initial value
COMSAS Detect Timeout timer	COMSAS detect timeout (see table 87 in 6.6.1)
Await ALIGN Timeout time	Await ALIGN timeout (see table 94 in 6.7.2.2)
Hot-Plug Timeout timer	Hot plug timeout (see table 93 in 6.7.1)
RCDT timer	RCDT (see table 95 in 6.7.4.2)
SNLT timer	SNLT (see table 95 in 6.7.4.2)

SNTT timer	SNTT (see table 95 in 6.7.4.2)
TLT timer	TLT (see table 95 in 6.7.4.2)
MTT timer	MTT (see table 95 in 6.7.4.2)

### 6.8.2 SP transmitter and receiver

The SP transmitter transmits OOB signals and dwords on the physical link based on messages from the SP state machine (see 6.8).

The SP transmitter receives the following messages from the SP state machine:

- a) Transmit COMINIT;
- b) Transmit COMSAS;
- c) Transmit COMWAKE;
- d) Transmit SATA Port Selection Signal;
- e) Transmit D10.2;
- f) Set Rate with a Physical Link Rate argument, and an SSC On or an SSC Off argument;
- g) Transmit ALIGN with an argument indicating the specific type (e.g., Transmit ALIGN (0));
- h) Transmit Phy Capabilities Bits;
- i) Transmit TRAIN Pattern;
- j) Transmit TRAIN\_DONE Pattern; and
- k) Transmit MUX Sequence.

When not otherwise instructed, the SP transmitter transmits D.C. idle.

Upon receiving a Transmit MUX Sequence message, the SP transmitter transmits:

- 1) MUX (LOGICAL LINK 0);
- 2) MUX (LOGICAL LINK 1);
- 3) MUX (LOGICAL LINK 0);
- 4) MUX (LOGICAL LINK 1);
- 5) MUX (LOGICAL LINK 0); and
- 6) MUX (LOGICAL LINK 1).

The SP transmitter shall complete any physical link rate change requested with the Set Rate message within RCDT (see table 95 in 6.7.4.2).

The SP transmitter sends the following messages to the SP state machine:

- a) COMINIT Transmitted;
- b) COMSAS Transmitted;
- c) COMWAKE Transmitted;
- d) SATA Port Selection Signal Transmitted;
- e) TRAIN\_DONE Pattern Transmitted;
- f) Phy Capabilities Bits Transmitted; and
- g) MUX Sequence Transmitted.

The SP receiver receives OOB signals and dwords from the physical link and sends messages to the SP state machine indicating what it has received.

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

- a) Set Rate with a Physical Link Rate argument, and an SSC On argument or an SSC Off argument;
- b) Start Training; and
- c) Abort Training.

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

- a) COMINIT Detected;
- b) COMSAS Detected;
- c) COMWAKE Detected;

- d) COMSAS Completed;
- e) COMWAKE Completed;
- f) ALIGN Received with an argument indicating the specific type (e.g., ALIGN Received (0));
- g) Phy Capabilities Bits Received with arguments indicating the supported settings bits received;
- h) Training Completed;
- i) TRAIN\_DONE Received; and
- j) Dword Received.

The ALIGN Received, Dword Received, and TRAIN\_DONE Received messages are only sent when the SP\_DWS state machine has achieved dword synchronization.

For SATA speed negotiation, the ALIGN Received (0) message includes an argument containing the physical link rate at which the ALIGN (0) primitives were detected. For SAS speed negotiation, only ALIGNs at the physical link rate specified by the last Set Rate message received by the SP transmitter cause ALIGN Received messages.

The SP transmitter relationship to other transmitters is defined in 4.3.2. The SP receiver relationship to other receivers is defined in 4.3.3.

### 6.8.3 OOB sequence states

#### 6.8.3.1 OOB sequence states overview

Figure 154 shows the OOB sequence states. These states are indicated by state names with a prefix of OOB.

[figure 154 is unchanged]

#### 6.8.3.2 SP0:OOB\_COMINIT state

##### 6.8.3.2.1 State description

This state is the initial state for this state machine.

Upon entry into this state, the phy shall:

- a) set the COMWAKE\_Received state machine variable to zero;
- b) send a Stop DWS message to the SP\_DWS state machine;
- c) send a Phy Layer Not Ready confirmation to the link layer;
- d) set the ATTACHED SATA DEVICE bit to zero in the SMP DISCOVER response (see 10.4.3.10);
- e) if this state was entered due to power on, set the ATTACHED SATA PORT SELECTOR bit to zero in the SMP DISCOVER response (see 10.4.3.10); and
- f) if this state was not entered because of a Disable Phy request, send a Transmit COMINIT message to the SP transmitter.

If this state was entered because of a Disable Phy request, then upon entry into this state, this state shall:

- a) ignore COMINIT Detected messages until this state is re-entered due to a power on, hard reset, or Management Reset request; and
- b) set the ResetStatus state machine variable to DISABLED.

If this state was entered due to power on or hard reset, then upon entry into this state, this state shall set the ResetStatus state machine variable to UNKNOWN.

If this state was entered because of a Management Reset request, then upon entry into this state, this state shall:

- a) if the ResetStatus state machine variable is not set to RESET\_IN\_PROGRESS, SPINUP\_HOLD, G1, G2, or G3, set the ResetStatus state machine variable to UNKNOWN; or

- b) if the ResetStatus state machine variable is set to RESET\_IN\_PROGRESS, SPINUP\_HOLD, G1, G2, or G3, set the ResetStatus state machine variable to RESET\_IN\_PROGRESS.

If this state was not entered due to a power on, hard reset, Disable Phy, or Management Reset request, then upon entry into this state, this state shall:

- a) if the ResetStatus state machine variable is not set to PHY\_RESET\_PROBLEM, SPINUP\_HOLD, or UNSUPPORTED\_PHY\_ATTACHED, set the ResetStatus state machine variable to UNKNOWN; or
- b) if the ResetStatus state machine variable is set to PHY\_RESET\_PROBLEM, SPINUP\_HOLD, or UNSUPPORTED\_PHY\_ATTACHED, not change the ResetStatus state machine variable.

If the phy supports SATA port selectors and this state receives a COMWAKE Detected message, this state shall:

- a) if the ResetStatus state machine variable is not set to PHY\_RESET\_PROBLEM, SPINUP\_HOLD, or UNSUPPORTED\_PHY\_ATTACHED, set the ResetStatus state machine variable to PORT\_SELECTOR;
- b) set the COMWAKE\_Received state machine variable to one; and
- c) if the ATTACHED SATA PORT SELECTOR bit is set to zero in the SMP DISCOVER response (see 10.4.3.10):
  - A) set the ATTACHED SATA PORT SELECTOR bit to one in the SMP DISCOVER response; and
  - B) send a SATA Port Selector Change confirmation to the link layer.

This state machine waits for receipt of a COMINIT Transmitted message and/or a COMINIT Detected message.

#### **6.8.3.2.2 Transition SP0:OOB\_COMINIT to SP1:OOB\_AwaitCOMX**

This transition shall occur if this state receives a COMINIT Transmitted message and has not received a COMINIT Detected message.

#### **6.8.3.2.3 Transition SP0:OOB\_COMINIT to SP3:OOB\_AwaitCOMINIT\_Sent**

This transition shall occur if this state receives a COMINIT Detected message and has not received a COMINIT Transmitted message.

#### **6.8.3.2.4 Transition SP0:OOB\_COMINIT to SP4:OOB\_COMSAS**

This transition shall occur if this state receives both a COMINIT Transmitted message and a COMINIT Detected message.

### **6.8.3.3 SP1:OOB\_AwaitCOMX state**

#### **6.8.3.3.1 State description**

Upon entry into this state, the phy shall initialize and start the Hot-Plug Timeout timer if this phy is:

- a) an expander phy; or
- b) an initiator phy or target phy implementing the Hot-Plug Timeout timer.

If the phy supports SATA port selectors and this state receives a COMWAKE Detected message, this state shall:

- a) if the ResetStatus state machine variable is not set to PHY\_RESET\_PROBLEM, SPINUP\_HOLD, or UNSUPPORTED\_PHY\_ATTACHED, set the ResetStatus state machine variable to PORT\_SELECTOR;
- b) set the COMWAKE\_Received state machine variable to one; and
- c) if the ATTACHED SATA PORT SELECTOR bit is set to zero in the SMP DISCOVER response (see 10.4.3.10):



- A) set the ATTACHED SATA PORT SELECTOR bit to one in the SMP DISCOVER response;  
and
- B) send a SATA Port Selector Change confirmation to the link layer.

#### **6.8.3.3.2 Transition SP1:OOB\_AwaitCOMX to SP0:OOB\_COMINIT**

This transition shall occur if the Hot-Plug Timeout timer expires.

If the COMWAKE\_Received state machine variable is set to zero and the ATTACHED SATA PORT SELECTOR bit is set to one in the SMP DISCOVER response (see 10.4.3.10), the state machine shall, before the transition:

- a) set the ATTACHED SATA PORT SELECTOR bit to zero in the SMP DISCOVER response; and
- b) send a SATA Port Selector Change confirmation to the link layer.

Before the transition, if this state was entered from SP0:OOB\_COMINIT, this state shall set the ResetStatus state machine variable to UNKNOWN.

#### **6.8.3.3.3 Transition SP1:OOB\_AwaitCOMX to SP4:OOB\_COMSAS**

This transition shall occur after receiving either a COMINIT Detected message or a COMSAS Detected message. If COMSAS Detected was received, this transition shall include a COMSAS Detected argument.

If the ATTACHED SATA PORT SELECTOR bit is set to one in the SMP DISCOVER response (see 10.4.3.10), the state machine shall, before the transition:

- a) set the ATTACHED SATA PORT SELECTOR bit to zero in the SMP DISCOVER response; and
- b) send a SATA Port Selector Change confirmation to the link layer.

#### **6.8.3.4 SP2:OOB\_NoCOMSASTimeout state**

##### **6.8.3.4.1 State description**

Upon entry into this state, the phy shall initialize and start the Hot-Plug Timeout timer if this phy is:

- a) this phy is an expander phy; or
- b) this phy is an initiator phy or target phy implementing the Hot-Plug Timeout timer.

##### **6.8.3.4.2 Transition SP2:OOB\_NoCOMSASTimeout to SP0:OOB\_COMINIT**

This transition shall occur if the Hot-Plug Timeout timer expires.

##### **6.8.3.4.3 Transition SP2:OOB\_NoCOMSASTimeout to SP4:OOB\_COMSAS**

This transition shall occur after receiving a COMINIT Detected message.

#### **6.8.3.5 SP3:OOB\_AwaitCOMINIT\_Sent state**

##### **6.8.3.5.1 State description**

This state waits for a COMINIT Transmitted message.

If the phy supports SATA port selectors and this state receives a COMWAKE Detected message, this state shall:

- a) if the ResetStatus state machine variable is not set to PHY\_RESET\_PROBLEM, SPINUP\_HOLD, or UNSUPPORTED\_PHY\_ATTACHED, set the ResetStatus state machine variable to PORT\_SELECTOR; and
- b) if the ATTACHED SATA PORT SELECTOR bit is set to zero in the SMP DISCOVER response (see 10.4.3.10):
  - A) set the ATTACHED SATA PORT SELECTOR bit to one in the SMP DISCOVER response;  
and
  - B) send a SATA Port Selector Change confirmation to the link layer.

##### **6.8.3.5.2 Transition SP3:OOB\_AwaitCOMINIT\_Sent to SP4:OOB\_COMSAS**

This transition shall occur after receiving a COMINIT Transmitted message.

**6.8.3.6 SP4:OOB\_COMSAS state****6.8.3.6.1 State description**

Upon entry into this state, the phy shall send a Transmit COMSAS message to the SP transmitter.

This state waits for receipt of a COMSAS Transmitted message and/or a COMSAS Detected message.

If the phy supports SATA port selectors and this state receives a COMWAKE Detected message, this state shall:

- a) if the ResetStatus state machine variable is not set to PHY\_RESET\_PROBLEM, SPINUP\_HOLD, or UNSUPPORTED\_PHY\_ATTACHED, set the ResetStatus state machine variable to PORT\_SELECTOR; and
- b) if the ATTACHED SATA PORT SELECTOR bit is set to zero in the SMP DISCOVER response (see 10.4.3.10):
  - A) set the ATTACHED SATA PORT SELECTOR bit to one in the SMP DISCOVER response; and
  - B) send a SATA Port Selector Change confirmation to the link layer.

**6.8.3.6.2 Transition SP4:OOB\_COMSAS to SP5:OOB\_AwaitCOMSAS\_Sent**

This transition shall occur if this state receives a COMSAS Detected message or this state was entered with a COMSAS Detected argument, and this state has not received a COMSAS Transmitted message.

If the ATTACHED SATA PORT SELECTOR bit is set to one in the SMP DISCOVER response (see 10.4.3.10), the state machine shall set the ATTACHED SATA PORT SELECTOR bit to zero in the SMP DISCOVER response and send a SATA Port Selector Change confirmation to the link layer before the transition.

**6.8.3.6.3 Transition SP4:OOB\_COMSAS to SP6:OOB\_AwaitNoCOMSAS**

This transition shall occur if this state receives both a COMSAS Transmitted message and a COMSAS Detected message.

If the ATTACHED SATA PORT SELECTOR bit is set to one in the SMP DISCOVER response (see 10.4.3.10), the state machine shall set the ATTACHED SATA PORT SELECTOR bit to zero in the SMP DISCOVER response and send a SATA Port Selector Change confirmation to the link layer before the transition.

**6.8.3.6.4 Transition SP4:OOB\_COMSAS to SP7:OOB\_AwaitCOMSAS**

This transition shall occur if this state receives a COMSAS Transmitted message and has not received a COMSAS Detected message.

**6.8.3.7 SP5:OOB\_AwaitCOMSAS\_Sent state****6.8.3.7.1 State description**

This state waits for receipt of a COMSAS Transmitted message.

**6.8.3.7.2 Transition SP5:OOB\_AwaitCOMSAS\_Sent to SP6:OOB\_AwaitNoCOMSAS**

This transition shall occur after receiving a COMSAS Transmitted message.

If this state received a COMSAS Completed message, it shall include a COMSAS Completed argument with the transition.

**6.8.3.8 SP6:OOB\_AwaitNoCOMSAS state****6.8.3.8.1 State description**

This state machine waits for a COMSAS Completed message, which indicates that COMSAS has been completely received.

**6.8.3.8.2 Transition SP6:OOB\_AwaitNoCOMSAS to SP0:OOB\_COMINIT**

This transition shall occur after receiving a COMINIT Detected message.

Before the transition, this state shall set the ResetStatus state machine variable to UNKNOWN.

**6.8.3.8.3 Transition SP6:OOB\_AwaitNoCOMSAS to SP8:SAS\_Start**

This transition shall occur after receiving a COMSAS Completed message, or shall occur if a COMSAS Completed argument was received in the transition.

**6.8.3.9 SP7:OOB\_AwaitCOMSAS state****6.8.3.9.1 State description**

Upon entry into this state, the phy shall initialize and start the COMSAS Detect Timeout timer.

**6.8.3.9.2 Transition SP7:OOB\_AwaitCOMSAS to SP2:OOB\_NoCOMSASTimeout**

This transition shall occur if the phy does not support SATA and the COMSAS Detect Timeout timer expires.

The state machine shall set the MgmtReset state machine variable to zero before the transition.

**6.8.3.9.3 Transition SP7:OOB\_AwaitCOMSAS to SP6:OOB\_AwaitNoCOMSAS**

This transition shall occur after receiving a COMSAS Detected message.

The state machine shall set the MgmtReset state machine variable to zero before the transition.

The state machine shall set the ATTACHED SATA PORT SELECTOR bit to zero in the SMP DISCOVER response (see 10.4.3.10). If the ATTACHED SATA PORT SELECTOR bit in the SMP DISCOVER response was set to one prior to this transition, the state machine shall send a SATA Port Selector Change confirmation to the link layer before the transition.

**6.8.3.9.4 Transition SP7:OOB\_AwaitCOMSAS to SP16:SATA\_COMWAKE**

This transition shall occur if:

- a) the phy supports SATA;
- b) the COMSAS Detect Timeout timer expires; and
  - A) the MgmtReset state machine variable is set to one; or
  - B) the phy does not implement SATA spinup hold.

The state machine shall set the MgmtReset state machine variable to zero before the transition.

The state machine shall set the ATTACHED SATA DEVICE bit to one in the SMP DISCOVER response (see 10.4.3.10) before the transition.

**6.8.3.9.5 Transition SP7:OOB\_AwaitCOMSAS to SP26:SATA\_SpinupHold**

This transition shall occur if:

- a) the phy supports SATA;
- b) the COMSAS Detect Timeout timer expires;
- c) the phy implements SATA spinup hold; and
- d) the MgmtReset state machine variable is set to zero.

The state machine shall set the ATTACHED SATA DEVICE bit to one in the SMP DISCOVER response (see 10.4.3.10) before the transition.

**6.8.4 SAS speed negotiation states****6.8.4.1 SAS speed negotiation states overview**

Figure 155 shows the SAS speed negotiation states, in which the phy has detected that it is attached to a SAS phy or expander phy rather than a SATA phy, and performs the SAS speed negotiation sequence. These states are indicated by state names with a prefix of SAS.

[figure 155 is unchanged.]

Figure 156 shows the SAS speed negotiation states related to SNW-3 and Train-SNW.

[figure 156 is unchanged.]

#### **6.8.4.2 SP8:SAS\_Start state**

##### **6.8.4.2.1 State description**

This is the initial state for the SAS speed negotiation sequence.

Upon entry into this state, the phy shall initialize and start the RCDT timer.

If this state is entered from SP6:OOB\_AwaitNoCOMSAS, then the Current SNW state machine variable shall be set to SNW-1. If this state is not entered from SP6:OOB\_AwaitNoCOMSAS, then the Current SNW state machine variable shall be set to:

- a) SNW-2 if the Current SNW state machine variable is set to SNW-1;
- b) SNW-3 if the Current SNW state machine variable is set to SNW-2, and either SNW-1 is invalid or SNW-2 is valid;
- c) Final-SNW if the Current SNW state machine variable is set to SNW-2, SNW-1 is valid, and SNW-2 is invalid;
- d) Final-SNW if the Current SNW state machine variable is set to SNW-3, SNW-3 is invalid, and SNW-2 is valid; or
- e) Unsupported Phy Attached if the Current SNW state machine variable is set to SNW-3, SNW-3 is invalid, and SNW-2 is invalid.

After the Current SNW state machine variable is updated, if the Current SNW state machine variable is not set to Unsupported Phy Attached, then this state shall:

- a) initialize and start the RCDT timer; and
- b) if the SNW specified by the Current SNW state machine variable is supported, send a Set Rate message to the SP transmitter and SP receiver with:
  - A) the Physical Link Rate argument set to 1.5 Gbps and an SSC Off argument, if the Current SNW state machine variable is set to SNW-1;
  - B) the Physical Link Rate argument set to 3 Gbps and an SSC Off argument, if the Current SNW state machine variable is set to SNW-2;
  - C) the Physical Link Rate argument set to 1.5 Gbps and an SSC Off argument, if the Current SNW state machine variable is set to Final-SNW, SNW-1 was valid and SNW-2 was invalid;
  - D) the Physical Link Rate argument set to 3 Gbps and an SSC Off argument, if the Current SNW state machine variable is set to Final-SNW and SNW-2 was valid; or
  - E) the Physical Link Rate argument set to 1.5 Gbps and either an SSC On argument or an SSC Off argument, if the Current SNW state machine variable is set to SNW-3.

During this state D.C. idle shall be transmitted.

##### **6.8.4.2.2 Transition SP8:SAS\_Start to SP0:OOB\_COMINIT**

This transition shall occur after receiving a COMINIT Detected message.

Before the transition, this state shall set the ResetStatus state machine variable to UNKNOWN.

##### **6.8.4.2.3 Transition SP8:SAS\_Start to SP1:OOB\_AwaitCOMX**

This transition shall occur if the Current SNW state machine variable is set to Unsupported Phy Attached.

Before the transition, this state shall set the ResetStatus state machine variable to UNSUPPORTED\_PHY\_ATTACHED.

#### **6.8.4.2.4 Transition SP8:SAS\_Start to SP9:SAS\_WindowNotSupported**

This transition shall occur after the RCDT timer expires if the SNW indicated by the Current SNW state machine variable is not supported.

#### **6.8.4.2.5 Transition SP8:SAS\_Start to SP10:SAS\_AwaitALIGN**

This transition shall occur after the RCDT timer expires if:

- a) the Current SNW state machine variable is not set to SNW-3; and
- b) the SNW indicated by the Current SNW state machine variable is supported.

#### **6.8.4.2.6 Transition SP8:SAS\_Start to SP27:SAS\_Settings**

This transition shall occur after the RCDT timer expires if:

- a) the Current SNW state machine variable is set to SNW-3; and
- b) SNW-3 is supported.

#### **6.8.4.3 SP9:SAS\_WindowNotSupported state**

##### **6.8.4.3.1 State description**

Upon entry into this state, the phy shall initialize and start the SNTT timer.

During this state D.C. idle shall be transmitted.

##### **6.8.4.3.2 Transition SP9:SAS\_WindowNotSupported to SP14:SAS\_Fail**

This transition shall occur after the SNTT timer expires.

#### **6.8.4.4 SP10:SAS\_AwaitALIGN state**

##### **6.8.4.4.1 State description**

Upon entry into this state, the phy shall:

- a) initialize and start the SNTT timer and SNLT timer;
- b) send a Start DWS message to the SP\_DWS state machine; and
- c) repeatedly send Transmit ALIGN (0) messages to the SP transmitter.

Each time this state receives a DWS Lost message, this state may send a Start DWS message to the SP\_DWS state machine to re-acquire dword synchronization without running a new link reset sequence.

##### **6.8.4.4.2 Transition SP10:SAS\_AwaitALIGN to SP0:OOB\_COMINIT**

This transition shall occur after receiving a DWS Lost message if this state does not send a Start DWS message, or after receiving a COMINIT Detected message.

Before the transition, this state shall set the ResetStatus state machine variable to UNKNOWN.

##### **6.8.4.4.3 Transition SP10:SAS\_AwaitALIGN to SP11:SAS\_AwaitALIGN1**

This transition shall occur if this state receives an ALIGN Received (0) message before the SNLT timer expires.

##### **6.8.4.4.4 Transition SP10:SAS\_AwaitALIGN to SP12:SAS\_AwaitSNW**

This transition shall occur if this state receives an ALIGN Received (1) message before the SNLT timer expires.

##### **6.8.4.4.5 Transition SP10:SAS\_AwaitALIGN to SP14:SAS\_Fail**

This transition shall occur if the SNTT timer expires.

#### **6.8.4.5 SP11:SAS\_AwaitALIGN1 state**

**6.8.4.5.1 State description**

This state shall repeatedly send Transmit ALIGN (1) messages to the SP transmitter.

Each time this state receives a DWS Lost message, this state may send a Start DWS message to the SP\_DWS state machine to re-acquire dword synchronization without running a new link reset sequence.

**6.8.4.5.2 Transition SP11:SAS\_AwaitALIGN1 to SP0:OOB\_COMINIT**

This transition shall occur after receiving a DWS Lost message if this state does not send a Start DWS message, or after receiving a COMINIT Detected message.

Before the transition, this state shall set the ResetStatus state machine variable to UNKNOWN.

**6.8.4.5.3 Transition SP11:SAS\_AwaitALIGN1 to SP12:SAS\_AwaitSNW**

This transition shall occur if this state receives an ALIGN Received (1) message before the SNLT timer expires. This indicates that the other phy has been able to achieve dword synchronization in the current SNW.

**6.8.4.5.4 Transition SP11:SAS\_AwaitALIGN1 to SP14:SAS\_Fail**

This transition shall occur if the SNTT timer expires. This indicates that the other phy has not been able to achieve dword synchronization in the current SNW.

**6.8.4.6 SP12:SAS\_AwaitSNW state****6.8.4.6.1 State description**

This state shall repeatedly send Transmit ALIGN (1) messages to the SP transmitter.

If the Current SNW state machine variable is set to Final-SNW, this state shall send a Start SL\_IR Receiver confirmation to the link layer.

Each time this state receives a DWS Lost message, this state may send a Start DWS message to the SP\_DWS state machine to re-acquire dword synchronization without running a new link reset sequence.

This state waits for the SNTT timer to expire or for a Stop SNTT request.

**6.8.4.6.2 Transition SP12:SAS\_AwaitSNW to SP0:OOB\_COMINIT**

This transition shall occur after receiving a DWS Lost message if this state does not send a Start DWS message, or after receiving a COMINIT Detected message.

Before the transition, this state shall set the ResetStatus state machine variable to UNKNOWN.

**6.8.4.6.3 Transition SP12:SAS\_AwaitSNW to SP13:SAS\_Pass**

This transition shall occur after the SNTT timer expires or after receiving a Stop SNTT request.

**6.8.4.7 SP13:SAS\_Pass state****6.8.4.7.1 State description**

This state determines if:

- a) another SAS SNW is required; or
- b) the SAS speed negotiation sequence is complete.

Each time this state receives a DWS Lost message, this state may send a Start DWS message to the SP\_DWS state machine to re-acquire dword synchronization without running a new link reset sequence.

**6.8.4.7.2 Transition SP13:SAS\_Pass to SP0:OOB\_COMINIT**

This transition shall occur after receiving a DWS Lost message if this state does not send a Start DWS message, or after receiving a COMINIT Detected message.

Before the transition, this state shall set the ResetStatus state machine variable to UNKNOWN.

#### **6.8.4.7.3 Transition SP13:SAS\_Pass to SP8:SAS\_Start**

This transition shall occur if the Current SNW state machine variable is not set to Final-SNW.

#### **6.8.4.7.4 Transition SP13:SAS\_Pass to SP15:SAS\_PHY\_Ready**

This transition shall occur if the Current SNW state machine variable is set to Final-SNW.

### **6.8.4.8 SP14:SAS\_Fail state**

#### **6.8.4.8.1 State description**

This state determines if:

- a) another SAS SNW is required; or
- b) the SAS speed negotiation sequence is complete.

#### **6.8.4.8.2 Transition SP14:SAS\_Fail to SP1:OOB\_AwaitCOMX**

This transition shall occur if the Current SNW state machine variable is set to Final-SNW.

Before the transition, this state shall set the ResetStatus state machine variable to PHY\_RESET\_PROBLEM.

#### **6.8.4.8.3 Transition SP14:SAS\_Fail to SP8:SAS\_Start**

This transition shall occur if the Current SNW state machine variable is not set to Final-SNW.

### **6.8.4.9 SP15:SAS\_PHY\_Ready state**

#### **6.8.4.9.1 State description**

This state waits for a COMINIT Detected message, a DWS Lost message, [a Power management \(Partial\) message](#), [a Power management \(Slumber\) message](#), or a DWS Reset message.

Upon entry into this state, the phy shall:

- a) if multiplexing is enabled (see table 100 in 6.7.4.2.3.3):
  - 1) send a Transmit MUX Sequence message to the SP transmitter; and
  - 2) after receiving MUX Sequence Transmitted, send a Phy Layer Ready (SAS) confirmation to the link layer to indicate that the physical link has been brought up successfully in SAS mode;
- b) if multiplexing is not enabled, send a Phy Layer Ready (SAS) confirmation to the link layer to indicate that the physical link has been brought up successfully in SAS mode;
- c) if the SP transmitter is transmitting at 1.5 Gbps, set the ResetStatus state machine variable to G1;
- d) if the SP transmitter is transmitting at 3 Gbps, set the ResetStatus state machine variable to G2; and
- e) if the SP transmitter is transmitting at 6 Gbps, set the ResetStatus state machine variable to G3.

While in this state dwords from the link layer are transmitted at the negotiated physical link rate at the rate established in the previous SNW.

If multiplexing is disabled, each time this state receives a DWS Lost message, this state may send a Start DWS message to the SP\_DWS state machine to re-acquire dword synchronization without running a new link reset sequence.

NOTE 42 - If multiplexing is enabled and this state receives a DWS Lost message, this state does not send a Start DWS message and the state machine transitions to SP0:OOB\_COMINIT.

**6.8.4.9.2 Transition SP15:SAS\_PHY\_Ready to SP0:OOB\_COMINIT**

This transition shall occur after:

- a) receiving a DWS Lost message, if this state does not send a Start DWS message;
- b) receiving a DWS Lost message followed by a COMINIT Detected message, if this state does not send a Start DWS message; or
- c) receiving a DWS Reset message.

This transition may but should not occur after receiving a COMINIT Detected message before receiving a DWS Lost message, or after receiving a COMINIT Detected message after sending a Start DWS message (i.e., the SP state machine should ignore COMINIT Detected messages unless the SP\_DWS state machine has indicated loss of dword synchronization).

**6.8.4.9.3 Transition SP15:SAS\_PHY\_Ready to SP31:SAS\_PM\_Partial**

This transition shall occur after receiving a Power management (Partial) message.

**6.8.4.9.3 Transition SP15:SAS\_PHY\_Ready to SP32:SAS\_PM\_Slumber**

This transition shall occur after receiving a Power management (Slumber) message.

**6.8.4.10 SP27:SAS\_Settings state****6.8.4.11 SP28:SAS\_TrainSetup****6.8.4.12 SP29:SAS\_Train state****6.8.4.13 SP30:SAS\_TrainingDone state**

*[Clauses 6.8.4.10 through 6.8.4.13 are unchanged.]*

**6.8.5 SATA host emulation states****6.8.5.1 SATA host emulation states overview**

Figure 157 shows the SATA host emulation states, in which the phy has detected that it is attached to a SATA phy and behaves as if it were a SATA host phy, initiating the SATA speed negotiation sequence. These states are indicated by state names with a prefix of SATA.

*The power management states defined in this standard are for SAS initiator phys that support SATA; expander devices that support SATA do not support power management in this standard.*

*[The italic wording and other wording in this clause do not permit expanders to support SATA power management. I think SATA power management support should be allowed.]*

**6.8.6 SATA port selector state SP25:SATA\_PortSel****6.8.7 SATA spinup hold state SP26:SATA\_SpinupHold**

*[no changes to the above clauses]*

**6.8.8 SAS phy power management states****6.8.8.1 SAS phy power management states overview**

Figure xxx shows the SAS phy power management states. These states are for SAS initiator phys, expander phys, and SAS target phys that support the phy power management option. Phys compliant with previous versions of this standard do not support SAS power management.

*[Need new state machine figure here for power management.]*

**6.8.8.2 SP31:SAS\_PM\_Partial state****6.8.8.2.1 State description**

This state waits for a COMINIT Detected message, a COMWAKE Detected message, or a Power Management (Exit) message.



Upon entry to this state, the phy shall be quiesced to match the partial low power mode requirement (see x.x).

#### **6.8.8.2.2 Transition SP31:SAS\_PM\_Partial to SP0:OOB\_COMINIT**

This transition shall occur after receiving a COMINIT Detected message.

Before the transition, this state shall set the ResetStatus state machine variable to UNKNOWN.

#### **6.8.8.2.3 Transition SP31:SAS\_PM\_Partial to SP33:SAS\_COMWAKE**

This transition shall occur after receiving a COMWAKE Detected message or a Power management (Exit) message.

If this state received a COMWAKE Detected message, it shall include a COMWAKE Detected argument with the transition.

### **6.8.8.3 SP32:SAS\_PM\_Slumber state**

#### **6.8.8.3.1 State description**

This state waits for a COMINIT Detected message, a COMWAKE Detected message, or a Power Management (Exit) message.

Upon entry to this state, the phy shall be quiesced to match the slumber low power mode requirement (see x.x).

#### **6.8.8.3.2 Transition SP32:SAS\_PM\_Slumber to SP0:OOB\_COMINIT**

This transition shall occur after receiving a COMINIT Detected message.

Before the transition, this state shall set the ResetStatus state machine variable to UNKNOWN.

#### **6.8.8.3.3 Transition SP32:SAS\_PM\_Slumber to SP33:SAS\_COMWAKE**

This transition shall occur after receiving a COMWAKE Detected message or a Power Management (Exit) message.

If this state received a COMWAKE Detected message, it shall include a COMWAKE Detected argument with the transition.

### **6.8.8.4 SP33:SAS\_COMWAKE state**

#### **6.8.8.4.1 State description**

Upon entry into this state, the phy shall send a Transmit COMWAKE message to the SP transmitter.

This state waits for a COMWAKE Transmitted message and/or a COMWAKE Detected message.

#### **6.8.8.4.2 Transition SP33:SAS\_COMWAKE to SP34:SAS\_AwaitCOMWAKE\_Sent**

This transition shall occur if this state receives a COMWAKE Detected message and this state has not received a COMWAKE Transmitted message, or shall occur if a COMWAKE Detected argument was received in the transition.

#### **6.8.8.4.3 Transition SP33:SAS\_COMWAKE to SP35:SAS\_AwaitNoCOMWAKE**

This transition shall occur if this state receives both a COMWAKE Transmitted message and a COMWAKE Detected message.

#### **6.8.8.4.4 Transition SP33:SAS\_PM\_COMWAKE to SP36:SAS\_AwaitCOMWAKE**

This transition shall occur if this state receives a COMWAKE Transmitted message and has not received a COMWAKE Detected message.

### **6.8.8.5 SP34:SAS\_AwaitCOMWAKE\_Sent state**

#### **6.8.8.5.1 State description**

This state waits for receipt of a COMWAKE Transmitted message.

#### **6.8.8.5.2 Transition SP34:SAS\_AwaitCOMWAKE\_Sent to SP35:SAS\_AwaitNoCOMWAKE**

This transition shall occur after receiving a COMWAKE Transmitted message.

If this state received a COMWAKE Detected message, it shall include a COMWAKE Detected argument with the transition.

#### **6.8.8.6 SP35:SAS\_AwaitNoCOMWAKE state**

##### **6.8.8.6.1 State description**

This state waits for a COMWAKE Detected message or a COMINIT Detected message.

##### **6.8.8.6.2 Transition SP35:SAS\_AwaitNoCOMWAKE to SP0:OOB\_COMINIT**

This transition shall occur after receiving a COMINIT Detected message.

Before the transition, this state shall set the ResetStatus state machine variable to UNKNOWN.

##### **6.8.8.6.3 Transition SP35:SAS\_AwaitNoCOMWAKE to SP37:SAS\_PM\_AwaitALIGN**

This transition shall occur after receiving a COMWAKE Detected message, or shall occur if a COMWAKE Detected argument was received in the transition.

#### **6.8.8.7 SP36:SAS\_AwaitCOMWAKE state**

##### **6.8.8.7.1 State description**

Upon entry into this state, the phy shall initialize and start the COMWAKE Detect Timeout timer [10 ms – new defined timer].

This state waits for a COMWAKE Detected message or a COMINIT Detected message.

##### **6.8.8.7.2 Transition SP36:SAS\_AwaitCOMWAKE to SP0:OOB\_COMINIT**

This transition shall occur if the COMWAKE Detect Timeout timer expires or a COMINIT Detected message is received.

Before the transition, this state shall set the ResetStatus state machine variable to UNKNOWN.

##### **6.8.8.7.3 Transition SP36:SAS\_AwaitCOMWAKE to SP37:SAS\_PM\_AwaitALIGN**

This transition shall occur after receiving a COMWAKE Detected message.

#### **6.8.8.8 SP37:SAS\_PM\_AwaitALIGN state**

##### **6.8.8.8.1 State description**

Upon entry into this state, the phy shall:

- a) Send a Set Rate message to the SP transmitter and SP receiver with:
  - A) the Physical Link Rate argument set to the last successfully negotiated link rate;
  - B) the receiver training settings are restored;
  - C) the SSC argument is restored; and
  - D) the multiplexing argument is restored;
- b) initialize and start the SNTT timer and SNLT timer;
- c) send a Start DWS message to the SP\_DWS state machine; and
- d) repeatedly send Transmit ALIGN (0) messages to the SP transmitter.

Each time this state receives a DWS Lost message, this state may send a Start DWS message to the SP\_DWS state machine to re-acquire dword synchronization without running a new link reset sequence.

##### **6.8.8.8.2 Transition SP37: SAS\_PM\_AwaitALIGN to SP0:OOB\_COMINIT**

This transition shall occur after receiving a DWS Lost message if this state does not send a Start DWS message (i.e., this happens after SNLT timer expires), or after receiving a COMINIT Detected message.

Before the transition, this state shall set the ResetStatus state machine variable to UNKNOWN.

#### **6.8.8.8.3 Transition SP37: SAS\_PM\_AwaitALIGN to SP38:SAS\_ReceivedALIGN0**

This transition shall occur if this state receives an ALIGN Received (0) message before the SNLT timer expires.

#### **6.8.8.8.4 Transition SP37: SAS\_PM\_AwaitALIGN to SP39:SAS\_TransmitALIGN1**

This transition shall occur if this state receives an ALIGN Received (1) message before the SNLT timer expires.

### **6.8.8.9 SP38:SAS\_ReceivedALIGN0 state**

#### **6.8.8.9.1 State description**

This state shall repeatedly send Transmit ALIGN (1) messages to the SP transmitter.

Each time this state receives a DWS Lost message, this state may send a Start DWS message to the SP\_DWS state machine to re-acquire dword synchronization without running a new link reset sequence.

This state waits for the SNLT timer to expire.

#### **6.8.8.9.2 Transition SP38:SAS\_ReceivedALIGN0 to SP0:OOB\_COMINIT**

This transition shall occur after receiving a DWS Lost message if this state does not send a Start DWS message, or after receiving a COMINIT Detected message.

Before the transition, this state shall set the ResetStatus state machine variable to UNKNOWN.

#### **6.8.8.9.3 Transition SP38:SAS\_ReceivedALIGN0 to SP39:SAS\_TransmitALIGN1**

This transition shall occur if this state receives an ALIGN Received (1) message before the SNLT timer expires. This indicates that the other phy has been able to achieve dword synchronization.

### **6.8.8.10 SP39:SAS\_TransmitALIGN1 state**

#### **6.8.8.10.1 State description**

This state shall repeatedly send Transmit ALIGN (1) messages to the SP transmitter.

Each time this state receives a DWS Lost message, this state may send a Start DWS message to the SP\_DWS state machine to re-acquire dword synchronization without running a new link reset sequence.

This state waits for the SNTT timer to expire or for a Stop SNTT request.

#### **6.8.8.10.2 Transition SP39:SAS\_TransmitALIGN1 to SP0:OOB\_COMINIT**

This transition shall occur after receiving a DWS Lost message if this state does not send a Start DWS message (i.e., this happens after SNTT timer expires), or after receiving a COMINIT Detected message.

Before the transition, this state shall set the ResetStatus state machine variable to UNKNOWN.

#### **6.8.8.10.3 Transition SP39:SAS\_TransmitALIGN1 to SP15:SAS\_PHY\_Ready**

This transition shall occur after the SNTT timer expires or after receiving a Stop SNTT request.

## **6.9 SP\_DWS (phy layer dword synchronization) state machine**

**6.10 Multiplexing**

**6.11 Spin-up**

[No changes to the above clauses]

**7 Link layer**

**7.1 Link layer overview**

The link layer defines primitives, address frames, and connections. Link layer state machines interface to the port layer and the phy layer and perform the identification and hard reset sequences, connection management, and SSP, STP, and SMP specific frame transmission and reception.

**7.2 Primitives**

**7.2.1 Primitives overview**

[No changes to this clause.]

**7.2.2 Primitive summary**

Table 107 defines the deletable primitives.

[No changes to table 107.]

Table 108 defines the primitives not specific to the type of connection.

[\[Add these items to table 108.\]](#)

Primitive	Use	from			to			Primitive Sequence Type
		I	E	T	I	E	T	
<a href="#">PMACK</a>	NoConn	I	E	T	I	E	T	Extended
<a href="#">PMNACK</a>	NoConn	I	E	T	I	E	T	Extended
<a href="#">PMREQ_P</a>	NoConn	I	E	T	I	E	T	Extended
<a href="#">PMREQ_S</a>	NoConn	I	E	T	I	E	T	Extended

Table 109 defines the primitives used only inside SSP and SMP connections.

[No changes to table 109.]

Table 110 lists the primitives used only inside STP connections and on SATA physical links.

[No changes to table 110.]

**7.2.3 Primitive encodings**

Table 111 defines the primitive encoding for deletable primitives.

[No changes to table 111.]

Table 112 defines the primitive encoding for primitives not specific to type of connection.

[\[Add these items to table 112.\]](#)

Primitive	character				hexadecimal
	1st	2nd	3rd	4th	
<a href="#">PMACK</a>	K28.5				
<a href="#">PMNACK</a>	K28.5				
<a href="#">PMREQ_P</a>	K28.5				
<a href="#">PMREQ_S</a>	K28.5				

Table 113 defines the primitive encodings for primitives used only inside SSP and SMP connections.

[no changes to table 113.]

Table 114 lists the primitive encodings for primitives used only inside STP connections and on SATA physical links.

[no changes to table 114.]

**7.2.4 Primitive sequences**

**7.2.5 Deletable primitives**

[no changes to the above clauses]

**7.2.6 Primitives not specific to type of connections**

**7.2.6.1 AIP (Arbitration in progress)**

AIP is sent by an expander device after a connection request to specify that the connection request is being processed and specify the status of the connection request.

The versions of AIP representing different statuses are defined in table 119.

**Table 119 — AIP primitives**

Primitive	Description
AIP (NORMAL)	Expander device has accepted the connection request. This may be sent multiple times (see 7.12.4.3).
AIP (RESERVED 0)	Reserved. Processed the same as AIP (NORMAL).
AIP (RESERVED 1)	
AIP (RESERVED 2)	
AIP (WAITING ON CONNECTION)	Expander device has determined the routing for the connection request, but either the destination phys are all being used for connections or there are insufficient routing resources to complete the connection request. This may be sent multiple times (see 7.12.4.3).
AIP (WAITING ON DEVICE)	Expander device has determined the routing for the connection request and forwarded it to the output physical link. This is sent one time (see 7.12.4.3).
AIP (WAITING ON PARTIAL)	Expander device has determined the routing for the connection request, but the destination phys are all busy with other partial pathways. This may be sent multiple times (see 7.12.4.3).
AIP (RESERVED WAITING ON PARTIAL)	Reserved. Processed the same as AIP (WAITING ON PARTIAL).

See 7.12 for details on connections.

**7.2.6.2 BREAK**

**7.2.6.3 BREAK\_REPLY**

**7.2.6.4 BROADCAST**

**7.2.6.5 CLOSE**

**7.2.6.6 EOAF (End of address frame)**

**7.2.6.7 ERROR**

**7.2.6.8 HARD\_RESET**

**7.2.6.9 OPEN\_ACCEPT**

[Clauses 7.2.6.2 through 7.2.6.9 are unchanged.]

**7.2.6.10 OPEN\_REJECT**

OPEN\_REJECT specifies that a connection request has been rejected and specifies the reason for the rejection. The result of some OPEN\_REJECTs is to abandon (i.e., not retry) the connection request and the result of other OPEN\_REJECTs is to retry the connection request.

All of the OPEN\_REJECT versions defined in table 122 shall result in the originating port abandoning the connection request.

**Table 122 — Abandon-class OPEN\_REJECT primitives**  
 [Table 122 is unchanged.]

All of the OPEN\_REJECT versions defined in table 123 shall result in the originating port retrying the connection request.

**Table 123 — Retry-class OPEN\_REJECT primitives**

Primitive	originator	Description
OPEN_REJECT (NO DESTINATION) a	Expander phy	An expander device in the pathway is not configuring and determines that: a) there is no such destination phy; b) the connection request routes to a destination expander phy in the same expander port as the source expander phy and the expander port is using the subtractive routing method; or c) the SAS address is valid for an STP target port in an STP/SATA bridge, but the initial Register - Device to Host FIS has not been successfully received (see 10.4.3.12).
OPEN_REJECT (PATHWAY BLOCKED) b	Expander phy	An expander device determined the pathway was blocked by higher priority connection requests.
OPEN_REJECT (RESERVED CONTINUE 0) c	Unknown	Reserved. Process the same as OPEN_REJECT (RETRY).
OPEN_REJECT (RESERVED CONTINUE 1) c		
OPEN_REJECT (RESERVED INITIALIZE 0) a	Unknown	Reserved. Process the same as OPEN_REJECT (NO DESTINATION).
OPEN_REJECT (RESERVED INITIALIZE 1) a		
OPEN_REJECT (RESERVED STOP 0) b	Unknown	Reserved. Process the same as OPEN_REJECT (PATHWAY BLOCKED).
OPEN_REJECT (RESERVED STOP 1) b		
OPEN_REJECT (RETRY)	Destination phy or zoning expander phy	Either: a) a phy with destination SAS address exists but is temporarily not able to accept connections (see 7.16.1, 7.17.5, and 7.18.3); <a href="#">[add power management reference here?]</a> b) an expander device in the pathway is configuring and would otherwise have returned OPEN_REJECT (NO DESTINATION)(see 4.7.2 and 7.12.4.2.5); c) an expander device in the pathway is locked and would otherwise have returned OPEN_REJECT (ZONE VIOLATION)(see 4.9.3.5 and 7.12.4.2.5); or d) an expander device in the pathway has reduced functionality (see 4.6.8 and 7.12.4.2.5).

- a If the I\_T Nexus Loss timer is already running, it continues running; if it is not already running, it is initialized and started. Stop retrying the connection request if the I\_T Nexus Loss timer expires.
- b If the I\_T Nexus Loss timer is already running, it continues running. Stop retrying the connection request if the I\_T Nexus Loss timer expires.
- c If the I\_T Nexus Loss timer (see 8.2.2) is already running, it is stopped.

NOTE 49 - Some SAS logical phys compliant with earlier versions of this standard also transmit OPEN\_REJECT (RETRY) if they receive an OPEN address frame while their SL\_CC state machines are in the SL\_CC5:BreakWait state (see 7.14.4.7).

#### **7.2.6.11 SOAF (Start of address frame)**

#### **7.2.6.12 TRAIN**

#### **7.2.6.13 TRAIN\_DONE**

[Clauses 7.2.6.11 through 7.2.6.13 are unchanged.]

#### **7.2.6.14 PMACK**

PMACK specifies the positive acknowledgement of a power management request primitive.

See 7.10 for details of power management.

#### **7.2.6.15 PMNACK**

PMNACK specifies the negative acknowledgement of a power management request primitive.

See 7.10 for details of power management.

#### **7.2.6.16 PMREQ\_P**

PMREQ\_P specifies a request to transition to partial power management state.

See 7.10 for details of power management.

#### **7.2.6.17 PMREQ\_S**

PMREQ\_S specifies a request to transition to slumberl power management state.

See 7.10 for details of power management.

### **7.2.7 Primitives used only inside SSP and SMP connections**

### **7.2.8 Primitives used only inside STP connections and on SATA physical links**

[Clauses 7.2.7 and 7.2.8 are unchanged.]

## **7.3 Physical link rate tolerance management**

### **7.4 Idle physical links**

### **7.5 CRC**

### **7.6 Scrambling**

### **7.7 Bit order of CRC and scrambler**

### **7.8 Address frames**

[Clauses 7.3 through 7.8 are unchanged.]

## **7.9 Identification and hard reset sequence**

### **7.9.1 Identification and hard reset sequence overview**

After the phy reset sequence (see 6.7) has been completed indicating the physical link is using SAS rather than SATA, each logical phy either:

- a) performs an identification sequence by transmitting one or three IDENTIFY address frames (see 7.8.2); or
- b) performs a hard reset sequence by transmitting a HARD\_RESET primitive sequence (see 7.2.6.8).

[Note: Is something needed here for power management recovery sequence?]

If multiplexing (see 6.10) is enabled (see table 100 in 6.7.4.2.3.3), the identification sequence and hard reset sequence begin with a multiplexing sequence (see 6.7.4.3).

Figure 176 shows two phys with multiplexing disabled performing the identification sequence. Only one IDENTIFY address frame is shown in this example.

[Figure 176 is unchanged.]

Figure 177 shows phy A performing the identification sequence and phy B performing the hard reset sequence. Multiplexing is disabled and only one IDENTIFY address frame is shown in this example.

[Figure 177 is unchanged.]

Each logical phy receives an IDENTIFY address frame or a HARD\_RESET primitive sequence from the logical phy to which it is attached. The combination of a phy reset sequence, an optional hard reset sequence followed by another phy reset sequence, and an identification sequence is called a link reset sequence (see 4.4.1).

If a phy receives a valid IDENTIFY address frame within 1 ms of phy reset sequence completion, the SAS address in the outgoing IDENTIFY address frame(s) and the SAS address in the incoming IDENTIFY address frame determine the port to which a phy belongs (see 4.1.4). The phy ignores subsequent IDENTIFY address frames and HARD\_RESET primitives until another phy reset sequence occurs.

If a phy receives a HARD\_RESET primitive sequence within 1 ms of phy reset sequence completion, it shall be considered a reset event and cause a hard reset (see 4.4.2) of the port containing that phy. If a phy does not receive a HARD\_RESET primitive sequence or a valid IDENTIFY address frame within 1 ms of phy reset sequence completion, it shall restart the phy reset sequence.

### 7.9.2 SAS initiator device rules

### 7.9.3 Expander device rules

### 7.9.4 SL\_IR (link layer identification and hard reset) state machines

[Clauses 7.9.2 through 7.9.4 are unchanged.]

[Note: I suspect changes may be needed to SL\_IR state machine for COMWAKE handling or power management.]

## 7.10 Power management

SAS interface power management may be supported in SSP.

SSP target ports may generate PMREQ\_P or PMREQ\_S. SSP initiator ports and expander ports may reply with PMNACK if power management is not permitted or PMACK if power management is supported and permitted.

[Note: Do we need more guidance on when partial or slumber modes should be invoked?]

SATA interface power management ~~is not~~ may be supported in STP.

STP initiator ports shall not generate SATA\_PMREQ\_P or SATA\_PMREQ\_S, ~~or SATA\_PMACK~~. If an STP initiator port receives SATA\_PMREQ\_P or SATA\_PMREQ\_S, it ~~shall~~ may reply with SATA\_PMNAK if power management is not supported or SATA\_PMACK if power management is supported.



If an expander device receives SATA\_PMREQ\_P or SATA\_PMREQ\_S from a SATA device while an STP connection is not open, it shall not forward it to any STP initiator port and shall reply with SATA\_PMNAK. ~~If one of these primitives arrives while an STP connection is open, it may forward the primitive to the STP initiator port.~~ If an expander device receives SATA\_PMREQ\_P or SATA\_PMREQ\_S from a SATA device while an STP connection is open, it may reply with SATA\_PMNAK if power management is not permitted or SATA\_PMACK if power management is supported and permitted.

SCSI idle and standby power conditions, implemented with the START STOP UNIT command (see SBC-3) and the Power Condition mode page (see SPC-4), may be supported by SSP initiator ports and SSP target ports as described in 10.2.10.

ATA idle and standby power modes, implemented with the IDLE, IDLE IMMEDIATE, STANDBY, STANDBY IMMEDIATE, and CHECK POWER MODE commands (see ATA8-ACS), may be supported by STP initiator ports. The ATA sleep power mode, implemented with the SLEEP command, shall not be used.

**7.11 SAS domain changes (Broadcast (Change) usage)**

**7.12 Connections**

**7.13 Rate matching**

**7.14 SL (link layer for SAS logical phys) state machines**

**7.15 XL (link layer for expander logical phys) state machine**

**7.16 SSP link layer**

**7.17 STP link layer**

**7.18 SMP link layer**

[Clauses 7.11 through 7.18 are unchanged.]

**10.2.7.4 Protocol-Specific Port mode page**

The Protocol-Specific Port mode page (see SPC-4) contains parameters that affect SSP target port operation. If the mode page is implemented by one logical unit in a SCSI target device, it shall be implemented by all logical units in the SCSI target device that support the MODE SELECT or MODE SENSE commands.

The mode page policy (see SPC-4) for this mode page shall be either shared or per target port. If a SAS target device has multiple SSP target ports, the mode page policy should be per target port.

Parameters in this mode page:

- a) shall affect all phys in the SSP target port if the mode page policy is per target port; or
- b) shall affect all SSP target ports in the SAS target device if the mode page policy is shared.

Table 216 defines the format of the page for SAS SSP.

**Table 216 — Protocol-Specific Port mode page for SAS SSP**

Bit	7	6	5	4	3	2	1	0
0	PS	SPF	PAGE CODE (19h)					
1	PAGE LENGTH (0Dh)							
2	ALLOW SLUMBER	CONTINUE AWT	BRCST ASYNCH EVENT	READY LED MEANING	PROTOCOL IDENTIFIER (6h)			

3		Reserved	
4	(MSB)	I_T NEXUS LOSS TIME	(LSB)
5			
6	(MSB)	INITIATOR RESPONSE TIMEOUT	(LSB)
7			
8	(MSB)	REJECT TO OPEN LIMIT	(LSB)
9			
10		Reserved	
15			

The PARAMETERS SAVEABLE (PS) bit is defined in SPC-4.

The SUBPAGE FORMAT (SPF) bit is defined in SPC-4 and shall be set to the value defined in table 216.

The PAGE CODE field is defined in SPC-4 and shall be set to the value defined in table 216.

The PAGE LENGTH field is defined in SPC-4 and shall be set to the value defined in table 216.

An ALLOW SLUMBER bit set to 1 specifies that the SAS port is allowed to initiate negotiation for slumber power management mode (see 7.10). Negotiation for slumber power management mode may occur when there are no commands outstanding for that SAS port. An ALLOW SLUMBER bit set to 0 specifies that the SAS port shall not initiate negotiation for slumber power management mode unless the SAS target device transitions to idle or standby power condition.

**Note:** Negotiation to enter partial power management mode is allowed any time the SAS port anticipates an idle period greater than xx usec.

A CONTINUE AWT bit set to one specifies that the SAS port shall not stop the Arbitration Wait Time timer and set the Arbitration Wait Time timer to zero when the SAS port receives an OPEN\_REJECT (RETRY). A CONTINUE AWT bit set to zero specifies that the SAS port shall stop the Arbitration Wait Time timer and set the Arbitration Wait Time timer to zero when it receives an OPEN\_REJECT (RETRY).

A BROADCAST ASYNCHRONOUS EVENT bit set to one specifies that the device server shall enable origination of Broadcast (Asynchronous Event) (see 4.1.13). A BROADCAST ASYNCHRONOUS EVENT bit set to zero specifies that the device server shall disable origination of Broadcast (Asynchronous Event).

The READY LED MEANING bit specifies the READY LED signal behavior (see 10.4.1). Regardless of the mode page policy (see SPC-4) for this mode page, the shared mode page policy shall be applied to the READY LED MEANING bit.

The PROTOCOL IDENTIFIER field is defined in SPC-4 and shall be set to the value defined in table 216 indicating that this is a SAS SSP specific mode page.

**10.4.3.10 DISCOVER function**

[text and tables unchanged until table 269.]

**Table 269 – DISCOVER response (1 of 3)**

Bit	7	6	5	4	3	2	1	0
Byte								
0	SMP FRAME TYPE (41h)							
1	FUNCTION (10h)							
2	FUNCTION RESULT							
3	RESPONSE LENGTH (00h or 1Ah)							
4	(MSB) EXPANDER CHANGE COUNT (LSB)							
5	Reserved							
6	Reserved							
7	Reserved							
8	Reserved							
9	PHY IDENTIFIER							
10	Reserved							
11	Reserved							
12	Resvd	ATTACHED DEVICE TYPE			ATTACHED REASON			
13	Reserved		PHY POWER STATE		NEGOTIATED LOGICAL LINK RATE			
14	Reserved				ATTACHE D SSP INITIATOR	ATTACHE D STP INITIATOR	ATTACHE D SMP INITIATOR	ATTACHE D SATA HOST
15	SATA PORT SELECTOR	Reserved			ATTACHE D SSP TARGET	ATTACHE D STP TARGET	ATTACHE D SMP TARGET	ATTACHE D SATA DEVICE
16	(MSB) SAS ADDRESS (LSB)							
23								
24	(MSB) ATTACHED SAS ADDRESS (LSB)							
31								
32								
33	Reserved					INSIDE ZPSDS PERSISTNT	REQUEST INSIDE ZPSDS	ATTACHE D BRK_REPL CAPABLE
34	Reserved							
39								
12	PROG MIN PHYSICAL LINK RATE				HARDW MIN PHYSICAL LINK RATE			
13	PROG MAX PHYSICAL LINK RATE				HARDW MAX PHYSICAL LINK RATE			
42	PHY CHANGE COUNT							

[Existing text is unchanged until added text below.]

If the phy is a physical phy and a SAS phy or expander phy is attached, the ATTACHED REASON field indicates the value of the REASON field received in the IDENTIFY address frame (see 7.8.2) during the identification sequence. If the phy is a physical phy and a SATA phy is attached, the ATTACHED REASON field shall be set to 0h after the initial Register - Device to Host FIS has been received. If the phy is a virtual phy, the ATTACHED REASON field shall be set to 0h.

The PHY POWER STATE field indicates the power management state for the referenced phy and is described in table new2.

**Table new2 — PHY POWER STATE field description**

Value	Description
00b	Active power state
01b	Partial power state
10b	Slumber power state
11b	Reserved

The NEGOTIATED LOGICAL LINK RATE field is defined in table 271 and indicates the logical link rate being used by the phy. For physical phys, this is negotiated during the link reset sequence. For virtual phys, this field should be set to the maximum physical link rate supported by the expander device. This field may be different from the negotiated physical link rate when multiplexing is enabled.

[No other changes in this clause.]

**10.4.3.15.4 DISCOVER LIST response SHORT FORMAT descriptor**

Table 293 defines the SHORT FORMAT descriptor.

**Table 293 – SHORT FORMAT descriptor**

Bit	7	6	5	4	3	2	1	0
Byte								
0	PHY IDENTIFIER							
1	FUNCTION RESULT							
2	Restrect	ATTACHED DEVICE TYPE			ATTACHED REASON			
3	Restricted		PHY POWER STATE		NEGOTIATED LOGICAL LINK RATE			
4	Restricted				ATTACHE D SSP INITIATOR	ATTACHE D STP INITIATOR	ATTACHE D SMP INITIATOR	ATTACHE D SATA HOST
5	SATA PORT SELECTOR	Restricted			ATTACHE D SSP TARGET	ATTACHE D STP TARGET	ATTACHE D SMP TARGET	ATTACHE D SATA DEVICE
6	VIRTUAL PHY	Reserved			ROUTING ATTRIBUTE			
7	REASON				Reserved			
8	ZONE GROUP							
9	Restricted for DISCOVER response byte 60		INSIDE ZPSDS PERSISTNT	REQUEST INSIDE ZPSDS	Reserve d	ZONE GROUP PERSISTNT	INSIDE ZPSDS	Reserve d
10	ATTACHED PHY IDENTIFIER							
11	PHY CHANGE COUNT							
12	(MSB) ATTACHED SAS ADDRESS							
19	(LSB)							
20	Reserved							
23								

The PHY IDENTIFIER field indicates the phy for which physical configuration link information is being returned.

The FUNCTION RESULT field indicates the value that is returned in the FUNCTION RESULT field in the SMP DISCOVER response for the specified phy (e.g., SMP FUNCTION ACCEPTED, PHY VACANT, or PHY DOES NOT EXIST). If the FUNCTION RESULT field is set to PHY VACANT or PHY DOES NOT EXIST, the rest of the fields in the SHORT FORMAT descriptor shall be ignored.

The fields in the SHORT FORMAT descriptor not defined in this subclause are defined in the SMP DISCOVER response (see 10.4.3.10).

#### **Annex K** (informative)

##### **Primitive encoding**

[Update table K.1 to include coding of the PMREQ\_P, PMREQ\_S, PMACK, and PMNAK primitives.]