From: Gerry Houlder, Seagate Technology <gerry.houlder@seagate.com>
Subj: SAS-2.1: Add low power transceiver options
Date: Oct. 27, 2008

Overview

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

This proposal allows targets, initiators, or expanders to initiate power management requests.

SATA uses 4 primitives for interface power management. SAS will also use 4 primitives but will use encodings chosen from the SAS encoding space. The primitives are:

(a) PM_REQ (PARTIAL) to request partial power condition (transceiver should recover within 35 us);
(b) PM_REQ (SLUMBER) to request slumber power condition (transceiver should recover within 5.03 msec);
(c) PM_ACK to accept change into the requested power condition;
(d) PM_NAK to reject the power condition request.

When a power condition 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, DFE settings, SSC, multiplexing). The intent is that both ends can restart and obtain synchronization without needing to change receiver settings or redo training. Repeating training would increase the recovery time so much that partial power condition may not be feasible.

When an initiator requests an expander to open a link that is in partial power condition, recovery time should be fast enough so any delay can be covered by returning AIP (WAITING ON CONNECTION) until the link is ready. When an initiator requests to open a link that is in slumber power condition, the expander should 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 condition has fast enough recovery so that it may be used by a target-expander link without explicit approval by an initiator. The slumber condition 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 a SAS specific mode page to enable/ disable power management in targets.
(b) use an SMP function to enable/ disable power management in expanders.

There is a desire to add information to SMP functions to indicate links that are in a low power condition. The DISCOVER function and the DISCOVER LIST function are extended to report this.

If a drive is removed while in partial or slumber and another drive is inserted, the drive will send a COMINIT 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 then a COMWAKE sent by an initiator will timeout, so these situations are covered.

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

6.6 Out of band (OOB) signals
[Clauses 6.6.1 through 6.6.4 are unchanged.]

6.6.5 Power Management conditions

During power management conditions (partial or slumber), the phy shall be in D.C. Idle bus condition. The phy requirements specified in Table 58 and Table 61 apply.

Table new5 defines the maximum recovery times from power management conditions.

Table new5 — Power management maximum recovery times

Description	Partial power management condition	Slumber power management condition
Phy wakeup time [a]	5 us	5 ms
Phy recovery time [a]	30 us	30 us
[a] See figure new2.		

[Note: Is this the place for a table describing power management recovery times or should it be in 7.10? I think it should be here because impending split into physical and protocol standards should group this and its referenced table in the physical standard.]

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 condition

Figure new1 shows the sequence to transition from active to power management condition.

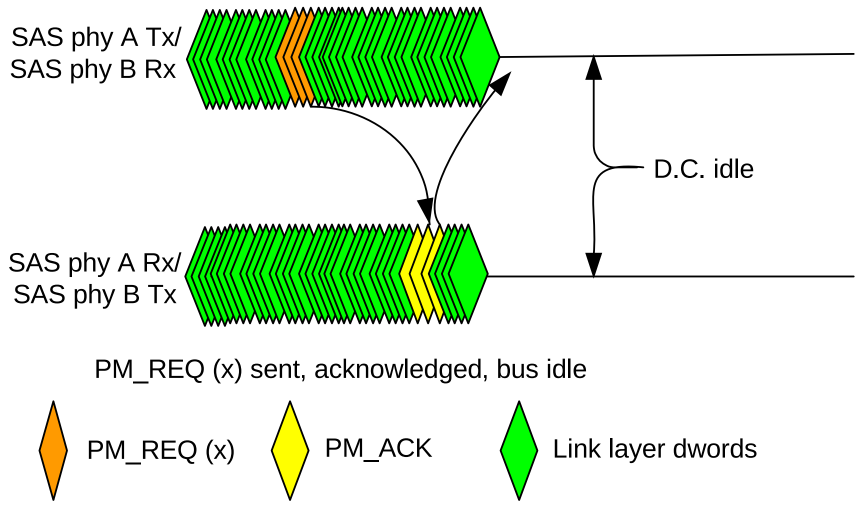

Figure new1 – SAS transition to power management

Figure new1 shows a transition from both ends sending link layer dwords to D.C. idle. The sequence proceeds as follows:

1) phy A sends PM_REQ (x) primitive to phy B;
2) phy B sends PM_ACK primitive to phy A;
3) both phys remember speed, training settings, SSC setting, and multiplexing setting; and
4) both phys transition to the requested D.C. idle condition.

6.7.6.2 COMWAKE sequence to recover from power management condition

The sequence to recover from either partial or slumber power management condition is shown in figure new2.

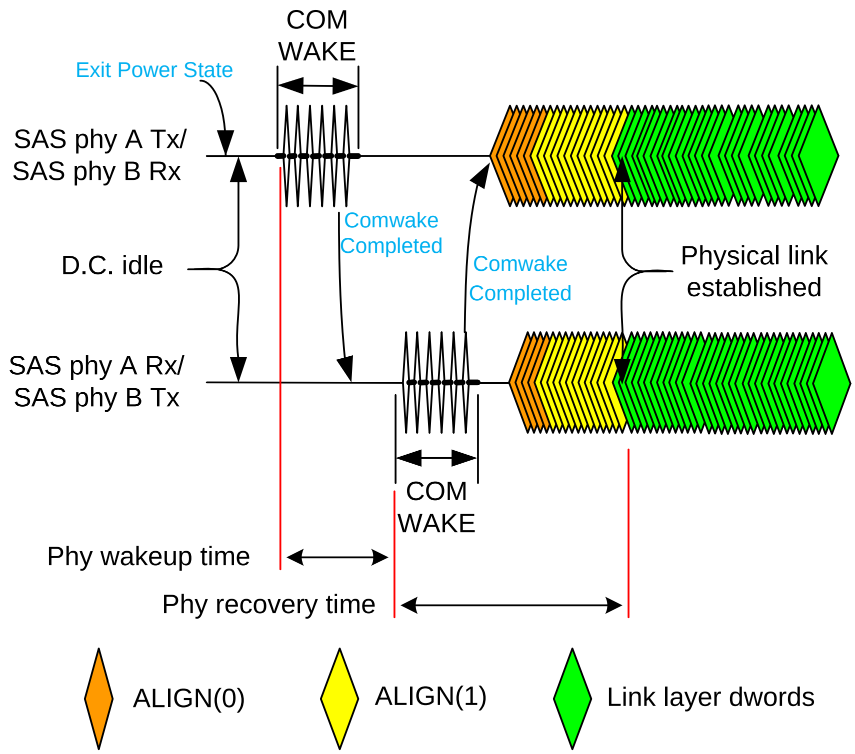

Figure new2 – SAS COMWAKE wakeup sequence

The COMWAKE recovery sequence is as follows:

1) phy A initiates exit power management action by transitioning to active condition;
2) phy A transmits COMWAKE;
3) phy B detects COMWAKE, initiates exit power management action and transmits COMWAKE back to phy A;
4) both phys transmit ALIGN(0) primitives at previously negotiated settings;
5) when each receiver synchronizes on ALIGN(0)s, that transmitter changes to sending ALIGN(1)s;
6) when each receiver synchronizes on ALIGN(1)s, that transmitter changes to link layer dwords; and
7) if phys are multiplexed, the link layer sends the multiplexing sequence (see 6.7.4.3) to align the logical links.

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 condition.

6.7.6.3 Recovery from hot plug events during power management

Figure new3 shows examples of hot-plug scenarios that may occur during power management recovery.

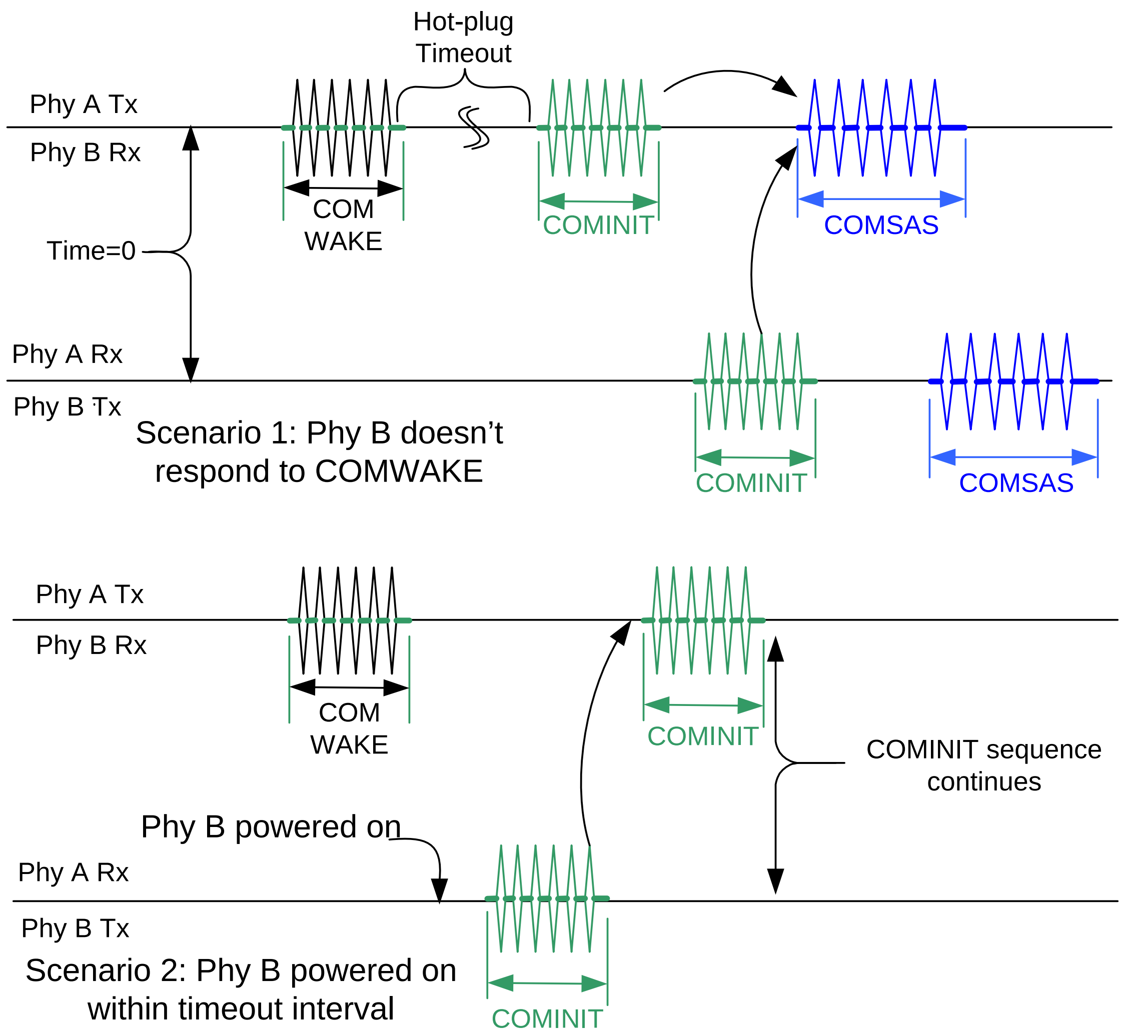

Figure new3 – COMWAKE hot-plug examples

Scenario 1 depicts:

1) phy A starting a power management exit request;
2) phy B target doesn't respond within timeout limit; and
3) phy A recovers by changing to COMINIT sequence.

Scenario 2 depicts:

1) phy A starting a power management exit request;
2) phy B is powered on and begins COMINIT sequence within the timeout limit; and
3) phy A proceeds by continuing the COMINIT sequence.

6.8 SP (phy layer) state machine

[Changes to this clause are detailed in latest revision of 08-206.]

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 112 defines the deletable primitives.
[No changes to table 112.]

Table 113 defines the primitives not specific to the type of connection.
[Add these items to table 113.]

Primitive	Use	from			to			Primitive Sequence Type
		I	E	T	I	E	T	
PM_ACK	NoConn	I	E	T	I	E	T	Extended
PM_NAK	NoConn	I	E	T	I	E	T	Extended
PM_REQ (PARTIAL)	NoConn	I	E	T	I	E	T	Extended
PM_REQ (SLUMBER)	NoConn	I	E	T	I	E	T	Extended

Table 114 defines the primitives used only inside SSP and SMP connections.
[No changes to table 114.]

Table 115 lists the primitives used only inside STP connections and on SATA physical links.
[No changes to table 115.]

7.2.3 Primitive encodings

Table 116 defines the primitive encoding for deletable primitives.
[No changes to table 116.]

Table 117 defines the primitive encoding for primitives not specific to type of connection.
[Note: Add these items to table 117. Need to assign primitive encodings.]

Primitive	character				hexadecimal
	1st	2nd	3rd	4th	
PM_ACK	K28.5				
PM_NAK	K28.5				
PM_REQ (PARTIAL)	K28.5				
PM_REQ (SLUMBER)	K28.5				

Table 118 defines the primitive encodings for primitives used only inside SSP and SMP connections.
[No changes to table 118.]

Table 119 lists the primitive encodings for primitives used only inside STP connections and on SATA physical links.
[No changes to table 119.]

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 124.

Table 124 — 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 127 shall result in the originating port abandoning the connection request.

Table 127 — Abandon-class OPEN_REJECT primitives

[Table 122 is unchanged.]

All of the OPEN_REJECT versions defined in table 128 shall result in the originating port retrying the connection request.

Table 128 — Retry-class OPEN_REJECT primitives

<table>
<tr><th>Primitive</th><th>originator</th><th>Description</th></tr>
<tr><td>OPEN_REJECT (NO DESTINATION) [a]</td><td>Expander phy</td><td>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).</td></tr>
<tr><td>OPEN_REJECT (PATHWAY BLOCKED) [b]</td><td>Expander phy</td><td>An expander device determined the pathway was blocked by higher priority connection requests.</td></tr>
<tr><td>OPEN_REJECT (RESERVED CONTINUE 0) [c]</td><td rowspan="2">Unknown</td><td rowspan="2">Reserved. Process the same as OPEN_REJECT (RETRY).</td></tr>
<tr><td>OPEN_REJECT (RESERVED CONTINUE 1) [c]</td></tr>
<tr><td>OPEN_REJECT (RESERVED INITIALIZE 0) [a]</td><td rowspan="2">Unknown</td><td rowspan="2">Reserved. Process the same as OPEN_REJECT (NO DESTINATION).</td></tr>
<tr><td>OPEN_REJECT (RESERVED INITIALIZE 1) [a]</td></tr>
<tr><td>OPEN_REJECT (RESERVED STOP 0) [b]</td><td rowspan="2">Unknown</td><td rowspan="2">Reserved. Process the same as OPEN_REJECT (PATHWAY BLOCKED).</td></tr>
<tr><td>OPEN_REJECT (RESERVED STOP 1) [b]</td></tr>
<tr><td>OPEN_REJECT (RETRY)</td><td>Destination phy or zoning expander phy</td><td>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);[Note: add power management reference here?]
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</td></tr>
</table>

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

See 7.10 for details of power management.

7.2.6.15 PM_NAK
PM_NAK specifies the negative acknowledgement of a power management request primitive.

See 7.10 for details of power management.

7.2.6.16 PM_REQ (PARTIAL)
PM_REQ (PARTIAL) specifies a request to transition to partial power management condition.

See 7.10 for details of power management.

7.2.6.17 PM_REQ (SLUMBER)
PM_REQ (SLUMBER) specifies a request to transition to slumber power management condition.

See7.10 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
[Clauses 7.3 through 7.7 are unchanged.]

7.8 Address frames

[Changes to clauses 7.8 and 7.9 are described in latest revision of 08-249.]

7.10 Power management

SAS interface power management may be supported on SAS phys and expander phys when no connections are active.

Power management is enabled in SCSI target devices using the SAS Protocol Specific mode page (see 10.2.7.4). [Note: Should we use CONFIGURE GENERAL function for expander control to enable power management on all attached phys?]

If power management is enabled and the most recently received IDENTIFY address frame has the PARTIAL CAPABLE bit set to one (see 7.8.2), then SAS phys and expander phys may generate PM_REQ (PARTIAL). If power management is enabled and the most recently received IDENTIFY address frame has the SLUMBER CAPABLE bit set to one (see 7.8.2), then SAS phys and expander phys may generate PM_REQ (SLUMBER). If power management is enabled, then SAS phys and expander phys may reply with PM_ACK to accept a power management request. If power management is disabled, SAS phys and expander phys shall reject a power management request by replying with PM_NAK or ignoring the request.

If an expander phy attached to a SAS device is in partial power management condition and the expander receives an open address frame for that device, then the expander initiates the exit power management procedure (see x.x) on that phy and may respond with AIP (NORMAL) or AIP (WAITING ON CONNECTION) until the open address frame is sent to the SAS device.

If an expander phy attached to a SAS device is in slumber power management condition and the expander receives an open address frame for that device, then the expander initiates the exit power management procedure (see x.x) on that phy and shall respond with OPEN REJECT (RETRY) until a phy ready condition is established with the SAS device.

If a SCSI target device requires receipt of a NOTIFY (ENABLE SPINUP) primitive before transitioning to a higher SCSI power condition (see 10.2.10), then it shall not generate PM_REQ (PARTIAL) or PM_REQ (SLUMBER) requests and shall not accept PM_REQ (PARTIAL) or PM_REQ (SLUMBER) requests until a NOTIFY (ENABLE SPINUP) is received.

[Note: is special handling needed for NOTIFY (POWER LOSS EXPECTED)? What about the reserved NOTIFYs?)]
[Note: For BROADCAST primitives, should power managed phys be woken up to transmit the broadcast or should the phy be skipped (i.e., no BROADCAST sent to that phy)?]

[Note: Do we need more guidance on when partial or slumber conditions should be invoked and what responses should result?]

SATA interface power management is not supported in STP.

STP initiator ports shall not generate SATA_PMREQ_P, SATA_PMREQ_S, or SATA_PMACK. If an STP initiator port receives SATA_PMREQ_P or SATA_PMREQ_S, it shall reply with SATA_PMNAK.

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.

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)

[Changes to clauses 7.11 through 7.18 are in latest revision of 08-249.]

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 223 — Protocol-Specific Port mode page for SAS SSP

Byte \ Bit	7	6	5	4	3	2	1	0
0	PS	SPF	PAGE CODE (19h)					
1	PAGE LENGTH (0Dh)							
2	ENABLE SLUMBER	CONTINUE AWT	BRCAST ASYNCH EVENT	READY LED MEANING	PROTOCOL IDENTIFIER (6h)			
3	ENABLE PARTIAL							
4	(MSB)	I_T NEXUS LOSS TIME						
5								(LSB)
6	(MSB)	INITIATOR RESPONSE TIMEOUT						
7								(LSB)
8	(MSB)	REJECT TO OPEN LIMIT						
9								(LSB)
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 ENABLE SLUMBER bit set to 1 specifies that the SAS port is enabled to send PM_REQ (SLUMBER) and respond with PM_ACK when a PM_REQ (SLUMBER) is received (see 7.10). An ENABLE SLUMBER bit set to 0 specifies that the SAS port shall not send PM_REQ (SLUMBER) and shall respond with PM_NAK when a PM_REQ (SLUMBER) is received.

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.

An ENABLE PARTIAL bit set to 1 specifies that the SAS port is enabled to send PM_REQ (PARTIAL) and respond with PM_ACK when a PM_REQ (PARTIAL) is received (see 7.10). An ENABLE PARTIAL bit set to 0 specifies that the SAS port shall not send PM_REQ (PARTIAL) and shall respond with PM_NAK when a PM_REQ (PARTIAL) is received.

[No other changes to this clause.]

10.4.3.10 DISCOVER function

[Text and tables unchanged until table 279.]

Table 279 – DISCOVER response (1 of 3)

Byte \ Bit	7	6	5	4	3	2	1	0
0	SMP FRAME TYPE (41h)							
1	FUNCTION (10h)							
2	FUNCTION RESULT							
3	RESPONSE LENGTH (00h or 1Ah)							
4	(MSB)	EXPANDER CHANGE COUNT						
5								(LSB)
6	Reserved							
7	Reserved							

8	Reserved							
9	PHY IDENTIFIER							
10	Reserved							
11	Reserved							
12	Resvd	ATTACHED DEVICE TYPE			ATTACHED REASON			
13	Reserved		PHY PM CONDITION		NEGOTIATED LOGICAL LINK RATE			
14	Reserved				ATTACHED SSP INITIATOR	ATTACHED STP INITIATOR	ATTACHED SMP INITIATOR	ATTACHED SATA HOST
15	SATA PORT SELECTOR	Reserved			ATTACHED SSP TARGET	ATTACHED STP TARGET	ATTACHED SMP TARGET	ATTACHED SATA DEVICE
16	(MSB)	SAS ADDRESS						
23								(LSB)
24	(MSB)	ATTACHED SAS ADDRESS						
31								(LSB)
32	ATTACHED PHY IDENTIFIER							
33	Reserved					INSIDE ZPSDS PERSISTNT	REQUEST INSIDE ZPSDS	ATTACHED 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 table and 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 PM CONDITION field indicates the power management condition for the referenced phy and is described in table new4.

Table new4 — PHY PM CONDITION field description

Value	Description
00b	Active power condition
01b	Partial power condition
10b	Slumber power condition
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 303 defines the SHORT FORMAT descriptor.

Table 303 – SHORT FORMAT descriptor

Byte \ Bit	7	6	5	4	3	2	1	0
0	PHY IDENTIFIER							
1	FUNCTION RESULT							
2	Restrct	ATTACHED DEVICE TYPE			ATTACHED REASON			
3	Restricted		PHY PM CONDITION		NEGOTIATED LOGICAL LINK RATE			
4	Restricted				ATTACHED SSP INITIATOR	ATTACHED STP INITIATOR	ATTACHED SMP INITIATOR	ATTACHED SATA HOST
5	SATA PORT SELECTOR	Restricted			ATTACHED SSP TARGET	ATTACHED STP TARGET	ATTACHED SMP TARGET	ATTACHED 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	Reserved	ZONE GROUP PERSISTNT	INSIDE ZPSDS	Reserved
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 PM_REQ (PARTIAL), PM_REQ (SLUMBER), PM_ACK, and PM_NAK primitives.]