

Note: Substitute for section 6.8 – 6.12.11.3

Revision

1. Change RCD time from 320ns (480UI) to 750,000 UI to assist PLL lockup mechanism
2. Change the OOB state machine to fix problems found in simulation.
3. Remove the CHANGE notification state (SP17: PHY_CHANGE and SP:25 SATA_PHY_CHANGE) from SP state machine preceding entry to the PHY_READY and SATA_PHY_READY states. CHANGE should be scheduled after IDENTIFY for SAS devices in the IR state machine and in whatever SATA state machine that gets enabled by SATA_PHY_READY. Determination of delivery is not known until wide paths are established and this does not happen until after IDENTIFY. Likewise CHANGE is schedule on hot plug exit from the PHY_READY and the SATA_PHY_READY states by the IR and SATA enabled state machines.

1.1 Out of band (OOB) signals

SATA defines “out of band” (OOB) signals used for phy resets. OOB signals are low-speed signal patterns detected by the phy that do not appear in normal data streams. They consist of bursts (repeated sequences) of ALIGNs followed by defined amounts of idle time. During the ALIGN burst time, ALIGN ~~or ALIGN(1)~~ primitives are sent repeatedly for the duration of the burst. During the idle time, the transmitter shall be enabled and transmit either a low or high level but shall not toggle between the levels. The signals are differentiated by the length of idle time between the ~~bursts of~~ ALIGN bursts.

SATA defines two OOB signals: COMINIT/COMRESET and COMWAKE. COMINIT and COMRESET are used in this standard interchangeably. SAS devices identify themselves with an additional SAS-specific OOB signal called COMSAS.

Table 1 defines the timing specifications for OOB signals.

Table 1. OOB signal timing specifications

Parameter	Nominal	Minimum	Maximum	Comments
Unit interval during OOB (UI(OOB)) ^a	666,667 ps	666,600 ps	666,734 ps	The time basis for ALIGN bursts and idle times used to create OOB signals. Based on 1,5 Gbps clock tolerance.
Hot-Plug Timeout	100 ms	10 ms	500 ms	How often a device should resend COMINIT to detect if a device has been attached.
Rate change detect (RCD)	480 UI(OOB) [Editor's note: changing to 7570 000 UI(OOB)]			The time the transmitter shall transmit idle between rates during speed negotiation. Used by the transmitter and receiver to calculate the width of the Speed Negotiation Window.
Speed Negotiation Transmit Time (SNTT)	109,23 μs	109,22 μs	109,24 μs	The time during which ALIGN or ALIGN(1) is transmitted at each rate during negotiation. Derived from: UI(OOB) * 4 096 * 40. Used by the transmitter and receiver to calculate the width of the Speed Negotiation Window.
Speed Negotiation Lock Time (SNLT)			104,475 <u>102,400</u> μs	Receiver's time during the speed negotiation window to detect ALIGN or ALIGN(1) and reply with ALIGN(1). Derived from: SNTT - (3 * (COMSAS Negotiation Time)) <u>or UI(OOB) * 3840 * 40.</u>
COMSAS detect timeout period		10,88 <u>13,653</u> μs		The time a receiver shall allow to detect COMSAS after sending COMSAS. <u>UI(OOB) * 512 * 40.</u>
ALIGN Detect Timeout			880 μs	The time during SATA speed negotiation that an initiator phy shall allow for an ALIGN to be received after detecting the COMWAKE negation period.

^a This is not the same as UI(OOB) defined in SATA; SAS has tighter clock tolerance.

Table 2 describes the OOB signal transmitter requirements for the burst time and idle time that comprise each OOB signal. When the link is synchronized, before sending an OOB signal, the transmitter shall transmit idle for 1 ms to ensure the receiver drops link synchronization and detects the OOB signal.

Table 2. OOB signal transmitter requirements

Signal	ALIGN Burst time	Idle time
COMWAKE	160 UI(OOB)	160 UI(OOB)
COMINIT/RESET	160 UI(OOB)	480 UI(OOB)
COMSAS	160 UI(OOB)	1 440 UI(OOB)

Table 3 describes the OOB signal receiver requirements for the idle times that distinguish each OOB signal.

Table 3. OOB signal receiver requirements

Signal	Idle time		
	may detect	shall detect	shall not detect
COMWAKE	55 ns to 175 ns	101,3 ns to 112 ns	< 55 ns or > 175 ns
COMINIT/ COMRESET	175 ns to 525 ns	304 ns to 336 ns	< 175 ns or > 525 ns
COMSAS	525 ns to 1 575 ns	911,7 ns to 1008 ns	< 525 ns or > 1 575 ns

To send an OOB signal, a transmitter shall send the ALIGN burst followed by idle time at least six times. It is not an error to receive more than six idle time/ALIGN burst pairs. After the last ALIGN burst a COMINIT negation time or COMSAS negation time shall occur. The last idle time serves as the OOB signal negation time and is longer than the normal idle time for that OOB signal. A receiver shall detect an OOB signal after receiving at four consecutive bursts of transitions and idle times. It shall not detect the same signal again until it has detected a different OOB signal (e.g., if the idle time changes) or has detected lack of transitions for greater than the preceding idle time (i.e., a COMINIT negation time for a COMINIT idle time or a COMSAS negation time for a COMSAS idle time). ~~the OOB signal negation time.~~

Table 3 shows the OOB signal negation times for each OOB signal

Table 3. OOB signal negation times

Parameter	Time
COMWAKE negation time	175 ns
COMINIT/COMRESET negation time	525 ns
COMSAS negation time	1 575 ns

The ALIGNs used in OOB signals are not required to be at generation 1 (G1) rates (1,5 Gbps), as this rate might not be supported in future generations of SAS devices. The ALIGNs are only required to generate an envelope for the detection circuitry, as required for any signaling that may be AC coupled. If G2 ALIGNs are used, the number of ALIGNs doubles compared with G1 ALIGNs. The idle time between the burst of ALIGNs is used to distinguish between the OOB signals.

Nominally a SAS transmitter should use ALIGNs at the G1 rate to create the burst portion of the OOB signal, but may use ALIGNs at its slowest supported rate if it does not support the G1 rate and shall not send them at a rate faster than its slowest supported rate.

A SAS receiver shall detect OOB signals comprised of ALIGNs transmitted at any rate up to its highest supported rate. This includes rates below its lowest supported rates.

Figure 1 illustrates the timing required for the generation and detection of all the SAS OOB signals. The signals labeled ~~ComXxxDet-ComWAKEDet, ConInitDet, ComSasDet and ComXxxSent- ComWakeSent, ComInitSent and ComSasSent (where Xxx is Wake, Init, or Sas)~~ are inputs to the SAS phy layer state machine. ~~ComXxxDet-ComWakeDet, ComInitDet and ComSasDet are~~ is an outputs from the OOB detection circuitry that indicates ~~the detection of that~~ an incoming COMINIT, COMWAKE, or COMSAS signal has been detected on the link. ~~ComXxxSent-ComWakeSent, ComInitSent and ComSasSent are~~ is an outputs from the OOB generation circuitry that indicates that a COMWAKE, COMINIT, or COMSAS signal has been completely output onto the link.

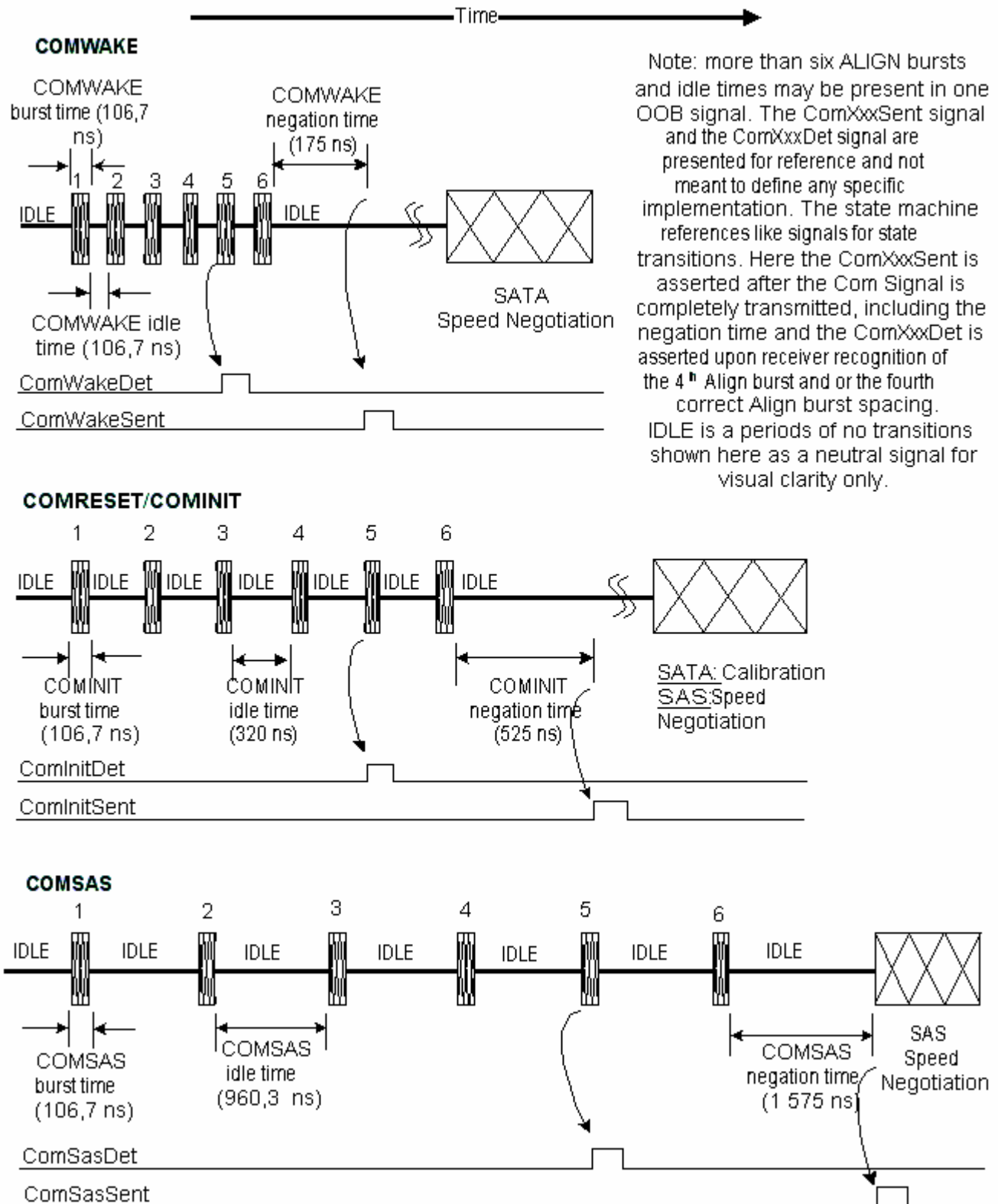


Figure 1. OOB signal transmission and detection

Expanders shall not pass OOB signals. An expander device shall run the link reset sequence independently on each physical link (initiator phy to expander phy, expander phy to expander phy, or expander phy to target phy).

The phy reset sequence shall only affect the phy, not the port or device containing the phy or other phys in the same port or device.

1.2 Phy reset sequences

1.2.1 Phy reset sequence overview

The phy reset sequence shall only affect the phy, not the port or device containing the phy or other phys in the same port or device.

The phy reset sequence consists of SAS OOB sequence and speed negotiation sequence.

After receiving a HARD_RESET, a ~~device-SAS port~~ should start the phy reset sequence within 250 ms.

1.2.2 SATA phy ~~reset-OOB~~ sequence

The link reset sequence between a SATA initiator device and a SATA target device is as defined in SATA.

1.2.3 SAS to SATA OOB sequence

To initiate a phy reset sequence a SAS phs shall transmit a COMINIT. If a SATA phy is attached to the link it either:

- a) misinterprets the COMINIT to be a COMRESET and responds with a COMINIT; or
- b) ignores the COMINIT and provides no response within the COMSAS detect timeout.

Either response indicates to the SAS phy that a SATA phy is attached. As a result the SAS phy shall initiate a SATA reset sequence (i.e., transmit a COMRESET).

Figure 3 defines a SAS phy reset sequence between a SAS phy and a SATA phy (i.e., between an expander device and a SATA target device, or between a SAS initiator device and a SATA target device).

~~Figure 36 depicts the link reset sequence between a SATA initiator device and target device. See SATA for detailed requirements.~~

Figure 2. SATA phy reset sequence

~~1.2.31.2.4~~ SAS to SATA phy reset sequence

SAS phys shall send COMSAS during the SATA calibration phase to identify themselves as SAS phys rather than SATA phys.

~~Figure 37~~ shows a reset sequence between a SAS phy and a SATA phy (i.e., between an expander device and a SATA target device, or between a SAS initiator device and a SATA target device). The two possible cases are presented. The first case is that a legacy SATA phy misinterprets the COMSAS to be a COMRESET and responds with a COMINIT. The second case is that the SATA phy ignores the COMSAS and provides no response within the COMSAS detect timeout. The SAS phy state machine treats these two cases the same, and determines that a SATA phy is attached. The normal SATA reset sequence shall be used thereafter.

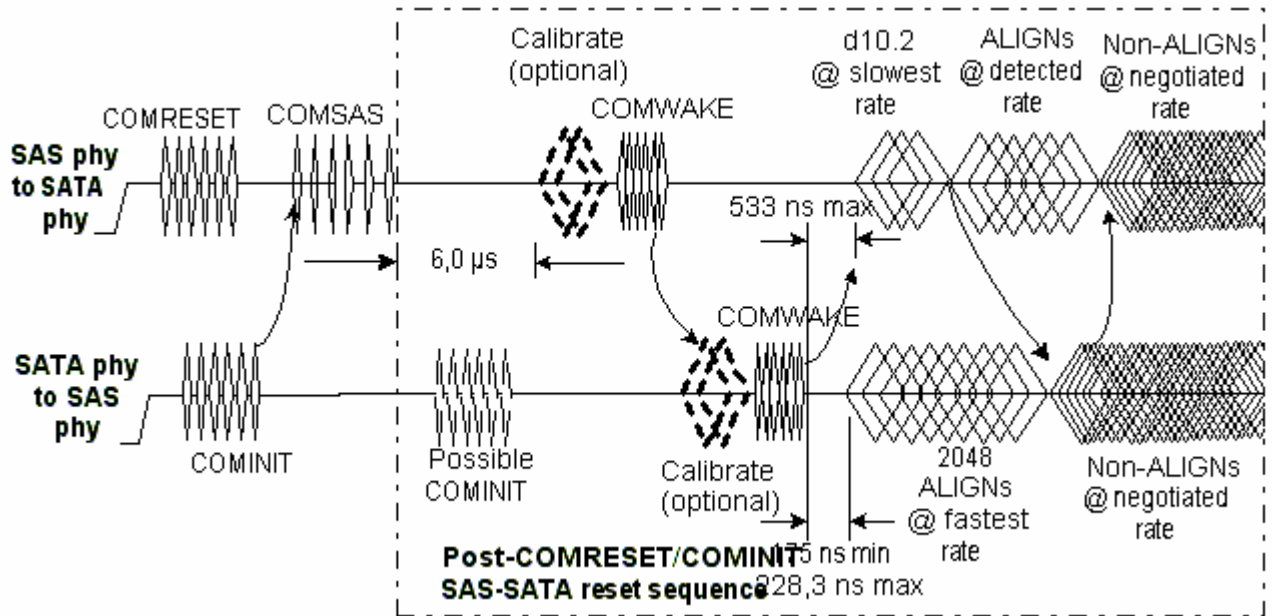


Figure 3. SAS device to SATA device phy reset sequence

4.2.41.2.5 SAS to SAS phy reset sequence

A SAS device shall distinguish between COMINIT and COMSAS and continue with a SAS speed negotiation sequence rather than the SAS reset sequence.

To initiate a SAS OOB sequence a SAS phy shall transmit a COMINIT. If there is no COMINIT or a COMSAS received within a hot plug timeout then the SAS phy shall transmit another COMINIT.

On receipt of a COMINIT a SAS phy shall either:

- a) transmit a COMINIT, if the receiving SAS phy has not yet transmitted a COMSAS, followed by a COMSAS; or
- b) transmit a COMSAS, if the receiving SAS phy has transmitted a COMINIT.

After completing the transmission of a COMSAS and the successful receipt a COMSAS the SAS OOB sequence is complete and the SAS speed negotiation sequence begins.

Figure 1 illustrates several different sequences based on whether a SAS phy powers up before, after, or at the same time as the SAS phy to which it is attached.

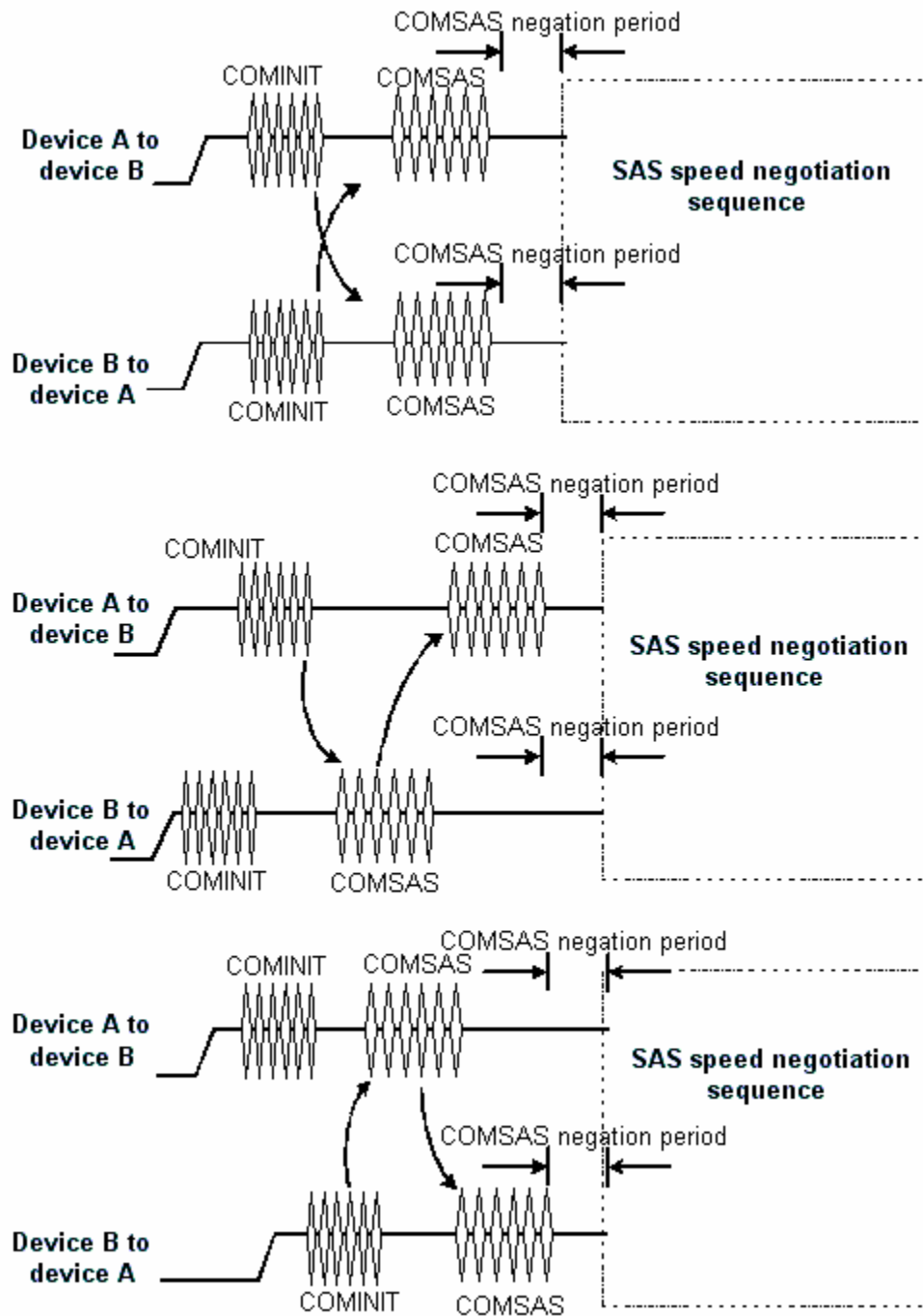


Figure 4. SAS-to-SAS phy reset sequence

1.2.6 SAS speed negotiation sequence

The SAS speed negotiation sequence is a peer-to-peer negotiation technique that does not assume initiator device and target device roles like the SATA speed negotiation sequence. The sequence consists of a set of Speed Negotiation Windows for each link rate, starting with 1,5 Gbps, then 3,0 Gbps, then the next rate. The length of this speed negotiation sequence is determined by the number of link rates supported by the negotiating phys.

The speed negotiation window shall consist of the following transmission sequence for each speed negotiation window:

- 1) transmission of idle for a rate change delay;
- 2) transmission of ALIGNs at that link rate if the phy supports the link rate or transmission of idles for the entire speed negotiation window if the phy does not support the link rate.

If the receiving phy supports the link rate, it shall attempt to synchronize on an incoming series of dwords at the same rate for the speed negotiation lock time. The received dwords may be ALIGNs or ALIGN(1)s. If the phy achieves dword synchronization within the speed negotiation lock time, it shall change from transmitting ALIGNs to transmitting ALIGN(1)s for the remainder of the speed negotiation transmit time (i.e., the remainder of the speed negotiation window). If the device does not achieve dword synchronization within the speed negotiation lock time, it shall continue transmitting ALIGNs for the remainder of the speed negotiation transmit time.

At the end of the speed negotiation transmit time, if a phy is both transmitting and receiving ALIGN(1)s, it shall consider that link rate valid. The phy shall then proceed to the next speed negotiation window. A phy shall participate in all speed negotiation windows:

- up to its highest supported link rate plus one;
- up to a speed negotiation window that does not detect a valid link rate or
- until it has detected a valid link rate in the most recent speed negotiation window and then runs a speed negotiation window that does not detect a valid link rate.

If the phy has detected a valid link rate in the previous speed negotiation window, it shall enter the final speed negotiation window using the highest previously successful link rate.

Any time a SAS phy fails speed negotiation, it shall wait the Hot Plug Timeout Delay before attempting a retry.

For examples of speed negotiations between devices that support various speeds see Annex A.

Each Speed Negotiation Window lasts a Rate Change Delay plus a Speed Negotiation Transfer Time. The sequence may be shortened if a device does not support faster rates.

Figure 5 illustrates a SAS Speed Negotiation Window. First, each device shall transmit idle, (no signal transitions), for a Rate Change Delay. Then, each device shall start transmitting ALIGNs at that link rate if it supports the link rate or shall transmit idle for the entire Speed Negotiation Window if it does not support the link rate.

If the device supports the link rate, it shall attempt to synchronize on an incoming series of dwords at the same rate for the Speed Negotiation Lock Time. The incoming dwords may be ALIGNs or ALIGN(1)s. If the device achieves dword synchronization within the Speed Negotiation Lock Time, it shall change from transmitting ALIGNs to transmitting ALIGN(1)s for the remainder of the Speed Negotiation Transmit Time (i.e., the remainder of the Speed Negotiation Window). If the device does not achieve dword synchronization within the Speed Negotiation Lock Time, it shall continue transmitting ALIGNs for the remainder of the Speed Negotiation Transmit Time. Figure 39 illustrates the Speed Negotiation Window.

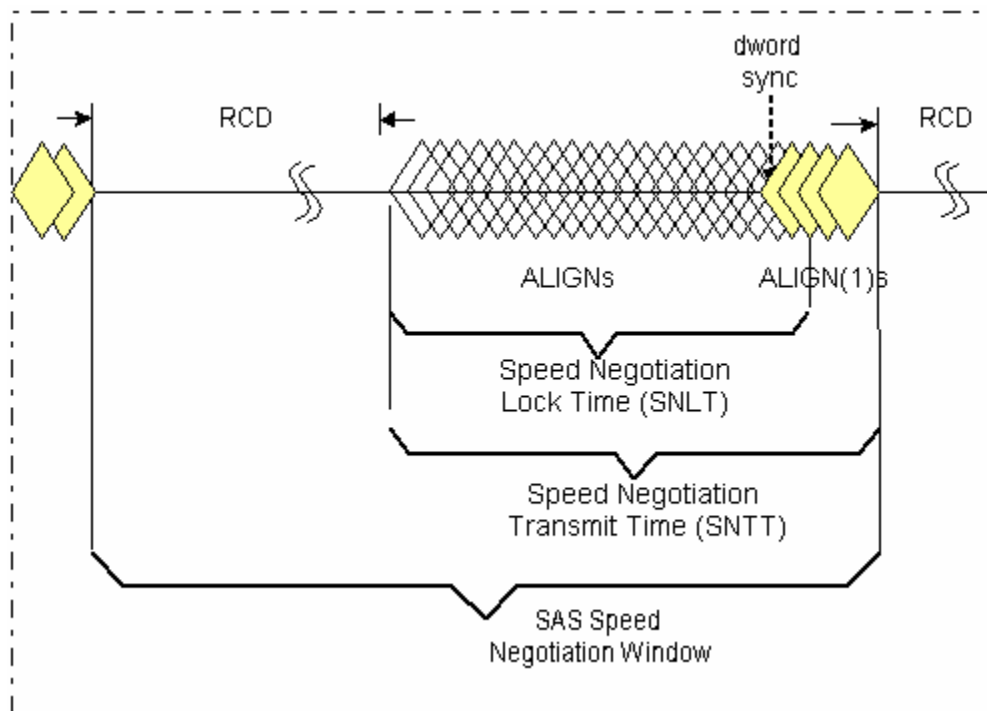


Figure 5. Speed Negotiation Window

At the end of the Speed Negotiation Transmit Time, if a device is both transmitting and receiving ALIGN(1)s, it shall consider that link rate valid for later use. It shall then proceed to the next Speed Negotiation Window. A device shall participate in all Speed Negotiation Windows up to its highest supported link rate plus one, or until it has detected a valid link rate in the most recent Speed Negotiation Window and runs a Speed Negotiation Window that does not detect a valid link rate. At this point, if it has detected a valid link rate in the previous Speed Negotiation Window, it shall enter the final Speed Negotiation Window using the highest previously successful link rate.

Figure 6 shows a speed negotiation between a SAS device A that supports only the G1 link rate attached to a SAS device B that only supports the G1 link rate. They both run the G1 Speed Negotiation Window (valid) and the G2 Speed Negotiation Window (invalid), then select G1 for the final Speed Negotiation Window used to establish the negotiated rate.

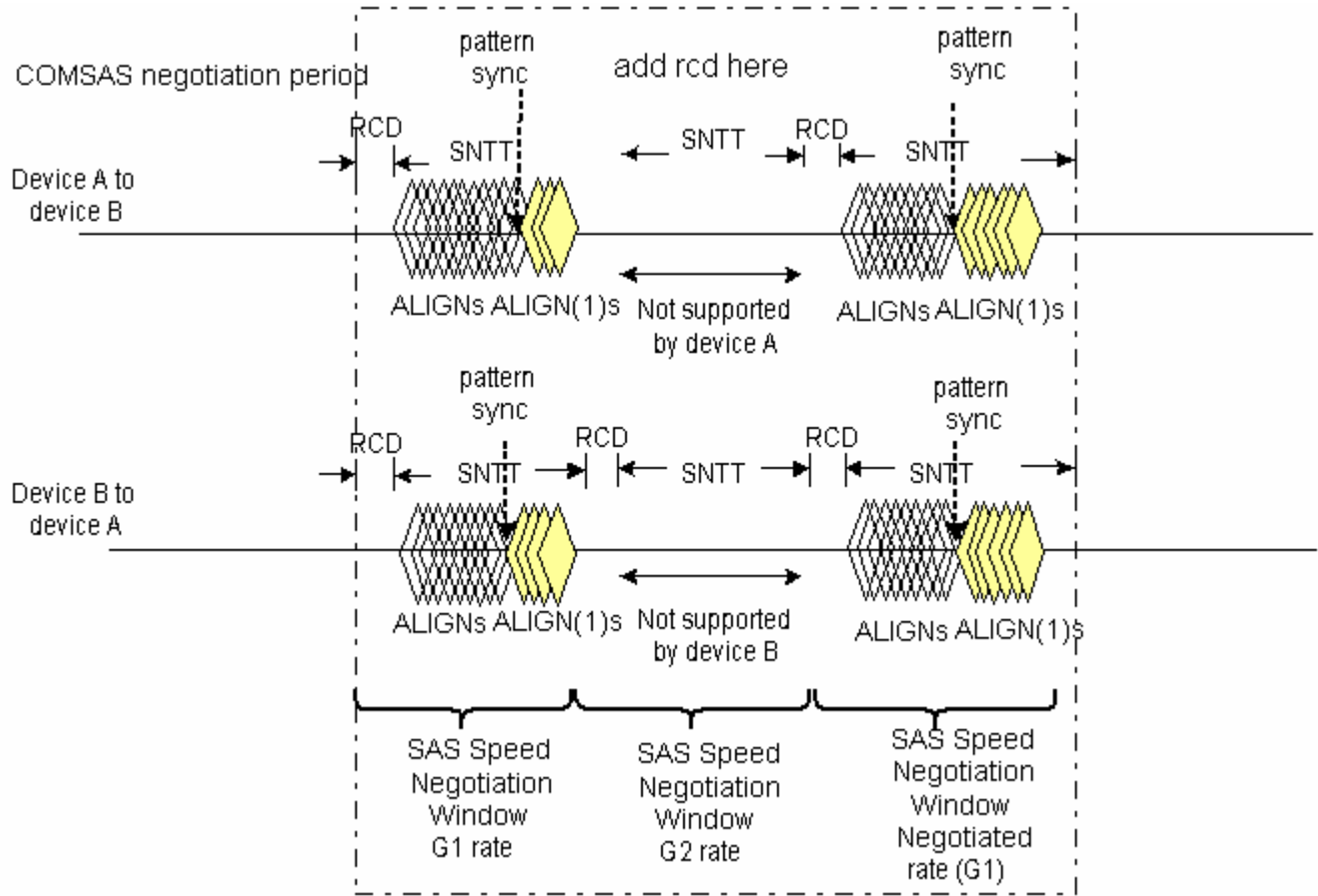


Figure 6. SAS speed negotiation sequence (G1 only with G1 only)

Figure 6 shows a speed negotiation between a SAS device A that supports G1 thru G3 link rates attached to a SAS device B that only supports G1 and G2 link rates. They both run the G1 Speed Negotiation Window (valid), the G2 Speed Negotiation Window (valid), and the G3 Speed Negotiation Window (supported by device A but not by device B, so invalid), then select G2 for the final Speed Negotiation Window used to establish the negotiated rate.

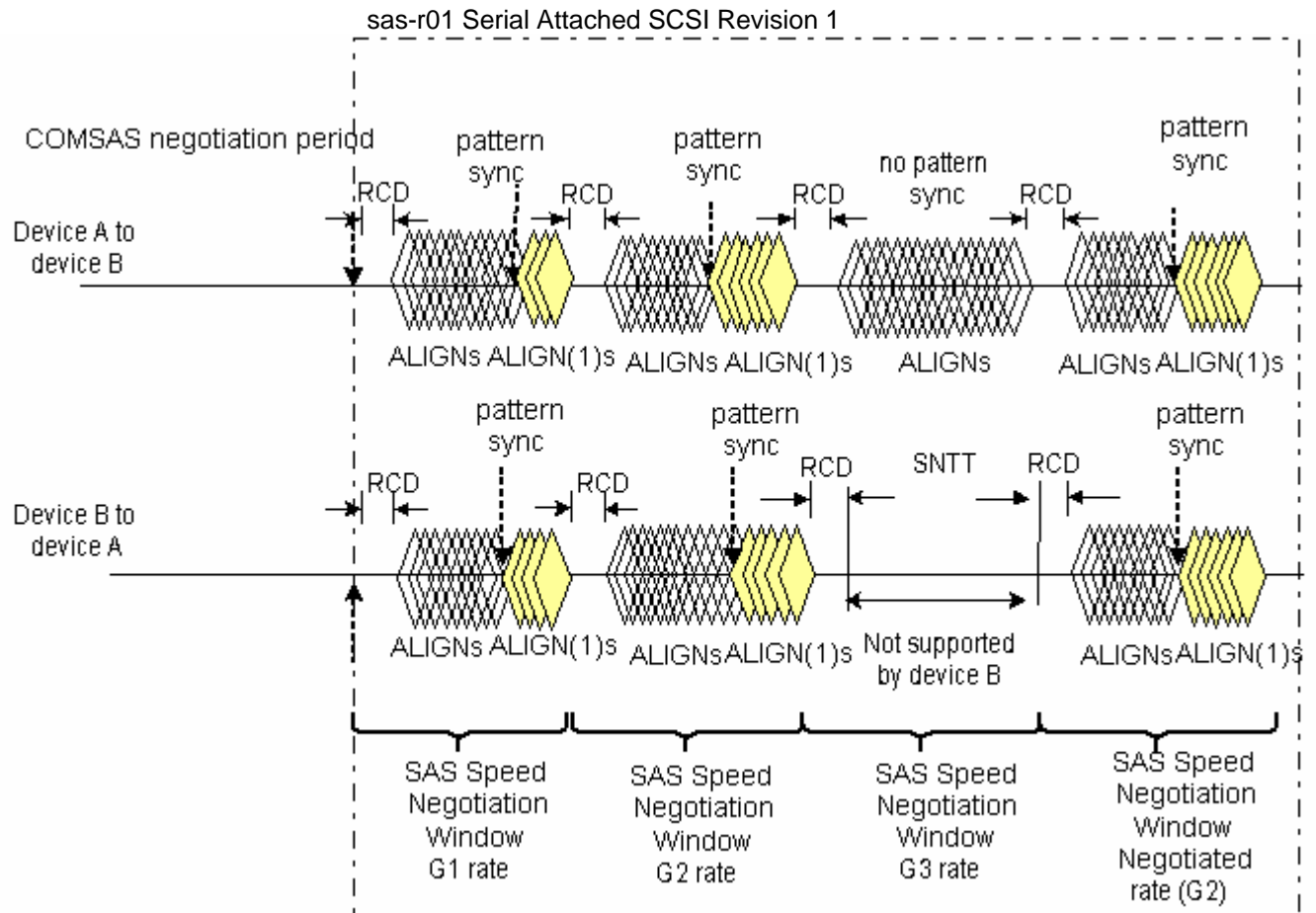


Figure 7. SAS speed negotiation sequence (-G1, G2, G3 with G1, G2)

Figure 7 shows speed negotiation between a SAS device A that supports G1 thru G3 link rates and a SAS device B that only supports the G2 link rate. They both run the G1 Speed Negotiation Window (supported by device A but not by device B, so invalid), the G2 Speed Negotiation Window (valid), and the G3 Speed Negotiation Window (supported by device A but not by device B, so invalid), then select G2 for final Speed Negotiation Window to establish the negotiated rate.

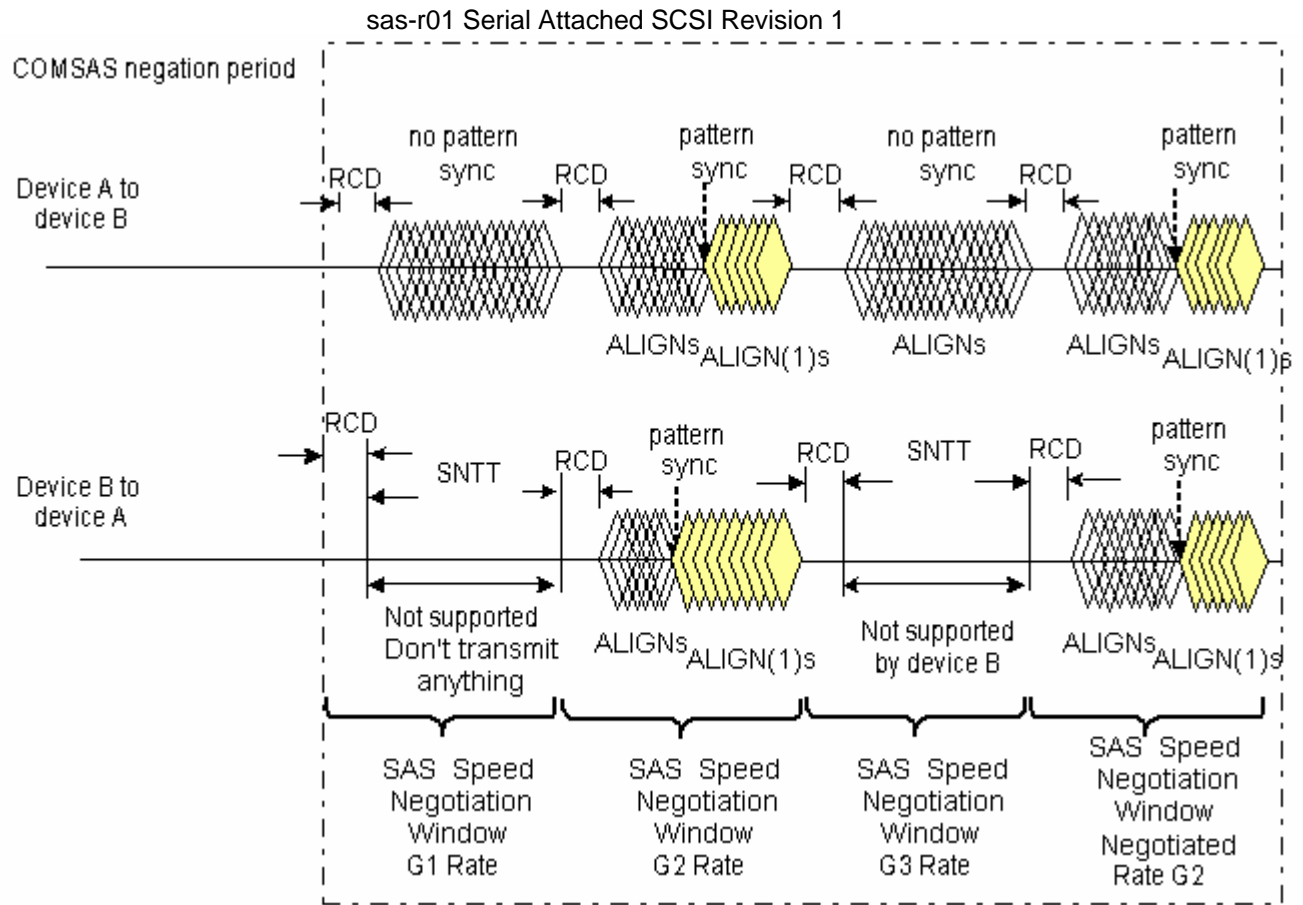


Figure 8. SAS speed negotiation sequence (G1, G2, G3 with G2 only)

Figure 9 shows the same speed negotiation sequence as in Figure 37, except for some reason, one of the SAS device's PLLs does not obtain dword synchronization during the final speed negotiation window. If this occurs, the handshake is not complete and the OOB sequence shall be retried starting with COMINIT, forcing the phy to retry the whole reset sequence.

Any time a SAS phy fails speed negotiation, it shall wait the Hot Plug Timeout Delay before attempting a retry.

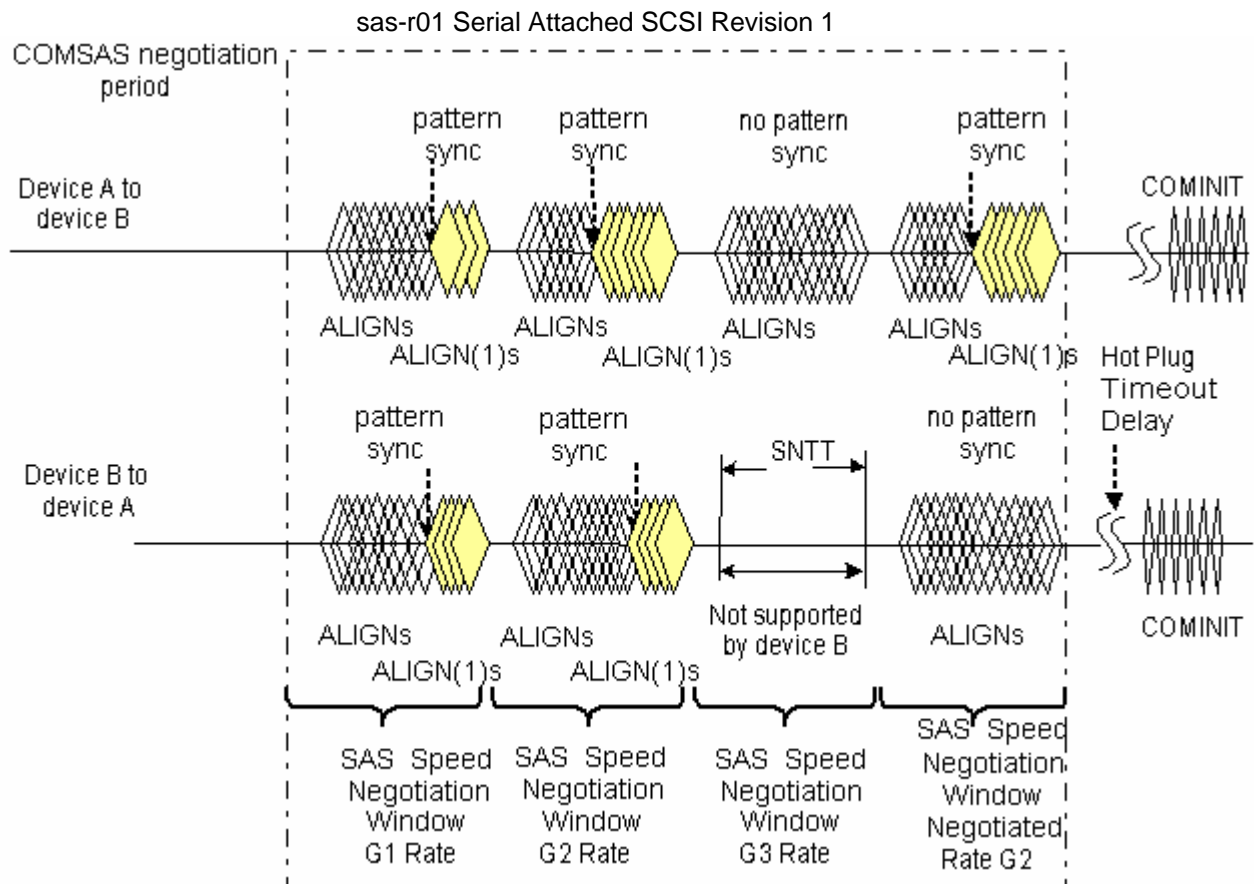


Figure 9. SAS speed negotiation sequence (G1,G2,G3 with G1,G2 and failure)

1.3 Phy reset sequence after signal cable insertion

Since SATA and SAS signal cable connectors do not include power lines or presence indications, it is not possible to detect the physical insertion of the signal cable connector into a receptacle. In order to get around this limitation and SAS phys should periodically transmit a COMINIT sequence if the link is not already established and they have not detected a COMINIT sequence within a Hot-Plug Timeout period.

Figure 10 illustrates how two active SAS phys complete the phy reset sequence if the phys are not attached at power on. In this scenario, SAS phy B is connected to SAS phy A via a cable some time before SAS phy B's second Hot-Plug timeout occurs. SAS phy B's OOB detection circuitry detects a COMINIT from SAS phy A's transmitter after the connection is made. Since it has both sent and detected a COMINIT sequence, and therefore SAS phy B is free to transmit the COMSAS sequence, since it has both sent and detected a COMINIT sequence. Upon detection of COMSAS, SAS phy A transmits its own COMSAS sequence bypassing the sent/received COMINIT requirement. The normal SAS speed negotiation process follows.

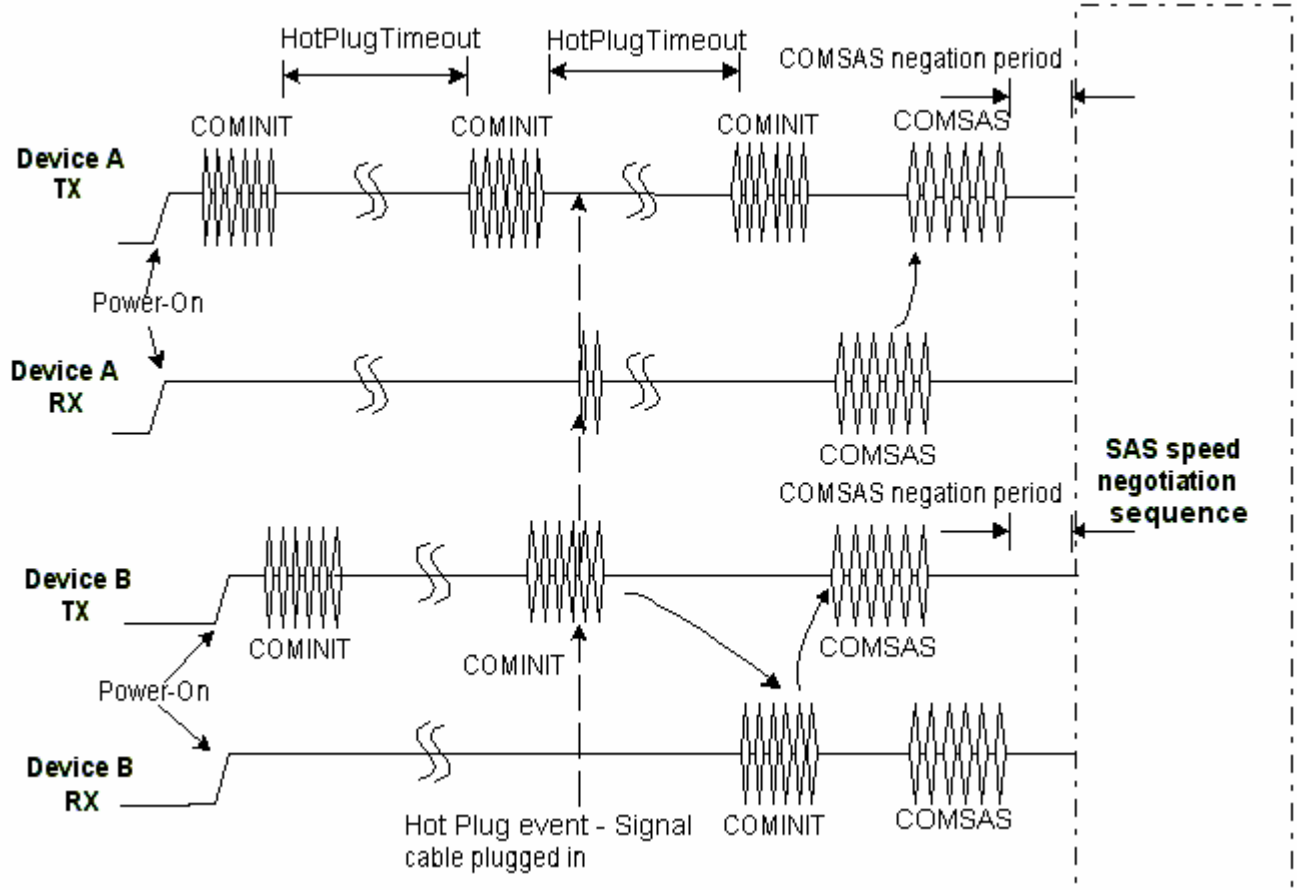


Figure 10. Hot-plug and the phy reset sequence

1.4 SAS phy state machine [NEW]

1.4.1 SAS phy state machine overview

The SAS phy (SP) state machine can be broken down into three sections:

- COMINIT/COMSAS handshake states;
- SAS speed negotiation states; and
- SATA host emulation states.

The SP states are:

- SP0: SAS_PWR_ON
- SP1: SAS_RESET
- SP2: SAS_AWAIT_COMX
- SP3: SAS_AWAIT_COMINIT_SENT
- SP4: SAS_COMSAS
- SP5: SAS_AWAIT_COMSAS_SENT
- SP6: SAS_AWAIT_NO_COMSAS
- SP7: SAS_AWAIT_COMSAS
- SP8: SN_START
- SP9: SN_RATE_NOT_SUPPORTED
- SP10: SN_AWAIT_ALIGN
- SP11: SN_AWAIT_ALIGN
- SP12: SN_AWAIT_SNW
- SP13: SN_PASS

- r) SP14 SN_FAIL
- s) SP15: SN_FALLBACK
- t) SP16: SN_INC_SPEED
- u) SP18: PHY_READY
- v) SP19: SATA_COMWAKE
- w) SP20: SATA_AWAIT_COMWAKE
- x) SP21: SATA_AWAIT_NO_COMWAKE
- y) SP22: SATA_AWAIT_ALIGN
- z) SP23: SATA_ADJUST_SPEED
- aa) SP24: SATA_SEND_ALIGN
- bb) SP26: SATA_PHY_READY
- cc) SP27: PM_SLUMBER
- dd) SP28: PM_PARTIAL

The SP state machine receives the following requests from the link layer:

- ee) ResetPhy.

The SP state machine sends no confirmations to the link layer.

The SP state machine sends the following parameters to the DWS state machines:

- ff) PhyReady(SAS); and
- gg) PhyReady(SATA).

The SAS phy layer shall maintain the following timers:

- hh) OOB Timeout timer,
- ii) Hot Plug Timeout timer;
- jj) Rate Change Delay (RCD) timer,
- kk) Speed Negotiation Lock Timeout (SNLT) timer,
- ll) Speed Negotiation Window Timeout (SNWTO) timer; and
- mm) COMSAS detect timeout timer.

The SP state machine functions to bring SAS links to a working state. There are multiple entry mechanisms to get the SP state machine to its initial SP0:SAS_PWR_ON state.

A power on event forces the SP state machine into the SP0:SAS_PWR_ON state..

Upper protocol layers may force a transition to the SP0:SAS_PWR_ON state. In devices supporting SMP, the PHY CONTROL function may be used to trigger this transition.

The Dword Synchronization state machine (DWS) sets policy on the operational state of the link by monitoring received errors on the link. Entry into the DWS0:AcquireSync state indicates that the link has lost dword synchronization and needs to re-establish synchronization. The SP state machine shall transition to the SP1:SAS_Reset state.

The DWS state machine is active after power-up and may be monitored by the SP state machine to help determine receipt of ALIGN and ALIGN1 primitives. The SP state machine actively monitors the DWS state machine in the PhyReady(SATA/SAS) state to determine if it needs to re-run OOB.

1.4.2 COMINIT/COMSAS handshaking states

Figure 11 shows the COMINIT/COMSAS handshake states, in which the SAS device transmits an initial COMINIT and COMSAS sequences, and awaits to receive a COMINIT and COMSAS sequences back. These states include a prefix of SAS_ in the state name. Once a COMINIT has been detected, the SAS device transmits a COMSAS sequence instead or in place of the normal SATA calibration pattern, and waits to see if the connected device responds with a COMSAS. If the SAS device does not detect a COMSAS sequence within a COMSAS Detect Timeout period, it assumes that a SATA Device is attached, and jumps to the SATA Host emulation states. Otherwise, a SAS-capable port is attached and the SAS speed negotiation states are entered.

1.4.2.2 SP1:SAS_Reset state

1.4.2.2.1 State description

This state is entered under the following conditions:

- a. From SP0:SAS_PWR_ON or when a general reset is requested from the ULP.
- b. It is also entered if COMINIT is detected in states SP18:PHY_Ready or SP26:SATA_PHY_Ready
- c. ALIGN primitives are not detected within an ALIGN Detect Timeout period after entering state SP22:SATA_AwaitALIGN.
- d. This state is entered from state SP18:PHY_Ready or SP26:Sata_Phy_Ready if dword synchronization is lost. Enable Disable Link Layer(SATA/SAS Disable) are real time indications that dword synchronization has been lost. When asserted, this indicates that a significant problem with the link exists,

The state machine exits this state when it has been determined that a COMINIT burst has been sent or detected,

PhyRdy(SAS/SATA) are cleared to 0 upon entering this state. The Fallback and Success flags are also cleared upon entering this state.

1.4.2.2.2 Transition SP1:SAS_Reset to SP2:SAS_AwaitCOMX

Occurs when a COMINIT burst has been sent, a COMINIT was not detected.

1.4.2.2.3 Transition SP1:SAS_Reset to SP3:SAS_AwaitCOMINTSent

Occurs when a COMINIT burst has not been completely transmitted but COMINIT was detected.

1.4.2.2.4 Transition SP1:SAS_Reset to SP4:SAS_COMSAS

Occurs when PwrOnRst is de-asserted, a COMINIT burst has been sent, a COMINIT was detected.

1.4.2.3 SP2:SAS_AwaitCOMX state

1.4.2.3.1 State description

This state is entered when a COMINIT sequence has been sent during state SP0:SAS_RESET and a COMINIT sequence was not detected. Upon entering this state, a Hot-Plug Timeout counter shall be initialized and enabled to count down. The state machine exits this state when it detects a COMINIT, or a COMSAS sequence, or if a Hot-Plug timeout occurs before any of the OOB signals are detected.

This state is also entered from state SP14:SN_Fail if there was not a previous successful lock and the max frequency has been attempted or there was a successful lock but the fallback speed failed. The state machine shall wait the Hot Plug timeout delay before attempting the connection again.

This state is also entered if COMSAS has been sent and COMSAS is not detected in the timeout window and a SATA connection is not supported.

1.4.2.3.2 Transition SP2:SAS_AwaitCOMX to SP1:SAS_Reset

Occurs when a Hot-Plug timeout occurs before any OOB signals are detected.

1.4.2.3.3 Transition SP2:SAS_AwaitCOMX to SP4:SAS_COMSAS

Occurs when a COMINIT has been sent and received or COMSAS sequence is detected. This indicates that a connection has been detected, so the SAS device transmits a COMSAS sequence to inform the other end of the link that it supports SAS protocol.

1.4.2.4 SP3:SAS_AwaitCOMINTSent state

1.4.2.4.1 State description

This state is entered when a COMINIT sequence has been detected but the COMINIT initiated in SP1:SAS_Reset has not been completely sent. The state machine shall wait in this state until the completion of the COMINIT transmission.

1.4.2.4.2 Transition SP3:SAS_AwaitCOMINTSent to SP4:COMSAS

The state machine shall exit this state at the completion of COMINIT transmission being completed.

1.4.2.5 SP4:SAS_COMSAS state

1.4.2.5.1 State description

This state is entered when a COMINIT has been sent and detected. If COMINIT was detected before COMINIT transmission was completed, this state will be entered from the SP3:SAS_AwaitCominitSent state when COMINIT transmission is complete. It can also be entered from SP2:SAS_AwaitCOMX if the full COMINIT handshake is complete or COMSAS has been detected. The state machine remains in this state until the COMSAS sequence is transmitted or detected.

1.4.2.5.2 Transition SP4:SAS_COMSAS to SP5:SAS_AwaitCOMSASSent

Occurs if a COMSAS sequence is detected while COMSAS is still being transmitted. This acknowledges that two SAS ports are connected but COMSAS is not completely sent.

1.4.2.5.3 Transition SP4:SAS_COMSAS to SP6:SAS_AwaitNoCOMSAS

This transition occurs if a COMSAS has been sent and received. This acknowledges that two SAS ports are connected.

1.4.2.5.4 Transition SP4:SAS_COMSAS to SP7:SAS_AwaitCOMSAS

This transition occurs if a COMSAS sequence was not detected while COMSAS was being transmitted. It cannot be determined yet if a SAS or SATA port is connected.

1.4.2.6 SP5:SAS_AwaitCOMSASSent state

1.4.2.6.1 State description

This state is entered when a COMSAS sequence is detected in state SP4:SAS_COMSAS but the full COMSAS sequence has not been completely transmitted.

The state machine remains in this state until the entire COMSAS sequence is transmitted, at which point it will transition to SP6:SAS_AwaitNoCOMSAS to insure that the COMSAS sequence being received is complete.

1.4.2.6.2 Transition SP5:SAS_AwaitCOMSASSent to SP6:SAS_AwaitNoCOMSAS

This transition occurs when the entire COMSAS sequence is transmitted.

1.4.2.7 SP6:SAS_AwaitNoCOMSAS state

1.4.2.7.1 State description

This state is entered when a COMSAS sequence **has been both transmitted and** detected in state SP4:SAS_COMSAS, state SP5:SAS_AwaitCOMSASSent, or state SP7:SAS_AwaitCOMSAS.

The state machine remains in this state until the end of the COMSAS sequence is detected, at which point the SAS speed negotiation begins.

1.4.2.7.2 Transition SP6:SAS_AwaitNoCOMSAS to SP8:SN_Start

This transition occurs when the end of the COMSAS sequence is detected.

1.4.2.8 SP7:SAS_AwaitCOMSAS state

1.4.2.8.1 State description

This state is entered when a COMSAS sequence has been transmitted, but a COMSAS has not yet been detected in state SP4:SAS_COMSAS.

Upon entering this state the COMSAS Timeout counter shall be initialized and enabled to count down. The state machine remains in this state until either a COMSAS sequence is detected or a COMSAS detect timeout period occurs.

1.4.2.8.2 Transition SP7:SAS_AwaitCOMSAS to SP6:SAS_AwaitNoCOMSAS

This transition occurs when a COMSAS sequence is detected.

1.4.2.8.3 Transition SP7:SAS_AwaitCOMSAS to SP19:SATA_COMWAKE

All SAS devices that support SATA target attach shall transition to this state when a COMSAS sequence is not detected within the COMSAS detect timeout period..

1.4.2.8.4 Transition SP7:SAS_AwaitCOMSAS to SAS_AwaitNoCOMX

SAS devices that don't support SATA target attach shall transition to this state when a COMSAS sequence is not detected within the COMSAS detect timeout period.

1.4.3 SAS speed negotiation states

Error! Reference source not found. shows the SAS Speed Negotiation states, in which the SAS phy has detected that it is attached to a SAS phy, and engages in the SAS speed negotiation procedure. These states are indicated by state names with a prefix of SN or PHY.

SAS phy layer state machine - SAS speed negotiation states

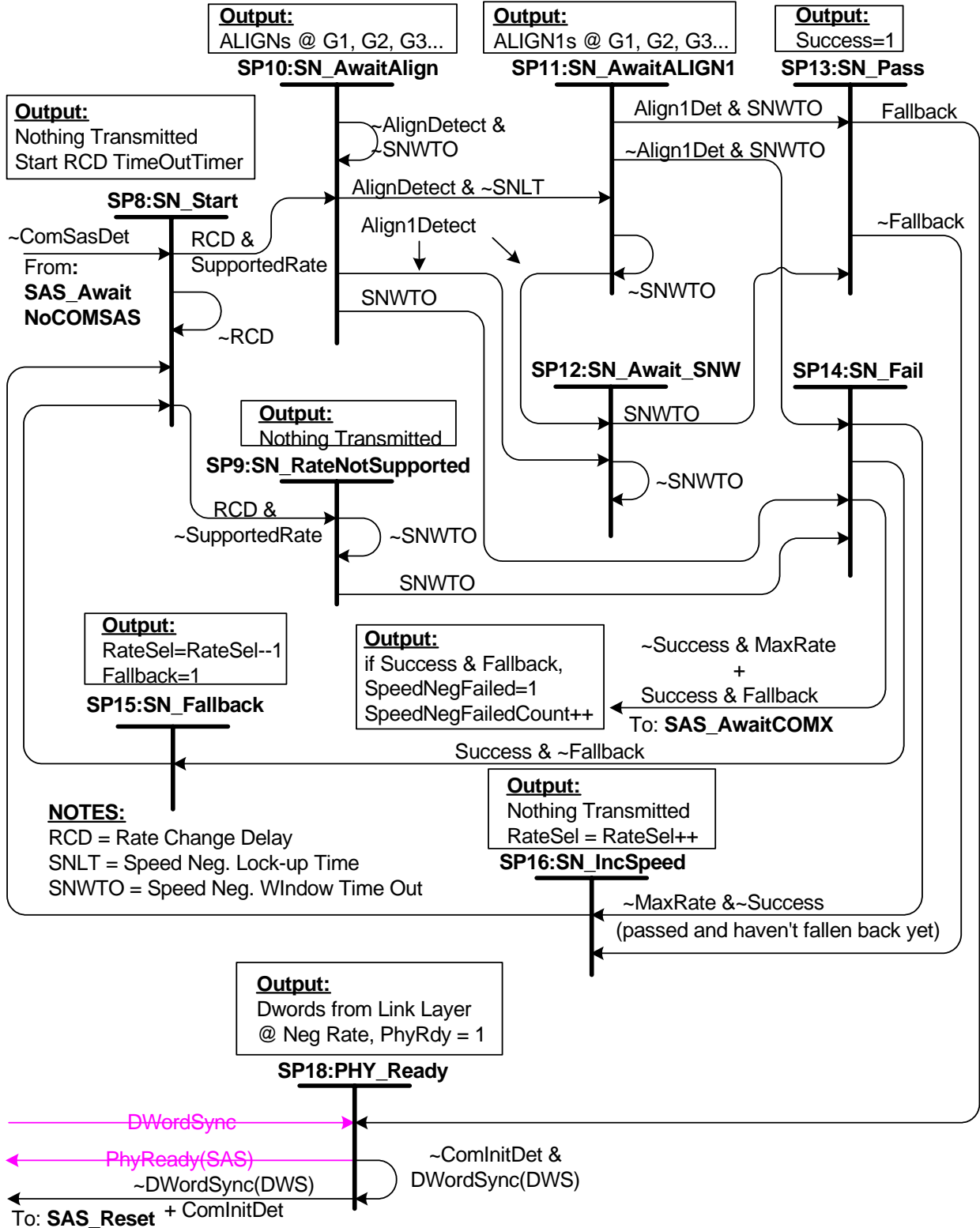


Figure 12. SAS phy state machine - SAS speed negotiation states

1.4.3.1 SP8:SN_Start state**1.4.3.1.1 State description**

This state marks the beginning of the SAS speed negotiation process. It is used to get the transmit lines to an idle state in between SAS speed negotiation windows. A SAS speed negotiation window is defined to be 109,2 μ s (4096 Gen1 UIs). This state is entered from state SP6:SAS_AwaitNoCOMSAS after the end of a COMSAS sequence. It can also be entered from states SP16_Inc Speed or SP15:SN_Fallback for another pass, after a speed negotiation window occurs. Upon entering this state, the Rate Change Delay timer is loaded with an initial value and enabled to count down. This allows time required for a transmitter to switch to either the next higher or next lower supported speed. The state machine remains in this state until a Rate Change Delay has elapsed. The transmitter is turned on but nothing is transmitted during this state. Occurs when the Rate Change Delay timer expires and the current rate is not supported.

1.4.3.1.2 Transition SP8:SN_Start to SP10:SN_AwaitAlign

Occurs when the Rate Change Delay timer expires and the current rate is supported.

1.4.3.1.3 Transition SP8:SN_Start to SP9:SN_RateNotSupported

Occurs when the Rate Change Delay timer expires and the current rate is NOT supported.

1.4.3.2 SP9:SN_RateNotSupported state**1.4.3.2.1 State description**

This state is entered after a Rate Change Delay (RCD) from state SP8:SN_Start when the currently selected rate is not supported. During this state nothing is transmitted for an entire speed negotiation window. Upon entering this state the SNWTO counter is loaded and enabled to count down. The state machine exits from this state after the SNWTO counter expires.

1.4.3.2.2 Transition SP9:SN_RateNotSupported to SP14:SN_Fail

Occurs after the SNWTO counter expires.

1.4.3.3 SP10:SN_AwaitAlign state**1.4.3.3.1 State description**

The state machine starts transmitting ALIGN characters at the current supported rate (Gen1, Gen2, Gen3...) in this state. Upon entering this state, the Speed Negotiation Window Timeout (SNWTO) counter is loaded and enabled to count down. The state machine exits this state when a SNWTO occurs without the pattern synchronization or if synchronization occurs before the Speed Negotiation Lock Time (SNLT) expires.

1.4.3.3.2 Transition SP10:SN_AwaitAlign to SP11:SN_AwaitALIGN1

Occurs when ALIGNs at the current rate are received and recognized before an SNLT period.

1.4.3.3.3 Transition SP10:SN_AwaitAlign to SP12:SN_Await_SNW

Occurs when ALIGN1s at the current rate are received and recognized before an SNLT period.

1.4.3.3.4 Transition SP10:SN_AwaitAlign to SP14:SN_FAIL

Occurs at the end of the Speed Negotiation Window and dword synchronization had not been obtained within the SNLT.

1.4.3.4 SP11:SN_AwaitALIGN1 state**1.4.3.4.1 State description**

This state is entered from state SP10:SN_AwaitAlign if the PLL has locked and obtained Dword synchronization and have detected ALIGNs within the SNLT. During this state the ALIGN1 primitive is transmitted. This state is exited when the SNWTO expires or when ALIGN1s are received before the SNW time out.

1.4.3.4.2 Transition SP11:SN_AwaitALIGN1 to SP14:SN_Fail

Transition to the SN_Fail state occurs if no ALIGN1s were detected within an SNWTO period. This indicates that the other device must not have been able to lock at the current rate.

1.4.3.4.3 Transition SP11:SN_AwaitALIGN1 to SP14:SN_Await_SNW state

Occurs if ALIGN1s were detected within the SNWTO period. This indicates that the other device has been able to lock at the current rate.

1.4.3.5 SP12:SN_Await_SNW state

1.4.3.5.1 State description

This state is entered from state SP10:SN_AwaitAlign or SP11:SN_AwaitAlign1 if the PLL has locked and obtained the appropriate pattern synchronization within an SNLT. During this window the state machine latches the fact that ALIGN1s were received, the speed negotiation is successful and now must only wait till the end of the Speed Negotiation Window.

1.4.3.5.2 Transition SP12:SN_AwaitALIGN1 to SP13:SN_Pass

Speed negotiation at this rate was successful and the state machine needs only to wait till the end of the Speed Negotiation window. At the SNWTO, the state machine will advance to SN_Pass

1.4.3.6 SP13:SN_Pass state

1.4.3.7 State description

This state is entered from state SP11:SN_AwaitALIGN1 or SP12:SN_Await_SNW after an SNWTO occurs. This is a single clock state that set the Success flag, indicating that there was at least one successful speed negotiation.

1.4.3.7.1 Transition SP13:SN_Pass to SP16:SN_IncSpeed

If the state machine has not fallen back, this means that it has not progressed past the breaking point so the speed is incremented for the next attempt.

1.4.3.7.2 Transition SP13:SN_Pass to SP18:PHY_Ready

The transition to the SP18:PHY_Ready state is taken if the speed negotiation has completed. This is indicated by the Fallback flag. If this flag is set it indicates that the speed negotiation has progressed to where it failed and then had fallen back to the last negotiated speed and then subsequently passed.

1.4.3.8 SP14:SN_Fail state

1.4.3.8.1 State description

This state is entered from the SP11:SN_AwaitALIGN state or SP11:SN_AwaitALIGN1 state if ALIGN1s are not received indicating that the other end does not support the rate or has NOT locked at this rate.

1.4.3.8.2 Transition SP14:SN_Fail to SP2:SAS_AwaitCOMX

Occurs when the state machine has not been successful, meaning that the max rate has been attempted and there haven't been any successful rate negotiated or when the speed negotiation fails after dropping back to the last working frequency.

1.4.3.8.3 Transition SP14:SN_Fail to SP15:SN_Fallback

Transition to fallback if the speed negotiation has failed and there was a previous success.

1.4.3.8.4 Transition SP14:SN_Fail to SP16:SN_IncSpeed

Transition to SP16:SN_IncSpeed if the speed negotiation has not previously passed and the maximum supported rate has not yet been attempted.

1.4.3.9 SP15:SN_Fallback state

1.4.3.9.1 State description

This state is entered from state SP14:SN_Fail after an SNWTO occurs indicating that the current tested speed failed but there was a former success. This is a one clock state that sets the Fallback flag to indicate that the sequence has progressed through a successful negotiation to one that fails and now needs to fall back to the highest mutually supported speed.

1.4.3.9.2 Transition SP15:SN_Fallback to SP8:SN_Start

Occurs on the next clock.

1.4.3.10 SP16:SN_IncSpeed state

This is a one clock state that increments the speed for the next attempt. This state is entered when the transmitter has not reached its maximum supported rate during the SAS speed negotiation process. This can either be when the previous window has failed with no previous successes or passed but the highest supported rate hasn't been tested.

1.4.3.10.1 Transition SP16:SN_IncSpeed state to SP8:SN_Start

Occurs on the next clock with the transmitter set for the next highest rate.

1.4.3.11 SP18:PHY_Ready state

1.4.3.11.1 State description

This state is entered from state SP13:SN_Pass state if the Fallback flag is set, which is an indication that the SAS speed negotiation procedure has completed successfully.

This state enables the SAS phy dword synchronization state machine (DWS) to provide rule checking for dword synchronization and determination of link failure. The input from DWS to indicate that Dword Synchronization is Acquired (DWordSync) and is used in the SP:18:Phy Ready and the SP26:SATA_Phy Ready states to determine if it is necessary to re-run OOB. Until this time, the DWS input is ignored.

The state machine remains in this state until a COMINIT sequence is detected or loss of DWordSync.. PhyReady(SAS) is set to one to indicate that the link has been brought up successfully in SAS mode. While in this state, the PhyRdy(SAS) signal to the link layer shall be set to one and dwords from the link layer are transmitted at the negotiated rate

1.4.3.11.2 Transition SP18:PHY_Ready to SP1:SAS_Reset

Transition to SAS_Reset occurs if DWordSync is de-asserted by the DWS state machine or a COMINIT is detected. This is an indication that a device has been unplugged from the port or reset from the link.

1.4.4 SATA host emulation states

Error! Reference source not found. shows the SATA Host emulation states, in which the SAS device has detected that it is connected a SATA target and behaves as if it were a SATA Host as far as OOB is concerned, initiating the speed negotiation. These states are indicated by state names with a prefix of SATA_. During SATA Host emulation, the SAS device transmits a COMWAKE sequence and then waits to receive a COMWAKE. Once the COMWAKE sequence is detected, the SAS device follows the speed negotiation sequence defined in the SATA 1.0 specification.

SAS initiator devices, target devices and expander devices should use similar state machines to the one defined in this specification, with the exception that SAS initiators should implement the SATA Host emulation power management states.

The power management states that are defined in this specification are for SAS initiators that intend to attach directly to SATA targets. For SAS-only initiators, these states are not needed. The power management states are prefixed with PM.

SAS phy layer state machine - SATA host emulation states

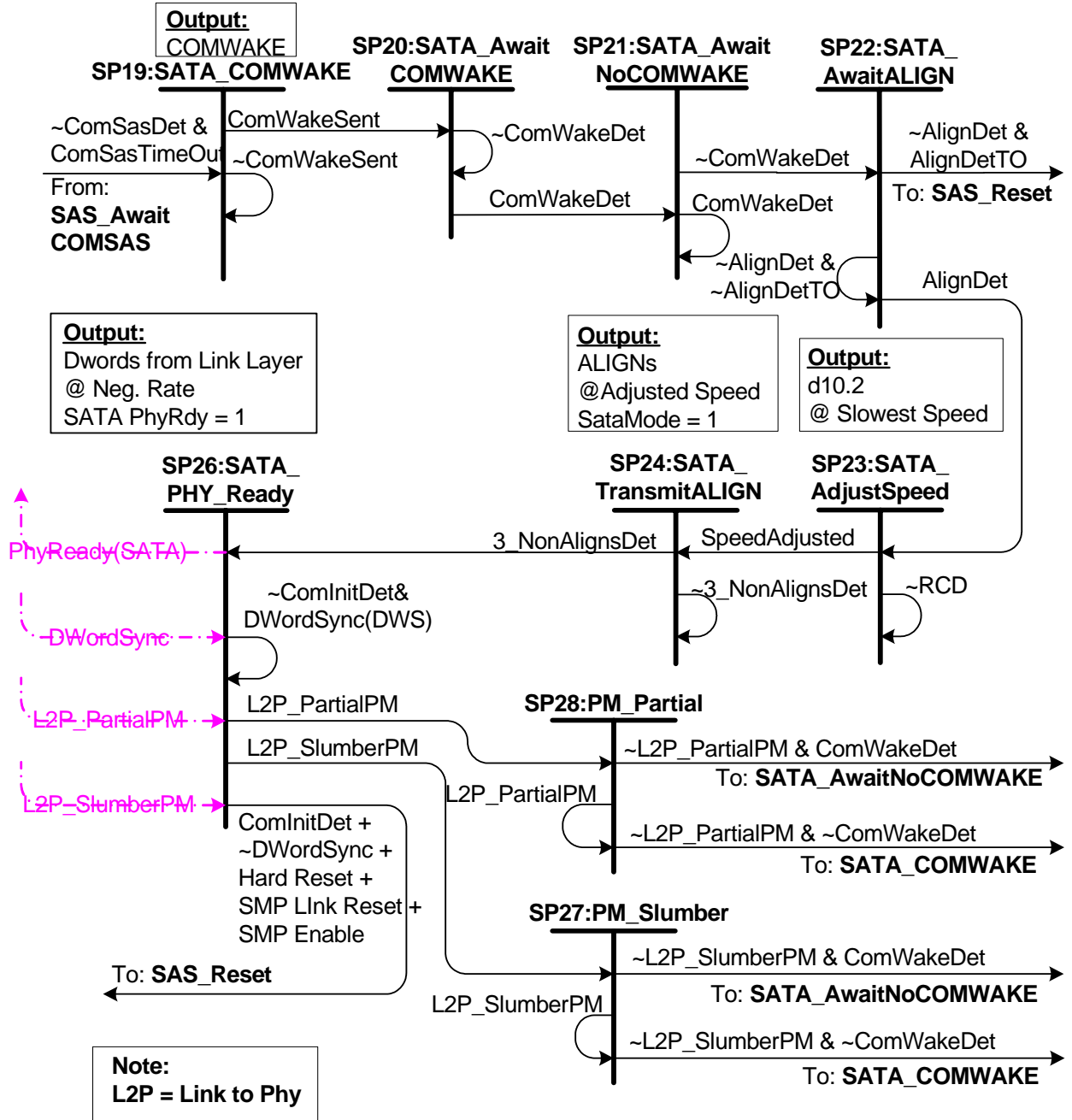


Figure 13. SAS phy state machine - SATA host emulation states

1.4.4.1 SP19:SATA_COMWAKE state

1.4.4.1.1 State description

This is the first state of the SATA Host Emulation states. This state is entered from state SP7:SAS_AwaitCOMSAS when a COMSAS is not detected within a COMSAS detect timeout period of 3,5 μ s. The state machine remains in this state until a COMWAKE sequence has been transmitted.

1.4.4.1.2 Transition SP19:SATA_COMWAKE to SP20:SATA_AwaitCOMWAKE

Occurs when the COMWAKE sequence has been transmitted.

1.4.4.2 SP20:SATA_AwaitCOMWAKE state

1.4.4.2.1 State description

This state is entered when a COMWAKE sequence has been sent during state SP19:SATA_COMWAKE. The state machine remains in this state until a COMWAKE sequence has been detected.

1.4.4.2.2 Transition SP20:SATA_AwaitCOMWAKE to SP21:SATA_AawitNoCOMWAKE

Occurs when a COMWAKE sequence is detected.

1.4.4.3 SP21:SATA_AwaitNoCOMWAKE state

1.4.4.3.1 State description

This state is entered when a COMWAKE sequence is detected in state SATA_AwaitCOMX. The state machine remains in this state until the end of the COMWAKE sequence is detected.

1.4.4.3.2 Transition SP21:SATA_AwaitNoCOMWAKE to SP22:SATA_AwaitALIGN

Occurs when the end of the COMWAKE sequence has been detected.

1.4.4.4 SP22:SATA_AwaitALIGN state

1.4.4.4.1 State description

This state is entered from state SP21:SATA_AwaitNoCOMWAKE after the end of a COMWAKE sequence has been detected. While in this state, the SAS device is transmitting D10.2 characters at its slowest supported transmit speed. The state machine remains in this state until either an ALIGN primitive is detected at any of its supported rates, or 880 μ s (32K Gen1 dwords) have elapsed without detecting any ALIGN primitives.

The SAS device shall start transmitting D10.2 characters no later than 20 Gen1 dwords (533 ns) after COMWAKE was de-asserted.

1.4.4.4.2 Transition SP22:SATA_AwaitALIGN to SP23:SATA_AdjustSpeed

Occurs when the SAS device detects an ALIGN primitive at any of its supported rates.

1.4.4.4.3 Transition SP22:SATA_AwaitALIGN to SP1:SAS_Reset

Occurs when the SAS device does not detect any ALIGN primitives within 880 μ s (32K Gen1 dwords)

1.4.4.5 SP23:SATA_AdjustSpeed state

1.4.4.5.1 State description

This state is entered from state SP22:SATA_AwaitALIGN when the SAS device detects an ALIGN primitive at any of its supported rates. During this state, the SAS device's PHY transmitter circuitry adjusts to the same rate of the ALIGN primitives that were detected by the receiver circuitry. The state machine remains in this state until the transmitter rate has been adjusted appropriately.

1.4.4.5.2 Transition SP23:SATA_AdjustSpeed to SP24:SATA_TransmitALIGN

Occurs when the transmitter rate has been adjusted to the rate of the received ALIGN primitives.

1.4.4.6 SP24:SATA_TransmitALIGN state**1.4.4.6.1 State description**

This state is entered from state SP23:SATA_AdjustSpeed after the transmitter rate has been adjusted to the rate of the received ALIGN primitives. While in this state the SAS device transmits ALIGN primitives at the adjusted rate until the receiver circuitry detects three back-to-back non-ALIGN primitives, at which point the state machine transitions to state SP26:SATA_PHY_Ready.

1.4.4.6.2 Transition SP24:SATA_TransmitALIGN to SP26:SATA_PHY_Ready

Occurs when the SAS device's receiver circuitry detects three back-to-back non-ALIGN primitives signaling the completion of a SATA speed negotiation.

1.4.4.7 SP26:SATA_PHY_Ready state**1.4.4.7.1 State description**

This state is entered from state SP25:SATA_TransmitAlign. This state enables the SAS phy dword synchronization state machine (DWS) to provide rule checking for dword synchronization and determination of link failure. The input from DWS to indicate that Dword Synchronization is Acquired (DWordSync) and is used in the SP:18:Phy Ready and the SP26:SATA_Phy Ready states to determine if it is necessary to re-run OOB. Until this time the DWS input is ignored.

The state machine remains in this state until a COMINIT sequence is detected or loss of DWordSync.. PhyRdy(SATA) is set to one to indicate that the link has been brought up successfully in SATA mode. While in this state, the PhyRdy(SATA) signal to the link layer shall be set to one and dwords from the link layer are transmitted at the negotiated rate

1.4.4.7.2 Transition SP26:SATA_PHY_Ready to SP1:Reset

Occurs when a COMINIT sequence is detected or loss of DWordSync from the DWS state machine indicating that the incoming signal is not compliant with the encoding defined for SAS/SATA.

1.4.4.8 SP27:PM_Partial state**1.4.4.8.1 State description**

This state is entered from SP26:SATA_PHY_Ready if L2P_PartialPM is received from the link layer. Exit from this state is driven from receipt of COMWAKE or de-assertion of the partial signal.

The phy shall remember if COMWAKE was detected during SP27:PM_Partial to determine if the wakeup request was originated from the host or the phy.

1.4.4.8.2 Transition SP27:PM_Partial to SP21:SATA_AwaitNoCOMWAKE

Occurs when the receiver circuitry detects COMWAKE.

1.4.4.8.3 Transition SP27:PM_Partial to SP20:SATA_COMWAKE

The device shall transition to SP20:SATA_COMWAKE if the link layer de-asserts the slumber signal to the PHY interface.

1.4.4.9 SP28:PM_Slumber state**1.4.4.9.1 State description**

This state is entered from SP26:SATA_PHY_Ready since the device is operating in SATA mode and L2P_SlumberPM is received from the link layer. Exit from this state is driven from receipt of COMWAKE or de-assertion of the slumber signal.

The phy shall remember if COMWAKE was detected during SP28:PM_Slumber to determine if the wakeup request was originated from the link layer or the phy.

1.4.4.9.2 Transition SP28:PM_Slumber to SP21:SATA_AwaitNoCOMWAKE

Occurs when the receiver circuitry detects COMWAKE from the device at the other end of the link. This transition should not occur until the phy interface is recovered from Slumber mode and is ready to initiate communications.

1.4.4.9.3 Transition SP28:PM_Slumber to SP20:SATA_COMWAKE

The device shall transition to SP20:SATA_COMWAKE if the link layer de-asserts the slumber signal to the PHY interface.

1.5 Dword SAS phy dword synchronization state machine

1.5.1 Dword SAS phy dword synchronization state machine overview

A SAS receiver monitors and decodes the incoming data stream and forces K28.5 characters into the first byte position to effectively perform dword alignment. The receiver continues to reestablish dword alignment by forcing received K28.5 characters into the first byte position until a valid SAS primitive is detected. The resultant primitives, dwords and valid dword indicators (encoding error indicators, etc) are sent to the Dword synchronization state machine to enable the state machine to determine the dword synchronization policy.

Upon Power Up or loss of previous synchronization, the DWS0:AcquireSync state is entered. On entry into this state, the Enable Disable Link Layer(SAS Disable) or Enable Disable Link Layer(SATA Disable) confirmation is sent to the link layer qualified by the PhyReady parameter from the SP state machine. When the requirements for dword synchronization are met and the DWS state machine enters the DWS3:SyncAcquired state the Enable Disable Link Layer(SATA Enable or SAS Enable) will be output. These signals will be qualified with the phy initialization state machine output PhyReady(SATA or SAS) and presented to the link layer. The link layer uses this qualified version to enable/disable operation. The phy initialization state machine will use the unqualified version of the Enable Disable Link Layer to determine whether to re-run OOB. The unqualified signal will be monitored only in the PhyReady(SATA or SAS) states and not in the other OOB and speed negotiation states. Upon receipt of the Enable Disable Link Layer (SATA Disable) or Enable Disable Link Layer (SAS Disable), the phy initialization state machine will re-run OOB.

Each SAS phy includes a SAS phy dword synchronization state machine (DWS). The DWS state machine establishes the same dword receive/loss policy. The DWS state machine requires an error free string of dwords to obtain synchronization. The DWS state machine tolerates invalid dwords as defined in the state machine before losing dword synchronization. While not in dword synchronization, information received is invalid and not passed to the SAS link layer state machine.

This state machine may be running continually, initially set in DWS0: AcquireSync by a power on event.

The DWS state machine contains the following states:

- nn) DWS0:AcquireSync;
- oo) DWS1:Valid1;
- pp) DWS2:Valid2;
- qq) DWS3:SyncAcquired;
- rr) DWS4:Lost1;
- ss) DWS4aDWS5:Lost1Recovered;
- tt) DWS5DWS6:Lost2;
- uu) DWS4aDWS7:Lost2Recovered;
- vv) DWS6DWS8:Lost3; and
- ww)DWS6aDWS9:Lost3Recovered.

The DWS state machine receives no requests from the link layer.

The DWS state machine sends the following confirmations to the link layer:

- xx) Enable Disable Link Layer(SAS Enable);
- yy) Enable Disable Link Layer(SATA Enable);
- zz) Enable Disable Link Layer(SAS Disable); and
- aaa) Enable Disable Link Layer(SATA Disable).

The DWS state machine receives the following parameters from the SP state machine:

- bbb) PhyReady(SAS); and

Error! Reference source not found. shows the DWS state machine.

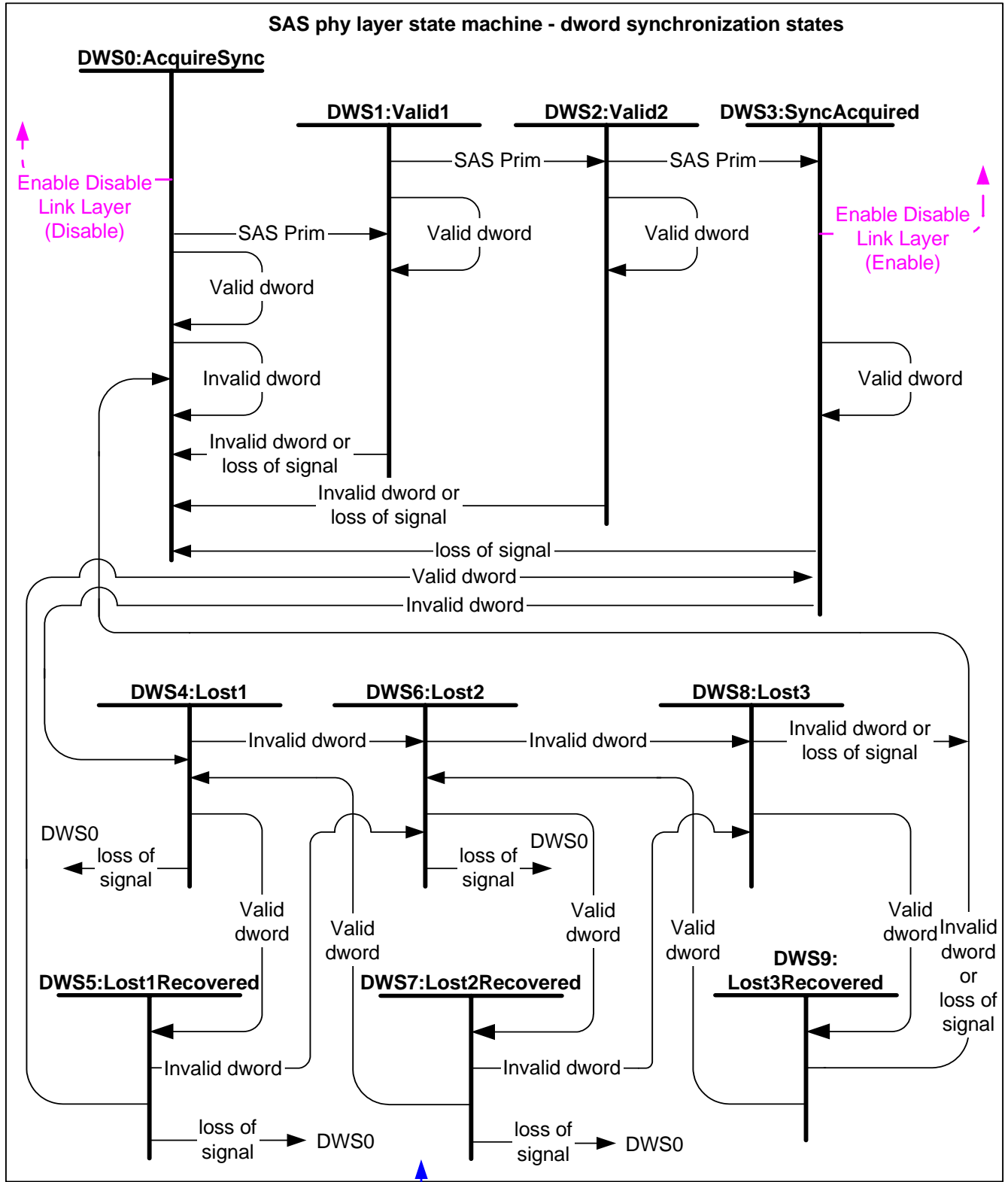


Figure 4. Dword synchronization state machine

1.5.2 DWS0: ACQ Sync StartDWS0:AcquireSync state

1.5.2.1 State description

Upon power-up or loss of previous dword synchronization, the DWS0:AcquireSync state is entered.

1.5.2.2 Transition DWS0:AcquireSync to DWS1:Valid1

This transition shall occur when a valid SAS primitive is detected. This indicates the present dword alignment is valid for this word.

1.5.2.3 Transition DWS0:AcquireSync to DWS0a:AcquireSync

This transition shall occur when a valid dword is detected that is not a SAS primitive.

1.5.2.4 Transition DWS0:AcquireSync to DWS0b:AcquireSync

This transition shall occur when an invalid dword or a loss of signal is detected. The receiver continues to monitor information received and forces the first K28.5 character into the first byte position for word alignment.

1.5.3 DWS1:Valid1 state

1.5.3.1 State description

The receiver monitors information received until a valid SAS primitive, an invalid dword, or a loss of signal is detected.

1.5.3.2 Transition DWS1:Valid1 to DWS2:Valid2

This transition shall occur when a valid SAS primitive is detected. This indicates the present dword alignment is valid for this word.

1.5.3.3 Transition DWS1:Valid1 to DWS1:Valid1

This transition shall occur when a valid dword is detected that is not a SAS primitive.

1.5.3.4 Transition DWS1:Valid1 to DWS0:AcquireSync

This transition shall occur when an invalid dword or a loss of signal is detected. A new attempt at word alignment is started.

1.5.4 DWS2:Valid2 state

1.5.4.1 State description

The receiver has detected Valid2 two valid SAS primitives using the current word alignment. The receiver monitors information received until a valid SAS primitive, an invalid dword, or a loss of signal is detected.

1.5.4.2 Transition DWS2:Valid2 to DWS3:SyncAcquired

This transition shall occur when a valid SAS primitive is detected. This indicates the present dword alignment is valid for this word.

1.5.4.3 Transition DWS2:Valid2 to DWS2:Valid2

This transition shall occur when a valid dword is detected that is not a SAS primitive.

1.5.4.4 Transition DWS2:Valid2 to DWS0:AcquireSync

This transition shall occur when an invalid dword or a loss of signal is detected. A new attempt at word alignment is started.

1.5.5 DWS3:SyncAcquired state

1.5.5.1 State description

Dword synchronization state machine has detected three valid SAS primitives using the current word alignment. The Enable Disable Link Layer(SAS Enable) or Enable Disable Link Layer(SATA Enable) confirmation is sent to the link layer qualified by the PhyReady(SATA/SAS) parameter from the SP state machines. The current SAS primitive is valid for processing. The receiver monitors information received until an invalid dword is received or a loss of signal is signaled by the receiver.

1.5.5.2 Transition DWS3:SyncAcquired to DWS4:Lost1

This transition shall occur when an invalid dword is detected. An expander forwarding the dword to another link shall replace the invalid dword with ERROR or SATA_ERROR (see **Error! Reference source not found.**).

1.5.5.3 Transition DWS3:SyncAcquired to DWS3:SyncAcquired

This transition shall occur when a valid dword is detected.

1.5.5.4 Transition DWS3:SyncAcquired to DWS0:AcquireSync

This transition shall occur when a loss of signal is detected. A new attempt at word alignment is started.

1.5.6 DWS4:Lost1 state

1.5.6.1 State description

The receiver has detected an invalid dword. This is the first state in the process that may result in losing dword synchronization. The receiver monitors information received.

1.5.6.2 Transition DWS4:Lost1 to DWS5:Lost1Recovered

This transition shall occur when a valid dword is detected. This indicates the present dword alignment may still be valid for this dword.

1.5.6.3 Transition DWS4:Lost1 to DWS6:Lost2

This transition shall occur when an invalid dword is detected. An expander forwarding the dword to another link shall replace the invalid dword with ERROR or SATA_ERROR (see **Error! Reference source not found.**).

1.5.6.4 Transition DWS4:Lost1 to DWS0:AcquireSync

This transition shall occur when a loss of signal is detected. A new attempt at dword alignment is started.

1.5.7 DWS5:Lost1Recovered state

1.5.7.1 State description

The receiver has detected a valid dword after an invalid dword. Dword synchronization may still be valid. The receiver monitors information received for valid dwords, invalid dwords and loss of signal.

1.5.7.2 Transition DWS5:Lost1Recovered to DWS3:SyncAcquired

This transition shall occur when a valid dword is detected. Two valid dwords have been received since the last invalid dword.

1.5.7.3 Transition DWS5:Lost1Recovered to DWS6:Lost2

This transition shall occur when an invalid dword is detected. An expander forwarding the dword to another link shall replace the invalid dword with ERROR or SATA_ERROR (see **Error! Reference source not found.**).

1.5.7.4 Transition DWS5:Lost1Recovered to DWS0:AcquireSync

This transition shall occur when a loss of signal is detected. A new attempt at dword alignment is started.

1.5.8 DWS6:Lost2 state

1.5.8.1 State description

The receiver has detected an invalid dword without sufficient valid dwords to nullify a previous invalid dwords. The receiver monitors information received for valid dwords, invalid dwords or and loss of signal.

1.5.8.2 Transition DWS6:Lost2 to DWS7:Lost2Recovered

This transition shall occur when a valid dword is detected. This indicates the present dword alignment is still valid for this dword.

1.5.8.3 Transition DWS6:Lost2 to DWS8:Lost3

This transition shall occur when an invalid dword is detected. An expander forwarding the dword to another link shall replace the invalid dword with ERROR or SATA_ERROR (see **Error! Reference source not found.**).

1.5.8.4 Transition DWS6:Lost2 to DWS0:AcquireSync

This transition shall occur when a loss of signal is detected. A new attempt at dword alignment is started.

1.5.9 DWS7:Lost2Recovered state

1.5.9.1 State description

The receiver has detected a valid dword after an invalid dword. Dword synchronization may still be valid. The receiver monitors information received for valid dwords, invalid dwords and a loss of signal is detected.

1.5.9.2 Transition DWS7:Lost2Recovered to DWS4:Lost1

This transition shall occur when a valid dword is detected. Two valid dwords have been received since the last invalid dword.

1.5.9.3 Transition DWS7:Lost2Recovered to DWS8:Lost3

This transition shall occur when an invalid dword is detected. An expander forwarding the dword to another link shall replace the invalid dword with ERROR or SATA_ERROR (see **Error! Reference source not found.**).

1.5.9.4 Transition DWS7:Lost2Recovered to DWS0:AcquireSync

This transition shall occur when a loss of signal is detected. A new attempt at dword alignment is started.

1.5.10 DWS8:Lost3 state

1.5.10.1 State description

The receiver has detected an invalid dword without sufficient valid dwords to nullify a previous invalid dwords. The receiver monitors information received for valid dwords, invalid dwords and a loss of signal.

1.5.10.2 Transition DWS8:Lost3 to DWS9:Lost3Recovered

This transition shall occur when a valid dword is detected. This indicates the present dword alignment may still valid for this dword.

1.5.10.3 Transition DWS8:Lost3 to DWS0:AcquireSync

This transition shall occur when an invalid dword or loss of signal is detected. If an invalid dword is detected, at least four invalid dwords have been received without sufficient intervening valid dwords to nullify their affect. Dword synchronization is lost and must be reacquired; a new attempt at dword alignment is started.

1.5.11 DWS9:Lost3Recovered state

1.5.11.1 State description

The receiver has detected a valid dword after an invalid dword. Dword synchronization may still be valid. The receiver monitors information received for valid dwords, invalid dwords and loss of signal.

1.5.11.2 Transition DWS9:Lost3Recovered to DWS6:Lost2

This transition shall occur when a valid dword is detected. Two valid dwords have been received since the last invalid dword.

1.5.11.3 Transition DWS9:Lost3Recovered to DWS0:AcquireSync

This transition shall occur when an invalid dword or loss of signal is detected. If an invalid dword is detected, at least four invalid dwords have been received without sufficient intervening valid dwords to nullify their affect. Dword synchronization is lost and must be reacquired; a new attempt at dword alignment is started.