To: T10 Technical Committee
From: Amr Wassal, PMC-Sierra (amr_wassal@pmc-sierra.com)
Stephen Finch, STMicroelectronics (steve.finch@st.com)
Date: 16 November 2006
Subject: 06-515r0 (formerly 06-324) SAS-2 Modifications to the SAS Speed Negotiation

Revision history

Revision 0 (7 July 2006) First revision
Revision 1 (24 August 2006)
Revision 2 (7 September 2006) Review updates including: training pattern definition, changes to several diagrams where training pattern is depicted and state machine.
Revision 3 (20 September 2006) Updates based on 12 September 2006 Phy Working Group meeting. A significant re-organization, plus many changes to make consistent use of new terms like Final-SNW and Train-SNW. Updates to diagrams.
Revision 4 (29 September 2006) Updates per 28 September 2006 conference call.
Revision 5 (12 October 2006) Updates per 12 October 2006 conference call.
Revision 6 (19 October 2006) Updates per 19 October 2006 conference call. We reviewed down to the middle of section 6.7.4.2.4.
Revision 7 (30 October 2006) Updates per 26 October 2006 conference call. We reviewed down to the middle of section 6.7.4.2.4. Incorporated that portion of 06-363 relating to changes in clause 6 into this document and that content will be reviewed and updated as part of this document into the future.
Revision 8 (2 November 2006) Updates per 2 November 2006 conference call.
Revision 9 (8 November 2006) Updates per 7 November 2006 face-to-face working group meeting.
Revision 0 06-515 (16 November 2006) Editorial updates per 16 November 2006 conference call. No technical changes.

Related documents

sas2r04a - Serial Attached SCSI - 2 (SAS-2) revision 4a
05-397 SAS-2 Start-up training sequence (Harvey Newman, Infineon)
06-263 SAS-2 Spread-spectrum clocking (Rob Elliott, HP)
06-295 SAS-2 Speed negotiation (Amr Wassal and Robert Watson, PMC-Sierra)
06-354 SAS-2 Startup training sequence proposal (Steve Finch, ST Microelectronics)
06-355 SAS-2 SNW-3 definition (Amr Wassal and Robert Watson, PMC-Sierra)
06-363 SAS-2 SNW-3 bit definitions (Rob Elliot, HP)

Overview

The SAS speed negotiation defined in SAS-1 did not allow for training of the phy nor allow for exchange of parameters that are proving useful at G3 and higher rates. New proposals for SAS-2 proposed a training pattern in 05-397 and speed negotiation modifications in 06-295 that has provision for training and for exchanging parameters during SNW-3 as defined in 06-363.

Suggested additions to clause 3 Definitions, symbols, abbreviations, keywords, and conventions are in blue text. No changes to existing clause 3 content.

3.1 Definitions

3.1.x Actual Lock Time (ALT): The time at which actual lock to the received ALIGN (0) or ALIGN (1) occurs and the transmitter transitions form sending ALIGN (0) to sending ALIGN (1) (see 6.7.4.2.2).

3.1.x Actual Training Time (ATT): The time in which training of the phy's receiver is complete (see 6.7.4.2.2).

3.1.x Bit Cell Time (BCCT): The time period defined within an SNW-3 during which one bit phy capabilities is transmitted (see 6.7.4.2.2).

3.1.x Maximum Training Time (MTT): The maximum time for training to complete during a Train-SNW (see 6.7.4.2.2).

3.1.x Maximum Train-SNW Window Time (MTWT): The maximum duration of the Train-SNW (see 6.7.4.2.2).

3.1.x Training Lock Time (TLT): The maximum time during Train-SNW for a receiver device to complete training and a transmitter device to reply with TRAIN_DONE pattern.

3.1.x Train-SNW Window Time (TWT): The actual duration of the Train-SNW.

3.2 Symbols and abbreviations

ALT	Actual lock time
ATT	Actual training time
MTT	Maximum training time
MTWT	Maximum Train-SNW window time
SNW	Speed negotiation window
SNWT	Speed negotiation window time
TLT	Training lock time
TWT	Train-SNW window time

Suggested changes to clause 7 (Link Layer): Add definitions for TRAIN and TRAIN_DONE primitives. All changes are additions of blue text. No changes to existing clause 7 content.

7.2.2 Primitive summary

Table 84 defines the primitives not specific to the type of connection.

Table 84 — Primitives not specific to type of connection (part 1 of 2)

Primitive	Use[a]	From[b]			To[b]			Primitive sequence type[c]
		I	E	T	I	E	T	
AIP (NORMAL)	NoConn		E		I	E	T	Extended
AIP (RESERVED 0)								
AIP (RESERVED 1)								
AIP (RESERVED 2)								
AIP (RESERVED WAITING ON PARTIAL)								
AIP (WAITING ON CONNECTION)			E					
AIP (WAITING ON DEVICE)			E					
AIP (WAITING ON PARTIAL)			E					
ALIGN (0)	All	I	E	T	I	E	T	Single
ALIGN (1)								
ALIGN (2)								
ALIGN (3)								
BREAK	All	I	E	T	I	E	T	Redundant
BREAK_REPLY	All	I	E	T	I	E	T	Redundant
BROADCAST (CHANGE)	NoConn	I	E		I			Redundant
BROADCAST (SES)				T	I			
BROADCAST (EXPANDER)			E		I			
BROADCAST (ASYNCHRONOUS EVENT)				T	I			
BROADCAST (ZONE ACTIVATE)		I				E		
BROADCAST (RESERVED 4)								
BROADCAST (RESERVED CHANGE 0)					I			
BROADCAST (RESERVED CHANGE 1)					I			
CLOSE (CLEAR AFFILIATION)	STP	I					T	Triple
CLOSE (NORMAL)	Conn	I		T	I		T	
CLOSE (RESERVED 0)								
CLOSE (RESERVED 1)								

Primitive	Use[a]	From[b]			To[b]			Primitive sequence type[c]
		I	E	T	I	E	T	
EOAF	NoConn	I	E	T	I	E	T	Single
ERROR	All		E		I	E	T	Single
HARD_RESET	NoConn	I	E		I	E	T	Redundant
NOTIFY (ENABLE SPINUP)	All	I	E				T	Single
NOTIFY (POWER LOSS EXPECTED)		I	E				T	
NOTIFY (RESERVED 1)					I	E	T	
NOTIFY (RESERVED 2)								
OPEN_ACCEPT	NoConn	I		T	I		T	Single

Table 84 — Primitives not specific to type of connection (part 2 of 2)

Primitive	Use[a]	From[b]			To[b]			Primitive sequence type[c]
		I	E	T	I	E	T	
OPEN_REJECT (BAD DESTINATION)	NoConn		E		I		T	Single
OPEN_REJECT (CONNECTION RATE NOT SUPPORTED)		I	E	T				
OPEN_REJECT (NO DESTINATION)			E					
OPEN_REJECT (PATHWAY BLOCKED)			E					
OPEN_REJECT (PROTOCOL NOT SUPPORTED)		I		T				
OPEN_REJECT (ZONE VIOLATION)			E					
OPEN_REJECT (RESERVED ABANDON 1)								
OPEN_REJECT (RESERVED ABANDON 2)								
OPEN_REJECT (RESERVED ABANDON 3)								
OPEN_REJECT (RESERVED CONTINUE 0)								
OPEN_REJECT (RESERVED CONTINUE 1)								
OPEN_REJECT (RESERVED INITIALIZE 0)								
OPEN_REJECT (RESERVED INITIALIZE 1)								
OPEN_REJECT (RESERVED STOP 0)								
OPEN_REJECT (RESERVED STOP 1)								
OPEN_REJECT (RETRY)		I	E	T				
OPEN_REJECT (STP RESOURCES BUSY)			E	T				
OPEN_REJECT (WRONG DESTINATION)		I		T			T	
SOAF	NoConn	I	E	T	I	E	T	Single
TRAIN	SpNeg	I	E	T	I	E	T	Redundant
TRAIN_DONE	SpNeg	I	E	T	I	E	T	Redundant

[a] The Use column indicates when the primitive is used:

a) NoConn: SAS physical links, outside connections;
b) Conn: SAS physical links, inside connections;
c) All: SAS physical links, both outside connections or inside any type of connection; or
d) STP: SAS physical links, inside STP connections.
e) SpNeg: SAS physical links, during speed negotiation (see 6.7.4.2)

[b] The From and To columns indicate the type of ports that originate each primitive or are the intended destinations of each primitive:

a) I for SAS initiator ports;
b) E for expander ports; and
c) T for SAS target ports.

Expander ports are not considered originators of primitives that are passing through from expander port to expander port.

[c] The Primitive sequence type columns indicate whether the primitive is sent as a single primitive sequence, a repeated primitive sequence, a continued primitive sequence, an extended primitive sequence, a triple primitive sequence, or a redundant primitive sequence (see 7.2.4).

7.2.3 Primitive encodings

Table 87 defines the primitive encoding for primitives not specific to type of connection.

Table 87 — Primitive encoding for primitives not specific to type of connection (part 1 of 2)

Primitive	Character			
	1st	2nd	3rd	4th (last)
AIP (NORMAL)	K28.5	D27.4	D27.4	D27.4
AIP (RESERVED 0)	K28.5	D27.4	D31.4	D16.7
AIP (RESERVED 1)	K28.5	D27.4	D16.7	D30.0
AIP (RESERVED 2)	K28.5	D27.4	D29.7	D01.4
AIP (RESERVED WAITING ON PARTIAL)	K28.5	D27.4	D01.4	D07.3
AIP (WAITING ON CONNECTION)	K28.5	D27.4	D07.3	D24.0
AIP (WAITING ON DEVICE)	K28.5	D27.4	D30.0	D29.7
AIP (WAITING ON PARTIAL)	K28.5	D27.4	D24.0	D04.7
ALIGN (0)	K28.5	D10.2	D10.2	D27.3
ALIGN (1)	K28.5	D07.0	D07.0	D07.0
ALIGN (2)	K28.5	D01.3	D01.3	D01.3
ALIGN (3)	K28.5	D27.3	D27.3	D27.3
BREAK	K28.5	D02.0	D24.0	D07.3
BREAK_REPLY	K28.5	D02.0	D29.7	D16.7
BROADCAST (CHANGE)	K28.5	D04.7	D02.0	D01.4
BROADCAST (SES)	K28.5	D04.7	D07.3	D29.7
BROADCAST (EXPANDER)	K28.5	D04.7	D01.4	D24.0
BROADCAST (ASYNCHRONOUS EVENT)	K28.5	D04.7	D04.7	D04.7
BROADCAST (ZONE ACTIVATE)	K28.5	D04.7	D16.7	D02.0
BROADCAST (RESERVED 4)	K28.5	D04.7	D29.7	D30.0
BROADCAST (RESERVED CHANGE 0)	K28.5	D04.7	D24.0	D31.4
BROADCAST (RESERVED CHANGE 1)	K28.5	D04.7	D27.4	D07.3
CLOSE (CLEAR AFFILIATION)	K28.5	D02.0	D07.3	D04.7
CLOSE (NORMAL)	K28.5	D02.0	D30.0	D27.4
CLOSE (RESERVED 0)	K28.5	D02.0	D31.4	D30.0
CLOSE (RESERVED 1)	K28.5	D02.0	D04.7	D01.4
EOAF	K28.5	D24.0	D07.3	D31.4
ERROR	K28.5	D02.0	D01.4	D29.7
HARD_RESET	K28.5	D02.0	D02.0	D02.0

Table 87 — Primitive encoding for primitives not specific to type of connection (part 2 of 2)

Primitive	Character			
	1st	2nd	3rd	4th (last)
NOTIFY (ENABLE SPINUP)	K28.5	D31.3	D31.3	D31.3
NOTIFY (POWER LOSS EXPECTED)	K28.5	D31.3	D07.0	D01.3
NOTIFY (RESERVED 1)	K28.5	D31.3	D01.3	D07.0
NOTIFY (RESERVED 2)	K28.5	D31.3	D10.2	D10.2
OPEN_ACCEPT	K28.5	D16.7	D16.7	D16.7
OPEN_REJECT (BAD DESTINATION)	K28.5	D31.4	D31.4	D31.4
OPEN_REJECT (CONNECTION RATE NOT SUPPORTED)	K28.5	D31.4	D04.7	D29.7
OPEN_REJECT (NO DESTINATION)	K28.5	D29.7	D29.7	D29.7
OPEN_REJECT (PATHWAY BLOCKED)	K28.5	D29.7	D16.7	D04.7
OPEN_REJECT (PROTOCOL NOT SUPPORTED)	K28.5	D31.4	D29.7	D07.3
OPEN_REJECT (ZONE VIOLATION)	K28.5	D31.4	D02.0	D27.4
OPEN_REJECT (RESERVED ABANDON 1)	K28.5	D31.4	D30.0	D16.7
OPEN_REJECT (RESERVED ABANDON 2)	K28.5	D31.4	D07.3	D02.0
OPEN_REJECT (RESERVED ABANDON 3)	K28.5	D31.4	D01.4	D30.0
OPEN_REJECT (RESERVED CONTINUE 0)	K28.5	D29.7	D02.0	D30.0
OPEN_REJECT (RESERVED CONTINUE 1)	K28.5	D29.7	D24.0	D01.4
OPEN_REJECT (RESERVED INITIALIZE 0)	K28.5	D29.7	D30.0	D31.4
OPEN_REJECT (RESERVED INITIALIZE 1)	K28.5	D29.7	D07.3	D16.7
OPEN_REJECT (RESERVED STOP 0)	K28.5	D29.7	D31.4	D07.3
OPEN_REJECT (RESERVED STOP 1)	K28.5	D29.7	D04.7	D27.4
OPEN_REJECT (RETRY)	K28.5	D29.7	D27.4	D24.0
OPEN_REJECT (STP RESOURCES BUSY)	K28.5	D31.4	D27.4	D01.4
OPEN_REJECT (WRONG DESTINATION)	K28.5	D31.4	D16.7	D24.0
SOAF	K28.5	D24.0	D30.0	D01.4
TRAIN	K28.5	D30.3	D30.3	D30.3
TRAIN_DONE	K28.5	D30.3	D30.3	D10.2

The following are added clauses:

7.2.5.14 TRAIN

TRAIN is used during speed negotiation (see 6.7.4.2).

7.2.5.15 TRAIN_DONE

TRAIN_DONE is used during speed negotiation (see 6.7.4.2).

Suggested changes to clause 6.6.2. No other changes to clause 6.6. All changes are additions of blue text.

6.6.2 Transmitting OOB signals

Table 73 — OOB signal transmitter device requirements

Signal	Burst time	Idle time	Negation time	Signal time [g]
COMWAKE	160 OOBI [a]	160 OOBI [a]	280 OOBI [b]	2200 OOBI [h]
COMINIT/RESET	160 OOBI [a]	480 OOBI [c]	800 OOBI [d]	4600 OOBI [i]
COMSAS	160 OOBI [a]	1 440 OOBI [e]	2 400 OOBI [f]	12000 OOBI [j]

[a] 160 OOBI is nominally 106,6 ns (see table 72).
[b] 280 OOBI is nominally 186,6 ns.
[c] 480 OOBI is nominally 320 ns.
[d] 800 OOBI is nominally 533 ns.
[e] 1 440 OOBI is nominally 960 ns.
[f] 2 400 OOBI is nominally 1 600 ns.
[g] Signal time is defined as the total of six Burst times plus six Idle times plus one Negation time (see description below).
[h] 2 200 OOBI is nominally 1 467 ns.
[i] 4 600 OOBI is nominally 3 093 ns.
[j] 12 000 OOBI is nominally 8 000 ns.

Clause 6.7.4.2 is replaced in its entirety. Previously reviewed text is shown in black rather that being changed to blue.

6.7.4.2 SAS speed negotiation sequence

6.7.4.2.1 SAS speed negotiation sequence overview

The speed negotiation sequence establishes communications between the two phys of a physical link at the highest possible transmission rate.

The SAS speed negotiation sequence is a peer-to-peer negotiation technique that does not assume initiator and target (i.e., host and device) roles. The rules for speed negotiation are the same for both phys of a physical connection.

The speed negotiation sequence consists of a set of speed negotiation windows. Each speed negotiation window is identified by a name (e.g., Speed Negotiation Window 1 or SNW-1) that is used to identify the format, the timing of any transmissions within the window, and the transmission rate.

Speed negotiation windows conform to one of three defined formats:

a) Speed negotiation without training. SNW-1, SNW-2 and Final-SNW are of this type.
b) Speed negotiation with training. Train-SNW is of this type.
c) Link Capabilities Exchange. SNW-3 is of this type.

The speed negotiation window formats are defined in 6.7.4.2.3.

Many of the timing parameters used for defining the speed negotiation windows are common to multiple window types. All of the timing specifications for all speed negotiation window types are defined in section 6.7.4.2.2.

A SAS speed negotiation sequence may or may not include all three types of speed negotiation windows. Phys may implement a subset of speed negotiation windows provided that the subset implements a valid speed negotiation sequence. Speed negotiation sequences are defined in section 6.7.4.2.4.

6.7.4.2.2 Speed Negotiation Timing Specifications

Table 80 defines the timing specifications for the SAS speed negotiation windows. The definitions of the speed negotiation window types specify the timing parameters that are applicable for that type.

Table 80 — SAS speed negotiation sequence timing specifications

Acronym	Name	Time[a]	Comments
RCDT	Rate change delay time	750 000 OOBI[b]	The time the transmitter device shall transmit D.C. idle between rates during speed negotiation.
SNTT	Speed negotiation transmit time	163 840 OOBI[c]	The time during which ALIGN (0) or ALIGN (1) is transmitted at each physical link rate during the speed negotiation sequence.
SNLT	Speed negotiation lock time	153 600 OOBI[d]	The maximum time during SNW-1, SNW-2 or Final-SNW for a transmitter device to reply with ALIGN (1).
ALT	Actual lock time		The time at which actual lock to the received ALIGN (0) or ALIGN (1) occurs and the transmitter transitions form sending ALIGN (0) to sending ALIGN (1).
SNWT	Speed negotiation window time	913 840 OOBI[e]	The duration of SNW-1, SNW-2, SNW-3, or a Final-SNW.
MTT	Maximum training time	29 998 080 OOBI[f]	The maximum time for training to complete during a Train-SNW.
TLT	Training lock time	28 497 920 OOBI[g]	The maximum time during Train-SNW for a transmitter device to reply with TRAIN_DONE pattern.
ATT	Actual training time		The time in which training of the phy's receiver is complete.
TWT	Train-SNW window time		The actual duration of the Train-SNW . The duration of the Train-SNW may be less than MTWT if the Train-SNW is valid, or will be equal to MTWT if the Train-SNW is invalid.
MTWT	Maximum Train-SNW window time	30 748 080 OOBI[h]	The maximum duration of the Train-SNW . The duration of the Train-SNW may be less than MTWT.
BCT	Bit Cell Time	2 200 OOBI	Each BCT is either a COMWAKE or D.C. Idle.

[a] OOBI is defined in table 67.

[b] 750 000 OOBI is nominally 500 us. Derived from: OOBI x 18 750 x 40.

[c] 163 840 OOBI is nominally 109,226 µs. Derived from: OOBI x 4 096 x 40.

[d] 153 600 OOBI is nominally 102,4 µs. Derived from: OOBI x 3 840 x 40.

[e] 913 840 OOBI is nominally 609,226 µs. Derived from: RCDT + SNTT.

[f] 29 998 080 OOBI is nominally 19,99872 ms (~ 20 ms, an integer number of training patterns)

[g] 28 497 920 OOBI is nominally 18,998613 ms (~ 19 ms, an integer number of training patterns)

[h] 30 748 080 OOBI is nominally 20,5 ms. Derived from: RCDT + MTT.

6.7.4.2.3 Speed Negotiation Window Definitions

During each speed negotiation window, a phy shall either:

a) transmit and receive as defined for the current speed negotiation window; or
b) transmit D.C. Idle and not attempt to receive.

If a phy transmits as defined for the current speed negotiation window and receives the expected transmission, then the speed negotiation window is valid.

If a phy does not receive the expected transmission from the attached phy, then the speed negotiation window is invalid.

> Note x: If a phy transmits D.C. Idle during a speed negotiation window, then the attached phy may not receive the expected transmission and the speed negotiation window is invalid.

6.7.4.2.3.1 SNW-1, SNW-2 and Final-SNW Definition

Figure 122 defines the SAS speed negotiation window format for SNW-1, SNW-2 and Final-SNW, including:

a) speed negotiation window time (SNWT);
b) rate change delay time (RCDT);
c) speed negotiation transmit time (SNTT);
d) speed negotiation lock time (SNLT); and
e) actual lock time (ALT).

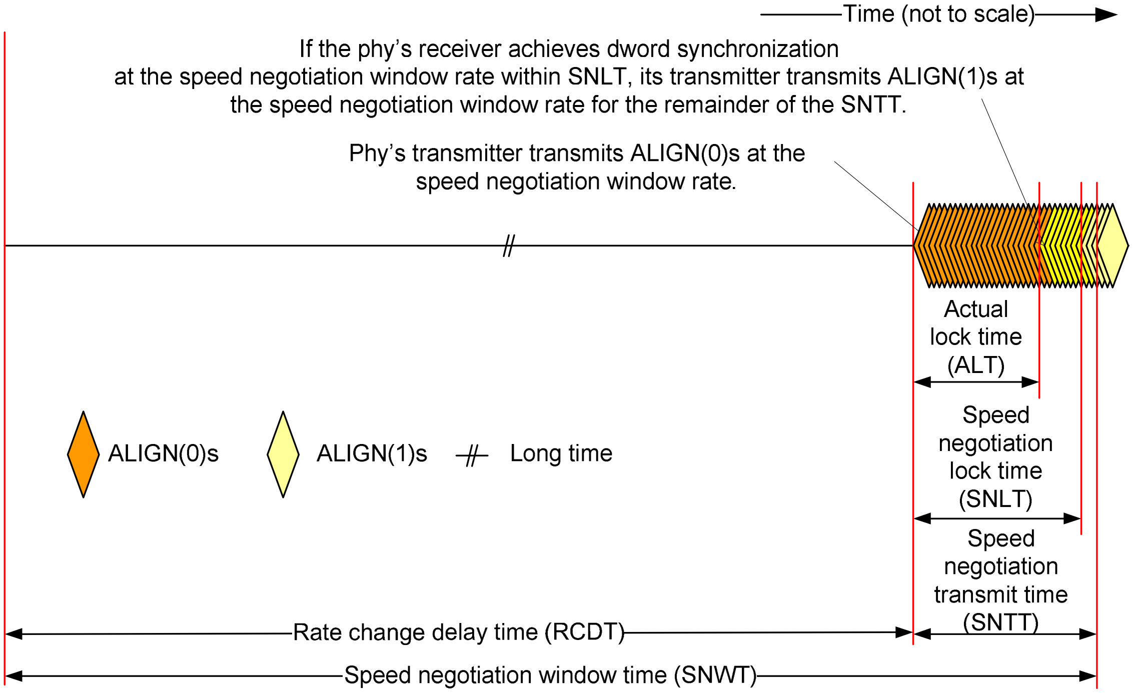

Time (not to scale)
If the phy's receiver achieves dword synchronization at the speed negotiation window rate within SNLT, its transmitter transmits ALIGN(1)s at the speed negotiation window rate for the remainder of the SNTT.
Phy's transmitter transmits ALIGN(0)s at the speed negotiation window rate.
Actual lock time (ALT)
Speed negotiation lock time (SNLT)
Speed negotiation transmit time (SNTT)
ALIGN(0)s
ALIGN(1)s
Long time
Rate change delay time (RCDT)
Speed negotiation window time (SNWT)

Figure 122 — SAS SNW-1, SNW-2 and Final-SNW

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

1) transmission of D.C. idle for an RCDT; and
2) if the phy supports the physical link rate or if this is the Final-SNW, transmission of ALIGNs at the physical link rate specified below for the remainder of the entire speed negotiation window time. If the phy does not support the physical link rate, transmission of D.C. idle for the remainder of the entire speed negotiation window time.

The speed negotiation window rate for SNW-1 shall be 1,5 Gbps.

The speed negotiation window rate for SNW-2 shall be 3,0 Gbps.

The speed negotiation window rate for Final-SNW shall be:

a) 1,5 Gbps if SNW-1 is valid and SNW-2 is invalid; or

b) 3,0 Gbps if SNW-2 is valid and SNW-3 (see 6.7.4.2.3.2) is invalid.

If the phy supports the speed negotiation window rate, it shall initially transmit ALIGN (0) and it shall attempt to synchronize on an incoming series of dwords at that rate for the SNLT. The received dwords may be ALIGN (0) or ALIGN (1) primitives. If the phy achieves dword synchronization within the SNLT, it shall change from transmitting ALIGN (0) primitives to transmitting ALIGN (1) primitives for the remainder of the SNTT (i.e., the remainder of the speed negotiation window time). The point at which the phy achieves dword synchronization is the Actual lock time (ALT). The change from transmitting ALIGN (0) primitives to transmitting ALIGN (1) primitives shall occur after ALT. If the phy does not achieve dword synchronization within the SNLT, it shall continue transmitting ALIGN (0) primitives for the remainder of the SNTT (i.e., the remainder of the speed negotiation window time).

At the end of the SNTT, if a phy is both transmitting and receiving ALIGN (1) primitives, it shall consider the speed negotiation window valid.

SNW-1, SNW-2 and Final-SNW shall be transmitted with spread spectrum clocking disabled.

6.7.4.2.3.2 SNW-3 Definition

SNW-3 allows the phys to exchange information about their capabilities (e.g., physical link rates supported information, SSC support information, and multiplexing support).

If a phy supports SNW-3, then the phy transmits 32 bits of information describing the capabilities of the phy.

If a phy supports SNW-3, then the phy receives 32 bits of information from the attached phy or D.C. Idle.

If a phy does not support SNW-3, then the phy transmits D.C. Idle.

If a phy does not support SNW-3, then the phy ignores received SNW-3 signaling.

The first bit of information is the start bit and shall be transmitted as a one. Each of the remaining 31 bits of information may be a one or a zero. The transmitter shall transmit 32 BCTs (see Table 80) after an RCDT and then transmit D.C. Idle for the remainder of SNTT.

Each BCT consists of the transmission of a COMWAKE signal (see 6.6.2) or the transmission of D.C. Idle. To transmit a one, a COMWAKE shall be transmitted in the BCT. To transmit a zero, a D.C. Idle shall be transmitted in the BCT.

Table 81 defines the information bits.

Figure 123 defines the SAS SNW-3, including:

a) speed negotiation window time;
b) speed negotiation window rate of 1,5 Gbps;
c) rate change delay time (RCDT); and
d) speed negotiation transmit time (SNTT).

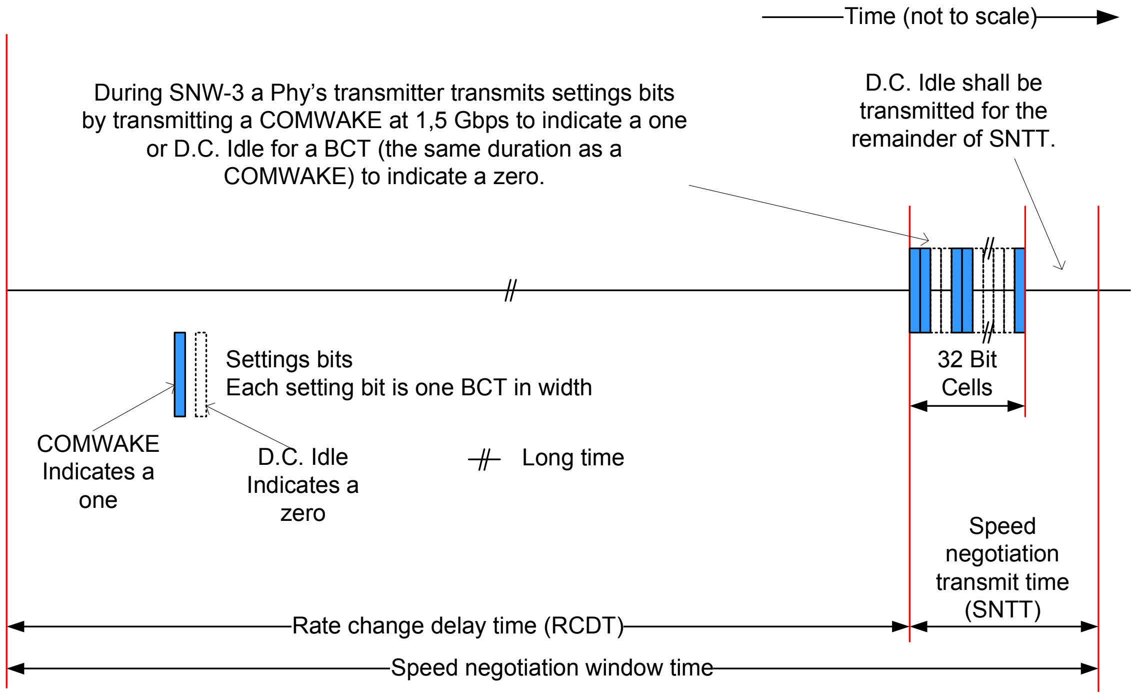

Time (not to scale)
During SNW-3 a Phy's transmitter transmits settings bits by transmitting a COMWAKE at 1,5 Gbps to indicate a one or D.C. Idle for a BCT (the same duration as a COMWAKE) to indicate a zero.
D.C. Idle shall be transmitted for the remainder of SNTT.
Settings bits
Each setting bit is one BCT in width
32 Bit Cells
COMWAKE Indicates a one
D.C. Idle Indicates a zero
Long time
Speed negotiation transmit time (SNTT)
Rate change delay time (RCDT)
Speed negotiation window time

Figure 123 — SAS SNW-3

Table 81 defines the SNW-3 information. For bits defined as reserved, the phy shall transmit zeros (i.e., D.C. Idle) in the outgoing SNW-3 information and shall ignore the bits in the incoming SNW-3 information.

Table 81 — SNW-3 settings bits

Bit(s)	Description	Changeable
Header		
(first bit) 0	START bit	No
1	TX SSC TYPE bit	No
2 to 7	Reserved [note: 05-381 Multiplexing claims bits 4:7]	No
Supported Settings		
8	G1 WITHOUT SSC SUPPORTED bit	Yes [a]
9	G1 WITH SSC SUPPORTED bit	Yes [a]
10	G2 WITHOUT SSC SUPPORTED bit	Yes [a]
11	G2 WITH SSC SUPPORTED bit	Yes [a]
12	G3 WITHOUT SSC SUPPORTED bit	Yes [a]
13	G3 WITH SSC SUPPORTED bit	Yes [a]
14 to 30	Reserved	No
Trailer		
31 (last bit)	PARITY bit	No
[a] Phys controlled by a management device server with the SMP PHY CONTROL function (see 10.4.3.18) may set this bit to one or zero in the SNW-3 INFORMATION CHANGEABLE field.		

The START bit shall be set to one. The phy's receiver shall use this bit to establish the timing for the subsequent bits.

A TX SSC TYPE bit set to one indicates that the phy's transmitter uses center-spreading SSC when SSC is enabled. A TX SSC TYPE bit set to zero indicates that the phy's transmitter uses down-spreading SSC when SSC is enabled, or that the phy does not support SSC.

> NOTE 1 - The phy receiver may use the TX SSC TYPE bit to optimize its CDR circuitry. This bit indicates the type of SSC used when attached to a SAS phy or an expander phy.

A G1 WITHOUT SSC SUPPORTED bit set to one indicates that the phy supports G1 without SSC. A G1 WITHOUT SSC SUPPORTED bit set to zero indicates that the phy does not support G1 without SSC. If the phy supports SNW-1 and supports SNW-3, then G1 WITHOUT SSC SUPPORTED bit shall be set to one.

A G1 WITH SSC SUPPORTED bit set to one indicates that the phy supports G1 (i.e., 1,5 Gbps) with SSC. A G1 WITH SSC SUPPORTED bit set to zero indicates that the phy does not support G1 with SSC.

A G2 WITHOUT SSC SUPPORTED bit set to one indicates that the phy supports G2 without SSC. A G2 WITHOUT SSC SUPPORTED bit set to zero indicates that the phy does not support G2 without SSC. If the phy supports SNW-2 and supports SNW-3, then G2 WITHOUT SSC SUPPORTED bit shall be set to one.

A G2 WITH SSC SUPPORTED bit set to one indicates that the phy supports G2 (i.e., 3 Gbps) with SSC. A G2 WITH SSC SUPPORTED bit set to zero indicates that the phy does not support G2 with SSC.

A G3 WITHOUT SSC SUPPORTED bit set to one indicates that the phy supports G3 without SSC. A G3 WITHOUT SSC SUPPORTED bit set to zero indicates that the phy does not support G3 without SSC.

A G3 WITH SSC SUPPORTED bit set to one indicates that the phy supports G3 (i.e., 6 Gbps) with SSC. A G3 WITH SSC SUPPORTED bit set to zero indicates that the phy does not support G3 with SSC.

The PARITY bit provides for error detection of all the SNW-3 information bits. The PARITY bit shall be set to one or zero such that the total number of SNW-3 information bits that are set to one is even, including the START bit and the PARITY bit. If the PARITY bit received is incorrect based upon the received bits, then the parity is bad and the phy shall consider it a phy reset problem (see 6.8.4.10.2). Phy reset problems may be counted and reported in the PHY RESET PROBLEM COUNT field in the SMP REPORT PHY ERROR LOG page (see 10.4.3.8) and the Protocol-Specific Port log page (see 10.2.8.1).

Table 82 lists examples of SNW-3 settings bits.

Table 82 — Example SNW-3 settings bits

Code[a]	Description
80540000h	Down-spreading SSC G1, G2, and G3 with SSC supported
80FC0001h	Down-spreading SSC G1, G2, and G3 with and without SSC supported
80A80000h	SSC not supported G1, G2, and G3 without SSC supported
C0FC0000h	Center-spreading SSC G1, G2, and G3 with and without SSC supported

[a] Expressed as a 32-bit value with bit 0 (i.e., the START bit) as the MSB and bit 31 as the LSB (i.e., the PARITY bit).

If SNW-3 is not supported, then the phy shall transmit D.C. Idle during SNW-3.

If SNW-3 is supported and a phy receives at least one COMWAKE, then SNW-3 is valid.

If SNW-3 is valid and the parity is bad, then a phy reset problem has occurred. Phy reset problems may be counted and reported in the PHY RESET PROBLEM COUNT field in the SMP REPORT PHY ERROR LOG page (see 10.4.3.8) and the Protocol-Specific Port log page (see 10.2.8.1).

The phy shall negotiate to the highest commonly supported settings based on the outgoing and incoming SNW-3 settings bits.

Table 83 defines the priority of the SNW-3 supported settings.

Table 83 — Negotiation priority of SNW-3 information supported settings

Priority	Supported Setting
Highest	G3 with SSC
...	G3 without SSC
...	G2 with SSC
...	G2 without SSC
...	G1 with SSC
Lowest	G1 without SSC

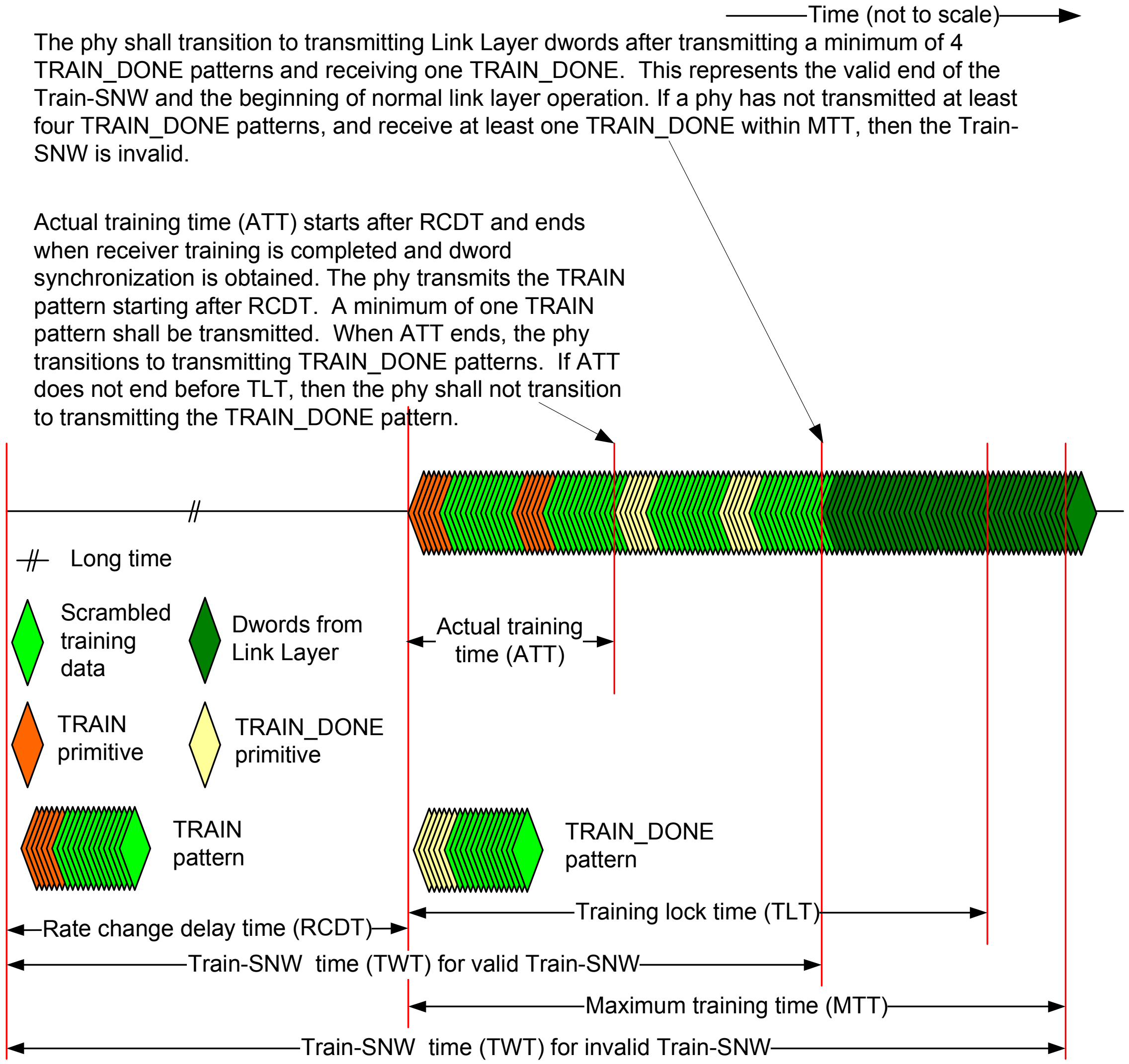
The transmitter shall use SAS signal output levels during the SAS speed negotiation sequence as described in 5.3.6.5.

6.7.4.2.3.3 Train-SNW Definition

Figure 124 defines the Train-SNW, including:

a) maximum Train-SNW window time (MTT);

b) rate change delay time (RCDT);
c) maximum train time (MTT);
d) train lock time (TLT); and
e) actual training time (ATT).



The phy shall transition to transmitting Link Layer dwords after transmitting a minimum of 4 TRAIN_DONE patterns and receiving one TRAIN_DONE. This represents the valid end of the Train-SNW and the beginning of normal link layer operation. If a phy has not transmitted at least four TRAIN_DONE patterns, and receive at least one TRAIN_DONE within MTT, then the Train-SNW is invalid.

Actual training time (ATT) starts after RCDT and ends when receiver training is completed and dword synchronization is obtained. The phy transmits the TRAIN pattern starting after RCDT. A minimum of one TRAIN pattern shall be transmitted. When ATT ends, the phy transitions to transmitting TRAIN_DONE patterns. If ATT does not end before TLT, then the phy shall not transition to transmitting the TRAIN_DONE pattern.

Time (not to scale)
Long time
Scrambled training data
Dwords from Link Layer
TRAIN primitive
TRAIN_DONE primitive
TRAIN pattern
TRAIN_DONE pattern
Actual training time (ATT)
Training lock time (TLT)
Rate change delay time (RCDT)
Train-SNW time (TWT) for valid Train-SNW
Maximum training time (MTT)
Train-SNW time (TWT) for invalid Train-SNW

Figure 124 — SAS Train-SNW

The Train-SNW utilizes TRAIN and TRAIN_DONE as defined in 7.2. TRAIN and TRAIN_DONE are used within training patterns.

There are two training patterns, the TRAIN pattern and the TRAIN_DONE pattern. Each training pattern consists of:

1) either a TRAIN if the pattern is a TRAIN pattern, or a TRAIN_DONE if the pattern is a TRAIN_DONE pattern; and
2) 58 dwords of value 0000_0000h that are transmitted scrambled and 8b10b encoded.

The scrambler is as defined for the link layer and shall be initialized at the end of RCDT within the Train-SNW. The scrambler shall not be re-initialized for the remainder of the Train-SNW.

The transmitter shall start transmitting TRAIN patterns at the end of RCDT portion of the Train-SNW with either starting disparity. The number of TRAIN patterns transmitted is determined by the time required for the phy's receiver to complete training. At least one TRAIN pattern shall be transmitted.

If the phy's receiver is trained and acquires dword synchronization before training lock time (TLT), then the phy shall stop transmitting TRAIN patterns and start transmitting TRAIN_DONE patterns. A minimum of four TRAIN_DONE patterns shall be transmitted.

If the phy transmits four or more TRAIN_DONE patterns and receives a minimum of one TRAIN_DONE before the maximum training time (MTT), then the phy shall stop transmitting TRAIN_DONE patterns, shall start transmitting dwords from the link layer and the Train-SNW is valid.

6.7.4.2.4 SAS speed negotiation sequence

The SAS speed negotiation sequence consists of a set of speed negotiation windows (see 6.7.4.2.3) for each physical link rate as follows:

1) SNW-1 (i.e., to negotiate 1,5 Gbps without training capabilities);
2) SNW-2 (i.e., to negotiate 3 Gbps without training capabilities); and
3) one of the following sequences:
   a) a Final-SNW negotiating 1,5 Gbps if SNW-1 is valid and SNW-2 is invalid; or
   b) an SNW-3 followed by one of the following:
      A) a Final-SNW negotiating 3,0 Gbps if SNW-2 is valid and SNW-3 is invalid; or
      B) one or more Train-SNWs if SNW-3 is valid, the parity is good (see Table 81 and Table 83) and at least one settings bit in common.

If SNW-1, SNW-2, and SNW-3 are all invalid, then a phy reset problem has occurred. Phy reset problems may be counted and reported in the PHY RESET PROBLEM COUNT field in the SMP REPORT PHY ERROR LOG page (see 10.4.3.8) and the Protocol-Specific Port log page (see 10.2.8.1).

If a Final-SNW is performed and is invalid, then a phy reset problem has occurred.

If a phy participates in a valid SNW-3 and parity is good, then the phy shall participate in a Train-SNW utilizing the highest commonly supported settings based on the outgoing and incoming SNW-3 settings bits (see Table 80).

If settings are exchanged during SNW-3 and the parity is bad or there are no common settings, then a phy reset problem occurred. Phy reset problems may be counted and reported in the PHY RESET PROBLEM COUNT field in the SMP REPORT PHY ERROR LOG page (see 10.4.3.8) and the Protocol-Specific Port log page (see 10.2.8.1).

If a Train-SNW is invalid and there are additional, untried, commonly supported settings exchanged during SNW-3, then a new Train-SNW shall be performed at the next highest, untried, commonly supported capability. If a Train-SNW is invalid and there are no additional, untried, commonly supported settings exchanged during SNW-3, then a phy reset problem occurred.

Figure 125 is an example of speed negotiation between a phy A that supports G1 through G3 and a phy B that only supports G2. Both phys participate in:

1) the SNW-1 (supported by phy A but not by phy B);
2) the SNW-2 (supported by both phys);
3) the SNW-3 (supported by phy A but not by phy B); and
4) the Final-SNW at G2.

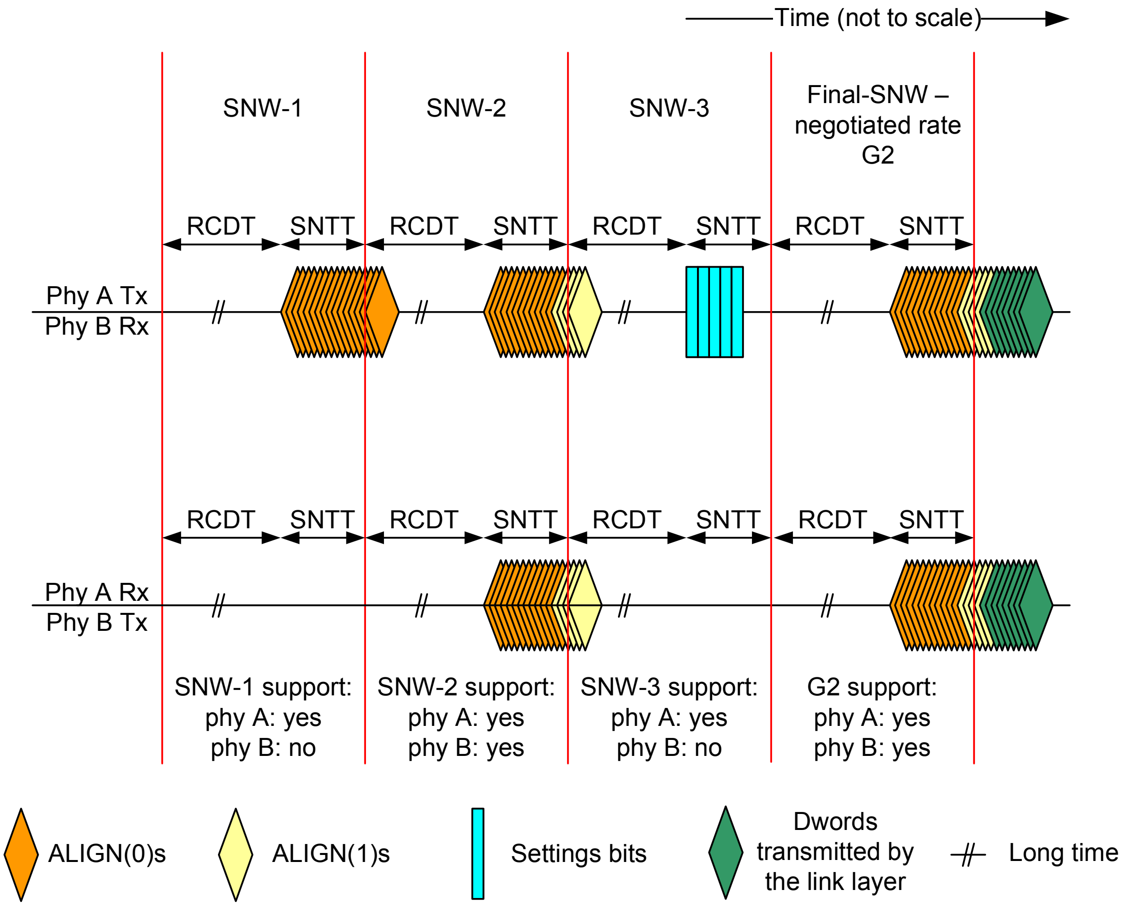


Figure 125 — SAS speed negotiation sequence (phy A: G1, G2, G3, phy B: G2 only)

Phy A and phy B detect an invalid SNW-1, a valid SNW-2 followed by an invalid SNW-3 and correctly proceeds to a Final-SNW at G2.

If the phy does not achieve dword synchronization during the Final-SNW, then the SAS speed negotiation sequence fails. This is a phy reset problem and may be counted and reported in the PHY RESET PROBLEM COUNT field in the SMP REPORT PHY ERROR LOG page (see 10.4.3.6) and the Protocol-Specific Port log page (see 10.2.8.1).

Figure 126 shows a speed negotiation sequence where phy B does not achieve dword synchronization during the Final-SNW. If this occurs, the handshake is not complete and the phy reset sequence is retried.

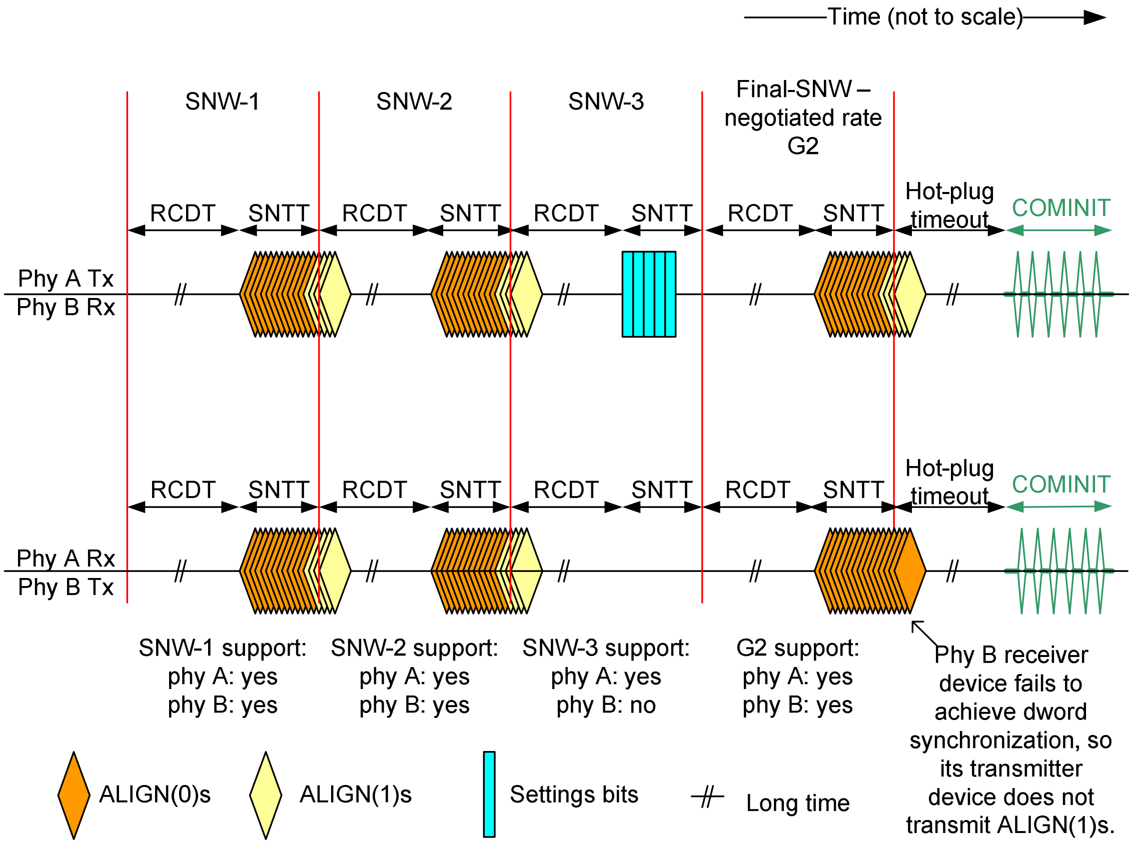

Time (not to scale)
SNW-1
SNW-2
SNW-3
Final-SNW – negotiated rate G2
RCDT
SNTT
Hot-plug timeout
COMINIT
Phy A Tx
Phy B Rx
Phy A Rx
Phy B Tx
SNW-1 support: phy A: yes phy B: yes
SNW-2 support: phy A: yes phy B: yes
SNW-3 support: phy A: yes phy B: no
G2 support: phy A: yes phy B: yes
Phy B receiver device fails to achieve dword synchronization, so its transmitter device does not transmit ALIGN(1)s.
ALIGN(0)s
ALIGN(1)s
Settings bits
Long time

Figure 126 — SAS Final-SNW phy reset problem

Figure 127 shows speed negotiation between a phy A and a phy B that both support G1 and G2 rates without requiring training and also support additional capabilities. Both phys participate in:

1) the SNW-1 (supported by both phys);
2) the SNW-2 (supported by both phys);
3) the SNW-3 (supported by both); and
4) the Train-SNW (supported by both phys).

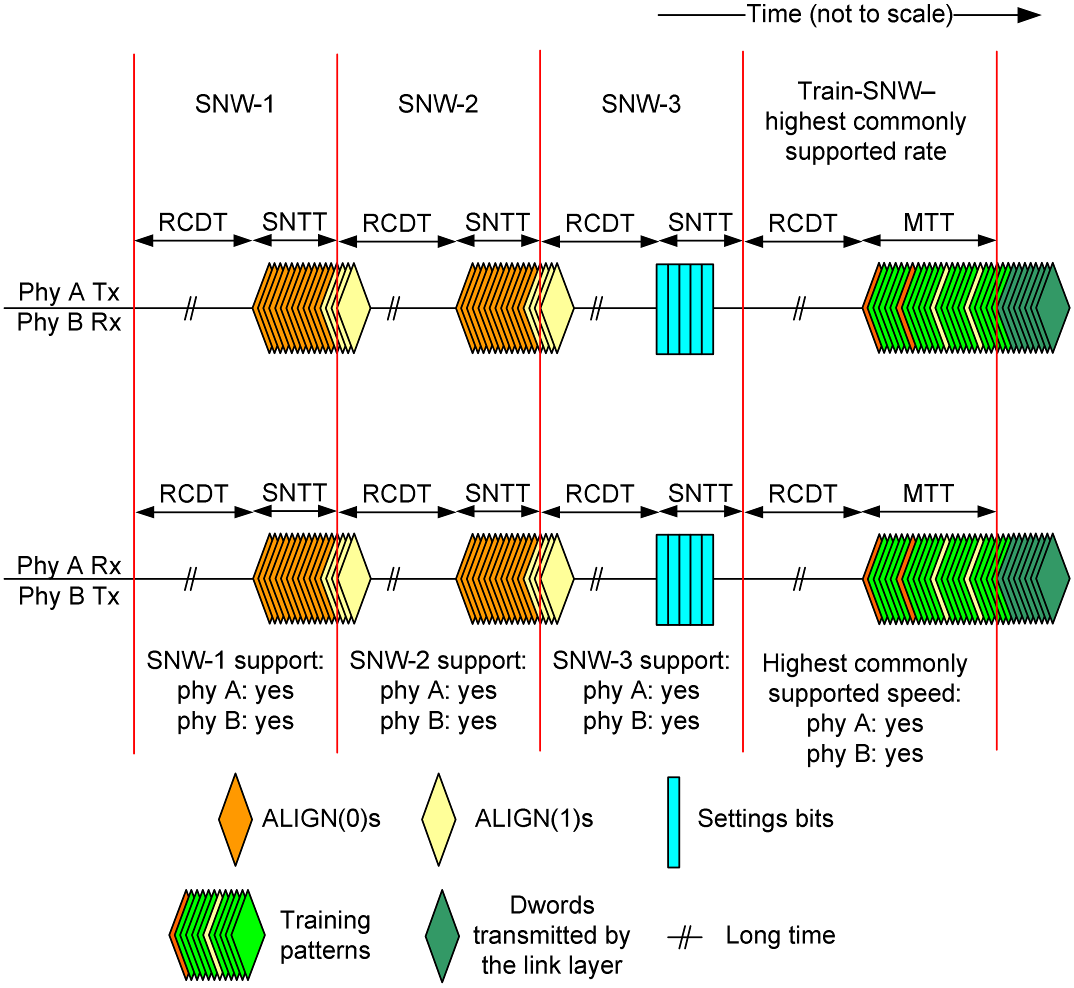


Figure 127 — SAS speed negotiation sequence (phy A: G1, G2, G3 or higher, phy B: G1, G2, G3 or higher)

A phy reset problem may be encountered if:

a) the phys do not exchange the settings bits properly (e.g., due to a parity error) in SNW-3 (see Figure 128); or
b) either phy does not complete training within the MTT interval of several training windows exhausting all common settings exchanged in SNW-3 (see Figure 129)..

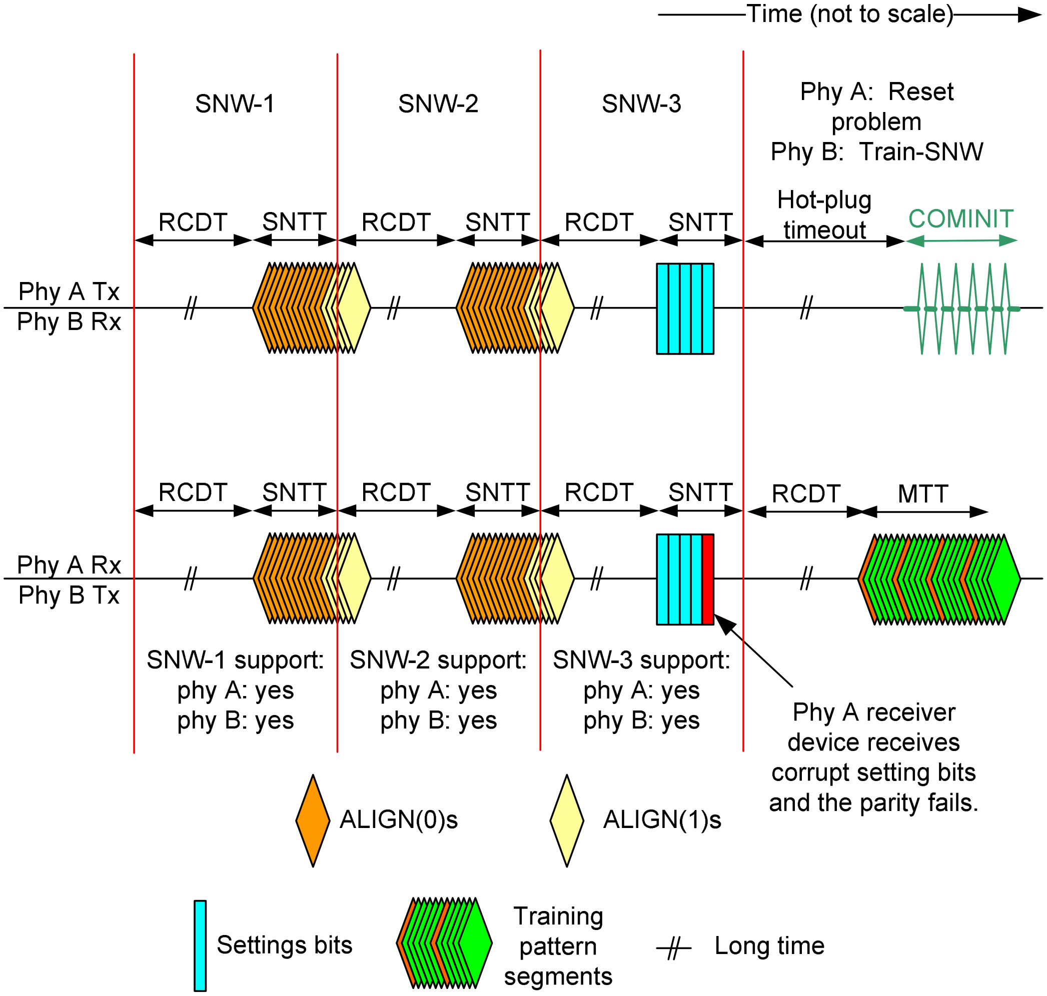


Figure 128 — SAS speed negotiation sequence - SNW-3 (maximum SNW) phy reset problem

These phy reset problems may be counted and reported in the PHY RESET PROBLEM COUNT field in the SMP REPORT PHY ERROR LOG page (see 10.4.3.6) and the Protocol-Specific Port log page (see 10.2.8.1).

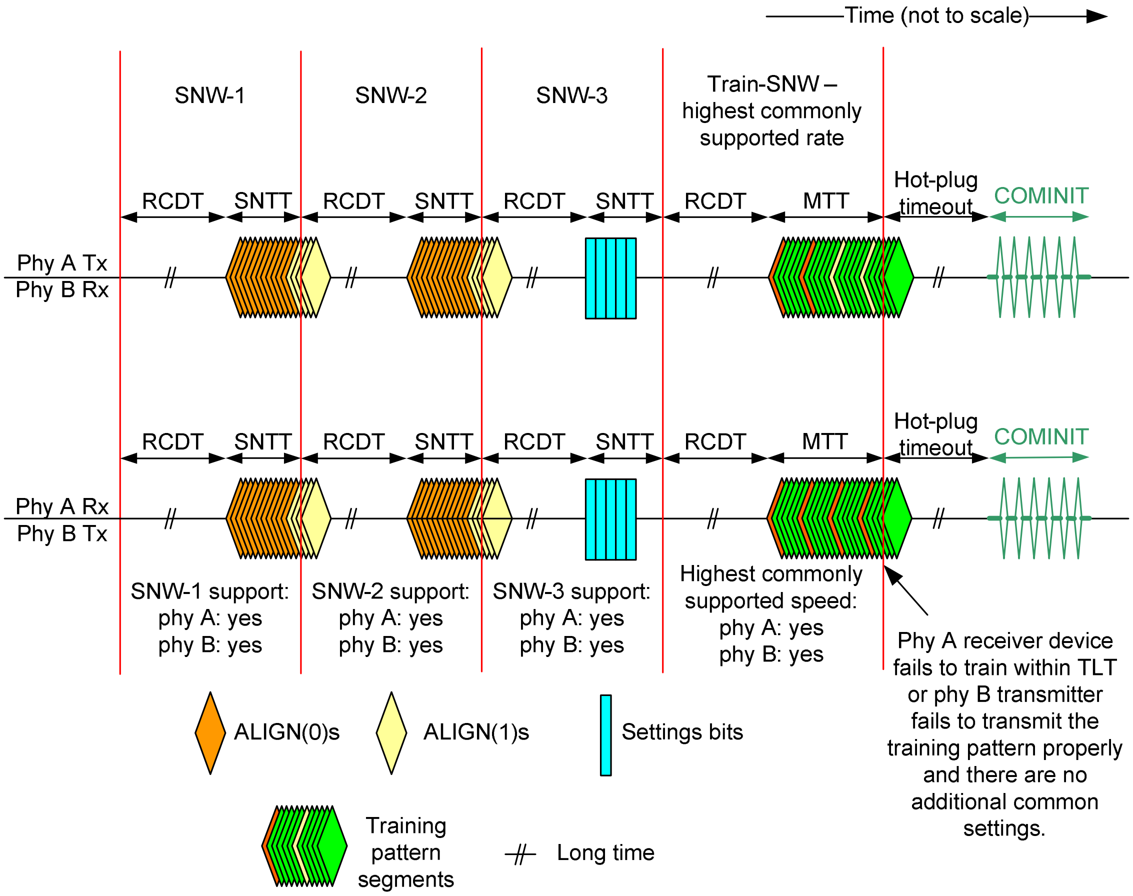


Figure 129 — SAS speed negotiation sequence - training (Train-SNW) phy reset problem

If settings bits are exchanged that contain more than one common setting and the Train-SNW using the highest common capability is invalid, a second Train-SNW is performed using the next highest common setting, as shown in Figure NEW-1.

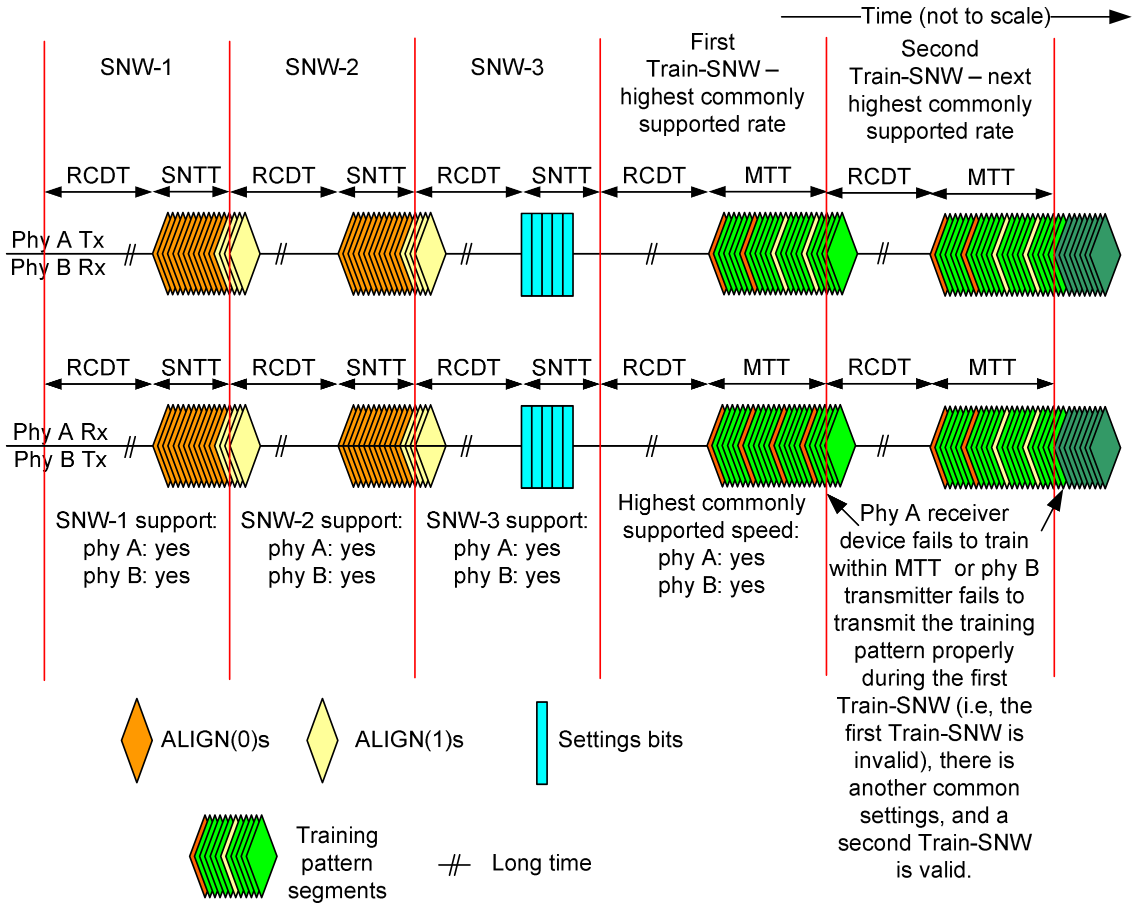

Time (not to scale)
SNW-1
SNW-2
SNW-3
First Train-SNW – highest commonly supported rate
Second Train-SNW – next highest commonly supported rate
RCDT
SNTT
RCDT
SNTT
RCDT
SNTT
RCDT
MTT
RCDT
MTT
Phy A Tx
Phy B Rx
RCDT
SNTT
RCDT
SNTT
RCDT
SNTT
RCDT
MTT
RCDT
MTT
Phy A Rx
Phy B Tx
SNW-1 support: phy A: yes phy B: yes
SNW-2 support: phy A: yes phy B: yes
SNW-3 support: phy A: yes phy B: yes
Highest commonly supported speed: phy A: yes phy B: yes
Phy A receiver device fails to train within MTT or phy B transmitter fails to transmit the training pattern properly during the first Train-SNW (i.e, the first Train-SNW is invalid), there is another common settings, and a second Train-SNW is valid.
ALIGN(0)s
ALIGN(1)s
Settings bits
Training pattern segments
Long time

Figure NEW1 — **SAS speed negotiation sequence – First Train-SNW is invalid, Second Train-SNW valid**

For more examples of speed negotiations between phys that support various speeds, see Annex C.

<Editor's Note: We should review what is in Annex C and see if more examples are needed.>

Clause 6.8
Suggested changes to clause 6.8, 6.8.1, and 6.8.2: All changes are additions of blue text.

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 sets of states:

a) OOB sequence (OOB) states;
b) SAS speed negotiation (SAS) states; and
c) 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 6.8.3.4);
e) SP4:OOB_COMSAS (see 6.8.3.4);
f) SP5:OOB_AwaitCOMSAS_Sent (see 6.8.3.4);
g) SP6:OOB_AwaitNoCOMSAS (see 6.8.3.4);
h) SP7:OOB_AwaitCOMSAS (see 6.8.3.4);
i) SP8:SAS_Start (see 6.8.3.4);
j) SP9:SAS_WindowNotSupported (see 6.8.3.4);
k) SP10:SAS_AwaitALIGN (see 6.8.3.4);
l) SP11:SAS_AwaitALIGN1 (see 6.8.3.4);
m) SP12:SAS_AwaitSNW (see 6.8.3.4);
n) SP13:SAS_Pass (see 6.8.3.4);
o) SP14 SAS_Fail (see 6.8.3.4);
p) SP15:SAS_PHY_Ready (see 6.8.3.4);
q) SP16:SATA_COMWAKE (see 6.8.3.4);
r) SP17:SATA_AwaitCOMWAKE (see 6.8.3.4);
s) SP18:SATA_AwaitNoCOMWAKE (see 6.8.3.4);
t) SP19:SATA_AwaitALIGN (see 6.8.3.4);
u) SP20:SATA_AdjustSpeed (see 6.8.3.4);
v) SP21:SATA_Transmit_ALIGN (see 6.8.3.4);
w) SP22:SATA_PHY_Ready (see 6.8.3.4);
x) SP23:SATA_PM_Partial (see 6.8.3.4);
y) SP24:SATA_PM_Slumber (see 6.8.3.4);
z) SP25:SATA_PortSel (see 6.8.3.4); and
aa) SP26:SATA_SpinupHold (see 6.8.3.4).
bb) SP27:SAS_Settings (see 6.8.3.4);
cc) SP28:SAS_TrainSetup (see 6.8.3.4);
dd) SP29:SAS_Train (see 6.8.3.4); and
ee) SP30:SAS_TrainingDone (see 6.8.3.4).

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 Speed Negotiation Window state variable to determine the current speed negotiation window type (e.g., SNW-2 or Final-SNW).

If the phy supports SNW-3, then the SP state machine shall maintain a Common Supported Settings state variable that contains supported settings (see Table 81) common between the transmitted settings bits and the received settings bits.

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 84.

Table 84 — SP state machine timers

Timer	Initial value
COMSAS Detect Timeout timer	COMSAS detect timeout (see table 66 in 6.6.1)
Await ALIGN Timeout timer	Await ALIGN timeout (see table 73 in 6.7.2.2)
Hot-Plug Timeout timer	Hot plug timeout (see table 72 in 6.7.1)
RCDT timer	RCDT (see table 77 in 6.7.4.2)
SNLT timer	SNLT (see table 77 in 6.7.4.2)
SNTT timer	SNTT (see table 77 in 6.7.4.2)
TLT timer	TLT (see table 77 in 6.7.4.2)
MTT timer	MTT (see table 77 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 argument or an SSC OFF argument;
g) Transmit ALIGN with an argument indicating the specific type (e.g., Transmit ALIGN (0));
h) Transmit Settings Bits;
i) Transmit TRAIN pattern; and
j) Transmit TRAIN_DONE pattern.

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

The SP transmitter shall complete any physical link rate change requested with the Set Rate message within RCDT (see table 77 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; and
f) Settings Bits 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) COMSAS Completed;
d) COMWAKE Detected;
e) COMWAKE Completed;
f) ALIGN Received with an argument indicating the specific type (e.g., ALIGN Received (0));
g) Dword Received;
h) Settings Bits Completed with arguments (i.e., received setting bits argument and parity good argument or parity bad argument;
i) Training Completed; and
j) TRAIN_DONE Received.

The ALIGN Received message, Dword Received message, and TRAIN_DONE Received message 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.

No changes to clause 6.8.3. But we will have to change the figure to show the transitions from figure 33 back to SP0 and SP1 on that diagram.

Suggested changes to clause 6.8.4: All changes are additions of blue text. Section numbers that do not appear below are not changed.

6.8.4 SAS speed negotiation states

6.8.4.1 SAS speed negotiation states overview

Figure 131 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.

Changes to Figure 132 are in RED and Underlined in the figure. The associated transition arrows are also added in RED.

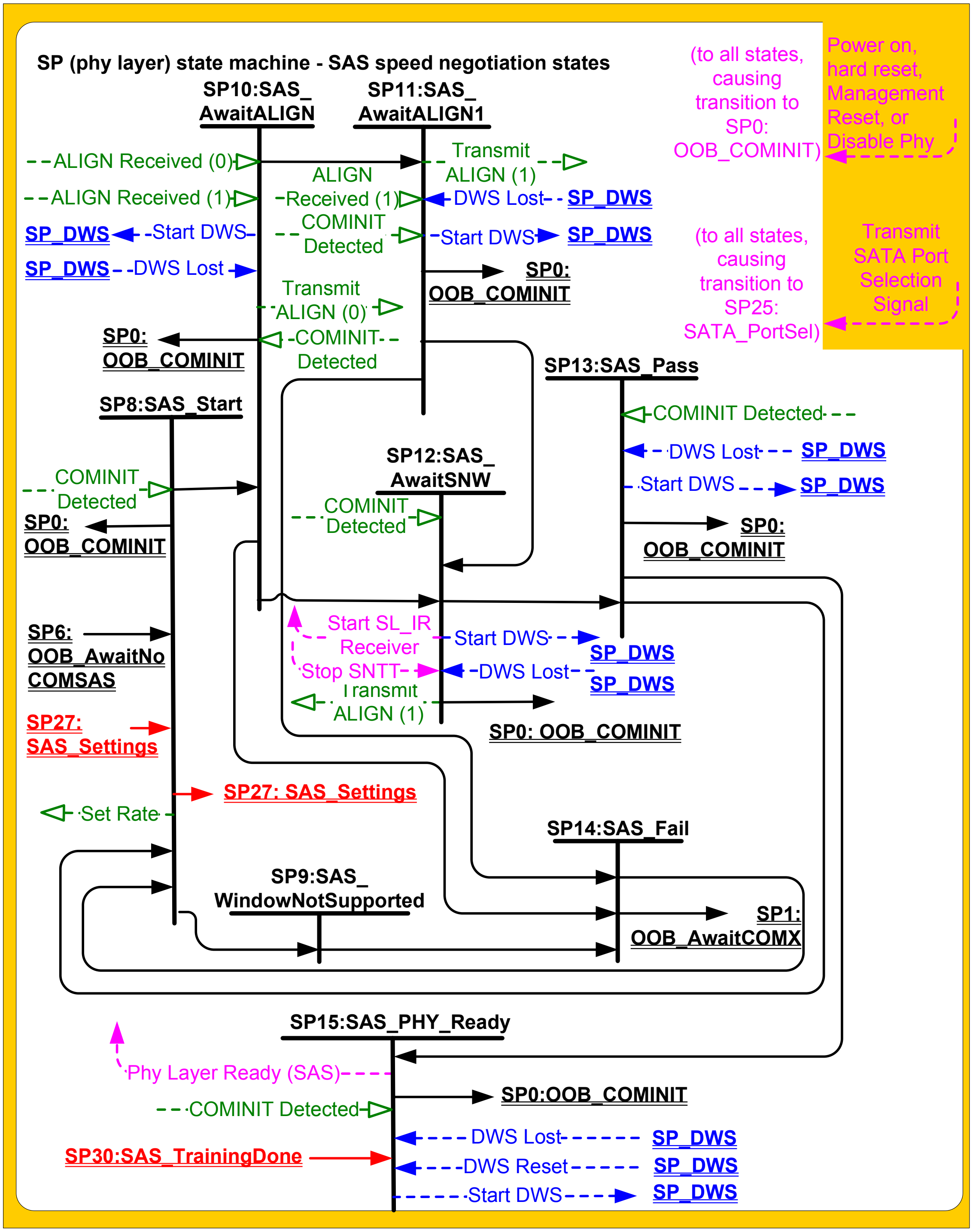


Figure 131— SP (phy layer) state machine - SAS speed negotiation states

Figure 132 shows the SAS speed negotiation states related to SNW-3 and the Train-SNW speed negotiation windows.

Figure 132 is a completely new figure.

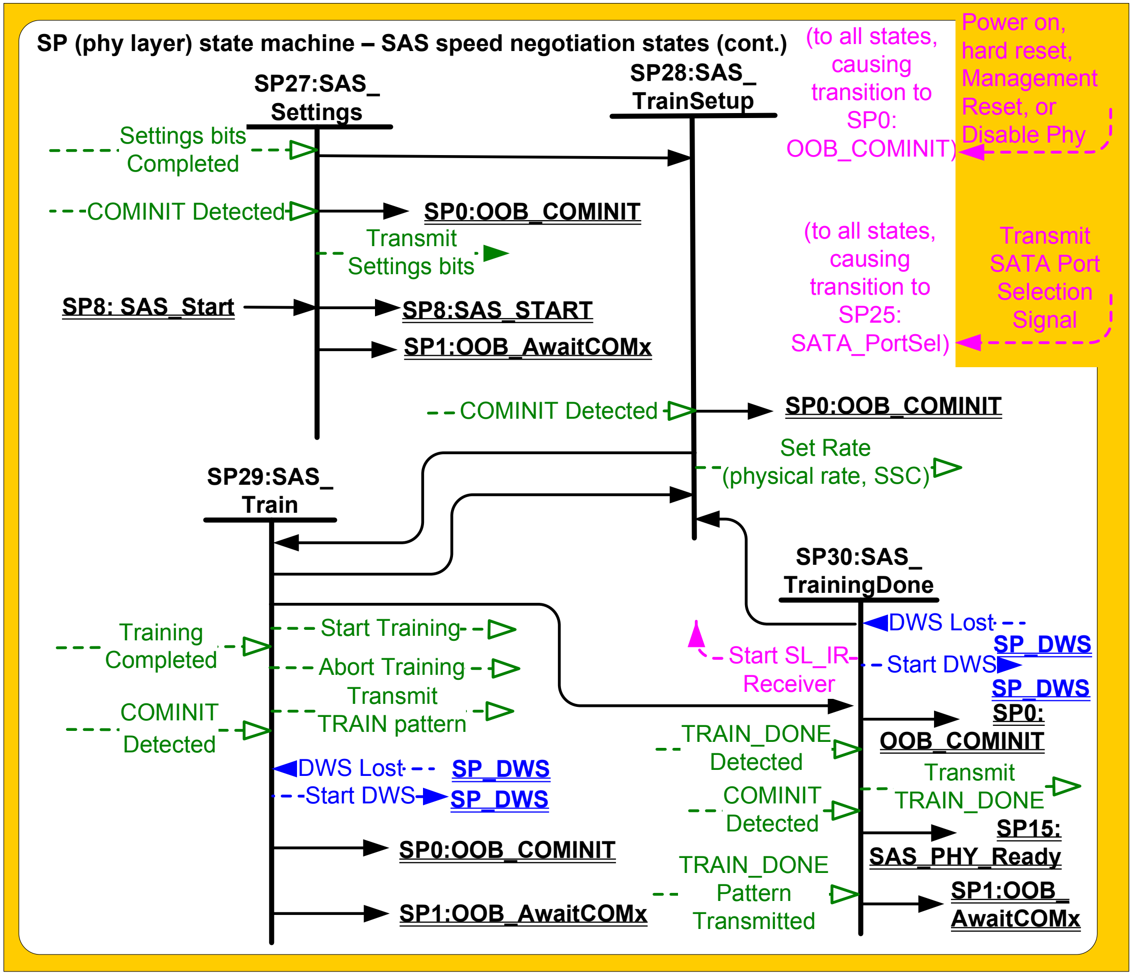


Figure 132 — SP (phy layer) state machine - SAS speed negotiation states related to SNW-3 and Final-SNW

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.

If this state is entered from SP6:OOB_AwaitNoCOMSAS, then the Current Speed Negotiation Window state variable shall be set to SNW-1. If this state is not entered from SP6:OOB_AwaitNoCOMSAS, then the Current Speed Negotiation Window state variable shall be set to:

a) SNW-2 if the present value of the Current Speed Negotiation Window state variable is SNW-1;
b) SNW-3 if the present value of the Current Speed Negotiation Window state variable is SNW-2, and either SNW-1 is invalid or SNW-2 is valid;

c) Final-SNW if the present value of the Current Speed Negotiation Window state variable is SNW-2, and SNW-1 is valid and SNW-2 is invalid;
d) Final-SNW if the present value of the Current Speed Negotiation Window state variable is SNW-3, SNW-3 is invalid and SNW-2 is valid; or
e) Phy Reset Problem if the present value of the Current Speed Negotiation Window state variable is SNW-3, SNW-3 is invalid and SNW-2 is invalid.

After the Current Speed Negotiation Window state variable is updated, and if the Current Speed Negotiation Window state variable is not set to Phy Reset Problem, then this state shall:

a) initialize and start the RCDT timer; and
b) if the speed negotiation window specified by the Current Speed Negotiation Window state 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 Speed Negotiation Window state variable is set to SNW-1;
   B) the Physical Link Rate argument set to 3,0 Gbps and an SSC OFF argument, if the Current Speed Negotiation Window state variable is set to SNW-2;
   C) the Physical Link Rate arguments set to 1,5 Gbps and an SSC OFF argument, if the Current Speed Negotiation Window state variable is set to Final-SNW, SNW-1 was valid and SNW-2 was invalid;
   D) the Physical Link Rate arguments set to 3,0 Gbps and an SSC OFF argument, if the Current Speed Negotiation Window state 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 Speed Negotiation Window state 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.

6.8.4.2.3 Transition SP8:SAS_Start to SP9:SAS_WindowNotSupported

This transition shall occur after the RCDT timer expires if the speed negotiation window specified by the Current Speed Negotiation Window state variable is not supported.

6.8.4.2.4 Transition SP8:SAS_Start to SP10:SAS_AwaitALIGN

This transition shall occur after the RCDT timer expires if the Current Speed Negotiation Window state variable is not set to SNW-3 and the speed negotiation window specified by the Current Speed Negotiation Window state variable is supported.

6.8.4.2.5 Transition SP8:SAS_Start to SP27:SAS_Settings

This transition shall occur after the RCDT timer expires if the Current Speed Negotiation Window state variable is set to SNW-3 and SNW-3 is supported.

6.8.4.2.6 Transition SP8:SAS_Start to SP1:OOB_AwaitCOMX

This transition shall occur if the Current Speed Negotiation Window state variable is set to Phy Reset Problem.

6.8.4.3 SP9:SAS_WindowNotSupported state

6.8.4.3.1 State description

Upon entry into this state the SNTT timer shall be initialized and started.

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 SNTT timer and SNLT timer shall be initialized and started and this state shall 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.

Upon entry into this state, this state shall send a Start DWS message to the SP_DWS state machine.

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.

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.

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 speed negotiation window.

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 speed negotiation window.

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 this is 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.

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 speed negotiation window 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.

6.8.4.7.3 Transition SP13:SAS_Pass to SP8:SAS_Start

This transition shall occur if the Current Speed Negotiation Window state 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 Speed Negotiation Window state 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 speed negotiation window 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 Speed Negotiation Window state variable is set to Final-SNW.

6.8.4.8.3 Transition SP14:SAS_Fail to SP8:SAS_Start

This transition shall occur if the Current Speed Negotiation Window state 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, or a DWS Reset message.

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

Upon entry into this state, this state shall 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.

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.

6.8.4.9.2 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).

The remaining changes to clause 6.8.4 are all insertions of new paragraphs – exact paragraph number to be determined at time of incorporation into the SAS-2 draft.

6.8.4.10 SP27:SAS_Settings state

6.8.4.10.1 State description

This state transmits and receives settings bits.

Upon entry to this state, the Common Supported Settings state variable shall indicate that there are no settings bits (see Table 81) in common.

Upon entry into this state, this state shall send Transmit Settings Bits message to the SP transmitter.

If a Settings Bits Detected message is received with the argument of parity good, then the Common Supported Settings state variable shall be set to reflect the bits from bit 8 through bit 13 that are set to one in both the transmitted and received settings bits.

This state waits for the SNTT timer to expire.

6.8.4.10.2 Transition SP27:SAS_Settings to SP0:OOB_COMINIT

This transition shall occur after receiving a COMINIT Detected message.

6.8.4.10.3 Transition SP27:SAS_Settings to SP1:OOB_AwaitCOMX

This transition shall occur after the SNTT timer expires if:

a) a Settings Bits Detected message is received with an argument of parity bad; or
b) no common settings exist after the Common Supported Settings state variable is set as a result of receiving a Settings Bits Detected message.

6.8.4.10.4 Transition SP27:SAS_Settings to SP8:SAS_Start

This transition shall occur if:

a) the SNTT timer expires; and
b) a Settings Bits Detected message is not received during this state.

6.8.4.10.5 Transition SP27:SAS_Settings to SP28:SAS_TrainSetup

This transition shall occur:

a) after the SNTT timer expires; and
b) if Common Supported Settings state variable indicates at least one settings bit in common.

6.8.4.11 SP28:SAS_TrainSetup

6.8.4.11.1 State description

Upon entry into this state, this state shall:

a) initialize and start the RCDT timer; and
b) send a Set Rate message to the SP transmitter and receiver with the arguments set to reflect the highest priority setting (see Table 83) contained in the Common Supported Settings state variable;

After the Set Rate message is sent, the Common Supported Settings state variable shall be set to indicate the settings bit used for the Set Rate message is no longer in common.

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

6.8.4.11.2 Transition SP28:SAS_TrainSetup to SP0:OOB_COMINIT

This transition shall occur after receiving a COMINIT Detected message.

6.8.4.11.3 Transition SP28:SAS_TrainSetup to SP29:SAS_Train

This transition shall occur after the RCDT timer expires.

6.8.4.12 SP29:SAS_Train state

6.8.4.12.1 State description

Upon entry into this state:

a) the MTT timer shall be initialized and started;
b) the TLT timer shall be initialized and started;

c) a Start Training message shall be sent to the SP receiver; and
d) a Start DWS message shall be sent to the SP_DWS state machine.

During this state Transmit TRAIN Pattern message shall be repeatedly sent to the SP transmitter.

Each time this state receives a DWS Lost message, this state shall send a Start DWS message to the SP_DWS state machine to re-acquire dword.

If the MTT timer expires, then this state shall send an Abort Training message to the SP receiver.

6.8.4.12.2 Transition SP29:SAS_Train to SP0:OOB_COMINIT

This transition shall occur after receiving a COMINIT Detected message.

6.8.4.12.3 Transition SP29:SAS_Train to SP1:OOB_AwaitCOMX

This transition shall occur if

a) the MTT timer expires; and
b) the Common Supported Settings state variable does not contain additional common settings.

This is a phy reset problem.

6.8.4.12.4 Transition SP29:SAS_Train to SP30:SAS_TrainingDone

This transition shall occur if:

a) this state receives a TRAIN Completed message before the TLT timer expires; and
b) dword synchronization is acquired.

6.8.4.12.5 Transition SP29:SAS_Train to SP28:SAS_TrainSetup

This transition shall occur if:

a) the MTT timer expires; and
b) the Common Supported Settings state variable contains additional common settings.

6.8.4.13 SP30:SAS_TrainingDone state

6.8.4.13.1 State description

During this state shall repeatedly send Transmit TRAIN_DONE pattern message 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 MTT timer to expire or a TRAIN_DONE Received message from the receiver.

This state shall send a Start SL_IR Receiver confirmation to the link layer when a TRAIN_DONE Received message is received.

6.8.4.13.2 Transition SP30:SAS_TrainingDone 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; or
b) receiving a COMINIT Detected message.

6.8.4.13.3 Transition SP30:SAS_TrainingDone to AwaitCOMX

This transition shall occur if:

a) TRAIN_DONE Received message is not received before the MTT timer expires; and
b) the Common Supported Settings state variable does not contain additional common settings bits.

This is a phy reset problem.

6.8.4.13.4 Transition SP30:SAS_TrainingDone to SP28:SAS_TrainSetup

This transition shall occur if:

a) the MTT timer expires; and
b) the Common Supported Settings state variable contains additional common settings bits.

6.8.4.13.5 Transition SP30:SAS_TrainingDone to SP15:SAS_PHY_Ready

This transition shall occur if this state receives:

a) at least four TRAIN_DONE Patterns Transmitted messages; and
b) a TRAIN_DONE Received message before the MTT timer expires.