

# Serial Attached SCSI Management upper layers and port layer



by Rob Elliott

HP Industry Standard Servers

Server Storage Advanced Technology

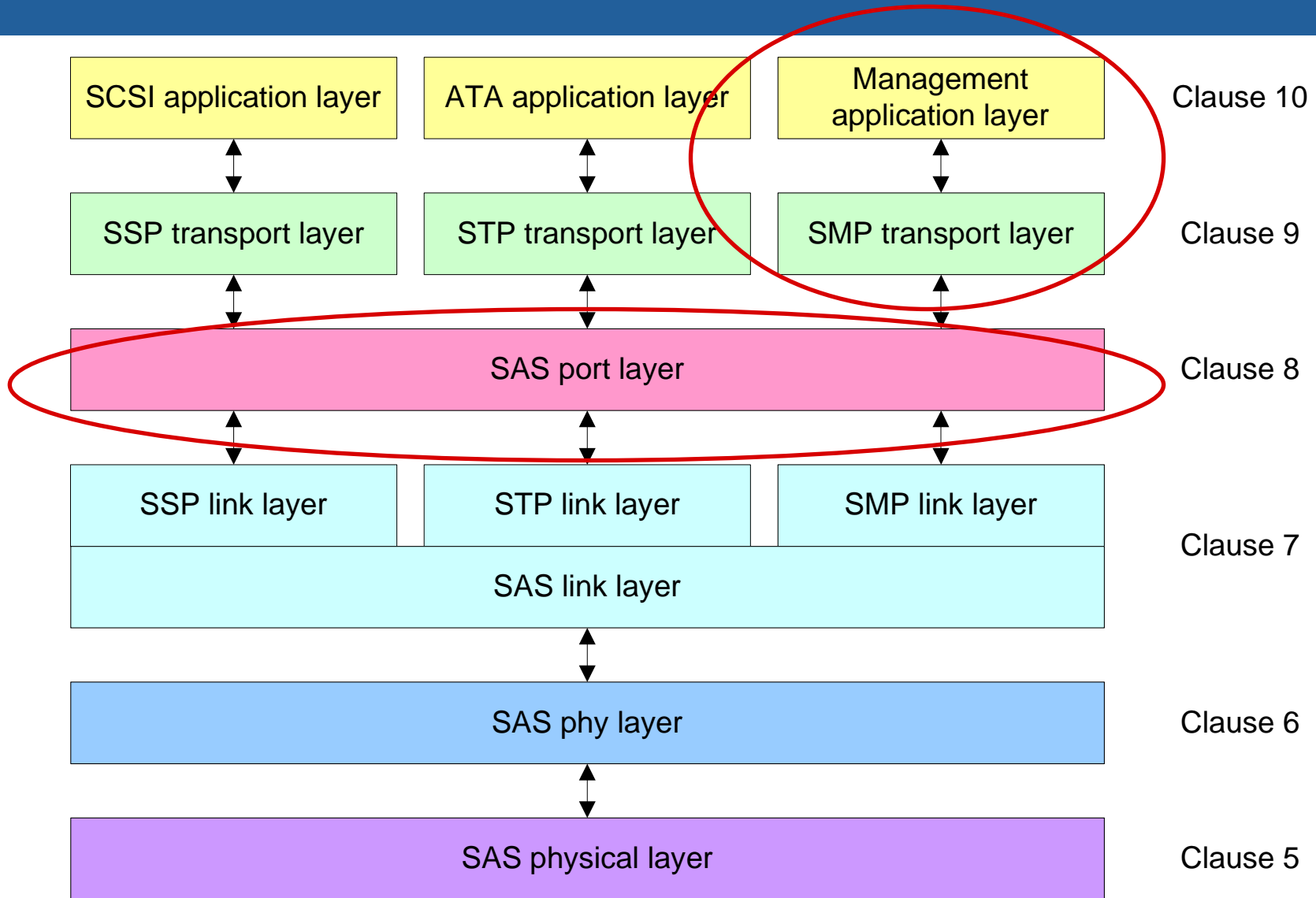
[elliott@hp.com](mailto:elliott@hp.com) <http://www.hp.com>

30 September 2003

- These slides are freely distributed by HP through the SCSI Trade Association (<http://www.scsita.org>)
- STA members are welcome to borrow any number of the slides (in whole or in part) for other presentations, provided credit is given to the SCSI Trade Association and HP
- This compilation is © 2003 Hewlett-Packard Corporation



# SAS standard layering



# Table of contents

- SMP transport layer
- SMP application layer
  - General read functions
  - Phy-specific read functions
  - Write functions
- Port layer
- Wrap up

# SMP transport layer

# SMP Request and Response frames



- **SMP Request** frame from initiator to target

Byte	Field(s)
0	SMP Frame Type (40h)
1 to n	Request bytes
m to (n-3)	Fill bytes, if needed
(n-3) to n	CRC

- **SMP Response** frame from target to initiator

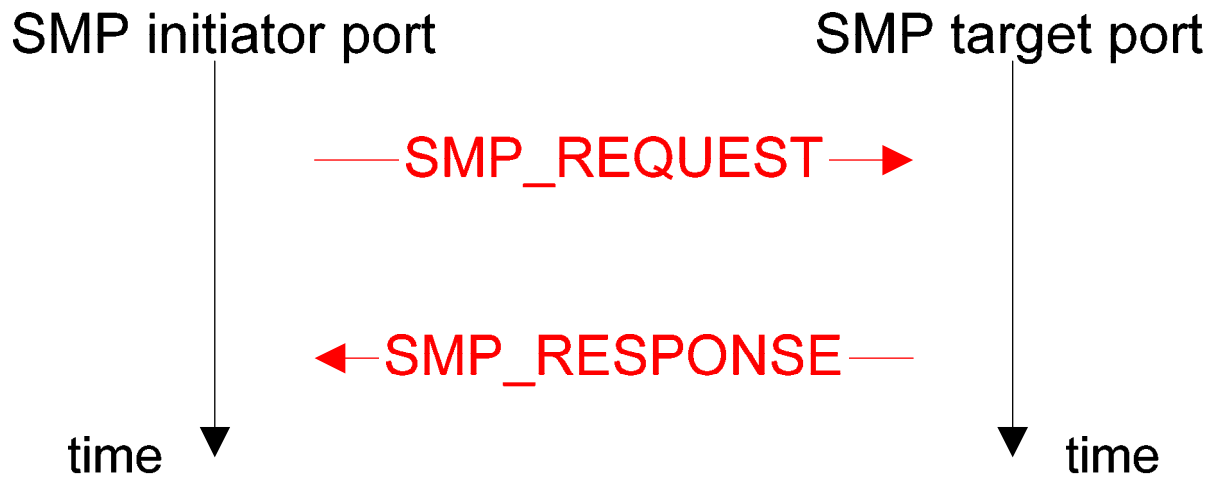
Byte	Field(s)
0	SMP Frame Type (41h)
1 to n	Response bytes
m to (n-3)	Fill bytes, if needed
(n-3) to n	CRC

- As with SSP, the link layer checks the **CRC** field
  - Target: BREAK the connection
  - Initiator: CLOSE the connection

# SMP frame sequence



- Initiator must open the connection
- One and only one frame transferred in each direction
- Both sides close the connection after SMP\_RESPONSE is transferred



# SMP transport layer state machines



- Separate state machines for initiator port and target port
  - **MT\_IP** – transport layer for SMP initiator ports
    - Upon request from SMP application layer
      1. Transmit an **SMP Request** frame
      2. Receive an **SMP Response** frame
  - **MT\_TP** – transport layer for SMP target ports
    - Upon arrival of a frame
      - Receive an **SMP Request** frame
      - Transmit an **SMP Response** frame

# SMP application layer

# SMP Application layer frame formats



- **SMP Request** frame from initiator to target
  - **Function** field (1 byte)
    - requests certain function be performed

Byte	Field(s)
0	SMP Frame Type (40h)
1	Function
2 to 3	Reserved
4 to (n-3)	Function Specific
(n-3) to n	CRC

- **SMP Response** frame from target to initiator
  - **Function** field (1 byte)
    - Indicates the function that was performed
  - **Function Result** field (1 byte)
    - Indicates the result

Byte	Field(s)
0	SMP Frame Type (41h)
1	Function
2	Function Result
3	Reserved
4 to (n-3)	Function Specific
(n-3) to n	CRC

# SMP functions



- Two functions apply to the whole expander

Frame type	Function name	Description
00h	REPORT GENERAL	Information about the expander
01h	REPORT MANUFACTURER INFORMATION	INQUIRY command-like information about the expander

# SMP functions



- The rest apply to a specific phy
- 00h – 7Fh are reads; 80h – FFh are writes

Frame type	Function name	Description
10h	DISCOVER	Miscellaneous information
11h	REPORT PHY ERROR LOG	Error counters
12h	REPORT PHY SATA	Information concerning attachment to a SATA device
13h	REPORT ROUTE INFORMATION	Read routing table entry
90h	CONFIGURE ROUTE INFORMATION	Write routing table entry
91h	PHY CONTROL	Configure various phy settings

# SMP function results



Value	Function Result	Description
00h	SMP Function Accepted	Good result
01h	Unknown SMP Function	Target doesn't support the request
02h	SMP Function Failed	Request failed for some reason
03h	Invalid Request Frame Length	Request frame length was invalid
10h	Phy Does Not Exist	For functions including a <b>Phy Identifier</b> field, the <b>Phy Identifier</b> was out of range
11h	Index Does Not Exist	For functions including an <b>Expander Route Index</b> field, the <b>Expander Route Index</b> was out of range (or the specified phy doesn't have a routing table at all)
12h	Phy Does Not Support SATA	REPORT PHY SATA requested, but no SATA device attached.
13h	Unknown Phy Operation	Unknown PHY CONTROL <b>Phy Operation</b> requested.

# SMP application layer – general read functions

# SMP REPORT GENERAL and REPORT MANUFACTURER INFORMATION request frames



- For the whole-expander read functions, the **SMP Request** frame contains nothing more than the function number

Byte	Field(s)
0	SMP Frame Type (40h)
1	Function (00 or 01h)
2 to 3	Reserved
4 to 7	CRC

# SMP REPORT GENERAL response frame



- Returns information about the expander as a whole
- No matter which phy of the expander is queried, the same result

Byte	Field(s)		
0	SMP Frame Type (41h)		
1	Function (00h)		
2	Function Result		
3	Reserved		
4 to 5	Expander Change Count		
6 to 7	Expander Route Indexes		
8	Reserved		
9	Number of Phys		
10	Reserved	Configuring	Configurable
11 to 27	Reserved		
28 to 31	CRC		

# SMP REPORT GENERAL response frame Expander Change Count field



- **Expander Change Count** field (2 bytes)
  - Every time the expander generates a BROADCAST (CHANGE), it increments this field
  - BROADCAST (CHANGE) includes no indication of where it came from
    - Requires querying all expanders
  - If the value has not changed, software knows the change did not come from behind this expander
    - Avoid searching through parts of the domain that were not the source
  - Since the counter wraps at  $2^{16}$  (FFFFh -> 0000h), there is a slight possibility of missing a change if exactly that many events happen between reads

# SMP REPORT GENERAL response frame Expander Route Indexes and Number of Phys fields



- **Expander Route Indexes** field (2 bytes)
  - Indicates the maximum depth of a route table
  - Each phy could have a different depth
  - This just indicates the depth of the phy with the largest table
  - Field is 16 bits, so 16K indexes maximum
- **Number of Phys** field (1 byte)
  - Indicates how many phys exist in the expander
  - numbered 0 .. N
  - Field is 8 bits, so 255 phys maximum
- Software must assume that the number of routing table entries for the expander is **Number of Phys x Expander Route Indexes**

# SMP REPORT GENERAL response frame Configurable and Configuring fields



- **Configurable** bit
  - An expander is not required to have programmable routing tables
  - It could fill in its tables itself
  - Means it contains SMP initiator port functionality (to probe other expanders)
  - If so, this bit is 1
- **Configuring** bit
  - If expander is not configurable, this bit indicates that it is currently changing its tables
  - Lets software know when it is safe to probe behind the expander
  - Lets software know when it is safe to read the routing tables

# SMP REPORT MANUFACTURER INFORMATION

## response frame



- Expanders don't do SCSI INQUIRY commands
- This provides a way to return three common fields from that command
- each field is an ASCII string
- T10 assigns these vendor IDs (see <http://www.t10.org> for a list)

Byte	Field(s)
0	SMP Frame Type (41h)
1	Function (01h)
2	Function Result
3 to 11	Reserved
12 to 19	Vendor Identification
20 to 35	Product Identification
36 to 39	Product Revision Level
40 to 59	Vendor Specific
28 to 31	CRC



# SMP application layer – phy-specific read functions

# SMP DISCOVER, REPORT PHY ERROR LOG, REPORT PHY SATA request frames



- For the per-phy read-only functions, the **SMP Request** frame contains the function number and the phy identifier

Byte	Field(s)
0	SMP Frame Type (40h)
1	Function (10h, 11h, or 12h)
2 to 8	Reserved
9	Phy Identifier
10 to 11	Reserved
12 to 15	CRC

# SMP DISCOVER response frame part 1



- Returns a lot of device about a phy and its current state

Byte	Field(s)	
0	SMP Frame Type (41h)	
1	Function (01h)	
2	Function Result	
3 to 8	Reserved	
9	Phy Identifier	
10 to 11	Reserved	
12	Attached Device Type	Reserved
13	Reserved	Negotiated Physical Link Rate
14	Reserved	Attached Initiator Port bits
15	Reserved	Attached Target Port bits
16 to 55	...see next slide...	

# SMP DISCOVER response frame part 2



- Early in SAS development, this functionality was split over two functions
- They were merged for simplicity of implementation
- Resulted in a large function result

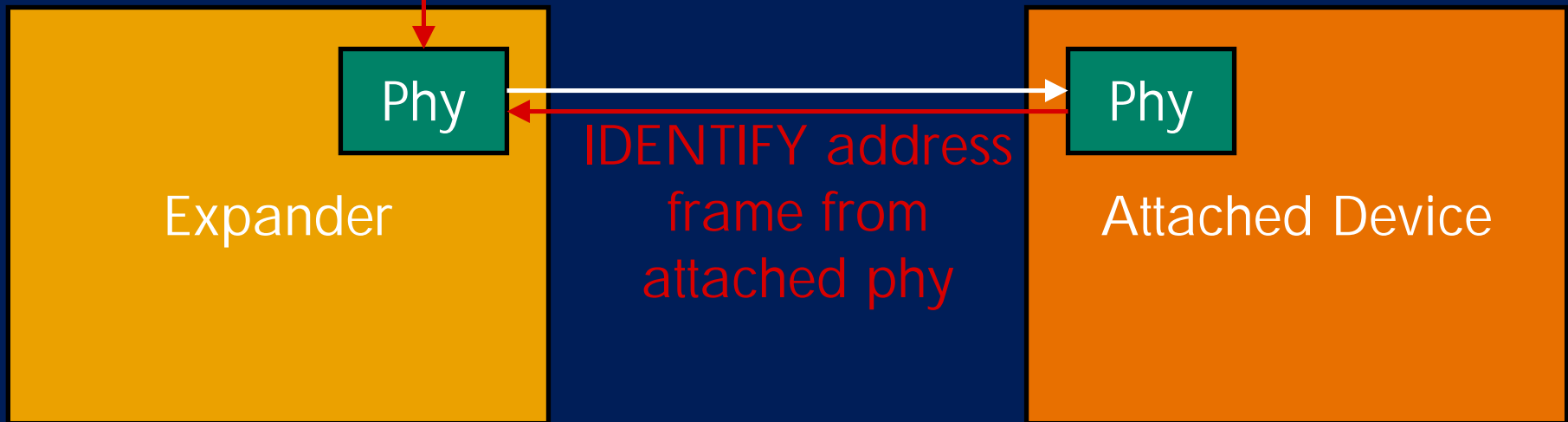
Byte	Field(s)		
0 to 15	...see previous slide...		
16 to 23	SAS Address		
24 to 31	Attached SAS Address		
32	Attached Phy Identifier		
33 to 39	Reserved		
40	Programmed Min Rate	Hardware Min Rate	
41	Programmed Max Rate	Hardware Max Rate	
42	Phy Change Count		
43	Virtual Phy	Reserved	Partial Pathway Timeout Value
44	Reserved		Routing Attribute
45 to 49	Reserved		
50 to 51	Vendor-specific		
52 to 55	CRC		

# SMP DISCOVER response frame Attached fields



- Several fields refer to the “attached” phy
- They return information from the IDENTIFY address frame received (by the phy being queried) during the link reset sequence

DISCOVER to this expander querying about this phy



# SMP DISCOVER response frame Attached Device Type field



- Attached Device Type field
  - Indicates whether probing beyond this phy is necessary during discovery

Value	Device type
00b	End device
01b	Edge expander
10b	Fanout expander

- If an end device is indicated, can start opening SSP, STP, or SMP connections (as indicated by the Attached Initiator and Target bits)
- If an expander is indicated, should open SMP connections to perform further discovery of its phys

# SMP DISCOVER response frame Attached Initiator and Target fields



- **Attached SSP/STP/SMP Initiator/Target** bits
  - Any number of SSP, STP, and/or SMP initiator and/or target bits may be set
  - Set based on the incoming IDENTIFY address frame
- **Attached SATA target** bit is set if the physical link is SATA, not SAS
  - There was no IDENTIFY address frame exchange
- **Attached SATA host** bit is optional
  - SATA hosts are not supported, but expander might be able to detect them and report them

7	6	5	4	3	2	1	0
Reserved				Attached SSP Initiator	Attached SSP Initiator	Attached SMP Initiator	Attached SATA Host
Reserved				Attached SSP Target	Attached SSP Target	Attached SMP Target	Attached SATA Target

# SMP DISCOVER response frame Negotiated Physical Link Rate field



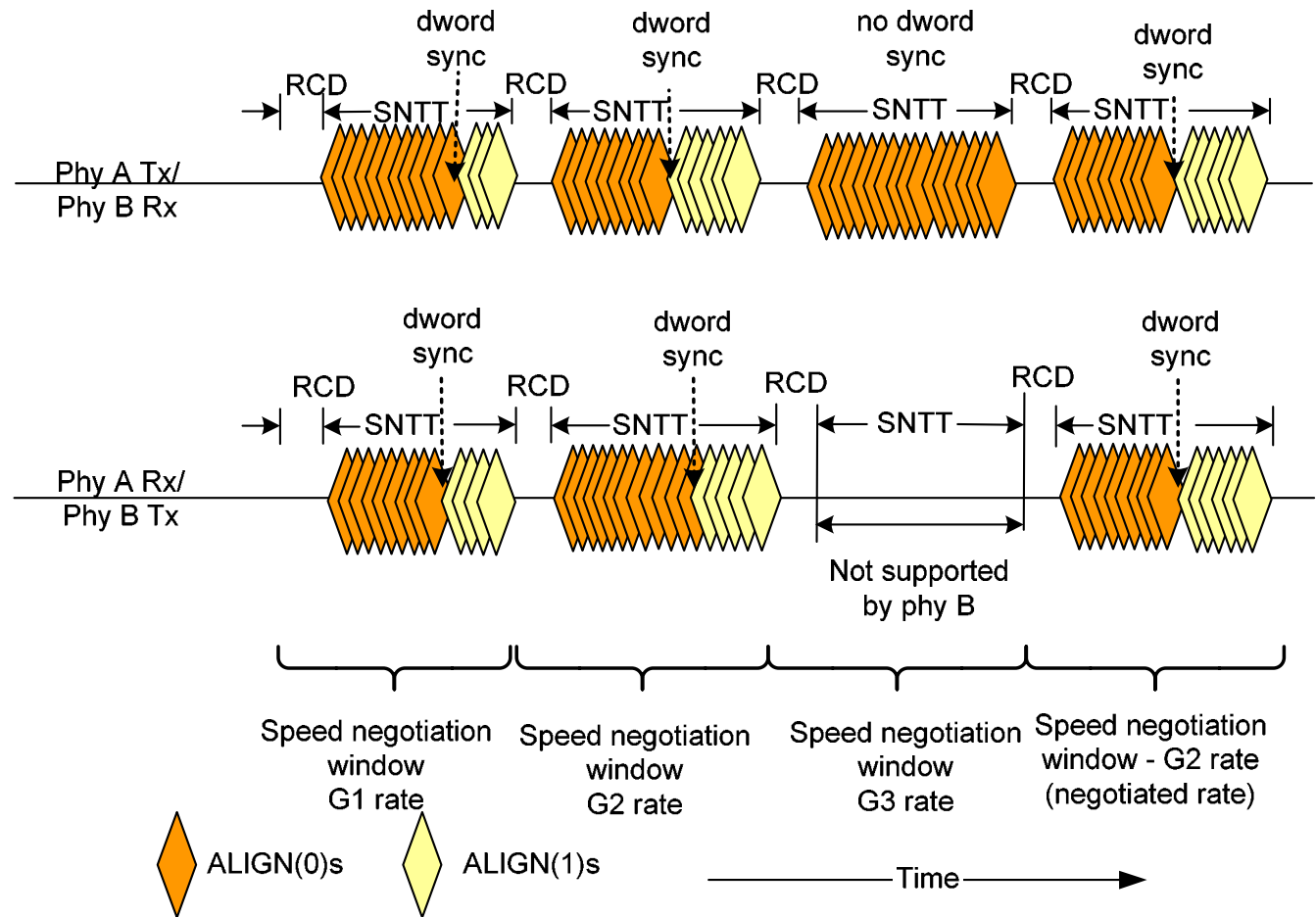
- **Negotiated Physical Link Rate** field (4 bits)
  - The current physical link rate in use by the phy
  - Also indicates phy state when it is not running at speed yet

Value	Negotiated physical link rate
0h	Unknown
1h	Phy is disabled (via the PHY CONTROL function)
2h	Phy reset problem (described later)
3h	SATA spinup hold (described later)
8h	1.5 Gbps
9h	3.0 Gbps

# Phy reset problem



- The last SAS speed negotiation window is expected to work
- If it fails, this is called a phy reset problem
- Phys must rerun the phy reset sequence from the beginning



# Phy reset problem reporting



- Reported in the DISCOVER function's **Negotiated Physical Link Rate** field
- Counted in the REPORT PHY ERROR LOG function's **Phy Reset Problem Count** field
- Resolution
  - Phy reruns the phy reset sequence and hopes for better results.
  - If it keeps failing, an initiator accessing the expander via another phy can:
    - Detect the problem with the DISCOVER and REPORT PHY ERROR LOG functions
    - Use the PHY CONTROL function's **Programmed Minimum/Maximum Physical Link Rate** fields to avoid the bad rate

# SATA spinup hold



- Serial ATA drives spinup automatically after finishing speed negotiation
  - They don't wait for a NOTIFY (ENABLE SPINUP) primitive like SAS drives
  - In an enclosure with SATA drives, they could all spinup simultaneously, stressing the power supply
- Only way to avoid this is to not finish speed negotiation automatically
  - When expander sees a SATA drive rather than SAS in the OOB sequence, it stops the phy reset sequence
  - Reports in the SMP DISCOVER function's **Negotiated Physical Link Rate** field
  - Waits for software to use the SMP PHY CONTROL function's **Phy Operation** field to request a Link Reset to proceed

# SMP DISCOVER response frame SAS Address fields



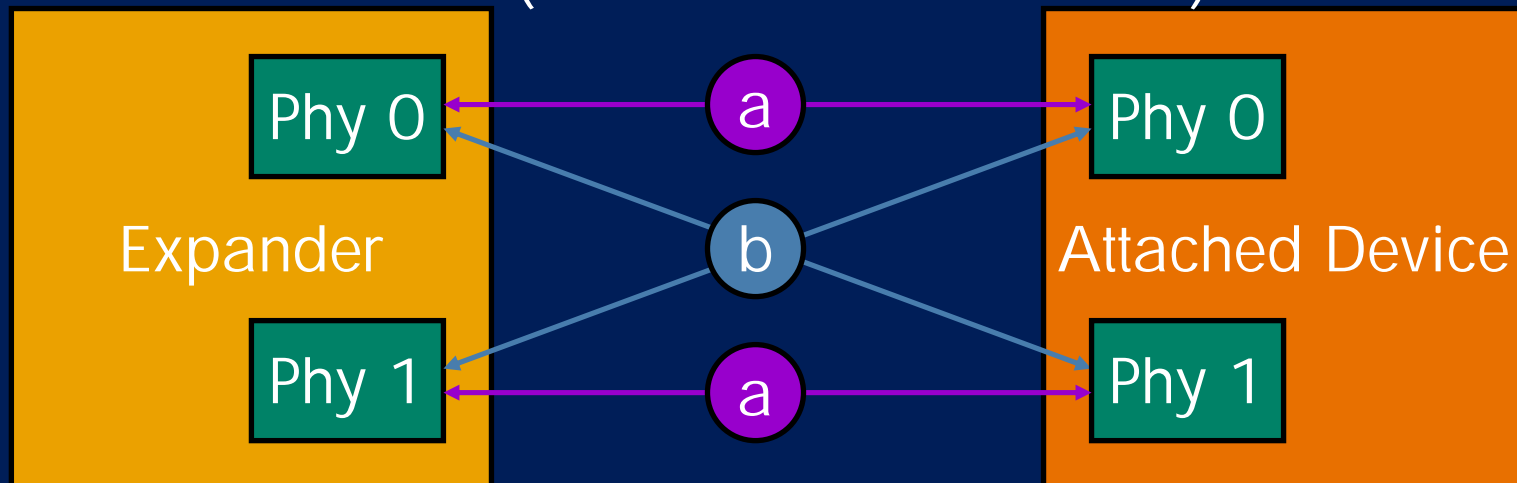
- **SAS Address** field (8 bytes)
  - The address of the expander itself
- **Attached SAS Address** field (8 bytes)
  - The address of the attached phy
  - If the **Attached Device Type** is an expander, this is an address to which another set of SMP requests is due

# SMP DISCOVER response frame Attached Phy Identifier field



- **Attached Phy Identifier** field (1 byte)
  - Returns the phy identifier of the attached phy
  - Essential for knowing the complete topology of a domain, phy-by-phy
  - Without this field, phys within wide ports would be indistinguishable
    - Yet they can have different link rates and be controlled on a per-phy basis

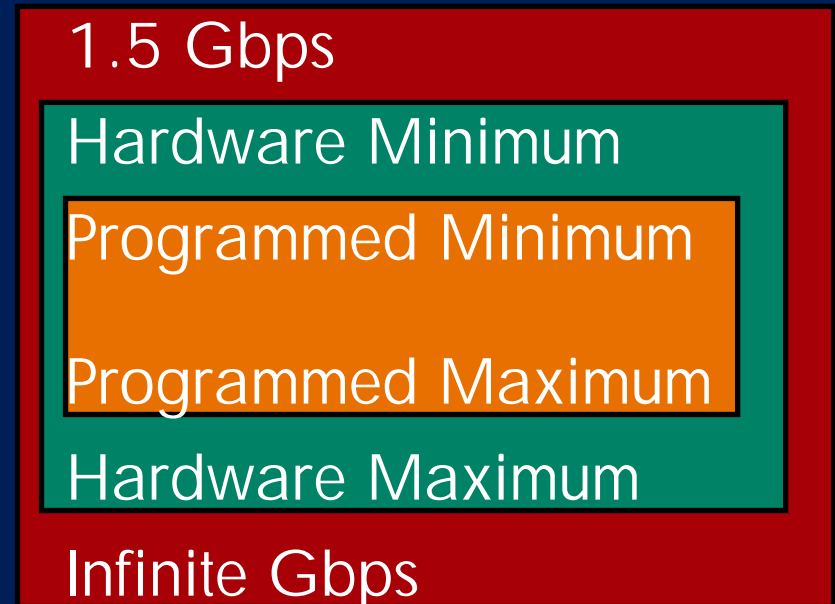
(either a or b could exist)



# SMP DISCOVER response frame Minimum/Maximum Physical Link Rate fields



- **Hardware Minimum/Maximum Physical Link Rate Fields**
  - A phy supports all physical link rates between these rates
  - No gaps allowed
  - SAS speed negotiation will not work properly if violated
- **Programmed Minimum/Maximum Physical Link Rate fields**
  - Using the PHY CONTROL function, software can restrict the range the phy will claim to support
  - Only applies until the phy loses power
  - Always a subset of the hardware range



Value	Minimum or maximum physical link rate
0h	Not programmable (for Programmed fields)
8h	1.5 Gbps
9h	3.0 Gbps

# SMP DISCOVER response frame other fields



- **Phy Change Count** field (1 byte)
  - The REPORT GENERAL function **Expander Change Count** field helps software determine the expander that caused a BROADCAST (CHANGE)
  - This field narrows it down to the phy
  - Same problem with wrapping (at FFh -> 00h)
- **Virtual Phy** bit
  - The phy is internal, not external
- **Partial Pathway Timeout Value** field
  - Deadlock detection time
- **Routing Attribute** field

Value	Routing attribute
0h	Direct
8h	Table
9h	Subtractive

# SMP REPORT PHY ERROR LOG response frame



- Returns error counts for the phy
- 4 specific low-level types of errors are counted
- Counters saturate at FFFFFFFFh
- SMP PHY CONTROL function used to clear the counters

Byte	Field(s)
0	SMP Frame Type (41h)
1	Function (11h)
2	Function Result
3 to 8	Reserved
9	Phy Identifier
10 to 11	Reserved
12 to 15	Invalid Dword Count
16 to 19	Invalid Disparity Error Count
20 to 23	Invalid Dword Synchronization Count
24 to 27	Phy Reset Problem Count
28 to 31	CRC

- Note that frame CRC errors are not included
  - Expanders do not examine frame contents (except for SMP frames addressed to themselves)

# SMP REPORT PHY SATA response frame



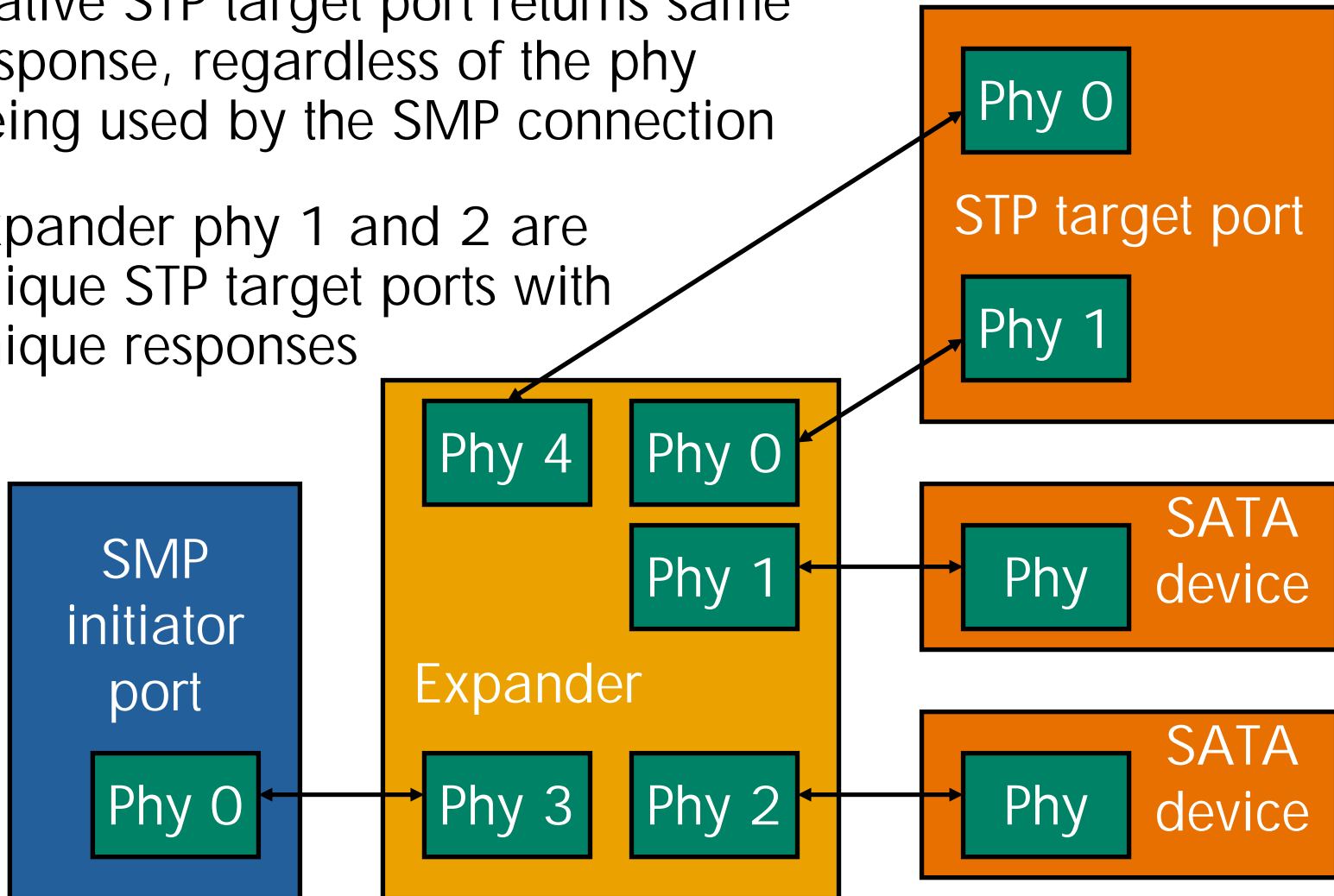
- Returns information about a phy that is attached to a SATA device
  - Each phy in an expander has its own response
- Also used by native STP targets
  - Same response via queries to any phy in the STP target port

Byte	Field(s)		
0	SMP Frame Type (41h)		
1	Function (12h)		
2	Function Result		
3 to 8	Reserved		
9	Phy Identifier		
10	Reserved		
11	Reserved	Affiliations Supported	Affiliation Valid
12 to 15	Reserved		
16 to 23	STP SAS Address		
24 to 43	Register Device to Host FIS		
44 to 47	Reserved		
48 to 55	Affiliated STP Initiator SAS Address		
56 to 59	CRC		

# SMP REPORT PHY SATA examples



- Native STP target port returns same response, regardless of the phy being used by the SMP connection
- Expander phy 1 and 2 are unique STP target ports with unique responses



# SMP REPORT PHY SATA response frame Affiliation fields



- **Affiliations Supported** bit
  - Indicates if this STP target port supports affiliations (allowing only one STP initiator at a time)
  - A native STP target port knows about SAS connections and knows which initiator is opening it
    - Can implement a SATA shadow task file register set per initiator and allow them to access concurrently
    - Use OPEN\_REJECT (RETRY) to temporarily hold off initiators
    - Select proper initiator for reconnections
    - Manage queue tags
- **Affiliation Valid** bit and **Affiliated STP Initiator SAS Address** field (8 bytes)
  - an affiliation is currently established by the identified STP initiator port

# SMP REPORT PHY SATA response frame Device-to-Host Register FIS field



- **Device-to-Host Register FIS** field (20 bytes)
  - After power on, a SATA device delivers a Device-to-Host Register FIS containing the power on signature (i.e. results of running diagnostics)
    - In SAS, no STP initiator to send it to (yet)
    - Could refuse to accept the FIS until an initiator opens a connection
      - Not helpful for multiple initiators
    - Instead, expander accepts the FIS and stores it here for access by any initiator
  - Be careful of endianness – uses a byte-by-byte copy of the SATA FIS, not a dword-by-dword copy
    - Byte 24 = FIS byte 0 (the **FIS Type** field)
    - Byte 25 = FIS byte 1 (includes the **PM Port** field)
    - ...
  - First byte is set to 00h whenever dword sync is lost to the device

# SMP REPORT ROUTE TABLE request frame



- Same as the per-phy read-only functions, but adds an **Expander Route Index** field
- Returns the expander route table entry for the specified **Phy Identifier** and **Expander Route Index**

Byte	Field(s)
0	SMP Frame Type (40h)
1	Function (13h)
2 to 5	Reserved
6 to 7	Expander Route Index
8	Reserved
9	Phy Identifier
10 to 11	Reserved
12 to 15	CRC

Expander Route Index	Phy Identifier			
	0	1	...	n
0	Entry	Entry		Entry
1	Entry	Entry		Entry
...				
M	Entry	Entry	...	Entry

- Must run this function many times to retrieve the whole table

# SMP REPORT ROUTE INFORMATION response frame



- Each entry contains a **Routed SAS address** field (8 bytes) and a **Disabled** bit
- Each entry is disabled after power on
- Entries enabled when written with **CONFIGURE ROUTE INFORMATION** function or by self-configuring expander

Byte	Field(s)
0	SMP Frame Type (41h)
1	Function (12h)
2	Function Result
3 to 5	Reserved
6 to 7	Expander Route Index
8	Reserved
9	Phy Identifier
10	Reserved
11	Disabled   Reserved
12 to 15	Reserved
16 to 23	Routed SAS Address
24 to 39	Reserved
40 to 43	CRC

# SMP application layer – write functions

# SMP CONFIGURE ROUTE INFORMATION request frame



- Used to write an expander route table entry
- Same fields as in REPORT ROUTE INFORMATION request and response frames
- Lost of reserved bits to match size of PHY CONTROL request frame

Byte	Field(s)	
0	SMP Frame Type (40h)	
1	Function (90h)	
2 to 5	Reserved	
6 to 7	Expander Route Index	
8	Reserved	
9	Phy Identifier	
10 to 11	Reserved	
12	Disable	Reserved
13 to 15	Reserved	
16 to 23	Routed SAS Address	
24 to 39	Reserved	
40 to 43	CRC	

# SMP CONFIGURE ROUTE INFORMATION response frame



- Simply indicates success or failure

Byte	Field(s)
0	SMP Frame Type (41h)
1	Function (90h)
2	Function Result
3	Reserved
4 to 7	CRC

# SMP PHY CONTROL request frame



- Change the phy state

Byte	Field(s)	
0	SMP Frame Type (40h)	
1	Function (91h)	
2 to 5	Reserved	
6 to 7	Expander Route Index	
8	Reserved	
9	Phy Identifier	
10	Phy Operation	
11	Reserved	Update PPTV
12 to 31	Reserved	
32	Programmed Min Rate	Reserved
33	Programmed Max Rate	Reserved
34 to 35	Reserved	
36	Reserved	PPTV
37 to 39	Reserved	
40 to 43	CRC	

# SMP PHY CONTROL response frame



- Simply indicates success or failure
- Same format as CONFIGURE ROUTE INFORMATION response frame

Byte	Field(s)
0	SMP Frame Type (41h)
1	Function (91h)
2	Function Result
3	Reserved
4 to 7	CRC

# SMP PHY CONTROL request frame PPTV fields



- Update PPTV bit and PPTV field (4 bits)
  - Writes the Partial Pathway Timeout Value
  - 0 to 15 microseconds
    - 0  $\mu$ s means immediately back off
    - 7  $\mu$ s is the recommended default
  - Used for deadlock detection
  - If a partial request waits on another partial request for this long, one of them is backed off with OPEN\_REJECT (PATHWAY BLOCKED)

# SMP PHY CONTROL request frame Programmed Minimum/Maximum Physical Link Rate fields



- Programmed Minimum/Maximum Physical Link Rate fields (4 bits each)
  - Can limit the phy to a smaller range of physical link rates during speed negotiation

Value	Meaning
0h	No change
8h	1.5 Gbps
9h	3.0 Gbps

- Phys must support a contiguous range
- Set Minimum = Maximum to only allow one physical link rate
- Note this controls the physical link rate not the connection rate



# SMP PHY CONTROL request frame Phy Operation field



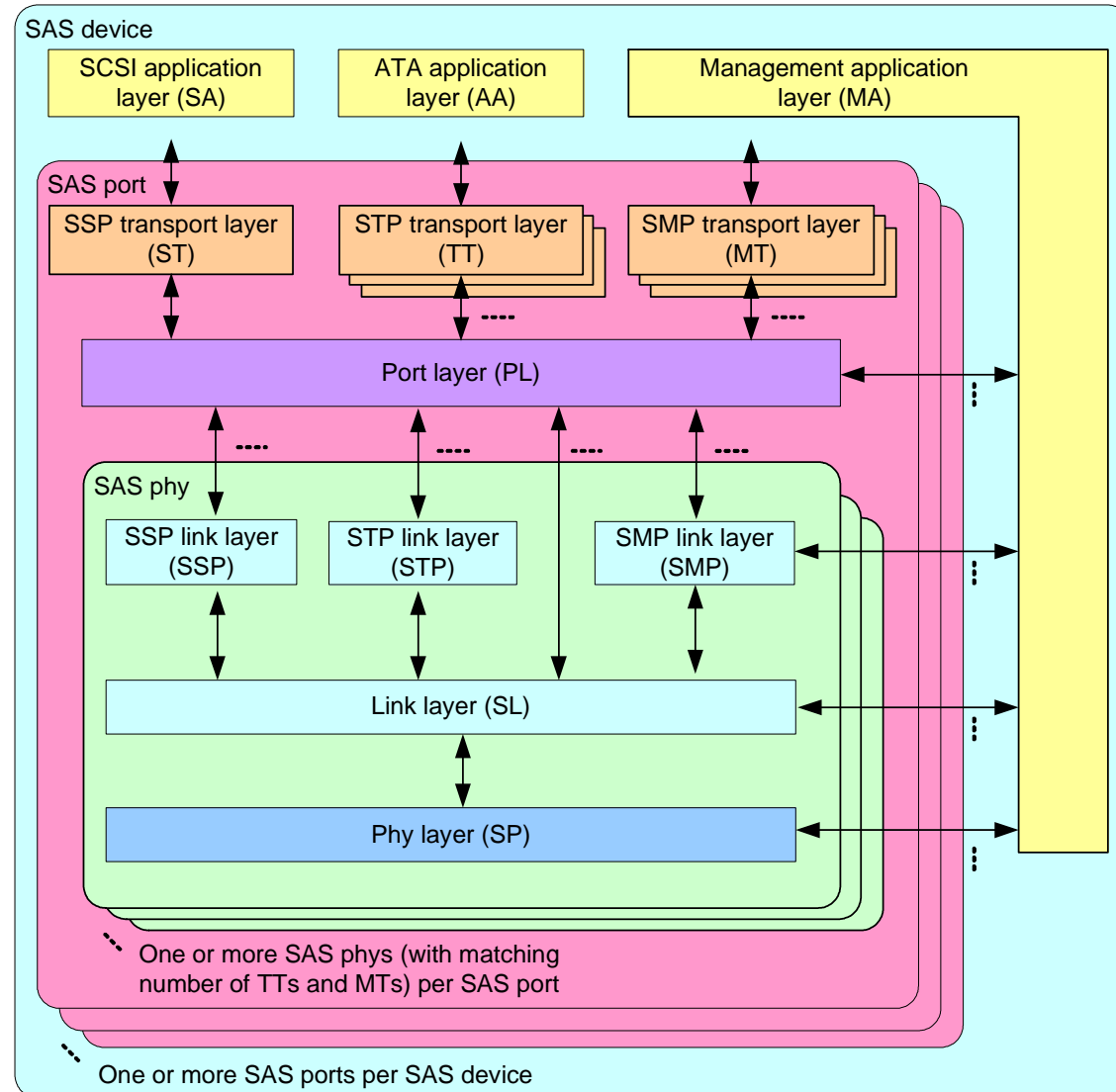
Value	Phy Operation	Description
00h	NOP	No operation.
01h	Link Reset	Run a link reset sequence: Phy reset sequence (OOB + speed negotiation) + identification sequence Maintains any affiliation.
02h	Hard Reset	Run a link reset sequence: Phy reset sequence (OOB + speed negotiation) + hard reset sequence (send HARD_RESET primitives) + a second phy reset sequence + identification sequence Clears any affiliation.
03h	Disable	Disable the phy.
05h	Clear Error Log	Clear the error logs reported with REPORT PHY ERROR LOG
06h	Clear Affiliation	Clear an affiliation if present (from the same initiator sending this SMP function).

# Port layer

# Port layer overview



- Transport layer defines frame contents
- Link layer manages connections and transmits frames
- The Port layer sits between the two
  - Part of the port object
  - Chooses the phy on which to send each frame
  - Tells the link layer of each phy in the port when to open and close connections



# Port layer timers



- Port layer implements several timers
  - **I\_T Nexus Loss** timer (Protocol-Specific Port mode page)
    - If a connection request is rejected for certain reasons, eventually notifies the application layer that the destination has disappeared
  - **Arbitration Wait Time** timer (OPEN address frames)
    - If a connection request is rejected for certain reasons, updates the AWT in the next Open Address Frame as appropriate
  - **Bus Inactivity Time Limit** timer (Disconnect-Reconnect mode page)
    - Measures time between sending frames in a connection
  - **Maximum Connect Time Limit** timer (Disconnect-Reconnect mode page)
    - Measures time that each connection is open

# Port layer state machines

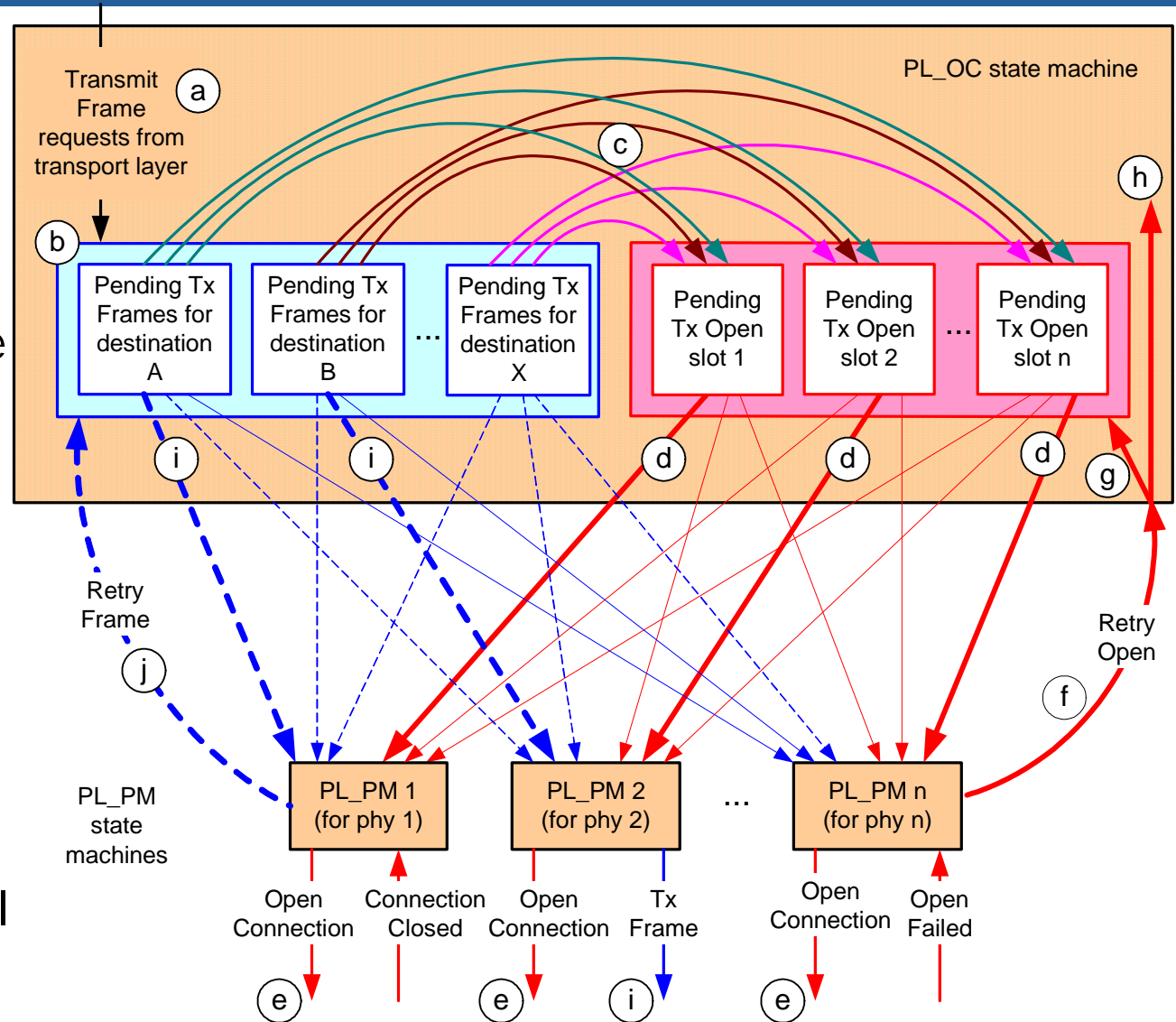


- Same state machines for initiator ports and target ports
- **PL\_OC (overall control)** “state machine”
  - One per port
  - **Pending Frames** pool
    - Pool of frames per destination (I\_T nexus)
    - Note that frames for different I\_T\_L and I\_T\_L\_Q share the same pool
  - **Pending Tx Opens** pool
    - Pool of connection requests to make (n slots for n phys)
  - Requests that PL\_PM open and close connections
  - Routes frames from the Pending Frames pool to the phys when connections are established
- **PL\_PM (phy manager)** state machine
  - One per phy (within the port)
  - Active whenever a connection is established
  - Sits in PL\_PM3:Connected state during a connection

# Port layer state machine usage



- A) Transport layer provides frames
- C) PL\_OC constantly looks at frame pool to see when Tx Opens are needed
- D) Open connection
- F) Problems opening
- I) Send frames to phys with open connections
- J) Connection lost; return frame to pool



# Port layer and wide ports



- Wide ports require the most complicated rules
  - More than one SSP connection can be open from this wide port to another wide port
  - OK to send two COMMAND frames simultaneously
  - During data transfers, must keep all pipelined data frames (in one direction) together until ACK/NAK count goes to zero
    - Then, can walk to a different connection and continue
    - Only one DMA engine active per command (per direction)
    - But, its phy can change
  - After sending a COMMAND on one phy, must wait for its ACK or NAK before sending a task management function ABORT TASK referencing it on another phy
    - Otherwise the context for the ACK is lost



# Wrap up

# Serial Attached SCSI tutorials



- General overview (~2 hours)
- Detailed multi-part tutorial (~3 days to present):
  - Architecture
  - Physical layer
  - Phy layer
  - Link layer
    - Part 1) Primitives, address frames, connections
    - Part 2) Arbitration fairness, deadlocks and livelocks, rate matching, SSP, STP, and SMP frame transmission
  - Upper layers
    - Part 1) SCSI application and SSP transport layers
    - Part 2) ATA application and STP/SATA transport layers
    - Part 3) Management application and SMP transport layers, plus port layer
  - SAS SSP comparison with Fibre Channel FCP

# Key SCSI standards



- Working drafts of **SCSI** standards are available on <http://www.t10.org>
- Published through <http://www.incits.org>
  - Serial Attached SCSI
  - SCSI Architecture Model – 3 (SAM-3)
  - SCSI Primary Commands – 3 (SPC-3)
  - SCSI Block Commands – 2 (SBC-2)
  - SCSI Stream Commands – 2 (SSC-2)
  - SCSI Enclosure Services – 2 (SES-2)
- **SAS connector** specifications are available on <http://www.sffcommittee.org>
  - SFF 8482 (internal backplane/drive)
  - SFF 8470 (external 4-wide)
  - SFF 8223, 8224, 8225 (2.5", 3.5", 5.25" form factors)
  - SFF 8484 (internal 4-wide)

# Key ATA standards



- Working drafts of **ATA** standards are available on <http://www.t13.org>
  - Serial ATA 1.0a (output of private WG)
  - ATA/ATAPI-7 Volume 1 (architecture and commands)
  - ATA/ATAPI-7 Volume 3 (Serial ATA standard)
- **Serial ATA II** specifications are available on <http://www.t10.org> and <http://www.serialata.org>
  - Serial ATA II: Extensions to Serial ATA 1.0
  - Serial ATA II: Port Multiplier
  - Serial ATA II: Port Selector
  - Serial ATA II: Cables and Connectors Volume 1

# For more information



- International Committee for Information Technology Standards
  - <http://www.incits.org>
- T10 (SCSI standards)
  - <http://www.t10.org>
  - Latest SAS working draft
  - T10 reflector for developers
- T13 (ATA standards)
  - <http://www.t13.org>
  - T13 reflector for developers
- T11 (Fibre Channel standards)
  - <http://www.t11.org>
- SFF (connectors)
  - <http://www.sffcommittee.org>
- SCSI Trade Association
  - <http://www.scsita.org>
- Serial ATA Working Group
  - <http://www.serialata.org>
- SNIA (Storage Networking Industry Association)
  - <http://www.snia.org>
- Industry news
  - <http://www.infostor.com>
  - <http://www.byteandswitch.com>
  - <http://www.wwpi.com>
  - <http://searchstorage.com>
- Training
  - <http://www.knowledgetek.com>



**i n v e n t**