To: T10 Technical Committee
From: Rob Elliott, HP (elliott@hp.com)
Date: 2 November 2005
Subject: 05-322r0 SAS-2 Wide SSP target port simultaneous connection rules

Revision history
Revision 0 (2 November 2005) First revision

Related documents
sas2r00 - Serial Attached SCSI - 2 revision 00
05-381r0 - SAS-2 multiplexing (Rob Elliott, HP)

Overview
SAS-1.1 includes no guidance about how a wide SSP target port (e.g., a SAS-attached RAID controller) should decide how many connections to attempt to open at one time.

An SSP initiator port (e.g., an HBA) makes this decision based on:

a) its own port width
b) the minimum port width on the pathway between it and the target port
c) the status of other connections from itself using the same potential pathways (to avoid issuing requests that wait on other requests from the same SSP initiator port)
d) how busy it is (if one phy suffices for all the traffic, then it might not bother with more than one connection at a time even if more are possible)

An SSP target port doesn't necessarily perform topology discovery, so cannot be relied on to determine b) or set up the destination group numbers (see below) for c).

A set of STP target ports in an expander device can make this decision (since they know they are sharing a logical link, they should avoid creating congestion too). A self-configuring expander device has sufficient information to set up destination group numbers on its own.

Editor's Note 1: "logical link" is introduced by 05-381 and represents a time-division multiplexed portion of a physical link. Almost everywhere that the link layer and higher layers refer to "physical link" are changed to "logical link" by 05-381. A few new sentences in this proposal follow suit.

This proposal lets that information be provided to the target port via a shared mode page. It is based on a "destination group" concept being implemented by several HBAs.

In these HBAs, the port classifies all the destination ports into destination groups based on the shared logical links needed to access them. For example, in figure 1, the initiator port considers all the target ports to be part of a destination group sharing the 4-wide link. Targets A, B, C, D, and Y are in another destination group sharing a 2-wide link.

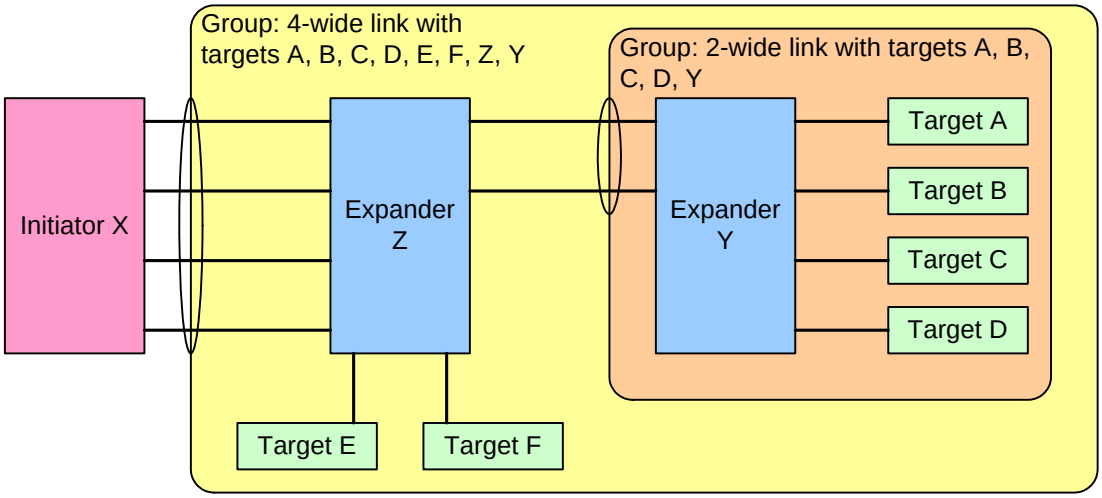

Figure 1 — Destination group example (perspective from initiator port)

Destinations may be in more than one destination group at a time (e.g., A, B, C, D, and Y are in both if the destination groups).

The HBA follows these rules:

a) Each port should not attempt to establish more connections to a destination group than the width of any of the wide links used to talk to those destination ports, so the connection request does not sit waiting on other connections involving the port to complete. Although waiting does not result in a deadlock (since the connections are expected to be regularly closed), it wastes bandwidth that could be used more productively establishing connections to other destinations.

b) Each port should keep track of the number of connections open to each destination group. When the port wants to establish a connection to a destination, it first checks to ensure that at least one logical link is available to that destination. If so, it makes the connection request; if not, it tries a destination that is not a member of a currently fully utilized destination group.

The same algorithm can be used by targets. Figure 2 shows the groups from target E's perspective (adding that target E and F are each two-wide).

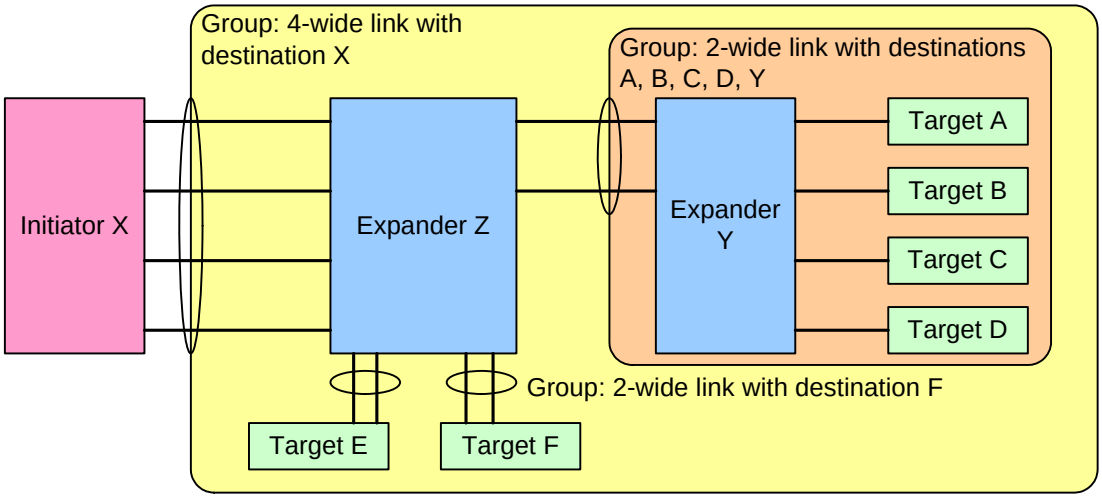

Figure 2 — Destination group example (target port E perspective)

Although the group holding initiator port X is 4-wide, target E cannot open more than two connections at a time because it is 2-wide. The target follows the same rules described above.

For both targets and initiators, the scheme is imperfect, since an incoming connection request could still arrive that consumes another logical link in the wide link after the port already sent its outgoing request on one of the logical links. In this case, the port is not advised to try to cancel its outgoing connection request with a BREAK; it just lets it sit and congest. This impairs performance but does not lead to starvation, livelock, or deadlock.

Also, the scheme does not account for other ports making connection requests in the fabric - the basic SAS rules requiring connections to be regularly closed must suffice. In a closed environment like dual RAID controllers sharing a set of disk drives, an initiator may choose to limit its requests based on knowledge of what the other initiators are doing. Targets are unlikely to have that coordination ability.

A new mode page and SMP function are proposed containing:

a) a list of SSP initiator port SAS addresses with up to 3 destination group numbers for each. Targets may support more than 3 destination group numbers.
b) a list of the maximum simultaneous connections available for each destination group
c) the SSTP initiator port SAS address that established the policy

The target would then implement the destination group algorithm and avoid congesting the fabric with its own requests. If a destination SSP port is not in the list, then accesses to it are not constrained.

Editor's Note 2: One problem is different application clients may have different algorithms to choose destination group numbers. If two application clients disagree on their assignments, whose policy should win? What if the initiator port that set a policy disappears - how do application clients behind other initiator ports know they should take control? This proposal records the SAS address of the initiator port that write the destination group information, so some other can take over ownership if that SAS address leaves the domain. No other coordination mechanisms are provided. If initiators are allowed to share a logical unit, then they are expected to know how to share the settings for this mode page, so this is not that different from any other mode page sharing problem.

Suggested changes

7.12.2 Opening a connection

7.12.2.1 Connection request

The OPEN address frame (see 7.8.3) is used to open a connection from a source port to a destination port using one source phy and one destination phy.

To make a connection request, the source port shall transmit an OPEN address frame through an available phy. The source phy shall transmit idle dwords after the OPEN address frame until it receives a response or aborts the connection request with BREAK.

After transmitting an OPEN address frame, the source phy shall initialize and start a 1 ms Open Timeout timer. Whenever an AIP is received, the source phy shall reinitialize and restart the Open Timeout timer. Source phys are not required to enforce a limit on the number of AIPs received before aborting the connection request. When any connection response is received, the source phy shall reinitialize the Open Timeout timer. If the Open Timeout timer expires before a connection response is received, the source phy shall transmit BREAK to abort the connection request (see 7.12.5).

The OPEN address frame flows through expander devices onto intermediate physical links. If an expander device on the pathway is unable to forward the connection request, it returns OPEN_REJECT (see 7.12.4). If the OPEN address frame reaches the destination, it returns either OPEN_ACCEPT or OPEN_REJECT unless the OPEN address frame passed an OPEN address frame from the destination with higher arbitration priority (see 7.12.3). Rate matching shall be used on any physical links in the pathway with negotiated physical link rates that are faster than the requested connection rate (see 7.13).

7.12.2.2 Results of a connection request

After a phy transmits an OPEN address frame, it shall expect one or more of the results listed in table 1.

Table 1 — Connection Results of a connection request

Result	Description
Receive AIP	Arbitration in progress. When an expander device is trying to open a connection to the selected destination port, it returns an AIP to the source phy. The source phy shall reinitialize and restart its Open Timeout timer each time it receives an AIP. AIP is sent by an expander device while it is internally arbitrating for access to an expander port.
Receive OPEN_ACCEPT	Connection request accepted. OPEN_ACCEPT is transmitted by the destination phy.
Receive OPEN_REJECT	Connection request rejected. OPEN_REJECT is transmitted by the destination phy or by an expander device in the partial pathway. The different versions are described in 7.2.5.11. See 4.5 for I_T nexus loss handling.
Receive OPEN address frame	If AIP has been previously detected, this indicates an overriding connection request. If AIP has not yet been detected, this indicates two connection requests crossing on the physical link. Arbitration fairness determines which one wins (see 7.12.3).
Receive BREAK	The destination phy or an expander device in the partial pathway may reply with BREAK indicating the connection is not being established.
Open Timeout timer expires	The source phy shall abort the connection request by transmitting BREAK (see 7.12.5). See 4.5 for I_T nexus loss handling.

After an OPEN_REJECT (CONNECTION RATE NOT SUPPORTED) has been received by a SAS target port, the SAS target device shall set the connection rate for future requests for that I_T_L_Q nexus to:

a) the last value received in a connection request from the SAS initiator port;
b) 1,5 Gbps; or
c) the connection rate in effect when the command was received.

7.16.1 Opening an SSP connection

An SSP phy that accepts an OPEN address frame shall transmit at least one RRDY in that connection within 1 ms of transmitting an OPEN_ACCEPT. If the SSP phy is not able to grant credit, it shall respond with OPEN_REJECT (RETRY) and not accept the connection request.

An SSP port shall never attempt to establish more connections to a destination port than the destination port width or the width of the narrowest logical link on the pathway to the destination port. An SSP port should not attempt to establish more connections with a group of destination ports than the width of the narrowest logical link on the pathway to that group of destination ports. For SSP target ports, the maximum number of connections allowed for each destination group and the destination ports in each destination group are specified in the Destination Group Control mode page (see 10.7.3.2.4).

7.17.6 Opening an STP connection

If no STP connection exists when the SATA host port in an STP/SATA bridge receives a SATA_X_RDY from the attached SATA device, the STP target port in the STP/SATA bridge shall establish an STP connection to the appropriate STP initiator port before it transmits a SATA_R_RDY to the SATA device.

Wide STP initiator ports shall not request more than one connection at a time to an STP target port. Wide STP target ports shall not request more than one connection at a time to an STP initiator port.

While a wide STP target port is waiting for a response to a connection request or has established a connection to an STP initiator port, it shall:

a) reject incoming connection requests from that STP initiator port with OPEN_REJECT (RETRY); and
b) if affiliations are supported, reject incoming connection requests from other STP initiator ports with OPEN_REJECT (STP RESOURCES BUSY).

While a wide STP initiator port is waiting for a response to a connection request to an STP target port, it shall not reject an incoming connection request from that STP target port because of its outgoing connection request. It may reject incoming connection requests for other reasons (see 7.2.5.11).

If a wide STP initiator port receives an incoming connection request from an STP target port while it has a connection established with that STP target port, it shall reject the request with OPEN_REJECT (RETRY).

The first dword that an STP phy sends inside an STP connection after OPEN_ACCEPT that is not an ALIGN or NOTIFY shall be an STP primitive (e.g., SATA_SYNC).

Expander devices shall never allow their STP ports (e.g., the STP target ports in STP/SATA bridges) to attempt to establish more connections to a destination port than the destination port width or the width of the narrowest logical link on the pathway to the destination port.

Expander devices should never allow their STP ports (e.g., the STP target ports in STP/SATA bridges) to attempt to establish more connections with a group of destination ports than the width of the narrowest logical link on the pathway to that group of destination ports.

Editor's Note 3: Expander devices are expected to figure out destination group memberships on their own based on their self-configuring capability.

10.2.7.2 Protocol-Specific Port mode page

10.2.7.2.1 Protocol-Specific Port mode page overview

The Protocol-Specific Port mode page (see SPC-3) contains parameters that affect SSP target port operation. If the mode page is implemented, all logical units in SCSI target devices in SAS domains supporting the MODE SELECT or MODE SENSE commands shall implement the page.

If a SAS target device has multiple SSP target ports, changes in the short page parameters for one SSP target port should not affect other SSP target ports.

Table 2 defines the subpages of this mode page.

Table 2 — Protocol-Specific Port mode page subpages

Subpage	Description	Reference
Short page	Short format	10.2.7.2.2
Long page 00h	Not allowed	
Long page 01h	Phy Control And Discover subpage	10.2.7.2.4
Long page 02h	Destination Group Control subpage	10.2.7.2.x
Long page E0h - FEh	Vendor specific	
Long page FFh	Return all subpages for the Protocol-Specific Port mode page	SPC-3
All others	Reserved	

10.2.7.2.2 Protocol-Specific Port mode page - short format

...

10.2.7.2.3 Protocol-Specific Port mode page - Phy Control And Discover subpage

...

10.2.7.2.4 Protocol-Specific Port mode page - Destination Group Control subpage [new section]

The Destination Group Control subpage contains port-specific parameters for wide SSP ports. The mode page policy (see SPC-3) for this subpage shall be shared.

Table 3 defines the format of the subpage for SAS SSP.

Table 3 — Protocol-Specific Port mode page for SAS SSP - Destination Group Control subpage

Byte\Bit	7	6	5	4	3	2	1	0
0	PS	SPF (1b)	PAGE CODE (19h)					
1	SUBPAGE CODE (02h)							
2	(MSB)	PAGE LENGTH (n - 3)						
3								(LSB)
4	Reserved							
5	Reserved				PROTOCOL IDENTIFIER (6h)			
6	Reserved							
7	NUMBER OF DESTINATION GROUPS (m - 2)							
8	CONFIGURED BY SAS ADDRESS							
15								
Destination group maximum connections list								
16	DESTINATION GROUP 1 MAXIMUM CONNECTIONS							
	...							
m + 16	DESTINATION GROUP M MAXIMUM CONNECTIONS							
Destination group member descriptor header								
m + 17	Reserved							
m + 18								
m + 19	NUMBER OF DESTINATION GROUP MEMBER DESCRIPTORS							
Destination group member descriptor list								
m + 20	Destination group member descriptor (first)(see table 4)							
...	...							
	Destination group member descriptor (last)(see table 4)							
n								

The PARAMETERS SAVEABLE (PS) bit is defined in SPC-3.

The SUBPAGE FORMAT (SPF) bit shall be set to one to access the long format mode pages.

The PAGE CODE field shall be set to 19h.

The SUBPAGE CODE field shall be set to 02h.

The PAGE LENGTH field shall be set to the length of the remaining bytes of the subpage.

The PROTOCOL IDENTIFIER field shall be set to 6h indicating this is a SAS SSP specific mode page.

The NUMBER OF DESTINATION GROUPS field contains the number of destination groups supported by the SSP target port. This field shall not be changeable. This field shall be set to a multiple of four and shall be less than or equal to 32 (i.e., 0, 4, 8, 12, 16, 20, 24, 28, or 32). Destination groups are numbered sequentially starting with 00h.

Each DESTINATION GROUP NN MAXIMUM CONNECTIONS field contains the maximum number of connections the SSP target port shall attempt to establish to destination SAS addresses in the corresponding destination group. The number of DESTINATION GROUP NN MAXIMUM CONNECTIONS fields is specified by the NUMBER OF DESTINATION GROUPS field.

The CONFIGURED BY SAS ADDRESS field indicates the SAS address of the SSP initiator port that wrote the values in the mode page, if any. A CONFIGURED BY SAS ADDRESS field set to 00000000_00000000h indicates the SAS address is unknown or the mode page was never written. This field is ignored by the MODE SELECT command.

The NUMBER OF DESTINATION GROUP MEMBER DESCRIPTORS field contains the number of destination group member descriptors that follow. For the MODE SELECT command, the destination group member descriptor list completely replaces the previous destination group member descriptor list.

Table 4 defines the destination group member descriptor. There shall be no more than one destination group member descriptor per destination SAS address.

Table 4 — Destination group member descriptor

Byte\Bit	7	6	5	4	3	2	1	0
0	DESTINATION SAS ADDRESS							
7								
8	(group 0)	DESTINATION GROUP NUMBER BITMASK						(group 7)
11	(group 24)							(group 31)

The DESTINATION SAS ADDRESS field contains the SAS address of a destination SSP port (i.e., an SSP initiator port).

The DESTINATION GROUP NUMBER BITMASK field contains a bitmask representing the destination group number(s) of the destination port specified by the DESTINATION SAS ADDRESS field. A bit set to one means the destination port is part of the corresponding destination group. A bit set to zero means the destination port is not part of the corresponding destination group.

[end of all new section]