T10/05-370 revision 1

Date: November 08, 2005

To: T10 Committee (SCSI)

From: George Penokie (IBM/Tivoli)

Subject: SAS-2, SAM-4, SPC-4, NOTIFY (POWER LOSS EXPECTED) Fixes

Overview

Some holes were found in the 05-167r5 that need to be fixed. Only the moving of the power loss timer from Protocol-Specific Logical Unit mode page to the Protocol-Specific Port mode page is a technical change, none of the changes constitute a technical change from the intent of 05-167r5.

SAS-2 changes

The changes marked in this proposal for SAS-2 represent changes from the combination of 05-176r5 and 05-307r0.

0.0.0.1

0.0.1 Primitive summary

Table 1 defines the primitives not specific to the type of connection.

Table 1 — 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	Single
AIP (RESERVED 0)	NoConn				I	E	T	Single
AIP (RESERVED 1)	NoConn				I	E	T	Single
AIP (RESERVED 2)	NoConn				I	E	T	Single
AIP (RESERVED WAITING ON PARTIAL)	NoConn				I	E	T	Single
AIP (WAITING ON CONNECTION)	NoConn		E		I	E	T	Single
AIP (WAITING ON DEVICE)	NoConn		E		I	E	T	Single
AIP (WAITING ON PARTIAL)	NoConn		E		I	E	T	Single
ALIGN (0)	All	I	E	T	I	E	T	Single
ALIGN (1)	All	I	E	T	I	E	T	Single
ALIGN (2)	All	I	E	T	I	E	T	Single
ALIGN (3)	All	I	E	T	I	E	T	Single
BREAK	All	I	E	T	I	E	T	Redundant
BROADCAST (CHANGE)	NoConn	I	E		I	E	T	Redundant
BROADCAST (SES)	NoConn			T	I	E	T	Redundant
BROADCAST (RESERVED 1)	NoConn				I	E	T	Redundant
BROADCAST (RESERVED 2)	NoConn				I	E	T	Redundant
BROADCAST (RESERVED 3)	NoConn				I	E	T	Redundant
BROADCAST (RESERVED 4)	NoConn				I	E	T	Redundant
BROADCAST (RESERVED CHANGE 0)	NoConn				I	E	T	Redundant
BROADCAST (RESERVED CHANGE 1)	NoConn				I	E	T	Redundant
CLOSE (CLEAR AFFILIATION)	STP	I					T	Triple
CLOSE (NORMAL)	Conn	I		T	I		T	Triple
CLOSE (RESERVED 0)	Conn				I		T	Triple
CLOSE (RESERVED 1)	Conn				I		T	Triple

Table 1 — 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	
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 ~~FAILURE~~ LOSS EXPECTED)	All	I	E				T	Single
NOTIFY (RESERVED 1)	All				I	E	T	Single
NOTIFY (RESERVED 2)	All				I	E	T	Single
OPEN_ACCEPT	NoConn	I		T	I		T	Single
OPEN_REJECT (BAD DESTINATION)	NoConn		E		I		T	Single
OPEN_REJECT (CONNECTION RATE NOT SUPPORTED)	NoConn	I	E	T	I		T	Single
OPEN_REJECT (NO DESTINATION)	NoConn		E		I		T	Single
OPEN_REJECT (PATHWAY BLOCKED)	NoConn		E		I		T	Single
OPEN_REJECT (PROTOCOL NOT SUPPORTED)	NoConn	I		T	I		T	Single
OPEN_REJECT (RESERVED ABANDON 0)	NoConn				I		T	Single
OPEN_REJECT (RESERVED ABANDON 1)	NoConn				I		T	Single
OPEN_REJECT (RESERVED ABANDON 2)	NoConn				I		T	Single
OPEN_REJECT (RESERVED ABANDON 3)	NoConn				I		T	Single
OPEN_REJECT (RESERVED CONTINUE 0)	NoConn				I		T	Single
OPEN_REJECT (RESERVED CONTINUE 1)	NoConn				I		T	Single
OPEN_REJECT (RESERVED INITIALIZE 0)	NoConn				I		T	Single
OPEN_REJECT (RESERVED INITIALIZE 1)	NoConn				I		T	Single
OPEN_REJECT (RESERVED STOP 0)	NoConn				I		T	Single
OPEN_REJECT (RESERVED STOP 1)	NoConn				I		T	Single
OPEN_REJECT (RETRY)	NoConn	I		T	I		T	Single
OPEN_REJECT (STP RESOURCES BUSY)	NoConn		E	T	I			Single
OPEN_REJECT (WRONG DESTINATION)	NoConn	I		T	I		T	Single
SOAF	NoConn	I	E	T	I	E	T	Single

[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.

[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, a triple primitive sequence, or a redundant primitive sequence (see 7.2.4).

0.0.2 Primitive encodings

Table 2 defines the primitive encoding for primitives not specific to type of connection.

Table 2 — 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
BROADCAST (CHANGE)	K28.5	D04.7	D02.0	D01.4
BROADCAST (SES)	K28.5	D04.7	D07.3	D29.7
BROADCAST (RESERVED 1)	K28.5	D04.7	D01.4	D24.0
BROADCAST (RESERVED 2)	K28.5	D04.7	D04.7	D04.7
BROADCAST (RESERVED 3)	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 2 — 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 ~~FAILURE~~ 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 (RESERVED ABANDON 0)	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

0.0.2.1 NOTIFY

NOTIFY may be transmitted in place of any ALIGN (see 7.2.5.2) being transmitted for clock skew management (see 7.3), rate matching (see 7.13), or STP initiator phy throttling (see 7.17.2). Substitution of a NOTIFY may or may not affect the ALIGN rotation (i.e., the NOTIFY may take the place of one of the ALIGNs in the rotation through ALIGN (0), ALIGN (1), ALIGN (2), or ALIGN (3) or it may delay the rotation). A specific NOTIFY shall not be transmitted in more than three consecutive dwords before at least three other dwords have been transmitted.

NOTIFY shall not be forwarded through expander devices. Expander devices shall substitute an ALIGN for a NOTIFY if necessary.

SAS target devices are not required to detect every transmitted NOTIFY.

The versions of NOTIFY representing different reasons are defined in table 3.

Table 3 — NOTIFY primitives

Primitive	Description
NOTIFY (ENABLE SPINUP)	Specify to an SAS target device that it may temporarily consume additional power while transitioning into the active or idle power condition state.
NOTIFY (POWER ~~FAILURE~~ LOSS ~~-~~EXPECTED)	Specify to an SAS target device that power may be lost within a time specified by the power ~~fail~~ loss timeout timer (see 0.0.2.1.1).
NOTIFY (RESERVED 1)	Reserved.
NOTIFY (RESERVED 2)	Reserved.

NOTIFY (ENABLE SPINUP) is transmitted by a SAS initiator port or expander port and is used to specify to an SAS target device that it may temporarily consume additional power (e.g., while spinning-up rotating media) while transitioning into the active or idle power condition state. The length of time the SAS target device consumes additional power and the amount of additional power is vendor specific. NOTIFY (ENABLE SPINUP) shall interact with the device's power condition state transitions, controlled by the Power Conditions mode page (see SPC-3) and/or the START STOP UNIT command (see SBC-2), as described in 10.2.10.

SAS initiator devices and expander devices shall use NOTIFY (ENABLE SPINUP) while attached to SAS target devices (i.e., devices that report SSP target support in their IDENTIFY address frames). They shall transmit one NOTIFY (ENABLE SPINUP) after power on when the enclosure is ready for initial spin-up. After the initial NOTIFY (ENABLE SPINUP), they shall transmit NOTIFY (ENABLE SPINUP) periodically. Otherwise, the selection of when and how often to transmit NOTIFY (ENABLE SPINUP) is outside the scope of this standard.

NOTE 1 - The SAS initiator device or expander device uses NOTIFY (ENABLE SPINUP) to avoid exceeding enclosure power supply capabilities during spin-up of multiple SAS target devices. It may choose to rotate transmitting NOTIFY (ENABLE SPINUP) across all of its ports, distributing it to N ports at a time if the enclosure power supply is capable of powering N SAS target devices spinning up at a time. An expander device may allow this timing to be configured by a NVROM programming with enclosure-specific sequencing patterns, or may employ more complex, dynamic interaction with the enclosure power supply.

NOTE 2 - NOTIFY (ENABLE SPINUP) should be transmitted as frequently as possible to avoid incurring application layer timeouts.

I_T nexus loss, logical unit reset, and hard reset shall not cause a SAS target device to spin-up automatically on receipt of NOTIFY (ENABLE SPINUP).

SAS target devices with multiple SAS target ports shall honor NOTIFY (ENABLE SPINUP) from all SAS target ports equivalently (e.g., NOTIFY (ENABLE SPINUP) received on SAS target port A serves as a wakeup for a START STOP UNIT command received through SAS target port B).

NOTIFY (POWER ~~FAILURE~~ LOSS ~~-~~EXPECTED) is transmitted by a SAS initiator port or expander port and is used to specify to a SAS target device that a power ~~failure~~ loss may occur within a time defined in the power ~~failure~~ loss timeout field of the Protocol-Specific Logical Unit mode page (see 0.0.2.1.1).

The NOTIFY (POWER ~~FAILURE~~ LOSS EXPECTED) shall be transmitted at least three times by the SAS initiator port or expander port.

If a SAS target device supports NOTIFY (POWER ~~FAILURE~~ LOSS EXPECTED) and receives NOTIFY (POWER ~~FAILURE~~ LOSS EXPECTED) on an SSP target port, then the device server for each logical unit to which ~~the~~ that SSP target port has access shall:

1) stop writing data to the media on a block boundary (e.g., all write activity shall continue until a block boundary is reached then all writing shall stop); ~~and~~

2) clear ~~the~~ all task sets (i.e., the device server acts as if it has received a CLEAR TASK SET task management function (see SAM-4) for each task set); and
3) establish a unit attention condition for the initiator port associated with every I_T nexus with the additional sense code set to ~~POWER FAIL EVENT COMMANDS CLEARED~~ COMMANDS CLEARED BY POWER LOSS NOTIFICATION.

If a target device supports NOTIFY (POWER ~~FAILURE~~ LOSS EXPECTED) and receives NOTIFY (POWER ~~FAILURE~~ LOSS EXPECTED) on an SSP port, then within 1 ms after the NOTIFY (POWER ~~FAILURE~~ LOSS EXPECTED) is received, each SAS phy within the target device shall:

a) if there is an SSP connection, then transmit a BREAK on that connection; and
b) response to SSP connection requests with OPEN_REJECT (RETRY)s until the power ~~fail~~ loss timeout timer expires (see 0.0.2.1.1) or power is lost.

NOTE 3 - A SAS target device may transmit a BREAK on an open connection and begin transmitting OPEN_REJECT (RETRY)s as soon as a NOTIFY (POWER ~~FAILURE~~ LOSS EXPECTED) is received.

If any frames are received by a SAS target device after a NOTIFY (POWER ~~FAILURE~~ LOSS EXPECTED) is received and before a connection is closed (i.e., by transmitting a BREAK in response to the NOTIFY (POWER ~~FAILURE~~ LOSS EXPECTED)), then the SAS target device shall discard the received frames.

~~If the power fail timeout timer expires, then each device server shall establish a unit attention condition for the initiator port associated with every I_T nexus with the additional sense code set to POWER FAIL EVENT COMMANDS CLEARED COMMANDS CLEARED BY POWER FAILURE NOTIFICATION~~. After power on the power loss timeout timer shall be initialized and stopped until a NOTIFY (POWER ~~FAILURE~~ LOSS EXPECTED) is received.

NOTIFY (RESERVED 1), and NOTIFY (RESERVED 2) shall be ignored by all devices.

SPC-4 changes

Add in a new ASC/ASCQ of 2Fh 01h COMMANDS CLEARED BY POWER ~~FAILURE~~ LOSS NOTIFICATION.

SAM-4 changes

0.0.3 When a SCSI initiator port aborts tasks received on other I_T nexuses

When a SCSI initiator port causes task(s) received on other I_T nexus(es) to be aborted, the SCSI initiator port associated with every other I_T nexus shall be notified that the task(s) have been aborted. The method of notification shall depend on the setting of the TAS bit in the Control mode page (see SPC-3) that applies to the SCSI initiator port(s) associated with the other I_T nexus(es).

If the TAS bit is set to zero, the method of notification shall be a unit attention condition. The additional sense code for the unit attention condition depends on the action that caused the task(s) to be aborted as described in table 4 (see 0.3.1).

If the TAS bit is set to one, the method of notification shall be the termination of each aborted task with a TASK ABORTED status. ~~The COMMANDS CLEARED BY ANOTHER INITIATOR unit attention condition shall not be established, however, the establishment of any other applicable unit attention condition shall not be affected.~~ A unit attention condition containing an additional sense code whose ADDITIONAL SENSE CODE field contains 2Fh (e.g., COMMANDS CLEARED BY ANOTHER INITIATOR, or COMMANDS CLEARED BY POWER LOSS NOTIFICATION), shall not be established. The establishment of any other applicable unit attention condition shall not be affected.

When a logical unit is aborting one or more tasks received on an I_T nexus using the TASK ABORTED status it should complete all of those tasks before entering additional tasks received on that I_T nexus into the task set.

0.1 Task and command lifetimes

This subclause specifies the events delimiting the beginning and end (i.e., lifetime) of a task or pending command from the viewpoint of the device server and application client.

The device server shall create a task upon receiving a **SCSI Command Received** indication unless the command represents a continuation of a linked command as described in 5.1.

The task shall exist until:

a) The device server sends a SCSI transport protocol service response for the task of TASK COMPLETE; or
b) The task is aborted as described in 0.3.

The application client maintains an application client task to interact with the task from the time the **Send SCSI Command** SCSI transport protocol service request is invoked until it receives one of the following SCSI target device responses:

a) A service response of TASK COMPLETE for that task;
b) Notification of a unit attention condition with one of the following additional sense codes;
   A) ~~COMMANDS CLEARED BY ANOTHER INITIATOR, if in reference to the task set containing the task~~ Any additional sense code whose ADDITIONAL SENSE CODE field contains 2Fh (e.g., COMMANDS CLEARED BY ANOTHER INITIATOR, or COMMANDS CLEARED BY POWER LOSS NOTIFICATION), if in reference to the task set containing the task; or
   B) Any additional sense code whose ADDITIONAL SENSE CODE field contains 29h (e.g., POWER ON, RESET, OR BUS DEVICE RESET OCCURRED; POWER ON OCCURRED; SCSI BUS RESET OCCURRED; BUS DEVICE RESET FUNCTION OCCURRED; DEVICE INTERNAL RESET; or I_T NEXUS LOSS OCCURRED);
c) Notification that the task manager has detected the use of a duplicate I_T_L_Q nexus (see 5.8.3);
d) A service response of FUNCTION COMPLETE following an ABORT TASK task management function directed to the specified task;
e) A service response of FUNCTION COMPLETE following an ABORT TASK SET or a CLEAR TASK SET task management function directed to the task set containing that task;
f) A service response of FUNCTION COMPLETE following an I_T NEXUS RESET task management function delivered on the I_T nexus used to deliver that task; or
g) A service response of FUNCTION COMPLETE in response to a LOGICAL UNIT RESET task management function directed to the logical unit.

0.2 Task management function lifetime

The task manager shall create a task management function upon receiving a Task Management Request Received indication (see 7.10). The task management function shall exist until:

a) the task manager sends a SCSI transport protocol service response for the task management function;
b) an I_T nexus loss (see 6.3.4);
c) a logical unit reset (see 6.3.3);
d) a hard reset (see 6.3.2); or
e) a power on condition (see 6.3.1).

The application client maintains an application client task to interact with the task management function from the time the **Send Task Management Request** SCSI transport protocol service request is invoked until it receives one of the following SCSI target device responses:

a) A service response of FUNCTION COMPLETE, FUNCTION SUCCEEDED, FUNCTION REJECTED, or SERVICE DELIVERY OR TARGET FAILURE is received for that task management function; ~~or~~
b) Notification of a unit attention condition with any additional sense code whose ADDITIONAL SENSE CODE field contains 29h (e.g., POWER ON, RESET, OR BUS DEVICE RESET OCCURRED; POWER ON

OCCURRED; SCSI BUS RESET OCCURRED; BUS DEVICE RESET FUNCTION OCCURRED; DEVICE INTERNAL RESET; or I_T NEXUS LOSS OCCURRED); or

c) Notification of a unit attention condition with an additional sense code of COMMANDS CLEARED BY POWER LOSS NOTIFICATION.

NOTE 4 - The names of the unit attention conditions listed in the subclause (e.g., SCSI BUS RESET OCCURRED) are based on usage in previous versions of this standard. The use of these unit attention condition names is not to be interpreted as a description of how the unit attention conditions are represented by any given SCSI transport protocol.

0.3 Aborting tasks

0.3.1 Mechanisms that cause tasks to be aborted

A task is aborted when an event or SCSI initiator device action causes termination of the task prior to its successful completion.

The following events cause a task or several tasks to be aborted:

a) The return of an **Execute Command** service response of SERVICE DELIVERY OR TARGET FAILURE as described in 5.1;
b) An I_T nexus loss (see 6.3.4);
c) A logical unit reset (see 6.3.3);
d) A hard reset (see 6.3.2); ~~or~~
e) A power on condition (see 6.3.1); or
f) SCSI transport protocol specific conditions.

An action transmitted via one I_T nexus may abort task(s) received on that I_T nexus and/or task(s) received on other I_T nexuses.

The following actions affect only the task(s) received on the I_T nexus on which the action is transmitted:

a) Completion of an ABORT TASK task management function directed to the specified task;
b) Completion of an ABORT TASK SET task management function under the conditions specified in 7.3;
c) Completion of an I_T NEXUS RESET task management function; or
d) Completion of a command with a CHECK CONDITION status, without establishing an ACA condition (see 5.8.1.3) or establishing an ACA condition (see 5.8.2.2), while the Control mode page (see SPC-3) contains fields that are set as follows:
   A) The QERR field set to 01b and the TST field set to 001b; or
   B) The QERR field set to 11b.

The actions shown in table 4 affect the task(s) received on the I_T nexus on which the action is transmitted and/or task(s) received on other I_T nexuses.

Table 4 — Actions that affect task(s) received on this or other I_T nexuses

Action	Unit attention additional sense code, if any (see 0.0.3)
Completion of a CLEAR TASK SET task management function referencing the task set containing the specified task	COMMANDS CLEARED BY ANOTHER INITIATOR
Completion of a command with a CHECK CONDITION status, with or without establishing an ACA condition, and the QERR field was set to 01b and the TST field was set to 000b in the Control mode page (see SPC-3)	COMMANDS CLEARED BY ANOTHER INITIATOR
Completion of a PERSISTENT RESERVE OUT command with a PREEMPT AND ABORT service action with a reservation key that is associated with the I_T nexus on which the task was received (see SPC-3)	COMMANDS CLEARED BY ANOTHER INITIATOR
Completion of a LOGICAL UNIT RESET task management function (see 7.7) directed to the logical unit	BUS DEVICE RESET FUNCTION OCCURRED
SCSI transport protocol specific conditions	As defined by the applicable SCSI transport protocol standard

If one or more tasks are cleared or aborted, the affected tasks are also cleared from the initiator ports in a manner that is outside the scope of this standard.