

To: T10 Technical Committee
 From: Rob Elliott, HP (elliott@hp.com)
 Date: 17 July 2007
 Subject: 06-026r4 SAM-4 TASK ABORTED status clarifications

Revision history

Revision 0 (3 May 2007) First revision

Revision 1 (10 May 2007) Incorporated comments from May 2007 CAP WG

Revision 2 (22 June 2007) Per request from Bill Galloway (Pivot3), changed so a SCSI device conditions caused by a particular I_T nexus like LOGICAL UNIT RESET does cause TASK ABORTED status.

Devices that cannot hold off handling the reset can simply not claim support for TASK ABORTED status.

Revision 3 (27 June 2007) Incorporated suggestions from Charles Binford (Sun)

Revision 4 (17 July 2007) Incorporated suggestions from July 2007 CAP WG

Related documents

sam4r09 - SCSI Architecture Model - 4 (SAM-4) revision 9

Overview

Based on the inconsistent implementations by target vendors, it is apparently not clear whether TASK ABORTED status is sent when tasks are aborted by a LOGICAL UNIT RESET task management function.

SPC-4 says the TAS bit set to one means "tasks aborted by the actions of an I_T nexus other than the I_T nexus on which the command was received shall be terminated with a TASK ABORTED status." That implies a LOGICAL UNIT RESET task management function does generate TASK ABORTED status values, since it is sent on a particular I_T nexus.

- a) Argument in favor of returning TASK ABORTED status: Conditions that were detected by a device server on only one of its target ports are not likely to be known by initiator ports attached to different SCSI domains in which the other target ports are attached. This status provides them notification that another initiator port caused their tasks to be aborted.
- b) Argument opposed to returning TASK ABORTED status: the device server might not be able to fend off the reset while it doles out TASK ABORTED statuses for each queued command. It is difficult to defer handling of power loss or hard reset. It is more conceivable to defer a LOGICAL UNIT RESET, which might involve firmware processing. However, an application could be sending LOGICAL UNIT RESET because the device server appears hung; asking the device server to send more status values is unlikely to unhang it.

Numerous changes are proposed to the "Aborting tasks" section that should make the rules clearer and more comprehensive.

This revision proposes clarifying that SCSI device conditions that abort tasks (and the task management functions that cause them - LOGICAL UNIT RESET and I_T NEXUS RESET) do cause TASK ABORTED status, along with commands and task management functions that abort tasks without causing SCSI device conditions (like ABORT TASK, ABORT TASK SET, and CLEAR TASK SET).

Some transport protocols like iSCSI still define a task management function called TARGET RESET that invokes a hard reset, although it has been obsolete since SAM-2. These protocols are expected to return TASK ABORTED status, since the hard reset condition was caused by one particular I_T nexus.

Suggested changes to SAM-4

5.6.1 Status codes

...

TASK ABORTED. This status shall be returned when a task is aborted by ~~another SCSI initiator port~~ by a command or task management function on another I_T nexus and the Control mode page TAS bit is set to one (see 5.6.3).

...

5.6 Aborting tasks

5.6.1 Mechanisms that cause tasks to be aborted

A task is aborted when an ~~event or SCSI initiator device action~~ SCSI device condition (see 6.3), command, or task management function causes termination of the task prior to its ~~successful~~ completion by the device server.

~~The following events cause a task or several tasks to be aborted~~ are capable of aborting tasks:

- ~~a) The return of an Execute Command service response of SERVICE DELIVERY OR TARGET FAILURE as described in 5.1;~~
- ~~b) A power on condition (see 6.3.1);~~
- ~~c) A hard reset (see 6.3.2);~~
- ~~d) A logical unit reset (see 6.3.3);~~
- ~~e) An I_T nexus loss (see 6.3.4);~~
- ~~f) A power loss expected (see 6.3.5); or~~
- ~~g) SCSI transport protocol specific conditions.~~

~~Editor's Note 1: resorted above list starting with power on and ending with power loss expected~~

~~An action~~ command or task management function 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~~ commands and task management functions affect only the task(s) received on the I_T nexus on which the ~~action~~ command or task management function 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 26 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 26 — Actions that affect task(s) received on this or other I_T nexuses

Action	Unit attention additional sense code, if any (see 5.6.3)
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
Receipt of a Power Loss Expected indication (see 6.3.5)	COMMANDS CLEARED BY POWER LOSS NOTIFICATION
SCSI transport protocol specific conditions	As defined by the applicable SCSI transport protocol standard

Table 27 lists the SCSI device conditions that cause tasks to be aborted in a SCSI initiator device.

Table 27 — SCSI device conditions that abort tasks in a SCSI initiator device

<u>SCSI device condition</u>	<u>Scope</u>	<u>Reference(s)</u>
<u>Power on</u>	<u>All tasks in the SCSI initiator device</u>	<u>6.3.1</u>
<u>Hard reset</u>	<u>All tasks with an I_T nexus involving the SCSI initiator port</u>	<u>6.3.2</u>
<u>I_T nexus loss</u>	<u>All tasks associated with the lost I_T nexus</u>	<u>6.3.4</u>
<u>SCSI transport protocol specific conditions</u>	<u>As defined by the applicable SCSI transport protocol standard</u>	

Table 28 lists the SCSI device conditions that cause tasks to be aborted in a SCSI target device.

Table 28 — SCSI device conditions that abort tasks in a SCSI target device

<u>SCSI device condition</u>	<u>Scope</u>	<u>Unit attention condition (see 5.8.7) additional sense code, if any</u>	<u>TASK ABORTED status</u> ^a	<u>Reference(s)</u>
<u>Power on</u>	<u>All tasks in the SCSI target device</u>	<u>See table 33 in 6.2</u>	<u>No</u>	<u>6.3.1</u>
<u>Hard reset</u>	<u>All tasks in all logical units to which the SCSI target port has access in the SCSI target device</u>		<u>Yes or no</u> ^c	<u>6.3.2</u>
<u>Logical unit reset</u> ^b	<u>All tasks in the logical unit</u>		<u>Yes or no</u> ^d	<u>6.3.3 and 7.7</u>
<u>I T nexus loss</u> ^b	<u>In each logical unit to which the SCSI target port has access, all tasks associated with the lost I T nexus</u>		<u>No</u>	<u>6.3.4 and 7.6</u>
<u>Power loss expected</u>	<u>All tasks in the SCSI target device</u>		<u>No</u>	<u>6.3.5</u>
<u>SCSI transport protocol specific conditions</u>	<u>As defined by the applicable SCSI transport protocol standard</u>			
^a "Yes" indicates that each task that is aborted on an I T nexus other than the one that caused the SCSI device condition is terminated with TASK ABORTED status, if the TAS bit is set to one in the Control mode page (see SPC-4). "No" indicates that no status is returned for aborted tasks. ^b This SCSI device condition is able to be invoked by a task management function listed in table 29. ^c If the hard reset is caused by a particular I T nexus (e.g., by a SCSI transport protocol-specific task management function), then "yes" applies". Otherwise, "no" applies. ^d If the logical unit reset is caused by a particular I T nexus (e.g., by a LOGICAL UNIT RESET task management function), then "yes" applies". Otherwise (e.g., if triggered by a hard reset), "no" applies.				

[Table 29 lists the task management functions that cause tasks to be aborted.](#)

Table 29 — Task management functions that abort tasks

Task management function	Scope	Unit attention condition (see 5.8.7) additional sense code, if any^a	TASK ABORTED status^b	Reference(s)
ABORT TASK (I T L Q Nexus)	Task specified by the I T L Q nexus argument	None	No	7.2
ABORT TASK SET (I T L Nexus)	All tasks in the task set with the same I T nexus as that specified by the I T L Nexus argument	None	No	7.3
CLEAR TASK SET (I T L Nexus)	All tasks in the task set^c	COMMANDS CLEARED BY ANOTHER INITIATOR	Yes	7.5
LOGICAL UNIT RESET (I T L nexus)	See table 28 description of the logical unit reset condition			
I T NEXUS RESET (I T nexus)	See table 28 description of the I T nexus loss condition			
<p>^a If if the TAS bit is set to zero in the Control mode page (see SPC-4), the device server creates this unit attention condition for each I T nexus that had task(s) aborted other than the I T nexus that delivered the task management function. If the TAS bit is set to one in the Control mode page (see SPC-4), the device server does not create this unit attention condition.</p> <p>^b “Yes” indicates that each task that is aborted on an I T nexus other than the one that delivered the task management function is terminated with TASK ABORTED status, if the TAS bit is set to one in the Control mode page. “No” indicates that no status is returned for aborted tasks.</p> <p>^c If the TST field is set to 001b (i.e., per-I T nexus) in the Control mode page (see SPC-4), there is one task set per I T nexus, so no other I T nexuses are affected and CLEAR TASK SET is equivalent to ABORT TASK SET.</p>				

Table 30 lists the command-related conditions that cause tasks to be aborted.

Table 30 — Command-related conditions that abort tasks

<u>Command-related condition</u>	<u>Scope</u>	<u>Unit attention condition (see 5.8.7) additional sense code, if any</u> ^a	<u>TASK ABORTED status</u> ^b	<u>Reference(s)</u>
Completion of a command with a CHECK CONDITION status if: a) the QERR field is set to 01b; and b) the TST field is set to 000b (i.e., shared) in the Control mode page (see SPC-4)	All tasks in the task set	COMMANDS CLEARED BY ANOTHER INITIATOR	Yes	5.8.1.3 and 5.8.2.2
Completion of a command with a CHECK CONDITION status if: a) the QERR field is set to 01b; and b) the TST field is set to 001b (i.e., per-I T nexus) in the Control mode page (see SPC-4)	All tasks in the task set ^c	None	No	5.8.1.3 and 5.8.2.2
Completion of a command with a CHECK CONDITION status if the QERR field is set to 11b in the Control mode page (see SPC-4)	All tasks in the task set with the same I T nexus as the command that was terminated	None	No	5.8.1.3 and 5.8.2.2
Processing 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-4)	All tasks from all I T nexuses with the specified reservation key	COMMANDS CLEARED BY ANOTHER INITIATOR	Yes	SPC-4
The return of an Execute Command service response of SERVICE DELIVERY OR TARGET FAILURE	The indicated task	None	No	5.1
Termination of an overlapped command	All tasks with the same I T nexus as the command that was terminated	None	No	5.8.3
<p>^a If the TAS bit is set to zero in the Control mode page (see SPC-4), the device server creates this unit attention condition for each I T nexus that had task(s) aborted other than the I T nexus that delivered the task management function. If the TAS bit is set to one in the Control mode page (see SPC-4), the device server does not create this unit attention condition.</p> <p>^b “Yes” indicates that each task that is aborted on an I T nexus other than the one that delivered the command is terminated with TASK ABORTED status, if the TAS bit is set to one in the Control mode page (see SPC-4). “No” indicates that no status is returned for aborted tasks.</p> <p>^c Since the TST field is set to 001b, there is one task set per I T nexus, so no other I T nexuses are affected.</p>				

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.

5.6.2 When a SCSI initiator port aborts tasks received on its own I_T nexus

~~When a SCSI initiator port causes task(s) received on its own I_T nexus to be aborted, no notification that the task(s) have been aborted shall be returned to the SCSI initiator port. When a device server receives a command or task management function on an I_T nexus that causes tasks on the same I_T nexus to be aborted, the device server shall not return any notification that those tasks have been aborted other than:~~

- a) ~~the completion response for the command or task management function **action** that caused the task(s) to be aborted; and~~
- b) ~~notification(s) associated with related effects of the **action** command or task management function (e.g., a reset unit attention condition).~~

5.6.3 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 26 (see 5.6.1):~~

~~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. 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 device server receives a command or task management function on an I_T nexus that causes tasks on other I_T nexuses to be aborted, the device server shall return notifications for those tasks based on the setting of the TAS bit in the Control mode page (see SPC-4):

- a) If the TAS bit is set to zero, the device server:
 - A) shall not return status for the tasks that were aborted; and
 - B) shall establish a unit attention condition for the SCSI initiator port associated with each I_T nexus containing tasks that were aborted with an additional sense code set as defined in table 27;
- b) If the TAS bit is set to one, the device server:
 - A) shall return TASK ABORTED status for each aborted task; and
 - B) shall not establish a unit attention condition for this reason.

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.

5.8 Command processing considerations and exception conditions

5.8.7 Unit Attention condition

Each logical unit shall generate a unit attention condition whenever one of the following events occurs:

- c) A power on (see 6.3.1), hard reset (see 6.3.2), logical unit reset (see 6.3.3), I_T nexus loss (see 6.3.4), or power loss expected (see 6.3.5) occurs;
- d) A removable medium may have been changed;
- e) The mode parameters associated with this I_T nexus have been changed by a task received on another I_T nexus (i.e., initiator ports share mode parameters, see SPC-3);
- f) The log parameters associated with this I_T nexus have been changed by a task received on another I_T nexus (i.e., initiator ports share log parameters, see SPC-3);
- g) The version or level of microcode has been changed (see SPC-3);

- h) Tasks received on this I_T nexus have been cleared by a task or a task management function associated with another I_T nexus and the TAS bit was set to zero in the Control mode page associated with this I_T nexus (see [5.6 SPC-3](#));
- i) INQUIRY data has been changed (see SPC-3);
- j) The logical unit inventory has been changed (see SPC-3);
- k) The mode parameters in effect for the associated I_T nexus have been restored from non-volatile memory (see SPC-3); or
- l) Any other event requiring the attention of the SCSI initiator device.

...

7.2 ABORT TASK

Request:

Service Response = ABORT TASK (IN (I_T_L_Q Nexus))

Description:

This function shall be supported by all logical units.

The task manager shall abort the specified task, if any, as described in [5.6-2](#). Previously established conditions, including ~~MODE-SELECT~~[mode](#) parameters, reservations, and ACA shall not be changed by the ABORT TASK function.

A response of FUNCTION COMPLETE shall indicate that the task was aborted or was not in the task set. In either case, the SCSI target device shall guarantee that no further requests or responses are sent from the task.

All SCSI transport protocol standards shall support the ABORT TASK task management function.

7.3 ABORT TASK SET

Request:

Service Response = ABORT TASK SET (IN (I_T_L Nexus))

Description:

This function shall be supported by all logical units.

The task manager shall abort all tasks in the task set that were received on the specified I_T nexus as described in 5.6. Tasks received on other I_T nexuses or in other task sets shall not be aborted.

This task management function performed is equivalent to a series of ABORT TASK requests.

[All pending status and sense data for the tasks that were aborted shall be cleared.](#) Other previously established conditions, including ~~MODE-SELECT~~[mode](#) parameters, reservations, and ACA shall not be changed by the ABORT TASK SET function.

All SCSI transport protocol standards shall support the ABORT TASK SET task management function.

7.4 CLEAR ACA

Request:

Service Response = CLEAR ACA (IN (I_T_L Nexus))

Description:

This function shall be supported by a logical unit if it supports ACA (see 5.2).

For the CLEAR ACA task management function, the task set shall be the one defined by the TST field in the Control mode page (see SPC-3).

An application client requests a CLEAR ACA using the faulted I_T nexus (see 3.1.39) to clear an ACA condition from the task set serviced by the logical unit. The state of all tasks in the task set shall be modified as described in 8.8. For a task with the ACA attribute (see 8.6.5) receipt of a CLEAR ACA function shall have

the same effect as receipt of an ABORT TASK function (see 7.2) specifying that task. If successful, this function shall be terminated with a service response of FUNCTION COMPLETE.

If the task manager clears the ACA condition, any task within that task set may be completed subject to the requirements for task set management specified in clause 8.

The service response for a CLEAR ACA request received from an I_T nexus other than the faulted I_T nexus shall be FUNCTION REJECTED.

All SCSI transport protocol standards shall support the CLEAR ACA task management function.

7.5 CLEAR TASK SET

Request:

Service Response = CLEAR TASK SET (IN (I_T_L Nexus))

Description:

This function shall be supported by [all](#) logical units.

~~All tasks in the task set shall be aborted~~[The task manager shall abort all tasks in the task set](#) as described in 5.6.

~~For the CLEAR TASK SET task management function, the task set shall be the one defined by the TST field in the Control mode page (see SPC-3). If the TST field is set to 001b (i.e., per-I T nexus) in the Control mode page (see SPC-4), there is one task set per I T nexus, so no other I T nexuses are affected and CLEAR TASK SET is equivalent to ABORT TASK SET (see 7.2).~~

All pending status and sense data for the task set shall be cleared. Other previously established conditions, including ~~MODE SELECT~~[mode](#) parameters, reservations, and ACA shall not be changed by the CLEAR TASK SET function.

All SCSI transport protocol standards shall support the CLEAR TASK SET task management function.

7.6 I_T NEXUS RESET

Request:

Service Response = I_T NEXUS RESET (IN (I_T_L Nexus))

Description:

SCSI transport protocols may or may not support I_T NEXUS RESET and may or may not require logical units accessible through SCSI target ports using such transport protocols to support I_T NEXUS RESET.

Each logical unit accessible through the SCSI target port shall perform the I_T nexus loss functions specified in 6.3.4 for the I_T nexus on which the function request was received, then the SCSI target device shall return a FUNCTION COMPLETE response. After returning a FUNCTION COMPLETE response, the logical unit(s) and the SCSI target port shall perform any additional functions specified by the SCSI transport protocol.

7.7 LOGICAL UNIT RESET

Request:

Service Response = LOGICAL UNIT RESET (IN (I_T_L Nexus))

Description:

This function shall be supported by all logical units.

Before returning a FUNCTION COMPLETE response, the logical unit shall perform the logical unit reset functions specified in 6.3.3.

NOTE 11 - Previous versions of this standard only required LOGICAL UNIT RESET support in logical units that supported hierarchical logical units.

All SCSI transport protocol standards shall support the LOGICAL UNIT RESET task management function.