Command Cleared Notification Proposal

(Round 3)

- General idea is for targets to notify initiators that their I/Os were cleared by the actions of another initiator
  - e.g. Target Reset, Clear Queue, Lun Reset, etc.
- Discussed twice in T11 working groups with FC only solutions.
- Recommendation was to take to T10 for a general SCSI solution (e.g. propose change in SAM-2)

Reason for Change

- Some environments use resets for more than error recovery.
  - Wolfpack uses resets in resource management: used to clear reservations
- Desire to restart I/Os from other initiators quickly
  - currently first indication is Power Up Unit Attention on **next** I/O (which may never come)

    OR
  - I/O timeout - 30, 60, 90 seconds typical

Propose New Status

- A new SCSI Status value of **0x40 - Command Cleared** is proposed. Sent by the target whenever an I/O is canceled by a task management function initiated by another initiator.
  - Sent for **each** I/O canceled
  - In FC, target must have Sequence Initiative (SI) to send.
  - Small window (on time scale) during life of command where initiator has SI. Target must wait for SI before sending FCP_RESP.
- Open Issue
  - Send / Don't send to initiator which performed Task Management function.

SAM 2 Status Codes

(for reference)

Table 13 — Status codes

Status Code	Status
0h	GOOD
2h	CHECK CONDITION
4h	CONDITION MET
8h	BUSY
10h	INTERMEDIATE
14h	INTERMEDIATE-CONDITION MET
18h	RESERVATION CONFLICT
22h	Obsolete
28h	TASK SET FULL
30h	ACA ACTIVE
All other codes	Reserved

Mode page to Enable

- Obviously this will cause problems if target uses new status and host is not aware.
- Enable via new bit in Control Mode Page (0x0A)
  - Suggest byte 3, bit 3
    (control mode page shown below for reference)

8.3.4 Control mode page

The control mode page (see table 158) provides controls over several SCSI features that are applicable to all device types such as tagged queuing, asynchronous event reporting, and error logging.

Table 158 — Control mode page

Bit / Byte	7	6	5	4	3	2	1	0
0	PS	Reserved	PAGE CODE (0Ah)					
1	PAGE LENGTH (0Ah)							
2	TST			Reserved			GLTSD	RECL
3	QUEUE ALGORITHM MODIFIER				Reserved	QERR		DQUE
4	Reserved	RAC	Reserved		SWP	RAERP	UAAERP	EAERP
5	Reserved							
6	(MSB)	READY AER HOLDOFF PERIOD						
7								(LSB)
8	(MSB)	BUSY TIMEOUT PERIOD						
9								(LSB)
10	(MSB)	EXTENDED SELF-TEST COMPLETION TIME						
11								(LSB)

A task set type field (TST) specifies the type of task set (see table 159). If the device maintains mode pages per initiator, the TST field, if changeable, shall reflect in all initiator pages the state selected by the most recent MODE SELECT. If the most recent MODE SELECT changes the setting of this field the device server shall establish a unit attention condition for all initiators except the one that issued the MODE SELECT command (see SAM-2). The device server shall set the additional sense code to MODE PARAMETERS CHANGED.

Table 159 — Task set type

Value	Description
000b	Task set per logical unit for all initiators
001b	Task set per initiator per logical unit
010b - 111b	Reserved

A global logging target save disable (GLTSD) bit of zero allows the target to provide a target-defined method for saving log parameters. A GLTSD bit of one indicates that either the target has disabled the target-defined method for saving log parameters or when set by the initiator specifies that the target-defined method shall be disabled.

A report log exception condition (RLEC) bit of one specifies that the device server shall report log exception conditions as described in 8.2. A RLEC bit of zero specifies that the device server shall not report log exception conditions.

The QUEUE ALGORITHM MODIFIER field (see table 160) specifies restrictions on the algorithm used for reordering tasks having the SIMPLE task attribute.

Table 160 — Queue algorithm modifier

Value	Description
0h	Restricted reordering
1h	Unrestricted reordering allowed
2h - 7h	Reserved
8h - Fh	Vendor-specific

A value of zero in the QUEUE ALGORITHM MODIFIER field specifies that the device server shall order the actual execution sequence of tasks having the SIMPLE task attribute such that data integrity is maintained for that initiator. This means that, if the transmission of new service delivery requests is halted at any time, the final value of all data observable on the medium shall have exactly the same value as it would have if all the tasks had been given the ORDERED task attribute. The restricted reordering value shall be the default value.

A value of one in the QUEUE ALGORITHM MODIFIER field specifies that the device server may reorder the actual execution sequence of tasks having the SIMPLE task attribute in any manner. Any data integrity exposures related to task sequence order shall be explicitly handled by the application client through the selection of appropriate commands and task attributes.

The queue error management (QERR) field specifies how the device server shall handle blocked tasks when another task receives a CHECK CONDITION status (see table 161). The task set type (see the TST field definition above) defines which tasks are blocked. If TST=000b, then all tasks from all initiators are blocked. If TST=001b, then only tasks from the initiator that receives the CHECK CONDITION status are blocked.

Table 161 — Queue error management (QERR) field

Value	Definition
00b	Blocked tasks in the task set shall resume after an ACA or CA condition is cleared (see SAM-2).
01b	All the blocked tasks in the task set shall be aborted when the CHECK CONDITION status is sent. A unit attention condition (see SAM-2) shall be generated for each initiator that had blocked tasks aborted except for the initiator to which the CHECK CONDITION status was sent. The device server shall set the additional sense code to COMMANDS CLEARED BY ANOTHER INITIATOR.
10b	Reserved
11b	Blocked tasks in the task set belonging to the initiator to which a CHECK CONDITION status is sent shall be aborted when the status is sent.

A disable queuing (DQUE) bit of zero specifies that tagged queuing shall be enabled if the device server supports tagged queuing. A DQUE bit of one specifies that tagged queuing shall be disabled. Any queued commands received by the device server shall be aborted. The method used to abort queued commands is protocol-specific.

The report a check (RAC) bit provides control of reporting long busy conditions or CHECK CONDITION status. A RAC bit of one specifies that a CHECK CONDITION status should be reported rather than a long busy condition (e.g., longer than the BUSY TIMEOUT PERIOD). A RAC bit of zero specifies that long busy conditions (e.g., busy condition during auto contingent allegiance) may be reported.

A software write protect (SWP) bit of one specifies that the logical unit shall inhibit writing to the medium after writing all cached or buffered write data, if any. When SWP is one, all commands requiring writes to the medium shall return CHECK CONDITION status and shall set the sense key to DATA PROTECT and the additional sense code to WRITE PROTECTED. When SWP is one and the device model defines a write protect (WP) bit in the

DEVICE-SPECIFIC PARAMETER field in the mode parameter header, the WP bit shall be set to one for subsequent MODE SENSE commands. A SWP bit of zero specifies that the logical unit may allow writing to the medium, depending on other write inhibit mechanisms implemented by the logical unit. When the SWP bit is zero, the value of the WP bit (if defined) is device model specific. For a list of commands affected by the SWP bit and details of the WP bit (if defined) see the command standard (see 3.1.11) for the specific device type.

The RAERP, UAAERP, and EAERP bits enable specific events to be reported via the asynchronous event reporting protocol. When all three bits are zero, the target shall not use asynchronous event reporting. AER is defined in SAM-2.

A ready AER permission (RAERP) bit of one specifies that the device server may issue an asynchronous event report upon completing its initialization sequence instead of generating a unit attention condition. A RAERP bit of zero specifies that the device server shall not issue an asynchronous event report upon completing its initialization sequence.

NOTE 52 If the device server's default value for the RAERP bit is one and it does not implement saved parameters or include a hardware switch, then it may be impossible to disable the initialization sequence asynchronous event reporting.

A unit attention AER permission (UAAERP) bit of one specifies that the device server may issue an asynchronous event report instead of creating a unit attention condition upon detecting an unit attention condition event (other than upon completing an initialization sequence). A UAAERP bit of zero specifies that the device server shall not issue an asynchronous event reporting instead of creating a unit attention condition.

An error AER permission (EAERP) bit of one specifies that the device server may issue an asynchronous event report upon detecting a deferred error condition instead of waiting to report the deferred error on the next command. An EAERP bit of zero specifies that the device server shall not report deferred error conditions via an asynchronous event reporting.

The READY AER HOLDOFF PERIOD field specifies the minimum time in milliseconds after the target starts its initialization sequence that it shall delay before attempting to issue an asynchronous event report. This value may be rounded up as defined in 5.2.

The BUSY TIMEOUT PERIOD field specifies the maximum time, in 100 milliseconds increments, that the initiator allows for the target to remain busy for unanticipated conditions which are not a routine part of commands from the initiator. This value may be rounded down as defined in 5.2. A 0000h value in this field is undefined by this standard. An FFFFh value in this field is defined as an unlimited period.

The EXTENDED SELF-TEST COMPLETION TIME field contains advisory data that an application client may use to determine the time in seconds that the device server requires to complete an extended self-test when the device server is not interrupted by subsequent commands and no errors occur during execution of the self-test. The application client should expect this time to increase significantly if other commands are sent to the logical unit while a self-test is in progress or if errors occur during execution of the self-test. Device servers supporting SELF-TEST CODE field values other than 000b for the SEND DIAGNOSTIC command (see 7.25), shall support the EXTENDED SELF-TEST COMPLETION TIME field.