

Date: January 5, 1994 X3T10/94-024 Rev 0  
To: X3T10 Committee (SCSI)  
From: Doug Hagerman (Digital)  
Subject: Error Handling for SCSI Controllers

This paper is a proposal for some additional error codes to handle situations encountered in storage subsystems, particularly RAID subsystems. This is intended to be incorporated into the SCSI Controller Commands (SCC) document.

## 8.0 Subsystem Environment

SCC describes subsystems that consist of addressable devices including DACLs (Disk Array Conversion Layers), disks, power supplies, fans, and operator consoles. Conventional SCSI devices, including all these except DACLs, may be considered as independent units since each reports only its own errors. The DACL device type is unique because it reports not only its own errors but also those resulting from events on lower level devices. A DACL is a controller, and has a slightly more complicated error reporting scheme as a result.

(Note that from the viewpoint of the initiator, there is no distinction between "controller errors" and "device errors handled by controller". Both types are reported to the initiator from the DACL LUN.)

## 8.1 Controller Errors

Subsystem controller (DACL) errors are those that occur in the controller itself, and are reported to the initiator using the appropriate SCSI mechanism, and the error type is indicated by the appropriate ASC/ASCQ combination for the SCC device type. An example of this method would be a controller memory error, in which case the error is not traceable to any underlying subsystem device.

In the case of a RAID subsystem, since the subsystem nominally represents itself to the initiator as a disk, the disk device type codes will be used. Additional error codes for controller specific situations are listed below.

## 8.2 Device Errors Handled by Controller

Errors in an underlying device can be handled automatically by the controller and reported to the initiator as subsystem exception conditions. An example of this situation is a disk error in a RAID subsystem, which would be handled by some method that was pre-arranged when the RAID subsystem was set up. The initiator would see only a subsystem exception condition, without the information about the details of the underlying disk error itself. Error codes and how they relate to underlying device errors are listed below.

## 8.3 Device Errors Handled by Initiator

Errors in an underlying device can also be handled by a pass-through mechanism at the controller. This method would typically be used for diagnostic or maintenance operations. The SCC addressing mechanism allows an initiator to send commands directly to any addressable device in the subsystem by simply specifying the LUN that represents the device. See the relevant addressing document. Errors that occur in this process cause a contingent allegiance condition on that LUN (task set, really) which is handled by the initiator in the normal SCSI fashion.

The controller's pass-through mechanism will report the ASC/ASCQ codes that are native to the device. No new codes will be needed for existing device types (disk, tape, etc.).

#### 8.4 Logging Device Errors

The subsystem can also optionally maintain a log of underlying device errors so that the initiator can find out the details of those errors for maintenance reasons.

#### 8.5 Status Values, Sense Key Codes, and ASC/ASCQ Values

This list includes new codes for conditions native to SCSI controllers, and those that the controller reports as a result of events triggered by underlying devices. Codes for existing device types (disks, etc.) are not listed here.

##### 8.5.1 Status Values

A controller may return any of the status codes described in the SCSI standard, including: GOOD, CHECK CONDITION, CONDITION MET, BUSY, INTERMEDIATE, INTERMEDIATE - CONDITION MET, RESERVATION CONFLICT, COMMAND TERMINATED, and QUEUE FULL. These status codes have the same meanings as described in the SCSI standard.

##### 8.5.2 Sense Key Codes and ASC/ASCQ Values

A controller may return the following sense key codes and ASC/ASCQ values. The following list shows the normal relationship between the codes and values, and the class of events that cause them to be reported. These sense key descriptions are in addition to the descriptions in the SCSI standard.

Sense Key Code	ASC	ASCQ	Event
-----	---	----	-----
NO SENSE			No specific sense key information to be reported.
RECOVERED ERROR			The last command completed successfully without data loss, with some recovery action performed by the controller. Data was not lost.
	xxh	xxh	Device unavailable, data regenerated.
	xxh	xxh	
NOT READY			The logical unit is not ready.
	xxh	xxh	Rebuild in progress.
	xxh	xxh	Recalculation in progress.
	xxh	xxh	Operator initiated activity.
MEDIUM ERROR			The last command terminated with a non-recovered error condition that was caused by a data storage condition. Data may have been lost.
	xxh	xxh	Redundancy failure.
	xxh	xxh	Spare not available.
	xxh	xxh	Check data error.
HARDWARE ERROR			The last command terminated with a non-recovered error condition that was caused by a non-data component of the system. Data may have been lost.
	xxh	xxh	Power supply failure.
	xxh	xxh	Fan failure.

ILLEGAL REQUEST	xxh xxh	There was an illegal parameter in the command or in the additional parameters.
	xxh xxh	Invalid bit specified.
	xxh xxh	Text string overflow.
	xxh xxh	Invalid P-LUI.
	xxh xxh	Invalide P-extent.
	xxh xxh	Invalid R-LUI.
	xxh xxh	Incompatible redundancy group parameter.
	xxh xxh	Invalid V-LUI.
	xxh xxh	Incompatible volume set parameter.
	xxh xxh	Invalid S-LUI.
	xxh xxh	Incompatible spare parameter.
UNIT ATTENTION		A data storage element was changed, or the device was reset.
DATA PROTECT		A command was attempted on a data area that is protected from this operation. The command is not executed.
BLANK CHECK		Blank or missing data area encountered.
VENDOR-SPECIFIC		This sense key is available for reporting vendor-specific conditions.
COPY ABORTED		Copy command aborted due to device error.
ABORTED COMMAND		The target aborted the command. The initiator may be able to recover by trying the command again.
EQUAL		SEARCH DATA found matching data.
VOLUME OVERFLOW		Data buffer end encountered.
MISCOMPARE		Data did not match.