memorandum

Hewlett-Packard Company
3000 Hanover Street
Palo Alto, CA 94304-1185
USA
www.hp.com

T10/05-154r3

To	**From**	**Subject**	**Date**
INCITS T10 Committee	Michael Banther, HP	SSC-3 TapeAlert enhancements	7 October 2005

Revision History

Revision 0 – Initial document.

Revision 1 – Incorporated comments from May SSC-3 working group:

- In 4.2.15.1:
  - Replaced the TapeAlert flag severity attributes with descriptive statements;
  - Removed the TapeAlert flag grouping;
  - Added a reference to Annex A;
- In 4.2.15.2.1:
  - Altered the text regarding checking for supported log pages to reflect the group's decision to make the TapeAlert log page mandatory;
  - Moved application client recommendations to Annex A and added statements indicating their deprecation;
- In 4.2.15.3:
  - Reworked the description of how flags are de-activated at power-off and may then be re-activated by power-on;
  - Changed LOG SELECT to LOG SENSE;
  - Added lettered lists enumerating the conditions and commands that can cause flag re-activation;
  - Changed 'per-initiator basis' to 'per I_T nexus basis';
  - Noted that the device server must de-activate TapeAlert flags on a per I_T nexus basis when the TAPLSD bit equals zero to maintain backwards compatibility.

Revision 2 – Incorporated comments from July SSC-3 working group:

- Added 4.2.10:
  - Moved Information sense data descriptor text and table from 8.2.3 to 4.2.10.3.
  - Extended the Information sense data descriptor to cover both a threshold condition met and an informational exception condition.
  - Formatted the TapeAlert bitmap as a proper Information sense data descriptor.
  - Renumbered 'Error conditions' clause as 4.2.10.4.
- In 4.2.15.2.1:
  - Removed the possibility that de-activation of a TapeAlert flag may generate an informational exception.
- In 4.2.15.2.2:
  - Changed the guidance for polling the TapeAlert log page or the TapeAlert Response log page at the beginning of a data transfer sequence to include prior to mounting a volume in order to cover the cases of cleaning cartridges, etc.
  - Changed the application client guidance to recommend reading TapeAlerts before de-mounting each volume regardless of volume spanning.
- In 4.2.15.2.3:
  - Removed the possibility that de-activation of a TapeAlert flag may generate an informational exception.
  - Added text describing the retrieval of TapeAlert flag information using the Information sense data descriptor.
- In 4.2.15.2.4: Modified text to describe the retrieval of TapeAlert flag information using the Information sense data descriptor.

- In 4.2.15.3:
  - Decoupled table x+3 from listing mandatory flags only.
  - Added a statement to de-activate all TapeAlert flags while processing a LOG SELECT command with the PARAMETER LIST LENGTH field set to zero and the PC field set to 11b.
  - Changed editor's note from 'SSC-2' to 'SSC-3.'
- Added 4.2.15.5: Added a model clause for the TapeAlert Response log page.
- In 8.2.3: Added text to enforce mutual exclusion between informational exception reporting and threshold condition met reporting.
- In 8.3.y:
  - Noted that the application client should set the TARPC bit to one when using the threshold usage model.
  - Added a cross reference to the Informational Exceptions Control mode page clause.
  - Removed the possibility that de-activation of a TapeAlert flag may generate an informational exception.
  - Renamed the TADIECR bit to TASER and redefined it to enforce mutual exclusion between informational exception reporting and threshold condition met reporting.
  - Added creation of a unit attention condition when the TASER bit causes a change in the ETC bit of a TapeAlert log parameter.
  - Added the TARPF bit to specify whether the device server respects or ignores the value of the LOG SENSE PPC and PARAMETER POINTER fields.
- In 8.3.6 and elsewhere:
  - Changed 'when' to 'if.'
  - Clarified that the device server resetting the REPORT COUNT/TEST FLAG NUMBER field only occurs on a MODE SENSE command for the Informational Exceptions Control mode page.
  - Added text to specify that the device server resets the TEST bit to zero on a MODE SENSE command for the Informational Exceptions Control mode page.
- In 8.4.z: Changed the value of the PAGE CODE field in table x+6 from B0h to B2h.
- In A.1.2:
  - Updated the application client guidance to match the fact that severity levels now have descriptive text instead of attributes.
  - Corrected a capitalization problem.

Revision 3 – Incorporated comments from September SSC-3 working group:

- In 4.2.15.1:
  - Added guidance to log the flag or warn the operator to all levels of severity.
  - Added a definition of Informational severity.
  - Removed an editorial note containing a question that the working group has resolved.
- In 4.2.15.2.3:
  - Changed 'upon activating a TapeAlert flag' and 'upon de-activating a TapeAlert flag' to 'due to an activated TapeAlert flag' and 'due to a deactivated TapeAlert flag' respectively.
- In 4.2.15.3:
  - Added a statement prohibiting deactivation of TapeAlert flags due to vendor-specific events/conditions.
  - Changed effect of LOG SELECT from deactivating all TapeAlert flags to not deactivating any of them.
- In 8.3.6:
  - Removed editorial notes containing questions that the working group has resolved.
- In 8.4.1:
  - Changed the PAGE CODE for the TapeAlert Supported Flags VPD page from B1h to B2h.

Background

The TapeAlert feature suffers from several deficiencies. These deficiencies have inhibited its adoption by application clients.

Table 1 lists the known deficiencies (SSC3r01c) and HP's proposed changes or enhancements:

Table 1 - TapeAlert Deficiencies and Proposed Corrections or Enhancements

	Deficiency	Proposed Correction or Enhancement
a	Clause 4.2.15.1 *Introduction to TapeAlert application client interface* exceeds the scope of the standard (see 1, *Scope*) in the constraints it seeks to place on application clients.	This proposal adds a clearly labelled Usage Model clause to 4.2.15. The new clause will describe the expected usage without mandating it.
b	Clause 4.2.15.1 *Introduction to TapeAlert application client interface* hints that an application client can configure a device server to report a TapeAlert flag state change as an informational exception and clause 8.3.6 *Informational Exceptions Control mode page* includes a vague reference to the additional sense code returned in such a case. The lack of detail requires a very close reading of the text and a fair amount of imagination to discover the usage model for this form of reporting. Reporting a TapeAlert flag state change as an informational exception may clash with reporting one using thresholds, yet an application client may wish to receive non-TapeAlert informational exceptions.	This proposal adds an explicit description of the informational exception usage model to 4.2.15. This proposal defines the Device Configuration Extension mode page. To this page, this proposal adds a mode bit that controls whether or not a TapeAlert flag state change results in an informational exception condition.
c	Clause 4.2.15.2 *TapeAlert log sense format* requires de-activation of all active TapeAlert flags on a per-initiator basis when a LOG SENSE command returns the TapeAlert log page. De-activating flags when read has not had the benefit to application clients originally anticipated as it requires the application client to remember the state of previously read flags. Keeping the log page on a per-initiator basis uses substantial memory in Fibre Channel devices.	To the Device Configuration Extension mode page, this proposal adds a mode bit that controls whether or not a LOG SENSE command for the TapeAlert log page de-activates all active flags. A bit value of zero causes de-activation of all active flags; a bit value of one prohibits de-activation. This proposal alters the existing text in SSC3r01b, 4.2.15 to strongly recommend keeping the TapeAlert log page on a per I_T nexus basis when the new mode bit equals zero and to permit keeping it on either a per I_T nexus or shared basis when the new mode bit equals one.
d	Clauses 4.2.15.2 *TapeAlert log sense format* and 8.2.3 *TapeAlert log page* define each TapeAlert flag as a separate log parameter. This format is inefficient and inconvenient for application clients that poll the TapeAlert log page.	This proposal adds an additional, optional TapeAlert log page with all flags in one parameter similar to the TapeAlert Response log page in ADCr07, 6.1.3.
e	The existing definition of TapeAlert does not provide the application client with information about which flags a device server supports.	This proposal defines the TapeAlert Supported Flag VPD page
f	Clause 4.2.15.2 *TapeAlert log sense format* states that "the page control bits in the LOG SENSE command are not applicable and shall be ignored by the device server." Nowhere does the standard state what value the device server assumes for the PC field of a LOG SENSE command. If the device server assumes Cumulative values (i.e., PC equal to 01b; the most likely choice), 8.2.3 *TapeAlert log page* requires the device server to contradict the behaviour for the DU bit required by SPC-3 (SPC3r21, 7.2.1 *Log page structure and page codes for all device types*).	To the Device Configuration Extension mode page, this proposal adds a mode bit that controls whether or not the device server ignores the PC field. A bit value of zero causes the device server to ignore the PC field and to behave as though the PC field were set to 01b. A bit value of one causes the device server to respond to the command consistent with the setting of the PC field. This proposal also removes the fixed value of zero from the DU bit of every parameter in the TapeAlert log page so that the device server can set DU to one when a parameter reaches its maximum value.

	Deficiency	Proposed Correction or Enhancement
g	Clause 4.2.15 sometimes describes TapeAlert flag states in terms of *activation* and *deactivation* and sometimes in terms of values, e.g., 0b or 1b.	This proposal changes the model clause, 4.2.15, to uniformly use activate and deactivate. It alters a sentence in 8.2.3 to clarify that an activated flag is represented by the value 1b and a deactivated flag is represented by the value 0b.
h	Clause 8.2.3 *TapeAlert log page* requires the device server to set the value of the TSD bit to zero. This value indicates that the device server, "implicitly saves the TapeAlert flag at vendor specific intervals" (SPC3r21b, 7.2.1 *Log page structure and page codes for all device types*). Saving TapeAlert flags doesn't make very good sense.	This proposal changes the mandated value for the TSD bit of every parameter in the TapeAlert log page. The mandated value changes to one. A separate proposal changes the defined value of the TSD bit in the TapeAlert Response log page in ADC-2.
i	Clause 8.2.3 *TapeAlert log page* unnecessarily prohibits a device server from implementing threshold values for TapeAlert flags.	This proposal removes the mandated value for the ETC bit and the TMC field of every parameter in the TapeAlert log page. This proposal also redefines the VALUE OF FLAG field of every parameter in the TapeAlert log page to reserve the most significant seven bits of the byte. Restricting the field to a single bit limits the maximum parameter value to 1b.
j	The text in clause 8.3.6 *Informational Exceptions Control mode page* is confusing to the point of being unreadable. It may or may not be compatible with the definition of the page in SPC-3 (the revision history for SPC3r21 records two changes to the mode page and two other changes that affect informational exceptions). The current SSC-3 text includes undefined terms such as "test/false informational exception condition", "true informational exception condition", "real informational exception condition", and "real CHECK CONDITION" to name a few. The current SSC-3 text requires specific behaviour if the TEST and DEXCEPT bits equal one and the TEST FLAG NUMBER field equals zero, but it is silent in the case where the TEST and DEXCEPT bits equal one and the TEST FLAG NUMBER field does not equal zero. In general, the text in 8.3.6 does not meet the level of clarity desired in a SCSI standard.	This proposal re-works clause 8.3.6 to make the meaning of the text transparent. Specifically, this proposal uses consistent terms and adds a definition for each term used. This proposal also adds text covering the case where the TEST and DEXCEPT bits equal one and the TEST FLAG NUMBER field does not equal zero.
k	Annex A is labelled as normative and contains a combination of normative and informative text for the 64 TapeAlert flags. Some of the informative text may exceed the scope of the standard. The text in Annex A needs clear labelling of what is normative and what is informative. Separation of the normative text – including the Code, Flag, Type, and Flag type columns – from the informative text may be in order.	This proposal moves the normative text from Annex A into the TapeAlert model clause, 4.2.15.

In the proposed text, text that is new compared to SSC-3 appears in blue, text that is deleted from SSC-3 appears in ~~red strikeout~~, and editorial comments appear in pink. Compared to revision 1 of this proposal, text that is new appears in blue underline.

Deleted: and text that has been deleted appears in a balloon

Changes to SSC-3

4.2.10 Error reporting (deficiency i)

4.2.10.3 ~~Error conditions~~ Information sense data descriptor

If reporting descriptor format sense data, the device server shall include an Information sense data descriptor for a unit attention condition generated:

a) By a TapeAlert specific informational exception condition; or

b) By a TapeAlert flag meeting its threshold criteria when the corresponding ETC bit in the TapeAlert log parameter is set to one.

Table x shows the format of the Information sense data descriptor.

Table x — Information sense data descriptor

Bit Byte	7	6	5	4	3	2	1	0
0	DESCRIPTOR TYPE (00h)							
1	ADDITIONAL LENGTH (0Ah)							
2	VALID (1b)	Reserved						
3	Reserved							
4	FLAG01h	FLAG02h	FLAG03h	FLAG04h	FLAG05h	FLAG06h	FLAG07h	FLAG08h
5	FLAG09h	FLAG0Ah	FLAG0Bh	FLAG0Ch	FLAG0Dh	FLAG0Eh	FLAG0Fh	FLAG10h
6	FLAG11h	FLAG12h	FLAG13h	FLAG14h	FLAG15h	FLAG16h	FLAG17h	FLAG18h
7	FLAG19h	FLAG1Ah	FLAG1Bh	FLAG1Ch	FLAG1Dh	FLAG1Eh	FLAG1Fh	FLAG20h
8	FLAG21h	FLAG22h	FLAG23h	FLAG24h	FLAG25h	FLAG26h	FLAG27h	FLAG28h
9	FLAG29h	FLAG2Ah	FLAG2Bh	FLAG2Ch	FLAG2Dh	FLAG2Eh	FLAG2Fh	FLAG30h
10	FLAG31h	FLAG32h	FLAG33h	FLAG34h	FLAG35h	FLAG36h	FLAG37h	FLAG38h
11	FLAG39h	FLAG3Ah	FLAG3Bh	FLAG3Ch	FLAG3Dh	FLAG3Eh	FLAG3Fh	FLAG40h

The VALID bit shall be set to one.

An active TapeAlert flag has the corresponding FLAGXX bit set to one. An inactive TapeAlert flag has the corresponding FLAGXX bit set to zero.

4.2.10.~~3~~4 Error conditions

4.2.15 TapeAlert application client interface (deficiencies a, b, c, g, and k)

Replace all of 4.2.15.1 through 4.2.15.3 with:

4.2.15.1 TapeAlert introduction

TapeAlert provides an application client with the capability to receive notification of various events and conditions arising in the target device. This standard defines 64 unique TapeAlert flags for a sequential-access device. Other standards (e.g. SMC-3) may define other TapeAlert flags.

TapeAlert flags fall into three categories of severity (see table x+1).

Table x+1 — TapeAlert flag severity

Severity	Code	Definition
Critical	C	Either a failure has already occurred or a failure is likely to occur immediately. Continued operation without corrective action is likely to cause a failure. The condition should be logged and/or an operator informed.
Warning	W	If this condition is not corrected a data loss failure may occur. Continued operation without corrective action may possibly cause a failure. The condition should be logged and/or an operator informed.
Information	I	The condition should be logged and/or the operator informed. No guidance about continued operation without corrective action is given by this standard.

Deleted: EDITOR'S NOTE: The working group has not agreed a definition for Information.

Formatted: Font color: Blue

Table x+2 defines the 64 TapeAlert flags for a sequential-access device. See Annex A for additional information about each TapeAlert flag.

Table x+2 — TapeAlert flags

Flag	Name	Type	Severity	Deactivation condition
01h	Read warning	O	W	Start of next medium load
02h	Write warning	O	W	Start of next medium load
03h	Hard error	M	W	Start of next medium load[a]
04h	Media	M	C	Start of next medium load[a]
05h	Read failure	M	C	Start of next medium load[a]
06h	Write failure	M	C	Start of next medium load[a]
07h	Media life	O	W	Start of next medium load
08h	Not data grade	O	W	Start of next medium load
09h	Write protect	O	C	Start of next medium load or removal of write protect
0Ah	No removal	O	I	After medium removal allowed
0Bh	Cleaning media	O	I	Start of next medium load
0Ch	Unsupported format	O	I	Start of next medium load or format change
0Dh	Recoverable mechanical cartridge failure	O	C	Start of next medium load
0Eh	Unrecoverable mechanical cartridge failure	O	C	After service resolution
0Fh	Memory chip in cartridge failure	O	W	Start of next medium load
10h	Forced eject	O	C	Start of next medium load
11h	Read only format	O	W	Start of next medium load or format change
12h	Tape directory corrupted on load	O	W	Start of next medium load
13h	Nearing media life	O	I	Start of next medium load
14h	Clean now	O	C	After successful cleaning or cause resolved
15h	Clean periodic	O	W	After successful cleaning
16h	Expired cleaning media	O	C	Start of next medium load
17h	Invalid cleaning tape	O	C	Start of next medium load
18h	Retension requested	O	W	After successful retension
19h	Dual-port interface error	O	W	After interface returns to operation
1Ah	Cooling fan failure	O	W	After service resolution
1Bh	Power supply failure	O	W	After service resolution
1Ch	Power consumption	O	W	After power consumption returns to within specification
1Dh	Drive maintenance	O	W	After service resolution
1Eh	Hardware A	O	C	After service resolution
1Fh	Hardware B	M	C	At power on event
20h	Interface	O	W	After interface returns to operation

21h	Eject media	O	C	Start of next medium load
22h	Down-load fail	O	W	Start of next firmware download
23h	Drive humidity	O	W	After humidity returns to within specification
24h	Drive temperature	O	W	After temperature returns to within specification
25h	Drive voltage	O	W	After voltage returns to within specification
26h	Predictive failure	O	C	After service resolution
27h	Diagnostics required	O	W	After service resolution
28h- 2Eh	Obsolete			
2Fh- 31h	Reserved			
32h	Lost statistics	O	W	Start of next medium load
33h	Tape directory invalid at unload	O	W	Start of next medium load
34h	Tape system area write failure	O	C	Start of next medium load
35h	Tape system area read failure	O	C	Start of next medium load
36h	No start of data	O	C	Start of next medium load
37h	Loading failure	O	C	Start of next medium load
38h	Unrecoverable unload failure	O	C	After service resolution
39h	Automation interface failure	O	C	After service resolution
3Ah	Firmware failure	O	W	After service resolution
3Bh	WORM Medium – Integrity Check Failed	O	W	Start of next medium load
3Ch	WORM Medium – Overwrite Attempted	O	W	Start of next medium load
3Dh–40h	Reserved			

Type Key: M=Mandatory
O=Optional

[a] Devices compliant with previous generations of this standard may deactivate this TapeAlert flag when de-mounting the current medium.

EDITOR'S NOTE: Based on comments received during the May SSC-3 working group meeting, I believe that the group should review the severity assignment for each TapeAlert flag once we reach consensus on the TapeAlert flag definitions.

EDITOR'S NOTE: Although I have changed many occurrences of 'when' to 'if', I have also left occurrences of 'when' that do not appear synonymous with 'if.' I would like to review these remaining occurrences with the group.

Deleted: EDITOR'S NOTE: SSC-2 specifies that a power-cycle deactivates the Hardware B flag. ADC specifies deactivation, 'after service resolution.' I've retained the SSC-2 text in this proposal. Do we want to keep the current SSC-2 text or move to the ADC text?¶

4.2.15.2 TapeAlert usage model

4.2.15.2.1 TapeAlert usage model introduction

This standard describes three methods for an application client to monitor activation of TapeAlert flags:

a) Polling either the TapeAlert log page or the TapeAlert Response log page;

b) Configuring the device server to establish an informational exception condition upon activation of one or more TapeAlert flags.

c) Establishing a threshold for one or more of the parameters in the TapeAlert log page.

An application client may use any of these methods or a mixture of them.

Prior to using the TapeAlert Response log page with method (a), an application client should determine whether the device server supports the TapeAlert Response log page. An application client may determine if a device server supports a log page by issuing a LOG SENSE command with the PAGE CODE field set to 00h and examining the data returned.

4.2.15.2.2 TapeAlert polling usage model

The application client configures the device server for the TapeAlert polling usage model by:

a) Setting the TASER bit in the Device Configuration Extension mode page to one (see 8.3.y); and

b) Setting the ETC bit of every parameter in the TapeAlert log page to zero (see 8.2.3).

NOTE: Devices that comply with earlier generations of this standard set the ETC bit in each TapeAlert log parameter to zero and do not allow the application client to change this value.

If using the TapeAlert polling usage model, the application client reads the TapeAlert log page or the TapeAlert Response log page without receiving notification from the device server that a TapeAlert flag has changed state. The application client may read the TapeAlert log page or the TapeAlert Response log page at any time (e.g. polled at a regular interval of 60 seconds). The application client should read either the TapeAlert log page or the TapeAlert Response log page:

a) Prior to mounting a volume and at the beginning of a data transfer sequence;

b) Immediately after detecting an unrecoverable error during the data transfer sequence;

c) Before de-mounting each volume; and

d) At the end of a data transfer sequence.

4.2.15.2.3 TapeAlert informational exception usage model

The application client configures the device server for the TapeAlert informational exception usage model by:

a) Setting the TASER bit in the Device Configuration Extension mode page to zero (see 8.3.y);

b) Setting the DEXCPT bit in the Informational Exceptions Control mode page to zero and the TEST bit in the Informational Exceptions Control mode page to zero (see 8.3.6);

c) Setting the MRIE field in the Informational Exceptions Control mode page to a supported value greater than zero (see 8.3.6); and

d) Setting the ETC bit of every parameter in the TapeAlert log page to zero (see 8.2.3).

NOTE: Devices that comply with earlier generations of this standard set the ETC bit in each TapeAlert log parameter to zero and do not allow the application client to change this value.

If using the TapeAlert informational exception usage model, the application client receives TapeAlert flag information upon or after receiving notification from the device server that an informational exception has occurred. The device server generates an informational exception condition due to an activated TapeAlert flag. The device server does not generate an informational exception condition due to a de-activated TapeAlert flag. The method used by the device server to report the informational exception condition depends on the value of the MRIE field (see SPC-3). If the device server returns descriptor format sense data (see SPC-3), the current state of all TapeAlert flags appears in the Information sense data descriptor (see 4.2.10.3). If the device server returns fixed format sense data (see SPC-3), the application client should read the TapeAlert log page to retrieve the state of the TapeAlert flags.

Deleted: upon activating a

Deleted: upon de-activating a

If the TEST bit is set to zero, a device server reporting an informational exception condition for a TapeAlert flag sets the additional sense code to FAILURE PREDICTION THRESHOLD EXCEEDED.

4.2.15.2.4 TapeAlert threshold usage model

The application client configures the device server for the TapeAlert threshold usage model by:

a) Setting the TASER bit in the Device Configuration Extension mode page to one (see 8.3.y);
b) Setting to one the ETC bit of each parameter in the TapeAlert log page for which the application client wishes to receive a unit attention condition (see 8.2.3);
c) Setting to zero the ETC bit of each parameter in the TapeAlert log page for which the application client does not wish to receive a unit attention condition (see 8.2.3); and
d) Establishing a threshold value and a threshold met criteria (TMC) value for each TapeAlert log page parameter with the ETC bit set to one (see SPC-3).

NOTE: Devices that comply with earlier generations of this standard set the ETC bit in each TapeAlert log parameter to zero and do not allow the application client to change this value. These devices do not support the TapeAlert threshold usage model.

If using the TapeAlert threshold usage model, the application client receives a unit attention when a TapeAlert log page parameter meets its threshold criteria. If the device server returns descriptor format sense data (see SPC-3), the current state of all TapeAlert flags appears in the Information sense data descriptor (see 4.2.10.3). If the device server returns fixed format sense data (see SPC-3), the application client should read the TapeAlert log page to retrieve the state of the TapeAlert flags.

The threshold and TMC values determine whether the device server generates a unit attention condition on TapeAlert flag activation or de-activation.

EDITOR'S NOTE: 05-308 removes the SPC-4 restriction that all counting log parameters on a page must freeze when one of them reaches its threshold.

4.2.15.3 TapeAlert flag activation and deactivation

The device server shall activate a TapeAlert flag upon detecting the condition or event specified in table x+3.

Table x+3 — TapeAlert flag activation conditions

Flag	Name	Activation condition
03h	Hard error	An unrecoverable read/write/positioning error.
04h	Media	An unrecoverable read/write/positioning error due to a faulty medium.
05h	Read failure	An unrecoverable read error due to either a faulty medium or faulty device hardware.
06h	Write failure	An unrecoverable write/positioning error due to either a faulty medium or faulty device hardware.
1Fh	Hardware B	An error during power-on self test.

EDITOR'S NOTE: At the July SSC-3 working group meeting, the group reached a consensus to add activation conditions for other TapeAlert flags to this table.

The device server may activate a TapeAlert flag not listed in table x+3 upon detection of a vendor-specific condition.

The device server shall deactivate a TapeAlert flag upon detecting the condition or event specified for that flag in table x+2. The device server shall not deactivate any TapeAlert flag due to a vendor-specific condition/event. The device server shall deactivate all TapeAlert flags:

a) Upon processing a LOG SENSE command with the PAGE CODE field set to 2Eh if the TAPLSD bit is set to zero (see 8.3.y); or
b) Upon detecting a logical unit reset condition (see SAM-3).

Deleted:);

The device server shall not deactivate any TapeAlert flag due to:

a) Processing a LOG SELECT command with the PCR field set to one (see SPC-3); or

Deleted: While p

b) Processing a LOG SELECT command with the PARAMETER LIST LENGTH field set to zero and the PC field set to 11b.

Deleted: While p

Initialization processing due to a power-on condition may activate some TapeAlert flags.

If the device server deactivates a TapeAlert flag by processing a LOG SENSE command with the PAGE CODE field set to 2Eh, the device server shall not activate the flag again until the device server:

a) Detects the deactivation condition given in table x+2;

b) Detects a logical unit reset condition; or

c) Processes a LOG SELECT command with the PCR field set to one.

If the device server deactivates a TapeAlert flag through some other mechanism, the device server may activate the flag before

a) Detecting the deactivation condition given in table x+2;

b) Detecting a logical unit reset condition; or

c) Processing a LOG SELECT command with the PCR field set to one.

If the TAPLSD in the Device Configuration Extension mode page (see 8.3.y) is set to zero, the device server should de-activate flags on a per I_T nexus basis such that active flags are available for reading by other I_T nexuses. If the TAPLSD in the Device Configuration Extension mode page (see 8.3.y) is set to one, the device server may de-activate flags on a per I_T nexus basis.

NOTE: The device server de-activating TapeAlert flags on any basis other than per I_T nexus when the TAPLSD bit is set to zero violates backwards compatibility with previous versions of this standard.

EDITOR'S NOTE: The corresponding sentence in the current SSC-2 TapeAlert text contains 'shall'.

EDITOR'S NOTE: At the July SSC-3 meeting, IBM raised the following question and the group reached the following consensus:

> Does the statement, 'the device server should de-activate flags on a per I_T nexus basis' apply to flags de-activated by a LOG SELECT command? Each tape drive vendor will tell me their implementation and, if we all do it the same way, I'll make the text match. Otherwise, I'll indicate that the de-activation due to a LOG SELECT command clears on a vendor-specific basis.

Between the July SSC-3 meeting and present, I have received the following information regarding IBM and Quantum implementations:

> IBM has at least one product where a LOG SELECT with PCR set to one clears all TapeAlert flags in the LUN without regard to I_T nexus.
>
> Quantum reports that LOG SELECT does not affect TapeAlert flags at all in a SDLT.

4.2.15.4 WORM TapeAlert flags (deficiency g)

Two TapeAlert flags exist to support Write Once Read Many (WORM) media:

a) 3Bh, WORM Medium, Integrity Check Failed; and

b) 3Ch, WORM Medium, Overwrite Attempted.

If the device server supports TapeAlert flag 3Bh, it shall activate ~~set~~ that flag ~~to one~~ upon detecting that the integrity of the medium may be compromised ~~comprimised~~. If the device server supports TapeAlert flag 3C, it shall activate ~~set~~ that flag ~~to one~~ if an application client attempts to overwrite or erase user data.

In addition to the deactivation conditions for all TapeAlert flags (see 4.2.15.2), the device server shall deactivate ~~set~~ TapeAlert flags 3Bh and 3Ch ~~to zero~~ upon:

a) Execution of a LOAD UNLOAD command with a load bit set to one (see 7.2) that results in a not ready to ready transition, or if both the medium and device server support MAM, that results in access to medium auxiliary memory only; or

b) Execution of an autoload operation (see SPC-3) that results in a not ready to ready transition, or if both the medium and device server support MAM, that results in access to medium auxiliary memory only.

4.2.15.5 TapeAlert Response log page (deficiency d)

The TapeAlert flags reported through the TapeAlert Response log page (see 8.2.1) represent states. This approach facilitates accurate reporting of the conditions encountered by the device and allows the application client to manage the information directly. The device server does not maintain unique TapeAlert information for each initiator port, and the state flags are not affected by port events (e.g., port logins).

The application client is responsible for determining which flags have changed state upon subsequent retrieval of the TapeAlert Response log page, requiring the application client to maintain at least one previously retrieved TapeAlert Response log page in order to detect differences. The application client may maintain a state change history.

If the TAPLSD bit (see 8.3.y) is set to one, the device server shall maintain the value of the flags in the TapeAlert Response log page independently of the TapeAlert flags reported through the TapeAlert log page (see 8.2.3). A LOG SENSE command that retrieves the TapeAlert Response log page shall not set the flags in that page to zero and shall not set the flags in the TapeAert log page to zero. A LOG SENSE command that retrieves the TapeAlert log page shall not set the flags in the TapeAert Response log page to zero.

8.2.1 Log parameters overview (deficiency d)

Table 51 – Log page codes

Page code	Description	Reference
	. . .	
12h	TapeAlert Response log page	ADC
~~11h~~13h – 2Dh	Reserved	
	. . .	

8.2.3 TapeAlert log page (deficiencies f, g, h, and i)

The TapeAlert log page (see Table 53) defines error and informational flags used for detailed device diagnostics and management (see 4.2.15 and Annex A). ~~Refer to table A.1 (see Annex A) for a description of the flags. Refer to SPC-3 for a description of the log page fields.~~

Table 53 – TapeAlert log page

Bit Byte	7	6	5	4	3	2	1	0
0	Reserved		PAGE CODE (2Eh)					
1	Reserved							
2	(MSB)	PAGE LENGTH (n-3 ~~140h~~)						
3								(LSB)
	TapeAlert log parameters							
4	First TapeAlert log parameter							
	. . .							
n	Last TapeAlert log parameter							
~~5n-1~~	~~(MSB)~~	~~PARAMETER CODE (n)~~						
~~5n~~								~~(LSB)~~
~~5n+1~~	~~DU (0)~~	~~DS (1)~~	~~TSD (0)~~	~~ETC (0)~~	~~TMC (0)~~		~~LBIN (0)~~	~~LP (0)~~
~~5n+2~~	~~PARAMETER LENGTH (1)~~							
~~5n+3~~	~~VALUE OF FLAG~~							

See SPC-3 for a description of the PAGE CODE and PAGE LENGTH fields.

~~The parameter length field shall be set to 140h to allow for the transfer of all 64 flags.~~

Table x+4 defines the format of a TapeAlert log parameter.

Table x+4 – TapeAlert parameter format

Byte \ Bit	7	6	5	4	3	2	1	0
0	(MSB)	PARAMETER CODE						
1								(LSB)
2	DU	DS (1)	TSD (1)	ETC	TMC		LBIN (0)	LP (0)
3	PARAMETER LENGTH (01h)							
4	Reserved							FLAG

The value of ~~n in~~ the parameter code field shall ~~be in the~~ range from ~~of~~ 1 to 64.

See SPC-3 for a description of the DU bit, DS bit, TSD bit, ETC bit, TMC field, LBIN bit, and LP bit. The DS bit, TSD bit, LBIN bit, and LP bit shall be set to the value shown in table x+4.

An active TapeAlert flag has the FLAG bit set to one. An inactive TapeAlert flag has the FLAG bit set to zero. ~~A value of one in the VALUE OF FLAG field specifies the flag is set. Any other value specifies that the flag is not set.~~

If processing a LOG SELECT command and if the application client sends parameter data for the TapeAlert log page with the TSD or DS bits set to zero, the LP, LBIN or FLAG bits set to one, or the PARAMETER LENGTH field set to a value other than 01h, the device server shall:

a) Terminate the command with CHECK CONDITION status;
b) Set the sense key to ILLEGAL REQUEST; and
c) Set the additional sense code to INVALID FIELD IN PARAMETER LIST.

If the TASER bit is set to zero (see 8.3.y), the device server shall, upon processing a LOG SELECT command where the application client has sent parameter data for the TapeAlert log page with an ETC bit set to one:

a) Terminate the command with CHECK CONDITION status;
b) Set the sense key to ILLEGAL REQUEST; and
c) Set the additional sense code to INVALID FIELD IN PARAMETER LIST.

8.3.1 Mode parameters overview (deficiency b)

Table 58 – Mode page codes

Page code	Subpage code	Description	Reference
		. . .	
10h	01h	Device Configuration Extension mode page	8.3.y
		. . .	

8.3.y Device Configuration Extension mode page (deficiencies b, c and f)

The Device Configuration Extension mode page (see table x+5), a subpage of the Device Configuration mode page (see 8.3.3), provides control over SCSI features specific to Sequential Access devices. If a device server supports the Device Configuration Extension mode page, then the device server shall provide access to the mode page using the shared mode page policy (see SPC-3).

Table x+5 – Device Configuration Extension mode page

Bit Byte	7	6	5	4	3	2	1	0
0	PS	SPF (1b)	PAGE CODE (10h)					
1	SUBPAGE CODE (01h)							
2	(MSB)	PAGE LENGTH (1Ch)						
3								(LSB)
4	Reserved				TARPF	TASER	TARPC	TAPLSD
5	Reserved							
31								

See SPC-3 for a description of the PS bit, SPF bit, PAGE CODE field, SUBPAGE CODE field, and PAGE LENGTH field.

A TapeAlert Prevent LOG SENSE De-activation (TAPLSD) bit set to one specifies that the device server shall not alter the value of implemented TapeAlert FLAG parameters (see 8.2.3) due to processing of a LOG SENSE command with the PAGE CODE field set to 2Eh. A TAPLSD bit set to zero specifies that, as part of the processing of a LOG SENSE command with the PAGE CODE field set to 2Eh, the device server shall deactivate every supported TapeAlert flag.

A TapeAlert Respect Page Control (TARPC) bit set to one specifies that the device server shall select the type of parameter values for the TapeAlert log page (see 8.2.3) using the value of the page control field in a LOG SELECT or LOG SENSE CDB (see SPC-3). A TARPC bit set to zero specifies that the device server shall select cumulative parameter values for the TapeAlert log page regardless of the value of the page control field in a LOG SELECT or LOG SENSE CDB.

> NOTE: An application client using the TapeAlert threshold usage model (see 4.2.15.2.4) should set the TARPC bit to one.

A TapeAlert Select Exception Reporting (TASER) bit set to one specifies that:

a) Activation of a TapeAlert flag shall not result in an informational exception condition (see 8.3.6); and

b) Activation or deactivation of a TapeAlert flag may result in the generation of a unit attention condition with the additional sense code set to THRESHOLD CONDITION MET depending on the value of the ETC bit in each parameter in the TapeAlert log page.

A TASER bit set to zero specifies that:

a) Activation of a TapeAlert flag shall result in an informational exception condition (see 8.3.6);

b) Activation or deactivation of a TapeAlert flag shall not result in the generation of a unit attention condition with the additional sense code set to THRESHOLD CONDITION MET; and

c) The device server shall set the ETC bit in each parameter of the TapeAlert log page to zero.

The device server may provide independent sets of TapeAlert log parameters for each I_T nexus (see SPC-3). If the device server does not support independent sets of TapeAlert log parameters and the processing of a MODE SELECT command with the TASER bit set to zero results in a change in the value of the ETC bit in a TapeAlert log parameter, then the device server shall establish a unit attention condition (see SAM-3) for the initiator port associated with every I_T nexus, except the I_T nexus on which the MODE SELECT command was received, with the additional sense code set to LOG PARAMETERS CHANGED.

A TapeAlert Respect Parameter Fields (TARPF) bit set to one specifies that the device server shall select the parameters of the TapeAlert log page (see 8.2.3) to report in response to a LOG SENSE command using the values of the PPC bit and the PARAMETER POINTER field (see SPC-3). A TARPF bit set to zero specifies that the device server shall report all TapeAlert log page parameters regardless of the values of the PPC bit and the PARAMETER POINTER field.

8.3.6 Informational Exceptions Control mode page (deficiency j)

In addition to support for all device types (see SPC-3), the ~~The~~ Informational Exceptions Control mode page (see table 70) specifies ~~is used to specify the~~ parameters for the control of TapeAlert specific informational exception conditions for a sequential-access device.

Table 70 – Information Exceptions Control mode page

Bit Byte	7	6	5	4	3	2	1	0
0	PS	spf (0b) ~~Rsvd~~	PAGE CODE (1Ch)					
1	PAGE LENGTH (0Ah)							
2	PERF	Rsvd	EBF	EWASC	DEXCPT	TEST	Rsvd	LOGERR
3	Reserved				MRIE			
4	(MSB)	INTERVAL TIMER						
5								
6								
7								(LSB)
8	(MSB)	REPORT COUNT/TEST FLAG NUMBER						
9								
10								
11								(LSB)

See SPC-3 for a description of the PS bit SPF bit, PAGE CODE field, ~~and~~ PAGE LENGTH field, PERF bit, EBF bit, EWASC bit, LOGERR bit, and INTERVAL TIMER field.

~~See SPC-3 for a description of the LOGERR bit.~~

SPC-3 defines the effect of setting the TEST bit to one if the REPORT COUNT/TEST FLAG NUMBER field is set to zero. In response to a MODE SENSE command reporting parameters from the Informational Exceptions Control mode page, the device server shall set the value of the TEST bit to zero. The device server shall not alter the value of any TapeAlert flags in response to an application client setting the TEST bit to one and the REPORT COUNT/TEST FLAG NUMBER field to zero.

EDITORIAL NOTE: The proposed text removes the restriction for the device server to ignore the INTERVAL TIMER field. I see no reason to retain this restriction.

~~A TEST bit of one specifies the target shall generate test/false informational exceptions conditions as follows:~~

a) ~~if the TEST bit is one and the TEST FLAG NUMBER value is zero, the target shall generate a false informational exception condition based on the MRIE field (the INTERVAL TIMER field is ignored and the REPORT COUNT field is used as the TEST FLAG NUMBER). When a false informational exception condition is posted, the TapeAlert flags in the log page shall not be modified. True informational exception conditions shall have priority over false informational exception conditions. The TEST bit shall be automatically set to zero when the false informational exception condition is posted on the first command (excluding INQUIRY, REPORT LUNS, and REQUEST SENSE commands) that is received with no real informational exception condition pending. The false informational exception condition shall be reported in the method specified by the MRIE value and the additional sense code shall be set to FAILURE PREDICTION THRESHOLD EXCEEDED (FALSE). If the TEST and DEXCPT bits are one and the TEST FLAG NUMBER value is zero, the target shall return CHECK CONDITION. The sense key shall be set to ILLEGAL REQUEST and the additional sense code shall be set to INVALID FIELD IN PARAMETER LIST; or~~

Deleted: EDITORIAL NOTE: The current draft standard text states that the device server will not report an informational exception condition generated by an application client setting the TEST bit to one if another informational exception condition exists to report. Should we keep this hierarchy (HP internally is opposed to keeping it)? Some ambiguity exists in the current text: does the device server retain the test informational exception condition or discard it? If we keep the hierarchy, should we mandate one behaviour or the other?¶
EDITORIAL NOTE: The current draft standard text states that the device server shall reset the TEST bit under a limited set of circumstances: upon reporting an informational exception condition due to an application client setting the TEST bit to one and the REPORT COUNT/TEST FLAG NUMBER to zero. The July, 2005 SSC-3 working group agreed to reset the bit regardless of the value of the REPORT COUNT/TEST FLAG NUMBER. Should we keep the exemption in the current draft standard text for INQUIRY, REPORT LUNS, or REQUEST SENSE? The definition of REPORT LUNS already contains a statement that will exempt it (see SPC3r21d, 6.21 REPORT LUNS). The INQUIRY and REQUEST SENSE command definitions do not include a statement that would exempt them.¶

Table 71 defines the effect of setting the TEST bit to one if the REPORT COUNT/TEST FLAG NUMBER field is set to a non-zero value. In response to a MODE SENSE command reporting parameters from the Informational Exceptions Control mode page, the device server shall set the value of the TEST bit to zero. If both the TEST bit and the DEXCEPT bit are set to one, the device server shall terminate the MODE SELECT command with CHECK CONDITION status, shall set the sense key set to ILLEGAL REQUEST, and shall set the additional sense code set to INVALID FIELD IN PARAMETER LIST.

~~b) if the TEST bit is one and the TEST FLAG NUMBER value is a valid non zero value, the target shall generate or clear a test informational exception condition based on the TEST FLAG NUMBER value as described in table 71.~~

Table 71 – TEST bit and TEST FLAG NUMBER field definition ~~TapeAlert test descriptions~~

TEST FLAG NUMBER ~~Value~~	Description
1 to 64	Activate the TapeAlert flag specified by the TEST FLAG NUMBER field ~~in the log page~~. Report the informational exception condition for the TapeAlert flag with an additional sense code of FAILURE PREDICTION THRESHOLD EXCEEDED (FALSE) and ~~Once the TapeAlert flag is activated it shall be processed normally~~ based on the DEXCPT, MRIE, INTERVAL TIMER ~~COUNT~~, and REPORT COUNT values.
-1 to -64	Deactivate the TapeAlert flag specified by the absolute value of the TEST FLAG NUMBER field ~~in the log page~~. Deactivating the flag in this way is equivalent to performing the specified corrective action for that flag~~, allowing a true informational exception condition to be activated if the true error condition occurs for that flag~~.
32767	Activate all ~~of the~~ supported TapeAlert flags ~~in the log page~~. Report the informational exception condition for the TapeAlert flags with an additional sense code of FAILURE PREDICTION THRESHOLD EXCEEDED (FALSE) and ~~Once the supported TapeAlert flags are activated, they shall be processed normally~~ based on the DEXCPT, MRIE, INTERVAL TIMER ~~COUNT~~, and REPORT COUNT values.
all others	Return CHECK CONDITION status. Set the sense key to ILLEGAL REQUEST and the additional sense code to INVALID FIELD IN PARAMETER LIST ~~Reserved~~.

SPC-3 defines the effect of setting the TEST bit to zero.

~~A TEST bit of zero specifies the target shall not generate any test/false informational exception conditions. The value of the TEST bit returned by a MODE SENSE command shall be zero.~~

EDITORIAL NOTE: The second sentence deleted above, regarding the value of the TEST bit returned by a MODE SENSE command, may have been due to an assumption that the device server will reset the test bit in all cases rather than the limited set of circumstances actually described. I see no benefit in keeping it. Since we've extended auto-resetting to all cases, the sentence is not necessary because the auto-reset will make the bit zero.

See SPC-3 for a description of the DEXCPT bit~~, EWASC bit, EBF bit, PERF bit~~. A device server shall not report non-TapeAlert informational exceptions with an additional sense code of FAILURE PREDICTION THRESHOLD EXCEEDED if the DEXCPT bit is set to zero and the TASER bit in the Device Configuration Extension mode page is set to zero (see 8.3.y).

See SPC-3 for a description of the MRIE field. ~~If the informational exception condition was generated by an event that caused a real CHECK CONDITION to occur, then this real CHECK CONDITION shall over ride (i.e., be used instead of) the CHECK CONDITION defined in MRIE modes 01h to 05h.~~ For MRIE modes 02h ~~01h~~ to 06h, an additional sense code of FAILURE PREDICTION THRESHOLD EXCEEDED specifies a TapeAlert event has occurred on the device. Detailed information about the event is stored in the TapeAlert log page. If multiple TapeAlert flags are active simultaneously, the device server shall report a single informational exception condition. ~~Note that the MRIE modes do not effect the logging of events in the TapeAlert log page.~~

EDITORIAL NOTE: I've struck the second sentence in the paragraph above because I believe that SPC-3 covers the same ground, albeit using somewhat convoluted text (see SPC3r21d, 7.4.11 Information Exceptions Control mode page).

NOTE: The value of the MRIE field does not affect parameters in the TapeAlert log page or the TapeAlert Response log page.

~~See SPC-3 for a description of the INTERVAL TIMER field.~~

The REPORT COUNT/TEST FLAG NUMBER field has a dual purpose~~ and is described as follows~~:

a) SPC-3 defines the operation of the REPORT COUNT/TEST FLAG NUMBER field if the TEST bit is set to zero. When reporting an informational exception condition associated with TapeAlert flags, upon activation of a TapeAlert flag the device server shall report to the application client the informational exception condition the number of times indicated by the value of the REPORT COUNT/TEST FLAG NUMBER field.

~~a) if the TEST bit is set to zero, the value of the REPORT COUNT/TEST FLAG NUMBER field represents the REPORT COUNT. The REPORT COUNT value shall be returned in response to a MODE SENSE command. When the REPORT COUNT field is set to zero, this specifies there is no limit on the number of times the target shall report the informational exception condition. When the REPORT COUNT field is not set to zero, this specifies the number of times to report an informational exception condition. If multiple TapeAlert flags are active simultaneously, the flags are reported as a single informational exception condition. The REPORT COUNT is controlled by the newest flag that is active.~~

b) If the TEST bit is set to one, the value of the REPORT COUNT/TEST FLAG NUMBER field represents the test flag number. ~~The TEST FLAG NUMBER value shall not be returned in~~ In response to a MODE SENSE command reporting parameters from the Informational Exceptions Control mode page, the device server shall set the value of the REPORT COUNT/TEST FLAG NUMBER field to zero. Table 71 defines ~~V~~valid values for the ~~of~~ TEST FLAG NUMBER field~~ are –64 to 64 and 32767~~. Negative numbers shall be represented using the 2's complement notation and shall be sign extended to 4 bytes. ~~If the TEST FLAG NUMBER is set to an invalid value, the device server shall return CHECK CONDITION. The sense key shall be set to ILLEGAL REQUEST and the additional sense code shall be set to INVALID FIELD IN PARAMETER LIST. If the device server does not support a valid TEST FLAG NUMBER and the TEST bit is set to one, the device server shall return CHECK CONDITION. The sense key shall be set to ILLEGAL REQUEST and the additional sense code shall be set to INVALID FIELD IN PARAMETER LIST.~~

EDITORIAL NOTE: I've struck the last three sentences in the paragraph above because I believe that SPC-3 covers the same ground (see SPC3r21d, 6.7 MODE SENSE).

8.4.1 VPD parameters overview and page codes (deficiency e)

Table 75 – Sequential-access device VPD page codes

Page code	VPD page name	Reference	Support Requirements
B0h	Sequential-access device capabilities VPD page	8.4.2	Optional
B1h	Reserved for this device type		
B2h	TapeAlert supported flags VPD page	8.4.z	Optional
B~~1~~2h – BFh	Reserved for this device type		

Deleted: B1h

8.4.z TapeAlert supported flags VPD page (deficiency e)

Table x+6 specifies the TapeAlert supported flags VPD page. This page provides the application client with the means to determine the TapeAlert flags supported by the device server.

Table x+6 TapeAlert Supported Flags VPD page

Bit Byte	7	6	5	4	3	2	1	0
0	PERIPHERAL QUALIFIER			PERIPHERAL DEVICE TYPE				
1	PAGE CODE (B2h)							
2	Reserved							
3	PAGE LENGTH (08h)							
4	FLAG01h	FLAG02h	FLAG03h	FLAG04h	FLAG05h	FLAG06h	FLAG07h	FLAG08h
5	FLAG09h	FLAG0Ah	FLAG0Bh	FLAG0Ch	FLAG0Dh	FLAG0Eh	FLAG0Fh	FLAG10h
6	FLAG11h	FLAG12h	FLAG13h	FLAG14h	FLAG15h	FLAG16h	FLAG17h	FLAG18h
7	FLAG19h	FLAG1Ah	FLAG1Bh	FLAG1Ch	FLAG1Dh	FLAG1Eh	FLAG1Fh	FLAG20h
8	FLAG21h	FLAG22h	FLAG23h	FLAG24h	FLAG25h	FLAG26h	FLAG27h	FLAG28h
9	FLAG29h	FLAG2Ah	FLAG2Bh	FLAG2Ch	FLAG2Dh	FLAG2Eh	FLAG2Fh	FLAG30h
10	FLAG31h	FLAG32h	FLAG33h	FLAG34h	FLAG35h	FLAG36h	FLAG37h	FLAG38h
11	FLAG39h	FLAG3Ah	FLAG3Bh	FLAG3Ch	FLAG3Dh	FLAG3Eh	FLAG3Fh	FLAG40h

The PERIPHERAL QUALIFIER field and the PERIPHERAL DEVICE TYPE field are defined in SPC-3.

The PAGE LENGTH field specifies the length of the following VPD page data. If the allocation length value in the INQUIRY command descriptor block is too small to transfer all of the VPD page data, the page length shall not be adjusted to reflect the truncation.

Each FLAGXX bit indicates whether the device server supports the corresponding TapeAlert flag or not. A supported TapeAlert flag has the corresponding FLAGXX bit set to one. A TapeAlert flag that the device server does not support has the corresponding FLAGXX bit set to zero.

Annex A (deficiency k)

(~~normative~~ informative)

A.1 Application client recommendations for using TapeAlert

A.1.1 Introduction to application client recommendations for using TapeAlert

The previous version of this standard included some recommendations for application client behaviour when using TapeAlert. This standard deprecates these recommendations as they fall outside of its scope. To minimise the effort of referencing these recommendations for those application client designers that have incorporated them into products, this standard gathers them together and presents them below.

A.1.2 Recommendations for using TapeAlert

Upon detecting an active TapeAlert flag, the application client should:

a) Communicate an error message (see Annex A), the flag's severity (see table x+2), and the applicable description for that severity (see table x+1) to the user interface; and

b) Store the error message, the flag's severity, and the applicable attributes for that severity in a log.

At the beginning of each set of TapeAlert error messages, the application client should identify the target device that initiated them. For a medium-related flag (e.g. flags 04h, 07h, and 0Fh), the application client should include the software label of the medium in the TapeAlert error message so that the user is aware what piece of media the error refers to. Such information may be displayed with the messages for other flags.

EDITOR'S NOTE: The example list of medium-related flags is not exhaustive.

Each time the application client reads the TapeAlert log page or the TapeAlert Response log page, it should check all returned flags (see table x+2) to detect active flags. More than one flag may be active at a time.

The information read in the TapeAlert flags should not, in itself, cause the application client to stop a data transfer sequence (e.g., a backup or restore operation).

EDITOR'S NOTE: The sentence above is a 'shall' statement in the current SSC-2 TapeAlert text.

A.~~1~~2 TapeAlert ~~log page parameter codes (~~flag~~s)~~ associated information

Table A.1 ~~specifies~~ lists the TapeAlert ~~log page parameter codes~~ flags and associated information.

Table A.1 — TapeAlert ~~log page parameter codes~~ associated information

EDITORIAL NOTE: Delete the Type and Flag Type columns from the table that follows.