

From: Paul R. Nitza  
 To: X3T9.2 SCSI Committee Members  
 Date: 18 February 1989  
 Subj: Review of SCSI-2 Revision 7

Below is a list of possible problems and areas of concern that I compiled while reviewing revision 7 of the SCSI-2 specification. I believe these things need to be investigated before we forward the document for public review and hope we can cover this list during the February plenary meeting. Those page number shown in bold may involve other than simple editorial changes.

<b>Section 5</b>
------------------

Page 5-1, Section 5.1.1 - BUS FREE Phase, Fourth Paragraph:

Paragraph 4 which lists the possible reasons for a BUS FREE phase should include the following:

- (7) after an ABORT TAG message is successfully received by a target
- (8) after a CLEAR QUEUE message is successfully received by a target

Page 5-4, Section 5.1.4.1 - Reselection, Third Paragraph:

Paragraph 3 is missing the sentence shown in italics below:

After the target detects the BSY signal is true, it shall also assert the BSY signal and wait at least two deskew delays and then release the SEL signal. *The target may then change the I/O signal and the DATA BUS. After the reselected initiator detects the SEL signal is false, it shall release the BSY signal. The target shall continue asserting the BSY signal until it relinquishes the SCSI bus.*

Page 5-12, Section 5.1.10 - Signal Restriction Between Phases, First Paragraph, Number (1):

REQB and ACKB should be added to the list of signals that shall not change.

Page 5-15, Section 5.3 - SCSI Bus Phase Sequences, Third Paragraph:

In the last sentence the LINKED COMMAND COMPLETE message should be removed since it is not followed by a BUS FREE phase. Normally the LINKED COMMAND COMPLETE message is followed by a COMMAND phase.

Page 5-19, Last Paragraph on the Page:

A note should be added that if tagged queuing is being used the queue tag message must follow the IDENTIFY message. Suggested wording:

If the first message is an IDENTIFY message, then it may be immediately followed by other messages, such as the first of a pair of SYNCHRONOUS DATA TRANSFER REQUEST messages. Note: If tagged queuing is implemented the queue tag message must immediately follow the IDENTIFY message (see 5.6.17.). The IDENTIFY message establishes a logical connection between the initiator and the specified logical unit or target routine within the target known as an I\_T\_L nexus or I\_T\_R nexus. After the RESELECTION phase, the target's first message shall be IDENTIFY. This allows

Page 5-21, Section 5.6.2 - ABORT TAG, First Paragraph:

In the first sentence the words "I/O Process queuing" should be replaced by "tagged queuing". The ABORT TAG message is only used for tagged commands not both tagged and untagged queuing.

Page 5-21, Section 5.6.4 - CLEAR QUEUE, First Paragraph:

In the first sentence the words "tagged queuing" should be replaced by "I/O process queuing". The CLEAR QUEUE message can be used for both tagged and untagged queuing.

With the above change the third sentence of this same paragraph should also be changed to eliminate the reference to the ABORT TAG message. Suggested wording for the first four sentences of this paragraph are:

The CLEAR QUEUE message shall be implemented if tagged I/O process queuing is implemented. The target shall go to the BUS FREE phase following successful receipt of this message. The target shall perform an action equivalent to receiving a series of ABORT TAG messages ~~for all I/O processes currently queued for the logical unit or target routine from all~~ each initiators. All I/O processes, from all initiators, in the queue for the specified logical unit or target routine shall be cleared from the queue. All executing I/O processes.....

Section 6
-----------

Page 6-5, Section 6.2.2 - Logical Unit Number, First Paragraph:

Since the IDENTIFY message is now mandatory the last part of the second sentence in this paragraph should be deleted as shown below:

See the IDENTIFY message for a description. The target shall ignore the logical unit number specified within the command descriptor block ~~if an IDENTIFY message was received~~. It is recommended that the logical unit number in the command descriptor block be set to zero.

Page 6-7, Section 6.3 - Status, First Paragraph:

This paragraph should be reworded as shown below to include the new SCSI-2 terminating conditions:

A status byte shall be sent from the target to the initiator during the STATUS phase at the termination of each command as specified in Tables 6-6 and 6-7 unless the command is cleared by ~~an ABORT message, by a BUS DEVICE RESET message, a hard reset condition, or by an unexpected BUS FREE condition (see 5.1.1)~~ one of the following conditions:

- (1) an ABORT message
- (2) a BUS DEVICE RESET message
- (3) a hard reset condition
- (4) an unexpected BUS FREE condition (see 5.1.1)
- (5) an ABORT TAG message
- (6) a CLEAR QUEUE message
- (7) a RELEASE RECOVERY message.

Page 6-12, Section 6.5.2 - Duplicate Nexus:

This whole section was rewritten for revision 7 and I believe the meaning has been changed. This section now says that a duplicate nexus cannot be established, but section 6.8.1 and 5.6.1 specifically allow a duplicate nexus to be established for the purpose of aborting a command. This section used to be titled "Redundant Commands to an Active Logical Unit". I think the wording in revision 6/6A should be restored with the following changes (Note: the words in italics are only required if the Terminate I/O proposal is excepted):

**6.5.2 Redundant Commands to an Active Logical Unit**

~~An initiator should never attempt to send a second command to a logical unit until the command in progress is terminated, unless both the initiator and the target support tagged command queuing against a logical unit. The initiator is normally informed that a command is terminated by the target when the target returns a COMMAND COMPLETE message. Termination may also result from a hard RESET condition or from unusual soft RESET conditions.~~

An initiator should never attempt to send a redundant command to a logical unit until the first command in progress has completed (see 5.1.1 and 6.3). For targets that do not support any type of command queuing a redundant command is defined as a command received from any initiator for a logical unit that already has a command in progress. For targets that support untagged queuing a redundant command is defined as a command from an initiator for a logical unit that already has a command in progress from the same initiator. For targets that support tagged command queuing a redundant command is defined as a command from an initiator for a logical unit that has the same queue tag number as another command already in progress from the same initiator.

**IMPLEMENTORS NOTES:**

(1) Attempting to send a redundant command to a logical unit is considered a catastrophic failure on the part of the initiator. Therefore, special vendor-specific error recovery procedures may be required to establish guarantee the data integrity of the data on the logical unit in such a case on the medium. The target may return additional sense data may be presented to assist in to aid in this error recovery procedure (e.g., sequential-access devices may return the residue of blocks remaining to be written or read at the time the second command was received).

(2) It is permissible for the initiator to establish a duplicate nexus (i.e. establish a nexus with the same I T x y as another I/O process) for the sole purpose of sending an ABORT, ABORT TAG, BUS DEVICE RESET, CLEAR QUEUE or TERMINATE I/O PROCESS message (see 5.6.1 and 5.6.xx). In this case the message should be sent within the same message phase as the IDENTIFY and queue tag messages.

~~If a target receives an untagged command from an initiator for a logical unit that already has a command in progress from the same initiator a redundant command and if no a soft RESET condition has not occurred since the original command was identified, the target shall take the following action:~~

~~The target shall internally stop execution of both commands and shall present abort both commands and shall return CHECK CONDITION status. The sense key on behalf of these commands shall be set to ABORTED COMMAND and the additional sense code shall be set to OVERLAPPED COMMANDS ATTEMPTED. Only one status is returned for both commands.~~

~~If a target receives a command from an initiator for a logical unit that already has a command in progress from the same initiator a redundant command and if a soft RESET condition has occurred, the soft RESET action (3) shall be taken as the target shall meet the requirements of 5.2.2.2. described in section 5.2.2.2. The target shall clear the original command and perform the new command instead. The assumption is made that the new command is a repetition of the original command in this special case.~~

Page 6-12, Section 6.5.3 - Selection of an Invalid Logical Unit, Second Paragraph, Number (2):

The following changes should be made in the first sentence of number (2) so it corresponds with the wording used in the INQUIRY command:

(2) the target supports the logical unit, but the ~~logical unit~~ peripheral device is not attached to the target. In response to an INQUIRY command.....

Page 6-13, First Paragraph, Number (3):

The following changes should be made in the first sentence of number (3) so it corresponds with the wording used in the INQUIRY command:

(3) the target supports the logical unit and ~~it is attached, but~~ the peripheral device is attached but not operational. In response to an INQUIRY command.....

Page 6-13, Second Paragraph, Number (4):

The following changes should be made in the first sentence of number (4) so it corresponds with the wording used in the INQUIRY command:

(4) the target supports the logical unit but is incapable of determining ~~the~~ if the peripheral device is attached or is not operational when it is not ready. In response to an INQUIRY command.....

Page 6-14, Section 6.5.5 - Asynchronous Event Notification, Sixth Paragraph:

The words "sequential-access" should be removed from the first sentence of this paragraph since many devices can implement write caching. The new sentence would read:

An example of the first case above is a ~~sequential-access~~ device that implements write caching. Notification of an unable to write condition can be sent to.....

Page 6-14, Section 6.5.5 - Asynchronous Event Notification, Eighth Paragraph:

The word "tape" should be replaced by the words "sequential-access device". The new sentence would read:

An example of the fourth case above is a ~~tape~~ sequential-access device performing a REWIND command with the immediate bit set. Asynchronous event notification.....

Page 6-16, Section 6.6- Contingent Allegiance Condition, First Paragraph:

Suggest the following wording for the last sentence in the first paragraph:

Those targets that do not maintain independent recovery operations, including sense information, for each I\_T\_x nexus shall implement contingent allegiance. This guarantees that error information is available to the initiator ~~returned~~ that received the CHECK CONDITION status.

Page 6-17, Third Paragraph:

Suggest the third paragraph be modified as shown below:

Only untagged commands received from the device to which the INITIATE RECOVERY message was sent shall be executed. If the initiator sends a tagged command the target shall respond with a MESSAGE REJECT message to the queue message and continue the I/O process as if it was an untagged I/O process. Any commands remaining in the queue after the extended contingent allegiance condition is cleared shall be executed as if the event had not occurred.

Page 6-18, Section 6.8.2- Tagged Queuing, First Paragraph:

Suggest the following be added to paragraph 1

Tagged queuing allows a target to accept multiple commands from the same or different initiators until the logical unit's I/O process queue is full. A new I/O process may be initiated any time the BUS FREE phase exists. If the disconnect privilege is granted, the target may elect to disconnect after the queue tag message has been received, but prior to requesting the command descriptor block. If the disconnect privilege is not granted for a tagged command the target shall return BUSY status to the new I/O process.



Page 6-18, Section 6.8.2- Tagged Queuing, Eighth Paragraph:

Paragraph 8 states "A command received without a queue tag message is managed by the rules for untagged queuing". However, there is no place in the document where it states what should happen when the target receives an untagged command (i.e. when should it be executed) while it has 1 or more tagged commands. Should the untagged command be treated as a head of queue command, an ordered queue command or a simple queue command? I am open to treating an untagged command in any one of the three manners and suggest paragraph 8 be changed as follows:

A command received without a queue tag message is managed by the rules for untagged queuing target as if a HEAD OF QUEUE TAG message was received. Note: Only one untagged command for each I T x nexus may be accepted at a time (see 6.8.1).

Page 6-22, Section 6.9- Unit Attention Condition, Third Paragraph:

Paragraph 3 of this section precludes targets from queuing unit attention conditions. Under certain conditions this may be desirable or even necessary (e.g. a power on unit attention followed by a microcode has changed unit attention). I suggest that wording of paragraph 3 be changed as shown below and an implementors note be added.

If an INQUIRY command is received from an initiator with a pending unit attention condition (before the target reports CHECK CONDITION status), the target shall perform the INQUIRY command, report GOOD status, and shall not clear the unit attention condition. If the INQUIRY command ~~or any other command~~ is received after the target has reported CHECK CONDITION status to the initiator for a pending unit attention condition, then the unit attention condition shall be cleared, the target shall perform the command, and the target shall report GOOD status. If any other command is received after the target has reported CHECK CONDITION status to the initiator for a pending unit attention condition, then the unit attention condition shall be cleared, and if no other unit attention condition is pending the target shall perform the command and report GOOD status. If another unit attention condition is pending the target shall not perform the command and shall report CHECK CONDITION status.

Implementors Note: Some targets may queue unit attention conditions. After the first unit attention condition is cleared, another (different) unit attention condition may exist (e.g. a power on unit attention followed by a microcode has changed unit attention). The initiator can insure all unit attention conditions have been cleared by repeatedly issuing REQUEST SENSE commands until a sense key other than UNIT ATTENTION is returned.

Section 7
-----------

Page 7-8, Section 7.2.3- COPY Command:

In the list of COPY function codes just before section 7.2.3.1 the destination direct-access device is listed as writable. This is not true for the COMPARE command which references this section. I suggest that the following paragraph be added after the list:

For the COMPARE command the destination direct-access device does not have to be a writable device type.

Page 7-11, First Paragraph:

The first paragraph on this page is missing the words shown in italics below:

The direct-access device number of blocks field specifies the number of *blocks* in the current segment to be copied. A value of zero indicates that no blocks shall be transferred in this segment.

Page 7-11, Second to last Paragraph:

The second to last paragraph on this page is missing the words shown in italics below:

The number of blocks field specifies the number of blocks to be transferred *from the source* device during this segment. A value of zero indicates that no blocks shall be transferred.

Page 7-13, Second full Paragraph:

The second full paragraph on this page is missing the words shown in italics below:

The number of blocks field specifies the number of blocks to be transferred *from the source* device during this segment. A value of zero indicates that no blocks shall be transferred.

Page 7-27, LOG SELECT Command, Second to last Paragraph:

The second to last paragraph on this page is missing the word "be" in the first line and the last sentence is redundant since this section for the LOG SELECT command. The suggested changes are shown below:

The current cumulative values may be updated by the target to reflect the cumulative number of events experienced by the target. Fields in the parameter control byte (7.3.2) of each log parameter control the updating and saving of the current cumulative parameters. ~~Current cumulative values can also be updated by the initiator via the LOG SELECT command.~~

Page 7-28, LOG SELECT Command, First Paragraph:

In the first paragraph the word "allocation" should be changed to "parameter list" as shown below:

The target shall set all cumulative parameters to their default cumulative values in response to a LOG SELECT command with the PC field set to 11b and the ~~allocation~~ parameter list length set to zero.

Page 7-28, LOG SELECT Command, Last Paragraph:

Suggest the wording for generating a unit attention condition in the last paragraph be changed as shown below:

The target may provide independent sets of log parameters for each logical unit or for each combination of logical units and initiators. If the target does not support independent sets of log parameters and any log parameters are changed that affect other initiators, then ~~if the target shall generate an unit attention condition for all initiators except the one that issued the LOG SELECT command (see 6.1.3) if any log parameters are changed that affect other initiators.~~ This unit attention condition is returned with an additional sense code of LOG PARAMETERS CHANGED.

Page 7-29, Section 7.2.7- LOG SENSE Command, Second Paragraph:

Paragraph 2, number (1) (PPC bit set to 1) seems to say that the LOG SENSE command will return a 2 byte integer value instead of a page. However, section 7.3.2 states that the LOG SENSE command will always return page formatted data (or no data) according to the page code specified in the CDB. Is a PPC bit of 1 really supposed to return a two byte value as this paragraph indicates?

Page 7-36, Section 7.2.10.4- Saved Values, Second Paragraph (Implementors Note):

The addition of the words "if the page affects the medium format" added in rev 7 change the meaning of this implementors note. These new words imply that the pages are not saved until after a FORMAT UNIT command if the medium format is changed. This conflicts with sections 7.2.8 (page 7-32, paragraph 3) and section 8.3.3.3 (page 8-75, implementors note at the bottom of the page) that state that all pages will be saved when a MODE SELECT command with the SP bit set to 1 is processed by the target. In addition, these pages may or may not be saved at the next FORMAT UNIT command depending on the value of the Disable Saved Pages bit in the defect list header.

Page 7-50, First Paragraph:

The last sentence of the first paragraph on this page states that "The subsequent execution of a REQUEST SENSE command shall recover the deferred error sense information". This statement conflicts with section 6.9 (Unit Attention Condition, page 6-22) that states that a target may "(2) report the unit attention condition, discard any pending sense data and clear the unit attention condition for that initiator". Two possible solutions are: (1) add a note that deferred sense information may be lost if a unit attention condition is pending or (2) remove the option that allows the target to destroy the sense information (page 6-22, last paragraph, number 2). I would prefer the latter option since discarding error information (any error information) does not sit well with me.

Page 7-50, Last Paragraph (Number (5)):

Suggest the wording of this paragraph be changed as shown below. Two changes have been made, change the word operating to executing for consistency and correcting an editing mistake that occurred in rev 7 (change the word deferred to current in the middle of the paragraph).

(5) If a current command has not yet started executing, and a deferred error occurs, the command shall be terminated with a CHECK CONDITION status and deferred error information posted in the sense data. By convention the current command is considered to have started execution if the target has changed phase from the COMMAND phase to the next normal phase of the command sequence. If a deferred error occurs while a current command is ~~operating~~ executing and the current command has been affected by the error, the command shall be terminated by a CHECK CONDITION status and ~~deferred~~ current error information shall be posted in the sense



data. In this case, if the current error information does not adequately define the deferred error condition, a deferred error may be posted after the current error information has been recovered. If a deferred error occurs while a current command is ~~operating~~ executing and the current command completes successfully, the target may choose to post the deferred error information after the completion of the current command.

Page 7-52, Unit Attention Sense Key:

The reference to section 6.1.3 should be changed to reference section 6.9.

Page 7-67, Fourth Paragraph:

The words "LOG SELECT and" should be added as shown below.

For cumulative log parameter values (indicated by the PC field of the LOG SELECT and LOG SENSE command descriptor block), the disable update (DU) bit is defined as follows:

Page 7-69, The Second and Fourth Paragraphs:

The reference to section 7.5.3 in both of these paragraphs should be changed to reference section 7.3.3.1.

Page 7-72, Third Paragraph:

This paragraph may conflict with paragraph 2 on page 7-69. Page 7-69 does not mention that ALL data counters cease operation when any one data counter in a page reaches its maximum value. Why do we want to suspend the counting of other errors just because one error counter reached its maximum value? If we do this for the Error Counter pages shouldn't we also do this for the Buffer Over-Run/Under-Run counter pages? I suggest that this paragraph be deleted since this information is covered on page 7-69. If this paragraph is not deleted I suggest the following changes:

When any counter ~~in a log~~ within an error counter page reaches its maximum value, incrementing of all counters in that log page shall cease until re-initialized by the initiator via a LOG SELECT command. If the RLEC bit of the control mode page is one, then the target shall report the exception condition as described in ~~7.3.4.1~~ 7.3.3.1.

Page 7-76, Second Paragraph:

The last sentence in the second paragraph is redundant and should be deleted.

Page 7-77, First Paragraph following Table 7.65:

The reference to section 7.3 should be changed to reference section 7.3.3.

Page 7-77, Last Paragraph:

In the last sentence of this paragraph the words "MODE SENSE" should be changed to "MODE SELECT".

Page 7-78, First Paragraph:

In the first and last sentence of this paragraph the words "parameter length field" should be changed to "page length field".

Page 7-86, Fifth Paragraph:

The first sentence should be changed as follows:

The page length field specifies the length of the ~~supported page list~~ following page data. If the allocation length of the command descriptor block is too small to transfer all of the page, the page length shall not be adjusted to reflect the truncation.

Section 8
-----------

Page 8-17, Fifth and Sixth Paragraphs:

The error "PRIMARY DEFECT LIST NOT FOUND" is now an ASCQ (changed between rev 6 and rev 7). This error should be changed to "DEFECT LIST NOT FOUND" which is the correct additional sense code.

Actually, a cleaner implementation of these errors would be to add two additional ASCQs:

19	02	PRIMARY DEFECT LIST ERROR
19	03	GROWN DEFECT LIST ERROR

Then change paragraphs 4, 5 and 6 on page 8-17 as follows:

The stop format (STPF) bit controls the behavior of the target when one of the following events occurs:

- (1) The target has been requested to use the primary defect list (DPRY is set to zero) or the grown defect list (CmpLst is set to zero), and the target cannot locate the list nor determine whether the list exists.
- (2) The target has been requested to use the primary defect list (DPRY is set to zero) or the grown defect list (CmpLst is set to zero), and the target ~~cannot locate the list nor determine whether the list exists~~ encounters an error while accessing the defect list.

A STPF bit of zero indicates that, if one or both of the above conditions occurs, the target shall continue to execute the FORMAT UNIT command. The target shall return CHECK CONDITION status at the completion of the FORMAT UNIT command. The sense key shall be set to RECOVERED ERROR and the additional sense code shall be set to to either ~~PRIMARY DEFECT LIST NOT FOUND~~ if condition one occurred, or DEFECT LIST ERROR if condition two occurred.

A STPF bit of one indicates that, if one or both of the above conditions occurs, the target shall terminate the FORMAT UNIT command with CHECK CONDITION status. The sense key shall be set to MEDIUM ERROR and the additional sense code shall be set to to either ~~PRIMARY DEFECT LIST NOT FOUND~~ if condition one occurred, or DEFECT LIST ERROR if condition two occurred.

Page 8-22, Second Paragraph (Implementors Note):

The Implementors Note on page 8-22 should be moved to page 8-21 since this is a global note that applies to all defect descriptor formats.

Page 8-24, Last Paragraph (Implementors Note):

The last sentence of this note says that the release of an overlapped area does not effect the lock of another initiator. What about an overlapped lock from the same initiator? I suggest the note be reworded as follows:

IMPLEMENTORS NOTE: Multiple locks may be in effect from ~~more than one~~ the same or different initiators. ~~Locks may overlap~~ Locked areas may partially or completely overlap other locked areas. An unlock of an ~~overlapped~~ area does not release any overlapping locked areas ~~the lock of another initiator.~~

Page 8-27, Last Paragraph (Implementors Note):

The implementors note at the bottom of the page should be changed as shown and moved below the first paragraph.

IMPLEMENTORS NOTE: Targets with cache memory may have default values for ~~these~~ the cache control bits which may affect the READ(6) command, however no default value is defined by this standard. If explicit control is required, the READ(10) command should be used.

Page 8-28, Second Paragraph:

The second paragraph should be changed as shown below to take into account the retention priority fields in the cache page.

A disable page out (DPO) bit of one indicates that the target shall assign the logical blocks accessed by this command the lowest priority for being fetched into or retained by the cache. The DPO bit set to one overrides any retention priority specified in the cache page (see 8.3.3.1). A DPO bit of zero indicates the priority ~~assigned~~ shall be determined by the ~~target in a vendor-specific manner~~ retention priority fields in the cache page. All other aspects of the algorithm implementing the cache replacement strategy are not defined by this standard.

Page 8-32, Bottom of the Page:

The following Implementors note was removed from rev 7 and should be put back in.

IMPLEMENTORS NOTE: The vendor-specific use of the logical block format is used for the return of information specific to different implementations. There is no universal model that sensibly defines the meaning of the logical block address of a defect. Usually a defect that has been reassigned no longer has a logical block address.

Page 8-33. First Paragraph:

The first sentence of this paragraph states that the physical format may not include defects in initiator inaccessible areas while the second sentence states that it will. I suggest the following change:

A defect list returned in ~~either the vendor specific use of the~~ logical block format or physical sector format may not include defects in initiator inaccessible areas. A defect list returned in ~~either the bytes-from-index or physical sector~~ format shall be a complete list. The complete list may contain defects in areas not within the capacity returned in the READ CAPACITY command.

Page 8-34. First Paragraph:

An addition, as shown below, should be added to the first paragraph to further clarify the Read Long command.

The READ LONG command (Table 8-20) requests that the target transfer data to the initiator. The data passed during the READ LONG command is implementation specific, but shall include the data bytes and the ECC bytes recorded on the medium for one logical block. The most recent data written in the addressed logical block shall be returned.

Page 8-38. Last Paragraph (Implementors Note):

I found this implementors note hard to understand and follow. I suggest the following rewording in an attempt to make it easier to understand.

IMPLEMENTORS NOTE: When a target implements independent storage of mode parameters for each initiator, a third-party RESERVE command ~~effects a transfer of copies~~ the current mode parameters of the initiator that sent the RESERVE command to the current mode parameters for the initiator specified as. ~~Those set up by the initiator of the RESERVE are to be set as the mode parameters used for commands from~~ the third-party device (usually a copy master device). The target creates a unit attention condition to notify the third-party device of changed mode parameters due to the ~~transfer reservation~~. A successful third-party RELEASE command ~~leaves the transferred parameters intact~~ does not return the third-party devices current mode parameters back to their previous (before the RESERVE command) values. This allows the mode parameters to be transferred and a following copy operation to take place without having the device reserved. The third-party device can issue MODE SENSE and MODE SELECT commands to query and modify the mode parameters.

Page 8-53. Last Paragraph (Implementors Note):

The implementors note at the bottom of the page should be changed as shown and moved below the first paragraph.

IMPLEMENTORS NOTE: Targets with cache memory may have default values for ~~these the~~ cache control bits which may affect the WRITE(6) command, however no default value is defined by this standard. If explicit control is required, the WRITE(10) command should be used.

Page 8-60, Table 8-42, Translate Address Page:

The "Page Parameter Length" field should be changed to "Page Length" or the field name in paragraph two should be changed from "Page Length" to "Page Parameter Length".

Page 8-63, First Paragraph after the Notes for Table 8-44:

Why does this paragraph say the device specific parameter field is reserved when the following table and next five paragraphs specify what this field is used for?

Page 8-79 and 8-80:

The last paragraph on page 8-79 and the first paragraph on page 8-80 are the same paragraph.

That's all Folks -- Thank-You