Date: November 17, 2000

To: T10 Committee (SCSI)

From: Rob Basham (IBM)

Subject: Explicit State Change Proposal for SSC-2

1. Introduction

1.1 Author Information

Document Owner: Rob Basham

Document Owner Email Address: robbyb@us.ibm.com

Document Owner Phone Number: 520-799-4923

1.2 Change History

1.2.1 Revision 0

- Initial Proposal

1.2.2 Revision 1

- Command Descriptor Blocks, etc, added

1.2.3 Revision 2

- Explicit block address model added
- Implicit block address model added
- Write append bit added
- Removed filemark encountered check for read type commands running the explicit block address model
- Miscellaneous fixes

1.3 Purpose

This document describes a set of changes to the SCSI Tape command set that makes all state change requests explicit. This new command set is called the explicit command set. The legacy command set as currently documented in SSC is called the implicit command set. The implicit command set is such that the meaning of many commands is dependent on preceding commands. For example, the READ command does not contain a logical block number. Which block is read depends on the current position on tape.

The goal of the changes is to allow a device driver to safely redrive commands, perform alternate pathing when a path has a hard failure, perform command queueing where in-order delivery is not guaranteed, etc. without the need for complicated logic to regain positioning agreements between the Initiator and the Target.

2. Implicit State Issues

The following are some of the problems in the current command set that this proposal resolves:

- Relative nature of the SPACE command
- Implicit logical block address for READ, SPACE, WRITE, WRITE FILEMARKS, VERIFY, RECOVER BUFFERED DATA and ERASE commands.

2.1 Relative Nature of SPACE Command

The SPACE command is problematic in that it asks the drive to space relative to the current position on tape. If for some reason that command failed on the interface, redriving the command is problematic. Did the SPACE command get to the tape drive or not? On a command timeout at the device driver, there is currently no way to simply redrive the command because the driver has no idea if the first time around any movement in logical block position took place.

2.2 Implicit Logical Block Number Problem

Many of the commands in the SSC command set have no logical block number associated with them. The drive executes the command based on the current logical block position on tape. If something happens where the Initiator and Target get out of synchronization, there is nothing obvious in the commands themselves that identifies a problem.

3. Proposed Changes

Below are proposed additions to the SSC-2 Revision 0 document.

Global change request: Search for all instances of the term "block number" and replace with "logical block address".

Section 4.1 on page 7 needs some introductory words on the implicit and explicit address command sets and their associated models.

The following definitions need to be aded to section 5.1 on page 8.

logical block address - A unique identifier for a logical block. The first logical block in a partition has an address of 0. If a logical block has an address of number n, the logical block immediately before that block has an address of n-1 and the logical block immediately after that block has an address of n+1. Filemarks and setmarks have a block address.

logical file address - A unique identifier for a range of zero or more blocks ending in a filemark. The first logical file address in a partition has an address of zero. If a logical file has an address of n, the logical file immediately before that has an address of n-1 and the logical file immediately after has an address of n+1.

logical set address - A unique identifier for a range of zero or more blocks ending in a setmark. The first logical set address in a partition is zero. If logical set has an address of n, the logical set immediately before that has an address of n-1 and the logical set immediately after has and address of n+1.

explicit address command set - The command set in which logical block addresses are explicitly included in the command descriptor blocks where appropriate. Support for this command set is mandatory.

implicit address command set - This optional command set is included for support with legacy systems. A logical block address is generally not included in the command descriptor blocks and positioning is implied based on the current position. Command queueing with this command set is prohibited.

Section 5.2.4, first paragraph, remove the second sentence. Discussion about which commands manage control and transfer is moved to the explicit address and implicit address command set sections.

Section 5.2.8 needs text to define the relationship of identifiers to their respective logical block address, logical file address and logical set address.

Remove Section 5.2.6, on Tagged Command Queueing. This material will be covered in the implicit and explicit address command set sections.

Remove Section 5.2.8, on Direction and Position Definitions. This material will be covered in the implicit and explicit address command set sections.

After 5.2.8 Direction and Position Information, add the following sections:

3.0.1 Explicit Address Model

With the explicit address command set, positioning is specified using the logical block address by the commands themselves. Current position on tape is not usually relevent.

The explicit address command set is listed in the table below:

Table 1 - Explicit Address Commands Set

Command Name	Operation code	Write type command	Read type command	Type
ERASE (16)		Y	N	M
FORMAT MEDIUM		Y	N	O
INQUIRY		N	N	O
LOAD UNLOAD		N	N	O
LOCATE (16)		N	N	O
LOG SELECT		N	N	O
LOG SENSE		N	N	O
MODE SELECT (6)		Y or N[1]	Y or N[1]	M
MODE SELECT (10)		Y or N[1]	Y or N[1]	O
MODE SENSE (6)		N	N	M
MODE SENSE (10)		N	N	O
MOVE MEDIUM		N	N	O
MOVE MEDIUM ATTACHED		N	N	O
PERSISTENT RESERVE IN		N	N	O
PERSISTENT RESERVE OUT		N	N	O
PREVENT ALLOW MEDIA REMOVAL		N	N	O
READ (16)		N	Y	M
READ BLOCK LIMITS		N	N	M
READ BUFFER		N	N	O
READ ELEMENT STATUS		N	N	O
READ ELEMENT STATUS ATTACHED		N	N	O
READ POSITION		N	N	M
READ REVERSE (16)		N	Y	O
RECEIVE DIAGNOSTIC RESULTS		N	N	O
RECOVER BUFFERED DATA		N	Y	O
RELEASE(6)		N	N	M
RELEASE(10)		N	N	M
REPORT DENSITY SUPPORT		N	N	M
REPORT LUNS		N	N	M
REQUEST SENSE		N	N	M
RESERVE(6)		N	N	M
RESERVE(10)		N	N	M
REWIND		N	N	M
SEND DIAGNOSTIC		Y or N[1]	Y or N[1]	M
SET CAPACITY		Y	N	O
TEST UNIT READY		N	N	M
VERIFY (16)		N	Y	O
WRITE (16)		Y	N	M
WRITE BUFFER		N	N	O
WRITE FILEMARKS (16)		Y	N	M

Key: M = Command implementation is mandatory.
O = Command implementation is optional.

Notes:

1. This command has some specific actions that fall under write type commands and some that fall under read type commands.

The explicit address model is designed for both implementations that support command queueing and those that don’t.

Below is a description of the the explicit address model as applied in cases where there is no command queueing:

On read type commands (see Table 1 for list of read type commands) any positioning and the data to be transferred are specified explicitly by the command descriptor block. The position on tape is still maintained on a command by command basis. A command may be appropriately issued from the host to read any logical block within any partition at any time. In that respect, current position on tape has lost significance. It is expected that blocks will still be generally read sequentially, but mainly for performance reasons.

Similarly on write type commands (see Table 1), current position is no longer significant. The location at which to write is explicitly contained in the command descriptor blocks, specified using a logical block address.

Below is a description of the explicit address model as applied to command queueing:

Generally, transport mechanisms deliver commands in order. The explicit address model is built with the assumption that the transport layer used with the command set may not deliver queued commands in the proper order. The model is broken into three states: neutral, write capable, and read capable.

There is no direct transition between the write capable state and the read capable state. When in the neutral state, the drive may transition to either the write capable state or the read capable state. A drive is in the neutral state at power up. It transitions to the neutral state from either the read capable or write capable state by successful completion of a LOAD UNLOAD command, REWIND command, or a LOCATE command with the CODE field set to the 0 (the locate block option). The LOAD UNLOAD, REWIND, and LOCATE commands shall not be queued at any time. MODE SELECT and SEND DIAGNOSTIC commands shall not be queued in cases where the commands may cause logical or physical tape positioning or affect the state of buffered data. A transition from the neutral state to the write capable state is made by issuing any queued write type command. A transition from the neutral state to the read capable is made by issuing any queued read type command. See Table 1 for the definition of which commands are read type and which are write type.

When in the read capable state, the drive may execute any command except write type commands. If a write type command or a command from the implicit address model is received, the drive shall return status of CHECK CONDITION with associated sense data sense key of ILLEGAL REQUEST and an additional sense code qualifier of SEQUENTIAL POSITIONING ERROR. On read type commands any positioning and the data to be transferred is specified explicitly by the command descriptor block. The position on tape is still maintained on a command by command basis. A command may be appropriately issued by the host to read any logical block within any partition at any time. In that respect, current position on tape has lost significance. It is expected that blocks will still be generally read sequentially, but mainly for performance reasons.

When in the write capable state, the drive may execute any command except read type commands. If a read type command or a command from the implicit address model is received, the drive shall return status of CHECK CONDITION with associated sense data sense key of ILLEGAL REQUEST and an additional sense code qualifier of SEQUENTIAL POSITIONING ERROR. By their very nature, writes are sequential after the append point since all data within that partition following logical block address specified in the write is rendered invalid. It is important that the drive be aware of where the append is. With queued commands in an environment where in order deliver is not guaranteed, one can imagine a scenario where the first command sent to the drive is misrouted and a subsequent command gets there first. Because of this requirement, a special append point bit has been added to all write type commands so that the drive can know for sure which write commands are at non-sequential append points. A write type command with an append point bit set can be accepted out of sequence for any block in any partition. Subsequent commands without the append point bit would be expected to follow in sequence. Commands not received in strict sequence can be either held to see if subsequent commands come in that put them in sequence or they may be rejected if there is some confidence that the transport layer delivers commands in order. In the case of the reject because of a sequence mismatch, a CHECK CONDITION shall be returned with sense key ILLEGAL REQUEST and an additional sense code qualifier of SEQUENTIAL POSITIONING ERROR. During the Auto Contingent Allegiance created as a result of reporting the error, it is expected that the intiator would abort all outstanding write commands for which the append bit is not set with a logical block address greater than the one on which the SEQUENTIAL POSITIONING ERROR was reported. If a command is aborted, it can also leave a gap that would cause the SEQUENTIAL POSITIONING ERROR to be reported. In any case, the drive shall not attempt to execute an out of sequence write type command for which the append point bit is not set. The drive may attempt data transfer for write commands that do not have the write append bit set, but shall not send status for a command until all commands with a small logical block address in the request are complete and status has been sent. There is no requirement for a wait on confirmation that the host has received ending status.

When in the neutral state, the drive may execute only queued commands that are of neither the read nor the write type. It may execute unqueued commands of any type.

3.0.2 Implicit Address Model

The implicit address command set is defined for legacy applications and is optional. With this model, positioning is specified based on the current position on tape and is implicit. In general, the commands have no explicit positioning information. Command queueing is prohibited using this model. This model defines tape behavior as it has previously been known in the prior version of the SSC document and in SCSI-2.

The implicit address command set is listed in the table below:

Table 2 - Implicit Address Command Set

Command Name	Operation Code	Write Type Command	Read Type Command	Type
ERASE (6)		Y	N	M
FORMAT MEDIUM		Y	N	O
INQUIRY		N	N	M
LOAD UNLOAD		N	N	O
LOCATE (10)		N	N	O
LOCATE (16)		N	N	O
LOG SELECT		N	N	O
LOG SENSE		N	N	O
MODE SELECT (6)		Y or N[1]	Y or N[1]	M
MODE SELECT (10)		Y or N[1]	Y or N[1]	O
MODE SENSE (6)		N	N	M
MODE SENSE (10)		N	N	O
MOVE MEDIUM		N	N	O
MOVE MEDIUM ATTACHED		N	N	O
PERSISTENT RESERVE IN		N	N	O
PERSISTENT RESERVE OUT		N	N	O
PREVENT ALLOW MEDIA REMOVAL		N	N	O
READ (6)		N	Y	M
READ BLOCK LIMITS		N	N	M
READ BUFFER		N	N	O
READ ELEMENT STATUS		N	N	O
READ ELEMENT STATUS ATTACHED		N	N	O
READ POSITION		N	N	M
READ REVERSE (6)		N	Y	O
RECEIVE DIAGNOSTIC RESULTS		N	N	O
RECOVER BUFFERED DATA		N	Y	O
RELEASE(6)		N	N	M
RELEASE(10)		N	N	M
REPORT DENSITY SUPPORT		N	N	M
REPORT LUNS		N	N	M
REQUEST SENSE		N	N	M
RESERVE(6)		N	N	M
RESERVE(10)		N	N	M
REWIND		N	N	M
SEND DIAGNOSTIC		Y or N[1]	Y or N[1]	M
SET CAPACITY		Y	N	O
SPACE(6)		N	Y	M
SPACE(16)		N	Y	O
TEST UNIT READY		N	N	M
VERIFY (6)		N	Y	O
WRITE (6)		Y	N	M
WRITE BUFFER		Y	N	O
WRITE FILEMARKS (6)		Y	N	M

Key: M = Command implementation is mandatory if implementing this model.
O = Command implementation is optional.

The implicit address model is not designed to support command queueing.

Below is a description of the the implicit address model:

On read type commands the current position is used to control where to start execution of the command. The logical block address is implicit. The LOCATE command is an exception to this rule. Position is maintained on a command by command basis.

Similarly on write type commands, current position is the determining factor on where to write append. It is implicit and no other write append point is possible.

3.0.3 Simultaneous support for Implicit and Explicit Address Models

The explicit address model is mandatory. For those drives that additionally support the implicit address model, a mechanism is needed to manage the transition. The mechanism to make a dynamic transition between operation in each model is defined below:

A transition from the explicit address model to the implicit address model can be made when the drive is in the neutral state. The transition is made by issuing a command from the implicit address command set. A drive actively using the explicit address model that is in the write capable or read capable state that receives a command from the implicit address command set rejects the command as described in x.x.x.x explicit address model behavior.

A transition from the implicit address model to the explicit address model is made by issuing a non-queued command from the explicit address model command set. This transitions the drive to the neutral state of the explicit address command model.

On page 27 change the ERASE command to ERASE(6).

Table 1 on page 17 needs to be updated to account for the new commands.

Table 3 on page 20 needs to be updated to account for the new commands.

Table 9 on page 24 needs to be updated to account for the new commands.

Table 10 gets changed to the form below(subclauses and op codes purposely omitted):

Table 3 - Commands for sequential-access devices

Command Name	Flush Write Data	Operation code	Subclause
ERASE (6)	Yes		
ERASE (16)	Yes		
FORMAT MEDIUM	No		
INQUIRY	No		
LOAD UNLOAD	Yes		
LOCATE (10)	Yes		
LOCATE (16)	Yes		
LOG SELECT	No		
LOG SENSE	No		
MODE SELECT (6)	Yes		
MODE SELECT (10)	Yes		
MODE SENSE (6)	No		
MODE SENSE (10)	No		
MOVE MEDIUM	Yes		
MOVE MEDIUM ATTACHED	Yes		
PERSISTENT RESERVE IN	No		
PERSISTENT RESERVE OUT	No		
PREVENT ALLOW MEDIA REMOVAL	No		
READ (6)	Yes		
READ (16)	Yes		
READ BLOCK LIMITS	No		
READ BUFFER	Yes		
READ ELEMENT STATUS	No		
READ ELEMENT STATUS ATTACHED	No		
READ POSITION	No		
READ REVERSE (6)	Yes		
READ REVERSE (16)	Yes		
RECEIVE DIAGNOSTIC RESULTS	No		
RECOVER BUFFERED DATA	May		
RELEASE(6)	No		
RELEASE(10)	No		
REPORT DENSITY SUPPORT	No		
REPORT LUNS	No		
REQUEST SENSE	No		
RESERVE(6)	No		
RESERVE(10)	No		
REWIND	Yes		
SEND DIAGNOSTIC	Yes		
SET CAPACITY	May		
SPACE(6)	May		
SPACE(16)	May		
TEST UNIT READY	No		
VERIFY (6)	Yes		
VERIFY (16)	Yes		
WRITE (6)	No		
WRITE (16)	No		
WRITE BUFFER	Yes		
WRITE FILEMARKS (6)	May		
WRITE FILEMARKS (16)	May		

3.1 ERASE (12)

The ERASE(12) command ... *add in text here just as it appears in 5.3.1 ERASE Command*

Table 4 - ERASE (12) command

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE							
1	RESERVED					WR_APP	IMMED	LONG
2	PARTITION							
3	LOGICAL BLOCK ADDRESS							
10								
11	CONTROL							

Add the following text for the WR_APP *field:*

The WR_APP field is used to indicate that this command is the first in a string of sequential queued write commands. If the command is not queued, the value of this bit shall be ignored. If the command is queued and this bit is set, an append may occur at any valid position on tape prior to or at End Of Data in any specified partition. If the command is queued and this bit is not set, then data may only be written if the LOGICAL BLOCK ADDRESS and PARTITION specified match the current position on tape. If the position doesn't match, the drive shall return CHECK CONDITION with sense key ILLEGAL REQUEST and an additional sense code qualifier of SEQUENTIAL POSITIONING ERROR.

Add the following text for the logical block number and partition fields:

The LOGICAL BLOCK ADDRESS and PARTITION fields indicate the position at which the ERASE command should start.

The LOCATE (16) command ... *add in text here just as it appears in 5.3.1 LOCATE (16) Command*

Table 5 - LOCATE (92) command

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE							
1	RESERVED			BOUND	CODE			IMMED
2	LOGICAL BLOCK ADDRESS							
9								
10	RESERVED							
13								
14	PARTITION							
15	CONTROL							

Add the following text for the BOUND *field:*

The BOUND field is valid only when used with the CODE field set to value of BLOCK (000b). When used with CODE set to any other value, the drive shall respond with CHECK CONDITION status, with sense data sense key of ILLEGAL REQUEST and additional sense data of INVALID FIELD IN CDB.

The BOUND field specifies that the locate block will stay within the current file and set boundaries. If a filemark is encountered when locating to a block, the drive shall return CHECK CONDITION status, and the FILEMARK and VALID bits shall be set to one in the ssense data. The sense key shalll be set to FILEMARK DETECTED. The information field shall be set to the requed count minus the actual number of blocks spaced over. The logical position shall be on the end-of-partition side of the filemark if movement was in the backward direction and on the beginning-of-partition side of the filemark if movement was in the reverse direction (in other words, the filemark is not traversed).

If a setmark is encountered while locating to a bock with the BOUND field set and the RSMK is set in the device configuration page (see x.x.x.x) the command shall be terminated, CHECK CONDITION status shall be returned and the FILEMARK and VALID bits shall be set to one in the sense data. The sense key shall be set to NO SENSE, and the additional sense code and an aditional sense code qualifier shall be set to SETMARK DETECTED. The information field shall be set to the requested count minus the acual number of blocks or filemarks spaced over. The logical position shall be on the end-of-partition side of the setmark if movement was in the backward direction and on the beginning-of-partition side of the setmark if movement was in the reverse direction (in other words, the setmark is not traversed).

Note: An important purpose of the BOUND field is to allow device drivers to completely map the implicit command set to the explicit command set. The space block function specified in the SPACE comand would not be possible without this field.

Add the following text for the code field:

The code field is defined in the table below:

Table 6 -

Code	Description	Support
000b	Block	Mandatory
001b	Filemark	Mandatory
010b	Sequential filemarks	Optional
011b	End-of-data	Optional
100b	Setmark	Optional
101b	Sequential setmarks	Optional
110b	BT	Optional
111b	CP	Optional

The text describing the the details of each code come out of the current LOCATE (10) and SPACE (6) command text.

On page 32 change the READ command to READ(6).

3.2 READ (16)

The READ(16) command ... *add in text here just as it appears in 5.3.6 READ Command*

Table 7 - READ (16) command

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE							
1	RESERVED						SILI	FIXED
2	PARTITION							
3	LOGICAL BLOCK ADDRESS							
10								
11	TRANSFER LENGTH							
13								
14	RESERVED							
15	CONTROL							

Add the following text for the logical block number and partition fields:

The LOGICAL BLOCK ADDRESS and PARTITION indicates the position at which the READ command should start. If the current position does not match what is in the LOGICAL BLOCK ADDRESS and PARTITION field the drive should locate to that position and return the block as requested.

Also, text should be changed to reflect that filemarks and setmarks are not traversed when encountered.

3.3 READ REVERSE (16)

The READ REVERSE(16) command ... *add in text here just as it appears in 5.3.9 READ*

REVERSE Command

Table 8 - READ REVERSE (16) command

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE							
1	RESERVED					BYTORD	SILI	FIXED
2	PARTITION							
3	LOGICAL BLOCK ADDRESS							
10								
11	TRANSFER LENGTH							
13								
14	RESERVED							
15	CONTROL							

Add the following text for the logical block number and partition fields:

The LOGICAL BLOCK ADDRESS and PARTITION indicates the position at which the READ REVERSE command should start. If the current position does not match what is in the LOGICAL BLOCK ADDRESS and PARTITION field the drive should locate to that position and return the block as requested.

3.4 RECOVER BUFFERED DATA (16)

The RECOVER BUFFERED DATA(16) command ... *add in text here just as it appears in 5.3.10*

RECOVER BUFFERED DATA Command

Table 9 - RECOVER BUFFERED DATA(16) command

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE							
1	RESERVED						SILI	FIXED
2	PARTITION							
3	LOGICAL BLOCK ADDRESS							
10								
11	TRANSFER LENGTH							
13								
14	RESERVED							
15	CONTROL							

Add the following text for the logical block number and partition fields:

The LOGICAL BLOCK ADDRESS and PARTITION indicates the position at which the RECOVER BUFFERED DATA command should start. If the current position does not match what is in the LOGICAL BLOCK ADDRESS and PARTITION field the drive should locate to that position and return the block as requested.

3.5 SPACE (16)

If it's not too late, I would propose obseleting the SPACE(16) command and requiring use of the LOCATE (16) command.

3.6 VERIFY (16)

The VERIFY(16) command ... *add in text here just as it appears in 5.3.16 VERIFY Command*

Table 10 - VERIFY (16) command

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE							
1	RESERVED					IMMED	BYTCMP	FIXED
2	PARTITION							
3	LOGICAL BLOCK ADDRESS							
10								
11	VERIFICATION LENGTH							
13								
14	RESERVED							
15	CONTROL							

Add the following text for the logical block number and partition fields:

The LOGICAL BLOCK ADDRESS and PARTITION indicates the position at which the VERIFY command should start. If the current position does not match what is in the LOGICAL BLOCK ADDRESS and PARTITION field the drive should locate to that position and return the block as requested.

3.7 WRITE (16)

The WRITE (16) command ... *add in text here just as it appears in 5.3.17 WRITE Command*

Table 11 - WRITE (16) command

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE							
1	RESERVED						WR_APP	FIXED
2	PARTITION							
3	LOGICAL BLOCK ADDRESS							
10								
11	TRANSFER LENGTH							
13								
14	RESERVED							
15	CONTROL							

Add the following text for the WR_APP *field:*

The WR_APP field is used to indicate that this command is the first in a string of sequential queued write commands. If the command is not queued, the value of this bit shall be ignored. If the command is queued and this bit is set, an append may occur at any valid position on tape prior to or at End Of Data in any specified partition. If the command is queued and this bit is not set, then data may only be written if the LOGICAL BLOCK ADDRESS and PARTITION specified match the current position on tape. If the position doesn't match, the drive shall return CHECK CONDITION with sense key ILLEGAL REQUEST and an additional sense code qualifier of SEQUENTIAL POSITIONING ERROR.

Add the following text for the logical block number and partition fields:

The LOGICAL BLOCK ADDRESS and PARTITION fields indicate the position at which the WRITE command should start.

3.8 WRITE FILEMARKS (16)

The WRITE FILEMARKS (16) command ... *add in text here just as it appears in 5.3.17 WRITE*

FILEMARKS Command

Table 12 - WRITE FILEMARKS (16) command

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE							
1	RESERVED					WR_APP	WSMK	FIXED
2	PARTITION							
3	LOGICAL BLOCK ADDRESS							
10								
11	TRANSFER LENGTH							
13								
14	RESERVED							
15	CONTROL							

Add the following text for the WR_APP *field:*

The WR_APP field is used to indicate that this command is the first in a string of sequential queued write commands. If the command is not queued, the value of this bit shall be ignored. If the command is queued and this bit is set, an append may occur at any valid position on tape prior to or at End Of Data in any specified partition. If the command is queued and this bit is not set, then data may only be written if the LOGICAL BLOCK ADDRESS and PARTITION specified match the current position on tape. If the position doesn't match, the drive shall return CHECK CONDITION with sense key ILLEGAL REQUEST and an additional sense code qualifier of SEQUENTIAL POSITIONING ERROR.

Add the following text for the logical block number and partition fields:

The LOGICAL BLOCK ADDRESS and PARTITION fields indicate the position at which the WRITE FILEMARKS command should start.