Satsuma SCSI
DASD Interface Specification
Release 4.00

Basic Features

PSSD Development.
Fujisawa.
FU-R89.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

EC HISTORY

- REV. 1.0 P/N. 66G9703 12/22/93
- REV. 1.01 P/N. 66G9703 01/25/94
- REV. 2.00 P/N. 66G9703 03/07/94
- REV. 3.00 P/N. 66G9703 03/28/94
- REV. 3.01 P/N. 66G9703 04/05/94
- REV. 3.02 P/N. 66G9703 04/13/94
- REV. 3.03 P/N. 66G9703 04/15/94
- REV. 4.00 P/N. 66G9703 06/08/94

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

Contents

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

Figures

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

Tables

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.0 SCSI COMMAND SET

Summaries of the SCSI commands supported by the file are listed below. where O=optional, M=mandatory, E=extended, R=reserved and V=vendor unique. The column “SCSI-1” refers to ANSI version 1 standard. The column “CCS” refers to the ANSI sub-committee Common Command Subset for DASD devices. The column “SCSI-2” refers to ANSI version 1 standard.

SCSI–1	CCS	SCSI–2	CODE	COMMAND
M	M	M	04h	FORMAT UNIT
E	M	M	12h	INQUIRY
O	O	O	15h	MODE SELECT
O	O	O	1Ah	MODE SENSE
O	O	O	34h	PRE–FETCH
M	M	M	08h	READ
E	M	M	25h	READ CAPACITY
E	M	M	28h	READ EXTENDED
R	O	O	3Ch	READ BUFFER
R	R	O	37h	READ DEFECT DATA
R	R	O	3Eh	READ LONG
O	O	O	07h	REASSIGN BLOCKS
O	M	M	17h	RELEASE
M	M	M	03h	REQUEST SENSE
O	M	M	16h	RESERVE
O	O	O	01h	REZERO UNIT
O	O	O	0Bh	SEEK
O	O	O	2Bh	SEEK EXTENDED
O	M	M	1Dh	SEND DIAGNOSTICS
O	O	O	1Bh	START/STOP UNIT
R	R	O	35h	SYNCHRONIZE CACHE
O	M	M	00h	TEST UNIT READY
O	O	O	2Fh	VERIFY
M	M	M	0Ah	WRITE
E	M	M	2Ah	WRITE EXTENDED
O	O	O	2Eh	WRITE AND VERIFY
R	O	O	3Bh	WRITE BUFFER
R	R	O	3Fh	WRITE LONG

Figure 1. SCSI Commands Supported. (In Alphabetical order)

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

SCSI–1	CCS	SCSI–2	CODE	COMMAND
O	M	M	00h	TEST UNIT READY
O	O	O	01h	REZERO UNIT
M	M	M	03h	REQUEST SENSE
M	M	M	04h	FORMAT UNIT
O	O	O	07h	REASSIGN BLOCKS
M	M	M	08h	READ
M	M	M	0Ah	WRITE
O	O	O	0Bh	SEEK
E	M	M	12h	INQUIRY
O	O	O	15h	MODE SELECT
O	M	M	16h	RESERVE
O	M	M	17h	RELEASE
O	O	O	1Ah	MODE SENSE
O	O	O	1Bh	START/STOP UNIT
O	M	M	1Dh	SEND DIAGNOSTICS
E	M	M	25h	READ CAPACITY
E	M	M	28h	READ EXTENDED
E	M	M	2Ah	WRITE EXTENDED
O	O	O	2Bh	SEEK EXTENDED
O	O	O	2Eh	WRITE AND VERIFY
O	O	O	2Fh	VERIFY
O	O	O	34h	PRE–FETCH
R	R	O	35h	SYNCHRONIZE CACHE
R	R	O	37h	READ DEFECT DATA
R	O	O	3Bh	WRITE BUFFER
R	O	O	3Ch	READ BUFFER
R	R	O	3Eh	READ LONG
R	R	O	3Fh	WRITE LONG

Figure 2. SCSI Commands Supported. (By Command Code)

1.1 Flag and Link Bits

Many of the structures in this section have fields names FLAG and LINK. The meaning of these fields is defined below.

FLAG The Flag bit specifies which message the target shall return to the initiator if the link bit is one and the command completes without any error. If Link is zero, Flag must also be zero. If Link is one and the command terminates successfully, the file will send either the LINKED COMMAND COMPLETE message (FLAG=0) or the LINKED COMMAND COMPLETE WITH FLAG message (FLAG=1). Typically this bit is used to cause an interrupt in the initiator between linked commands.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

LINK This bit is set to one to indicate that the initiator desires an automatic link to the next command upon successful completion of the current command. Upon successful completion of the command, the file will return INTERMEDIATE GOOD status and then send one of the two messages defined under Flag above.

Upon unsuccessful completion of the command, the file will return CHECK CONDITION status or RESERVATION CONFLICT status and then send the COMMAND COMPLETE message. No further commands in the chain are executed.

1.2 Abbreviations

These abbreviations are used throughout the following sections:

LUN. Logical Unit Number. An encoded three bit identifier for the logical unit.

VU. Vendor Unique bits.

LBA. Logical Block Address.

RSVD. Reserved.

MSB. Most Significant bit.

LSB. Least Significant bit.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.3 FORMAT UNIT (04)

	BIT 7	6	5	4	3	2	1	0
BYTE 0	Command code (04h)							
BYTE 1	LUN			Fmt Data	Cmp Lst	List Format		
BYTE 2	VU = 0							
BYTE 3	(MSB)							
BYTE 4	Interleave Factor (LSB)							
BYTE 5	VU = 0		RSVD = 0				Flag	Link

Figure 3. FORMAT UNIT (04)

The FORMAT command performs a physical formatting of the file media. This includes handling of defective sectors, and the overwriting of all data areas with a constant data pattern. (Reserved areas of the media are not affected by the FORMAT command.)

- **FmtDt** set to one specifies that a Data Out phase follows the Command phase. The Data Out phase consists of a defect list header followed by **zero defect descriptors**. FmtDt set to zero specifies that no Data Out phase follows.
- **CmpIt** set to one specifies that the GList (Grown Defect List) existing prior to the format **not** be used and is discarded. The Drive is formatted with PList and DList (if specified). DList becomes the new GList.

  Note: The file manages two internal defect lists and one external. The primary defect list (“P”List) is created at time of manufacture and cannot be altered. The grown defect list (“G”List) is built after time of manufacture by the Initiators use of the REASSIGN BLOCK command and the Automatic Reallocate function, The data defect list (“D”List) is an external list. It is supplied by the initiator in the DATA OUT phase of the FORMAT UNIT command.
- **List Format** specifies the format of the defect descriptor transferred to the Target when FmtData bit is set to one.
- **Interleave Factor** may be zero or one, either of which specifies an interleave of 1:1. Other Interleave Factors are not supported because of the extensive buffering implemented in the file.

1.3.1 Defect List

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

	BIT 7	6	5	4	3	2	1	0
BYTE 0	Reserved = 0							
BYTE 1	FOV	DPRY =0	DCRT	STPF	IP =0	DSP =0	Immd	0
BYTE 2	Defect list length MSB							
BYTE 3	Defect list length LSB							

Figure 4. Format of Defect List Header. Format of the defect list header sent during the data out phase when FmtDt set to one.

	BIT 7	6	5	4	3	2	1	0
BYTE 0 – 3	Defect Descriptor 0							
BYTE 4n – 4n + 3	Defect Descriptor n							

Figure 5. Format of Defect List. Format of the defect list sent during the data out phase when FmtDt set to one.

The Target has a limited implementation of the Format Option bits located in Bits 2 through 7 of Byte 1 of the Defect List Header (See Figure 4). If the Initiator attempts to select any function not implemented by the Target, the Target terminate the command with *Check Condition Status*. The sense key is set to *Illegal Request* and the additional sense code is set to *Invalid Field in Parameter List*.

- **FOV** (Format Options Valid) bit of zero causes the Target to verify that the setting for the DPRY (Disable Primary), DCRT (Disable Certification), STPF (stop Format), IP (Initialize Pattern), and DSP (Disable Saving Parameters) bits are zero. If any of these bits are not zero, the Target terminates the command with *Check Condition Status*. The sense key is set to *Illegal Request* and the additional sense code is set to *Invalid Field in Parameter List*.

  Note: When FOV bit is one there is only one combination of the DPRY, DCRT, STPF, IP and DSP bits allowed. Any other combinations return a *Check Condition Status* With a sense key of *Illegal Request* and an additional sense code of *Invalid Field In Parameter List*. The supported combination are:

  DPRY=0 DCRT=1 STPF=1 IP=0 DSP=0

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

- **DPRY**(Disable Primary) bit set to zero indicates that the Target does not use portions of the medium identified as defective in the primary defect PList for Initiator addressable logical blocks. If the Target cannot locate the PList or it cannot determine whether a PList exists, the target terminates the Format Unit command as described for STPF=1.
- **DCRT** (Disable certification) bit must be set to one. The Target does not generate a CList (certification list) nor perform a certification process while executing the Format Unit Command.
- **STPF** (stop Format) bit must be set to one. If one or both of the following conditions occurs, the Target terminates the Format Unit command with *Check Condition Status*. The sense key is set to *Medium Error* and the additional sense code is set to either *Defect List Not Found* if the first condition occurred, or *Defect List Error* if the second condition occurred.
  - The Target cannot locate a required defect list nor determine that the list exists.
  - The Target encounters an unrecoverable error while accessing a required defect list.
- **IP** (Initialization Pattern) bit must be set to zero. The Target initializes all data with zeros.
- **DSP** (Disable Saving Parameters) bit must be set to zero. The Target saves all the Mode Select savable parameters during the format operation.
- **IMMD** (immediate) bit set to zero requests that status be returned at the end of the format operation. An immediate bit set to one requests that status be returned immediately. *Good Status* is returned following the CDB validation and transfer of data in the Data Out phase. If the immediate format operation terminates in error, Deferred Error Sense data is generated. With the immediate bit set to one, the Link bit must be set to zero.

The Defect List Length field specifies the total length in bytes of the defect descriptors that follow. The Target has an implementation limitation for number of defect descriptors. The number of defect descriptor shall be less than **128**. The defect list length must be equal to four times the number of defect descriptors to follow, otherwise the command is terminated with *Check Condition Status* The sense key is set to *Illegal Request* and the additional sense code is set to *Invalid Field In Parameter List*. The defect descriptors must specify the defect based on the current Format Device parameters reported by the Mode Sense command.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.4 INQUIRY (12)

	BIT 7	6	5	4	3	2	1	0
BYTE 0	Command Code = 12h							
BYTE 1	LUN			RSVD = 0				EVPD
BYTE 2	PAGE CODE							
BYTE 3	RSVD = 0							
BYTE 4	ALLOCATION LENGTH							
BYTE 5	VU = 0		RSVD = 0				FLAG	LINK

Figure 6. INQUIRY (12)

The INQUIRY command requests the parameters of the target to be sent to the initiator.

An EVPD bit of one specifies that the file shall return the vital product data page identified by the Page Code field in the CDB.[1] **Page code** specifies which page of vital product data information the file shall return.

EVPD	PAGE CODE	Description
0	0	The file returns the standard INQUIRY data.
0	Non Zero	The file returns CHECK CONDITION status with the sense key of ILLEGAL REQUEST and the additional sense code of INVALID FIELD IN CDB.
1	Supported	The file returns the vital product data of page code requested.
1	Unsupported	The file returns CHECK CONDITION status with the sense key of ILLEGAL REQUEST and the additional sense code of INVALID FIELD IN CDB.

[1] The available VPD pages are defined in the addendum provided for each different file model in the section entitled **Inquiry Data Format**.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

Allocation Length specifies the number of bytes that the initiator has allocated for INQUIRY data to be returned. An allocation length of zero implies that no data is to be returned. The file will terminate the DATA IN phase when all available INQUIRY data has been transferred or when allocation length bytes have been transferred, whichever is less.

If an INQUIRY command is received from an initiator with a pending unit attention condition (before the target reports CHECK CONDITION status), the file processes the INQUIRY command. The unit attention condition is not cleared by this action.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.5 MODE SENSE (1A)

	BIT 7	6	5	4	3	2	1	0
BYTE 0	Command code = 1Ah							
BYTE 1	LUN			RSVD = 0				
BYTE 2	PCF		Page Code					
BYTE 3	RSVD = 0							
BYTE 4	Allocation Length							
BYTE 5	VU = 0		RSVD = 0				FLAG	LINK

Figure 7. MODE SENSE (1A)

The MODE SENSE command provides a means for the file to report various device parameters to the initiator. It is the complement to the MODE SELECT command.

Allocation Length indicates the maximum number of bytes the initiator has set aside for the DATA IN phase. A value of zero is not considered an error. If the allocation length is smaller than the amount available, then that portion of the data up to the allocation length will be sent. It is noted that this may result in only a portion of a multi-byte field being sent.

Page Control Field: PCF (Page Control Field) defines the type of Page Parameter values to be returned.

PCF **Meaning**

0 0 **Report current values.** The file returns the current values under which the logical unit is presently configured for the page code specified. The current values returned are:

1. The parameters set in the last successful MODE SELECT command.
2. The saved values if a MODE SELECT command has not been executed since the last power-on, hard RESET condition, or BUS DEVICE RESET message .

Note: The file will not process the Mode Select command until the completion of spin-up. Therefore, the initiator cannot modify the current values prior to the saved values being read in.

0 1 **Report changeable value.** The file returns the changeable values for the page code specified. The page requested shall be returned containing information that indicate which fields are changeable. All bits of parameters that are changeable shall be set to one. Parameters that

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

are *defined by the file* shall be set to zero. If any part of a field is changeable all bits in that field shall be set to one.

Note: For a value field such as the buffer ratios of page 2, the bit field will not indicate the range of supported values but rather that the field is supported.

1 0 **Report default value.** The file returns the default values for the page code specified. The parameters not supported by the file are set to zero.

1 1 **Report saved value.** The file returns the saved value for the page code specified.

Saved values are one of following :

- the values saved as a result of MODE SELECT command
- identical to the default values
- zero when the parameters are not supported

The Page Length byte value of each page returned by the file indicates up to which fields are supported on that page.

Page Code: This field specifies which page or pages to return. Page code usage is defined in Figure 8.

Page Code	Description
01h – 38h	Return specific page.
3Fh	Return all available pages.

Figure 8. Page Code Usage

1.5.1 Mode Parameter List

The mode parameter list contain a header, followed by zero or more block descriptors, followed by zero or more variable-length pages.

1.5.1.1 HEADER

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

	7	6	5	4	3	2	1	0
BYTE 0	Mode Data Length							
BYTE 1	Medium Type = 0							
BYTE 2	WP	RSVD = 0						
BYTE 3	Block Descriptor Length (= 0 or 8)							

Figure 9. MODE Parameter List (Header)

- **Mode Data Length**. When using the MODE SENSE command, the mode data length field specifies the length in bytes of the following data that is available to be transferred. The mode data length does not include the length byte itself. When using the MODE SELECT command, this field is reserved.
- **Medium Type** field is always set to zero in the file. (Default medium type)
- **WP**. When used with the MODE SELECT command, the write protect (WP) bit is reserved.

  When used with the MODE SENSE command, a write protect (WP) bit of zero indicates that the medium is write enabled.
- **Block Descriptor Length**. This field specifies the length in bytes of the block descriptors.

  When used with the MODE SELECT command, zero or eight are supported by the file.

  When used with the MODE SENSE command, the file returns eight to indicate that only a single block descriptor is available.

1.5.1.2 Block Descriptor

BYTE 0	Density code = 0
BYTE 1 BYTE 2 BYTE 3	Number of Blocks (MSB) (LSB)
BYTE 4	RSVD = 0
BYTE 5 BYTE 6 BYTE 7	Block Length = 200h

Figure 10. MODE Parameter Block Descriptor

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

The Block descriptor provides formatting information about the Number of Blocks (user addressable) to format at the specified Block Length.

- **Number of Blocks**

  When used with the MODE SELECT command, the **Number of Blocks** field must be;

  – Zero to indicate all available blocks
  – The exact number of blocks in the data area of the file, which can be obtained with the MODE SENSE command.

  Any other value is invalid, and causes the command to fail with CHECK CONDITION status.

  When used with the MODE SENSE command, the field contain exact number of blocks.

- **Block Length**

  When used with the MODE SELECT command, the **Block length** field must contain 512, or the file will terminate the command with CHECK CONDITION status.

  When used with the MODE SENSE command, the field will return always contain 512 .

1.5.1.3 Page Descriptor

BYTE 0	PS	RSVD=0	Page Code
BYTE 1			Page Length
BYTE 2–n			Mode Parameters

Figure 11. MODE Parameter Page Format

Each mode page contains a page code, a page length, and a set of mode parameters.

When using the MODE SENSE command, a parameter savable (PS) bit of one indicates that the mode page can be saved by the file in the reserved area of the file.

A PS bit of zero indicates that the supported parameters cannot be saved. When using the MODE SELECT command, the PS bit is reserved (zero).

The pages supported are described in the product specific specification.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.6 MODE SELECT (15)

	BIT 7	6	5	4	3	2	1	0
BYTE 0	Command Code = 15h							
BYTE 1	LUN			PF=1	RSVD = 0			SP
BYTE 2	RSVD = 0							
BYTE 3	RSVD = 0							
BYTE 4	Parameter List Length							
BYTE 5	VU = 0		RSVD = 0				FLAG	LINK

Figure 12. MODE SELECT (15)

The MODE SELECT command provides a means for the initiator to specify LUN or device parameters to the Target. It also allows an Initiator to specify options the Target uses in error recovery and Caching.

There is a single set of Mode Page parameters shared by all initiators.

PF A PF(Page Format) bit value of 1 indicates the data sent by the Initiator after the Mode Select Header and the Block Descriptor, if any, complies to the Page Format. The Target ignores this field since it only accepts mode parameters in the Page Format.

SP Save Pages. This indicates;

0 The drive shall not save the pages sent during the Data Out phase but will use them for all following commands until the power is removed, a reset is received or a new mode select command is received.

1 The drive will save the data in the reserved area of the disk. It will be used for all following commands until another mode select command is issued, this information is maintained over a power cycle or reset of the file.

Parameter List Length

This specifies the number of bytes to be sent from the initiator. A parameter list length of zero suppresses data transfer and is not considered as an error.

The MODE SELECT parameter list contains a four-byte header, followed by zero or one block descriptor followed by zero or more pages. The pages which are valid with this command are defined in the addendum under the heading **Mode Select Data**. as they vary with the file model.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.6.1.1 Application Note

The initiator should issue a MODE SENSE command requesting all Changeable values (see PCF field in byte two of the CDB in 1.5, “MODE SENSE (1A)” on page 17) prior to issuing a MODE SELECT command. This is necessary to find out which pages are implemented by the file and the length of those pages. The file will return, in the Pages of the MODE SENSE command, the number of bytes supported for each Page. The Page Length set by the initiator in the MODE SELECT command must be the exact value as that returned by the file in MODE SENSE Page Length. If this is not true, the file will return CHECK CONDITION status with sense key of ILLEGAL REQUEST. See 1.5, “MODE SENSE (1A)” on page 17.

Note: If an initiator sends a MODE SELECT command that changes any parameters that apply to other initiators, the file shall generate an unit attention condition for all initiators except the one that issued the MODE SELECT command. The file shall set the additional sense code to PARAMETERS CHANGED (2Ah).

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.7 PRE-FETCH (34)

	BIT 7	6	5	4	3	2	1	0
BYTE 0	Command Code = 34h							
BYTE 1	LUN			Reserved = 0			Immed	RelAdr = 0
BYTE 2 BYTE 3 BYTE 4 BYTE 5	(MSB) Logical Block Address							(LSB)
BYTE 6	Reserved = 0							
BYTE 7 BYTE 8	(MSB) Transfer Length							(LSB)
BYTE 9	VU = 0			Reserved = 0			FLAG	LINK

Figure 13. Pre-Fetch (34)

The PRE-FETCH command requests the file to transfer data to the cache. No data is transferred to the initiator.

Immed Immediate.

If the Immediate (Immed) bit of the CDB is zero:

- If an error occurs while reading,error recovery procedures are attempted. The Drive returns GOOD status or CHECK CONDITION status based on the setting of the MODE SELECT Page 1 parameters.
- If there is enough room in the segment for all of the Requested Data or if the Transfer Length is zero and no error occurs while reading,the Drive returns CONDITION MET status when the command completes.
- If there is not enough room in the segment,the Transfer Length is not zero, and no error occurred while reading,the Drive returns GOOD status when the command completes.

If the Immediate (Immed) bit of the CDB is one:

- If there is enough room in the segment for all of the Requested Data or if the Transfer Length is zero,the Drive returns CONDITION MET status as soon as the CDB is verified.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

- If there is not enough room in the segment and the Transfer Length is not zero,the Drive returns GOOD status as soon as the CDB is verified.
- The reading of data is handled the same as Read-Ahead operation.This implies the prefetch may be terminated upon receipt of another command.
- if an error is encountered:
    1. The file terminates the Pre-Fetch operation and does not attempt to recover the data.
    2. The error is not reported to the Initiator for the current command. (the error will be reported during the next command if the next command is a Read command that requests the block which encountered the error.)
    3. The blocks which were successfully read prior to the block in error are retained in the cache.

RelAdr Relative Block Address. **Must be set to zero**, indicating that the logical block address field specifies the first logical block of the range of logical blocks to be operated on by this command. Relative address is not supported.

Transfer length The transfer length field specifies the number of contiguous blocks of data that are to be transferred into the cache. A transfer length of zero indicates that blocks are to be transferred into the cache until the segment is filled or there are no more blocks on the media.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.8 READ (08)

	BIT 7	6	5	4	3	2	1	0
BYTE 0	Command Code = 08h							
BYTE 1	LUN			(MSB)	LBA			
BYTE 2	LOGICAL BLOCK ADDRESS							
BYTE 3	LOGICAL BLOCK ADDRESS (LSB)							
BYTE 4	TRANSFER LENGTH							
BYTE 5	VU = 0		RSVD = 0				FLAG	LINK

Figure 14. READ (08)

The READ command requests the file to transfer the specified number of blocks of data to the initiator starting at the specified logical block address.

Logical block address This field specifies the logical unit at which the read operation shall begin.

Transfer length This field specifies the number of blocks to be transferred. A value of zero implies 256 blocks are to be transferred.[2]

Note: Errors are handled by ERP(error recovery procedure). ERPs are controlled by the error recovery parameters specified by MODE SELECT command.

[2] Block is 512 bytes in length.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.9 READ CAPACITY (25)

	BIT 7	6	5	4	3	2	1	0
BYTE 0	Command Code = 25h							
BYTE 1	LUN			RSVD = 0				RelAdr
BYTE 2	(MSB)	Logical Block Address						
BYTE 3								
BYTE 4								
BYTE 5								(LSB)
BYTE 6	RSVD = 0							
BYTE 7	RSVD = 0							
BYTE 8	RSVD = 0							PMI
BYTE 9	VU = 0		RSVD = 0				FLAG	LINK

Figure 15. READ CAPACITY (25)

The READ CAPACITY command returns information regarding the capacity of the file.

- **RelAdr**. A Relative Address is not supported. Must be set to zero.
- **Logical Block Address** is used in conjunction with the PMI bit.
- **PMI**, Partial Medium Indicator indicates;

PMI	Description
0	The drive returns the last logical block address of the file.
1	The drive returns the last logical block address on the same track where the LBA in the command is located. This option provides the information the initiator needs to determine the amount of space available on the same track which is accessible without a head switch or seek.

1.9.1.1 Returned Data Format

The data returned to the initiator in response to the READ CAPACITY command is described here. The data is returned in the DATA IN phase.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

	BIT 7	6	5	4	3	2	1	0
BYTE 0	(MSB)		Logical Block Address					
BYTE 1								
BYTE 2								
BYTE 3								(LSB)
BYTE 4	(MSB)		Block Length					
BYTE 5								
BYTE 6						= 512		
BYTE 7								(LSB)

Figure 16. Format of READ CAPACITY command reply

- **Block Length** specifies the length in bytes of the block. It is set to 512.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.10 READ DEFECT DATA (37)

	BIT 7	6	5	4	3	2	1	0
BYTE 0	COMMAND CODE = 37h							
BYTE 1	LUN			Rsvd = 0				0
BYTE 2	Rsvd = 0			Plist	Glist	Defect List Format		
BYTE 3 BYTE 4 BYTE 5 BYTE 6	Rsvd = 0							
BYTE 7 BYTE 8	Allocation length (MSB) (LSB)							
BYTE 9	VU = 0		RSVD = 0				FLAG	LINK

Figure 17. Read Defect Data (37)

The READ DEFECT DATA command requests that the Target transfers the medium defect data to the initiator.

If the target is unable to access any medium defect data it will return a Check Condition status with the appropriate sense key. The sense key will be set to either Medium Error(03h) if a medium error occurred or No Sense(00h) if the list does not exist and the additional sense code will be set to Defect List Error(19h).

Plist The Primary Defect List (Plist) bit set to one indicates that the target returns the primary list of defects. A Plist bit of zero indicates that the target shall not return the Primary Defect list of defects.

Glist The Grown Defect List (Glist) bit set to one indicates that the target returns the grown defect list. A Glist bit of zero indicates that the target shall not return the Grown Defect list of defects.

Note: With both bits set to one Plist and Glist the target will return both the Primary and Grown defect lists. With both bits set to zero, the target will return only a four-byte Defect List Header.

Defect List format The Defect List Format Field is used by the initiator to indicate the preferred format for the defect list.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

The Defect List Format of '100 (Bytes from Index Format) ' and '101 (Physical Sector Format)' are supported. If the requested format is not supported by the file, it will return the defect list in its default format '101' then terminates the command with Check Condition status. The sense key will be set to Recovered Error(01h) and the additional sense code will be set to Defect List Not Found(1Ch).

The file sends defect list (Defect Descriptors) in a four byte ABA (absolute block address) format which follows a four byte Defect List Header.

The target will transfer all of the Read Defect Data up to the number of bytes allocated by the initiator.

Note: The file will terminate the Data In phase when the Allocation Length has been transferred or when all available Defect Data has been transferred to the initiator, whichever is less.

The Read Defect Data contains a four byte header, followed by zero or more defect descriptors.

1.10.1 Defect List Header

	BIT 7	6	5	4	3	2	1	0
	Defect List Header							
BYTE 0	Rsvd = 0							
BYTE 1	Rsvd = 0				Plist	Glist	Defect List Format	
BYTE 2	Defect List Length (MSB)							
BYTE 3	(LSB)							

Figure 18. Defect List Header

1.10.2 Bytes from Index Format (100b)

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

	Defect Descriptors	
BYTE 0 BYTE 1 BYTE 2	Cylinder Number of Defect	(MSB) (LSB)
BYTE 3	Head Number of Defect	
BYTE 4 BYTE 5 BYTE 6 BYTE 7	Defect Bytes from Index	(MSB) (LSB)

Figure 19. Defect Descriptors of Bytes from Index Format

Defect Bytes from Index is gotten using the following equation:

```
Bytes from Index = (Physical Sector Number) * N

      Where:  N = Bytes per sector (512 Bytes)
```

1.10.3 Physical Sector Format (101b)

	Defect Descriptors	
BYTE 0 BYTE 1 BYTE 2	Cylinder Number of Defect	(MSB) (LSB)
BYTE 3	Head Number of Defect	
BYTE 4 BYTE 5 BYTE 6 BYTE 7	Defective Sector Number	(MSB) (LSB)

Figure 20. Defect Descriptors of Physical Sector Format

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

The defect list format field specifies the format of the defect list data returned by the target.

The Defect List Length field specifies the length in bytes of the defect descriptors that follow. The Defect List Length is equal to eight times the number of defect descriptors.

If the Allocation Length is insufficient to transfer all of the defect descriptors, the Defect List Length will not be adjusted to reflect the truncation. The target will not create a CHECK CONDITION status.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.11 READ EXTENDED (28)

	BIT 7	6	5	4	3	2	1	0
BYTE 0	Command Code = 28h							
BYTE 1	LUN			DPO = 0	FUA	Reserved = 0		RelAdr = 0
BYTE 2	(MSB) Logical Block Address							
BYTE 3								
BYTE 4								
BYTE 5								(LSB)
BYTE 6	Reserved = 0							
BYTE 7	(MSB) Transfer Length							
BYTE 8								(LSB)
BYTE 9	VU = 0		Reserved = 0				FLAG	LINK

Figure 21. Read Extended (28)

The READ EXTENDED command requests the file to transfer data to the initiator. The transfer length field permits larger than can be specified in the READ command.

DPO Disable page out. **Must be set to zero** Disable page out is not supported.

FUA Force unit access. A FUA bit of 1 indicates that the data is read from the media and not from the cache. A FUA bit of 0 allows the data to be read from either the media or the cache.

RelAdr Relative Block Address. **Must be set to zero**, indicating that the logical block address field specifies the first logical block of the range of logical blocks to be operated on by this command. Relative address is not supported.

Transfer length The number of contiguous blocks to be transferred. If the transfer length is zero,the seek occurs,but no data is transferred. This condition is not considered an error. If read ahead is enabled, a read ahead is started after the seek completes.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.12 READ BUFFER (3C)

	BIT 7	6	5	4	3	2	1	0
BYTE 0	Command Code = 3Ch							
BYTE 1	LUN			RSVD = 0		MODE		
BYTE 2	Buffer ID = 0							
BYTE 3	(MSB) Buffer Offset							
BYTE 4								
BYTE 5								(LSB)
BYTE 6	(MSB) Allocation length							
BYTE 7								
BYTE 8								(LSB)
BYTE 9	VU = 0		RSVD = 0				FLAG	LINK

Figure 22. READ BUFFER (3C)

The READ BUFFER command is used in conjunction with the WRITE BUFFER command as a diagnostic function for testing the file's memory and the SCSI bus integrity. This command does not alter the medium.

The function of this command and the meaning of fields within the command descriptor block depend on the contents of the mode field.

MODE	Description
000	Read combined header and data
010	Data
011	Descriptor
All others	Not supported.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.12.1 Combined Header And Data (Mode 000)

In this mode, a four byte header followed by data bytes are returned to the initiator during the DATA IN phase. The buffer ID and the buffer offset field are reserved.

The file terminates the DATA IN phase when allocation length bytes of header plus data have been transferred or when the header and all available data have been transferred to the initiator, whichever is less.

The four-byte READ BUFFER header (Figure 23) is followed by data bytes from the file's data buffer.

	BIT 7	6	5	4	3	2	1	0
BYTE 0	RSVD = 0							
BYTE 1	(MSB) Buffer Capacity							
BYTE 2								
BYTE 3								(LSB)

Figure 23. READ BUFFER Header

The buffer capacity specifies the total number of data bytes that are available in the file's data buffer. This number is not reduced to reflect the allocation length nor is it reduced to reflect the actual number of bytes written using the WRITE BUFFER command.

Following the READ BUFFER header, the file will transfer data from its data buffer.

1.12.2 Read Data (Mode 010b)

In this mode, the DATA IN phase contains buffer data.

Buffer ID This field must be set to zero, indicating the data transfer buffer If other value is specified, the command is terminated with CHECK CONDITION status. File shall set sense key to ILLEGAL REQUEST and additional sense code to ILLEGAL FIELD IN CDB.

Buffer Offset This specifies the offset of the memory space specified by the Buffer ID. The initiator should conform to the offset boundary requirements returned in the READ BUFFER descriptor. If the value exceeds the buffer specified, the command is terminated with CHECK CONDITION status. File shall set sense key to ILLEGAL REQUEST and additional sense code to ILLEGAL FIELD IN CDB.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

Allocation Length The file terminates the DATA IN phase when allocation length bytes of data have been transferred or when the header and all available data have been transferred to the initiator, whichever is less.

1.12.3 Descriptor (Mode 011b)

In this mode, a maximum of four bytes of READ BUFFER descriptor information are returned. The file returns the descriptor information for the buffer specified by the buffer ID.

Buffer ID This field should normally be set to zero indicating the file data transfer buffer. If any other value is specified the file returns all zeros in the READ BUFFER descriptor.

Buffer Offset This field is reserved.

Allocation Length This must be set to four or greater. The file transfers the lesser of the allocation length or four bytes of READ BUFFER descriptor. The READ BUFFER descriptor is defined in Figure 24.

	BIT 7	6	5	4	3	2	1	0
BYTE 0	Offset Boundary							
BYTE 1	(MSB)	Buffer Capacity						
BYTE 2								
BYTE 3								(LSB)

Figure 24. READ BUFFER DESCRIPTOR

The value contained in the Buffer Offset field of subsequent WRITE BUFFER and READ BUFFER commands should be a multiple of two to the power of the offset boundary. The offset boundary is always set to nine, which indicates Sector(512 bytes) Boundaries.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.13 READ LONG (3E)

	BIT 7	6	5	4	3	2	1	0
BYTE 0	Command Code = 3Eh							
BYTE 1	LUN				Reserved = 0		CORT = 0	RelAdr = 0
BYTE 2 BYTE 3 BYTE 4 BYTE 5	(MSB) LOGICAL BLOCK ADDRESS (LSB)							
BYTE 6	Reserved							
BYTE 7 BYTE 8	(MSB) Byte Transfer Length (LSB)							
BYTE 9	VU = 0			RSVD = 0			FLAG	LINK

Figure 25. READ LONG (3E)

The READ LONG command requests the file to transfer **one block** of data to the initiator. The transfer data includes;

- 512 bytes of data
- ECC field data
- **CORT**

  0 A corrected bit of zero causes the logical block to be read without any correction made by the file.

  1 Not supported by the file. (A corrected bit of one causes the data to be corrected by ECC before transferring the data to the initiator.)

- **RelAdr** Relative Block Address is not supported by the file.
- **LOGICAL BLOCK ADDRESS** field specifies the logical block at which the read operation shall occur.
- **Byte Transfer Length**. This field must exactly specify the number of bytes of data that are available for transfer. If a non-zero byte transfer length does not match the available data length, the target terminates the command with CHECK CONDITION status, the sense key is set to ILLEGAL REQUEST and an additional sense code set to INVALID FIELD IN CDB. The valid and ILI bits is set

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

to one and the information field is set to the difference of the requested length minus the actual length in bytes. Negative values is indicated by two's complement notation.

The transfer length is calculated as follows:
transfer length = logical block size + 16

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.14 REASSIGN BLOCKS (07)

	BIT 7	6	5	4	3	2	1	0
BYTE 0	Command Code = 07h							
BYTE 1	LUN				RSVD = 0			
BYTE 2	RSVD = 0							
BYTE 3	RSVD = 0							
BYTE 4	RSVD = 0							
BYTE 5	VU = 0		RSVD = 0				FLAG	LINK

Figure 26. REASSIGN BLOCKS (07)

The REASSIGN BLOCKS command requests the file to reassign a logical block to an available spare. The REASSIGN BLOCKS command attempts to allocate spare blocks on a spare track. The logical block address is transferred to the file during the DATA OUT phase. One to four block(s) may be specified for relocation per REASSIGN BLOCKS command.

Reassignment is complete upon the completion of the REASSIGN BLOCKS command. At this time, the defective logical block address has been added to the grown (“G” list) defect list.

Data contained at the logical block address being reassigned is not preserved by the file, and is filled with a constant pattern.

Following is the format of the data sent by the initiator during the DATA OUT phase:

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

	BIT 7	6	5	4	3	2	1	0
BYTE 0	RSVD = 0							
BYTE 1	RSVD = 0							
BYTE 2	(MSB) Defect list length = 4/8/12/16							
BYTE 3								(LSB)
BYTE 4	(MSB) Defective							
BYTE 5	Logical							
BYTE 6	Block							
BYTE 7	Address –1							(LSB)
BYTE 8	(MSB) Defective							
BYTE 9	Logical							
BYTE 10	Block							
BYTE 11	Address –2							(LSB)
BYTE 12	(MSB) Defective							
BYTE 13	Logical							
BYTE 14	Block							
BYTE 15	Address –3							(LSB)
BYTE 16	(MSB) Defective							
BYTE 17	Logical							
BYTE 18	Block							
BYTE 19	Address –4							(LSB)

Figure 27. Format of REASSIGN BLOCKS data

Note: If the file finds a defective block by verifying ECC before it finds a spare, the file will not start the REASSIGN BLOCKS process, but will return CHECK CONDITION status with sense key set to MEDIUM ERROR.

- **Defect list length** must be 4,8,12 or 16. Otherwise, the drive returns Check Condition with Sense key = Illegal request.
- **Defective logical block address** is 4 byte length. The initiator can specify from 1 to 4 of Defective logical block address according to the Defect list length from 4 to 16, respectively. Defective logical block addresses must be ordered in ascending order, or the drive returns Check Condition.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.15 RELEASE (17)

	BIT 7	6	5	4	3	2	1	0
BYTE 0	Command Code = 17h							
BYTE 1	LUN			3rdPty	3rd Party ID			Ext=0
BYTE 2	Reservation Identification							
BYTE 3	RSVD = 0							
BYTE 4	RSVD = 0							
BYTE 5	VU = 0		RSVD = 0				FLAG	LINK

Figure 28. RELEASE (17)

The RELEASE command is used to release a LUN previously reserved.

Note: It is not an error for an initiator to release a LUN that is not currently reserved.

- **3rdPty** bit indicates that :
  - **1** This release process is for a third party which is specified by 3rd Party ID.
  - **0** This release process is for the initiator itself.
- **3rd Party ID** specifies the ID of the third party for which the LUN is reserved.[3]
- **Ext**ents must be 0. Extension is not supported by the file.
- **Reservation Identification** field is ignored.

[3] Refer 1.17, “RESERVE (16)” on page 43

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.16 REQUEST SENSE (03)

	BIT 7	6	5	4	3	2	1	0
BYTE 0	Command Code = 03h							
BYTE 1	LUN			RSVD = 0				
BYTE 2	RSVD = 0							
BYTE 3	RSVD = 0							
BYTE 4	ALLOCATION LENGTH							
BYTE 5	VU = 0		RSVD = 0				FLAG	LINK

Figure 29. REQUEST SENSE (03)

The REQUEST SENSE command requests the file to transfer sense data.

The sense data shall be available when following conditions,

- The previous command to the specified I_T_L nexus terminated with CHECK CONDITION status.[4]
- An other information (e.g. medium position) is available in any fields.
- The previous command to the specified I_T_L nexus ended unexpected BUS FREE error.

If REQUEST SENSE command with a invalid LUN is received, file return GOOD status and report a sense key of ILLEGAL REQUEST and an additional sense code of LOGICAL UNIT NOT SUPPORTED.

If the file has no sense data available to return, it shall return a sense key of NO SENSE and an additional sense code of NO ADDITIONAL SENSE INFORMATION.

The sense data shall be preserved by the file for the initiator until retrieved by the REQUEST SENSE command or until any other command for the same I_T_L nexus. Sense data shall be cleared upon receipt subsequent command including REQUEST SENSE to the same I_T_L nexus.

Separate sense data is maintained by the device for each initiator. Therefore, there is no requirement for an initiator to expeditiously clear a CHECK CONDITION as this will not affect other initiators in a multi-initiator system.

[4] **I_T_L nexus .** A nexus which exists between an initiator, a target and a logical unit.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

The file will return the number of bytes in the allocation length or 32 bytes whichever is less.

The contents of the sense data is defined in 3.0, “SCSI MESSAGE SYSTEM” on page 65.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.17 RESERVE (16)

	BIT 7	6	5	4	3	2	1	0
BYTE 0	Command Code = 16h							
BYTE 1	LUN			3rdPty	3rd Party ID			Ext=0
BYTE 2	Reservation Identification							
BYTE 3	(MSB) Extent List Length = 0							
BYTE 4								(LSB)
BYTE 5	VU = 0		RSVD = 0				FLAG	LINK

Figure 30. RESERVE (16)

The RESERVE command is used to reserve a LUN for an initiator. This reservation can be either for;

- The initiator which sends this command.
- The third party which is specified in this command.

This command results in reserving the entire LUN for the initiator until one of the following occurs:

- The reservation is superseded by another valid RESERVE command from the initiator that made the reservation.
- The LUN is released by a RELEASE command from the same initiator.
- A hard reset condition occurs. (A SCSI bus Reset assertion)
- A BUS DEVICE RESET message is received from any initiator.
- Power off/on occurs.
- **3rdPty** bit is to indicates that :

  1 This reservation is for a third party which is specified by 3rd Party ID.

  0 This reservation is for the initiator itself.
- **3rd Party ID** specifies the ID of the third party for which the LUN is reserved.

  Note: The LUN may be only released by the initiator who sent the RESERVE command.
- **Ext**ents must be 0. Extension is not supported by the file.
- **Reservation Identification** is ignored.
- **Extent List** length must be zero. Extent List length is not supported.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.17.1 Command Processing While Reserved

When a reservation for a LUN is made by an initiator for the same or a different initiator (Third Party option), the following rules apply.

- If the issuing initiator is the one that made the reservation and also the one to receive the reservation then :
  - All commands are permitted.
- If the issuing initiator is neither the one that made the reservation nor the one that receive the reservation then :
  - A REQUEST SENSE or INQUIRY command are permitted.
  - A RELEASE command is permitted but is ignored.
  - Any other command result in a RESERVATION CONFLICT status.
- If the issuing initiator is the one that made the reservation but is not the one to receive the reservation then :
  - A REQUEST SENSE, INQUIRY, RELEASE or RESERVE command is permitted.
  - Any other command result in a RESERVATION CONFLICT status.
- If the issuing initiator is not the one that made the reservation but is the one to receive the reservation then :
  - RESERVE command result in a RESERVATION CONFLICT status.
  - A RELEASE command is permitted but is ignored.
  - Any other command is permitted.

Note: A third party reservation (3rd pty bit equal to one) with the Third Party ID set equal to SCSI ID of the initiator that issued RESERVE command is not distinguishable from a RESERVE command with the Third Party bit equal to zero.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.18 REZERO UNIT (01)

	BIT 7	6	5	4	3	2	1	0
BYTE 0	Command Code = 01h							
BYTE 1	LUN				RSVD = 0			
BYTE 2	RSVD = 0							
BYTE 3	RSVD = 0							
BYTE 4	RSVD = 0							
BYTE 5	VU = 0		RSVD = 0				FLAG	LINK

Figure 31. REZERO UNIT (01)

The REZERO UNIT command requests that the target seek to logical block address 0.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.19 SEEK (0B)

	BIT 7	6	5	4	3	2	1	0
BYTE 0	Command Code = 0Bh							
BYTE 1	LUN			(MSB)	LBA			
BYTE 2	LOGICAL BLOCK ADDRESS							
BYTE 3	LOGICAL BLOCK ADDRESS (LSB)							
BYTE 4	Reserved							
BYTE 5	VU = 0		RSVD = 0				FLAG	LINK

Figure 32. SEEK (0B)

The SEEK command requests the file to seek to the specified logical block address.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.20 SEEK EXTENDED (2B)

	BIT 7	6	5	4	3	2	1	0
BYTE 0	Command Code = 2Bh							
BYTE 1	LUN			RSVD = 0				0
BYTE 2	(MSB)		Logical Block Address					
BYTE 3								
BYTE 4								
BYTE 5								(LSB)
BYTE 6	RSVD = 0							
BYTE 7	RSVD = 0							
BYTE 8	RSVD = 0							
BYTE 9	VU = 0		RSVD = 0				FLAG	LINK

Figure 33. SEEK EXTENDED (2B)

The SEEK EXTENDED command requests the file to seek to the specified logical block address.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.21 SEND DIAGNOSTIC (1D)

	BIT 7	6	5	4	3	2	1	0
BYTE 0	Command Code = 1Dh							
BYTE 1	LUN			RSVD = 0		SlfTst = 1	DevOfl = 0	UntOfl = 0
BYTE 2	RSVD = 0							
BYTE 3	(MSB) Parameter List Length							
BYTE 4								(LSB)
BYTE 5	VU = 0		RSVD = 0				FLAG	LINK

Figure 34. SEND DIAGNOSTIC (1D)

The SEND DIAGNOSTIC command requests the file to perform its self-diagnostic test.

- **SlfTst** must be 1, Indicating default self-tests.
- **DevOfl** must be 0, Indicating that the diagnostic operations don't effect subsequent command.
- **UntOfl** must be 0, Indicating that the diagnostic operations don't effect subsequent command.
- **Parameter List Length** is ignored by the Target.

Upon command completion, the following status is returned:

- GOOD status for successful test completion.
- CHECK CONDITION status for unsuccessful test completions.

The self diagnostics consists of two parts :[5]

- The first part is executed immediately after power up. This test is performed to verify all hardware which is not related to the disk drive. The local microprocessor, ROM (checksum), RAM (scratchpad and buffer), and control electronics are included here.
- The second part is executed after the spindle motor is started. This includes disk access (seek), R/W channel, and error correction circuitry verification. A reserved area on the disk is used for this test.

[5] See 4.19.2, “Diagnostics Command” on page 110 for a detailed listing of the operations carried out by the Diagnostics Command.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

Both tests are performed as a result of the SEND DIAGNOSTIC command. The SEND DIAGNOSTICS will fail with CHECK CONDITION status if it is issued while the spindle motor is not turning. (Such as after STOP command has been received.)

Note: The self diagnostic is also performed at Power On Reset time.

Note: The SCSI bus signals will not be corrupted when the device is executing the SEND DIAGNOSTIC command.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.22 START/STOP UNIT (1B)

	BIT 7	6	5	4	3	2	1	0
BYTE 0	Command Code = 1Bh							
BYTE 1	LUN			RSVD = 0				Immed
BYTE 2	RSVD = 0							
BYTE 3	RSVD = 0							
BYTE 4	RSVD = 0							Start
BYTE 5	VU = 0		RSVD = 0				FLAG	LINK

Figure 35. START/STOP Unit (1B)

The START/STOP UNIT command is used to spin up or stop the spindle motor.

- **Immed** bit is to specify

  0 Status is to be returned at the end of the operation.
  1 GOOD status shall always be returned immediately after command has been received. The TEST UNIT READY command may be used to determine when the file becomes ready after a spin-up.

- **Start** bit is to specify:

  0 Stop the spindle.
  1 Start the spindle.

  Note: Once the drive has become ready (after a power on) the Start/Stop UNIT command can be used without any errors, regardless of the state of the motor, stopped or spinning.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.23 SYNCHRONIZE CACHE (35)

	BIT 7	6	5	4	3	2	1	0
BYTE 0	Command Code = 35h							
BYTE 1	LUN			RSVD = 0			Immed = 0	RelAdr = 0
BYTE 2 BYTE 3 BYTE 4 BYTE 5	(MSB) Logical Block Address (LSB)							
BYTE 6	Reserved							
BYTE 7 BYTE 8	(MSB) Number of Blocks (LSB)							
BYTE 9	VU = 0		RSVD = 0				FLAG	LINK

Figure 36. SYNCHRONIZE CACHE (35)

The SYNCHRONIZE CACHE Command ensures that logical blocks in the cache have their most recent data value recorded on the media.

- **Logical Block Address** is to specify:
  where the operation is to begin.
- **Number of Blocks** specifies:
  the total number of contiguous logical blocks within the range. Number of Blocks of zero indicates that all remaining logical blocks on the logical unit shall be within the range.
- **Immed** (immediate) must be zero.
  An immediate bit of zero indicates that the status shall not be returned until the operation has completed.
  If the Immd bit is set to one, the drive returns a Check Condition status. The sense key shall be set to Illegal Request and the additional sense code shall be set to Invalid Field in CDB.
- **RelAdr** (relative address) must be zero.
  The drive does not support the relative addressing.
  If the RelAdr bit is set to one, the drive returns Check Condition status. The sense key shall be set to Illegal Request and the additional sense code shall be set to Invalid Field in CDB.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.24 TEST UNIT READY (00)

	BIT 7	6	5	4	3	2	1	0
BYTE 0	Command Code = 00h							
BYTE 1	LUN				RSVD = 0			
BYTE 2	RSVD = 0							
BYTE 3	RSVD = 0							
BYTE 4	RSVD = 0							
BYTE 5	VU = 0		RSVD = 0				FLAG	LINK

Figure 37. TEST UNIT READY (00)

The TEST UNIT READY command allows the initiator to check if the file is READY. The SCSI specification defines READY as the condition where the device will accept a media-access command without returning CHECK CONDITION status.

The file will first verify that the motor is spinning at the correct speed.

- If the spindle motor is not spinning at the correct speed, CHECK CONDITION status is returned with sense key of NOT READY.
- If the motor is spinning at the correct speed, the file accepts normal media access commands.

The TEST UNIT READY command is not intended as a diagnostic. No self diagnostic is performed by the device as a result of this command.

The TEST UNIT READY command has special significance for power sequencing using the START UNIT command with an Immediate bit of 1. In this mode the START UNIT command returns COMMAND COMPLETE status before the completion of motor spin-up and expects the initiator to issue TEST UNIT READY commands to determine when the motor has reached the proper speed.

Note: The spindle automatically starts in automatic spin-up Mode. The file does not execute any commands other than TEST UNIT READY, INQUIRY or REQUEST SENSE command until the the Power On sequence is complete. The file will return CHECK CONDITION status with NOT READY sense key and IN PROCESS OF BECOMING READY sense code for all other commands during the Power On period.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.25 VERIFY (2F)

	BIT 7	6	5	4	3	2	1	0
BYTE 0	Command Code = 2Fh							
BYTE 1	LUN			RSVD = 0			ByteChk	0
BYTE 2	(MSB) Logical Block Address							
BYTE 3								
BYTE 4								
BYTE 5								(LSB)
BYTE 6	RSVD = 0							
BYTE 7	(MSB) Transfer Length							
BYTE 8								(LSB)
BYTE 9	VU = 0		RSVD = 0				FLAG	LINK

Figure 38. VERIFY (2F)

The VERIFY command requests that the file verify the data written on the media. A verification length of zero indicates that no data will be transferred. This condition is not considered an error.

- **ByteChk** indicates;

  0 The verification is performed by ECC check. No data transfer from the initiator is performed in this case. If an ECC check is detected on all re-reads and the data was not corrected(either because it was uncorrectable or the correction was not attempted), a Check Condition status is returned with a Medium Error sense key.

  1 Byte-by-byte comparison is not supported.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.26 WRITE (0A)

	BIT 7	6	5	4	3	2	1	0
BYTE 0	Command Code = 0Ah							
BYTE 1	LUN			(MSB) LBA				
BYTE 2	LOGICAL BLOCK ADDRESS							
BYTE 3	LOGICAL BLOCK ADDRESS (LSB)							
BYTE 4	TRANSFER LENGTH							
BYTE 5	VU = 0		RSVD = 0				FLAG	LINK

Figure 39. WRITE (0A)

The WRITE command requests the file to write the specified number of blocks of data from the initiator to the medium starting at the specified logical block address.

See 1.8, “READ (08)” on page 25 for the parameters.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.27 WRITE EXTENDED (2A)

	BIT 7	6	5	4	3	2	1	0
BYTE 0	Command Code = 2Ah							
BYTE 1	LUN			DPO = 0	FUA		RSVD = 0	RelAdr = 0
BYTE 2 BYTE 3 BYTE 4 BYTE 5	(MSB) Logical Block Address							(LSB)
BYTE 6	RSVD = 0							
BYTE 7 BYTE 8	(MSB) Transfer Length							(LSB)
BYTE 9	VU = 0		RSVD = 0				FLAG	LINK

Figure 40. WRITE EXTENDED (2A)

The WRITE EXTENDED command requests that the file write the data transferred from the initiator. This command is processed like the standard WRITE command except for the longer transfer length .

DPO Disable page out. **Must be set to zero** Disable page out is not supported.

FUA Force unit access. A FUA bit of 1 indicates that the Target must write the data to the media before returning Good Status. A FUA bit of 0 indicates the Target may return Good Status prior to writing the data to the media.

RelAdr Relative Block Address. **Must be set to zero**, indicating that the logical block address field specifies the first logical block of the range of logical blocks to be operated on by this command. Relative address is not supported.

Transfer length The number of contiguous blocks to be transferred. If the transfer length is zero,the seek occurs,but no data is transferred. This condition is not considered an error.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.28 WRITE AND VERIFY (2E)

	BIT 7	6	5	4	3	2	1	0
BYTE 0	Command Code = 2Eh							
BYTE 1	LUN			DPO = 0	RSVD = 0		ByteChk = 0	RelAdr = 0
BYTE 2	(MSB) Logical Block Address							
BYTE 3								
BYTE 4								
BYTE 5								(LSB)
BYTE 6	RSVD = 0							
BYTE 7	(MSB) Transfer Length							
BYTE 8								(LSB)
BYTE 9	VU = 0		RSVD = 0				FLAG	LINK

Figure 41. WRITE AND VERIFY (2E)

WRITE AND VERIFY command requests that the file writes the data transferred from the initiator to the medium and then verify that the data is correctly written.

- **ByteChk** the options:

ByteChk	Description
0	The data is read back from the disk and verified using ECC after the successful write operation. If an ECC error is detected in the verify process, CHECK CONDITION status is returned with sense key set to MEDIUM ERROR.
1	Not supported by the file.

- DPO(Disable page out) **Must be set to zero** Disable page out is not supported.
- Relative Block Address **Must be set to zero**, indicating that the logical block address field specifies the first logical block of the range of logical blocks to be operated on by this command. Relative address is not supported.
- A transfer length of zero indicates that no data is transferred.
- If caching is enabled, the command performs an implied Force Unit Access (FUA) and an implied Synchronize Cache before starting the operation. This insures that the medium, not the cache, is being verified.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.29 WRITE BUFFER (3B)

	BIT 7	6	5	4	3	2	1	0
BYTE 0	Command Code = 3Bh							
BYTE 1	LUN			RSVD = 0		MODE		
BYTE 2	Buffer ID							
BYTE 3	(MSB) Buffer Offset							
BYTE 4								
BYTE 5								(LSB)
BYTE 6	(MSB) Parameter list length							
BYTE 7								
BYTE 8								(LSB)
BYTE 9	VU = 0		RSVD = 0				FLAG	LINK

Figure 42. WRITE BUFFER (3B)

The WRITE BUFFER command is used in conjunction with the READ BUFFER command as a diagnostic function for testing the file's memory and the SCSI bus integrity. This command does not alter the medium of the file. Additional modes are provided for downloading microcode and for downloading and saving microcode.

This command will cause the entire cache to be emptied.

The function of this command and the meaning of fields within the command descriptor block depend on the contents of the mode field.

MODE	Description
000	Write combined header and data
010	Data
100	Download Microcode
101	Download Microcode and Save
-	All other modes are not supported by the file.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.29.1 Combined Header And Data (Mode 000b)

In this mode, the data to be transferred is preceded by a four-byte header.

Buffer ID This field must be zero. If another value is specified, no download function are performed and the command is terminated with CHECK CONDITION status. And File shall set sense key to ILLEGAL REQUEST and additional sense code to ILLEGAL FIELD IN CDB.

Buffer Offset This field must be zero. If another value is specified, no download function are performed and the command is terminated with CHECK CONDITION status. And File shall set sense key to ILLEGAL REQUEST and additional sense code to ILLEGAL FIELD IN CDB.

Parameter List Length This field specifies the number of bytes that shall be transferred during the DATA OUT phase. This number **includes** four bytes of header, so the data length to be stored in the file's buffer is transfer length minus four. If the length exceeds the buffer size, the command is terminated with CHECK CONDITION status. And File shall set sense key to ILLEGAL REQUEST and additional sense code to ILLEGAL FIELD IN CDB.

The four-byte header consists of all reserved bytes.

	BIT 7	6	5	4	3	2	1	0
BYTE 0	RSVD = 0							
BYTE 1	RSVD = 0							
BYTE 2	RSVD = 0							
BYTE 3	RSVD = 0							

Figure 43. WRITE BUFFER Header

1.29.2 Write Data (Mode 010b)

In this mode, the DATA OUT phase contains buffer data.

Buffer ID This field must be set to zero, indicating the data transfer buffer If other value is specified, the command is terminated with CHECK CONDITION status. File shall set sense key to ILLEGAL REQUEST and additional sense code to ILLEGAL FIELD IN CDB.

Buffer Offset This specifies the offset of the memory space specified by the Buffer ID. The initiator should conform to the offset boundary requirements returned in the READ BUFFER descriptor. If the value exceeds the buffer specified, the command is terminated with CHECK CONDITION status. File shall set

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

sense key to ILLEGAL REQUEST and additional sense code to ILLEGAL FIELD IN CDB.

Parameter List Length This field specifies the Parameter List Length. It must be;

- less than the capacity of the buffer size.
- on a sector boundary. In other words, it must be a multiple of 512.

If a invalid value is specified, the command is terminated with CHECK CONDITION status. File shall set sense key to ILLEGAL REQUEST and additional sense code to ILLEGAL FIELD IN CDB.

1.29.3 Download Microcode (Mode 100b)

In this mode, the microcode is transferred to the control memory space of the file. Once downloaded the file will operate with the newly downloaded code until the next power cycle.

Buffer ID The buffer ID field is used to indicate which portion of the microcode image is being downloaded. If it is set to '00'x and the length of the download is '8000'x then just the main microprocessor code is updated. If the buffer ID is '00'x and the length is '10000'x then both the main micro and the HDC picocode are updated. If the buffer ID is '01'x and the length is '8000'x then just the HDC pico code is updated. Using combinations of buffer ID and length it is possible to updated both sets of code using a single write buffer command or else to individually update the codes.

Any other value for the buffer ID except '00'x and '01'x will cause the command to terminate with CHECK CONDITION status. The file shall set sense key to ILLEGAL REQUEST and additional sense code to ILLEGAL FIELD IN CDB.

Buffer Offset This specifies the starting address of the downloaded Microcode. It must be zero. If an invalid value is specified, the command is terminated with CHECK CONDITION status . File shall set sense key to ILLEGAL REQUEST and additional sense code to ILLEGAL FIELD IN CDB.

Transfer Length Total Microcode length must be specified. As detailed in the section on the buffer ID the length field may take a value 0f '8000'x to update a single processors code or else '10000'x to update both processors. It may also be set '0000'x in which case no code is updated. If an invalid value is specified, the command is terminated with CHECK CONDITION status. File shall set sense key to ILLEGAL REQUEST and additional sense code to ILLEGAL FIELD IN CDB.

Note: This process generates a unit attention condition for MICROCODE HAS BEEN CHANGED.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.29.4 Download Microcode and Save (Mode 101b)

In this mode, the microcode is transferred to the file and saved into the System reserved area on the disk. The downloaded code becomes effective after each Power On Reset until it is supplanted in another download microcode and save option.

The code to download in this operation will be supplied to the using system as a single binary image of 160Kb in size. This contains code for the main processor, it's overlays and also the HDC pico processor. The host system may download this entire image in one write buffer command or else it may split the code into multiple blocks of '8000'x bytes in length and send them using multiple write buffer commands. If this latter course of action is taken then file will store this code on a temporary area of the disk until the reception of the last block of code. The file will then checksum the code and perform other required checks and then write the code onto the disk replacing the current code within the file. It will also load the complete code into the processors memory maps.

Buffer ID

The buffer ID field is used to indicate which portion of the microcode image is being downloaded. If it is set to '00'x and the length of the download is '28000'x then all the code is to be downloaded in a single command and will replace the current microcode.

If the code is to be downloaded in blocks then the buffer ID must be set to '00'x on the first block, '01'x on the second block etc. The blocks must be sent to the file in the correct logical order, with no omissions or resends of blocks. On all write buffer commands except the final one the file merely stores the code and no action is taken. When the final block is received the file then attempts to use the new microcode. In between downloading the code blocks the file will continue to accept and process other commands from this and any other initiator. It is not until the final block of code has been received that the code the processors is running is changed. In addition it is legal to send some but not all of the code and then to start sending a different level of code as long as this second level of code is started from the beginning, ie buffer ID '00'x.

Any value for the buffer ID except '00'x or an incrementing value based on the last block sent will cause the command to terminate with CHECK CONDITION status. The file shall set sense key to ILLEGAL REQUEST and additional sense code to ILLEGAL FIELD IN CDB.

Buffer Offset

This specifies the starting address of the downloaded Microcode. It must be zero. If an invalid value is specified, the command is terminated with CHECK CONDITION status . File shall set sense key to ILLEGAL REQUEST and additional sense code to ILLEGAL FIELD IN CDB.

Transfer Length

Total Microcode length must be specified. As detailed in the section on the buffer ID the length field may take a value 0f '8000'x to update a single processors code or else '28000'x . It may also be set '0000'x in which case no code is updated. If an invalid value is specified, the command is terminated with CHECK CONDITION status. File shall set sense key to ILLEGAL REQUEST and additional sense code to ILLEGAL FIELD IN CDB.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

Note: This process generates a unit attention condition for MICROCODE HAS BEEN CHANGED.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1.30 WRITE LONG (3F)

	BIT 7	6	5	4	3	2	1	0
BYTE 0	Command Code = 3Fh							
BYTE 1	LUN			Reserved = 0				RelAd = 0
BYTE 2 BYTE 3 BYTE 4 BYTE 5	(MSB) LOGICAL BLOCK ADDRESS (LSB)							
BYTE 6	Reserved = 0							
BYTE 7 BYTE 8	(MSB) Byte Transfer Length (LSB)							
BYTE 9	VU = 0		RSVD = 0				FLAG	LINK

Figure 44. WRITE LONG (3F)

The WRITE LONG command requests the file to write **one block** of data transferred from the initiator. The transfer data must include;

- 512 bytes of data
- 16 bytes of ECC data

Parameters are;

- **RelAd** (Relative Block Address). This is not supported by the file.
- **LOGICAL BLOCK ADDRESS** field specifies the logical block at which the write operation shall occur.
- **Byte Transfer Length**. This field must exactly specify the number of bytes of data that are available for transfer. If a non-zero byte transfer length does not match the available data length, the target terminates the command with CHECK CONDITION status, the sense key is set to ILLEGAL REQUEST and an additional sense code set to INVALID FIELD IN CDB. The valid and ILI bits is set to one and the information field is set to the difference of the requested length minus the actual length in bytes. Negative values is indicated by two's complement notation.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

2.0 SCSI Status Byte

Upon the completion of a command, a status byte is sent to the initiator. Additional sense information may also be available depending on the contents of the status byte. The following section describes the possible values for the status byte and sense data.

	BIT 7	6	5	4	3	2	1	0
STATUS	Reserved = 0				Status Code			RSVD

Figure 45. SCSI Status Byte. Format of the SCSI STATUS byte. All Reserved fields(R) are set to zero.

STATUS BYTE **Description**

00h GOOD
The command has been successfully completed.

02h CHECK CONDITION
An error, exception, or abnormal condition has been detected. The sense data is set by the file. The REQUEST SENSE command should be issued to determine the nature of the condition.

04h CONDITION MET
This status indicates that the requested operation is satisfied. (See "Pre-Fetch" Command)

08h BUSY
This condition is returned when disconnect privilege is not granted while the file is BUSY processing the other command for the other initiator. The normal initiator recovery action is to issue the command at a later time, or reissue the command and grant the disconnect privilege.

10h INTERMEDIATE/GOOD
This status is returned for every command in a series of linked commands (except the last command), unless an error, exception, or abnormal condition causes a CHECK CONDITION status or a RESERVATION CONFLICT status to be set. If this status is not returned, the chain of linked commands is broken, and no further commands in the series are executed.

14h INTERMEDIATE/CONDITION MET
This status is the combination of CONDITION MET and INTERMEDIATE /GOOD.

18h RESERVATION CONFLICT
This status is returned whenever an SCSI device attempts to access the file, but it has been reserved by another initiator. (See 1.17, “RESERVE (16)” on page 43.)

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

28h QUEUE FULL

This status indicates that the targets command queue is full. If tagged command queuing feature is enabled and there is no room on the command queue, this status is returned when the initiator sends a command. For this status, sense is not valid.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

3.0 SCSI MESSAGE SYSTEM

This chapter details how the message system is implemented on the file. Included is a functional description of the supported messages.

3.1 Supported Messages

The message supported by the file is shows in Figure 46.

MESSAGE	CODE (hex)	Direction		Negate ATN Before last ACK
COMMAND COMPLETE	00	IN		———
SYNCHRONOUS DATA TRANSFER REQUEST	010301	IN	OUT	Yes
SAVE DATA POINTER	02	IN		———
RESTORE POINTERS	03	IN		———
DISCONNECT	04	IN		———
DISCONNECT	04		OUT	Yes
INITIATOR DETECTED ERROR	05		OUT	Yes
ABORT	06		OUT	Yes
MESSAGE REJECT	07	IN	OUT	Yes
NO OPERATION	08		OUT	Yes
MESSAGE PARITY ERROR	09		OUT	Yes
LINKED COMMAND COMPLETE	0A	IN		———
LINKED COMMAND COMPLETE (w/FLAG)	0B	IN		———
BUS DEVICE RESET	0C		OUT	Yes
ABORT TAG	0D		OUT	Yes
CLEAR QUEUE TAG	0E		OUT	Yes
SIMPLE QUEUE TAG	20XX	IN	OUT	No
HEAD OF QUEUE TAG	21XX		OUT	No
ORDERED QUEUE TAG	22XX		OUT	No
IDENTIFY	80–FF	IN		———
IDENTIFY	80–FF		OUT	No

Key: IN = Target to Initiator, OUT = Initiator to target.
YES = Initiator shall negate ATN before last ACK of message.
NO = Initiator may or may not negate ATN before last ACK of message.
——— = Not applicable
XX = Queue Tag

Figure 46. Supported Messages

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

If an unsupported message is received, the file will send the *MESSAGE REJECT* message to the initiator.

3.1.1 COMMAND COMPLETE (00)

The file sends this message to the initiator to indicate that the execution of a command has terminated and that valid status has been sent to the initiator. After successfully sending this message, the file releases all bus signals and goes to BUS FREE phase.

3.1.2 SYNCHRONOUS DATA TRANSFER REQUEST (01,03,01H)

Byte	Value	Description
0	01H	Extended message
1	03H	Extended message length
2	01H	SYNCHRONOUS DATA TRANSFER REQUEST code
3	M	Transfer period (M times 4 nanoseconds)
4	X	REQ/ACK offset

Figure 47. Synchronous Data Transfer Request.

A pair of Synchronous Data Transfer Request (SDTR) messages shown in Figure 47 are exchanged between an initiator and a Target to establish the synchronous data transfer mode between the two devices. The message exchange establishes the permissible transfer period and REQ/ACK offset for a synchronous data transfer between the two devices. The initiator may initiate a synchronous data transfer negotiation at any time after the LUN has been identified. A Synchronous Data Transfer Request(SDTR) message exchange shall be initiated by an SCSI device whenever a previously arranged data transfer agreement may have become invalid. *The agreement becomes invalid after any condition which may leave the data transfer agreement in an indeterminate state such as;*

1. after a Power-on Reset
2. after a SCSI Bus "hard" reset condition
3. after a Bus Device Reset message

In addition, a SCSI device may initiate a SDTR message exchange whenever it is appropriate to negotiate a new data transfer agreement(either synchronous or asynchronous).

M The transfer period(M above) is the minimum time allowed between leading edges of successive REQ pulses and of successive ACK pulses to meet the device requirements for successful reception of data. The file supports transfer period in the range 100 nSec to 475 nSec in 25 nSec increments.

REQ/ACK Offset

The ACK/REQ offset(X above) is the maximum number of REQ pulses allowed to be outstanding before the leading edge of its corresponding ACK pulses is received at the file. A REQ/ACK offset value of zero indicate asynchronous data transfer mode. The file supports REQ/ACK offset values in the range 0 through 15.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

If ATN is negated before all bytes of a multiple-byte extended message is received, the file will go to **BUS FREE** to signal a catastrophic error.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

3.1.2.1 Synchronous Negotiation Started by the Initiator

The file responds to each Initiator requested transfer period as shown in the following figure Figure 48:

Initiator Request	Target Response	Target Transfer Period	Maximum Burst Rate
0 <= Mi <= 25	Mt = 25	100 nSec	10.00 MB/s
26 <= Mi <= 31	Mt = Mi	125 nSec	8.00 MB/s
32 <= Mi <= 37	Mt = Mi	150 nSec	6.67 MB/s
38 <= Mi <= 43	Mt = Mi	175 nSec	5.71 MB/s
44 <= Mi <= 50	Mt = Mi	200 nSec	5.00 MB/s
51 <= Mi <= 56	Mt = Mi	225 nSec	4.44 MB/s
57 <= Mi <= 62	Mt = Mi	250 nSec	4.00 MB/s
63 <= Mi <= 68	Mt = Mi	275 nSec	3.64 MB/s
69 <= Mi <= 75	Mt = Mi	300 nSec	3.33 MB/s
76 <= Mi <= 81	Mt = Mi	325 nSec	3.08 MB/s
82 <= Mi <= 87	Mt = Mi	350 nSec	2.86 MB/s
88 <= Mi <= 93	Mt = Mi	375 nSec	2.67 MB/s
94 <= Mi <= 100	Mt = Mi	400 nSec	2.50 MB/s
101 <= Mi <= 106	Mt = Mi	425 nSec	2.35 MB/s
107 <= Mi <= 112	Mt = Mi	450 nSec	2.22 MB/s
113 <= Mi <= 118	Mt = Mi	475 nSec	2.11 MB/s
119 <= Mi <= 255	Mt = Mi	(Asynchronous mode)	N/A

Figure 48. Initiator Request/Target Response

3.1.2.2 Synchronous Negotiation Started by the Target

If the file recognize that negotiation is required, the file sends a SDR message to the initiator with transfer period equal to 200 nSec (M = 50). The file interprets the Initiator corresponding transfer period as shown in the following figure Figure 49 on page 69:

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

Initiator's Response	Target Transfer Period	Maximum Burst Rate
0 <= Mi <= 24	Send Message Reject(Async mode)	N/A
25 <= Mi <= 25	100 nSec	10.00 MB/s
26 <= Mi <= 31	125 nSec	8.00 MB/s
32 <= Mi <= 37	150 nSec	6.67 MB/s
38 <= Mi <= 43	175 nSec	5.71 MB/s
44 <= Mi <= 50	200 nSec	5.00 MB/s
51 <= Mi <= 56	225 nSec	4.44 MB/s
57 <= Mi <= 62	250 nSec	4.00 MB/s
63 <= Mi <= 68	275 nSec	3.64 MB/s
69 <= Mi <= 75	300 nSec	3.33 MB/s
76 <= Mi <= 81	325 nSec	3.08 MB/s
82 <= Mi <= 87	350 nSec	2.86 MB/s
88 <= Mi <= 93	375 nSec	2.67 MB/s
94 <= Mi <= 100	400 nSec	2.50 MB/s
101 <= Mi <= 106	425 nSec	2.35 MB/s
107 <= Mi <= 112	450 nSec	2.22 MB/s
113 <= Mi <= 118	475 nSec	2.11 MB/s
119 <= Mi <= 255	Send Message Reject(Async mode)	N/A

Figure 49. Target Response to Initiator's Transfer Period

3.1.3 SAVE DATA POINTER (02)

This message is sent from the file to direct the initiator to copy the active data pointer to the saved data pointer. The SAVE DATA POINTER message is only sent if the initiator has previously indicated the ability to accommodate disconnection and reconnection via the IDENTIFY message .

The file will send the SAVE DATA POINTER message to the initiator prior to sending a DISCONNECT message to the initiator if a data phase has occurred and another data phase is required to successfully complete the command.

3.1.4 RESTORE POINTERS (03)

This message is sent from the file to direct an initiator to copy the most recently saved pointers to the corresponding command, data, and status pointers. Command and status pointers should be restored to the beginning of the present command and status areas. The data pointer should be restored to the value at the beginning of the data area in the absence of a SAVE DATA POINTER message or to the value at the point at which the last SAVE DATA POINTER message occurred. Also see 3.4, “SCSI Bus Related Error Handling Protocol” on page 77.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

3.1.5 DISCONNECT (04)

This message is sent from the file to inform an initiator that the present connection is going to be broken. A later reconnect will be required in order to complete the current command. The disconnection is to free the SCSI bus while the file performs a relatively long operation that does not require the bus. These messages are only sent if the initiator previously indicated (via the IDENTIFY message) the ability to accommodate disconnection and reconnection.

The DISCONNECT message may also be sent from the initiator to the file to disconnect from the SCSI bus. If the file supports disconnecting at the time the DISCONNECT message is received from the initiator, the file will switch to the MESSAGE IN phase, send a DISCONNECT message to the initiator (possibly preceded by a SAVE DATA POINTER message), and then go to the BUS FREE phase. The file will not participate in another ARBITRATION phase for at least a disconnection delay. If the file does not support disconnecting at the time the Disconnect message is received from the initiator, the file will respond by sending a MESSAGE REJECT message to the initiator.

3.1.6 INITIATOR DETECTED ERROR (05)

This message is sent from an initiator to inform the file that an error has been detected that does not preclude the file from retrying the previous COMMAND, DATA and STATUS phase. The source of the error may be either related to previous activities on the SCSI bus or may be internal to the initiator and unrelated to any previous SCSI bus activity

If the initiator intends to send this message, the initiator must assert the ATN signal prior to its release of ACK for the last byte transferred in the information phase that is to be retried. This provides an interlock so the file can determine which information phase to retry.

After receiving this message, the file may retry the previous phase by sending a RESTORE POINTERS message to the initiator and then repeating the previous COMMAND, DATA, or STATUS phase.

3.1.7 ABORT (06)

This message is sent from the initiator to direct the file to clear the present operation for this initiator and logical unit, including queued command(s). If a logical unit has been identified, then all pending data and status for the issuing initiator and this logical unit will be cleared and the file will go to the BUS FREE phase. Pending data and status for other logical unit and initiators will not be cleared. If a logical unit has not been identified, the file will go to the BUS FREE phase without affecting an operation on any logical unit for this initiator or any other initiator. In either case, no status or ending message will be sent to the initiator for this operation. It is not an error to send the ABORT message to a logical unit that is not currently performing an operation for the initiator.

Note: It is permissible for an initiator to select the file/LUN after the file has disconnected from the initiator, for the purpose of sending an IDENTIFY message followed by an ABORT message. This will abort the command on the specified logical unit.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

3.1.8 MESSAGE REJECT (07)

This message is sent from either the initiator or the file to indicate that the last message received was inappropriate or has not been implemented.

If the initiator intends to send this message, the initiator must assert the ATN signal prior to its release of ACK for the REQ/ACK handshake of the message byte that is to be rejected. This provides an interlock so the file can determine which message is rejected.

If the file intends to send this message, the file will change to the MESSAGE IN phase and send the MESSAGE REJECT message to the initiator prior to transferring any additional message bytes (or any other information phase bytes) from the initiator regardless of ATN signal. This provides an interlock so the initiator can determine which message is rejected. After the file sends a MESSAGE REJECT message and if ATN signal is still asserted then it shall return to the MESSAGE OUT phase. the subsequent MESSAGE OUT phase shall begin with first byte of a message.

3.1.9 NO OPERATION (08)

This message is sent from the initiator to the file when the initiator does not currently have any other valid message to send. This message is ignored by the file and will not affect any operation.

3.1.10 MESSAGE PARITY ERROR (09)

This message is sent from the initiator to inform the file that the last message byte received had a parity error.

If the initiator intends to send this message, the initiator must assert the ATN signal prior to its release of ACK for the REQ/ACK handshake of the message byte that has the parity error. This provides an interlock so the file can determine which message byte has the parity error.

If the file receives this message under any other circumstance, the file will change to BUS FREE to signal a catastrophic error. After receiving this message, the file will retry sending the previous message to the initiator.

3.1.11 LINKED COMMAND COMPLETE (0A)

The file sends this message to the initiator to indicate that execution of a linked command (with flag bit equal to zero) has completed and that valid status has been sent to the initiator. After successfully sending this message, the file goes to COMMAND phase to receive the next command.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

3.1.12 LINKED COMMAND COMPLETE WITH FLAG (0B)

The file sends this message to the initiator to indicate that the execution of a linked command with flag bit set to one has completed and that valid status has been sent to the initiator. After successfully sending this message, the file goes to COMMAND phase to receive the next command.

3.1.13 BUS DEVICE RESET (0C)

This message is sent from an initiator to direct the file to clear all current commands. This message forces a hard reset condition which will reset the file to an initial state with no operations pending for any initiator. After receiving this message, the file will go to the BUS FREE phase.

3.1.14 ABORT TAG (0D)

When the target receives this message successfully, it clears the current I/O process and go to Bus Free. If the target has already started execution of I/O process, the execution will be halted. Pending status, data and commands for other active or queued I/O process shall not be affected.

3.1.15 CLEAR QUEUE TAG (0E)

All I/O process for all initiators shall be cleared. All active I/O process shall be terminated. The target shall go to the Bus Free phase following successfully receipt of this message.

3.1.16 QUEUE TAG MESSAGES(20h, 21h, 22h)

Byte	Value	Description
0	20H	Simple Queue Tag message
	21H	Head of Queue Tag message
	22H	Ordered Queue Tag message
1	XXh	Queue Tag

Figure 50. Queue Tag Messages

Queue Tag messages are used to specify an identifier, called a Queue Tag, for an I/O process which establish the I_T_L_Q nexus. The queue tag filed is an 8-bit unsigned integer assigned by the initiator during an initial connection. The Queue Tag for every I/O process for each I_T_L nexus must be unique. If the target receives a Queue Tag that is currently in use for the I_T_L nexus it will respond as "Incorrect Initiator Response". A Queue Tag becomes available for re-assignment when I/O process ends. The numeric value of a Queue Tag has no effect on the order of execution.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

Whenever an initiator connects to the target, the appropriate Queue Tag message must be sent immediately following the Identify message and within the same MESSAGE OUT phase to establish the I_T_L_Q nexus for the I/O process.

Whenever the target reconnects to an initiator to continue a tagged I/O process, the Simple Queue Tag message is sent immediately following the Identify and within the same MESSAGE IN phase to revive the I_T_L_Q nexus for the I/O process.

3.1.16.1 SIMPLE QUEUE TAG (20h)

The Simple Queue Tag Message specifies that the current I/O process be placed in the command queue. The order of execution, with respect to other I/O processes received with Simple Queue Tag Messages, is up to the discretion of the target. The target will send a Simple Queue Tag Messages after reselection for I/O processes that were received with either Simple, Ordered, or Head of Queue Tag messages.

3.1.16.2 HEAD OF QUEUE TAG (21h)

Commands with this tag should be inserted into the head of the queue. When a command is being executed, this tagged command will be inserted to the head of queue to be executed after the command being currently executed. The previous command being executed will not terminated by this tagged command. This tagged command will wait until the previous command is through. If plural head-of-queue tagged commands are received, those command will be executed in LIFO(Last in First out) order.

3.1.16.3 ORDERED QUEUE TAG (22h)

This tagged command is executed in the order received. All commands received before this command should be executed before this command, and all commands received after this commands should be executed after this command.

3.1.17 IDENTIFY (80 - FF)

This message is set by either the initiator or the file to establish the logical path connection between the two devices.

The IDENTIFY message is defined as follows:

Bit 7 This bit is always set to one to distinguish the IDENTIFY message from other messages.

Bit 6 This bit is only set to one by the initiator to grant the file the privilege of disconnecting. If this bit is zero, the file will not disconnect, unless the initiator instructs the file to disconnect by sending a DISCONNECT Message to the file. This bit is set to zero when the file sends an IDENTIFY message to the initiator.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

Bits 5-3 These bits are reserved and must be zero for an IDENTIFY message.

Note: If an invalid Identify message is received with these bits not equal to zero, then the file sends a MESSAGE REJECT message to the initiator and goes to the Bus Free phase to signal a catastrophic error condition.

Bits 2-0 These bits specify the logical unit number (LUN).

Only one LUN may be identified for any one selection sequence. If the file receives an IDENTIFY message with a new LUN after the LUN had previously been identified, the file will go to the BUS FREE phase to signal a catastrophic error. The initiator may send more than one Identify message during a selection sequence in order to toggle disconnect/reconnect permission if the specified LUN remains the same.

When the IDENTIFY message is sent from the file to the initiator during reconnection, an implied RESTORE POINTERS message must be performed by the initiator.

3.2 Supported Message Functions

The implementation of the supported messages will also include the following functions.

- Retry SCSI Command, DATA IN, DATA OUT, or STATUS phase
  The retry will be caused by the following error condition.
  - The file detected SCSI bus parity error(Command phase)
  - The file receives INITIATOR DETECTED ERROR MESSAGE during or at the conclusion of an information transfer phase (Command, Data In, Data Out or Status Phase)

    Note: The initiator may send the INITIATOR DETECTED ERROR message as a result of an initiator detected SCSI Bus parity error or an internal error.
- Retry MESSAGE IN phase
  - The retry will be caused by the receipt of a MESSAGE PARITY ERROR message immediately following a MESSAGE IN phase.

    Note: The Initiator may send the MESSAGE PARITY ERROR message as a result of an Initiator detected SCSI Bus parity error during the Message In phase.
- Receipt of multiple Identify message
  - The initiator is allowed to send a multiple IDENTIFY message out in order to toggle the disconnect/reconnect permission bit. This may be used to selectively enable or disable disconnect/reconnect permission during portion of a command. Note that this function does not effect the operation of the Forced Disconnect function.
- MESSAGE REJECT during Target Disconnection
  - If the Initiator rejects the SAVE DATA POINTER message, the file will disable disconnect/reconnect permission. This is equivalent to receiving an IDENTIFY message with bit 6 equal to zero. This will cause to file to inhibit the pending disconnection.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

– If the initiator rejects the DISCONNECT message, the file will not disconnect but may attempt to disconnect at a later time. This function may be used to selectively disable disconnection during portions of a command.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

3.3 Attention Condition

The attention condition allows an initiator to inform the file that a MESSAGE OUT phase is desired. The initiator may create the attention condition by asserting the ATN signal at any time except during the ARBITRATION or BUS FREE phases.

The initiator must create the attention condition by asserting the ATN signal least two deskew delays before releasing ACK for the last byte transferred in a bus phase to guarantee that the attention condition will be honoured before transition to a new bus phase. This will guarantee a predictable file response to message received during the MESSAGE OUT phase for this attention condition. If the ATN signal is asserted later, it might be honoured in the current bus phase or the next bus phase and then may not result in the expected action.

After the initiator asserts the ATN signal, the file will respond with the MESSAGE OUT phase as follows:

Current Phase	Response
COMMAND	Message Out phase will occur after part or all of the Command Descriptor Block has been transferred to the file. The initiator must continue REQ/ACK handshakes during the Command phase until the file enters the MESSAGE OUT phase.
DATA	The MESSAGE OUT phase will occur after part or all of the data bytes have been transferred and not necessarily on a logical block boundary. The initiator must continue REQ/ACK handshakes (asynchronous transfer) until it detects the phase change. **Note:** In synchronous transfer, the initiator must continue sending ACK pulses to reach an offset of zero.
STATUS	The MESSAGE OUT phase will occur after the REQ/ACK handshake of the status byte has been completed.
MESSAGE IN	The MESSAGE OUT phase will occur before the file sends another message.
SELECTION	If ATN occurs during a SELECTION phase and before the initiator releases the BSY signal, the MESSAGE OUT phase will occur immediately after that SELECTION phase.
RESELECTION	The MESSAGE OUT phase will occur after the file has sent its IDENTIFY message for that RESELECTION phase. (First the file tries to complete the reselection.)

The initiator must keep the ATN signal asserted if more than one message byte is to be transferred during the MESSAGE Out phase. The file will process each message byte (multiple-bytes for an extended message) prior to receive the next message from the initiator. The file will continue to handshake and process byte(s) in the MESSAGE OUT phase until ATN goes false unless one of the following condition occurs:

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

1. The file receives an illegal or inappropriate message and goes to the MESSAGE IN phase to send a MESSAGE REJECT message.
2. The file detects a catastrophic error condition and goes to the BUS FREE phase.

3.4 SCSI Bus Related Error Handling Protocol

This protocol is used to handle error that threaten the integrity of a connection between the Target and an Initiator.

3.4.1 Unexpected BUS FREE Phase Error Condition

There are several error conditions that will cause the file to immediately change to the BUS FREE phase, regardless of the state of the ATN signal. The file will not attempt to reconnect to the initiator to complete the operation that was in progress when the error condition was detected. The initiator should interpret this as a catastrophic error condition.

If the LUN was identified by the file prior to the error condition, then the file will abort the active command for this initiator/LUN and generate sense data for this initiator/LUN to describe the cause of the catastrophic error. The initiator may retrieve this sense data by issuing a REQUEST SENSE command to this LUN. Note however, that the REQUEST SENSE command may fail if the catastrophic error condition persists.

If the LUN was not identified by the file prior to the error condition, then the file will not affect the sense data or the operation of any currently executing command for this initiator or any other initiator.

3.4.2 MESSAGE OUT Phase Parity Error

The file will, depending on the model[6], optionally retry the message phase and if it still fails abort the current command with CHECK CONDITION status and sense data of ABORTED COMMAND / SCSI PARITY ERROR .

3.4.3 MESSAGE IN Phase Parity Error (Message Parity Error)

The file will, depending on the model[6], optionally retry the message phase and if it still fails abort the current command, go to bus free, setting sense data of ABORTED COMMAND / SCSI PARITY ERROR .

3.4.4 COMMAND Phase Parity Error

The file will, depending on the model[6], optionally retry the command phase after sending a restore pointers message. If it still fails it will abort the current command with CHECK CONDITION status and sense data of ABORTED COMMAND / SCSI PARITY ERROR .

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

3.4.5 DATA OUT Phase Parity Error

If the file detects a parity error during DATA OUT phase, the file will abort the current command with CHECK CONDITION status and sense data of ABORTED COMMAND / SCSI PARITY ERROR.

[6] The action for a particular file model is defined in the interface spec addendum for each model.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

3.4.6 INITIATOR DETECTED ERROR Message

An INITIATOR DETECTED ERROR message is valid after a COMMAND, DATA IN/OUT or STATUS phase has occurred. If another phase has occurred, the message is rejected.

The file will, depending on the model, optionally retry the previous phase is it is command or status. If this fails or the previous phase was a data transfer the file will generate a CHECK CONDITION status and a Sense key of ABORTED COMMAND with additional sense code of INITIATOR DETECTED ERROR.

3.4.7 MESSAGE REJECT Message

The file will take the following actions after receiving the MESSAGE REJECT message in response to messages listed below.

DISCONNECT The file will not disconnect but remains connected.

COMMAND COMPLETE No error, continue to bus free.

IDENTIFY Command aborted - bus freed - Sense data set to MESSAGE REJECT ERROR.

LINKED CMD CMPLT Command aborted - link broken - bus freed - sense data set to MESSAGE REJECT ERROR.

MESSAGE REJECT Command aborted - STATUS phase executed with CHECK CONDITION - sense data set to MESSAGE REJECT ERROR.

RESTORE POINTERS Command aborted - status set to CHECK CONDITION - sense will be set with the error that caused the RESTORE POINTERS message to be issued. (Assuming that error recovery is in progress)

SAVE DATA POINTER The file will not disconnect from the SCSI bus. It will not be considered an error.

No previous Msg The command is aborted, the bus freed, and Sense data is set to MESSAGE REJECT ERROR. This occurs when the file has not sent a message, but gets a MESSAGE REJECT from the initiator.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

4.0 Additional Information

This chapter provides additional information or descriptions of various functions, features, or operating models supported by the Target that are not fully described in previous chapters.

4.1 SCSI Protocol

There are various operating conditions that prevent the Target from executing a SCSI command. This section describes each of these operating conditions and their relative priority.

4.1.1 Priority of SCSI Status Byte Reporting

After establishing the I_T_L nexus or I_T_L_Q nexus, the Target must first determine whether command execution is allowed. Execution is deferred until a later time if the command must be added to the command queue. Execution may also be prevented by an internal Target condition that requires the reporting of a Check Condition, Busy or Reservation Conflict Status. There are several different internal conditions to be active at the same time. The order in which the Target checks for each of these conditions determines their priority(highest priority first) as follows:

1. Check Condition status for invalid Logical Unit Number(see 4.1.2, “Invalid LUN in Identify Message” on page 81)
2. Check Condition status for Incorrect Initiator Connection(see 4.1.3, “Incorrect Initiator Connection” on page 81)
3. Busy Status or Queue Full Status, or add command to command queue(see 4.1.4, “Command Processing During Execution of Active I/O process” on page 82)
4. Check Condition status for Unit Attention condition(see 4.1.5, “Unit Attention Condition” on page 84)
5. Check Condition status during Start-up and Format operations(see 4.1.6, “Command Processing During Start-up and Format Operations” on page 85)
6. Check Condition status for Internal Error Condition(see 4.1.7, “Internal Error Condition” on page 85)
7. Check Condition status for Deferred Error Condition(see 4.1.8, “Deferred error” on page 86)
8. Check Condition status for Degraded Mode Condition(see 4.1.9, “Degraded Mode” on page 86)
9. Reservation Conflict status(see 4.1.11, “Command Processing While Reserved” on page 88)
10. Check Condition status for invalid command opcode
11. Check Condition status for invalid command descriptor block

The highest priority internal condition that prevents command execution is reported by the Target, provided there is no bus error.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

For all Check Conditions, Sense data is built by the target provided a valid LUN address is known. Sense data is cleared by the Target upon receipt of any subsequent command to the LUN from the initiator receiving the Check Condition.

4.1.2 Invalid LUN in Identify Message

There are three different circumstances defined within the SCSI protocol when the response to an invalid LUN will occur. Each of these result in a different response.

4.1.2.1 Case 1 - Selection message sequence with Inquiry command

The INQUIRY command is a special case in SCSI. It is used to configure the bus when file IDs and LUNs are not known. The proper response is to return the inquiry data with a peripheral drive type of 1Fh which indicates that the specified LUN is not supported.

4.1.2.2 Case 2 - Selection message sequence with any other command

Any other commands, except REQUEST SENSE, return CHECK CONDITION status when an invalid LUN is specified in the message sequence following selection. In response to a REQUEST SENSE command the target shall return sense data. The sense key shall be set to ILLEGAL REQUEST and the additional sense code shall be set to LOGICAL UNIT NOT SUPPORTED.

4.1.2.3 Case 3 - After selection message sequence

It is permissible for the initiator to issue multiple IDENTIFY messages during a single command sequence provided to LUN remains the same. If the LUN is altered, the file goes to a Bus Free Phase.

4.1.3 Incorrect Initiator Connection

It is an Incorrect Initiator Connection error if any of the following occurs:

- an Initiator attempts to establish an I_T_L nexus when an I/O process (either queued or active) with an I_T_L nexus already exists from a previous connection with the same initiator.
- an Initiator attempts to establish an I_T_L_Q nexus when an I_T_L nexus already exists from a previous connection with the same initiator.
- an Initiator attempts to establish an I_T_L nexus when an I_T_L_Q nexus already exists from a previous connection with the same initiator.

  Note: It is not an Incorrect Initiator Connection to send a Request Sense command without a Queue tag message when sense is pending on the logical unit for the Initiator that issues the Request Sense command.
- an Initiator attempts to establish an I_T_L_Q nexus when an I/O process (either queued or active) with an I_T_L_Q nexus already exists from a previous connection with the same initiator.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

If any of the above errors occur, all queued I/O processes and active I/O processes associated with the issuing Initiator on the specified logical unit are terminated. The current I/O process is ended with a CHECK CONDITION status, the sense key is set to ABORTED COMMAND and the additional sense code is set to OVERLAPPED COMMANDS ATTEMPTED. Status is only returned for the current I/O process.

4.1.4 Command Processing During Execution of Active I/O process

When the Target is not executing any active I/O processes, a new I/O process is permitted to execute (unless execution is prevented by another internal Target condition listed in 4.1.1, “Priority of SCSI Status Byte Reporting” on page 80).

If an active I/O process does exist when the Target receives a new command, then the Target determines if:

- Check Condition Status with Sense Key = Aborted Command is returned for an Overlapped Commands Attempted error
- the command is permitted to execute
- the command is added to the command queue
- Queue Full Status is returned
- Busy Status is returned

If an active I/O process does exist when the Target receives a new command, then the Target determines how the new command should be handled based on the following rules:

- Check Condition Status is returned with Sense Key set to Aborted Command for an Overlapped Commands Attempted error if:
  - See 4.1.3, “Incorrect Initiator Connection” on page 81
- the command is permitted to execute if
  - the command is an Inquiry or Request Sense command
- Check Condition Status is returned with Sense Key set to Logical Unit Not Ready if:
  - the start-up operation or format operation is an active process.
- the command is permitted to execute if
  - the conditions to execute concurrently are met. (See 4.5, “Concurrent I/O Process” on page 95)
- the command is added to the command queue for an I_T_L nexus if:
  - no Queue Tag message was received during the connection which established the I/O process,and
  - Untagged Queuing is enabled(UQE = 1),and
  - disconnection is allowed for the current I/O process, and
  - there is no queued I/O process or active I/O process corresponding to the I_T_L nexus for the current I/O process, and

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

  - the command is not linked to a previous command.

- the command is added to the command queue for an I_T_L_Q nexus if:
  - a Queue Tag message was received during the connection which established the I/O process, and
  - Tagged Queuing is enabled(DQue = 0), and
  - an I/O process(either active or queued) exists at the Target for this Initiator, and
  - disconnection is allowed for the current I/O process, and
  - there is no queued I/O process or active I/O process corresponding to the I_T_L_Q nexus for the current I/O process, and
  - the command is not linked to a previous command.

  Note: Both Tagged and Untagged Queuing must be enabled(DQue = 0 and UQE = 1) for the Target to the queue tagged I/O processes from multiple Initiators.

- Queue Full Status is returned if:
  - the command would otherwise be queued(according to the rules described above)but the command queue is full and all slots are utilized, or
  - the command would otherwise be queued(according to the rules described above)but all of the available command queue slots not reserved for use by another initiator are utilized, or
  - Tagged Queuing is enabled(DQue = 0) and a Format Unit command was previously queued but has not yet begun execution, or
  - Tagged Queuing is enabled(DQue = 0) and a Start Unit command was previously queued but has not yet begun execution.

- Busy Status is returned if:
  - Tagged Queuing is disabled(DQue = 1) and a Format Unit command was previously queued but has not yet begun execution, or
  - Tagged Queuing is disabled(DQue = 1) and a Start Unit command was previously queued but has not yet begun execution, or
  - the command would otherwise be queued(according to the rules described above)but disconnection is not allowed for the current I/O process, or
  - the command would otherwise be queued(according to the rules described above)but Untagged Queuing is disabled(UQE = 0) and an I/O process (either active or queued) exists at the Target from a different Initiator.

If a command is queued, command execution may still be prevented at a later time when the command is dequeued to become an active I/O process. This occurs if command execution is prevented by another internal Target condition listed in 4.1.1, “Priority of SCSI Status Byte Reporting” on page 80 at the time the command is dequeued.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

4.1.5 Unit Attention Condition

The file will generate a unit attention condition for each initiator whenever:

- The file has been reset.
  This includes Power On Reset, SCSI Bus Reset, SCSI BUS DEVICE RESET message.
- The mode parameters in effect for this initiator has been changed by another initiator.
- The microcode has been changed.
  WRITE BUFFER command has been executed to download microcode. In this case, a unit attention condition is generated for all initiators except the one that issued the command.
- Commands are cleared by another initiator.
  This condition is generated against the initiator that has queued commands, if ...
  - Clear Queue Message is received.
  - Contingent Allegiance Condition is cleared when QERR (in Mode Page 0A) is 1.
  - DQue is set to 1 while queued command exist.

The unit attention condition persists for each initiator until that initiator clears the condition as described in the following paragraphs.

If the file receives a command from each initiator before reporting a CHECK CONDITION status for a pending unit attention condition for that initiator , the file's response varies with the command as follows.

INQUIRY The file executes the command with GOOD status and preserves the unit attention condition.

REQUEST SENSE

If the file has an available pending sense data for the initiator, the file sends the pending sense data and preserves the unit attention condition for the initiator.

If the file does not have an available pending sense data for the initiator, the file sends sense data for the unit attention condition and clear the unit attention condition for the initiator.

ALL OTHER The file terminates the command with a CHECK CONDITION status and preserve the unit attention condition.

If the file receives a command form each initiator after reporting a CHECK CONDITION status for a pending unit attention condition for that initiator , the file's response varies with the command as follows.

REQUEST SENSE The file sends the sense data for a pending unit attention condition and returns GOOD status. And the file clear the unit attention condition for the initiator.

ALL OTHER The file execute the command with GOOD status and clear the unit attention condition unless another unit attention condition exists. And then the sense data for the unit attention condition is lost.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

4.1.6 Command Processing During Start-up and Format Operations

If the Target receives a command from an Initiator while the Target is executing a start-up or format operation, The Target's response varies with the command as follows:

INQUIRY The file sends a inquiry data and returns appropriate status.

REQUEST SENSE Execute the command, returns a Sense key of NOT READY and an Additional Sense Code of LOGICAL UNIT NOT READY and return GOOD STATUS.

The Additional Sense Code Qualifier that is returned dependent on type of I/O processes that are active:

For the START/STOP UNIT and the Auto-start operation, the qualifier returned is LOGICAL UNIT IS IN PROCESS OF BECOMING READY. For the FORMAT UNIT command, the qualifier returned is LOGICAL UNIT NOT READY,FORMAT IN PROGRESS, and the Sense key specific bytes are set to return the progress indication.

START/STOP UNIT If Untagged Queuing is enabled(UQE = 1), and the start-up operation is an active process and a Start/Stop Unit command(either active or queued) does not exist at the Target from this initiator, and disconnection is allowed for the current I/O process then: The command is added to the command queue.

Otherwise: Do not execute the command and return Check Condition Status. The Sense data generated is described in Request Sense above.

ALL OTHER The file terminates the command with CHECK CONDITION status. The Sense data generated is described in Request Sense above.

4.1.7 Internal Error Condition

The Target generates an Internal Error condition for all Initiators when:

- an internally initiated operations with an unrecoverable error. the following is a list of internally initiated error conditions:
  - During the execution of the start-up sequence for Auto Start after the SCSI bus has been enabled and prior to completion of the bring-up sequence.
  - Following a SCSI H/W reset or a SCSI Bus Device Reset message if the reset was received during a start-up sequence with the Auto Start function enabled. The start-up sequence is executed if it has not been previously executed and completed.
- an recoverable error occurs during an internal Target idle time function

An Internal Error condition causes Sense data to be generated and saved for all Initiators. The Error Code field of the Sense is set for a Current Error(70h) and the Sense Key is set to HARDWARE ERROR. Recovered errors are not reported. Any outstanding Deferred Error condition is cleared for all initiators and the associated Sense data is lost.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

The Internal Error condition persists for each Initiator until that Initiator clears the condition from the logical unit as described below. Several commands are handled as special cases during an Internal Error condition. These cases are also discussed.

If the Target receives a command from an Initiator while an Internal Error condition exists for that Initiator, the Target's response varies with the command as follows:

INQUIRY The file executes the command with GOOD status and do not clear the Internal Error condition.

REQUEST SENSE The file executes the command, return the sense data generated by the Internal Error condition, return Good Status, and clear the Internal Error condition for that Initiator.

ALL OTHER The file terminates the command with a CHECK CONDITION status and clear the Internal Error condition.

4.1.8 Deferred error

Error code (71h) of sense data indicates that the Check Condition status returned is the result of an error or exception condition that occurred during execution of a previous command for which Good status has already been returned.
The drive creates an Deferred Error condition when :

- Execution of a Start/Stop Unit command with the immediate bit of one ends with an error.
- Execution of a Format Unit command with the immediate bit of one ends with an error.
- Execution of a Write command with WCE (write cache enable) bit of one ends with an error.

4.1.9 Degraded Mode

There are certain errors or conditions which may impair the file's ability to function normally. Rather than fail hard, the file is designed to be as responsive as possible. Also, in most cases, some action on the part of the initiator may be used to restore normal operation. This mode of limited operation is called Degraded Mode.

4.1.10 Degrade mode handling

If any of degrade mode condition occurs, Hardware error condition will be created. An initiator can receive the sense key of Hardware error (4h). After clearing the sense key, the drive accepts commands according to the status of the degrade mode.

The following table shows the degrade mode status, acceptable command and additional sense code. If the degrade mode exists, the Hard ware error condition caused by the degrade mode will be recreated any time after POR, SCSI Reset or Bus Device Reset message. Following list shows the various operation mode.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

4.1.10.1 Degrade Mode Entry Condition/Consequence State

Degrade-Mode	Entry Reason	Accepting Request	Description and sense code
Power on Self Test fail	• Failure of a Send Diagnostic self-test • Failure of a start-up sequence	• Request Sense • Inquiry • Start/Stop unit • Write Buffer(except download and save)	The integrity of the drive is questionable. Sense code = 4080 (diagnostic fail), Sense code = 4081 (HDC), 4082 (HIC), 4083 (other) 4084 (RAM error)
Spin-up	Can not start up motor	• Request Sense • Inquiry • Start/Stop unit • Write Buffer(except download and save)	Return Spin-up Degrade Mode. Sense code = 0400
U-code	Can not read u-code	Same as Spin-up Degrade	Return U-code Degrade Mode. Sense code = 4085 (u-code download fail)

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

Degrade-Mode	Entry Reason	Accepting Request	Description and sense code
Reserved Area	• Fail to read Push Table • Fail to read saved Mode Parameter • Fail to read Defect list	All commands.	Reserved area sector valid check failed. Defective sector found in reserved area. (Hard Error) The integrity of the drive is questionable. Sense code = 4080 (diagnostic fail), 1900 (Defect list error),
Config	Fail to read Drive Configuration Table	All commands	Configuration sector valid check failed. The integrity of the drive is questionable. May return blanks in Inquiry Command. Sense code = 4C00 (Self configuration fail)

4.1.11 Command Processing While Reserved

A logical unit is reserved after successful execution of the Reserve command. Each time a Reserve command is executed successfully, the Target records the SCSI ID of the Initiator that made the reservation and the SCSI ID of the Initiator that is to receive the reservation. This information is needed to determine whether subsequent commands should be permitted or if the Reservation Conflict Status should be reported. The Initiator that made the reservation is the Initiator that issued the Reserve command. The Initiator to receive the reservation may be either the same or a different Initiator(third-party reservation).

If the logical unit is reserved when a new command is received, the Target examines the command opcode and the SCSI ID of the issuing Initiator to determine whether a Reservation Conflict Status should be returned based on the following rules:

1. If the issuing Initiator is the one that made the reservation and also the one to receive the reservation then:
   - All commands are permitted.
2. If the issuing Initiator is neither the one that made the reservation nor the one to receive the reservation then:
   - A Request Sense or Inquiry command is permitted.
   - A Release command is permitted but is ignored.
   - Any other command results in a Reservation Conflict Status.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

3. If the issuing Initiator is the one that made the reservation but is not the one to receive the reservation then:
   - An Inquiry, Request Sense, Reserve, or Release command is permitted.
   - Any other command results in a Reservation Conflict Status.
4. If the issuing Initiator is not the one that made the reservation but is the one to receive the reservation then:
   - An Reserve command results in a Reservation Conflict Status.
   - A Release command is permitted but is ignored.
   - Any other command is permitted.

If a Reservation Conflict Status is not reported and the command is permitted, then the Target checks the next highest priority internal condition to determine whether execution is allowed. See 4.1.1, “Priority of SCSI Status Byte Reporting” on page 80

4.2 Priority Commands

Certain SCSI commands always execute without returning a Busy Status, Reservation Conflict Status in response to the command. These commands are:

- Inquiry
- Request Sense

These commands do not disconnect from the SCSI bus prior to completion. They are executed prior to attempting to complete the execution of any other pending command that has disconnected from the SCSI bus. Therefore, a second priority command cannot be received during the execution of a priority command.

These commands are never queued whether or not the command is sent with a queue tag. However, the rule for an Incorrect Initiator Connection still apply to priority commands.(see 4.1.3, “Incorrect Initiator Connection” on page 81)

4.3 Command queuing

When the initiator specifies that the file shall disable command queuing, the initiator must send only untagged command. When the initiator specifies that the target shall enable command queuing, the initiator may send either tagged or untagged command, but shall not use both at the same time.

The following commands are never queued and will be immediately executed without Bus disconnection

- Priority Commands(i.e.: Request Sense and Inquiry)

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

- Commands linked to previous commands. These are defined to be part of a single I/O process. (Linked commands are always executed immediately following the previous command from the same initiator. No other Initiator's command are allowed to be executed between two linked commands.)
- Commands for which disconnection is not allowed.(These may result in a Busy Status.)
- Commands in which a SCSI bus error occurred between selection and first disconnection following the receipt of the CDB.
- Commands for an invalid LUN.
- Commands which cause an OVERLAPPED COMMANDS ATTEMPTED error(see 4.1.3, “Incorrect Initiator Connection” on page 81)

4.3.1 Queue depth

The file has 32 queue elements. Seven of these are reserved for the seven possible initiator's that may connect to the file. This leaves a pool of 25 queue elements which are shared by all initiators on a first come first served basis. Any initiator can queue at least one command at any time irrespective of the actions of any other initiators in the system. A single initiator may queue up to 26 commands, if no other initiator has more than one command in the queue, although at times this maximum may be reduced as the file can reserve command blocks for internal use.

4.3.2 Tagged queuing

Commands with a tag message are saved in the command queue. Queued commands will be reordered in the method of 'Single side elevator'. See the section on Reordering 4.4, “Command reordering” on page 93 for details.

4.3.3 Untagged queuing

When Untagged Queuing is enabled, the target supports queuing one I/O process from each initiator. If the target receives an untagged I/O process while executing an I/O process from a different initiator, the untagged I/O process may be queued.

Untagged I/O process are treated by the target as though they were received with Simple Queue Tag messages for purposes of queuing.

Note: There is no guarantee that I/O processes are executed in the order they were received in a multiple initiator environment when Untagged Queuing is enabled.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

4.3.4 Command queuing rule

Commands can be received during an active I/O process if Bus is free. If CPE (concurrent process enable) bit of Mode page 0 is 0, only a single command except 2 commands (Inquiry and Request Sense) can be executed at the same time. In that case, a command starts to be executed after the previous command has completed. If CPE (concurrent process enable) bit of Mode page 0 is 1, I/O process of Read(6), Read extend(10), Write(6) and Write extend(10) can be active at the same time. See 4.5, “Concurrent I/O Process” on page 95 for details.

When the target receives a new command, the target do the following things.

1. Check if the command is overlapped. A target that detects an Incorrect Initiator Connection shall abort all (queued and active) I/O processes for the initiator and shall return Check Condition status. The sense key shall be set to Abort Command (0Bh) and the additional sense code shall be set to Overlapped Commands Attempted (4E00h).

   An incorrect initiator connection occurs when an initiator:

   a. attempts to establish an I_T_L nexus when an I_T_L nexus already exists from a previous connection.
      (For example, an initiator sends an untagged command when an untagged command is queued or being currently executed for the initiator.) or,
   b. attempts to establish an I_T_L_Q nexus when an I_T_L nexus already exists from a previous connection.
      (For example, an initiator sends a tagged command when an untagged command is queued or being currently executed for the initiator.) or,
   c. attempts to establish an I_T_L nexus when an I_T_L_Q nexus already exists from a previous connection.
      (For example, an initiator sends an untagged command when a tagged command is queued or being currently executed for the initiator.) or,
   d. attempts to establish an I_T_L_Q nexus when the same I_T_L_Q nexus already exists from a previous connection.
      (For example, an initiator sends a tagged command when a tagged command with the same Queue Tag is queued or being currently executed for the initiator.)

   It is not an Incorrect Initiator Connection to reconnect to an already established I_T_L or I_T_L_Q nexus if the initiator sends an Abort, Abort Tag, Clear Queue, Bus Device Reset during the same Message Out phase as the Identify message.

2. Check if the command is valid.

   a. Check if the command code is valid. If the command code is invalid, the target sends Check Condition and sets the sense key Illegal request (05h) and the additional sense code Invalid command (2000h).
   b. Check if LINK = 0 and FLAG = 1. If LINK = 0 and FLAG = 1 then the target sends Check Condition and sets the sense key Illegal request (05h) and the additional sense code Invalid CDB (2400h).
   c. Check if parity error occurred on CDB receipt. If parity error occurred, the target sends Restore pointer message (03h) to retry. If parity error occurred again, the target sends Check

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

Condition and sets the sense key Aborted Command (0Bh) and the additional sense code SCSI Parity (4700h).

d. Check if the reserved bits are off. If any reserved bit in CDB is on, the target sends Check Condition and sets the sense key Illegal Request (05h) and the additional sense code Invalid CDB (2400h).

3. Check if the command must be immediately executed. If the command is one of either Inquiry or Request Sense sent as untagged and unlinked command, it must be executed immediately without disconnecting Bus.
4. Check if the queue is full. If the command queue is full, the target sends Queue Full Status (28h).
5. Check if Busy. Busy Status (08h) is returned if:
   - disconnection is not allowed for the incoming tagged command.
   - disconnection is not allowed when queued or active process exists.
6. If all above is satisfied, add the command into the queue.

4.3.5 Queue Full status

This status is returned when a Simple Queue tag, Ordered Queue tag or Head of Queue tag message is received and the command queue is full. The I/O process is not placed in the command queue. Since one queue element is reserved for each initiator, any untagged command that doesn't cause Incorrect Initiator Connection will not cause Queue Full status.

4.3.6 Device behaviour on Command queuing

1. Initiators must send a Queue tag immediately after the Identify message in Message Out phase just after Selection. Targets send a simple queue tag immediately after the Identify message in Message In phase just after Reselection.
2. Each initiator can issue either one of tagged command or untagged command exclusively at the same time. There can exist other initiator using the other.
3. When DQue (Disable queue) of mode page 0Ah is 1, if an initiator issues a tagged command, the drive returns "Message Reject" message (07h) and receives that command as an untagged command.
4. Queue Tag number doesn't affect the order to execute.
5. An initiator issues a command with a queue tag which is same as the current I/O process or queued I/O process, the target returns Incorrect Initiator connection.
6. A series of linked commands are a single I/O process, and are assigned the queue tag established in the initial selection. A command received with a Head-of-Queue tag message shall not suspend a series of linked commands for which the target has begun execution.
7. If DQue is changed to 1 while queued commands exist, all queued commands for the ITL nexus issuing the mode select will be aborted. All future commands received, from any initiator, with a

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

queue tag will be processed as untagged commands, with a message reject message being returned immediately after the qtag is received by the target.

4.4 Command reordering

Command reordering function is supported under tagged command queuing enabled (DQue = 0). Reorder feature reorders Read/Write commands in order to minimize seek time between commands. This function will improve total thruput of the drive.

4.4.1 Reorder method

The reordering algorithm of the drive is 'Single side elevator'. The queued commands are reordered as ascending order of the requesting LBA from the LBA of the current command. Commands of smaller LBA will be queued after the commands of greater LBA.

For example, suppose that a Read command requesting LBA 1000 is being executed. And the commands queued are ...

- Read of LBA 100.
- Read of LBA 400.
- Read of LBA 1200.
- Read of LBA 1400.

These commands must be reordered ...

1. Read of LBA 1200.
2. Read of LBA 1400.
3. Read of LBA 100.
4. Read of LBA 400.

While executing the current command, if the incoming commands are Read of LBA 1500,Read of LBA 120, and Read of LBA 35, after the commands are received, the new queue becomes ...

1. Read of LBA 1200.
2. Read of LBA 1400.
3. Read of LBA 1500.
4. Read of LBA 35.
5. Read of LBA 100.
6. Read of LBA 120.
7. Read of LBA 400.

Reordering is done each time a command is received. The order of the commands previously received, reordered and queued will not be changed. Reordering is done only by inserting a incoming command into the best place in the command queue.

4.4.2 The restriction of reordering

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

4.4.2.1 SCSI restriction

Tag message: The initiator can send an untagged command or 3 kinds of tagged message to define reordering manner.

- Untagged command.
  The initiator does not care the reordering feature. An untagged command might be passed by a command sent by another initiator. Untagged command will not create data integrity problem by the same initiator, because the initiator cannot send more than one untagged command at the same time.
- Simple Queue Tag message (20h).
  This tagged command can be reordered by the drive unless it violates the rule of the Head-of-queue tag and the Ordered-queue tag message.
- Head of Queue Tag message (21h).
  Commands with this tag should be inserted into the head of the tag queue. While a command is being executed, this tagged command will be inserted to the head of queue to be executed after the command being currently executed. The previous command being executed will not be terminated by this tagged command. If the plural head-of-queue tagged commands are received, those commands will be executed in LIFO (Last in First out) order.
- Ordered Queue Tag message (22h).
  This tagged command is executed in the order received. All commands received before this command should be executed before this command, and all commands received after this commands should be executed after this command.

Mode parameter (Control Mode page 0Ah): Queue Algorithm modifier (7-4 bit of byte 3) specifies restriction on the algorithm used for reordering commands that are tagged with Simple-queue tag message.

0h : Restricted reordering. The target shall order the actual execution sequence of the queued commands from each initiator such that data integrity is maintained for that initiator.

1h : Un-restricted reordering allowed. The target may reorder the actual execution sequence of the queued commands in any manner it selects. Any data integrity exposures related to the command sequence order are explicitly handled by the initiator.

2h-Fh : Reserved.

4.4.2.2 Implementation restriction

There are some restrictions that are not clearly stated in SCSI specifications, but they are implemented in the drive.

1. Priority commands. Untagged and unlinked Request sense and Inquiry
   These are executed immediately after receiving CDB without disconnecting SCSI bus. They do not follow the rule of reordering. The execution of these command will not affect data integrity.
2. Commands other than Read/Write.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

Commands such as Mode Select, Reserve, Release or Start Stop Unit may affect the condition of execution or data of commands that follows. If Queue algorithm modifier is 0 (Restricted reordering), all commands including these commands except Read/Write and priority commands (untagged and unlinked Request sense and Inquiry) will not be reordered even if untagged or Simple Queue tagged. In that case, these commands will be always treated as Ordered tagged command.

3. Concurrent I/O.
   Commands with Head of Queue tag or Ordered Queue tag will not be executed concurrently. Only untagged and Simple-queue tagged Read/Write command can be executed concurrently.

4.4.3 Reordering general rules

1. Reordering follows SCSI restrictions and implementation restrictions.
2. If queue algorithm modifier = 1 (Unrestricted reordering), untagged and Simple-queue tagged Read and Write commands will be reordered in 'Single side elevator' method of LBA, regardless of breaking data integrity. Non Read/Write commands will be passed by a Read/Write command unless it violates tag message restrictions.
3. If queue algorithm modifier = 0 (Restricted reordering), untagged and Simple-queue tagged Read and Write commands will be reordered in 'Single side elevator' method of LBA, as far as it maintains data integrity. It implies that a Read command whose requested LBAs overlap those of a Write command previously received will not pass the Write command. A Write command will be the same. Non Read/Write commands will not be passed by any Read/Write or non Read/Write command.

4.5 Concurrent I/O Process

The Concurrent I/O process is that plural I/O processes are active (not queued) on the same logical unit at the same time. The target may starts the data phase of an I/O process while another I/O process is not completed. The following I/O processes are allowed to executed concurrently.

- Unlinked and untagged Request Sense and Inquiry during execution of other commands.
- When CPE (Concurrent process enable) bit is 1, one of the following commands can be executed during another one or the same one of the following commands is being executed, if those are untagged or simple tagged commands.
  - Read(6), Read extend(10)
  - Write(6), Write extend(10)

When an I/O process ends in Check Condition Status, the drive enters the Contingent Allegiance Condition and other queued I/O processes from all initiators on the same logical unit will not reconnect and will not complete the execution until the sense data is cleared. See 4.17, “Contingent allegiance Condition” on page 107 for details. If an I/O process (P-1) encounters an error while another I/O process (P-2) is active, the drive returns Check Condition to P-1 and P-2. The drive may continue P-2 until its convenient point to suspend, but may not send a Status. After the initiator clears the Contin-

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

gent Allegiance condition, the drive will resume or terminate P-2 according to QErr bit of Control mode page.
If the drive gets an error of P-2 before suspending the execution of P-2, it will keep the sense data separately from the sense data for P-1. The sense data for P-2 will be set after the Contingent Allegiance condition caused by P-1 is cleared.

4.6 Back to Back Write

Back to Back Write allows plural write commands requesting sequential LBA to write without losing a motor revolution.

The target will write back to back under the following conditions:

- Two Write(6) or Write extend(10) commands executed consecutively.
- The write commands address consecutive LBA. The LBA specified by the 2nd write command is the next to the last LBA by the 1st write command.
- The data of the LBA for the 2nd write command is received before the last LBA of the 1st write command is written to the media.
- The CPE (Concurrent process enable) bit is 1 or the WCE (Write Cache enable) bit is 1.

If the CPE (Concurrent process enable) is 1 and the WCE (Write cache enable) is 0, the Data phase of the 2nd write command is started after the Data phase of the 1st write command but before the Status phase of it. After seek complete, the data of both 1st and 2nd write command will be written onto the media. Then the drive reconnects to the nexus of the 1st write command to send Status before sending Status to the 2nd write command.

If the CPE (Concurrent process enable) is 0 and the WCE (Write cache enable) is 1, the drive returns Good Status immediately after the Data phase of the 1st write command but before actually writing the data to the media. The I/O process of the 2nd write command begins after the drive sends Good Status to the 1st write command. And then Good Status is returned to the 2nd write command. After seek complete, the data of both 1st and 2nd write command will be written onto the media.

If the drive returns Check Condition to the 1st write command and enters the contingent allegiance condition, execution of the 2nd write command is suspended. The 2nd write command will resumed execution when the contingent allegiance condition is cleared.

4.7 Write Cache

If the WCE (Write cache enable) bit is 1, the drive returns Good Status and Command Complete message and goes to Bus Free immediately after receiving the data of the last sector before actually writing the data onto the media. The drive will accept and queue a command, but it can not start to execute the command after sending a Good Status except the following condition.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

- The incoming command is one of priority commands.
- The CPE (Concurrent process enable) is 1 and the incoming command is Read(6), Read extend(10), Write(6) Write extend(10).

The drive behaviour according to the mode parameter.

When Write Cache is enabled and Concurrent I/O process is enabled,

the drive may start and complete following Read/Write commands before the actual write operation (writing on the media) is completed. So, any following Read/Write commands or priority commands returning a Good Status. do not guarantee the completion of the Write command.

Under the current implementation, a command except Read/Write commands or priority commands returning a Good Status following a Write command can guarantee that the data is written to the media.

A Synchronize Cache command always performs this function regardless of the current implementation.

When Write Cache is enabled and Concurrent I/O process is disabled,

the drive may not start any following commands except priority commands before a previous Write command is completed and the drive sends a Status. So, under the current implementation, any following commands except priority commands returning a Good Status guarantees that the data is written to the media.

A Synchronize Cache command always performs this function regardless of the current implementation.

When Write Cache is disabled and Concurrent I/O process is enabled,

the drive may start and complete following Read/Write commands before a previous Write command is completed. But the Write command will not be completed before the data is written to the media.

A Synchronize Cache command is not needed in order to assume that the data is written to the media.

When Write Cache is disabled and Concurrent I/O process is disabled,

the drive may not start any following commands except priority commands before a previous Write command is completed and the drive sends a Status. And the Write command will not be completed before the data is written to the media.

A Synchronize Cache command is not needed in order to assume that the data is written to the media.

If the drive detects an error after it returns a Good Status, the drive sets a Differed Error (Error Code of sense data = 71h) and a following command will be returned with Check Condition and the Contingent allegiance condition is established. Under the Contingent allegiance condition, all queued processes including commands from other initiators are suspended.

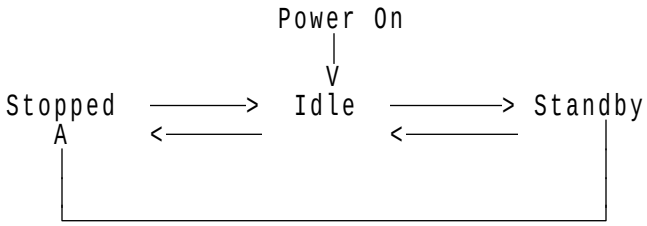
4.8 Power Saving Mode

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

4.8.1 Overview

Power save function will save power consumption while the drive is idle. The drive automatically transfers its operating mode according to the event and timer.

4.8.1.1 Mode Transition



Mode Characteristics

Mode Name	Idle	Standby	Stopped
HDC	On	Off	Off
SRV	On	Off	Off
R/W Channel	Off	Off	Off
Content of sector buffer (Write Cache, Look Ahead & etc.)	Kept	Lost	Lost
Spindle motor	On	Off	Off
Status	Good	Good	Not Ready

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

Trigger of Mode Transition

To From	Idle	Standby	Stopped
Idle	---	Expiration of Standby Timer & Read/Write Idle	SCSI Stop Unit Command
Standby	Any SCSI Command	---	SCSI Stop Unit Command
Stopped	SCSI Start Unit Command	N/A	---

Action for Mode Transition

To From	Idle	Standby	Stopped
Idle	---	Act_Lock Channel_off VCM_off Stop_spindle	Act_Lock Channel_off VCM_off Stop_spindle
Standby	Channel_on VCM_on HDC_on Start_spindle	---	Not Ready Status
Stopped	Channel_on VCM_on HDC_on Start_spindle	N/A	---

4.9 Automatic Rewrite/Reallocate

The target supports Auto and Recommended Reallocate for READ, WRITE, WRITE and VERIFY, VERIFY.

Automatic and Recommend Reallocate operate from within the read/write command. When an automatic reallocation occurs, the read or write command takes longer to complete. During this time, the target disconnects from the SCSI bus, if allowed, and reconnects before ending the command.

Following is a description of the target behaviour for each setting of ARRE. ARRE setting effects all data errors.(No Sector Found, Data Sync Byte Errors and Data ECC Errors.)

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

ARRE=1 : An error site determined to need rewriting or reassignment during a read is automatically rewritten or reallocated at the conclusion of the read and prior to sending the status. If the site cannot be automatically rewritten or reallocated, then a recommendation for reassignment is given. The site will be automatically rewritten or reallocated only if the data has been successfully read.

ARRE=0 : An error site determined to need rewriting or reassignment during a read is recommended for rewriting or reassignment at the conclusion of the read.

The setting of the ARRE bit is checked and the target will automatically rewrite/reallocate or recommend rewrite/reassign for the following commands.

- Read(6)
- Read(10)

Target will recommend rewrite/reallocate but will not auto rewrite/ reallocate for the following commands.

- Verify
- Verify Portion of Write and Verify

For all other commands the ARRE setting is ignored and the target will not automatically rewrite/reallocate or recommend rewrite/reassign.

Following is a description of the target behaviour for each setting of AWRE. AWRE setting effects only No Sector Found Errors on writes.

AWRE=1 : An error site determined to need reassignment during a write is automatically reallocated at the conclusion of the write and prior to sending the status. If the site cannot be automatically reallocated, then a recommendation for reassignment is given. The site will be automatically reallocated only if the write recovery succeeded at the conclusion of the write.

AWRE=0 : An error site determined to need reassignment during a write is recommended for reassignment at the conclusion of the write.

The setting of the AWRE bit is checked and the target will automatically reallocate or recommend reassign for the following commands.

- Write(6)
- Write(10)
- Write portion of Write and Verify

For all other commands the AWRE setting is ignored and the target will not automatically reallocate or recommend reassign.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

Auto/Recommend Reallocate information is communicated via the sense data returned following a command during which a site was determined to need rewriting or reassignment. The LBA returned in the sense data is the LBA that determined to need rewriting or reassignment.

The sense data combinations with auto/recommend rewrite/reallocate are listed below.

Key	Code	Qual	Description
1	17	00	Recovered Data without ECC.
1	17	06	Recovered Data without ECC - Auto Reallocated.
1	17	07	Recovered Data without ECC - Recommend Reassign.
1	17	09	Recovered Data without ECC - Data Rewritten.
1	18	00	Recovered Data with ECC.
1	18	02	Recovered Data with ECC - Auto Reallocated.
1	18	05	Recovered Data with ECC - Recommend Reassign.
1	18	07	Recovered Data with ECC - Data Rewritten.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

4.10 Segmented Caching

4.10.1 Overview

Segmented Cashing divides the data buffer into several smaller buffers. Each buffer is used as Read/Write/Read-Ahead buffer.

The size of segmented buffer is controlled by number of segmented buffer field of read cache page (page 08h). But file may adapt size of segmented buffer by checking access pattern.

4.10.2 Read Ahead

The Read Ahead function consists of reading data that the Initiator has not yet requested to the file buffer. This function is intended to improve performance for an initiator that frequently accesses sequential data with successive SCSI read commands. The Read Ahead function works when RCD (the read cache disable) bit of read cache page (page 08h) is set to 0.

The file will continue to read subsequent logical blocks following the rules below.

1. At least the same number of sectors succeeding the sectors requested by the last Read command will be read automatically if they are not available on the sector buffer.
2. The drive has at least 2 segmented buffers. Data stored by the most recent Read ahead function will not be discarded by Write command.
3. Data in the segmented buffers will not be replaced unless it is against the rule 1.
4. The Read Ahead operations continue across all physical boundaries (such as tracks and cylinders).

Commands that affects the Read ahead function.

The drive initiates the Read ahead function

- RCD is 0, and
- Read(6), Read extended(10), Read Verify, and Write and Verify is received, and
- The consecutive LBA of the requested LBA is not available on the buffer.

The action of each command for previously started Read ahead function is listed below. If SCSI reset or bus device reset message is received, all contents of segmented buffer is flushed.

Code	Command	Action
00h	Test Unit Ready	No Effect

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

Code	Command	Action
01h	Rezero Unit	Abort Read Ahead
03h	Request Sense	No Effect
04h	Format Unit	Abort Read ahead and flush all segments
07h	Reassign Blocks	Abort Read ahead and flush all segments
08h	Read(6)	Abort Read ahead if data not in active segment
0Ah	Write(6)	Abort Read ahead and flush the LRU segment
0Bh	Seek(6)	Abort Read Ahead
12h	Inquiry	No Effect
15h	Mode Select(6)	Abort Read ahead and flush all segments
16h	Reserve	No Effect
17h	Release	No Effect
1Ah	Mode Sense(6)	Abort Read ahead and flush all segments
1Bh	Start/Stop Unit	Abort Read ahead and flush all segments
1Dh	Send Diagnostic	Abort Read ahead and flush all segments
25h	Read Capacity	No Effect
28h	Read extended(10)	Abort Read ahead if data not in active segment
2Ah	Write extended(10)	Abort Read ahead and flush the LRU segment

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

Code	Command	Action
2Bh	Seek extended(10)	Abort Read Ahead
2Eh	Write and Verify	Abort Read ahead and flush the LRU segment
2Fh	Verify	Abort Read ahead and flush the LRU segment
34h	Pre-Fetch	Abort Read ahead if data not in active segment
35h	Synchronize Cache	Abort Read ahead and flush all segments
37h	Read Defect Data	Abort Read ahead and flush all segments
3Bh	Write Buffer	Abort Read ahead and flush all segments
3Ch	Read Buffer	Abort Read Ahead
3Eh	Read Long	Abort Read ahead and flush the LRU segment
3Fh	Write Long	Abort Read ahead and flush all segments
4Dh	Log sense	Abort Read ahead and flush all segments

Table 1. Read Ahead handling per each command

Even if an error occurs during Read ahead, the error will not be reported to the Initiator. The data read before the error occurs will be stored as a valid data by Read ahead.

4.11 Reselection Timeout

If reselection fails, depending on the file model, it will be retried 1 or more times. Please see the model specific specifications for the number of retries allowed.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

4.12 Single Initiator Selection

For single initiator systems, it is not an error to have only the target ID bit present during selection. Disconnection is not allowed for Single Initiator Selection with only one ID bit present during selection. The initiator must not send an Identify message with the disconnect permission bit(6) on.

4.13 Non-arbitrating systems

The Target cannot detect whether other SCSI devices on the SCSI bus use arbitration prior to selection. As a consequence, the Target allows disconnect permission to be enabled by the Identify message independent of the initiators use of arbitration prior to selection. A non-arbitrating initiator must ensure that disconnect permission in the Identify message is disabled (bit 6=0)for proper operation.

4.14 Selection without ATN

If the target is selected without ATN signal active, no Identify message is received from the initiator. In this case, the LUN is identified from the CDB and disconnect permission is disabled. The target does not perform any phase retries. The target still responds to a subsequent attention condition. However, the LUN is not considered to be known if a fatal error is detected during the Command phase. That is a Command phase parity error or a fatal message error in response to attention condition during Command phase is handled as a Bus Free error with no sense data. The target also knows the use of linked commands if selected without ATN. The target does not initiate synchronous data transfer negotiation if selected without ATN.

Phase retries and target initiated negotiations may be allowed if a subsequent Identify message is received.

4.15 Multiple Initiator Environment

4.15.1 Initiator Sense Data

Separate sense data is reserved for each initiator. Each initiator's sense data is maintained independent of commands from other initiators.

4.15.2 Initiator Mode Select/Mode Sense Parameters

A single shared copy of the Mode Select/Mode Sense parameters is maintained by the file. This includes both the current and saved parameters.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

4.15.3 Initiator Data Transfer Mode Parameter

A separate data transfer mode parameters area is reserved and maintained for each initiator.

4.16 Options Jumper Block

The Drive provides an Options Jumper Block which allows the drive to be customized using jumpers instead of software settings. Refer to the product Hardware Specifications for detailed information on the location and function of these jumpers.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

4.17 Contingent allegiance Condition

The contingent allegiance condition shall exist following the return of Check Condition, except Check Condition caused by Invalid LUN. Execution of all queued commands shall be suspended until the contingent allegiance condition is cleared.

The contingent allegiance condition can be cleared by the initiator in one of the following ways:

- By issuing a REQUEST SENSE command to the Target and receiving the sense data. This is most recommended way.
- By issuing any other command to the I_T_x nexus that reported the fault.
- By issuing an Abort message to the I_T_x nexus that reported the fault. This will also abort the current and queued I/O process from that initiator.
- By issuing a Bus Device Reset message to the Target. This will also abort all current and queued I/O processes.
- By generating a RESET condition on the bus. This MUST be the last resort.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

4.18 Reset

The Reset condition is used to clear all SCSI devices from the bus. This condition takes precedence over all other phases and conditions. After a reset condition is detected and the reset actions completed, the target returns to a 'SCSI bus enabled' state that allows the target to accept SCSI commands.

This device uses the Hard reset option as defined in the SCSI-2 standard.

4.18.1 Reset Sources

There are four sources of resets detected by the target:

Reset Name	Reset Source
Power-On Reset	The is the signal generated by the hardware at initial power-on
Self-Initiated reset	This is a software-generated reset that occurs when a catastrophic error is detected by the microcode.
SCSI Bus Reset	This is a reset generated when the SCSI bus control line RST goes active.
SCSI Bus Device Reset Message	This is the reset generated by the SCSI Bus Device Reset Message(0Ch).

4.18.2 Reset Actions

The action taken by the Drive following a reset is dependent on the source of the reset.

4.18.2.1 Power-On reset and Self-Initiated reset

These two reset conditions cause the following to be performed in the order shown.

1. A power-up sequence
2. A start-up sequence is necessary to put the Drive in a ready state

4.18.2.2 SCSI Bus reset and SCSI Bus Device Reset message

These two reset conditions cause the following to be performed.

- If reset goes active while the power-up sequence is in progress, the power-up sequence is started over.
- If the Auto Start pin is grounded and a start-up sequence has not yet completed, a start-up sequence will be re-attempted from the beginning.

  Note: The power-up sequence, having already completed, is not rerun.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

- If reset occurs while a physical sector is being written, the write operation is disabled after the current physical sector is written. Data is not lost as long as power stays valid until the physical sector being written is completed.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

4.19 RAS Characteristics

The file will execute a self test at power on or when a Send Diagnostics command is issued with the self test bit set.

The diagnostics are to assure the correct operation of the file and to verify that the check circuits detect fault conditions.

4.19.1 Power on Diagnostics

At power on time the following tests are executed:

1. Test the microprocessor's:
    a. Internal Timers.
    b. Internal RAM.
2. Do a sum check on the microprocessor's external ROM.
3. Test the adapter section of the file as follows:
    a. Test the registers.
    b. Test HDC.
4. Do a read/write test on the microprocessor's external RAM.
5. Do a read/write test on sector buffer RAM.
6. Check if the 12 volt line is active.
7. Initialize and check the servo system is functioning correctly.
8. Check the spindle's RPM.
9. Seek to a correct cylinder.

4.19.2 Diagnostics Command

The tests executed as a result of the Send Diagnostics command with the self test bit set to a one differs from the tests executed at power on. The spindle motor must be at the correct speed for the Diagnostics command to be executed. If the motor is not at the correct speed, a NOT READY status will be returned in the sense byte.

The Diagnostics command will execute the following tests:

1. Test the adapter section of the file as follows:
    a. Test the registers.
    b. Assure the ECC circuits work correctly.
    c. Do a read/write test on sector buffer RAM.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

2. Check the spindle's RPM.
3. Seek to a correct cylinder.
4. Write the CE cylinder.
5. Read from the CE cylinder.

4.19.3 Diagnostics Fault Reporting

Faults detected before successful completion of the adapter section could prevent the file from responding to a selection.

Faults detected after the successful completion of the adapter section will be reported as Check Condition status to the initiator on the first command issued after a fault is detected except for the Inquiry command. The Inquiry command will always respond with good status. Detecting a fault during power on will not terminate execution of the diagnostics nor will it terminate the power on process.

Faults detected during a Send Diagnostics command will report a Check Condition as end status.

4.20 Idle Time Function

The execution of a function by the drive during idle times may result in delays of commands requested by SCSI initiators while idle time functions are in progress. 'Idle time' is defined as time spent by the drive not executing a command requested by a SCSI initiator. The type of a function performed during idle time is

- Random Seek Idle

4.20.1 Random Seek Idle

The Random Seek Idle function is built into the drive microcode to prolong disk life. It is designed to move the heads over the disk surface during idle periods to prevent disk lubrication migration problems.

If the drive has been idle for about 30 min, the internal microcode moves the heads to the randomly selected location for 1 second.

The execution time that a SCSI command could be delayed is typically less than 1 second.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

5.0 SCSI SENSE DATA

5.1 SCSI Sense Data Format

Format of the sense data returned by the file in response to the REQUEST SENSE command.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

	BIT 7	6	5	4	3	2	1	0
BYTE 0	Valid	Error Code (70h or 71h)						
BYTE 1	RSVD = 0							
BYTE 2	0			ILI	0	Sense Key		
BYTE 3 — 6	(MSB) Information Bytes (LSB)							
BYTE 7	Additional Sense Length							
BYTE 8 —11	(MSB) Product Specific Information (LSB)							
BYTE 12	Additional Sense Code							
BYTE 13	Additional Sense Code Qualifier							
BYTE 14	FRU = 0							
BYTE 15	SKSV	Sense—Key Specific Bits						
BYTE 16 BYTE 17	Sense—Key Specific Bytes							
BYTE 18 —19	Reserved = 0							
BYTE 20 BYTE 21	Unit Error Code							
BYTE 22 BYTE 23	Reserved = 0							
BYTE 24 —27	Product Specific Information							
BYTE 28 —31	Reserved = 0							

Figure 51. Format of Sense Data. Format of the Sense Data returned by the file in response to the REQUEST SENSE command

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

5.1.1 Sense Data Description

5.1.1.1 Valid Bit

Bit 7 of byte 0

0 The Information Bytes are not defined.
1 The Information Bytes contain a valid logical block address.

5.1.1.2 Error Code

Bit 6 - 0 of byte 0.

70h Current Error. This indicates an error for the current command.
71h Deferred Error. This indicates that the error is for a previous command that has already returned a GOOD status. Such commands are associated with the immediate bit, or write caching. FORMAT UNIT command is an example of a command that may return a deferred error.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

5.1.1.3 Sense Key

Bit 3 - 0 of byte 2.

The sense key provides generic categories in which error and exception conditions can be reported. Initiators would typically use sense keys for high level error recovery procedures.

0 NO SENSE
There is no sense key information to be reported for the logical unit.

1 RECOVERED ERROR
The last command completed successfully with some recovery action performed by the file. More detailed information is available in the Additional Sense Code.

2 NOT READY
The logical unit addressed cannot be addressed. More detailed information is available in the Additional Sense Code.

3 Medium Error
The command terminated with an unrecoverable error condition caused by a flaw in the media or an error in the recorded data. More detailed information is contained in the Additional Sense Code.

4 HARDWARE ERROR
The file detected a unrecoverable hardware error while performing a command or during a diagnostic test. More detailed information is contained in the Additional Sense Code.

5 ILLEGAL REQUEST
There was an illegal parameter in the command descriptor block or additional parameter supplied as data. If an invalid parameter is found in the CDB, then the command is terminated without altering the medium. If an invalid parameter is found in parameters supplied as data, then the file might have altered the medium.

6 UNIT ATTENTION
Indicates that the file entered in the 'Unit Attention Condition'. (See 4.1.5, “Unit Attention Condition” on page 84.)

7-8 Not used

9 Vendor Specific

A Not used

B ABORTED COMMAND
The file aborted the command.

C-D Not Implemented

E Not Used

F Reserved

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

5.1.1.4 Information Bytes

Byte 3 - 6
This field is only valid when Valid Bit is one, and contains the unsigned LBA associated with the sense key.

5.1.1.5 Additional Sense Length

Byte 7
Indicates the remaining number of bytes in the sense data. (It is set to 18h in the file.)

5.1.1.6 Command Specific Information

Byte 8 - 11

The values in this field vary by product. Please see the product specific specification for more details.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

5.1.1.7 Additional Sense Code/Qualifier

Byte 12 / 13

Key	Code	Qual	Description
0	00	00	No error.
1	01	00	Recovered write error no index
1	03	00	Recovered write error - write fault
1	14	01	Recovered write error Id not found
1	16	00	Recovered write error DAM not found
1	17	00	Recovered read error without ECC applied.
1	17	01	Recovered read error with retries.
1	17	06	Recovered read error without ECC applied. Auto reallocated.
1	17	07	Recovered read error without ECC applied. Recommended reassign. This value can be returned only when ARRE = 0.
1	17	09	Recovered read error without ECC applied. Data re-written. This value can be returned only when ARRE = 1.
1	18	00	Recovered read error with ECC applied. This value can be returned only when ARRE = 0.
1	18	02	Recovered read error with ECC applied. Auto reallocated. This value can be returned only when ARRE = 1.
1	18	05	Recovered read error with ECC applied. Recommended reassign. This value can be returned only when ARRE = 0.
1	18	07	Recovered read error with ECC applied. Data Rewritten. This value can be returned only when ARRE = 1.
1	1C	01	Primary Defect list Not Found. Requested Defect List Format is not supported. Default Lsi Format is returned.(Read Defect Data Only)
1	1C	02	Grown Defect list Not Found. Requested Defect List Format is not supported. Default Lsi Format is returned.(Read Defect Data Only)
1	44	00	Internal target failure
2	04	00	Not ready. Start spindle motor fail.
2	04	01	Not ready. In process of becoming ready.
2	04	02	Not ready. Initializing command required. (Start Unit)
2	04	04	Not ready. Format in progress.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

Key	Code	Qual	Description
2	31	00	Not ready. Media format corrupt. A format operation was interrupted (power down, reset) prior to completion of a Format Unit command. The Format Unit command should be re-issued and must complete successfully for this error condition to be removed.
2	40	80	Diag Fail - Bring-Up Fail
2	40	85	Diag Fail - RAM Microcode Not Loaded
2	4C	00	Degraded Mode - Self Configuration Fail Configuration/RAM Microcode not loaded
3	10	00	Medium error. ID CRC error.
3	11	00	Medium error. Unrecovered read error.
3	14	01	Medium error. Record not found.
3	16	00	Medium error. Data synchronization mark error. (DAM error)
3	19	00	Medium error. Defect list error. A defect list error occurs when a data error is detected while reading the manufacturing defect list or while reading or writing the grown defect list.
3	31	01	Medium error. Medium Format Corrupted Reassign Failed
4	01	00	H/W error. No index or sector.
4	02	00	H/W error. No seek complete.
4	03	00	H/W error. Write fault.
4	09	00	H/W error. Track following error.
4	11	00	H/W error. Unrecovered read error in reserved area.
4	31	00	Degrade mode. Format corrupt.
4	32	00	H/W error. No defect spare location available. A no defect spare location available sense code indicates that the Reassign Block command can not proceed the process because all spare sectors have been used, or it will exceed implementation limitation of defect handling of the file.
4	40	80	Degrade Mode. Diagnostic Fail. Configuration sector valid check fail. Reserved area sector valid check fail.
4	40	81	Degrade mode. HDC error.
4	40	82	Degrade mode. HIC error.
4	40	83	Degrade mode. Other LSI error.
4	40	84	Degrade mode. RAM error.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

Key	Code	Qual	Description
4	40	85	Degrade Mode. RAM Microcode Not Loaded
4	44	00	H/W error. Internal target failure
4	47	00	H/W error. SCSI parity error.
5	1A	00	Illegal request. Parameter list length error. The number of parameters supplied is not equal to the value the expected.
5	20	00	Illegal request. Illegal command operation code. This command is also returned when an unsupported command code is received.
5	21	00	Illegal request. Logical block address out of range.
5	24	00	Illegal request. Invalid field in CDB.
5	26	00	Illegal request. Invalid fields in the parameter list.
6	28	00	Unit attention. Not ready to ready transition.(Format completed)
6	29	00	Unit attention. Power on reset or Bus device reset occurred.
6	2A	01	Unit attention. Mode select parameter changed.
6	2F	00	Unit attention. Command cleared by another initiator.
6	3F	01	Unit attention. Micro code has been changed.
B	1B	00	Aborted command. Synchronous data transfer error. (Extra ack detected)
B	25	00	Aborted command. Unsupported LUN. The drive supports LUN 0 only.
B	43	00	Aborted command. Message reject error. A message reject error occurs when an inappropriate or unexpected message reject is received from the initiator or the initiator rejects a message twice.
B	45	00	Aborted command. Selection/Reselection failed. A selection/reselection error occurs when the initiator fails to respond to a reselection within 250 milliseconds after the drive starts reselection. The reselection is attempted a second time before setting selection/reselection failed sense code.
B	47	00	Aborted command. SCSI parity error.
B	48	00	Aborted command. Initiator detected error message received. An initiator detected error occurs when the initiator detects an error, sends a message to retry, detects the error again, and sends the retry message a second time. The drive then sets check condition status with Initiator Detected Error.
B	49	00	Aborted command. Inappropriate/illegal message. An inappropriate or illegal message occurs when the initiator sent a message that either is not supported or is not in a logical sequence.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

Key	Code	Qual	Description
B	4E	00	Aborted command. Overlapped commands attempted.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

5.1.1.8 FRU (Field Replaceable Unit)

Byte 14
The FRU (Field Replaceable Unit) field value will always be zero.

5.1.1.9 Sense Key Specific

Byte 15 - 17
The definition of this field is determined by the value of the sense key field.

Illegal Request Case: Error filed pointer is returned.

	BIT 7	6	5	4	3	2	1	0
BYTE 15	SKSV	C/D	Reserved		BPV	Bit Pointer		
BYTE 16	(MSB)	Field Pointer						
BYTE 17								(LSB)

Figure 52. Field Pointer Bytes

SKSV Sense-key specific valid

C/D Command/Data

0 Indicates that the illegal parameter is in the data parameters sent by the initiator during DATA OUT phase

1 Indicates that the illegal parameter in the command descriptor block.

BPV Bit Pointer Valid

0 Indicates the bit pointer field is not valid.

1 Indicates the bit pointer field is significant.

Bit Pointer Bit Pointer indicates which bit of the byte number reported in Field Pointer is the bit in error. When a multiple-bit field is in error, the pointer point to the most significant bit of the field.

Field Pointer Indicates which bytes of the command descriptor block or of the parameter data was in error. Bytes are numbered starting from zero, as shown in the tables describing the commands and parameters. When a multiple-byte field id in error, the pointer point to the most significant byte of the field.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

Recovered, Hardware or Medium Error Case: Actual Retry Count is reported.

	BIT 7	6	5	4	3	2	1	0
BYTE 15	SKSV	Reserved						
BYTE 16	(MSB)	Actual Retry Count						
BYTE 17								(LSB)

Figure 53. Actual Retry Count Bytes

SKSV Sense-key specific valid

0 Indicates that Actual Retry Count is not valid.

1 Indicates that Actual Retry Count is valid.

Actual Retry Count Actual number of retries used in attempting to recover from the error condition.

Not Ready Case: Progress indication is returned. These fields are only defined for the FORMAT UNIT command with the Immediate bit set to one.

	BIT 7	6	5	4	3	2	1	0
BYTE 15	SKSV	Reserved						
BYTE 16	(MSB)	Progress Indication						
BYTE 17								(LSB)

Figure 54. Format Progress Indication Bytes

SKSV Sense-key specific valid

0 Indicates that Progress Indication is not valid.

1 Indicates that Progress Indication is valid.

Progress Indication Indicates a percent complete in which the returned value is the numerator that has 10000h as its denominator.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

5.1.1.10 Reserved

Byte 18 - 19

5.1.1.11 Unit Error Code(UEC)

Byte 20 - 21
The UEC gives detailed information about the error. It contains a unique code which describes where the error was detected and which piece of hardware or microcode detected the error. The UEC code is only valid if POR related-failure occurs.

UEC Code	Description
0001h	ROM error
0002h	Internal RAM error
0004h	HDC Register Error
0008h	HDC Abort function error
0010h	HDC Sequencer error
0020h	HDC ECC error
0040h	Servo Register failure
0080h	Sector Buffer error
0100h	PICO error
0200h	Servo MPU error
0400h	Motor error

Figure 55. UEC code definition

5.1.1.12 Reserved

Byte 22 - 23

5.1.1.13 Physical Error Record

Byte 24 - 27

The values in this field vary by product. Please see the product specific specification for more details.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

5.1.1.14 Reserved

Byte 28 - 31

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

6.0 Recovery Procedure

The following sections describe the recovery procedure for each of the various types of errors for which recovery is attempted. The error recovery procedure depends on the file model. Please see the model specific specifications for the description.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

Index

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

M

N

O

P

Q

R

S

T

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.

This document is the property of IBM. Its use is authorized only for responding to a request for quotation or for the performance of work for IBM. All questions must be referred to the IBM purchasing department.