

9914 ENHANCED SCSI INTERFACE

USER MANUAL

Document Ref. 95 124768

Issue 1

Parts List Number 95 124768 10

© Copyright 1992 Storage Technology Corporation - All Rights Reserved

Storage Technology reserve the right to change this manual without notice

This page has no technical content

Sales Offices

USA

Storage Technology Corporation
2270 South 88th Street
Louisville
Colorado 80028-0237
USA

Tel: 303 673 3126
Fax: 303 673 2559

UK

Storage Technology Corporation
81 High Street
Esher
Surrey
KT10 9QA
England

Tel: 372 468188
Fax: 372 463162

Germany

Storage Technology Corporation
Berner Strasse 35
D-6000 Frankfurt 50
Germany

Tel: 069/5001 295
Fax: 069/5001 293

France

Storage Technology Corporation
BP613
78056 Saint-Quentin-en-Yvelines
Cedex
France

Tel: 33-1-3012-3681
Fax: 33-1-3012-3500

INTERNAL CODE LICENSE

PLEASE READ THIS NOTICE CAREFULLY. THIS NOTICE IS A LEGAL AGREEMENT BETWEEN YOU (EITHER AN INDIVIDUAL OR ENTITY), THE ENDUSER, AND STORAGE TECHNOLOGY CORPORATION ("STORAGETEK") THE MANUFACTURER OF THE EQUIPMENT. BY USING ANY UNIT OF EQUIPMENT DESCRIBED IN THIS DOCUMENT, YOU AGREE TO BECOME BOUND BY THE TERMS OF THIS AGREEMENT. IF YOU DO NOT AGREE WITH TERMS OF THIS AGREEMENT, DO NOT USE THE EQUIPMENT. IF YOU DO NOT HAVE THE AUTHORITY TO BIND YOUR COMPANY, DO NOT USE THE EQUIPMENT. IF YOU HAVE ANY QUESTION, CONTACT THE AUTHORIZED STORAGETEK DISTRIBUTOR OR RESELLER FROM WHO YOU ACQUIRED THIS EQUIPMENT.

- (a) Licensed Internal Code ("Code") is a set of instructions which direct, order or control device operation, that is either embedded into or is to be loaded into Equipment, is an integral part of, and is required by, Equipment and executes below the external user interface. The Equipment you have acquired by purchase or lease is manufactured by StorageTek and contains Code. By accepting and operating this Equipment you acknowledge that StorageTek or its licensor(s) owns all copies of Code, including all copies made from them, and that Code is copyrighted by StorageTek or its licensor(s).

- (b) StorageTek grants you, as an enduser of the Equipment, a license to use the Code (or any replacement provided by StorageTek or your authorized StorageTek distributor or reseller) which license authorizes you :
 - (i) to execute the Code to enable the Equipment to perform in accordance with StorageTek's official published specifications (Specifications);

 - (ii) to make a backup or archival copy of the Code (unless one is made available to you by StorageTek or your authorized StorageTek distributor or reseller, and provided you reproduce the copyright notice and any other legend of ownership on the copy), which copy you may use only to replace the original in the Equipment; and

 - (iii) to execute and display the Code as necessary to maintain the Equipment.

- (c) Your license is limited to use of the Code as set forth in (b) above. You may not, for example, do any of the following :
 - (i) otherwise copy display, transfer, adapt, modify or distribute the Code (electronically or otherwise), except as may be authorized in the Specifications or by StorageTek or your authorized StorageTek distributor or reseller in writing;

 - (ii) reverse assemble, reverse compile, decode, reverse engineer or translate the Code; or

 - (iii) sub-license, assign or lease the Code or any copy of it.

- (d) You may transfer possession of the Code to another party only with the transfer of the Equipment and any party receiving Code from you receives such bound by the terms and conditions of this License, and accepts these terms by using the Code. It is your responsibility to inform the party acquiring the Equipment of that party's limited right to the Code as set forth in this license. Your license to use the Code is discontinued when you are no longer an owner or a rightful possessor of the Equipment. In addition, the terms and conditions of this license apply to Code you acquire from another party.

- (e) This license does not convey to you any license to or title to any StorageTek diagnostic software programs, or any copy of such, even if a copy of such diagnostic software is supplied with a unit of Equipment, sent to you and/or included on the same media as Code, all for the convenience of StorageTek or your authorised distributor or reseller customer services personnel, all rights in and to such diagnostic software being reserved by StorageTek.

Table of Contents

1. Introduction	
1.1 Identification	1 - 1
2. Physical Interface	
2.1 Mechanical	2 - 1
2.2 Single-ended Signals	2 - 1
2.2.1 Inputs	2 - 1
2.2.2 Outputs	2 - 1
2.2.3 Single-ended Pin Assignments	2 - 2
2.3 Differential Signals	2 - 3
2.3.1 Signal Levels	2 - 3
2.3.2 Differential Pin Assignments	2 - 3
2.4 Interface Cables	2 - 4
3. SCSI Protocol	
3.1 Message Protocol	3 - 2
3.2 Command Set	3 - 6
4. Command Descriptions	
4.1 Test Unit Ready	4 - 1
4.2 Rewind	4 - 2
4.3 Request Sense	4 - 3
4.4 Read Block Limits	4 - 8
4.5 Read Forward	4 - 9
4.6 Write Data	4 - 15
4.6.1 Variable Mode Unbuffered	4 - 15
4.6.2 Fixed Mode Unbuffered	4 - 16
4.6.3 Write Buffered Mode Variable/Fixed	4 - 17
4.6.4 Write Long Block	4 - 18
4.7 Read Reverse	4 - 20
4.8 Write Filemarks	4 - 21
4.9 Space	4 - 23
4.10 Inquiry	4 - 26
4.11 Verify	4 - 31
4.11.1 Variable Mode	4 - 31
4.11.2 Fixed Mode	4 - 32
4.11.3 Variable or Fixed Mode	4 - 33
4.12 Recover Buffered Data	4 - 34
4.13 Mode Select	4 - 35
4.13 Reserve/Release LUN	4 - 40
4.15 Copy	4 - 41
4.15.1 Truncated Segments, PAD/CAT Operation	4 - 44
4.15.2 Typical Copy Operation	4 - 44
4.15.3 Error Conditions	4 - 45

4.16	Erase	4 - 47
4.17	Mode Sense	4 - 48
4.18	Unload	4 - 54
4.19	Receive Diagnostic Results	4 - 55
4.20	Send Diagnostic	4 - 59
4.21	Locate	4 - 61
4.22	Write Buffer	4 - 63
4.23	Read Buffer	4 - 65
4.24	Read Position	4 - 67
4.25	Long Block Read	4 - 70
4.26	Log Sense	4 - 73
	4.26.1 Using 'Log Sense'	4 - 74
4.27	Log Select	4 - 75
	4.27.1 Using 'Log Select'	4 - 76
4.28	Limited Long Block Write	4 - 77

5. Completion Status

A. Set Up

A.1	Dual-in-Line Switch Settings	A - 1
	A.1.1 PROM 124610	A - 2

B. Interface Termination

C. Streamer Manuals

D. Customer Specific Variants

E. Diagnostic Program CDBs & Parameter Lists

E.1	Non-Page Formats	E - 1
E.2	Page Code 00 Format	E - 1
	E.2.1 Parameter List for Page Code 00	E - 1
E.3	Page Code 80 Formats	E - 2
	E.3.1 Parameter List for Page Code 80	E - 2
E.4	Page Code 81 Formats	E - 3
	E.4.1 Parameter List for Page Code 81	E - 4
E.5	Page Code 82 Formats	E - 5
	E.5.1 Parameter List for Page Code 82	E - 5

User Manual Amendment Record

Revision	Details	Date
ISSUE 1	First circulated revision Derived from publication 95 121798 24	Jun 1992

1.

Introduction

The Super SCSI version of the Streamer Buffered Interface (SBI) is a microprocessor-based controller designed to interface host computer systems, via the SCSI bus, to Industry Standard, Pertec Micro-formatter compatible, half-inch magnetic tape units. As with its predecessors, the SCSI SBI has been designed to fit within the spare slot in the 9914 card cage.

The Super SCSI SBI presents itself to the SCSI bus as a TARGET DEVICE. An Emulex FAS216/226 is utilized as the SCSI control element, providing DISCONNECT/RESELECT, and full ARBITRATION capabilities. Both SINGLE ENDED and DIFFERENTIAL interface configurations are available.

The SBI implements the bulk of the SCSI MESSAGE and COMMAND protocol, including the RECEIVE/SEND DIAGNOSTIC commands based on the powerful ON-LINE DIAGNOSTIC capability of the *StorageTek* 9914 Streamer.

The main features of the Super SCSI SBI are

- Embeds within the 9914
- Single-ended / differential capability
- Up to 7 Mbytes/second asynchronous data rate
- Up to 5 Mbytes/second normal synchronous data rate
- Up to 10 Mbytes/second fast synchronous data rate
- Full arbitration
- Disconnect/reselect supported
- Comprehensive command/message set
- 1 Mbyte or 2 Mbyte data buffer
- External tape parity generation/checking.

1.1 Identification

This manual applies to *StorageTek* Super SCSI interface boards, other SCSI boards (eg designed for AS/400 application) are covered by separate publications.

All interfaces are plug-in printed circuit boards (12.1 by 7.7 in) and identified by the batch label, which is marked with the parts list (PL) number as in the following table.

	1 Mbyte	2 Mbyte
Dual Interface	95 124600	95 124601
Single-ended	95 124602	95 124604
Differential	95 124603	95 124605

This page has no technical content

2.

Physical Interface

2.1 Mechanical

The Super SCSI SBI board has 2 standard connector positions, one position supporting SINGLE ENDED, the other DIFFERENTIAL, with pin-outs as set out in tables later in this section. A 3M-type 50-way male connector is usually fitted into one, to communicate with the SCSI bus.

In the case of the dual interface, male connectors are fitted into both board positions.

StorageTek offer 3M-type bus connectors, unshielded (4650-6000) or shielded (3565-2002, or Sun-style 8350-8005), with a box or plate arrangement, allowing streamer/SCSI combinations to be easily altered according to customers' preferred connector type.

2.2 Single-ended Signals

The user should be aware that ANSI do not recommend using the Fast SCSI synchronous method of data transfer with single-ended signals.

2.2.1 Inputs

Characteristic signal levels (as defined by ANSI) are shown in the following table

Logic State	True/False	Voltage at the SBI	TTL equivalent
Logical 1	True	0.0 to 0.8 V	low
Logical 0	False	2.0 to 5.25 V	high

2.2.2 Outputs

Single-ended outputs are a mixture of 3-state and open collector drivers, all capable of sinking 48 mA at 0.5 V, with characteristic levels as shown in the following table.

Logic State	True/False	Voltage at the SBI	TTL equivalent
Assertion	True	0.0 to 0.4 V	low
Negation	False	2.5 to 5.25 V	high

2.2.3 Single-ended Pin Assignments

Table 1.2.1 shows the SCSI signal names against their connector pin numbers, the full signal names and detailed descriptions of their functions are set out in the ANSI specification.

Signal name	Pin number	Pin number	Signal name
GRND	1	2	-DB 0
GRND	3	4	-DB 1
GRND	5	6	-DB 2
GRND	7	8	-DB 3
GRND	9	10	-DB 4
GRND	11	12	-DB 5
GRND	13	14	-DB 6
GRND	15	16	-DB 7
GRND	17	18	-DB P
GRND	19	20	GRND
GRND	21	22	GRND
GRND	23	24	GRND
n.c.	25	26	TERM PWR
GRND	27	28	GRND
GRND	29	30	GRND
GRND	31	32	-ATN
GRND	33	34	GRND
GRND	35	36	-BSY
GRND	37	38	-ACK
GRND	39	40	-RST
GRND	41	42	-MSG
GRND	43	44	-SEL
GRND	45	46	-C/D
GRND	47	48	-REQ
GRND	49	50	-I/O

TABLE 2.2.3 SINGLE-ENDED PIN ASSIGNMENTS

Notes:

- i) Appendix B explains TERM PWR.
- ii) n.c. = no connection.

2.3 Differential Signals

2.3.1 Signal Levels

All signals consist of 2 lines denoted +S and -S. A signal is true when +S is more positive than -S (with a minimum difference of 1 V). The interface has been implemented using TI 75176 or equivalent differential driver/receivers.

2.3.2 Differential Pin Assignments

Signal name	Pin number	Pin number	Signal name
GRND	1	2	GRND
+DB 0	3	4	-DB 0
+DB 1	5	6	-DB 1
+DB 2	7	8	-DB 2
+DB 3	9	10	-DB 3
+DB 4	11	12	-DB 4
+DB 5	13	14	-DB 5
+DB 6	15	16	-DB 6
+DB 7	17	18	-DB 7
+DB P	19	20	-DB P
DIFFSENS	21	22	GRND
GRND	23	24	GRND
TERM PWR	25	26	TERM PWR
GRND	27	28	GRND
+ATN	29	30	-ATN
GRND	31	32	GRND
+BSY	33	34	-BSY
+ACK	35	36	-ACK
+RST	37	38	-RST
+MSG	39	40	-MSG
+SEL	41	42	-SEL
+C/D	43	44	-C/D
+REQ	45	46	-REQ
+I/O	47	48	-I/O
GRND	49	50	GRND

TABLE 2.3.2 DIFFERENTIAL PIN ASSIGNMENTS

Note:

- i) Appendix B explains TERM PWR and DIFFSENS.

2.4 Interface Cables

Suitable cables, for shielded applications and to meet some RFI requirements will need a compatible connector and conductive shell. The actual cable should be selected after consulting the ANSI specification X3T9 2/86-109 rev 10h, particularly if the fast synchronous method of data transfer is to be used. Some of the ANSI requirements are stated next.

Cable type	25-signal twisted pair (recommended), <i>or</i> 50-way flat
Maximum cumulative length	25 metres (differential) 6 metres (single-ended)

TABLE 2.4.1 GENERAL CABLE REQUIREMENTS

When the fast synchronous method of data transfer is used, the cable should have the electrical characteristics as set out in Table 2.4.2.

Characteristics impedance	90 - 132 Ω
Conductor size	0.08042 mm ² (28 AWG) min
Signal attenuation	0.095 Db max per metre @ 5 MHz
Pair to pair propagation delay delta	0.20 ns max per metre
DC resistance	0.230 Ω max per metre @ 20 °C

TABLE 2.4.2 FAST CABLE CHARACTERISTICS

3.

SCSI Protocol

The SBI matches the basic SCSI specification for SEQUENTIAL ACCESS devices as detailed in ANSI X3.131. It implements all STANDARD operations and the majority of EXTENDED and OPTIONAL features including RESELECTION.

A typical operating sequence for the SBI might be:

- SCSI control element enabled for SELECTION.
- INITIATOR selects SBI.
- SBI checks if INITIATOR has set ATN
 - ATN not set. SBI notes that disconnection is not supported, and that the LUN address will be contained in the CDB.
 - ATN set. SBI receives the IDENTIFY message, checks if disconnection is supported, and stores the LUN address. (LUN address within the CDB now ignored).
- SBI requests COMMAND DESCRIPTOR BLOCK from the INITIATOR.
- SBI executes command;
 - requests/receives data if necessary;
 - disconnects, if possible, whenever any delay is anticipated (eg tape motion);
 - carries out any tape-related operations.
- SBI RESELECTS INITIATOR if necessary.
- SBI sends COMPLETION STATUS.
- SBI sends COMMAND COMPLETE message;
 - LINK not set, SBI disconnects and awaits selection;
 - LINK set, SBI requests next CDB without disconnecting.

The SBI supports a multi-INITIATOR environment as in 3rd PARTY RESERVE/RELEASE and reporting of the ATTENTION sense key.

3.1 Message Protocol

The SCSI board supports the majority of the messages available under the SCSI MESSAGE PROTOCOL, as listed in Table 3.1.

The SCSI board will attempt to receive a message from the INITIATOR as soon as possible after ATN has signalled its availability; any data phase will be abandoned immediately and the message out phase entered (if necessary, the data phase will be restarted via a RESTORE POINTERS message).

Command name	Direction	
COMMAND COMPLETE	in	
SAVE DATA POINTER	in	
RESTORE POINTERS	in	
DISCONNECT	in	
INITIATOR DETECTED ERROR		out
ABORT		out
MESSAGE REJECT	in	out
NO OPERATION		out
MESSAGE PARITY ERROR		out
LINKED COMMAND COMPLETE	in	
LINKED COMMAND COMPLETE(flag)	in	
BUS DEVICE RESET		out
SYNCH DATA TRANSFER REQUEST	in	out
IDENTIFY	in	out

Note: in/out is referenced to the INITIATOR.

Table 3.1 SCSI MESSAGE PROTOCOL

Command Complete

Sent by the SCSI board to indicate that a command (or sequence of LINKED commands) has just been completed, and that valid status has been sent. Once sent, the SCSI board enters the BUS FREE phase and awaits selection.

Save Data Pointer

Sent by the SCSI board prior to disconnecting during a data transfer command. The SCSI board intends to reconnect at a later stage and resume data transfers. The SCSI board expects the INITIATOR to respond to this message by saving the DATA POINTER for the current LUN.

Restore Pointers

Issued by the SCSI board whenever it wants to retry (send or receive) the previous transfer. This will normally occur whenever a parity error is detected on the SCSI bus, but may also occur if a DATA phase is cut short by the Host requesting a MESSAGE OUT phase.

Disconnect

Sent by the SCSI board to inform the INITIATOR that the SCSI board is about to disconnect, but intends to reconnect at a later stage to complete the command. The SCSI board expects the INITIATOR to save the COMMAND and STATUS pointers for the current LUN in response to this message.

Initiator-detected Error

When received, the SCSI board issues a Restore Pointers message and re-sends the faulty data.

Abort

Received by the SCSI board; results in any current LUN operation being halted and the command abandoned. The SCSI board goes directly to the BUS FREE phase without sending STATUS or COMMAND COMPLETION.

Any buffered data from previous commands is preserved following an Abort message.

Message Reject

Sent by the SCSI board if the message received is not implemented.

Received by the SCSI board; this should only happen in response to the SCSI board's SAVE DATA POINTER and DISCONNECT messages. If this occurs, the impending disconnect will be cancelled, and the command completed without disconnection.

No Operation

Ignored if received; present function will continue as normal.

Message Parity Error

Received by the SCSI board if the INITIATOR detected a parity error in the last message sent. The SCSI board will re-try up to 4 times before going directly to the BUS FREE phase.

Linked Command Complete

Sent by the SCSI board to indicate that a LINKED command has completed, and that valid INTERMEDIATE STATUS has been sent.

Linked Command Complete (with flag)

Sent by the SCSI board to indicate that a LINKED command with the FLAG bit of the CDB has completed, and that valid INTERMEDIATE STATUS has been sent.

Bus Device Reset

Received by the SCSI board; treated by the SCSI board like a HARD RESET on the SCSI bus. Any current operations are cleared, and the SCSI board goes directly to the BUS FREE phase. All reservations are cancelled, and all LUNs returned to VARIABLE mode.

Identify

Received by the SCSI board immediately following the SELECTION phase if the INITIATOR had set ATN during it.

This single-byte message contains further encoded information in bits 0-6 as detailed next.

Bit	Function
6	When set to 1, indicates that the INITIATOR can support DISCONNECTION. The SCSI board will not attempt any later disconnection if this bit is set to 0.
5 - 3	<i>Not used.</i>
2 - 0	These bits specify the LOGICAL UNIT NUMBER for the current command. This LUN address will be used rather than the one contained in the CDB.

The IDENTIFY message is also sent by the SCSI board immediately following any RESELECTION by it. The current LUN address will be set in bits 0-2, as above, with bits 3-6 set to 0. The SCSI board expects the INITIATOR to implement an implied RESTORE POINTERS operation as a result of this message.

Synchronous Data Transfer Request

This is a multi-byte extended message with the format shown:

Byte	Value	Description
0	01h	Extended message
1	03h	Extended message length
2	01h	Synchronous Data Transfer Request code
1	03h	Extended message length
3	m	Transfer Period (m times 4 ns)
4	x	REQ/ACK offset count

A pair of SDTR messages will be exchanged between the SCSI board and the Initiator whenever there is a need to establish a new, or known, data transfer mode. The SCSI board will initiate the message exchange following Power-on, a SCSI Bus Reset or a Bus Device Reset message. In addition, the Initiator may initiate a message exchange whenever it wants to.

a) Host-initiated Exchange

The SCSI board responds to this message by entering the Message In phase and sending a Synchronous Data Transfer Request (SDTR) message to the Initiator. If the SCSI board is able to support the period and offset parameters requested by the Initiator, the same values are returned, if not, the SCSI board sets its own values to establish the operational limits.

If the initiator re-asserts ATN and issues a Message Reject message following the SCSI board's SDTR message, the SCSI board will revert to Asynchronous data transfer mode and proceed to the next phase.

If the initiator re-asserts ATN and issues a Message Parity Error message following the SCSI board's SDTR message, the SCSI board will retry the message 4 times before reverting to Asynchronous data transfer mode and going bus free.

b) SBI-initiated Exchange

The SCSI board will enter the Message In phase following the first Command Phase after Power-on, Bus Reset, or Bus Device Reset message and send an SDTR message indicating its operational limits for synchronous transfers. It expects the Initiator to respond with a further SDTR message indicating that it can support the SCSI board's parameters, or setting its own lower operational limits. The values returned by the Initiator will then be used for synchronous communication.

If the Initiator sends Message Parity Error or Initiator Detected Error messages, the SCSI board will retry 4 times before reverting to asynchronous mode and going Bus Free. If the Initiator sends a Message Reject message, the SCSI board will revert to asynchronous mode and proceed with the command. If any other message is received, the SCSI board will revert to asynchronous mode and go Bus Free.

The Transfer Period, determined by byte 3, is the minimum time allowed between the leading edges of consecutive Request and of consecutive Acknowledge pulses.

The SCSI board supports a minimum transfer period of 100 nano seconds, equating to a data transfer rate of 10 MB/s and giving a minimum value for 'm' of 25.

The REQ/ACK Offset Count, determined by byte 4, is the maximum number of REQ pulses that can be outstanding before the leading edge of the first corresponding ACK pulse.

The SCSI board supports an offset range of 0-15, where 0 implies asynchronous mode.

The Super SCSI has 2 synchronous modes of operation, NORMAL and FAST as defined by the SCSI-2 specification. Both modes are identical in general operation, except that NORMAL mode will negotiate for a synchronous transfer rate of 5 MB/s while FAST will negotiate for a synchronous transfer rate of 10 MB/s. However, various setup, assertion and hold timing parameters associated with synchronous data transfer are different between the 2 modes, in line with the SCSI-2 specification. This means that setting the board for FAST mode, and allowing the Initiator to limit the synchronous transfer rate to 5 MB/s, does not give the same operating conditions as 5 MB/s in NORMAL mode which essentially provides greater margins at the same nominal data rate.

3.2 Command Set

The commands supported are listed in Table 3.2, and are described in full in Section 4.

All commands and parameter data are checked for zero Reserved Fields. Any non-zero reserved fields will result in a Check Condition/Illegal Request/Illegal Bit in CDB (or Illegal Bit in Parameter list) response.

The SBI carries out a 2-part RAM check at power on. A minimum level of good ram is checked for, followed by a full buffer check. If the minimum level check fails, the board shuts down and will not respond on the SCSI bus. If the data buffer check fails, the SBI will return Check Condition/Hardware Error on all commands except Inquiry (operates normally) and the first command received which will return Check Condition/Attention/Power-on as normal.

Command name	Code(h)
TEST UNIT READY	00
REWIND	01
REQUEST SENSE	03
READ BLOCK LIMITS	05
READ FORWARD	08
WRITE DATA	0A
READ REVERSE	0F
WRITE FILEMARKS	10
SPACE	11
INQUIRY	12
VERIFY	13
RECOVER BUFFERED DATA	14
MODE SELECT	15
RESERVE UNIT	16
RELEASE UNIT	17
COPY	18
ERASE	19
MODE SENSE	1A
UNLOAD	1B
RECEIVE DIAG. RESULTS	1C
SEND DIAGNOSTIC	1D
LOCATE	2B
WRITE BUFFER	3B
READ BUFFER	3C
READ POSITION	34
LONG BLOCK READ	07
LOG SENSE	4D
LOG SELECT	4C
LIMITED LONG BLOCK WRITE	0C

TABLE 3.2 COMMANDS SUPPORTED

4.

Command Descriptions

For each SCSI command, this section

- sets out the Command Descriptor Block (CDB),
- describes the function of the command,
- notifies any limitations,
- illustrates the format of returned data.

4.1 Test Unit Ready

bit	7	6	5	4	3	2	1	0
byte 0	0	0	0	0	0	0	0	0
byte 1	LUN			reserved				
byte 2	reserved							
byte 3	reserved							
byte 4	reserved							
byte 5	0	0	reserved			flag	link	

The SBI checks the specified LUN to see if it is READY.

GOOD STATUS is returned if the specified LUN's ON-LINE and READY status lines are both detected true.

CHECK CONDITION status is returned with NOT READY set in the sense data if either ON-LINE or READY is detected false, eg if the LUN is not present, or is connected and on-line, but rewinding.

4.2 Rewind

bit	7	6	5	4	3	2	1	0
byte 0	0	0	0	0	0	0	0	1
byte 1	LUN			reserved			IMM	
byte 2	reserved							
byte 3	reserved							
byte 4	reserved							
byte 5	0	0	reserved			flag	link	

This command rewinds the specified LUN to BOT.

If the IMM bit is set within the CDB, the SBI will issue the rewind command, check that the LUN responds, and then complete without waiting for LDPT status to be set. Otherwise the SBI waits for the LUN to reach BOT and set LDPT status.

4.3 Request Sense

bit	7	6	5	4	3	2	1	0	
byte 0	0	0	0	0	0	0	1		
byte 1	LUN			reserved					
byte 2	reserved								
byte 3	reserved								
byte 4	Allocation Length								
byte 5	0	VU1	reserved				flag	link	

The SBI responds to this command by sending all or part of its SENSE DATA block for the specified LUN. The amount of data sent is specified by the INITIATOR in the ALLOCATION BYTE of the CDB. The amount of data available for return will depend on whether the vendor unique bit VU1 is set, and/or whether any Copy Target data is available as shown in Table 4.3.1 below.

Although the SBI will implement this command at any time, the sense data returned is only valid if the previous command to the same LUN resulted in a CHECK CONDITION status.

This data is preserved by the SBI for any particular LUN until

- a) a REQUEST SENSE has been processed for the original INIT/LUN pair;
- b) any other command has been processed for the original INIT/LUN pair.

The REQUEST SENSE command itself will only return error status if a parity error is detected on the SCSI bus, or any RESET has taken place since the original error. If this does occur, the original data is lost, new data is set, and the command terminates immediately with CHECK CONDITION status.

The SBI always operates in EXTENDED SENSE mode with the format shown in Table 4.3.1.

Byte 0

The VALID bit when set indicates that bytes 3,4,5 and 6 contain valid information (see below).

The ERROR CODE bits are used to indicate whether the CHECK CONDITION and sense data apply to the command that returned the CHECK CONDITION (error code set to 70h) or to a previous command, such as a buffered WRITE, which had already returned GOOD STATUS (deferred error, error code 71h).

bit	7	6	5	4	3	2	1	0	
byte									
byte 0	Valid		Error Code						
byte 1	Copy Command			Current Segment Number					
byte 2	FM	EOM	ILI	0	Sense Key				
byte 3	Information Byte (msb)								
byte 4	Information Byte								
byte 5	Information Byte								
byte 6	Information Byte (lsb)								
byte 7	Additional Sense Length								
byte 8	Copy Command		Source Sense Data Offset						
byte 9	Copy Command		Destination Sense Data Offset						
byte 10	0	0	0	0	0	0	0	0	
byte 11	0	0	0	0	0	0	0	0	
byte 12	Additional Sense Code								
byte 13	Additional Sense Code Qualifier								
byte 14	0	0	0	0	0	0	0	0	
byte 15	0	0	0	0	0	0	0	0	
byte 16	0	0	0	0	0	0	0	0	
byte 17	0	0	0	0	0	0	0	0	
if VU1 = 0									
byte 18-n	Copy Target(s) Additional Sense Data								

TABLE 4.3.1

continued

... continued

or, if VU1 = 1

byte 18	FPTD	RWDG	ONL	RDY	EOT	LDPT	DBY	FBY
byte 19	0	0	NRZ	FM	CER	HER	0	0
byte 20	Full Status Byte						F1	
byte 21	Full Status Byte						F2	
byte 35	Full Status Byte						F16	
byte 36	Full Status Byte						F17	
byte 37-n	Copy Target(s) Additional Sense Data							

TABLE 4.3.1

Byte 1

This byte is only valid for sense data associated with the COPY command. It determines the Segment Descriptor (0-255) that was being processed when the error occurred.

Byte 2

The Filemark (FM) bit indicates that the LUN encountered a filemark when executing the last command. This may or may not be the cause of the CHECK CONDITION status.

The End of Medium (EOM) bit indicates that the LUN is either at LDPT or EOT. The INITIATOR will either know which from past events, or can tell from the additional sense bytes (see below). Again this bit may or may not be the cause of the error.

The Illegal Length (ILI) bit indicates that the size of the block read from tape did not match the length allowed for in the CDB.

The SENSE KEY bits are as defined in Table 4.3.2.

Bytes 3, 4, 5 & 6

The INFORMATION BYTES are used to indicate any difference (residue) between the requested length (bytes or blocks) and the actual length in any given command, including block residues associated with COPY command segment descriptors.

The SBI supports negative residues which are represented in 2's complement notation (see SPACE command).

Byte 7

The ADDITIONAL SENSE LENGTH byte indicates the amount of additional sense data available. This will normally be set 10, but can be 28 or 46 if extra sense is available for a COPY command.

Code	Meaning
0h	No sense data available. CHECK CONDITION likely to have been caused by FM, EOM or ILI being set.
1h	<p>This sense key indicates that the previous WRITE or READ command required some re-try action before eventually completing successfully.</p> <p>The SBI considers the command to have been successful, and will not set any residue information; a fixed WRITE/READ will continue to completion after a successful re-try operation.</p> <p>The reporting of this sense key can be inhibited by an option switch, see Appendix A under 'SW2-4' entries.</p>
2h	NOT READY. Addressed LUN was not ready for a new command.
3h	MEDIUM ERROR. Non-recoverable tape error was encountered.
4h	<p>Hardware Error</p> <ul style="list-style-type: none"> a. parity error on SCSI bus, b. ram error detected during self-test, c. hardware error detected during LUN diagnostic checks.
5h	Illegal Request. Usually indicates an illegal parameter in the CDB, or a non-permitted action requested.
6h	<p>UNIT ATTENTION. This status is set to indicate that the LUN media may have been removed, or the SBI reset since the last command was given. When set, the current command is effectively aborted (except INQUIRY) and no further action taken. All subsequent commands are then implemented as normal. Note that all INITIATORS will be informed in turn if and when they attempt to address the LUN for which the condition was detected.</p> <p>The SBI sets the ATTENTION status in response to</p> <ul style="list-style-type: none"> a. power-on, b. BUS DEVICE RESET message, c. RESET on SCSI bus, d. LUN has been detected off-line at some time since last command issued. <p>Note that a), b) and c) will result in ATTENTION being set for all LUNs.</p> <p>This sense key can be inhibited by an option switch SW2-3, see Appendix A for details.</p>
7h	FILE PROTECTED. Set if any WRITE/ERASE command is requested on a LUN which is write protected.

TABLE 4.3.2(a) SENSE KEY CODES 0-7 IN BYTE 2

Code	Meaning
8h	BLANK TAPE. Set if the LUN has not detected any data/filemark within approximately 20 feet of the current position during any forward READ or SPACE operation.
9h	DENSITY MISMATCH. Set by the SBI if a density mismatch between LUN and tape is detected on a READ from BOT.
Ah	COPY ABORTED. Indicates that the COPY command was aborted due to a failure on the Source and/or Destination devices.
Bh	ABORTED COMMAND. Indicates that the TARGET has aborted the command.
Ch	<i>Not used by the SBI.</i>
Dh	VOLUME OVERFLOW. Indicates that the last command has left data in the buffer which hasn't been written to tape.
Eh	MISCOMPARE. Indicates an error detected during a VERIFY command.
Fh	<i>Not used by the SBI.</i>

TABLE 4.3.2(b) SENSE KEY CODES 8-F IN BYTE 2

Byte 8

This byte represents the offset from byte 0 of the 1st byte of sense data of a COPY command's Source device. A value of 0 indicates that no such sense data is available.

Byte 9

This byte represents the offset from byte 0 of the 1st byte of sense data of a COPY command's Destination device. A value of 0 indicates that no such sense data is available.

Bytes 12 and 13

The Additional Sense Code and Additional Sense Code Qualifier bytes will often contain further information related to the error or exception condition that has occurred. Their content is as defined by the SCSI-2 Working Draft Proposal, rev 10b.

Bytes 18-n

The contents of these bytes depend on whether the vendor unique bit VU1 in the CDB is set, and/or if any Copy Target data is available, see Table 4.3.1 above.

If VU1 is set, 2 status bytes which reflect the LUN's Pertec interface lines are returned followed by the LUN's internally held status byte block.

Any Copy Target data is returned last. Bytes 8 and 9 of the sense data define the amount of data available and their position within the returned sense data.

4.4 Read Block Limits

bit	7	6	5	4	3	2	1	0
byte 0	0	0	0	0	0	1	0	1
byte 1	LUN			reserved				
byte 2	reserved							
byte 3	reserved							
byte 4	reserved							
byte 5	0	0	reserved				flag	link

This command causes the SBI to send 6 bytes of data as shown in Table 4.4 which define the minimum and maximum block lengths that the SBI can handle.

The minimum block length is always set to 1.

The maximum block length is always set to FFFFFFFh.

bit	7	6	5	4	3	2	1	0
byte 0	0	0	0	0	0	0	0	0
byte 1	Maximum Block Length (msb)						(FFh)	
byte 2	Maximum Block Length (FFh)							
byte 3	Maximum Block Length (lsb)						(FFh)	
byte 4	Minimum Block Length (msb)						(00h)	
byte 5	Minimum Block Length (lsb)						(01h)	

TABLE 4.4.1

Note that although normal buffered READ/WRITE operations are limited to 64K data blocks (10000h), the SBI can write and read 16 Mbyte (FFFFFFh) blocks. This involves a special algorithm which the SBI invokes automatically whenever the transfer length bytes of a variable WRITE/READ exceed 10000h. This is essentially transparent to the host, but does require it to have a minimum transfer rate capability; see Section 4.5.4 for full details.

4.5 Read Forward

bit	7	6	5	4	3	2	1	0
byte 0	0	0	0	0	1	0	0	0
byte 1	LUN			reserved			SILI	FXD
byte 2	Transfer Length (msb)							
byte 3	Transfer Length							
byte 4	Transfer Length (lsb)							
byte 5	0	LAR	reserved				flag	link

This command causes data to be read from tape in the forward direction and transferred to the INITIATOR. The amount of data read/transferred depends on the transfer length requested and whether the command is fixed or variable as defined by the FXD bit of the CDB.

4.5.1 Variable Mode

In this mode the SBI reads a single block of data from the tape and transfers it to the INITIATOR before completing.

The TRANSFER LENGTH (TL) bytes of the CDB specify the number of bytes the INITIATOR has allocated/expects for returned data. The amount of data actually returned depends on how TL compares with the block length read (BLR) from tape, as shown below.

a) BLR = TL

The data block is transferred as read from tape, and GOOD STATUS sent to complete.

b) BLR < TL

The data block is sent as read, but with CHECK CONDITION status set. VALID and ILI are set in the sense data, with the INFORMATION bytes set to (TL-BLR).

c) BLR > TL

First TL only bytes are sent with CHECK CONDITION status, VALID and ILI set in the sense data, and the INFORMATION bytes set to (TL-BLR), (negative residue in 2's comp notation, see SPACE command).

The above error reporting on wrong block length is suppressed if the SILI bit of the CDB is set; GOOD STATUS is returned, with the number of bytes transferred following the normal rules.

Note that if the TRANSFER LENGTH is set to zero, no tape motion takes place, and GOOD STATUS is returned.

Several abnormal conditions can occur which will result in CHECK CONDITION status being set, these are described next.

d) LUN in Fixed Mode

ILLEGAL REQUEST will be set in the sense data with no tape motion implemented.

e) Transfer Length too large

ILLEGAL REQUEST will be set in the sense data with no tape motion implemented.

f) Filemark Encountered

NO SENSE and FM are set in the sense data. The tape is left positioned on the EOT side of the FM read.

g) Irrecoverable Tape Error

If the SBI exhausts its re-try sequence, MEDIUM ERROR is set in the sense data, and no data is sent. The tape is left positioned on the EOT side of the faulty block.

h) EOT Encountered

Note: EOT will be ignored by the read command if an option switch is set, see Appendix A under 'SW3-2' entries.

NO SENSE and EOM are set in the sense data. Note that the SBI will not abort the READ in progress, but will carry on and attempt to read the block. Data is then transferred with any further sense data set as appropriate.

Note also that the SBI will implement a VARIABLE READ command even if EOT is already set. It is left to the INITIATOR and any LUN safety mechanisms to preserve tape integrity.

i) Physical end of tape

If the LUN's limit point is reached, MEDIUM ERROR and EOM are set in the sense data. Valid is set, with the Information Bytes set to the Transfer Length.

4.5.2 Fixed Mode

In this mode, the SBI transfers a number of contiguous blocks of data from the tape to the INITIATOR before completing. The number of blocks to be transferred is specified by the TRANSFER LENGTH bytes of the CDB.

On receipt of the command, the SBI self-configures its buffer into a number of sub-buffers which it then uses to try and maintain an optimum data flow between tape and INITIATOR.

The command completes with GOOD STATUS when the specified number of blocks have been transferred to the INITIATOR.

Several abnormal conditions can occur which result in CHECK CONDITION status being sent:

a) LUN in Variable Mode

ILLEGAL REQUEST will be set in the sense data with no tape motion implemented.

b) Filemark Encountered

NO SENSE and FM are set in the sense data (see below for VALID). The tape is left positioned on the EOT side of the FM.

c) EOT Encountered

Note: EOT will be ignored by the read command if an option switch is set, see Appendix A under 'SW3-2' entries.

NO SENSE and EOM are set in the sense data.

The command is terminated after the data block during which EOT was detected has been transferred to the INITIATOR. The tape is left positioned after this data block.

As in variable mode, the SBI will implement a FIXED READ even if EOT is already set. The command will always complete as above, however, after the first block read.

d) Physical End of Tape

If the LUN's limit point is reached, MEDIUM ERROR and EOM are set in the sense data.

e) Irrecoverable Tape Error

If the SBI exhausts its re-try sequence on any one block, the command is terminated and MEDIUM ERROR set in the sense data (see below for VALID).

The tape is left on the EOT side of the faulty block, which is not transferred to the INITIATOR.

f) Wrong Block Size

The command is terminated if a block is encountered which is larger or smaller than the current block length (see MODE SELECT). NO SENSE and ILI are set in the sense data (see below for VALID). The tape is left on the EOT side of the "faulty" block, which is not transferred to the INITIATOR.

The use of the SILI bit is not allowed in fixed mode. If the SILI and FXD bits are both set in the CDB, CHECK CONDITION with ILLEGAL REQUEST sense key will be set, with no tape motion taking place.

Note 1: in b) to d) of the above, the VALID bit is set in the sense data, and the INFORMATION BYTES set to the difference between the number of blocks specified (TL), and the number actually transferred to the INITIATOR.

Note 2: if the TRANSFER LENGTH is set to zero, no tape motion takes place, and GOOD STATUS is returned.

4.5.3 Look-ahead Read

Look-ahead Read (LAR) is essentially a vendor unique option on the normal READ FORWARD command, designed to maximize system throughput. It is recommended for use whenever the host anticipates a significant number of sequential READ operations, and is implemented by setting bit 6 of byte 5 within the CDB, or by setting option the HIGH PERFORMANCE MODE switch (see Appendix, 'SW1-4' entries) to its off position before power-on (see below).

The look-ahead aspect of the command is wholly transparent to the host, with the exception of any throughput changes. The host 'sees' a normal READ operation with all aspects of the normal READ described above still applying.

The basic LAR operation is as follows:

- read next block from tape;
- transfer data to host, and complete SCSI command;
- await new command;
- while awaiting new command, tape motion is continued and subsequent blocks (including file marks) read into the SBI's buffer in anticipation of further LAR commands;
- tape motion continues until the data buffer is full (less than 64K RAM free) or an error condition is encountered. If the buffer full condition occurs, further blocks are read from tape as space becomes available following further LAR commands from the host;
- an error condition will result in the tape motion being halted; the SBI will then continue to supply data from the buffer in response to further LAR commands until it catches up with the faulty lock, when CHECK CONDITION will be returned in the normal way.

Possible error conditions are:

- i) Irrecoverable read error (re-trying occurs in the normal way);
- ii) blank tape encountered;
- iii) EOT encountered;
- iv) block length greater than 64K (10000h) encountered;
- v) end of recorded data.

'End of recorded data' in the above is defined as a preset number of consecutive filemarks, or the last block which the SBI knows it recorded on this tape. The ability to determine the last block written is lost if the drive is taken off-line at any time following the WRITE operation, and is not available in a read-only environment. The number of consecutive filemarks required to halt the LAR operation defaults to 2 at power-on, and can be altered via the MODE SELECT command, or the front panel option settings available in the 9914 Streamer.

The look-ahead action is also halted if any command other than a further LAR, SPACE, INQUIRY, TEST UNIT READY, REQUEST SENSE, MODE SELECT, or MODE SENSE is received from the host. This will stop any tape motion and reposition the tape back to its logical position before the command is implemented. (The REWIND command will rewind directly to BOT).

Any block length errors detected when reading from the buffer are reported/handled as per normal operation, but will not abort the look-ahead action (except greater than 64K).

As indicated, the INQUIRY command may be implemented without affecting the look-ahead READ operation. This command returns 2 bytes which reflect the LUN's Pertec interface status lines. If initiated during a LAR operation, the bytes returned will reflect the status of the tape unit at the time the associated data block was read from tape, ie the tape units logical position, and not its current physical position.

The LAR option can be "permanently" enabled following power-on via the HIGH PERFORMANCE MODE option switch (see Appendix A, 'SW1-4' entries), ie all 'normal' reads from the host are implemented as Look-ahead READS. This mode stays in operation until cancelled by the MODE SELECT command (bit 6, byte 5 of the MODE SELECT CDB set). Further LAR operations are then implemented by setting bit 6, byte 5 within the READ command's CDB on each READ operation.

4.5.4 Read Long Block

This is a special variation of the normal variable READ command which can be used to read block lengths from 1 to 16,777,215 bytes (normal reading is limited to 65,536 bytes). The long read algorithm is invoked automatically whenever a transfer length greater than 65536 is requested.

When operating the long read algorithm, the SBI operates in a pseudo-synchronous mode with the host. Although the data is transferred in true asynchronous mode as normal, it is transferred to the host in 65,534 byte segments (plus any residue); special DMA techniques are employed to ensure that the data being continuously presented by the tape unit's true synchronous interface is accepted, temporarily buffered, then passed on to the host. Note that irrespective of the block length, only a single CDB is required to initiate the operation, and only one Status/Complete sequence presented as normal on completion.

Receipt of a Read Long Block command cancels any look-ahead read action and forces low speed non-LAR for all subsequent reads, long block or not. A rewind command restores the ability to execute look-ahead reads.

The special hardware action involved is transparent to the host with the exception that it must be capable of accepting data from the SBI at a minimum rate equivalent to the tape data rate. The necessary minimum performance is determined by the tape unit's speed and density.

Read Long Blocks are done at the speed set by the MODE SELECT command, giving the minimum required host data rates shown next.

Density & Speed			Data Rate
800 bpi NRZ	Low speed	42 ips	34 kB/s
	High speed	125 ips	100 kB/s
1600 bpi PE	Low speed	42 ips	67 kB/s
	High speed	125 ips	200 kB/s
6250 bpi GCR	Low speed	42 ips	260 kB/s
	High speed	125 ips	781 kB/s

Error checking and reporting is similar to the normal READ command. If a tape error occurs, the error is reported as normal, and the tape left positioned in the gap following the faulty block; no automatic retrying takes place.

The following abnormal conditions can occur and will result in CHECK CONDITION being sent:

a) Tape Error

MEDIUM ERROR is set in the sense data, no retrying takes place.

b) EOT Encountered

The EOM bit is set in the sense data if the command completes beyond the tape's EOT marker. If the sense key is set to NO SENSE, then a complete data block was read as normal. If the sense key is set to ABORT (0Bh), then it should be assumed that an incomplete block was encountered, see below.

c) Physical Limit Encountered

The 9914 normally implements a tape motion inhibit point at approximately 12 feet beyond the EOT marker. All forward commands are terminated by the streamer at this point, and no further forward motion commands are accepted.

If the Long Read encounters this situation, the read is abandoned and Check Condition sent with the EOM bit and the ABORT (0Bh) sense key set.

If the 9914's configuration is set to override the tape motion inhibit point, the greatest care must be taken to ensure the physical end of tape is never reached.

d) Wrong Block Length Encountered

Block length checking, length error reporting, residue calculation etc, operate as per normal READ.

Note that block lengths greater than 16 Mbytes can be 'read' and an indication of tape error or not obtained. Such blocks will always terminate with a CHECK CONDITION/ILLEGAL LENGTH, however, with the residue set to 0 to indicate unknown, and the host transfer limited to the transfer length set in the CDB.

4.6 Write Data

bit	7	6	5	4	3	2	1	0
byte 0	0	0	0	0	1	0	1	0
byte 1	LUN			reserved			FXD	
byte 2	Transfer Length (msb)							
byte 3	Transfer Length							
byte 4	Transfer Length (lsb)							
byte 5	0	0	reserved			flag	link	

This command causes data to be transferred from the INITIATOR and written to tape at the current tape position. The amount of data transferred/written depends on the TRANSFER LENGTH requested and the mode of operation as defined by the CDB and MODE SELECT data. There are 4 modes of operation available, Variable buffered, Variable unbuffered, Fixed buffered and Fixed unbuffered. The current mode is essentially set by power-on default parameters, and/or by a previous MODE SELECT command. The FXD bit of the Write CDB is simply used as a cross check that the INITIATOR 'knows' the current variable/fixed setting. Each mode reacts differently to various situations, as described below in (.1) to (.4).

4.6.1 Variable Mode Unbuffered

In variable mode, the TRANSFER LENGTH bytes (TL) specify the number of bytes to be transferred from the INITIATOR. The SBI does this and writes the data to tape as a single block. GOOD STATUS is then returned, and the tape left positioned after the block just written.

Several abnormal conditions can occur which will result in CHECK CONDITION status being set; these are described next.

a) LUN in Fixed Mode

ILLEGAL REQUEST is set in the sense data. No data is transferred and no tape motion takes place.

b) Block too Large

ILLEGAL REQUEST is set in the sense data if the transfer length requested is greater than the maximum block size permitted (see READ BLOCK LIMITS command).

No data is transferred and no tape motion takes place.

c) EOT or Early Warning Detected

If EOT (reflective marker) is encountered, or the EARLY WARNING indication (EW, which is available from the 9914, as Early EOT) is received, the data is written to tape as normal, but CHECK CONDITION status is sent to complete, with VALID, EOM and NO SENSE set in the sense data, and the Information Bytes set to 0.

Note that a variable write is implemented as normal even if EOT/EW is already set when the command is given. It will always complete as above, however, with any other sense data set as required.

d) Irrecoverable Write Error

If the SBI exhausts its re-try sequence trying to write the data to tape, VALID and MEDIUM ERROR are set in the sense data, with the Information Bytes set to the Transfer Length. The final attempt at writing the block is left on tape with the head positioned on either the BOT or EOT side of the faulty block, as set by the option switch SW3-3.

e) Physical End of Tape

If the LUN's physical tape limit is reached, CHECK CONDITION is set with VALID, EOM and VOLUME OVERFLOW set in the sense data. The Information bytes will be set to the Transfer Length.

4.6.2 Fixed Mode Unbuffered

In FIXED MODE, the TRANSFER LENGTH bytes specify the number of contiguous data blocks to be written, each of a size equal to the CURRENT BLOCK LENGTH (see MODE SELECT command).

Even in unbuffered mode, the fixed write operation can result in several blocks of data being held in the buffer waiting to be written to tape, and hence maintaining tape streaming. Unlike true buffered mode, however, the command will not complete until all associated data has actually been written to tape.

Several abnormal conditions can occur which will result in CHECK CONDITION status being sent, these conditions are described next.

a) LUN in Variable Mode

ILLEGAL REQUEST is set in the sense data, with no data being transferred and no tape motion taking place.

b) EOT or Early Warning Detected

If EOT is encountered or EW received, the command is terminated when the current block is written to tape. CHECK CONDITION status is sent; VALID, EOM and NO SENSE are set in the sense data, with the INFORMATION BYTES set to the difference between the transfer length requested and the number of blocks successfully written to tape.

Note that the SBI will attempt a fixed write even if EOT/EW is already set when the command is given. It will always terminate as above, however, after the first block has been written.

c) Irrecoverable Write Error

If the SBI exhausts its re-try sequence on any given block, the command is terminated. CHECK CONDITION status is sent; VALID, EOM and MEDIUM ERROR are set in the sense data, with the INFORMATION BYTES set to the difference between the transfer length requested and the number of blocks successfully written to tape. The final attempt at writing the block is left on tape with the head positioned on either the BOT or EOT side of the faulty block, as set by option switch SW3-3.

d) Physical End of Tape

If the LUN's physical tape limit is reached, CHECK CONDITION is set with VALID, EOM and VOLUME OVERFLOW set in the sense data. The INFORMATION BYTES will be set to the difference between the transfer length requested and the number of blocks successfully written to tape.

Note that the above errors can result in data blocks having been transferred from the INITIATOR, but not yet written to tape. This data can be recovered in the normal way by using the RECOVER BUFFERED DATA command. It will not be force written, however, by the WRITE FILEMARKS command.

4.6.3 Write Buffered Mode, Variable/Fixed

When in BUFFERED mode, which is set/reset via the MODE SELECT command, the SBI essentially transfers data in fixed and variable mode as before, except that GOOD STATUS is returned, and the command "completed", as soon as all of the host data has been transferred into the buffer.

The SBI then starts (or continues) to write the data to tape as normal, but can at the same time receive a further command from the Host. If this command is a further write data command (or filemark with IMM set), it is implemented immediately, and its associated data (or filemarks) transferred into the buffer as and when space is available. Depending on the block length/buffer size, several commands worth of data may be stacked up in this way waiting to be written to tape. Normally any commands other than WRITE are held and not implemented until all buffered data has been written to tape. The following commands, however, can be given on-the-fly without affecting the buffered WRITE:

ERASE
TEST UNIT READY
MODE SELECT
MODE SENSE
REQUEST SENSE.

If the EW indication is received from the tape unit, the SBI effectively suspends buffered mode and attempts to write all buffered data and filemarks to tape before returning a CHECK CONDITION (either on the current command, if still active, or on the next command received from the same INITIATOR). EOM and VALID will be set, with the sense key and information bytes depending on the mode as detailed below.

The SBI behaves similarly if EOT (reflective marker) is detected, except that it will halt after the current block has been written to tape, and will not attempt to empty its buffer; the sense data will also follow the rules which are described next.

a) Buffered Mode, Buffer Empty

If no buffered data or filemarks remain to be written, the sense key is set to No Sense, and the Information Bytes set to 0.

b) Buffered Mode, Buffered Data/Filemarks, Variable Mode

The sense key is set to Volume Overflow, and the Information Bytes set to the total number of unwritten bytes remaining in the buffer. Any buffered filemarks are not included in this total.

c) Buffered Mode, Buffered Data/Filemarks, Fixed Mode

The sense key is set to Volume Overflow, and the Information Bytes set to the total number of data blocks and filemarks remaining to be written (ie all buffered data and filemarks from previous commands, plus all data blocks from this command)

The SBI will attempt to implement a WRITE command even when EOT/EW is already set. It will always complete with a CHECK CONDITION, however, even if successful, with the sense data as above.

If Physical EOT is encountered, CHECK CONDITION is sent with Valid and Volume Overflow set in the sense data. The Information bytes will be set as above.

An irrecoverable write error will also halt the tape action. Valid and Medium Error will be set in the sense data, with the Information bytes following the above rules.

If any of the above occur before GOOD STATUS has been returned, the command terminates in the normal way as per the non-buffered mode. If GOOD STATUS has already been sent, the appropriate sense data is held by the SBI until a further command is received for the same INIT/LUN pair. When received, or if already pending, this command is terminated immediately with a CHECK CONDITION and the Deferred Error code set in the sense data; REQUEST SENSE can then be implemented in the normal way. An exception to this is the REQUEST SENSE command itself, which will return the sense data, valid or otherwise, without a further CHECK CONDITION.

4.6.4 Write Long Block

This is a special variation of the normal variable write command which can be used to write block lengths from 1 to 16,777,215 bytes (normal writing is limited to 65,536 bytes). The long write algorithm is invoked automatically whenever a transfer length greater than 64K (10000h) is requested.

When the long block algorithm is in operation, the SBI operates in a pseudo-synchronous mode with the host. Data is transferred from the host in 65,534 byte segments (plus any residue); special DMA techniques are employed to ensure data is continuously available at the tape unit's true synchronous interface.

This special hardware action is transparent to the host with the exception that it must be capable of transferring data to the SBI at a minimum rate equivalent to the tape data rate. The necessary minimum performance is determined by the tape unit's speed and density. Long blocks writes are done at the speed set by the MODE SELECT command, giving the minimum required host data rates shown in Table 4.6.4.

Error checking and reporting is similar to the normal WRITE command. Note, however, that as the data is not actually retained in the buffer, no automatic retrying is possible. If a tape error occurs, the error is reported as normal, and the tape left positioned in the gap following the faulty block.

Density & Speed			Data Rate
800 bpi NRZ	Low speed	42 ips	34 kB/s
	High speed	125 ips	100 kB/s
1600 bpi PE	Low speed	42 ips	67 kB/s
	High speed	125 ips	200 kB/s
6250 bpi GCR	Low speed	42 ips	260 kB/s
	High speed	125 ips	781 kB/s

TABLE 4.6.4 MINIMUM HOST DATA RATES

Certain abnormal conditions can occur and will result in CHECK CONDITION being sent, these conditions are described next.

a) Tape Error

MEDIUM ERROR is set in the sense data, no retrying takes place.

b) EOT Encountered

The EOM bit is set in the sense data if the command completes beyond the tape's EOT marker. If the sense key is set to NO SENSE, then the data block has been written successfully. If the sense key is set to ABORT (0B), then the data block has not been completed, see below.

c) Physical Limit Encountered

The 9914 normally implements a tape motion inhibit point at approximately 12 feet beyond the EOT marker. All forward commands are terminated by the streamer at this point, and no further forward motion commands are accepted.

If the Long Write encounters this situation, the WRITE is abandoned and Check Condition sent with the EOM bit and the ABORT (0B) sense key set.

If the streamer configuration is set to override the tape motion inhibit point, the greatest care must be taken to ensure the physical end of tape is never reached.

In all modes, GOOD STATUS is returned, with no tape motion taking place, if the TRANSFER LENGTH bytes are set to zero.

4.7 Read Reverse

byte	bit	7	6	5	4	3	2	1	0
byte 0		0	0	0	0	1	1	1	1
byte 1		LUN			reserved		SILI	FXD	
byte 2		Transfer Length (msb)							
byte 3		Transfer Length							
byte 4		Transfer Length (lsb)							
byte 5		0	0	reserved			flag	link	

This command is essentially identical to the READ FORWARD command, except that tape motion is in the reverse direction, and BOT (LDPT) is monitored in place of EOT.

Notes:

- i) Data is transferred to the INITIATOR in the reverse order in which it was written.
- ii) Unlike the SPACE command, the reverse direction is implicit in the command code, and that TRANSFER LENGTH/RESIDUE values are interpreted identically to READ FORWARD.
- iii) No attempt is made to read beyond BOT under any circumstances. Detection of BOT when tape is moving will immediately abort the command.
- iv) Any reverse command which completes successfully, but is still beyond EOT, will return a CHECK CONDITION status with NO SENSE set in the sense data; EOM is not set (the LUN's Pertec interface bytes in the INQUIRY data can be used to confirm that the LUN is beyond EOT).

4.8 Write Filemarks

bit	7	6	5	4	3	2	1	0
byte 0	0	0	0	1	0	0	0	0
byte 1	LUN			reserved			IMM	
byte 2	Number of Filemarks (msb)							
byte 3	Number of Filemarks							
byte 4	Number of Filemarks (lsb)							
byte 5	0	0	reserved			flag	link	

This command results in the required number of filemarks, as specified in the CDB, being written to tape at the current logical tape position.

A Number of Filemarks field of zero will result in no filemarks directly associated with this command being written, but will force write any previously buffered data and filemarks.

If the IMM bit is 0, the command will not complete until all previously buffered data, filemarks and the requested number of filemarks have been written to tape.

If the IMM bit is 1, and the SBI is in buffered mode, the command will complete as soon as the requested number of filemarks has 'entered' the SBI's buffer

CHECK CONDITION with an ILLEGAL REQUEST sense key will be set if the IMM bit is 1, and the SBI is in unbuffered mode.

If the EARLY WARNING (EW) indication is received from the tape unit, and the LUN is in buffered mode, the SBI will attempt to empty its buffer before completing. If EOT (reflective marker) is encountered, or EW in unbuffered mode, the current data block or filemark will be completed normally and the tape halted. In all cases, CHECK CONDITION status will be sent with EOM set in the sense data; the sense key, valid bit and information bytes will depend on the current operating mode of the SBI as detailed next.

a) Buffered Mode, Buffer Empty

If no buffered data or filemarks remain to be written, the sense key is set to No Sense, the Valid bit set to 1, and the Information Bytes set to 0.

b) Buffered Mode, Buffered Data/Filemarks, Variable Mode

The sense key is set to Volume Overflow, the Valid bit set to 1, and the Information Bytes set to the total number of unwritten filemarks (including buffered filemarks from previous commands).

c) Buffered Mode, Buffered Data/Filemarks, Fixed Mode

The sense key is set to Volume Overflow, the Valid bit set to 1, and the Information Bytes set to the total number of data blocks and filemarks remaining to be written (ie all buffered data and filemarks from previous commands, plus all filemarks for this command).

d) Unbuffered Mode

The sense key is set to No Sense, the Valid bit set to 1, and the Information bytes set to the requested number of filemarks less the number of filemarks written to tape.

Note that for a), b) and c) above, if the IMM bit is also set, the CHECK CONDITION will be reported on the first command following the error condition.

The SBI will implement a WRITE FILEMARKS command even if EOT/EW is already set when the command is given. It will always terminate as above, however, after the first filemark, buffered filemark or buffered data block has been written; it is left up to the INITIATOR and any LUN safety mechanisms to preserve tape integrity.

If physical EOT is encountered, CHECK CONDITION will be returned with Valid and Volume Overflow set in the sense data, and the Information Bytes set as above.

4.9 Space

bit	7	6	5	4	3	2	1	0
byte 0	0	0	0	1	0	0	0	1
byte 1	LUN			reserved			CD1	CD0
byte 2	Count (msb)							
byte 3	Count							
byte 4	Count (lsb)							
byte 5	0	0	reserved				flag	link

The SPACE command provides a variety of positioning functions which are determined by the COUNT and CODE settings within the CDB.

The action of SPACE upon encountering EOT is determined by option switch SW3-2 (or its NVR equivalent), as set out in Appendix A.

The functions available are

CD1	CD0	Function
0	0	space blocks
0	1	space filemarks
1	0	space consecutive filemarks
1	1	space to end of data

The COUNT field specifies both the number of blocks etc to be covered, and the direction of tape travel (in 2's complement notation).

Note: 2's complement notation means that if the MS bit of the quantity is set to 1, the contents are interpreted as a negative value, and tape motion is in the reverse direction. Some resulting SPACE actions are

MS	Count field	LS	Action
00	00	02	space forward 2 blocks
00	00	00	no tape motion
FF	FF	FE	space reverse 2 blocks

a) Space Blocks

Tape is spaced in the required direction over COUNT blocks.

Tape motion is halted and CHECK CONDITION status sent if a Filemark, EOT (forward) or BOT (reverse) are detected (see below for VALID). EOM and NO SENSE are set in the sense data.

b) Space Filemarks

Tape is spaced in the required direction over COUNT filemarks. Any intervening data blocks are ignored.

Tape is halted if EOT (forward) or BOT (reverse) is detected. EOM and NO SENSE are set in the sense data, see below for VALID.

c) Space Consecutive Filemarks

Tape is spaced in the required direction until COUNT consecutive filemarks are encountered. Any data block detected, effectively resets the filemark count.

Tape is halted immediately if EOT (forward) or BOT (reverse) is detected, and CHECK CONDITION status sent. EOM and NO SENSE are set in the sense data, see below for VALID.

If any of the above are terminated early, CHECK CONDITION status is sent, with NO SENSE and VALID set in the sense data. The INFORMATION BYTES are set to the difference between the requested COUNT and the number actually spaced.

Note that the value in the INFORMATION BYTES still carries direction information as per the original count. The table below shows the results if space 4 was requested and only 2 were encountered.

Count (hex)			Action	Residue		
MS		LS		MS		LS
00	00	04	space forward 4	00	00	02
FF	FF	FC	space reverse 4	FF	FF	FE

d) Space to End of Data

The SBI defines end of data as 2 consecutive filemarks. This option moves the tape forward until 2 consecutive filemarks are found, reverses back over the second one, leaving the tape between the 2 filemarks ready to append data.

If EOT is detected first, the tape is halted and CHECK CONDITION returned, with Valid and EOM set in the sense data (the information bytes will be set to 2 to indicate that 2 consecutive filemarks could not be found).

e) Beyond EOT

All SPACE forward commands (except **end of data**) will be implemented even if EOT is already set when the command is given. They will terminate as above, however, after the first block (data or filemark) has been detected.

As with REVERSE READ, any space reverse commands which complete successfully, but are still beyond EOT, will return a CHECK CONDITION status with NO SENSE set in the sense data; EOM is **not** set (the LUN's Pertec interface bytes in the INQUIRY data can be used to determine that the LUN is still beyond EOT).

4.10 Inquiry

bit	7	6	5	4	3	2	1	0
byte 0	0	0	0	1	0	0	1	0
byte 1	LUN			reserved				
byte 2	reserved							
byte 3	reserved							
byte 4	Allocation Length							
byte 5	0	0	reserved				flag	link

The INQUIRY command enables the INITIATOR to determine various parameters concerning the SBI and any attached LUNs.

The ALLOCATION LENGTH byte of the CDB specifies the number of bytes that the INITIATOR has allocated for returned data. The amount of data actually returned is as shown below:

AL (decimal)	Bytes returned
0	none (GOOD STATUS sent)
1 <= AL <= 55	AL bytes
56 <= AL	55 bytes

The returned data is as shown in Table 4.10 and described below.

a) Peripheral Device Type

Always set to 01h for LUN address 0, otherwise set to 01h if the LUN is detected on-line, or 7Fh to indicate 'LUN not present' if the LUN is off-line.

(Note that for any given LUN, the SBI does not know whether it is fitted directly into the LUN or is controlling it remotely. It can only tell if the LUN is present via its Pertec interface status lines. If a LUN is not detected on-line, therefore, the SBI assumes that it may have been physically disconnected from the Pertec bus).

b) Device Type Qualifier

Bits 6-0 of byte 1 are used to indicate switch option settings within the SBI (see Appendix A), and the basic recording status of the addressed LUN (only valid if the LUN is on-line). All bits are summarised in Table 4.10.1.

bit	7	6	5	4	3	2	1	0
byte 0	Peripheral Device Type							
byte 1	RMB	0	DE	AE	REE	0	NRZ	WE
byte 2	Version							
byte 3	Response Format							
byte 4	0	0	1	1	0	0	1	0
byte 5	0	0	0	0	0	0	0	0
byte 6	0	0	0	0	0	0	0	0
byte 7	0	0	0	0	0	0	0	0
byte 8	ascii S							
byte 9	ascii T							
byte 10	ascii K							
byte 11	ascii <SP>							
byte 12	ascii <SP>							
byte 13	ascii <SP>							
byte 14	ascii <SP>							
byte 15	ascii <SP>							
byte 16	ascii O							
byte 17	ascii P							
byte 18	ascii E							
byte 19	ascii N							
byte 20	ascii <SP>							
byte 21	ascii R							
byte 22	ascii E							
byte 23	ascii E							
byte 24	ascii L							

TABLE 4.10 INQUIRY DATA

continued ...

byte 25	ascii <SP>
byte 26	ascii T
byte 27	ascii A
byte 28	ascii P
byte 29	ascii E
byte 30	ascii <SP>
byte 31	ascii <SP>
byte 32	Firmware Revision
byte 33	Firmware Revision
byte 34	Firmware Revision
byte 35	Firmware Revision
byte 36	Firmware Revision
byte 37	Firmware Revision
byte 38	Firmware Revision
byte 39	ascii <SP>
byte 40	ascii <SP>
byte 41	ascii <SP>
byte 42	ascii <SP>
byte 43	ascii <SP>
byte 44	LUN 7..... LUN On-line StatusLUN 0
byte 45	LUN 7..... LUN Fixed StatusLUN 0
byte 46	DR 3R I2 I1 I0 R2 R1 R0
byte 47	LUN 7..... LUN Buffered StatusLUN 0
byte 48	FPTD RWDG ONL RDY EOT LDPT DBY FBY
byte 49	0 0 NRZ FM CER HER 0 0

TABLE 4.10 INQUIRY DATA

continued ...

byte 50	msb	Buffer Size Available								
byte 51		Buffer Size Available								
byte 52		Buffer Size Available								lsb
byte 53		0	0	0	0	6250	800	3200	1600	
byte 54		0	125	100	62	50	50 ss	42	25	

TABLE 4.10 INQUIRY DATA

Byte 1 (device type qualifier)

The bits have the following significance:

RMB	Always set to 1 to indicate removable media.
bit 6	Always 0.
DE	Set to indicate that the SBI will disconnect if possible whenever any significant delay is anticipated before completion.
AE	Set to indicate that the SBI will send CHECK CONDITION status with ATTENTION set in the sense data if the LUN has been detected off-line or RESET since the last command.
REE	Set to indicate that CHECK CONDITION status will be returned with the RECOVERED ERROR sense key set if any re-try operations were necessary for successful completion.
NRZ	Set when the LUN is in NRZ density.
WE	Set when the LUN is write enabled.

TABLE 4.10.1 BYTE 1 OF INQUIRY DATA

Version

Set to 09h to indicate compatibility with

ANSI X3.131-1986
ECMA-111

Response Data

Set to 2 to indicate the INQUIRY data format is as per SCSI-2 specification.

Vendor/Product Information

Bytes 8-43 inclusive contain ascii data giving various vendor and product information, Table 4.10 sets out the actual contents.

- Bytes 8-31 Vendor identification and tape type,
- Bytes 32-38 SBI Firmware revision.

Additional Information

The SBI makes 11 additional bytes available, as described next.

c) On-line Status

This byte has each bit corresponding to a LUN as shown:

b7	b6	b5	b4	b3	b2	b1	b0
LUN 7				LUN 0			

When set, the corresponding LUN is present and on-line.

d) Fixed Mode Status

Similar to a), but indicates which LUNs are currently operating in FIXED mode.

e) Reserved Status

This byte can be used to determine whether the addressed LUN is RESERVED and the identity of the RESERVER.

b0	R0	}	ID of INITIATOR to which LUN is currently reserved.
b1	R1		
b2	R2		
b3	I0	}	ID of INITIATOR which carried out 3rd party reservation.
b4	I1		
b5	I2		
b6	3R		LUN reserved, 3rd party
b7	DR		LUN reserved, directly

f) Buffered Status

As per a) and b) above, but indicating which LUNs are currently in buffered mode.

g) Bytes 48 & 49

Reflect the LUN's Pertec interface status lines as shown.

h) Bytes 50-52

Give the data buffer size available on the board. This is determined automatically following power-on and diagnostic self-test.

i) Bytes 53 and 54

Give the LUN densities and speeds available respectively.

CHECK CONDITION status will only be sent for the INQUIRY command if a parity error is detected on the SCSI bus during the command phase.

4.11 Verify

bit	7	6	5	4	3	2	1	0
byte 0	0	0	0	1	0	0	1	1
byte 1	LUN			reserved			BCP	FXD
byte 2	Verification Length (msb)							
byte 3	Verification Length							
byte 4	Verification Length (lsb)							
byte 5	0	0	reserved				flag	link

This command verifies one or more blocks of data from tape in the forward direction. The amount of data verified depends on the VERIFICATION LENGTH (VL) requested and whether the command is fixed or variable as defined within the CDB.

The method of verification is determined by the BCP bit of the CDB. When set to 0, the data block(s) are checked for Medium Error (CER, HER) only; no data is transferred from the INITIATOR. When set to 1, the SBI reads the next block from tape, transfers the "matching" data from the INITIATOR, and carries out a byte-for-byte comparison.

4.11.1 Variable Mode

A single block is checked in this mode as defined by the BCP bit of the CDB, with VL determining the number of bytes to be transferred from the INITIATOR.

The command will complete with CHECK CONDITION set if any of the following occur:

a) CER/HER or Miscompare

BCP state	Check Condition if
0 or 1	CER or HER status detected from the LUN (sets MEDIUM ERROR sense key).
1	A byte miscompare is detected (sets MISCOMPARE sense key).
1	The block length read from tape does not equal VL (ILLEGAL LENGTH with NO SENSE set in sense data).

TABLE 4.11.1 SOME VARIABLE MODE CHECK CONDITIONS

Further conditions which will result in CHECK CONDITION status being sent are

b) LUN in Fixed Mode

ILLEGAL REQUEST is set in the sense data, with no tape motion taking place

c) File Mark Detected

NO SENSE and FM are set in the sense data, with the tape left on the EOT side of the filemark.

d) EOT Detected

EOM and MEDIUM ERROR are set in the sense data.

As with the READ command, the SBI will not abort the read in progress, but will carry on and attempt to read/verify the data block before completing.

The SBI will also implement a VERIFY command even if EOT is already set. It is left to the INITIATOR and any LUN safety mechanisms to ensure tape integrity.

e) VL too Large

ILLEGAL REQUEST is set in the sense data, with no tape motion taking place.

4.11.2 Fixed Mode

In this mode the VL bytes of the CDB determine the number of blocks to be read from tape and checked, each of a length equal to the current length. The command completes when all of the blocks have been read and found error free.

The command will complete with CHECK CONDITION set if any of the following occur:

a) CER/HER or Miscompare

BCP state	Check Condition if
0 or 1	CER or HER status detected from the LUN (sets MEDIUM ERROR sense key).
1	A byte miscompare is detected (sets MISCOMPARE sense key).
1	The block length read from tape does not equal VL (ILLEGAL LENGTH with NO SENSE set in sense data).

TABLE 4.11.2 SOME FIXED MODE CHECK CONDITIONS

Further conditions which will result in CHECK CONDITION status being sent are

b) LUN in Variable Mode

ILLEGAL REQUEST is set in the sense data, with no tape motion taking place.

c) Filemark Detected

NO SENSE and FM are set in the sense data, with the tape left on the EOT side of the filemark (see below for VALID).

d) EOT Encountered

MEDIUM ERROR and EOM are set in the sense data.

The command is terminated after the data block during which EOT was detected has been read and checked, with the tape left positioned after this block. As with fixed read, the SBI will implement a fixed verify even if EOT is already set, but will terminate the command after the first block as above.

4.11.3 Variable or Fixed Mode

In Tables 4.11.1 and 4.11.2, when BCP is 1, VALID is set in the sense data and the INFORMATION BYTES set to the difference between the number of blocks specified (VL) and the number successfully verified.

GOOD STATUS is returned in both modes, without further action, if the verification length is 0.

4.12 Recover Buffered Data

bit	7	6	5	4	3	2	1	0
byte 0	0	0	0	1	0	1	0	0
byte 1	LUN			reserved			SILI	FXD
byte 2	Transfer Length (msb)							
byte 3	Transfer Length							
byte 4	Transfer Length (lsb)							
byte 5	0	0	reserved			flag	link	

This command enables the INITIATOR to recover data from the SBI's buffer which has been transferred from the INITIATOR, but not successfully written to tape (normally due to an irrecoverable tape error, or EOT being encountered in BUFFERED mode).

The command operates similarly to a normal READ command except that the associated data is read from the SBI's buffer and not from tape; no tape motion takes place as a result of the command. The data is read from the buffer in the same order as it would have been written to tape.

The various operations associated with Fixed/Variable, File mark Encountered, and Wrong Block Length are handled identically to a normal READ command.

If fixed mode is in operation, and more blocks are requested than are available in the buffer, CHECK CONDITION will returned with Valid, No Sense and EOM set in the sense data; the Information bytes will be set to the Transfer Length minus the actual number of blocks transferred.

If variable mode is in operation, and no data is available in the buffer, CHECK CONDITION will returned with Valid, No Sense and EOM set in the sense data; the Information bytes will be set to the Transfer Length.

A TRANSFER LENGTH of 0 will result in GOOD STATUS being sent, and no data transferred.

4.13 Mode Select

bit	7	6	5	4	3	2	1	0
byte 0	0	0	0	1	0	1	0	1
byte 1	LUN			PF	reserved			SP
byte 2	reserved							
byte 3	reserved							
byte 4	Parameter List Length							
byte 5	0	RLAR	reserved			flag	link	

The MODE SELECT command allows the INITIATOR to set up various control parameters for each individual LUN.

The PF bit determines whether any bytes following the descriptor data are to be interpreted as vendor unique data (PF = 0; ANSI '86 compatible) or as structured Page Data (PF = 1; SCSI-2 compatible).

The SP bit is ignored as the SBI does not support Saveable Pages.

The vendor unique RLAR bit within the CDB is used to cancel any Look-Ahead-Read mode that may have been in action since power-on. Normal READ operation will be resumed unless bit 6, byte 5 of the READ command's CDB is set (on each READ operation).

The PARAMETER LIST LENGTH (PL) contained in the CDB specifies the total number of bytes the INITIATOR has for transfer. This includes the 4-byte header, all Block Descriptor data and all vendor unique or Page data as appropriate. The SBI checks the value of PL, and returns CHECK CONDITION status with ILLEGAL REQUEST set in the sense data if the following conditions are not met:

- i) $PLL \geq 4$
- ii) $PLL - 4 \geq \text{Block Descriptor Length (BDL)}$
- iii) BDL is divisible by 8 if greater than 0.

(Note that the SBI only expects a single BLOCK DESCRIPTOR per LUN; any further descriptors will be ignored).

The expected data format is shown in Table 4.13.

bit	7	6	5	4	3	2	1	0
byte 0	0	0	0	0	0	0	0	0
byte 1	0	0	0	0	0	0	0	0
byte 2	0	Buffered			Speed			
byte 3	Block Descriptor Length							

BLOCK DESCRIPTOR

byte 0	Density Code							
byte 1	0	0	0	0	0	0	0	0
byte 2	0	0	0	0	0	0	0	0
byte 3	0	0	0	0	0	0	0	0
byte 4	0	0	0	0	0	0	0	0
byte 5	Block Length (msb)							
byte 6	Block Length							
byte 7	Block Length (lsb)							

either VENDOR UNIQUE DATA (if PF = 0),

byte 0	(msb)	Write/read Re-try Count	(lsb)
byte 1	(msb)	Read Re-try Count	(lsb)

or PAGE DATA (if PF = 1)

byte 0	0	0	Page Code
byte 1	Page Length (bytes)		
byte 2-n	Page Data		

TABLE 4.13 MODE SELECT DATA FORMAT

The SPEED bits are defined as shown:

0h 1h 2h-Fh	tape unit's high speed tape unit's low speed tape unit's high speed
-------------------	---

The BUFFERED bits are defined as shown:

0h 1h 2-7h	non-buffered buffered reserved
------------------	--------------------------------------

These bits condition the SBI to operate in a certain way when writing data; see the WRITE DATA command for a full description.

The Block Descriptor Length byte determines the amount of Descriptor data transferred; it does not include the vendor unique or Paged data.

If a BLOCK DESCRIPTOR has been received, the SBI will attempt to check/alter the LUN recording density during the MODE SELECT command. If this is not possible for any of the reasons listed below, CHECK CONDITION status is sent and ILLEGAL REQUEST set in the sense data; the entire mode select data is then ignored.

Conditions which will cause command "rejection" are:

- i) LUN not on-line,
- ii) new density requested, LUN not at LDPT,
- iii) new density requested, not available.

The DENSITY bits are defined as shown:

00h 01h 02h 03h 06h 80h	LUN's current density 800 bpi NRZ 1600 bpi PE 6250 bpi GCR 3200 bpi PE 3200 bpi PE
--	---

The BLOCK LENGTH (BL) bytes determine whether the LUN will subsequently operate in FIXED or VARIABLE mode.

If BL = 0, then the LUN is put into VARIABLE mode, such that on any subsequent WRITE / READ commands any block length can be handled (up to the maximum available) as defined by the TRANSFER LENGTH within the CDB.

If BL > 0, then the LUN is placed in FIXED mode with BL being stored as the CURRENT BLOCK LENGTH, such that on all subsequent WRITE/READ operations, the TRANSFER LENGTH will be interpreted as a number of blocks, each of length BL.

The SBI checks the value of BL received to ensure that it is less than or equal to 64K (10000h). If not, CHECK CONDITION status is sent and ILLEGAL REQUEST is set in the sense data; the entire mode select data is then ignored.

Re-try Count

If the PF bit is set to 0, the SBI will accept 2 vendor unique bytes which determine the re-try characteristics when writing and reading. The INITIATOR may send none, one or both of these as required. If neither byte is present, the SBI retains its current re-try settings (Power-on defaults to 16 write attempts and 4 read attempts). If the first byte only is present, this is used as the new re-try count for both write and read. If both bytes are present, then WRITE/READ becomes the new count for WRITE only, and READ the new read count.

Page Data

The SBI also supports the Page Data format of SCSI-2. As shown, this optional data is sent following the Header if the Descriptor Length is 0, or all of the Descriptor data if non-zero.

Each page has the basic form shown in Table 4.13. The Page Code distinguishes one page from another and defines the meaning of the following page data. The Page Length determines the number of bytes within the page (not including the Page Code and Page Length bytes themselves).

a) Page 20h

byte	R	R	1	0	0	0	0	0
byte 1	0	0	0	0	0	0	1	0
byte 3	Write Re-try Count							
byte 4	Read Re-try Count							

The values contained in bytes 3 and 4 replace the default re-try counts (16 and 4 respectively) set following power-on or reset, or any values set by a previous MODE SELECT command. The SBI does not support Saveable Pages.

b) Page 21h

byte 0	R	R	1	0	0	0	0	1
byte 1	0	0	0	0	0	0	1	0
byte 3	LAR termination, Consecutive Filemark Count							
byte 4	reserved							

The value of byte 3 replaces the default count of 2 or any other value previously set. This is then used in any subsequent look-ahead-read operation to define the end of recorded data.

The SBI will return a CHECK CONDITION, with a sense key of Illegal Request, if the Page Length byte for any given Page Code does not match the corresponding page length returned by the SBI during a previous Mode Sense command; thus pages 20h and 21h must have a page length of 2.

c) Other Pages Supported

The Super SCSI SBI supports the ANSI standard Device Configuration (01h) and Read-Write Error Recovery (10h) pages.

Device Configuration Page

Only the SOCF field of byte 8 is supported, allowing the LAR termination count to be set as above.

Read-Write Error Recovery Page

The PER bit of byte 2 is supported along with the Read Retry Count (byte 3) and Write Retry Count (byte 8).

4.14 Reserve/Release LUN

byte	bit	7	6	5	4	3	2	1	0
byte 0	Operation Code*								
byte 1	LUN			3-P	3-P ID			0	
byte 2	reserved								
byte 3	reserved								
byte 4	reserved								
byte 5	0	0	reserved			flag		link	

*Reserve = 16h
Release = 17h

These commands form a complementary pair which allow an INITIATOR to reserve a LUN for its own exclusive use. When a LUN is reserved, a command from any other INITIATOR will result in RESERVATION CONFLICT status being sent, and no action taken (see iii below).

The SBI supports 3rd party reservation/release. This enables an INITIATOR to reserve a LUN for the exclusive use of another INITIATOR whose ID is determined by the 3rd party code within the CDB.

The RELEASE command is essentially a mirror image of the RESERVE command. It releases a reserved LUN for use by all INITIATORS. If a LUN has been reserved directly by an INITIATOR, then it can only be released by that INITIATOR. If reserved 3rd party, then it can only be released 3rd party by the original reserving INITIATOR/3rd party combination.

Notes:

- i) "Illegal" attempts to release a reserved LUN are not considered an error; GOOD STATUS is returned, with no action taken.
- ii) All current reservations are released on receipt of a BUS DEVICE RESET message from any INITIATOR, or a HARD RESET on the SCSI bus.
- iii) 3 commands are "permitted" from other INITIATORS to a reserved LUN, and will attempt to complete. These are RESERVE and RELEASE themselves, and the INQUIRY command.

4.15 Copy

byte	bit	7	6	5	4	3	2	1	0
byte 0		0	0	0	1	1	0	0	0
byte 1		LUN			reserved				PAD
byte 2		Parameter List Length (msb)							
byte 3		Parameter List Length							
byte 4		Parameter List Length (lsb)							
byte 5		0	0	reserved				flag	link

The SBI implements copy function codes 0 and 1 allowing data to be copied between a Direct Access device (Target) on the SCSI bus and any of the LUNS connected to the SBI's tape interface. On receipt of the command, the SBI assumes the role of Copy Manager, becoming an INITIATOR with the Disc as a Target. The SBI's tape units are treated as pseudo-targets with all transfers being carried out internally off the SCSI bus. As a target, the disc drive is allowed to disconnect/reconnect as often as required during the disc transfers, thus maximising the efficiency of the SCSI bus during the COPY operation.

The SBI does not support 3rd party copies and cannot be used as Copy Manager for 2 other Targets on the bus.

The optional PAD bit is implemented in the SBI, and is used in conjunction with each segment descriptor's CAT bit to manage the writing/reading of truncated segments, where the source data does not fit exactly into an integer number of destination blocks (see below for detailed operation).

The Parameter List Length (PLL) specifies the number of bytes to be transferred from the host. This data is made up of a 4 byte Header and up to a maximum of 255 Segment Descriptors where each segment is 12 bytes in length.

The value of PLL must be in the range $0 \leq \text{PLL} \leq 3064$ with $(\text{PLL}-4)$ being divisible by 12. These checks are carried out before any data is transferred, and will result in a CHECK CONDITION with Illegal Request set in the sense data if not met. $\text{PLL} = 0$ will result in immediate command completion with GOOD STATUS.

Segment Descriptor Format

The Header and Descriptor formats are as shown below:

a) Header

byte	bit	7	6	5	4	3	2	1	0	
byte 0		Copy Function Code					Priority			
byte 1		0	0	0	0	0	0	0	0	
byte 2		reserved								
byte 3		reserved								

b) Segment Descriptor

byte 0	Source Address	0	CAT	Source LUN
byte 1	Destn Address	0	0	Destn LUN
byte 2	Streamer Block Length			(msb)
byte 3	Streamer Block Length			(lsb)
byte 4	Disc drive	Number of Blocks		(msb)
byte 5	"	"		
byte 6	"	"		
byte 7	Disc drive	Number of Blocks		(lsb)
byte 8	Disc drive	Logical Block Address		(msb)
byte 9	"	"		
byte 10	"	"		
byte 11	Disc drive	Logical Block Address		(lsb)

The **Copy Function Code** of the Header specifies the type of copy action to be carried out as shown:

Code 00h	Copy data from Direct Access Device to SBI's tape unit.
Code 01h	Copy data from SBI's tape unit to Direct Access Device.

All other codes will result in CHECK CONDITION/Illegal Request being set; no descriptors will be transferred, and Valid/Residue will not be set in the associated sense data.

The **Priority** field is unused, and is ignored by the SBI.

The **Source Address** and **Source LUN** fields specify the SCSI bus ID and logical unit of the device to copy data from for this segment (for a code 1 command, this must correspond to the SBI's SCSI ID and an on-line tape unit).

Similarly, the **Destination Address** and **Destination LUN** fields specify the SCSI bus ID and logical unit to copy data to for this segment (for a code 0 command, this must correspond to the SBI's SCSI ID and an on-line tape unit).

The **Disc drive Logical Block Address** field specifies the starting block address on the unit for the data associated with this segment. Note that the SBI automatically determines whether the disc drive supports the SCSI-2 10-byte WRITE/READ commands. These commands are used, if supported, in order to make available the increased address range (the DPO bit is set to 0, and the FUA bit set to 1).

The **Disc drive Number of Blocks** field specifies the number of logical blocks to be copied to/from the disc for this segment.

The **Streamer Block Length** field specifies the block length to be used when copying data to the tape, and the length to be expected when copying from the tape. CHECK CONDITION will be returned when copying to tape if segment parameters and the CAT/PAD settings do not allow a full block to be written. CHECK CONDITION will also result if a wrong block length is detected when copying from tape.

Other than the normal min/max (1/65,535) limits, there are no restrictions placed on the disc and tape block lengths that can be handled; the tape block length can be set to any size independently of the disc's length. Care should be taken, however, to utilize the PAD and CAT bits as detailed below if the disc and tape block sizes being used result in truncated segments, ie the source data to be transferred does not fit into an exact number of destination blocks.

4.15.1 Truncated Segments, PAD CAT Operation

The PAD and CAT bits are ignored if the disc block length and segment parameters result in exact segments, ie where the source data fits exactly into an integer number of destination blocks. Otherwise, the PAD and CAT bits are used as below to ensure that a complete tape block is always written/read to/from tape.

a) PAD = 0, CAT = 0

Any truncated segment will result in a CHECK CONDITION with Illegal Request, current segment number and Valid/Residue set in the associated sense data. Note that as much source data as possible will have been copied.

b) PAD = 1, CAT = 0

When copying from disc to tape, the SBI will compensate for any shortfall in disc data by padding out the resulting partial tape block with the 00h character, allowing a full block to be written to tape.

When copying from tape to disc, the residue of any tape block left after the specified number of disc blocks have been written will be treated as pad bytes and discarded. The first disc block of the next segment will be copied from a fresh tape block.

Note that this action applies to each segment individually.

c) PAD = X, CAT = 1

As will be seen, the CAT bit has no meaning in the final segment of the command; if the final segment is truncated, then the operation is dependent on the PAD bit as defined above. Otherwise the CAT bit is used to catenate the data of each segment with that of the next.

When copying from disc to tape, a partial tape block left at the end of any segment is filled with the disc data associated with the next segment.

When copying from tape to disc, the residue of any tape block will provide the first of the disc data for the next segment.

Note that this action essentially applies to each segment individually. For the catenation to ripple through all segments, each descriptor must have its CAT bit set.

4.15.2 Typical Copy Operation

Consider copying from disc to tape, the normal sequence of events would be:

- CDB received and validated;
- header received and validated;
- 1st descriptor received and validated;
- SBI disconnects from the Copy host and assumes role of Copy Manager;
- various commands issued to the Target to check compatibility:
TEST UNIT READY, flush unit attention if necessary;
INQUIRY, check the target is a disc drive;
MODE SENSE, determine disc block length for use in residue calculations etc;
- receive and buffer as much disc data as possible;

- start tape write, if last disc data received, check for required PAD/CAT operation and implement;
- reconnect to host, receive next descriptor, disconnect and validate;
- if invalid, complete tape transfer of 1st descriptor, reconnect and return CHECK CONDITION etc;
- if valid, check if compatible with previous descriptor; if compatible, input disc data as buffer space becomes available. If incompatible (different disc ID, tape block length etc) wait for tape write to complete before implementing;
- when final descriptor's associated disc data has been received, wait for tape write to complete (note that SCSI bus is free). Reconnect to host, send status and complete.

4.15.3 Error Conditions

A variety of error conditions can occur, resulting in a CHECK CONDITION being returned. The amount of sense data available via the REQUEST SENSE command also varies depending on the type of error. As well as the normal sense data associated directly with the COPY command, additional sense data may be available from the disc drive and/or the tape unit (for this purpose the tape unit is considered to be a Target responding to the SBI as INITIATOR). See the REQUEST SENSE command description for full details of the additional sense data format.

Notes:

a) The following error conditions will result in:

- i) **Illegal Request sense key.**
- ii) **Valid not set.**
- iii) **No additional disc or tape sense data.**

These error conditions are:

- * parameter List Length in CDB out of range (PLL > 3064);
- * PLL-4 not divisible by 12;
- * copy function code requested is not supported.

b) The following error conditions will result in:

- i) **Illegal Request Sense key.**
- ii) **Current Segment number set in sense data.**
- iii) **Valid set, information bytes set to the number of blocks not successfully copied for this segment.**
- iv) **No additional disc or tape sense data.**

These error conditions are:

- * sequential ID in descriptor does not match SBI's SCSI ID;
- * direct access ID same as sequential ID;
- * disc block address in descriptor out of range;
- * SBI unable to disconnect from host;
- * disc block length greater than 65,535;
- * truncated segment, unable to rectify via PAD/CAT;
- * disc block address goes out of range during copy operation;
- * unexpected disc operation, such as unexpected bus free phase.

c) The following error conditions will result in:

- i) **Copy Abort Sense key**
- ii) **Current Segment number set in sense data**
- iii) **Valid set, information bytes set to the number of blocks not successfully copied for this segment**
- iv) **Additional tape sense data available**
- v) **No additional disc data available**

These error conditions are:

- * tape unit is not on-line/ready
- * disc to tape copy requested, tape unit not write enabled
- * irrecoverable tape error when writing/reading
- * EOT encountered when writing
- * File Mark encountered when reading
- * wrong block length detected when reading
- * unusual tape status, eg gone off-line, command aborted, blank tape detected

d) The following error conditions will result in:

- i) **Copy Abort Sense key**
- ii) **Current Segment Number set in sense data**
- iii) **Valid set, information bytes set to the number of blocks not successfully copied for this segment**
- iv) **Additional disc sense data available**
- v) **No additional tape sense available**

These error conditions are:

- * target not a direct access device
- * disc not ready
- * CHECK CONDITION returned by disc to SBI during command operation.

4.16 Erase

bit	7	6	5	4	3	2	1	0
byte 0	0	0	0	1	1	0	0	1
byte 1	LUN			reserved			IMM	LNG
byte 2	reserved							
byte 3	reserved							
byte 4	reserved							
byte 5	0	0	reserved			flag		link

The ERASE command causes part or all of the tape to be erased from the current tape position. The length of tape to be erased is determined by the Long Gap (LNG) bit of the CDB as shown:

LNG	Action
0	erase next 3 inches
1	erase tape to EOT

Use of the IMM bit is not fully SCSI-2 compatible; it is ignored if a long ERASE is requested and/or non-buffered mode is selected. If a buffered mode WRITE is ongoing, however, a short ERASE with the IMM bit set will result in the ERASE command being stored for execution as appropriate and GOOD STATUS being returned immediately.

CHECK CONDITION status will be sent and FILE PROTECT set in the sense data if the LUN is write protected; no tape motion will take place.

CHECK CONDITION status will be sent and EOM/NO SENSE set in the sense data if EOT is encountered during a short erase. The short erase will also be implemented even if EOT is already set when the command is given. It is left to the INITIATOR and any LUN safety mechanisms to ensure tape integrity.

CHECK CONDITION status will be sent and MEDIUM ERROR/ERASE FAILURE set if any read back data is detected during the ERASE operation.

4.17 Mode Sense

bit	7	6	5	4	3	2	1	0
byte 0	0	0	0	1	1	0	1	0
byte 1	LUN			R	DBD	reserved		
byte 2	PCF		Page Code					
byte 3	reserved							
byte 4	Allocation Length							
byte 5	0	0	reserved				flag	link

The MODE SENSE command enables an INITIATOR to determine various parameters about the SBI/LUNS. Although complementary to the MODE SELECT command, it reflects the actual status of the LUN, and not necessarily what may have been requested by a previous SELECT command.

The DBD bit is ignored by the SBI. The amount of returned data is determined by the Allocation and Available lengths as described below.

The PCF (Page Control Field)/Page Code byte determines the Paged Data to be returned (see later for detailed description).

As the LUN's recording mode has to be checked, CHECK CONDITION status is sent and LUN NOT READY set in the sense data if the LUN is not on-line.

The Allocation Length of the CDB is the maximum number of bytes the INITIATOR has allowed for returned data. Allocation Length bytes are returned if the allocation length is less than the available length (Sense Data Length + 1), otherwise the Available Length is sent.

The data available for return is shown in Table 4.18.

bit	7	6	5	4	3	2	1	0
byte								
byte 0	Sense Data Length							
byte 1	Medium Type							
byte 2	WP	Buffered			Speed			
byte 3	0	0	0	0	1	0	0	0

BLOCK DESCRIPTOR

byte 0	Density Code							
byte 1	0	0	0	0	0	0	0	0
byte 2	0	0	0	0	0	0	0	0
byte 3	0	0	0	0	0	0	0	0
byte 4	0	0	0	0	0	0	0	0
byte 5	Block Length (msb)							
byte 6	Block Length							
byte 7	Block Length (lsb)							

either VENDOR UNIQUE DATA (if Non-page Mode),

byte 0	(msb)	Write Re-try Count	(lsb)
byte 1	(msb)	Read Re-try Count	(lsb)

or PAGE DATA (if Page Mode)

byte 0	0	R	Page Code
byte 1	Page Length		
byte 2-n	Page Data (dependent on page code)		

TABLE 4.18 MODE SENSE RETURNED DATA

Sense Data Length

This is the number of bytes available for return (not including the Sense Data Length byte itself).

Medium Type

Always set to 0

WP

Reflects the write enable status of the LUN as shown:

WP	Status
0	write enabled
1	write protected

Buffered Mode

These three bits reflect the current BUFFERED status of the LUN as shown:

00h	non-buffered
01h	buffered

Speed

The SBI defaults to high speed following power-on (or SCSI bus RESET), otherwise these bits reflect any previous SELECT command. They determine the LUN speed that will be requested on any subsequent commands as shown:

0h	high speed
1h	low speed
2h	high speed

Density Code

01h	LUN set for 800 bpi NRZ
02h	LUN set for 1600 bpi PE
03h	LUN set for 6250 bpi GCR
06h	LUN set for 3200 bpi PE

Block Length

The BLOCK LENGTH bytes are set to 0 following power-on (or SCSI bus RESET) to indicate that the LUN is in VARIABLE mode. Otherwise the BLOCK LENGTH bytes reflect the equivalent bytes of any previous SELECT command. If their value is greater than 0, then the LUN is in FIXED mode with BL equal to the CURRENT BLOCK LENGTH used in any WRITE/READ operations.

Vendor Unique Re-try Bytes

These bytes are sent as vendor unique data if Non-page mode is set in the CDB. They reflect the number of re-tries that will be attempted in WRITE and READ, respectively, before declaring an irrecoverable error.

Page Data Option

The SBI supports the Page Data format of SCSI-2. With Page mode set in the CDB, the INITIATOR can use the MODE SENSE command to determine those pages supported by the Target, and their characteristics, prior to any changes via the MODE SELECT command.

The data returned to the INITIATOR depends on the Page Code and the Page Control Field (PCF) of the CDB. The PCF bits determine the meaning of the page data as shown:

a) PCF = 00, Page Code = 20h

byte 0	0	R	1	0	0	0	0	0
byte 1	0	0	0	0	0	0	1	0
byte 2	Current Value of write re-try							
byte 3	Current Value of read re-try							

b) PCF = 00, Page Code = 21h

byte 0	0	R	1	0	0	0	0	1
byte 1	0	0	0	0	0	0	1	0
byte 2	Current Value of LAR FM count							
byte 3	reserved							

The Current Values will either be the default values set following power-on or reset, or values set by a previous MODE SELECT command.

c) PCF = 01, Page Code = 20h

byte 0	0	R	1	0	0	0	0	0
byte 1	0	0	0	0	0	0	1	0
byte 2	1	1	1	1	1	1	1	1
byte 3	1	1	1	1	1	1	1	1

d) PCF = 01, Page Code = 21h

byte 0	0	R	1	0	0	0	0	1
byte 1	0	0	0	0	0	0	1	0
byte 2	1	1	1	1	1	1	1	1
byte 3	0	0	0	0	0	0	0	0

The PCF code of 01 requests those bits of the page data that can be altered (by a MODE SELECT) to be set to 1.

e) PCF = 10, Page Code = 20h

byte 0	0	R	1	0	0	0	0	0
byte 1	0	0	0	0	0	0	1	0
byte 2	Write re-try Default Value (10h)							
byte 3	Read re-try Default Value (04h)							

f) PCF = 10, Page Code = 21h

byte 0	0	R	1	0	0	0	0	1
byte 1	0	0	0	0	0	0	1	0
byte 2	LAR FM termination default value (02h)							
byte 3	reserved							

The PCF code of 10 requests that the default values of all parameters are returned.

g) PCF = 11, Page Code = 20h, 21h or 3Fh

byte 0	0	R	Page Code						
byte 1	0	0	0	0	0	0	0	0	0

The PCF code of 11 requests that saved values are returned. The SBI does not support Saved parameters, hence returns a page length of 0.

All page codes other than 0h, 20h, 21h and 3Fh will return 2 bytes as shown:

byte 0	0	R	Page Code						
byte 1	0	0	0	0	0	0	0	0	0

A page code of 0 will result in the SBI setting a CHECK CONDITION with a sense key of ILLEGAL REQUEST.

h) Other Pages Supported

The Super SCSI SBI supports the ANSI standard Device Configuration (01h) and Read-Write Error Recovery (10h) pages

Device Configuration Page

Only the SOCF field of byte 8 is supported, allowing the LAR termination count to be set as above.

Read-Write Error Recovery Page

The PER bit of byte 2 is supported along with the Read Retry Count (byte 3) and Write Retry Count (byte 8).

4.18 Unload

byte	bit	7	6	5	4	3	2	1	0
byte 0		0	0	0	1	1	0	1	1
byte 1		LUN			reserved				0
byte 2		reserved							
byte 3		reserved							
byte 4		reserved					0	LD	
byte 5		0	0	reserved				flag	link

The UNLOAD command is selected when the LD bit of the CDB is set to 0 (the SBI does not support the LOAD command as such, but will return GOOD STATUS if received when the LUN is already on-line and hence loaded; CHECK CONDITION will be returned, however, with Not Ready sense key set if the LUN is not on-line).

The UNLOAD command causes the SBI to issue an OFF-LINE command to the LUN, with GOOD STATUS being returned when the LUN is detected off-line. If the LUN is not detected off-line within a short timeout (100 μ s), CHECK CONDITION status is sent with ABORT set in the sense data.

Note that it is left to the OPERATOR to ensure that the LUN's OFF-LINE/REMOTE UNLOAD option is set.

4.19 Receive Diagnostic Results

bit	7	6	5	4	3	2	1	0
byte 0	0	0	0	1	1	1	0	0
byte 1	LUN			reserved				
byte 2	reserved							
byte 3	Allocation Length (msb)							
byte 4	Allocation Length (lsb)							
byte 5	0	0	reserved			flag	link	

This essentially vendor unique command is used by the SBI to provide results data to the INITIATOR following completion of a SEND DIAGNOSTIC command. The SBI provides different results formats depending on the setting of the PF (Page Format) bit in the SEND DIAGNOSTIC CDB.

The ALLOCATION LENGTH (AL) bytes of the CDB specify the number of bytes the INITIATOR has allocated for returned data. The data available is determined by the format in operation and is detailed below.

Page Code 00h

Page Code 00h returns the data shown in Table 4.19(a) below. This is essentially a list of the page codes supported by the SBI.

bit	7	6	5	4	3	2	1	0
byte 0	Page Code					Set to 00h		
byte 1	0	0	0	0	0	0	0	0
byte 2	Page Parameter Length					Set to 00h		
byte 3	Page Parameter Length					Set to 04h		
byte 4	First Page Code Supported					Set to 00h		
byte 5	Second Page Code Supported					Set to 80h		
byte 6	Third Page Code Supported					Set to 81h		
byte 7	Fourth Page Code Supported					Set to 82h		

TABLE 4.19(a)

Page Code 80h

Page Code 80h can be used with the 9914 to access the condensed status bytes, returning the data shown in Table 4.19(b).

bit/data byte	7	6	5	4	3	2	1	0	
byte 0	Page Code						Set to 80h		
byte 1	0	0	0	0	0	0	0	0	
byte 2	Page Parameter Length						Set to 00h		
byte 3	Page Parameter Length						Set to 0Bh		
byte 4	FPTD	RWDG	ONL	RDY	EOT	LDPT	DBY	FBY	
byte 5	0	0	NRZ	FM	CER	HER	0	0	
byte 6	Condensed Status Byte C1								
byte 7	Condensed Status Byte C2								
byte 8	Condensed Status Byte C3								
byte 9	Condensed Status Byte C4								
byte 10	Condensed Status Byte C5								
byte 11	Condensed Status Byte C6								
byte 12	Condensed Status Byte C7								
byte 13	Condensed Status Byte C8								
byte 14	Condensed Status Byte C9								
The condensed status bytes are outlined in the 9914 User/Diagnostic Manual, and described fully in the 9914 Servicing Manual									

TABLE 4.19(b)

Page Code 81h

Page Code 81h returns the data shown in Table 4.19(c).

bit/data byte	7	6	5	4	3	2	1	0	
byte 0	Page Code						Set to 81h		
byte 1	0	0	0	0	0	0	0	0	
byte 2	Page Parameter Length						Set to 00h		
byte 3	Page Parameter Length						Set to 14h*		
byte 4	FPTD	RWDG	ONL	RDY	EOT	LDPT	DBY	FBY	
byte 5	0	0	NRZ	FM	CER	HER	0	0	
byte 6	Full Status Byte						F1		
byte 7	Full Status Byte						F2		
/									
byte 22	Full Status Byte						F17		
byte 23	Full Status Byte						F18		
The full status bytes are outlined in the 9914 User/Diagnostic Manual, and described fully in the 9914 Servicing Manual									

*Note: value is dynamically compiled to match the available number of status bytes.

TABLE 4.19(c)

Page Code 82h

Page Code 82h returns the data shown in Table 4.19(d). This page is intended to provide 9914 test results of special data diagnostic programs.

bit/data byte	7	6	5	4	3	2	1	0	
byte 0	Page Code						Set to 82h		
byte 1	0	0	0	0	0	0	0	0	
byte 2	Page Parameter Length (ms)						Set to 00h		
byte 3	Page Parameter Length (ls)						Set to 0Ah		
byte 4	FPTD	RWDG	ONL	RDY	EOT	LDPT	DBY	FBY	
byte 5	0	0	NRZ	FM	CER	HER	0	0	
byte 6	Program Number								
byte 7	Failure Code								
byte 8	reserved					DDP	ADP	P-AMP	
byte 9	Status after write to tape								
byte 10	Status after ADP E-E write								
byte 11	Status after DDP E-E write								
byte 12	Diagnostic Firmware Number								
byte 13	Diagnostic Firmware Revision								
The diagnostic programs are outlined in the 9914 User/Diagnostic Manual, and described fully in the 9914 Servicing Manual									

TABLE 4.19(d)

Notes to Table 4.19(d):

- The format of bytes 9, 10, & 11 is as follows:

byte 9	0	0	0	0	C	H	0	0
--------	---	---	---	---	---	---	---	---

H = set if soft error (CER) occurred,
 C = set if hard error (HER) occurred.

- Byte 12 number must equal the number of the tape unit's Data Diagnostic firmware in order to ensure compatibility, byte 13 is a diagnostic code revision within this number.

4.20 Send Diagnostic

bit	7	6	5	4	3	2	1	0
byte 0	0	0	0	1	1	1	0	1
byte 1	LUN			PF	0	S-TST	D-OFL	U-OFL
byte 2	reserved							
byte 3	Parameter List Length (msb)							
byte 4	Parameter List Length (lsb)							
byte 5	0	0	reserved				flag	link

This essentially vendor unique command allows the INITIATOR to make use of the SBI's self-test routine and the LUN's interface diagnostic capability (see Appendix C for relevant LUN User/Diagnostic or Servicing Manuals).

The PF bit determines whether the SBI is to operate in the SCSI-2 Page Format mode (PF = 1) or in its Non-page mode (PF = 0).

If either of the 2 qualifier bits (D-OFL or U-OFL) is not set, the SBI will "refuse" to carry out the command, including SELF-TEST. CHECK CONDITION status will be sent with ILLEGAL REQUEST set in the sense data.

Self Test

The SBI carries out its self-test routine when the S-TST bit of the CDB is set. The SBI checks the integrity of its data buffer to ensure that the number of contiguous good bytes is still equal to the value contained in the INQUIRY data set following power-on.

CHECK CONDITION status is sent with HARDWARE ERROR set in the sense data if an error is detected. The inquiry data is updated to reflect the new value determined during the self-test.

Note that if S-TST is set, the PARAMETER LIST LENGTH is ignored; no bytes will be transferred and no further checks carried out.

PF = 0

A PF bit of 0 indicates that the SBI is to operate in its non-page mode, which is not applicable to the 9914 Streamer.

PF = 1

A PF bit of 1 indicates that the SBI is to operate in the SCSI-2 page format. Each page format has a particular streamer application, those applicable to the 9914 are shown in Table 4.20.

The parameter list applicable to each page is shown in Appendix E.

The Page Code field specifies the page type and format for this page. The SBI currently supports Page Code 00h and the vendor unique Page Codes 80h, 81h and 82h.

Page Code 00h is used to instruct the SBI to supply a list of all the page codes supported. This data is returned during a subsequent RECEIVE DIAGNOSTIC RESULTS command. No diagnostic checks are actually carried out in response to this page.

The Page Parameter Length bytes define the number of bytes that will be transferred in the Page Parameter field (set to 0 for Page Code 00h).

For Page Codes 80h, 81h and 82h, the Page Parameter field essentially contains the sequence of diagnostic tests that the INITIATOR wants to be carried out.

mode	PF = 0	PF = 1			
page	-	00h	80h	81h	82h
9913	na	*1	*2	full	full
9914	na	*1	*2	full	full
na = not applicable. *1 = page codes supported only. *2 = condensed status bytes only.					

TABLE 4.20 PAGE CODE APPLICATIONS

Page Code 80h

Page Code 80h is not intended for use with the 9914 Streamer, but can be used to access the condensed status byte block; Appendix E shows sample Parameter lists.

Page Code 81h

Page Code 81h is intended for use with the 9914 Streamer, used as detailed in Appendix E to run LUN interface diagnostic programs.

It can be also be used (with a parameter list length of 0, see Appendix E) in order to obtain the LUN's Full Status Byte block with a subsequent RECEIVE DIAGNOSTIC RESULTS command.

Page Code 82h

Page Code 82h is specifically for use with the 9914 Streamer, making available various data checking and fault isolation programs. Parameter lists for various diagnostic programs are shown in Appendix E.

4.21 Locate

byte	bit	7	6	5	4	3	2	1	0	
byte 0		0	0	1	0	1	0	1	1	
byte 1		Logical Unit No.			reserved		0	0	0	
byte 2		reserved								
byte 3		reserved								
byte 4		Logical Block Address						(msb)		
byte 5		"	"	"						
byte 6		Logical Block Address						(lsb)		
byte 7		reserved								
byte 8		Partition Number								
byte 9		0	IER	reserved			flag	link		

This group one command causes the SBI to locate the specified logical block and leave the tape positioned on the BOT side of it. Any 'active' unwritten data in the SBI's buffer will be written to tape before implementing the locate operation. However, any data left in the buffer as a result of some write CHECK CONDITION, eg irrecoverable error, EOT encountered, etc., will not be force written to tape.

The **Partition** byte of the CDB is ignored by the SBI which treats the entire tape as a single partition. All blocks are numbered (addressed) sequentially - 1, 2, 3 - from BOT, with no distinction being made between data blocks and file marks.

A logical block address of 0 will result in the tape being rewound to BOT.

Several abnormal conditions which can occur are described next.

a) Irrecoverable Read Error Detected

Check condition will be returned with Valid and Medium Error set in the sense data; the Information bytes will be set to the difference between the Requested Address and the Current Block Address (tape positioned on the EOT side of the faulty block).

Note that the vendor unique Ignore Error (IER) bit of the Control byte, when set to 1, causes the SBI to ignore any read error indications from the tape while locating the specified block.

b) Physical End of Medium encountered

Tape position is assumed to have been lost if the physical end of the medium is encountered (not the reflective marker). Check condition will be returned with EOM and No Sense set in the sense data. Valid will not be set.

Note that the SBI will ignore the EOT marker when trying to locate the requested block.

c) LUN detected off-line

Tape position is assumed to be lost if the LUN is detected off-line. CHECK CONDITION will be returned with Not Ready set in the sense data. Valid will not be set.

Note that in b) and c) above, VALID not being set should be taken as confirmation that tape position has been lost.

4.22 Write Buffer

bit	7	6	5	4	3	2	1	0
byte 0	0	0	1	1	1	0	1	1
byte 1	0	0	0	0	0	Mode		
byte 2	0	0	0	0	0	0	0	0
byte 3	Start Address Offset (msb)							
byte 4	Start Address Offset							
byte 5	Start Address Offset (lsb)							
byte 6	0	0	0	0	0	0	0	0
byte 7	Byte Transfer Length (msb)							
byte 8	Byte Transfer Length (lsb)							
byte 9	0	0	reserved			flag	link	

This group 1 command can be used in conjunction with the complementary READ BUFFER command to check the integrity of the SBI's data buffer, allowing any buffer location to be written/read. The following command usage is recommended:

- issue READ BUFFER command (with offset set as required) to determine the Available Length, ie the number of buffer locations that can be written from any given offset;
- reserve the target if multi-initiator environment;
- issue WRITE BUFFER command to store test pattern;
- issue READ BUFFER command to retrieve stored data for comparison;
- repeat with different offsets/transfer lengths until full buffer checked. Release target.

The Mode field of byte 1 of the CDB determines the specific mode of operation of the command.

3 different modes are available, as described next.

Mode 000

This mode is incorporated for compatibility with older host systems, and only allows access to the first 65,531 bytes of the data buffer. The offset bytes 3, 4 and 5 of the CDB are ignored in this mode.

The BYTE TRANSFER LENGTH is the number of bytes that the host intends to transfer to the SBI. This includes 4 bytes of header information plus the test data pattern. Thus a transfer length of 65,535 would consist of 4 bytes of header information and 65,531 bytes of test data.

Note that the header bytes are stored separately and are not written into the data buffer under test. They are also essentially redundant, and can be of any value.

Mode 001

This mode is similar to the above, but utilizes the offset bytes of the CDB to allow access to the entire data buffer. Bytes 3, 4 and 5 are interpreted as a logical offset from the first buffer location. Note that this is an offset value and not an absolute address. The SBI takes care of all physical address translation.

The byte transfer length and header are as defined above for mode 000.

Mode 010

This mode is identical to mode 001 above, except that the header bytes are not used. The Byte Transfer Length is now the number of data bytes that the host wants to write into the SBI's buffer.

All Modes

CHECK CONDITION/ILLEGAL REQUEST will be returned and no data transferred if

- i) the BYTE TRANSFER LENGTH is greater than the AVAILABLE LENGTH (reported by the READ BUFFER command);
the SBI automatically takes account of the header length in modes 000 and 001;
- ii) mode field values other than 000, 001 and 010.

Example

Table 4.22 shows the parameters associated with a complete 256K buffer check using mode 001 with a basic transfer length of 65,535 and a sequence of 4 read/write/reads.

Transfer length	Offset	Available length	
FFFFh	000000h	FFFBh	read
	000000h		write
	000000h	FFFBh	read
FFFFh	00FFFBh	FFFBh	read
	00FFFBh		write
	00FFFBh	FFFBh	read
FFFFh	01FFF6h	FFFBh	read
	01FFF6h		write
	01FFF6h	FFFBh	read
EC34h	02FFF1h	EC30h	read
	02FFF1h		write
	02FFF1h	EC30h	read
Notes:			
i) The BYTE TRANSFER LENGTH can be less than the AVAILABLE LENGTH if necessary.			
ii) A transfer length of zero will result in no data being transferred and GOOD STATUS being returned.			

TABLE 4.22 EXAMPLE OF BUFFER CHECK

4.23 Read Buffer

bit	7	6	5	4	3	2	1	0
byte 0	0	0	1	1	1	1	0	0
byte 1	0	0	0	0	0	Mode		
byte 2	0	0	0	0	0	0	0	0
byte 3	Start Address Offset (msb)							
byte 4	Start Address Offset							
byte 5	Start Address Offset (lsb)							
byte 6	0	0	0	0	0	0	0	0
byte 7	Allocation Length (msb)							
byte 8	Allocation Length (lsb)							
byte 9	0	0	reserved			flag	link	

This Group 1 command can be used in conjunction with the complementary WRITE BUFFER command to check the integrity of the SBI's data buffer, allowing any buffer location to be written/read. The following command usage is recommended:

- issue READ BUFFER command (with offset set as required) to determine the Available Length, ie the number of buffer locations that can be written from any given offset;
- reserve the target if multi-initiator environment;
- issue WRITE BUFFER command to store test pattern;
- issue READ BUFFER command to retrieve stored data for comparison;
- repeat with different offsets/transfer lengths until full buffer checked. Release target.

The Mode field of byte 1 determines whether bytes 3, 4 and 5 of the CDB are valid. When set to 000, these bytes are ignored, allowing access to the first 65,531 buffer locations only. When set to 001 (vendor unique) or 010 (write data), bytes 3, 4 and 5 are interpreted as a logical offset from the first buffer location, thus allowing access to the entire data buffer.

The ALLOCATION LENGTH is the maximum number of bytes that the host has allocated for returned data from the SBI. This includes 4 bytes of header information plus the test data pattern in modes 000 and 001, and the data pattern only in mode 010. Thus an allocation length of 65,535 would consist of 4 bytes of header information and 65,531 bytes of test data in modes 000 and 001, and 65,535 bytes of test data in mode 010.

Note that the header bytes, if available, are assembled separately and are not part of the test data stored in the buffer under test

CHECK CONDITION/ILLEGAL REQUEST will be returned, and no data transferred, if:

- i) the Allocation Length is less than 4 in modes 000 and 001.
- ii) Mode field values other than 000, 001 and 010.

The header information of modes 000 and 001 is as shown in Table 4.23.

bit	7	6	5	4	3	2	1	0
byte 0	0	0	0	0	0	0	0	0
byte 1	0	0	0	0	0	0	0	0
byte 2	Available Length (msb)							
byte 3	Available Length (lsb)							

TABLE 4.23

The AVAILABLE LENGTH is the number of buffer locations available to be written into from any given offset (see below) by the WRITE BUFFER command. Note that the host command sequence/parameters will determine whether all or any of these locations contain meaningful data during a particular READ command execution.

Notes:

- i) If the Allocation Length is less than the Available Length (+4 in modes 000 and 001), the Allocation Length only byte will be sent.
- ii) If the Allocation Length is greater than the Available Length (+4 in modes 000 and 001), then Available Length (+4 in modes 000 and 001) number of bytes will be sent.
- iii) An Available Length of 0 is possible, and basically indicates that the Offset used in the CDB has gone beyond the final buffer location. This will not cause a CHECK CONDITION.
- iv) An Allocation length of zero will result in no data being transferred and GOOD STATUS being returned.

4.24 Read Position

byte	bit	7	6	5	4	3	2	1	0
byte 0		0	0	1	1	0	1	0	0
byte 1		Logical Unit No.				reserved			0
byte 2		reserved							
byte 3		reserved							
byte 4		reserved							
byte 5		reserved							
byte 6		reserved							
byte 7		reserved							
byte 8		reserved							
byte 9		0	0	reserved			flag	link	

This Group 1 command returns the current position of data blocks in the buffer, and the position of the medium. No tape motion takes place.

The SBI treats the entire tape as a single partition. All blocks are numbered (addressed) sequentially - 1, 2, 3 - from BOT, with no distinction being made between data blocks and file marks.

The SBI returns the data shown in Table 4.24.

byte	bit	7	6	5	4	3	2	1	0	
byte 0		BOP	EOP	0	0	0	BPU	0	0	
byte 1		0	0	0	0	0	0	0	0	
byte 2		reserved								
byte 3		reserved								
byte 4		First Block Location						(msb)		
byte 5		"	"	"						
byte 6		"	"	"						
byte 7		First Block Location						(lsb)		
byte 8		Last Block Location						(msb)		
byte 9		"	"	"						
byte 10		"	"	"						
byte 11		Last Block Location						(lsb)		
byte 12		reserved								
byte 13		Number of Blocks in Buffer						(msb)		
byte 14		"	"	"						
byte 15		Number of Blocks in Buffer						(lsb)		
byte 16		Number of Bytes in Buffer						(msb)		
byte 17		"	"	"						
byte 18		"	"	"						
byte 19		Number of Bytes in Buffer						(lsb)		

TABLE 4.24

A **Block Position Unknown (BPU)** bit of one indicates that no valid position information is available and that the remaining table data should be ignored. This could occur, for example, if the SBI receives a Hard Reset from the INITIATOR when not at BOT.

A **BOP** bit of 1 indicates that the tape is positioned at the BOT marker.

An **EOP** bit of 1 indicates that the tape is positioned after the EOT marker.

The **First Block Location** field indicates the position of the next data block to be transferred between the INITIATOR and the SBI if the previous command was a READ (forward) or WRITE command. It indicates the position of the last block transferred to the INITIATOR if the last command was a READ REVERSE.

The **Last Block Location** field indicates the position of the next block to be transferred between the buffer and the medium if the previous command was a READ (forward) or WRITE command. It indicates the last block to be read into the SBI's buffer if the previous command was a READ REVERSE.

The **Number of Blocks** field indicates the number of blocks (data and filemarks) stored in the SBI's buffer waiting to be written to tape or transferred to the host (look ahead READ).

The **Number of Bytes** field is the total number of data bytes in the buffer waiting to be written to tape or transferred to the host (look-ahead READ).

To illustrate the above, consider the buffer to contain the first 2 data blocks after BOT:

Previous Command	First Block Location	Last Block
Write	3	1
Read forward	1	3
Read reverse	3	1

4.25 Long Block Read

bit	7	6	5	4	3	2	1	0
byte 0	0	0	0	0	0	1	1	1
byte 1	LUN			reserved			SILI	0
byte 2	0	0	0	0	0	0	0	0
byte 3	Transfer Length (msb)							
byte 4	Transfer Length (lsb)							
byte 5	LFG	FFG	reserved				flag	link

This special vendor unique READ command allows any block length to be read from tape. It is intended for use by those hosts which cannot transfer more than 64 Kbytes per SCSI command, and hence cannot make use of the normal READ command's long block action.

Strictly speaking, each block of data is read from tape by a sequence of the above commands, utilising the first (FFG) and last (LFG) flags to control the overall action.

Consider the host reading a 1 Mbyte (1,048,576 bytes) data block:

- the host issues the first command of the sequence with the FFG bit set in the CDB, and the transfer length set to 65,534 (except for any final residue, hardware considerations require the data to be handled in segments divisible by 7, hence 65,534).
- on receipt, the SBI issues a READ command to the tape and activates its special long block algorithm to handle the continuous data stream, splitting it into 65,534 byte segments and providing temporary storage before transferring to the host.
- when available, the first segment is transferred to the host; status and command complete are then sent as per normal SCSI commands.
- the host will then issue 15 further commands (65,534 byte transfer length) with the LFG and FFG flags cleared.
- the host will then issue a 32-byte READ with the LFG flag set. After transferring the data to the host, the SBI sends STATUS and COMMAND COMPLETE as normal, terminates the sequence internally and prepares to receive the 'next' SCSI command.

It should be noted that tape data is transferred to the SBI across a synchronous interface which cannot be halted or delayed without incurring loss of data. Consequently the host must be capable of accepting data from the SBI at a minimum rate equivalent to the tape data rate. This is an averaged rate and must include all overheads associated with selection, CDB transfer, status phase etc. The necessary minimum performance is determined by the tape unit's speed and density. LONG BLOCK READs are done at the speed set by the MODE SELECT command, giving the minimum required host data rates shown next.

Density & Speed			Data Rate
800 bpi NRZ	Low speed	42 ips	34 kB/s
	High speed	125 ips	100 kB/s
1600 bpi PE	Low speed	42 ips	67 kB/s
	High speed	125 ips	200 kB/s
6250 bpi GCR	Low speed	42 ips	260 kB/s
	High speed	125 ips	781 kB/s

TABLE 4.25 MINIMUM HOST DATA RATES

Error checking and reporting is similar to the normal READ command. If a tape error occurs, the error is reported as normal, and the tape left positioned in the gap following the faulty block; no automatic retrying takes place.

The following abnormal conditions can occur and will result in CHECK CONDITION being sent:

i) Tape Error

MEDIUM ERROR is set in the sense data, no re-trying takes place.

ii) EOT Encountered

The EOM bit is set in the sense data if the command completes beyond the tape's EOT marker. If the sense key is set to NO SENSE, then a complete data block was read as normal. If the sense key is set to ABORT (0B), then it should be assumed that an incomplete block was encountered, see below.

iii) Physical Limit Encountered

The 9914 normally implements a tape motion inhibit point at approximately 12 feet beyond the EOT marker. All forward commands are terminated by the streamer at this point, and no further forward motion commands are accepted.

If the LONG BLOCK READ encounters this situation, the read is abandoned and Check Condition sent with the EOM bit and the ABORT (0Bh) sense key set.

If the streamer configuration is set to override the tape motion inhibit point, the greatest care must be taken to ensure the physical end of tape is never reached.

iv) Wrong Block Length Encountered

Block length checking, length error reporting, residue calculation etc, operate as per normal READ.

Special operational notes:

- * FFG flag set to indicate the first command in the string.
- * LFG flag set to indicate the last command in the string.
- * the FFG and LFG flags may be set together, allowing block lengths less than 64K to be read.
- * all commands except the last in the string (LFG set) must have a transfer length of 65,534 (or data will be lost at the end of each segment).
- * if a CHECK CONDITION is returned for any command, the host must assume the end of the string whether it had the LFG flag set or not.
- * as with normal READ, only the lesser of the (total) transfer length requested and the block length read will be transferred to the host.

4.26 Log Sense

bit	7	6	5	4	3	2	1	0
byte 0	0	1	0	0	1	1	0	1
byte 1	LUN			Reserved			PPC	SP
byte 2	PC		Page Code					
byte 3	Reserved							
byte 4	Reserved							
byte 5	msb							
byte 6	Parameter Pointer						lsb	
byte 7	msb							
byte 8	Allocation Length						lsb	
byte 9	Control							

The SCSI-2 LOG SENSE command is intended for use, with the LOG SELECT command, in managing statistical information pertaining to the Target/LUN.

The command conforms to the ANSI specification and implements a subset of it; viz two pages (2 and 3) and two parameter codes (3 and 6) within each page, giving the 4 error logs:

Error Type	Counter
write corrected	16-bit
write uncorrected	8-bit
read corrected	16-bit
read uncorrected	8-bit

The log values are retained until cleared/set by the LOG SELECT command, or cleared by a SCSI reset or power cycle; note that they cannot be 'saved' to the 9914's non-volatile memory, and are not retained through a power cycle.

As currently implemented, the commands should only be issued at the start/end of a write or read sequence as they will cause the buffer to flush in a write operation and cancel the LAR action on read, resulting in slower operation.

The term 'corrected error' means that the drive/SCSI recovered the error by means of its normal retry sequence; it has no direct correlation with the CER error signal on the internal Pertec interface. The corrected error log is only incremented once for any given data block, irrespective of the number of physical retries required.

The term 'uncorrected error' means that the drive/SCSI was unable to recover the error after its normal retry sequence; it has no direct correlation with the HER signal on the internal Pertec interface.

The following paragraphs are intended as a quick guide on how the LOG SENSE command can be used. They should be read/used in conjunction with the SCSI-2 description of the commands and associated parameters. It is considered that the command is much easier to use in practice than might appear likely from the formal description in the SCSI-2 specification.

4.26.1 Using 'Log Sense'

This command is used mainly to read the current value of a particular error log. It can also be used to read the current threshold value (as set by a previous LOG SELECT command) and the default threshold setting (as set at power-on).

A single page is accessed by each command; both parameter codes associated with each page can be accessed together, or individually.

The DU, DS, TSD, ETC, TMC and LP bits of the control byte are implemented as defined in SCSI-2, but saving is not supported.

The following command descriptor block examples will carry out the function indicated:

- i) determine the current write corrected error count (page 2, parameter 3)
4D, 00, 42, 00, 00, 00, 03, 00, 0A, 00
- ii) determine the current write uncorrected error count (page 2, parameter 6)
4D, 00, 42, 00, 00, 00, 06, 00, 09, 00
- iii) determine the current read corrected error count (page 3, parameter 3)
4D, 00, 43, 00, 00, 00, 03, 00, 0A, 00
- iv) determine the current read uncorrected error count (page 3, parameter 6)
4D, 00, 43, 00, 00, 00, 06, 00, 09, 00

4.27 Log Select

byte	bit	7	6	5	4	3	2	1	0
byte 0		0	1	0	0	1	1	0	0
byte 1		LUN			Reserved			PCR	SP
byte 2		PC		Reserved					
byte 3		Reserved							
byte 4		Reserved							
byte 5		Reserved							
byte 6		Reserved							
byte 7	msb								
byte 8		Parameter List Length							
byte 9		Control							
								lsb	

The SCSI-2 LOG SELECT command is intended for use, with the LOG SENSE command, in managing statistical information pertaining to the Target/LUN.

The command conforms to the ANSI specification and implements a subset of it; viz two pages (2 and 3) and two parameter codes (3 and 6) within each page, giving the 4 error logs:

Error Type	Counter
write corrected	16-bit
write uncorrected	8-bit
read corrected	16-bit
read uncorrected	8-bit

The log values are retained until cleared/set by the LOG SELECT command, or cleared by a SCSI reset or power cycle; note that they cannot be 'saved' to the 9914's non-volatile, and are not retained through a power cycle.

As currently implemented, the commands should only be issued at the start/end of a write or read sequence as they will cause the buffer to flush in a write operation and cancel the LAR action on read, resulting in slower operation.

The term 'corrected error' means that the drive/SCSI recovered the error by means of its normal retry sequence; it has no direct correlation with the CER error signal on the internal Pertec interface. The corrected error log is only incremented once for any given data block, irrespective of the number of physical retries required.

The term 'uncorrected error' means that the drive/SCSI was unable to recover the error after its normal retry sequence; it has no direct correlation with the HER signal on the internal Pertec interface.

The following paragraphs are intended as a quick guide on how the LOG SELECT command can be used. They should be read/used in conjunction with the SCSI-2 description of the commands and associated parameters. It is considered that the command is much easier to use in practice than might appear likely from the formal description in the SCSI-2 specification.

4.27.1 Using 'Log Select'

The LOG SELECT command is used mainly to clear the error logs and to set the threshold values that will trigger a UNIT ATTENTION condition. This feature may or may not be required; the 9914's thresholds default to maximum count.

All error logs can be cleared together or individually. Unlike the LOG SENSE command, all pages and parameters can be handled, if required, by a single command.

The following command descriptor block examples will carry out the function indicated:

- i) clear all error logs without need for parameter list

4C, 02, 40, 00, 00, 00, 00, 00, 00

- ii) clear read corrected error log

4C, 00, 40, 00, 00, 00, 00, 0A, 00 with the associated parameter list:

03	page code
00	reserved
00	page length msb
06	page length lsb
00	parameter code msb
03	parameter code lsb
04	control byte (TMC = 01)
02	parameter length
00	required value msb
00	required value lsb

4.28 Limited Long Block Write

bit	7	6	5	4	3	2	1	0
byte 0	0	0	0	0	1	1	0	0
byte 1	LUN			reserved				
byte 2	Transfer Length (msb)							
byte 3	Transfer Length							
byte 4	Transfer Length (lsb)							
byte 5	LFG	EOT	reserved				flag	link

This special vendor unique WRITE command is limited to a maximum block length of 16 Mbytes, and the first command of the sequence must indicate the overall block length. It is intended for use by those Hosts which cannot transfer more than 64 Kbytes per SCSI command, and hence cannot make use of the normal WRITE command's long block action.

Strictly speaking, each block of data is written to tape by a sequence of the above commands, utilising the LFG and EOT flags to control the overall action.

Consider the Host writing a 1 Mbyte (1,048,576 bytes) data block, as follows:

- the Host issues the first command of the sequence with the LFG bit of the CDB cleared, and the transfer length set to 1,048,576 bytes. Only 65,534 bytes will actually be transferred, however.
- on receipt, the SBI transfers the associated data into its buffer, then completes the command, going through the status/command complete/bus free sequence in the normal way.
- the Host will then issue 15 further commands with the LFG cleared, and the transfer length set to 65,534 bytes. As soon as the buffer is full during this sequence, the SBI will issue a write command to the tape unit, invoke its special long block algorithm, and start writing the data to tape.
- the Host will then issue a 32-byte WRITE with the LFG flag set. Completion of this command is carried out in the normal way, but will be held off until the tape write action is complete. Any tape related error will normally be reported with a check condition on this command.

If the EOT flag is set to 0, the SBI will abort the current command sequence when the tape unit reaches physical EOT marker. If the EOT flag is set to 1, the SBI will carry on writing through the EOT marker, and the host will not be given any indication that EOT has been reached until the command sequence is complete (see 'special operational notes' below).

It should be noted that tape data is transferred from the SBI to the tape unit across a synchronous interface which cannot be halted or delayed without incurring loss of data. Consequently the Host must be capable of transferring data to the SBI at a minimum rate equivalent to the tape data rate. This is an averaged rate and must include all overheads associated with selection, CDB transfer, status phase etc. The necessary minimum data rate is determined by the tape unit's speed and density, as shown in the next table.

Density & Speed			Data Rate
800 bpi NRZ	Low speed	42 ips	34 kB/s
	High speed	125 ips	100 kB/s
1600 bpi PE	Low speed	42 ips	67 kB/s
	High speed	125 ips	200 kB/s
6250 bpi GCR	Low speed	42 ips	260 kB/s
	High speed	125 ips	781 kB/s

Error checking and reporting is similar to the normal WRITE command. If a tape error occurs, the error is reported as normal, and the tape left positioned in the gap following the faulty block; no automatic retrying takes place.

The following abnormal conditions can occur and will result in CHECK CONDITION being sent:

a) Tape Error

MEDIUM ERROR is set in the sense data, no re-trying takes place.

b) Data Under Run

If the host fails to keep up with the tape at any point, the tape write is aborted, and MEDIUM ERROR set with INTERNAL TARGET FAILURE additional sense set.

c) EOT or EW Encountered

CDB/EOT state	
1	EOM/NO SENSE is set in the sense data if the command completes in the Early Warning area or beyond the tape's EOT marker.
0	EOM/NO SENSE is set in the sense data if the command completes in the Early Warning area. The tape write is aborted and EOM/MEDIUM ERROR set in the sense data if the physical EOT marker is encountered. The data block created is of indeterminate length, and is not a valid tape data block.

Special operational notes:

- This 'limited' version of the command has been specifically created to run on the Super SCSI version of the SBI, where the board's hardware architecture requires the block length being written to be known in advance of starting the tape write. It is also available on the Standard version of the board, however, and consequently should be the command of choice, if possible, for any Host application software.
- LFG flag set to indicate the last command in the string.
- the LFG flag may be set on the first command, allowing block lengths less than 64K to be written.
- all commands except the first and last in the string must have a transfer length of 65,534 (or spurious data will be written at the end of each segment).
- if a CHECK CONDITION is returned for any command, the Host must assume the end of the string whether it had the LFG flag set or not.
- it is recommended that the host system / operator size the tape in advance according to length, density and block size in order to avoid reaching the physical EOT marker.
- it is also recommended that the EOT bit is cleared on all commands; this will result in the command sequence being aborted if the physical EOT marker is detected. The host can then backspace, erase and close the tape.
- if the EOT bit is set, the host must maintain a log of remaining tape capacity in order to ensure tape integrity at EOT (ie avoid running off the end of tape), or restrict the maximum block length written to be equivalent to the Early Warning distance of 60 ft:

Density	Maximum Block Length (bytes)
800	576,000
1600	1,152,000
6250	4,500,000

This page has no technical content

5.

Completion Status

All commands contain a STATUS phase immediately prior to COMMAND COMPLETION. This consists of a single byte transferred from the SBI to the INITIATOR which basically informs the INITIATOR whether the command has completed successfully, or whether something has happened which might require the INITIATOR to check further, eg check the SENSE DATA.

The format of the status byte, and the information conveyed, is shown in Table 5.1.

Bits of status byte								Status
7	6	5	4	3	2	1	0	
0	0	0	0	0	0	0	0	GOOD STATUS
0	0	0	0	0	0	1	0	CHECK CONDITION
0	0	0	0	1	0	0	0	BUSY
0	0	0	1	0	0	0	0	INTERMEDIATE GOOD
0	0	0	1	1	0	0	0	RESERVATION CONFLICT

TABLE 5.1

Good Status (00h)

This indicates that the SBI has completed the command without an error or exception of any kind.

Check Condition (02h)

This indicates that some form of error, exception or abnormal condition has occurred. Valid sense data is now available for the REQUEST SENSE command if the INITIATOR wants to enquire further.

Busy Status (08h)

Indicates that the SBI is busy and cannot implement the command. Returned in the normal way following receipt of the CDB.

Intermediate Good (10h)

As GOOD above, but for commands within a series of linked commands (except for the final command).

Reservation Conflict (18h)

Sent whenever an INITIATOR attempts to access a LUN which has been reserved for the exclusive use of another INITIATOR.

This page has no technical content

A.

Set Up

This appendix details the various functional options/settings which may be determined by dual-in-line switches on the SCSI board. The settings are sampled once following power-on, and usually retained until a further power cycle takes place.

The 9914 GCR Streamer also supports control of the SCSI options from internal non-volatile RAM (NVR). At power-on, option byte 13 (bit 4) determines whether the switches are sampled or the NVR contents are sampled.

Byte 13, bit 4	Options taken from
' 1 ' at power-on	Non-volatile RAM
' 0 ' at power-on	SCSI board switches

The NVR contents may be changed by the operator via the operator's panel controls at the front of the 9914. If they are changed, byte 13 bit 4 is automatically set to 1 and subsequently (even if a power cycle takes place), the NVR settings override the dual-in-line settings until bit 4 is changed to 0 and the unit power cycled.

A switch set to 'off' corresponds to an NVR bit set to '1'.

A.1 Dual-in-Line Switch Settings

Various switched options are available as described next, the switch identities which invoke each option for a particular SCSI PROM revision are tabulated.

A.1.1 PROM 124610

Switch	Action	NVR bit
SW1-1	SCSI ID, lsb	B15, b0
SW1-2	SCSI ID	B15, b1
SW1-3	SCSI ID, msb	B15, b2
SW1-4	On Buffered write EOM flagged when physical EOT encountered	B15, b3
	Off Buffered write EOM flagged following buffer dump in EW area	

The 8 SCSI ID addresses are encoded as for other PROM revisions.

Switch	Action	NVR bit
SW2-1	On Differential Interface enabled Off Single-ended Interface enabled	B14, b7*
SW2-2	On DISCONNECT inhibited Off DISCONNECT enabled	B14, b6
SW2-3	On ATTENTION reporting inhibited Off ATTENTION reporting enabled	B14, b5
SW2-4	On RECOVERED ERROR reporting inhibited Off RECOVERED ERROR reporting enabled	B14, b4
SW2-5	On Normal SCSI Off Fast SCSI	B14, b3
SW2-6	Code compatibility	B14, b2
SW2-7	Code compatibility	B14, b1
SW2-8	Code compatibility	B14, b0

***Notes:**

1. This option must be set as required, even with a single-interface Super SCSI board.
2. The streamer must be power cycled following any change to this bit, in order for the new option to become effective.

Switch		Action	NVR bit
SW3-1	On	FM status during read or space	B15, b7
	Off	FM status during write, read or space	
SW3-2	On	EOT ignored when reading or spacing	B15, b6
	Off	Tape halted and Check Condition returned upon read / space into EOT	
SW3-3	On	Irrecoverable write errors finish with head on EOT side of block	B15, b5
	Off	Irrecoverable write errors finish with head on BOT side of block	
SW3-4		Code compatibility	B15, b4

The code compatibility bits are encoded as shown:

SW3-4	SW2-6	SW2-7	SW2-8	
off	off	on	off	<i>M4 Data standard</i>
off	off	off	on	<i>STK standard</i>
off	off	off	off	<i>RS/6000</i>
x	x	x	x	<i>reserved</i>

This page has no technical content

B.

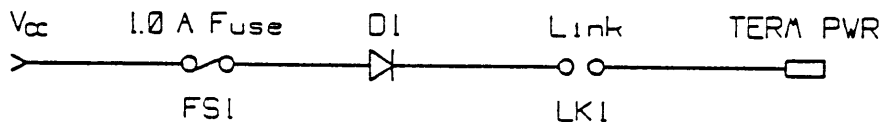
Interface Termination

None of the Super SCSI SBI boards provide on-board facilities for terminating the SCSI bus, whether using differential or single-ended signals.

The SCSI bus must be suitably terminated at the unit which is the last physical entity on the bus. If the last unit is a 9914 Streamer containing a *StorageTek* Super SCSI board, external terminators must be used.

Term Pwr

The Super SCSI SBI provides Terminating Power via a diode and fuse to pin 26 of its single-ended connector, and pins 25 & 26 of its differential connector. A removable link allows the SBI's power to be supplied to the SCSI bus, as illustrated next.



If Term Pwr is to be supplied to the SCSI bus, the link must be fitted.

If Term Pwr is not to be supplied to the SCSI bus, the link must be removed.

Diffsens

The DIFFSENS input signal of the connector is permanently connected as an active high enable for the differential drivers. If a single-ended device or terminator is inadvertently connected, this signal is grounded, disabling the differential drivers.

This page has no technical content

C.

Streamer Manuals

There are manuals associated with each streamer, giving performance figures, installation details, operating procedures, diagnostic program details, and status byte tables.

M G0595-A	9914 Product Specification
95 121797	9914 User/Diagnostic Manual
95 121798	9914 Servicing Manual

This page has no technical content

D.

Customer Specific Variants

Where customer-required variations of the standard operation have been implemented, these are normally accommodated by firmware changes only. The firmware is referenced by an identification number on the EPROM at the chip location as tabulated.

Board PL No.	1246XX	1246XX
Artwork Issue	A	B
EPROM location	U116	U99

This page has no technical content

E.

Diagnostic Program CDBs and Parameter Lists

E.1 Non-Page Formats

This page format is not applicable to the 9914.

E.2 Page Code 00 Format

The SEND DIAGNOSTIC CDB associated with Page Code 00 is:

bit	7	6	5	4	3	2	1	0	
byte 0	0	0	0	1	1	1	0	1	Command Code ID
byte 1	LUN			1	0	0	1	1	PF = 1 (SCSI-2 page format)
byte 2	reserved								
byte 3	0	0	0	0	0	0	0	0	Parameter Block Length (msb) = 0
byte 4	Parameter Block Length (lsb)								
byte 5	0	0	reserved				flag	link	

E.2.1 Parameter List for Page Code 00

The all-zero parameter list (to access the list of page codes supported) is:

bit	7	6	5	4	3	2	1	0	
byte 0	0	0	0	0	0	0	0	0	Page Code 00
byte 1	reserved								
byte 2	0	0	0	0	0	0	0	0	Page Parameter Length (msb) = 0
byte 3	0	0	0	0	0	0	0	0	Page Parameter Length (lsb) = 0

Note: this is the only use of Page Code 00.

E.3 Page Code 80 Formats

The SEND DIAGNOSTIC CDB associated with Page Code 80 is:

bit	7	6	5	4	3	2	1	0	
byte 0	0	0	0	1	1	1	0	1	Command Code 1D
byte 1	LUN			1	0	0	1	1	PF = 1 (SCSI-2 page format)
byte 2	reserved								
byte 3	0	0	0	0	0	0	0	0	Parameter Block Length (msb) = 0
byte 4	Parameter Block Length (lsb)								
byte 5	0	0	reserved			flag		link	

E.3.1 Parameter Lists for Page Code 80

A sample parameter list (for a stack of diagnostic programs) is:

bit	7	6	5	4	3	2	1	0	
byte 0	1	0	0	0	0	0	0	0	Page Code 80
byte 1	reserved								
byte 2	0	0	0	0	0	0	0	0	Page Parameter Length (msb) = 0
byte 3	0	0	0	0	1	0	0	0	Page Parameter Length (lsb) = 8
byte 4	0	0	0	0	1	1	0	1	Program 13 (set N)
byte 5	0	0	0	0	0	0	0	0	lsb of N
byte 6	0	0	1	0	1	0	0	0	msb of N] N = 40
byte 7	0	0	0	1	0	1	0	0	Program 20 (set all 0's data)
byte 8	0	0	0	1	1	0	1	0	Program 26 (write N blocks)
byte 9	0	0	0	1	0	1	0	1	Program 21 (set all 1's)
byte 10	0	0	0	1	1	0	1	0	Program 26 (write N blocks)
byte 11	0	0	0	1	1	1	1	0	Program 30 (rewind)

The all-zero parameter list (to access the condensed status bytes) is:

bit	7	6	5	4	3	2	1	0	
byte									
byte 0	1	0	0	0	0	0	0	0	Page Code 80
byte 1	reserved								
byte 2	0	0	0	0	0	0	0	0	Page Parameter Length (msb) = 0
byte 3	0	0	0	0	0	0	0	0	Page Parameter Length (lsb) = 0

E.4 Page Code 81 Formats

The SEND DIAGNOSTIC CDB associated with page 81 is:

bit	7	6	5	4	3	2	1	0	
byte									
byte 0	0	0	0	1	1	1	0	1	Command Code 1D
byte 1	LUN			1	0	0	1	1	PF = 1 (SCSI-2 page format)
byte 2	reserved								
byte 3	0	0	0	0	0	0	0	0	Parameter Block Length (msb) = 0
byte 4	Parameter Block Length (lsb)								
byte 5	0	0	reserved			flag		link	

E.4.1. Parameter Lists for Page Code 81

A sample parameter list (consisting of a stack of diagnostic programs) for Page Code 81 is:

bit	7	6	5	4	3	2	1	0	
byte 0	1	0	0	0	0	0	0	1	Page Code 81
byte 1	reserved								
byte 2	0	0	0	0	0	0	0	0	Page Parameter Length (msb) = 0
byte 3	0	0	0	0	0	1	0	1	Page Parameter Length (lsb) = 5
byte 4	0	0	0	0	0	1	0	0	Program 04 (servo check)
byte 5	0	0	0	1	1	1	1	0	Program 30 (rewind)
byte 6	0	0	0	1	1	1	1	1	Program 31 (write/check file mark)
byte 7	0	0	0	1	1	1	1	1	Program 31 (write/check file mark)
byte 8	0	0	0	1	1	1	1	0	Program 30 (rewind)

The all-zero parameter list (to access the full status bytes) is:

bit	7	6	5	4	3	2	1	0	
byte 0	1	0	0	0	0	0	0	1	Page Code 81
byte 1	reserved								
byte 2	0	0	0	0	0	0	0	0	Page Parameter Length (msb) = 0
byte 3	0	0	0	0	0	0	0	0	Page Parameter Length (lsb) = 0

E.5 Page 82 Formats

The SEND DIAGNOSTIC CDB associated with page 82 is:

bit	7	6	5	4	3	2	1	0	
byte									
byte 0	0	0	0	1	1	1	0	1	Command Code 1D
byte 1	LUN			1	0	0	1	1	PF = 1 (SCSI-2 page format)
byte 2	reserved								
byte 3	0	0	0	0	0	0	0	0	Parameter Block Length (msb) = 0
byte 4	Parameter Block Length (lsb)								
byte 5	0	0	reserved			flag		link	

E.5.1. Parameter Lists for Page Code 82

Specific parameter lists (for each relevant diagnostic program) for Page Code 82 are:

Program 01 - Confidence Check

bit	7	6	5	4	3	2	1	0	
byte									
byte 0	1	0	0	0	0	0	1	0	Page Code 82
byte 1	reserved								
byte 2	0	0	0	0	0	0	0	0	Page Parameter Length (msb) = 0
byte 3	0	0	0	0	0	0	0	1	Page Parameter Length (lsb)
byte 4	0	0	0	0	0	0	0	1	Page Parameter Start

Program 02 - Run Stack from Host

The SCSI board inserts its own 'run stack', ignoring this program if it is added to a stack.

Program 05 - Data Channel Check (non-corruptive)

bit	7	6	5	4	3	2	1	0	
byte 0	1	0	0	0	0	0	1	0	Page Code 82
byte 1	reserved								
byte 2	0	0	0	0	0	0	0	0	Page Parameter Length (msb) = 0
byte 3	0	0	0	0	0	1	1	0	Page Parameter Length (lsb) = 6
byte 4	0	0	0	0	0	1	0	1	Page Parameter Start
byte 5	S	0	0	0	C	H	X	1	S = hi speed C = ICER X = non-streaming H = IHER
byte 6	0	0	0	0	0	0	0	1	
byte 7	reserved								
byte 8	Block Length (msb)								
byte 9	Block Length (lsb)								

Program 06- Data Channel Check (corruptive)

bit	7	6	5	4	3	2	1	0	
byte 0	1	0	0	0	0	0	1	0	Page Code 82
byte 1	reserved								
byte 2	0	0	0	0	0	0	0	0	Page Parameter Length (msb) = 0
byte 3	0	0	0	0	0	1	1	0	Page Parameter Length (lsb) = 6
byte 4	0	0	0	0	0	1	1	0	Page Parameter Start
byte 5	S	0	0	0	C	H	X	1	S = hi speed C = ICER X = non-streaming H = IHER
byte 6	0	0	0	0	0	0	0	1	
byte 7	reserved								
byte 8	Block Length (msb)								
byte 9	Block Length (lsb)								

Program 09 - Data Channel Error Reporting

bit	7	6	5	4	3	2	1	0	
byte									
byte 0	1	0	0	0	0	0	1	0	Page Code 82
byte 1	reserved								
byte 2	0	0	0	0	0	0	0	0	Page Parameter Length (msb) = 0
byte 3	0	0	0	0	0	1	1	0	Page Parameter Length (lsb) = 6
byte 4	0	0	0	0	1	0	0	1	Page Parameter Start
byte 5	S	0	0	0	0	0	0	0	S = hi speed
byte 6	0	0	0	0	0	0	0	1	
byte 7	reserved								
byte 8	Block Length (msb)								
byte 9	Block Length (lsb)								

Program 10 - Set Parameters of Program 01

bit	7	6	5	4	3	2	1	0	
byte									
byte 0	1	0	0	0	0	0	1	0	Page Code 82
byte 1	reserved								
byte 2	0	0	0	0	0	0	0	0	Page Parameter Length (msb) = 0
byte 3	0	0	0	0	0	1	1	1	Page Parameter Length (lsb)
byte 4	0	0	0	0	1	0	1	0	Page Parameter Start
byte 5	reserved								
byte 7	Number of Blocks								
byte 8	reserved								
byte 8	Block Length (msb)								
byte 9	Block Length (lsb)								
byte 10	Data Pattern								14h = 0's 15h = 1's 16h = 1,0 17h = random

Program 20 - Set All Zeros Data

bit	7	6	5	4	3	2	1	0	
byte									
byte 0	1	0	0	0	0	0	1	0	Page Code 82
byte 1	reserved								
byte 2	0	0	0	0	0	0	0	0	Page Parameter Length
byte 3	0	0	0	0	0	0	0	1	Page Parameter Length
byte 4	0	0	0	1	0	1	0	0	Page Parameter Start

Program 21 - Set All Ones Data

bit	7	6	5	4	3	2	1	0	
byte									
byte 0	1	0	0	0	0	0	1	0	Page Code 82
byte 1	reserved								
byte 2	0	0	0	0	0	0	0	0	Page Parameter Length
byte 3	0	0	0	0	0	0	0	1	Page Parameter Length
byte 4	0	0	0	1	0	1	0	1	Page Parameter Start

Program 22 - Set Alternate 1/0 Data

bit	7	6	5	4	3	2	1	0	
byte									
byte 0	1	0	0	0	0	0	1	0	Page Code 82
byte 1	reserved								
byte 2	0	0	0	0	0	0	0	0	Page Parameter Length
byte 3	0	0	0	0	0	0	0	1	Page Parameter Length
byte 4	0	0	0	1	0	1	1	0	Page Parameter Start

Program 23 - Set Pseudo Random Data

bit	7	6	5	4	3	2	1	0	
byte 0	1	0	0	0	0	0	1	0	Page Code 82
byte 1	reserved								
byte 2	0	0	0	0	0	0	0	0	Page Parameter Length
byte 3	0	0	0	0	0	0	0	1	Page Parameter Length
byte 4	0	0	0	1	0	1	1	1	Page Parameter Start

Program 24 - Set Fail Limits

bit	7	6	5	4	3	2	1	0	
byte 0	1	0	0	0	0	0	1	0	Page Code 82
byte 1	reserved								
byte 2	0	0	0	0	0	0	0	0	Page Parameter Length (msb) = 0
byte 3	0	0	0	0	1	0	1	0	Page Parameter Length (lsb)
byte 4	0	0	0	1	1	0	0	0	Page Parameter Start
byte 5	reserved								
byte 6	reserved								
byte 7	reserved								
byte 8	reserved								
byte 9	reserved								
byte 10	Write Re-try Count								
byte 11	Write Error Limit								
byte 12	Read Re-try Count								
byte 13	Read Error Limit								

Program 25 - Write, Backspace, Read (N blocks)

bit	7	6	5	4	3	2	1	0	
byte									
byte 0	1	0	0	0	0	0	1	0	Page Code 82
byte 1	reserved								
byte 2	0	0	0	0	0	0	0	0	Page Parameter Length (msb) = 0
byte 3	0	0	0	0	0	1	1	0	Page Parameter Length (lsb)
byte 4	0	0	0	1	1	0	0	1	Page Parameter Start
byte 5	S	0	0	0	C	H	X	0	S = hi speed C = ICER X = non-streaming H = IHER
byte 6	Number of Blocks								
byte 7	reserved								
byte 8	Block Length (msb)								
byte 9	Block Length (lsb)								

Program 26 - Write N Blocks

bit	7	6	5	4	3	2	1	0	
byte									
byte 0	1	0	0	0	0	0	1	0	Page Code 82
byte 1	reserved								
byte 2	0	0	0	0	0	0	0	0	Page Parameter Length (msb) = 0
byte 3	0	0	0	0	0	1	1	0	Page Parameter Length (lsb)
byte 4	0	0	0	1	1	0	1	0	Page Parameter Start
byte 5	S	0	0	0	C	H	X	0	S = hi speed C = ICER X = non-streaming H = IHER
byte 6	Number of Blocks								
byte 7	reserved								
byte 8	Block Length (msb)								
byte 9	Block Length (lsb)								

Program 27 - Space Reverse N Blocks

bit	7	6	5	4	3	2	1	0	
byte									
byte 0	1	0	0	0	0	0	1	0	Page Code 82
byte 1	reserved								
byte 2	0	0	0	0	0	0	0	0	Page Parameter Length (msb) = 0
byte 3	0	0	0	0	0	1	1	0	Page Parameter Length (lsb)
byte 4	0	0	0	1	1	0	1	1	Page Parameter Start
byte 5	S	0	0	0	0	0	X	0	S = hi speed X = non-streaming
byte 6	Number of Blocks								
byte 7	reserved								
byte 8	Block Length (msb)								
byte 9	Block Length (lsb)								

Program 28 - Read Forward N Blocks

bit	7	6	5	4	3	2	1	0	
byte									
byte 0	1	0	0	0	0	0	1	0	Page Code 82
byte 1	reserved								
byte 2	0	0	0	0	0	0	0	0	Page Parameter Length (msb) = 0
byte 3	0	0	0	0	0	1	1	0	Page Parameter Length (lsb)
byte 4	0	0	0	1	1	1	0	0	Page Parameter Start
byte 5	S	0	0	0	C	H	X	0	S = hi speed C = ICER X = non-streaming H = IHER
byte 6	Number of Blocks								
byte 7	reserved								
byte 8	Block Length (msb)								
byte 9	Block Length (lsb)								

Program 29 - Read Reverse N Blocks

bit	7	6	5	4	3	2	1	0	
byte									
byte 0	1	0	0	0	0	0	1	0	Page Code 82
byte 1	reserved								
byte 2	0	0	0	0	0	0	0	0	Page Parameter Length (msb) = 0
byte 3	0	0	0	0	0	1	1	0	Page Parameter Length (lsb)
byte 4	0	0	0	1	1	1	0	1	Page Parameter Start
byte 5	S	0	0	0	C	H	X	0	S = hi speed C = ICER X = non-streaming H = IHER
byte 6	Number of Blocks								
byte 7	reserved								
byte 8	Block Length (msb)								
byte 9	Block Length (lsb)								

Program 32 - Space Forward N Blocks

bit	7	6	5	4	3	2	1	0	
byte									
byte 0	1	0	0	0	0	0	1	0	Page Code 82
byte 1	reserved								
byte 2	0	0	0	0	0	0	0	0	Page Parameter Length (msb) = 0
byte 3	0	0	0	0	0	1	1	0	Page Parameter Length (lsb)
byte 4	0	0	1	0	0	0	0	0	Page Parameter Start
byte 5	S	0	0	0	0	0	X	0	S = hi speed X = non-streaming
byte 6	Number of Blocks								
byte 7	reserved								
byte 8	reserved								
byte 9	reserved								

Multiple Programs

Page parameters may be strung together, within the same page format, provided that byte 3 reflects the number of page parameter bytes (which follow immediately in the list).

Example: to set up the data pattern to 'all ones' and write 5 blocks, each 1K in length:

bit	7	6	5	4	3	2	1	0	
byte 0	1	0	0	0	0	0	1	0	Page Code 82
byte 1	reserved								
byte 2	0	0	0	0	0	0	0	0	Page Parameter Length (msb) = 0
byte 3	0	0	0	0	0	1	1	1	Page Parameter Length (lsb) = 7
byte 4	0	0	0	1	0	1	0	1	Program 21 (set 'all ones' data)
byte 5	0	0	0	1	1	0	1	0	Program 26 (write N blocks)
byte 6	S	0	0	0	C	H	X	0	S = hi speed C = ICER X = non-streaming H = IHER
byte 7	0	0	0	0	0	1	0	1	Number of Blocks = 5
byte 8	reserved								
byte 9	0	0	0	0	0	1	0	0	msb } 1K Block Length
byte 10	0	0	0	0	0	0	0	0	lsb

This page has no technical content