DATAPOINT

Model 9330 Disk/Tape Subsystem

Technical Reference

61994

September 1985

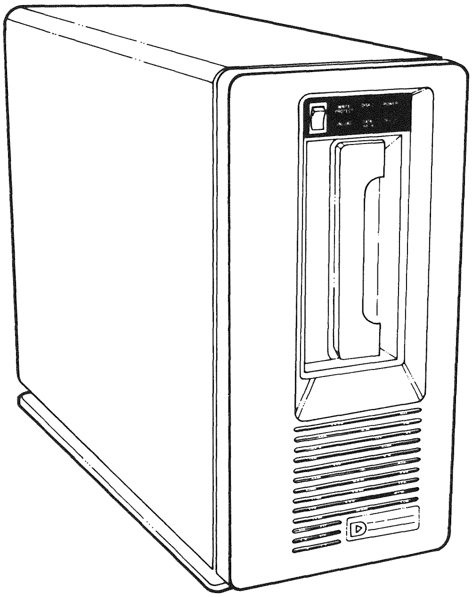
WARNING

This equipment generates, uses, and can radiate radio frequency energy and if not installed and used in accordance with the instructions manual, may cause interference to radio communications. It has been tested and found to comply with the limits for a Class A computing device pursuant to Subpart J of Part 15 of FCC Rules, which are designed to provide reasonable protection against such interference when operated in a commercial environment. Operation of this equipment in a residential area is likely to cause interference in which case the user at his own expense will be required to take whatever measures may be required to correct the interference.

Document No. 61994.

System features are subject to change without notice.

Model 9330 Disk/Tape Subsystem



Technical Reference

61994 September 1985

Preface

The DATAPOINT 9330 Series Disk/Tape Subsystems are compact data storage units for DATAPOINT 8600 Processors. The mid-capacity fixed-disk subsystem has streaming tape backup capability. This manual contains operation and programming information for those who require detailed technical information.

> Basic operating procedures are described in the *Model 9330 Disk/Tape Subsystem Operating Guide/Product Specification*, Document No. 61888. See DATAPOINT *Software & Documentation Price Schedule*, Document No. 60231.

CONTENTS

CONTENTS

CONTENTS

Chapter 1.
9330 DISK/TAPE SUBSYSTEM
Overview

Introduction

The DATAPOINT 9330 Series Disk/Tape Subsystems are data storage units for DATAPOINT processors. The midcapacity disk subsystem has streaming tape backup capability.

Description

The 9330 Disk/Tape Subsystem is contained in a slim vertical enclosure for efficient use of available space. Winchester disk and streaming cartridge tape technology provide reliable performance, quick backup, and minimum on-site maintenance.

Comment

All command code designations in this manual are in hexadecimal notation and are identified by the prefix HEX.

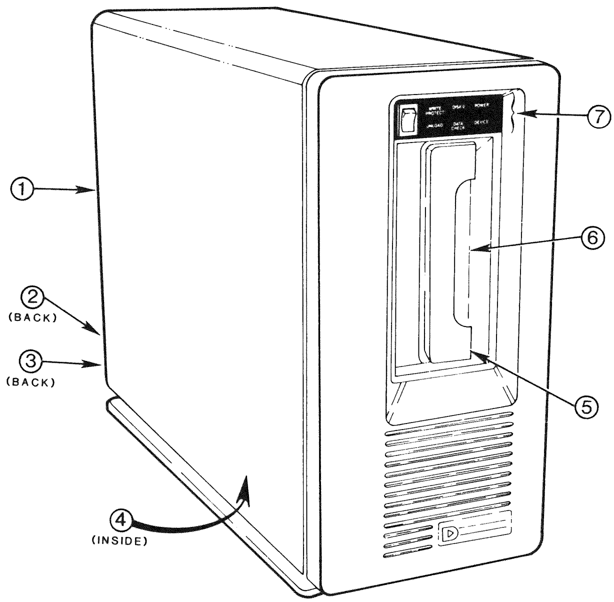
Contents

This chapter includes general subsystem information in:

- a Part/Function table,
- a subsystem diagram,
- a configuration table and diagrams,
- cable and connector descriptions, and
- a chapter contents table that briefly describes each chapter.

Parts of the Subsystem

Disk/Tape subsystem diagram



Part/Function table

The disk/tape subsystem's major parts and their functions are listed below.

KEY	PART	FUNCTION
1	Enclosure	Contains subsystem electronics
2	Power cord	Connects the device to the AC power source
3	Power switch	Turns device power on or off
4	Disk drive	Provides 28 or 65 MB disk storage
5	Tape drive	Provides 65 MB streaming tape storage
6	Ejection lever	Partially ejects the tape cartridge for easy removal
7	Control panel	Contains the write protect switch, disk light, and status lights

Configurations

Five versions of the 9330 Series Disk/Tape Subsystem are available. Each configuration is described in the following table.

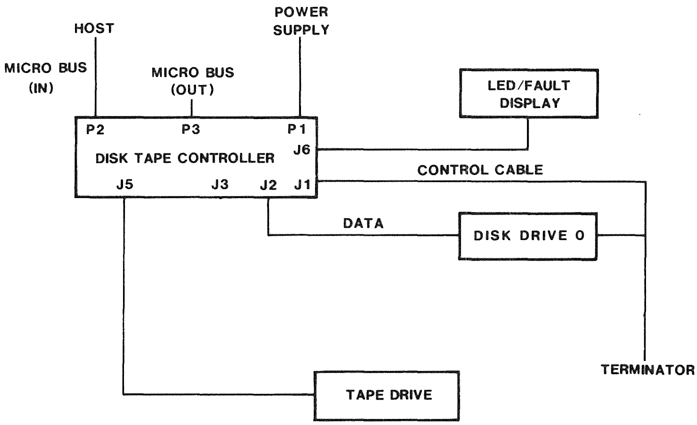
MODEL CODE	DESCRIPTION
9334	A disk/tape subsystem with a 65 MB streaming tape drive and a 28 MB disk drive
9335	A disk expansion unit with a 28 MB disk drive
9336	A disk/tape subsystem with a 65 MB streaming tape drive and a 65 MB disk drive
9337	A disk expansion unit with a 65 MB disk drive
9339	A tape-only configuration with a 65 MB streaming tape drive

Note:

Storage capacities are for a formatted disk or tape. The tape capacity is based on 8 K byte blocks.

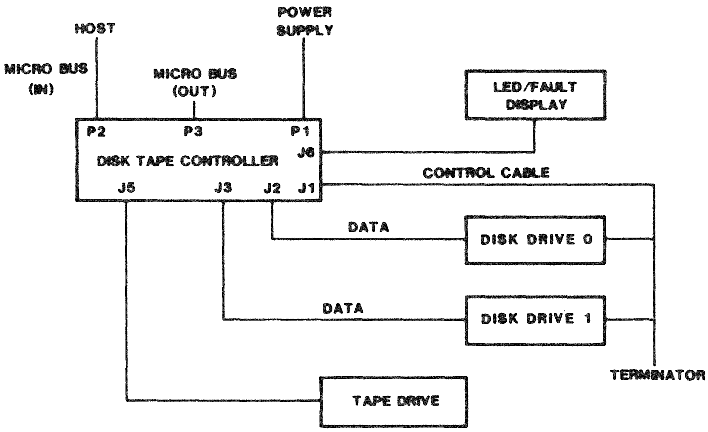
Disk/Tape subsystem block diagram

The following diagram illustrates the basic components of the disk/tape subsystem.



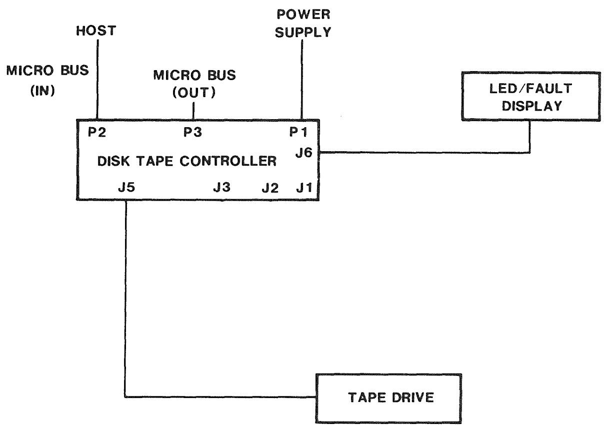
Disk expansion subsystem block diagram

The following diagram illustrates the basic components of the disk expansion subsystem.



Tape-only subsystem block diagram

The following diagram illustrates the basic components of the tape-only subsystem.



Introduction

The subsystem has two external cables:

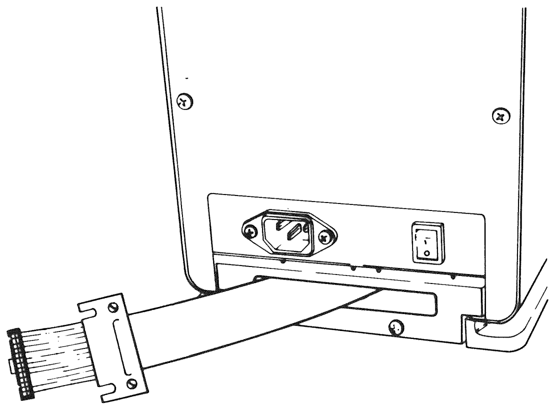
- a Micro Bus interface cable to a processor or other micro bus peripheral, and
- an AC power cord.

Interface cable

The interface cable connects the subsystem to your processor. It must be installed by a DATAPOINT Customer Service Engineer.

After installation, you can disconnect the subsystem only at the processor.

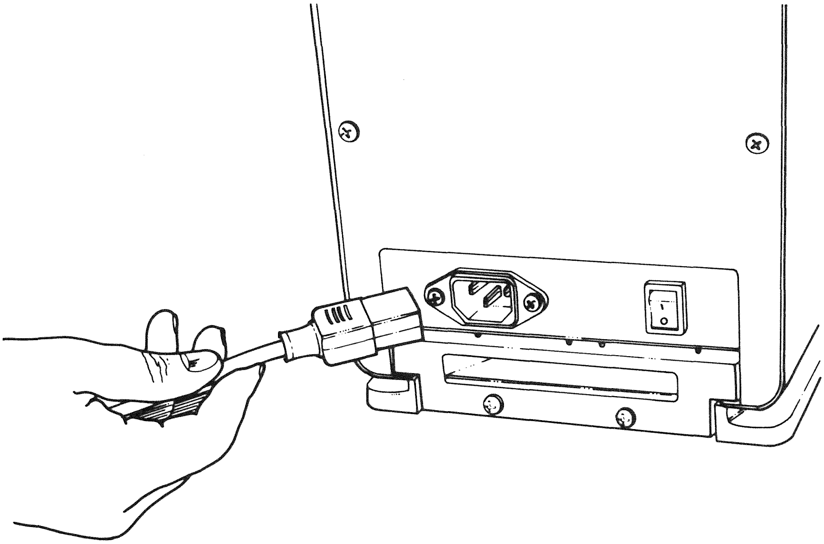
Interface cable diagram



AC power cord

The power cord for the subsystem is detachable. It is connected to the socket located on the back of the cabinet at the lower left.

Power cord connector diagram



Chapter Contents

Introduction

This reference contains information for system designers, system managers, or others needing detailed technical and programming information. The remaining chapters' contents are briefly described in the following table.

CHAPTER...	CONTAINS...
2, *SPECIFICATIONS*	• subsystem operating specifications, • subsystem operating requirements, and • regulatory compliance information.
3, *CONTROLLER*,	• a functional description, • connector pin designations, • configuration jumper tables, and • a multiple controller diagram.
4, *INTERFACE COMMAND STRUCTURE*,	• an interface command summary and format, • interrupt information, • the command transfer sequence, • abnormal termination information, and • interface command formats.
5, *DISK COMMAND STRUCTURE*,	detailed descriptions of the disk drive commands.

Chapter contents continued

CHAPTER...	CONTAINS...
6, *DISK DRIVE ERROR CODES*,	• a disk error code summary for quick reference, • detailed descriptions of the disk error codes, • firmware error routine information, and • alternate track assignment and handling information.
7, *TAPE COMMAND STRUCTURE*,	detailed descriptions of the tape drive commands.
8, *TAPE DRIVE ERROR CODES*,	• a tape drive error code summary and • detailed information on the Sense Status Block.
9, *DIAGNOSTIC COMMAND STRUCTURE*,	• detailed descriptions of the diagnostic command codes.

Chapter 2. SPECIFICATIONS

Overview

Introduction

The DATAPOINT 9330 Series Subsystem has three main technical components:

- a controller,
- a fixed disk drive, and
- a cartridge tape drive.

Subsystem operating requirements and performance criteria are specified in this chapter.

Contents

This chapter includes:

- controller specifications,
- disk drive specifications,
- tape drive specifications,
- physical, power, and cable specifications, and
- environmental requirements.

Controller specifications

The 9330 subsystem microprocessor-based controller is the interface between the host processor and the subsystem's components. The specifications are:

PART	SPECIFICATION
Interface	DATAPOINT Micro Bus with: • 8-bit command/address bus, • 8-bit data bus, and • polled or interrupt mode.
Disk interface	ST506
Tape interface	proprietary

Controller features

FEATURES	DESCRIPTION
Data buffers	Two 8 K buffers for disk/tape operations
ECC	Hardware 32-bit polynomial, 11-bit burst error correction
Diagnostics	Internal with on-board power/fault indication
Variable interleave	3 to 1 minimum achievable
Overlapped seek	Allows both drives to be positioned simultaneously
Head and cylinder switching	Automatic during a multisector transfer if the end of track or cylinder is reached. The controller issues a seek and resumes transfer.
Retry and error correction option	User selectable and programmable from the host

FEATURES	DESCRIPTION
Disk/Tape copy	Host independent; tape maintains streaming at any interleave factor
Write protect	Control panel switch
Defective/Alternate track processing	Automatic; transparent to the host

Disk Drive Specifications

Fixed disk description

General performance specifications of the fixed disk include:

DESCRIPTION	SPECIFICATION	
	MODEL 9334/35 28 MB DISK	MODEL 9336/37 65 MB DISK
Disk drive	28 MB Winchester technology	65 MB Winchester technology
Disk surfaces	5	9
Tracks per surface	697	924
Sectors per track	32	32
Bytes per sector	256	256
Bytes per track	8192	8192
Bytes per drive	28.5 MB (user capacity)	68.1 MB (user capacity)

Fixed disk timing

DESCRIPTION	SPECIFICATION	
	MODEL 9334/35 28 MB DISK	MODEL 9336/37 65 MB DISK
Bit transfer rate	5 MB	5 MB
Rotation speed	3600 RPM	3600 RPM
Rotation delay	8.33 ms	8.33 ms
Head positioning (average)	45 ms	35 ms
Track-to-track positioning time	9 ms	8 ms
Start time (maximum)	30 s	30 s

Tape drive timing

Tape drive timing specifications are:

DESCRIPTION	SPECIFICATION		
	MINIMUM	NOMINAL	MAXIMUM
Tape velocity	50 ips	55 ips	60 ips
Effective recording density	--	8000 bps	--
Initialization time (600-ft cartridge)	130 s	--	340 s
Access time (from rest)	--	61 ms	--
Start/Stop distance	1.20 in.	1.40 in.	1.90 in.
Interblock gap size	0.56 in.	0.60 in.	0.76 in.
Repositioning time	325 ms	350 ms	395 ms
Positioning time	270 ms	290 ms	325 ms
Command reinstruct time	--	--	5.4 ms
Head positioning time (track to adjacent track)	670 ms	740 ms	810 ms

Tape cartridge description

General performance characteristics of the tape cartridge (DATAPOINT Part Number 80925) include:

<table>
<tr><th>DESCRIPTION</th><th>SPECIFICATION</th></tr>
<tr><td>Tape cartridge</td><td>Removable, 1/4 in.</td></tr>
<tr><td>Block size</td><td>256 bytes to 8 K bytes</td></tr>
<tr><td>Formatted capacity</td><td>Capacity is a function of block size. The following table gives examples.
<table><tr><td>BLOCK SIZE</td><td>1K</td><td>2K</td><td>4K</td><td>8K</td></tr><tr><td>CAPACITY (MB)</td><td>47</td><td>57</td><td>64</td><td>68</td></tr></table></td></tr>
<tr><td>Average data transfer rate</td><td>Transfer rate is a function of block size. The following table gives examples.
<table><tr><td>BLOCK SIZE</td><td>1K</td><td>2K</td><td>4K</td><td>8K</td></tr><tr><td>TRANSFER RATE (K bits/sec)</td><td>264</td><td>321</td><td>360</td><td>383</td></tr></table></td></tr>
<tr><td>Read time</td><td>24 minutes maximum</td></tr>
</table>

Comment

The streaming tape drive operates at an instantaneous data transfer rate of 440 K bits/second. The host I/O subsystem must maintain an average data transfer rate as specified in the *Tape cartridge description* table to sustain streaming operation.

Equipment dimensions

DIMENSION	SPECIFICATION
Height	14.46 in. (36.73 cm)
Width	7.00 in. (17.78 cm)
Depth	19.12 in. (48.56 cm)
Weight	50 lb (22.68 kg) maximum

Power requirements

REQUIREMENT	SPECIFICATION
AC line cord	6 feet (1.8 meter) with 3-prong plug
AC input voltage	• 120 VAC, +10/-15% standard • Field configurable for operation at 100, 110, 220, 230, or 240 VAC +10/-15%
Frequency	50 to 60 Hz ±2%
Power consumption	220 watts maximum (active)

Cable Specifications

The following table lists the specifications for the interconnecting cables in the 9330 subsystem.

CONNECTOR	CABLE	DATAPOINT PART NUMBER
J1	Disk drive control	10-3372-6002 3M
J5	Tape drive control	10-3431-6002 3M
J2, J3	Data signals	10-3428-6002 3M
J6	Status lights	10-3428-6002 3M
J7	Write protect lines	10-3428-6002 3M
P1	Power supply	10-350431-1 AMP
P2, P3	Host interface	10-65823-073 BERG

Environmental Requirements

Temperature and humidity specifications

REQUIREMENT	SPECIFICATION
Operating temperature	50° to 104° Fahrenheit 10° to 40° Celsius
Storage temperature	14° to 122° Fahrenheit -10° to 50° Celsius
Relative humidity	20 to 80% noncondensing
Altitude	-983 feet to 9830 feet

Environmental conformance

DESCRIPTION	REGULATION
Acoustic level	DATAPOINT PNC 45
Radiated and conducted emissions	FCC part 15, subpart J, with proper installation, Class A, VDE pending
Regulatory compliance	UL 478, CSA C 22.2 No. 154-1975

Chapter 3. CONTROLLER

Overview

Introduction

The 9330 subsystem controller is designed for maximum flexibility of system configuration. It provides communications and controls between the DATAPOINT Micro Bus interface to the host processor and the subsystem's components. It also displays subsystem status information for the operator.

Contents

This chapter provides:

- functional descriptions of the controller components,
- detailed definitions of connector pins and signals,
- controller configuration jumper information,
- controller address jumper designations, and
- a multiple controller configuration diagram.

Comment

The 9330 subsystem controller can be connected to any Micro Bus-based DATAPOINT processor. The host processor must support interrupts and parity to fully employ the subsystem's features.

Controller Functional Description

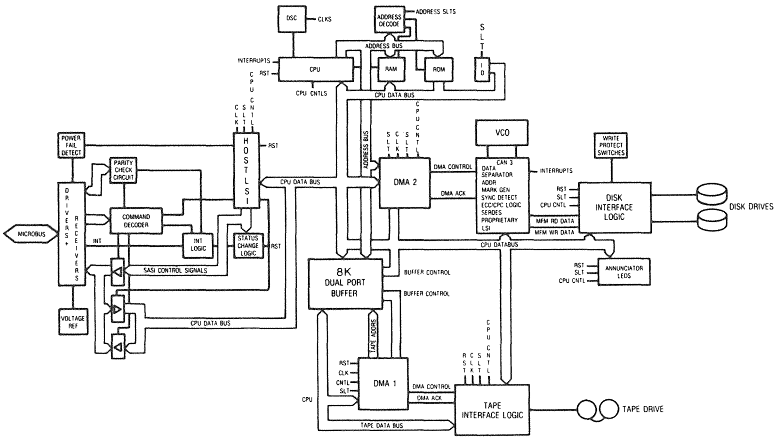
Introduction

The subsystem controller consists of six logical functions:

- the host interface,
- data buffers,
- disk drive interface,
- tape drive interface,
- microprocessor, and
- status light interface.

Each function is described in the following paragraphs.

Subsystem block diagram



Host interface

The Micro Bus interface transfers commands and data from the host to the disk or tape and returns status and data to the host from the disk or tape. It consists of the:

- line drivers and receivers,
- command decoder,
- status and interrupt logic, and
- power fail logic.

Interface functions

THE MICRO BUS CONTAINS...	THAT...
an interface	is a set of drivers, receivers, and latches that provide the electrical interface and timing synchronization between the host and the controller data bus.
a command decoder	is a multiplexer that decodes the 4-bit Micro Bus command into eight internal strobes.
status and interrupt logic	contains circuitry to inform the host of controller progress and control interrupts through the interrupt mask written by the host.
power fail logic	terminates any controller operation by holding the controller reset if a supply voltage falls below tolerance or if the Micro Bus +5 V sense line falls below tolerance.

Micro Bus lines

The Micro Bus uses 21 communications lines that include:

- an 8-bit Command/Address bus,
- an 8-bit bidirectional data bus line with parity,
- two strobe lines,
- a parity line,
- an Interrupt Request line, and
- an Interrupt Acknowledge line.

Remote power indicator

The Micro Bus power indication signal allows peripheral devices to slave their power supplies to the host processor. The subsystem can power up whenever the host processor is powered up.

The controller protects against transient conditions when the host system is powered up or down by:

- monitoring the power indication line and
- forcing a reset when it is inactive.

Bus parity

The subsystem interface contains circuitry for checking and generating Micro Bus parity. Parity is checked or generated on all Micro Bus transfers including interrupt acknowledge response.

- If parity is incorrect, the command is ignored and a Micro Bus parity error interrupt is generated.
- Configuration jumpers J8 through 13 can be set to allow operation on processors without parity logic.

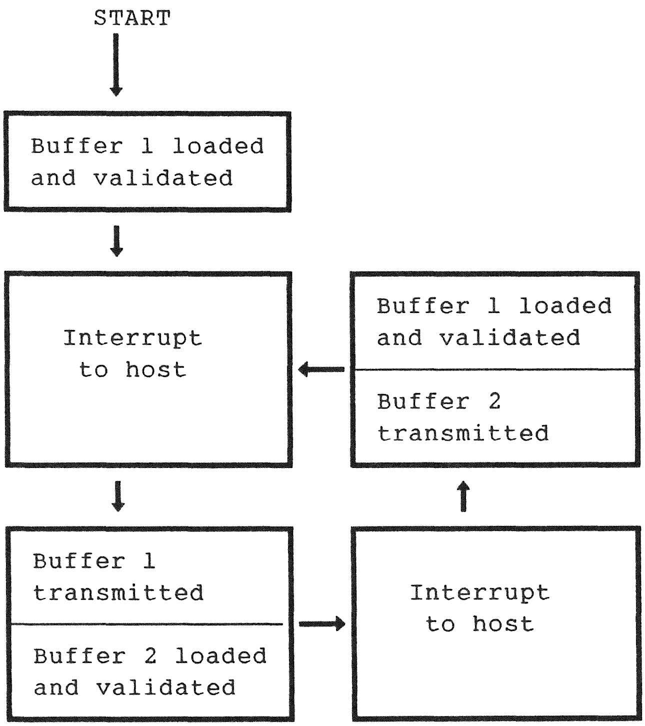
Data buffers

The controller's data buffers are two independent 8 K byte buffers with two 4-channel direct memory access (DMA) devices. Each buffer has its own address and data bus for the simultaneous data transfer between one buffer and the disk drive and the other buffer and the tape drive. This design allows the tape drive to maintain streaming at any interleave factor during disk and tape transfer operations.

The first DMA device controls data transfers between the tape drive and the buffer and between the Micro Bus and the buffer. The second controls data transfers between the disk drive and the buffer.

Controller data buffer operation

The buffers receive and transfer data as shown in the following diagram.



Disk drive interface

The disk drive interface consists of two Large Scale Integration (LSI) devices for operation and data control.

The disk control LSI:

- generates control and status signals to operate the disk and
- controls arm position and head selection.

The disk data LSI services read and write signals over the disk cable. They include:

- data separation,
- address mark generation and detection,
- synchronization detection,
- serial data to byte-wide data conversion, and
- error correction logic.

Tape drive interface

The tape drive interface provides:

- the communication path for microprocessor control of the tape drive,
- control for the data path between the buffer DMA and the tape drive, and
- parallel to serial data conversion.

The tape drive is controlled by the microprocessor through the command bus and strobes. Data flow is controlled with separate read and write strobes.

The microprocessor monitors tape drive operation and status.

Microprocessor

An 8-bit microprocessor oversees controller operation and is the high level command interface to the host. Its features are:

FEATURE	FUNCTION
8 KB RAM	local storage
PROM	controls program storage
Oscillator	provides 4 and 10 MHz clocks for microprocessor operation and controller circuit timing
Address decoder	controls the microprocessor input and output of all controller hardware
Configuration Register	allows the microprocessor to read the user-configurable jumpers

Status light interface

The status light interface provides controller and subsystem communications with the operator through the control panel status lights and the write protect switch.

Power-On-Reset (POR) diagnostics

At power on, the controller executes a set of internal self-tests that include:

- a ROM checksum test,
- a RAM buffer test, and
- an ECC circuit test.

During diagnostic execution, the power status light flashes at a 2-Hz rate. When the diagnostic execution is successful, the power status light stops flashing and remains on.

When the POR diagnostic fails, the power status light continues to flash at a 1-Hz rate. When this occurs, the controller allows any interface command described in Chapter 4, *INTERFACE COMMAND STRUCTURE*, except RESET.

Note:

The POR diagnostic completes execution within 10 seconds of power on.

Comment

When a tape cartridge is present on a disk/tape or tape only subsystem, the power-on diagnostic does not include tests that involve moving the tape.

Error correction circuitry (ECC)

The controller incorporates Error Correction Circuitry (ECC) using a 32-bit algorithm. It can detect burst errors up to 22 bits long and correct burst errors up to 11 bits long. The typical controller error correction time is 50 milliseconds. This is longer than the time needed for one disk revolution. A sector that causes an error is reread on the next disk revolution.

Note:

Disk ID fields incorporate error detection only; no correction is attempted.

Controller Connectors

Introduction

The controller contains all interconnections within the subsystem and to the host. It communicates between the host and the disk or tape drive through the:

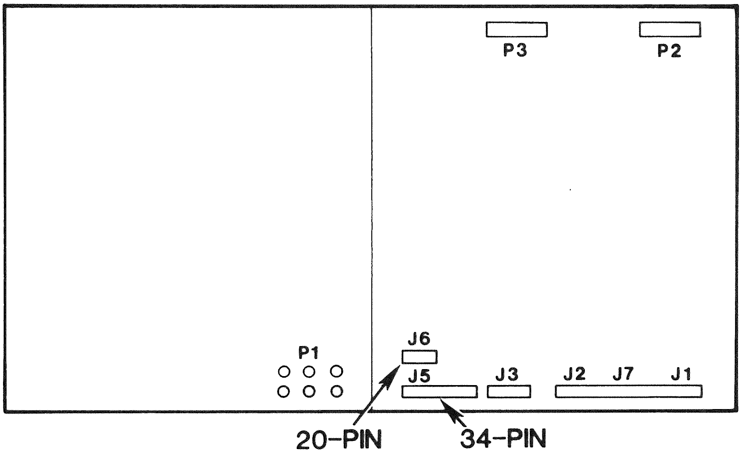
- power supply connector,
- Micro Bus connectors,
- disk drive connector, and
- tape drive connector.

The controller displays operating status information through the LED/Write protect connector.

Comment

The following sections define pins and signals for each of the controller connectors.

Controller connector layout



Power supply connector (P1)

The power supply connector pin assignments are:

PIN	VOLTAGE
1	REMOTE POWER SENSE
2	+12 Volts
3, 4	GROUND RETURN
5, 6	+5 V

Micro Bus connector (P2, P3)

The 26-pin Micro Bus connector pin assignments are:

PIN	SIGNAL NAME	SIGNAL NAME	PIN
1	GROUND	STROBE 1	2
3	GROUND	STROBE 2	4
5	GROUND	INTERRUPT ACKNOWLEDGE	6
7	GROUND	ADDRESS 0	8
9	ADDRESS 1	ADDRESS 2	10
11	ADDRESS 3	COMM 0	12
13	COMM 1	COMM 2	14
15	COMM 3	PARITY	16
17	INTERRUPT REQUEST	DATA 0	18
19	DATA 1	DATA 2	20
21	DATA 3	DATA 4	22
23	DATA 5	DATA 6	24
25	DATA 7	+5 volts (power indication)	26

Disk control connector (J1, J2, J3, J6, J7)

The disk control connector incorporates

- disk expansion lines,
- write protect control lines, and
- disk control lines

into a single 60-pin connector. The following tables list the pin assignments.

Disk drive control lines (J1)

PIN		
SIGNAL	GROUND	SIGNAL NAME
28	27	REDUCED WRITE CURRENT
30	29	HEAD SELECT 2^2
32	31	WRITE GATE
34	33	SEEK COMPLETE
36	35	TRACK 00

Disk drive control lines (J1) continued

PIN		
SIGNAL	GROUND	SIGNAL NAME
38	37	WRITE FAULT
40	39	HEAD SELECT 2^0
42	41	SPARE
44	43	HEAD SELECT 2^1
46	45	INDEX
48	47	READY
50	49	STEP
52	51	DRIVE SELECT 1
54	53	DRIVE SELECT 2
56	55	RESERVED
58	57	KEY
60	59	DIRECTION IN

Note:

REDUCED WRITE CURRENT may be configured as Head Select 2^3.

Disk data lines (J2, J3)

PIN		
SIGNAL	GROUND	SIGNAL NAME
1	2	DRIVE SELECTED
3	4	RESERVED
5	6	KEY
7	8	RESERVED
9	10	SPARE
11	12	GROUND
13		MFM WRITE DATA+
14		MFM WRITE DATA-
15	16	GROUND
17		MFM READ DATA+
18		MFM READ DATA-
19	20	GROUND

Disk LED/Write protect lines (J6)

PIN	SIGNAL
1	-DISK DRIVE 0 LED
2	+DISK DRIVE 0 LED
3	-DISK DRIVE 1 LED
4	+DISK DRIVE 1 LED
5	-CONTROLLER FAULT LED
6	+CONTROLLER FAULT LED
7	-WRITE PROTECT, DRIVE 0, INPUT
8	+WRITE PROTECT, DRIVE 0, LED POWER
9	WRITE PROTECT, DRIVE 0, GROUND
10	SPARE
11 through 14	RESERVED

Disk LED/Write protect lines (J6) continued

PIN	SIGNAL
15	-WRITE PROTECT, DRIVE 1, INPUT
16	+WRITE PROTECT, DRIVE 1, LED POWER
17	WRITE PROTECT, DRIVE 1, GROUND
18	KEY
19 through 20	SPARE

Note:

Connectors J6 and J7 have some duplicated lines. This design facilitates cabling of the disk expansion drive.

Write protect control lines (J7)

PIN	SIGNAL NAME
21	-DISK DRIVE 1 LED
22	+WRITE PROTECT DISK 1 POWER
23	-WRITE PROTECT DISK 1 INPUT
24	DISK TERMINATOR POWER
25	RESERVED
26	RESERVED

Tape drive control lines (J5)

The tape drive pin assignments and signals are shown in the following table.

PIN		
SIGNAL	GROUND	SIGNAL NAME
1		FILE MARK DETECT
2		KEY
3	4	DATA ERROR FLAG
5	6	DEVICE BUSY
7	8	READ DATA
9		READY
10		NOT USED
11	12	INTERRUPT
13	14	READ GATE
15	16	DATA CLOCK
17	18	COMMAND ACKNOWLEDGE

Tape drive control lines (J5) continued

PIN		
SIGNAL	GROUND	SIGNAL NAME
19	20	RESET
21	22	WRITE DATA
23	24	WRITE GATE
25	26	COMMAND BUS 0
27	28	COMMAND GATE
29	30	COMMAND BUS 1
31	32	COMMAND BUS 3
33	34	COMMAND BUS 2

Controller Configuration Jumpers

Introduction

The disk/tape subsystem controller supports 1 of 16 device addresses. The jumpers correspond to the Micro Bus address bits Address 0 through Address 3. When more than one device is used on the Micro Bus, each device must have a unique address. This section discusses the:

- configuration jumpers, and
- Micro Bus address jumpers.

Configuration jumpers (J8)

A 16-position configuration header designates:

- Micro Bus address selection,
- parity enable,
- diagnostic loopback,
- tape drive present, and
- drive type.

Configuration jumper assignments

The jumpers are in a 3 X 16 header with a storage location for each jumper not in use. Jumper assignments are shown in the following table.

PIN	SIGNAL NAME	DESCRIPTION
	CONTROLLER	
1	Address 3	Micro Bus address jumpers
2	Address 2	
3	Address 1	
4	Address 0	
	CONFIGURATION BYTE	
5	Bit 7	Drive type for Drive 1
6	Bit 6	
7	Bit 5	
8	Bit 4	
9	Bit 3	Drive type for Drive 0
10	Bit 2	
11	Bit 1	
12	Bit 0	

HEX...	SHOWS...
00	no disk
01	28 MB
02	65 MB

Note:
The configuration byte is read in byte 2, Request Controller Type command.

Configuration jumper assignments continued

PIN	SIGNAL NAME	DESCRIPTION
13	Disable Parity	Disable/enable Micro Bus parity checking
14	Diagnostic Loopback	Forces continuous execution of power-on diagnostics
15	Tape Drive Present	1 = tape drive present 0 = no tape drive present Can be read in byte 3, bit 0 of the Request Controller Type command
16	Spare	Unassigned, but can be read in byte 3, bit 1 of the Request Controller Type command

Controller address jumpers

The factory-installed jumper configuration sets the controller's address to 8. The address jumpers are binary encoded as shown in the following table.

CONTROLLER ADDRESS	J8-1	J8-2	J8-3	J8-4
Address 0	0	0	0	0
Address 3	0	0	1	1
Address 8	1	0	0	0
Address 10	1	0	1	0
Address 13	1	1	0	1

Diagnostic loopback

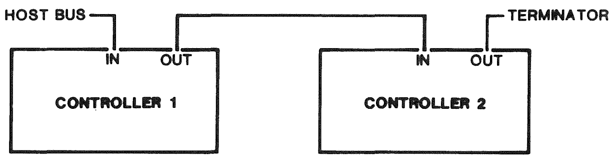
The diagnostic loopback function is invoked by setting jumpers J8 through J14 of the configuration jumpers. This function forces the controller to continually execute the power-on diagnostics. During execution of the loopback test, the power status light flashes at a 1-Hz rate. If a failure is detected, the power status light stops flashing and remains off.

Multiple Controllers

Introduction

A separate controller is required for each additional pair of disk drives or for an additional tape drive. The Micro Bus terminator is positioned on the last controller when multiple controllers are used. The following figure shows a system configured with two controllers.

Multiple controllers



Chapter 4. INTERFACE COMMAND STRUCTURE

Overview

Introduction

The disk/tape controller drives a DATAPOINT Micro Bus and can control two disk drives and a streaming tape drive.

The Controller/Micro Bus software interface consists of host software used to perform I/O operations with the 9330 subsystem. Programming for host/subsystem communications is primarily concerned with the Micro Bus lines that make up the:

- 8-bit Command/Address bus,
- 8-bit bidirectional data bus,
- Interrupt Request line, and
- Interrupt Acknowledge line.

Contents

The Controller/Micro Bus software interface is discussed in sections about the:

- interface command summary and format,
- interface status latch,
- interrupts,
- command transfer sequence,
- abnormal termination errors, and
- command formats.

Abbreviations

The following abbreviations are used in this chapter:

- LUN--Logical Unit Number,
- MSB--Most Significant Byte,
- MMSB--Middle Most Significant Byte, and
- LSB--Least Significant Byte.

Interface command summary

The eight acceptable commands are transferred to the interface using command bits COM0 through COM3 and Strobe 1. Strobe 2 is not used by the interface. The commands, their bit patterns, and their descriptions are shown in the following table.

USE THIS COMMAND...	WITH COM BITS... 3	2	1	0	TO...
READ DATA	0	1	1	1	put the interface in data mode to read data, command, or status from the controller.
ENABLE INT	0	1	1	0	clear the interrupt mask.
DISABLE INT	0	1	0	1	set the interrupt mask. Note: DISABLE INT disables all interrupts except Micro Bus parity error and NOREQ to the host. Interrupts are disabled at power up and at software reset. Polling can be done by disabling interrupts and reading the interface status latch.

Interface command summary continued

USE THIS COMMAND...	WITH COM BITS... 3	2	1	0	TO...
SELECT	0	1	0	0	set the select latch to indicate a command transaction beginning. The controller clears the select latch when it is ready to accept a command.
ID BYTE READ	0	0	1	1	return Hex 08 to identify this device in the Micro Bus address space.
WRITE DATA	0	0	1	0	put the interface in data mode to write data to or from the controller.
ISTATUS READ	0	0	0	1	allow the host to read the interface status latch. Status is returned on the Micro Bus data lines.

Interface command summary continued

USE THIS COMMAND...	WITH COM BITS... 3	2	1	0	TO...
RESET	0	0	0	0	cause a software reset without power down. This resets the controller to the initialized state. Note: RESET does not force diagnostic execution or affect tape or disk drives. The host must wait 20 usec after RESET to issue SELECT.

Interface command format

The 8-bit command/address bus transfers four address bits and four command bits from host to controller. Each command must contain the device address set by configuration jumpers 0 through 3. The command/address format is:

BIT...	7	6	5	4	3	2	1	0
IS...	COM3	COM2	COM1	COM0	AD3	AD2	AD1	AD0

Interface Status Latch

Introduction

The host processor determines controller status by reading the interface status latch on the Micro Bus data lines. The interface status latch is an 8-bit word the host reads by issuing an ISTATUS command. Bits, signals, and their descriptions are given in the following table.

BIT	SIGNAL	NAME AND DESCRIPTION
07	I/-O	Input/-Output indicates the transfer direction across the host bus. • If set to 1, transfer is from controller to host. • If reset to 0, transfer is from host to controller. • Valid only when REQ is set to 1.
06	BUSY	Busy is the controller acknowledgement to the host in response to SELECT. It indicates the controller is active or in use. The controller cannot be interrupted when BUSY is set to 1.
05	C/-D	Command/-Data indicates which information is to be transferred. • If set to 1, command or status information is transferred. • If reset to 0, data is transferred. • Valid only when REQ is set to 1.

Interface status latch continued

BIT	SIGNAL	NAME AND DESCRIPTION
04	MSG	When Message is set to 1, a status byte transfer is pending. MSG, REQ, and O/-O must all be set to 1 to transfer the byte that indicates the operation end. When complete, the controller resets all interface signal lines and returns to the idle state. MSG is valid only when REQ is set to 1.
03	REQ	Request validates I/-O, C/-D, and MSG. A 1 from the controller indicates an 8-bit byte is to be transferred on the host bus.
02	NOREQ	Invalid Request is set to 1 by the controller if a Micro Bus read or write command is received and REQ is reset to 0.
01	uPARERR	Micro Bus Parity Error indicates a parity error on the host bus. When an error is detected, the command is ignored, the status bit is set to 1, and an interrupt is generated.

Interface status latch continued

BIT	SIGNAL	NAME AND DESCRIPTION
00	POR	POR Diagnostics in Progress indicates: • the controller is executing POR diagnostics or • POR diagnostics have failed. Note: POR is set to 1 while the tests are running and reset to 0 on successful test completion.

Status line decode

Additional controller status information is available by decoding bits 4, 5, and 7 of the status latch.

The following table gives the meaning of these decoded bits.

Note:

X = don't care

STATUS LINES			
BIT 7	BIT 5	BIT 4	
I/-O	C/-D	MSG	MEANING
0	1	0	next command
0	0	X	output data
1	0	X	input data
1	1	0	completion byte
1	1	1	message byte

Interrupts

Introduction

The controller uses the interrupt features of the DATAPOINT Micro Bus. Interrupts generated by the controller are sent to the host over the interrupt request line (Micro Bus pin 17). Controller interrupts can occur on three occasions:

- during the command transfer phase,
- on completion of an operation, and
- when an error occurs that requires host service.

Interrupt sequence

The interrupt sequence is described in the following table.

STAGE	PROCESS
1	The controller sends an interrupt request to the host.
2	The host responds with an interrupt acknowledge.
3	The controller transfers an interrupt response byte.
4	The interrupt request and the interrupt response byte are cleared on completion of the host interrupt cycle.

Interrupt response byte

BIT	DEFINITION
07	I/-O (See the note with Bit 04.)
06	Micro Bus Parity Error/Invalid Request • This interrupt occurs when the controller detects a Micro Bus parity error or a Micro Bus read or write command is received when REQ is reset to 0. The Interface status latch shows that error caused the interrupt. • The controller need not be addressed to generate a Micro Bus Parity error; multiple units may respond to this error.
05	C/-D (See the note with Bit 04.)
04	MSG Note: MSG, C/-D (Bit 05), and I/-O (Bit 07) must be decoded to obtain the proper interrupt response that shows the controller state. See *Status line decode*.
03	CONTROLLER ADDRESS 03
02	CONTROLLER ADDRESS 02
01	CONTROLLER ADDRESS 01
00	CONTROLLER ADDRESS 00

Command Transfer Sequence

Introduction

All diagnostic, disk, and tape commands must follow the command sequence described in this section. When the host has a command for the controller, the sequence of events required to cause execution is:

- A SELECT command is issued.
- Command is fetch,
- Data is transferred if required, and
- A completion satus byte is generated.

Command execution begins after the command fetch sequence is completed. The following paragraphs describe these events.

Select phase

When a reset has occurred or a command completed, the controller is in the idle state. The command sequence begins when the host issues a SELECT command. The host then waits for a busy status from the controller.

Comment

The host must wait 20 microseconds following a RESET before issuing a SELECT command.

Command fetch phase

During the command fetch phase, the following sequence of events occurs:

STAGE	PROCESS
1	The controller sets C/-D to 1 to indicate a command mode transfer and resets I/-O to 0 to indicate output from the host.
2	The controller generates an interrupt to the host indicating that the controller is ready to accept the command transfer.
3	The command is transferred by the host placing the first byte of the command on the Micro Bus data lines and issuing a WRITE DATA command.
4	The controller sets REQ to 1 to request the next byte of the command.
5	The command transfer continues with the host placing the next command byte on the data lines and issuing the WRITE DATA command.
6	When the six command bytes have been transferred, the command byte fetch phase ends after REQ is reset to 0.

Data transfer phase

A read or write data transfer requires the following steps.

STAGE	PROCESS
1	The controller resets C/-D to 0 to indicate data mode.
2	I/-O on the host bus is: • set to 1 for data transfer from controller to host or • reset to 0 for data transfer from host to controller.
3	The controller generates an interrupt to the host, indicating that the controller is ready for the data transfer.
4	Data is transferred by the host placing the first byte of data on the Micro Bus data lines and issuing a WRITE DATA command.
5	The controller exits the data transfer phase when all bytes have been transferred.

Comment

If a command requires no data transfer, this phase is omitted.

Completion status byte phase

After all data bytes have been transferred, a completion status byte is placed on the data bus by the controller. The following table describes the sequence of events.

STAGE	PROCESS
1	The controller • sets C/-D to 1, indicating status information is to be transferred, • sets I/-O to 1, indicating transfer is from controller to host, and • resets MSG to 0.
2	The controller generates an interrupt to the host to identify the status phase.
3	The host reads the status byte using the READ DATA command.

Message byte phase

After the status byte is transferred, the controller places the message byte on the data bus. The following table describes the sequence of events.

STAGE	PROCESS
1	The controller: • sets C/-D to 1, indicating that status information is to be transferred, • sets I/-O to 1, indicating transfer is from controller to host, • sets MSG to 1, indicating the message byte transfer is pending, and • sets REQ to 1, validating C/-D, I/-O, and MSG.
2	The controller generates an interrupt to the host to identify the message phase.
3	The host reads the message byte using the READ DATA command.
4	After the message byte is read, the controller resets REQ, BUSY, and all other lines to 0. This completes the status and message phase with the controller ready for the next command.

Nonrecoverable errors

A nonrecoverable error during the command transfer sequence causes the controller to immediately terminate the command and enter the status and message phase. Error conditions and their results are given in the following table.

ANY...	CAUSES...	AND IS REPORTED IN THE...	WITH ERROR CODE IN THE...
Micro Bus parity error	an interrupt to the host, command termination,	interrupt response byte	status byte.
drive interface or controller error	an interrupt to the host, command termination,	completion status byte	message byte.
read/write channel error	the command to terminate in the normal manner,	completion status byte	message byte.

Note:

A sector or more of data may be transferred before an error is reported.

Comments

For drive interface, controller-related, or read/write channel errors, a Request Sense command can be issued to retrieve error information. If the host does not care about errors, any other command can be issued.

Multiple recoverable errors can occur during a multisector I/O operation. In this event, only the location and cause of the last error that occurred is reported.

Multisector transfer errors

All 256 bytes of a sector are transferred or none are transferred. A partial sector is never transferred.

If an error occurs during a multisector data transfer, the Request Sense Status (HEX 03) returns the logical address of the failed sector.

Caution

The recoverable error status information is lost when a nonrecoverable error occurs after a recoverable error.

Command Formats

Introduction

The host sends a six-byte block to the controller to specify the controller operation. The following figure shows the block composition.

Command block format

BIT	7	6	5	4	3	2	1	0
BYTE 0	Bits 0 through 7 contain the opcode.							
BYTE 01	Bit 7 is always 0.	Bits 5 through 6 contain the LUN. Only 00 and 01 are valid for disk; 10 specifies tape.		Bits 0 through 4 contain the disk address MSB.				
BYTE 02	Bits 0 through 7 contain the middle disk address.							
BYTE 03	Bits 0 through 7 contain the disk address LSB.							
BYTE 04	Bits 0 through 7 specify interleave or sector count.							
BYTE 05	Bits 0 through 7 contain the control field. See *Control field byte options*.							

Control field byte options

The control field (byte 5) of the command block contains optional flags defining controller command processing.

The block format is:

BIT	7	6	5	4	3	2	1	0
BYTE 05	Disable Retry = 1	Disable ECC	Alternate Format Pattern = 1	Always 0				

The control field bits are defined in the following paragraphs.

Disable retry
Bit 7

The disable retry bit determines whether or not the controller attempts to retry a command. Tape commands are retried as specified for the particular command. The bit is interpreted as shown in the following table.

IF THE DISABLE RETRY BIT IS...	THEN..
set to 1,	the controller does not attempt to retry the command.
reset to 0,	the controller tries a disk command 4 times before reporting an error.

Note:

After a successful retry, the error is reported as correctable.

Disable ECC error correction
Bit 6

The Disable ECC error correction bit is interpreted as shown in the following tables.

IF...	THEN...
the Disable ECC error correction bit is set to 1 when a data error occurs,	the controller does not attempt to correct the data read from the disk.
a data error occurs during a multisector transfer,	the command is aborted and a nonrecoverable error is reported. The sector containing the error is not transferred.
the Disable ECC error correction bit is reset to 0 when a recoverable data error occurs,	the controller attempts one reread.
the reread is successful,	error code 1B is flagged to indicate a reread was required.

Note:

Data that is ECC-recoverable is corrected and transferred to the host with a flag for an ECC-recoverable error.

Disable ECC error correction continued
Bit 6

IF...	THEN...
an error persists after the reread,	ECC is applied.
an error persists during a multisector transfer,	transfer is continued. Errors are reported on transfer completion.
the error persists after 4 ECC tries,	the command is aborted and a nonrecoverable-ECC error is reported.

Note:

The sector containing the error is not transferred.

Comment

The Disable ECC error correction and Disable retry bits perform independent functions and can be used in any combination.

When Disable ECC error correction and Disable retry are not reset to 0, ECC correction has priority over retries. If a recoverable data error occurs, it is corrected and returned rather that retried.

Alternate format pattern
Bit 5

The Alternate Format Pattern bit is interpreted as shown in the following table.

WHEN THE ALTERNATE FORMAT PATTERN BIT IS...	THEN...
set to 1 during a format command,	the command uses the data field format pattern previously written to the controller buffer using the Write buffer command.
reset to 0,	the default pattern HEX 6C is used.

Completion status byte

The completion status byte is returned on command execution completion. It contains status information on the command execution.

The byte format is:

BIT...	7	6	5	4	3	2	1	0
CONTAINS...	0	D	D	0	WP	0	ERR	0

Bit descriptions

<table>
<tr><th>BIT</th><th>DEFINITION</th></tr>
<tr><td>7</td><td>always 0</td></tr>
<tr><td>6, 5</td><td>the logical unit number of the drive. They are interpreted as:
<table>
<tr><th>BIT</th><th>6</th><th>5</th></tr>
<tr><td>DISK DRIVE 00</td><td>0</td><td>0</td></tr>
<tr><td>DISK DRIVE 01</td><td>0</td><td>1</td></tr>
<tr><td>TAPE DRIVE</td><td>1</td><td>0</td></tr>
</table>
<u>Note:</u>
If both tape and disk drives are active, the LUN specifies the tape drive.</td></tr>
<tr><td>4</td><td>always 0</td></tr>
<tr><td>3</td><td>when set to 1, the selected disk drive is write protected</td></tr>
<tr><td>2</td><td>always 0</td></tr>
<tr><td>1</td><td>when set to 1, an error has occurred during command execution</td></tr>
<tr><td>0</td><td>always 0</td></tr>
</table>

Message byte

The message byte is transferred to the host after the completion status byte has been transferred.

- If a command execution error occurred, the message byte contains the error code.
- If no error occurred, all bits are reset to 0.
- If multiple errors occurred, only the last occurring error is reported.

The byte format is:

BIT...	7	6	5	4	3	2	1	0
CONTAINS...	Bit 7 is set to 1 if an error was flagged in the completion status byte. This indicates that the Request Sense Status block contains valid error information.	Bits 0 through 6 contain the error code when an error occurred. When no error has occurred, bits 0 through 6 contain 0.						

Chapter 5.
DISK COMMAND STRUCTURE

Overview

Introduction

This chapter contains detailed programming information about the fixed disk commands. Both the command name and its equivalent HEX code are given.

Abbreviations

The following abbreviations are used in this chapter:

- LUN--Logical Unit Number,
- DCF--Device Command Field,
- MSB--Most Significant Byte,
- MMSB--Middle Most Significant Byte, and
- LSB--Least Significant Byte.

Summary of disk commands

The following table provides a summary of the 20 disk commands and includes a list of the corresponding HEX control codes. The commands are listed in ascending numerical order.

COMMAND	HEX CODE
Test Drive Ready	00
Recalibrate	01
Request Sense Status	03
Format Drive	04
Check Track Format	05
Format Track	06
Format Bad Track	07
Read Data	08
Read Sectors No Data Transfer	09
Write Data	0A

Summary of disk commands continued

COMMAND	HEX CODE
Seek	0B
Initialize Drive Characteristics	0C
Read ECC Burst Error Length	0D
Format Alternate Track	0E
Write Sector Buffer	0F
Read Sector Buffer	10
Request Controller Type	11
Read Identifier Physical	12
Read Identifier Logical	13
Check Track ECC	14

Organization of disk commands

The following table shows the disk commands grouped according to logical functions.

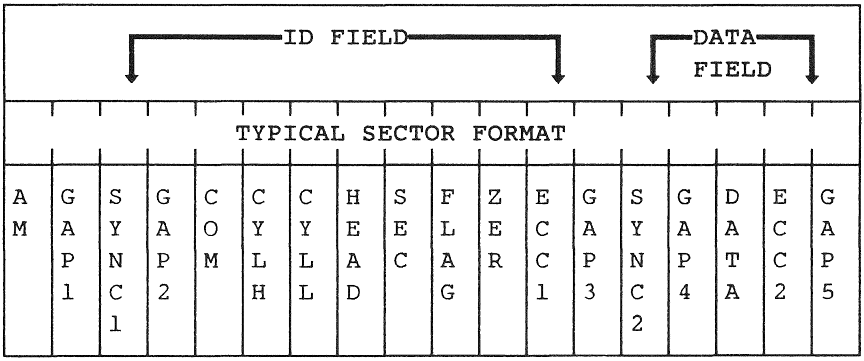
LOGICAL FUNCTION	COMMAND LABEL	HEX CODE
Disk and Controller	Test Drive Ready	00
	Request Sense Status	03
	Initialize Drive Characteristics	0C
	Request Controller Type	11
Disk	Format Drive	04
	Recalibrate	01
Buffer	Write Sector Buffer	0F
	Read Sector Buffer	10
Track and format	Check Track Format	05
	Format Track	06
	Format Bad Track	07
	Format Alternate Track	0E

Organization of disk commands continued

LOGICAL FUNCTION	COMMAND LABEL	HEX CODE
Disk read, write, and seek functions	Seek	0B
	Read Data	08
	Read Sectors No Data Transfer	09
	Write Data	0A
	Read ECC Burst Error Length	0D
	Read Identifier Physical	12
	Read Identifier Logical	13
	Check Track ECC	14

Disk sector format

The 9330 subsystem disk is formatted with 256 byte sectors. The following diagram shows the disk sector format and the names of the fields of the information traveling over the controller-disk interface.



Sector field descriptions

FIELD	BYTES	DESCRIPTION
AM	4	Address mark
GAP1	5	Zero gap (0)
SYNC1	1	ID sync byte (HEX 01)
GAP2	2	ID zero byte gap (0)
COM	1	ID compare byte (HEX C2)
CYLH	1	Cylinder high (MSB)
CYLL	1	Cylinder low (LSB)
HEAD	1	Head number
SEC	1	Sector number
FLAG	1	Flag byte
ZER	1	Zero byte
ECC1	4	ID ECC byte
GAP3	16	Zero byte gap
ECC2	4	Data field ECC bytes
GAP5	21	Interrecord zero gap (0)
TOTAL	323	256 byte sector

Sector interleaving

Variable sector interleaving is supported by the controller. Interleaving allows logical contiguous sectors of data on a given track to be mapped onto nonadjacent physical sectors. When any format command is issued, any interleave value up to the number of sectors minus one must be passed in the command string. The interleave factor can be adjusted for maximum system performance.

Interleaving example

An interleave factor of five causes every fifth physical sector to be transferred as the next contiguous logical data sector. <u>This does not mean that five sectors of data are transferred in one revolution.</u>

If the operation is READ and the interleave factor is five, then:

- a sector of data is read into the sector buffer, and
- while the heads are passing over the next four physical sectors of the disk,
- the data is being transferred to the host.

If the host cannot transfer the full sector of data during the four sector times available,

- the controller must wait a full revolution before the next logical sector can be read from the disk.

If this happens, the interleave factor is too low and should be increased until an increase in system operating speed is apparent.

Track format example

The following table shows an example of 32 sectors per track with an interleave factor of 5.

PHYSICAL SECTOR	LOGICAL SECTOR	PHYSICAL SECTOR	LOGICAL SECTOR	PHYSICAL SECTOR	LOGICAL SECTOR	PHYSICAL SECTOR	LOGICAL SECTOR	PHYSICAL SECTOR	LOGICAL SECTOR	PHYSICAL SECTOR	LOGICAL SECTOR	PHYSICAL SECTOR	LOGICAL SECTOR
0	0	5	1	10	2	15	3	20	4	25	5	30	6
1	13	6	14	11	15	16	16	21	17	26	18	31	19
2	26	7	27	12	28	17	29	22	30	27	31		
3	7	8	8	13	9	18	10	23	11	28	12		
4	20	9	21	14	22	19	23	24	24	29	25		

Comment

To take full advantage of the controller interleaving feature, the operating system should perform multiple sector data transfers. If single sector transfers are employed, the difference in speed with various interleave factors may not be dramatic.

Logical address calculation

The logical address of a disk sector is calculated by the formula:

```
LOGICAL ADDRESS = ((CYADR X HDCYL) + HDADR) X SETRK + SEADR
```

Where: CYADR = Cylinder address

HDCYL = Number of heads per cylinder

HDADR = Head address

SETRK = Number of Sectors per track

SEADR = Sector address

Disk defect map

The next to last cylinder on the disk drive is dedicated for defect map use. The defect map occupies one track on the cylinder, and the remaining tracks contain copies of the defect map. Defects on the defect-cylinder are ignored when formatting or writing to that cylinder; however, when a sector is unreadable due to a defect, the information is available from the same sector number in any of the defect map copies. A defect-map sector contains:

- a four-byte header that includes the hash value and the number of entries in the sector and
- up to 63 four-byte entries.

Header

A valid defect map sector begins with a four-byte header in the following format.

BYTE	DESCRIPTION
0	Hash value LSB
1	Hash value MSB
2	Hash value MMSB
3	Number of defect entries in this sector

Note:

Hash values are calculated over bytes 3 through 255 for a total of 253 bytes. This includes all bytes except those containing the hash value.

Defect map entries

A defect-map entry is a four-byte block as shown in the following table.

BYTE	DESCRIPTION
0	Bad linear track number MMSB
1	Bad linear track number MSB
2	Bad linear track number LSB
3	Error flag

Note:

The linear track number is calculated as follows:

LINEAR TRACK NO. = (CYLINDER NO. X NUMBER OF HEADS) + HEAD NO.

Linear track calculation examples

LINEAR TRACK NO.	CYLINDER NO.					
	0	1	2	3	. . .	n
Surface 0	1	4	8	12	. . .	n
Surface 1	2	5	9	13	. . .	n
Surface 2	3	6	10	14	. . .	n
Surface 3	4	7	11	15	. . .	n

Error flag (byte 03)

The error flag (byte 03) is defined as follows:

BIT	DESCRIPTION
0	Set to 1, entire track is faulty
1	Set to 1, defect found by DATAPOINT diagnostic
2	Set to 1, defect found by DATAPOINT operating system
3	Not used

Error Codes

See Chapter 6, *DISK DRIVE ERROR CODES*, for information on error codes and alternate track handling.

Test Drive Ready (HEX 00)

Description

The Test Drive Ready command selects the addressed device and tests it for readiness.

The logical unit number (LUN) specifies the selected device, and the device state is returned in the Completion status byte. The following table describes the Completion status byte.

IF...	THEN...
all status bits are in the correct state,	• the error flag is reset to 0, and • the message byte contains 00.
any status bits are not correct,	• the error flag is set to 1, and • the message byte contains the error code.

Note:

The error flag is in the Completion byte.

HEX 00 device command field

The following diagram illustrates the HEX 00 DCF format.

Note:

dd = drive LUN

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	0	0	0	0	0
BYTE 1	0	d	d	NOT USED				
BYTE 2	NOT USED							
BYTE 3	NOT USED							
BYTE 4	NOT USED							
BYTE 5	NOT USED							

Recalibrate (HEX 01)

Description

The drive specified by the LUN moves the drive arm to the track 0 position. The Recalibrate command steps the drive one cylinder at a time while waiting for the track 0 signal from the drive to become active. If an error occurs when retries are enabled, the controller is automatically recalibrated.

Purpose

Use the Recalibrate command solely to correct a drive position error, since the command is slower than a direct seek to track 0.

HEX 01 device command field

The following diagram illustrates the HEX 01 DCF format.

Note:

dd = drive LUN
r = retries

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	0	0	0	0	1
BYTE 1	0	d	d	NOT USED				
BYTE 2	NOT USED							
BYTE 3	NOT USED							
BYTE 4	NOT USED							
BYTE 5	r	0	0	0	0	0	0	0

Request Sense Status (HEX 03)

Description

The Request Sense Status command causes the controller to transfer four bytes of drive and controller status to the host. The host can issue this command to obtain more detailed error information when the completion status byte indicates an error.

HEX 03 device command field

The following diagram illustrates the HEX 03 DCF format.

Note:

dd = drive LUN

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	0	0	0	1	1
BYTE 1		0	d	d		NOT USED		
BYTE 2				NOT USED				
BYTE 3				NOT USED				
BYTE 4				NOT USED				
BYTE 5				NOT USED				

HEX 03 error status field

When an error occurs on a multisector data transfer, the Request Sense Status returns the failed sector's logical address in bytes 1 through 3. The following diagram illustrates the format of the returned information.

Note:

dd = drive LUN

BIT	7	6	5	4	3	2	1	0
BYTE 0	ERROR CODE							
BYTE 1	0	d	d	LOGICAL ADDR 2 (MSB)				
BYTE 2	LOGICAL ADDR 1							
BYTE 3	LOGICAL ADDR 0 (LSB)							

Error status field description

The following table describes the contents of the error status field.

IF HEX 03 IS ISSUED AFTER ANY FORMAT OR CHECK TRACK FORMAT COMMAND...	THEN...
and no error occurred,	the logical address returned by the controller points to one sector beyond the last track formatted or checked.
and an error occurred,	the logical address returned points to the track and sector in error.

Error code format
Byte 0

The following diagram illustrates the format of the error code returned in byte 0 by the Request Sense Status command.

Note:

```
c = error code
v = valid bit
```

BIT	7	6	5	4	3	2	1	0
NAME	v	0	c	c	c	c	c	c
BYTE 00			ERROR CODE					
		ALWAYS 0						
	ADDRESS VALID							

Address valid
Bit 7

The Address valid bit indicates that the logical sector address in bytes 1, 2, and 3 is the sector address where the error occurred. If the failed command does not require a logical block address, the Address valid bit is reset to 0.

Format Drive (HEX 04)

Description

The Format Drive command causes the drive specified by the LUN to be formatted with the interleave factor in byte 4 of the command. All cylinders are formatted from the specified logical sector number (LSN) through the maximum cylinder address specified by the Assign Drive Characteristics command.

Formatting process

When a Format Drive command is issued, the process is:

<table>
<tr><th>STAGE</th><th>DESCRIPTION</th></tr>
<tr><td>1</td><td>The specified drive is recalibrated.</td></tr>
<tr><td>2</td><td>A seek is made to the address specified in bytes 1, 2, and 3 of the command string.</td></tr>
<tr><td>3</td><td>The spindle speed is timed. Allowable spindle speeds are as follows:
<table>
<tr><th>FOR...</th><th>THE SPEED IS...</th></tr>
<tr><td>embedded servo drives,</td><td>3536 RPM ±3%.</td></tr>
<tr><td>nonembedded drives,</td><td>3600 RPM ±3%.</td></tr>
<tr><td colspan="2">After format, all drives must maintain a ±1% speed tolerance.</td></tr>
</table>
</td></tr>
<tr><td>4</td><td>The track is divided into equal sectors; header field is written out for all sectors.</td></tr>
<tr><td>5</td><td>The controller reads back the header field and writes the data field for all sectors.</td></tr>
<tr><td>6</td><td>The data pattern is defaulted to HEX 6C unless the host specifies a pattern in the Write Sector Buffer command by setting bit 5 in the control field.</td></tr>
</table>

Comment

All format operations start at the first sector of the specified track, even when the address specified in the command did not point to the track boundary.

HEX 04 device command field

The following diagram illustrates the HEX 04 DCF format.

Note:

```
dd = drive LUN
r  = retries
X  = don't care
p  = 0 format data pattern default to HEX 6C;
     1 format data pattern specified in disk
     buffer
```

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	0	0	1	0	0
BYTE 1	0	d	d	X	LOGICAL ADDR 2(MSB)			
BYTE 2	LOGICAL ADDR 1							
BYTE 3	LOGICAL ADDR 0 (LSB)							
BYTE 4	0	0	0	INTERLEAVE				
BYTE 5	r	0	p	0	0	0	0	0

Note:

The controller changes the disk buffer contents only when byte 5, bit 5 is reset to 0.

Hard errors while formatting

If the Format Drive command encounters a hard error while formatting a track, the format operation stops and the error is reported. To continue formatting, the host software must provide the data fields for all logical sectors following the sector in error using the Write Data command and resume formatting with the Format Disk command at the beginning of the next track.

Check Track Format (HEX 05)

Description

The Check Track Format command causes a track to be checked against a specified format.

- The track to be checked is specified by the logical sector address in bytes 1 through 3 of the command.
- The format to be matched is specified by the interleave factor located in byte 4 of the command.

Check track format process

The following table describes the stages of the Check Track Format command:

STAGE	ACTION
1	The command generates an interleave table.
2	A seek is made to the target track without recalibrating.
3	Each sector ID is read.
4	While reading, the controller checks for a bad ID field and compares the current sector address with the value generated in the interleave table. <u>Note:</u> Reading begins with the first sector after the index. The controller does not read the data field.
5	If a discrepancy occurs between the sector number read and the value in the interleave table, the controller reports the error.

HEX 05 device command field

The following diagram illustrates the HEX 05 DCF format.

Note:

dd = drive LUN
X = don't care
r = retries

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	0	0	1	0	1
BYTE 1	0	d	d	X	LOGICAL ADDR 2 (MSB)			
BYTE 2	LOGICAL ADDR 1							
BYTE 3	LOGICAL ADDR 0 (LSB)							
BYTE 4	0	0	0	INTERLEAVE				
BYTE 5	r	0	0	0	0	0	0	0

Format Track (HEX 06)

Description

The Format Track command causes a specific track to be formatted. The track is specified by the logical sector address in bytes 1 through 3 of the command.

Purpose

The command can be used to clear the defective or alternate track flags or to reformat a specific track. On an unformatted drive, the command will recalibrate, reseek, and format the specified track.

Formatting process

The following table describes the three format track stages.

<table>
<tr><th>STAGE</th><th>DESCRIPTION</th></tr>
<tr><td>1</td><td>A seek without recalibration is made to the track specified by the logical sector address in command bytes 1 through 3.</td></tr>
<tr><td>2</td><td>The controller attempts to verify its position by reading the track ID.
<table>
<tr><th>IF THE CONTROLLER...</th><th>THEN...</th></tr>
<tr><td>can verify its track position,</td><td>the track is formatted.</td></tr>
<tr><td>cannot verify its track position,</td><td>the controller recalibrates, reseeks, and formats the track.</td></tr>
</table></td></tr>
<tr><td>3</td><td>The ID and data field are written using the interleave value specified in byte 4 of the command.</td></tr>
</table>

HEX 06 device command field

The following diagram illustrates the HEX 06 DCF format.

Note:

```
dd = drive LUN
 X = don't care
 r = retries
 p = 0 format data pattern default to 6C;
     1 format data pattern specified in disk
     buffer
```

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	0	0	1	1	0
BYTE 1	0	d	d	X	LOGICAL ADDR 2 (MSB)			
BYTE 2	LOGICAL ADDR 1							
BYTE 3	LOGICAL ADDR 0 (LSB)							
BYTE 4	0	0	0	INTERLEAVE				
BYTE 5	r	0	p	0	0	0	0	0

Format Bad Track (HEX 07)

Description

The Format Bad Track command is the same as the Format Track command except that the bad track flag is set in the ID field (bit 1) of the specified track.

Purpose

The Format Bad Track command is used to set the bad track flag in the ID field. Any subsequent accesses to the track cause an error code 19 to be returned.

HEX 07 device command field

The following diagram illustrates the HEX 07 DCF format.

Note:

dd = drive LUN
X = don't care
r = retries
p = 0 format data pattern default to 6C;
1 format data pattern specified in disk buffer

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	0	0	1	1	1
BYTE 1	0	d	d	X	LOGICAL ADDR 2 (MSB)			
BYTE 2	LOGICAL ADDR 1							
BYTE 3	LOGICAL ADDR 0 (LSB)							
BYTE 4	0	0	0	INTERLEAVE				
BYTE 5	r	0	0	0	0	0	0	0

Read Data (HEX 08)

Description

The Read Data command transfers the number of sectors specified in byte 4 to the host. The logical sector address in bytes 1 through 3 specifies the starting sector.

Number of sectors transferred

From 1 to 256 sectors can be transferred with a single command.

IF...	THEN...
byte 4 is reset to 0,	256 sectors are transferred.
an error occurs during any read operation,	recovery is performed as specified by byte 5 of the command.

HEX 08 device command field

The following diagram illustrates the HEX 08 DCF format.

Note:

```
dd = drive LUN
 X = don't care
 r = retries
 a = ECC correction
```

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	0	1	0	0	0
BYTE 1	0	d	d	X	LOGICAL ADDR 2 (MSB)			
BYTE 2	LOGICAL ADDR 1							
BYTE 3	LOGICAL ADDR 0 (LSB)							
BYTE 4	# OF SECTORS							
BYTE 5	r	a	0	0	0	0	0	0

Read Sectors No Data Transfer (HEX 09)

Description

The Read Sectors No Data Transfer command is identical to the Read command except that no data is transferred. The ID fields and the data fields are checked for errors and the errors are reported.

Note:

This command overwrites the contents of the disk data buffer.

HEX 09 device command field

The following diagram illustrates the HEX 09 DCF format.

Note:

dd = drive LUN
X = don't care
r = retries
a = ECC correction

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	0	1	0	0	1
BYTE 1	0	d	d	X	LOGICAL ADDR 2 (MSB)			
BYTE 2	LOGICAL ADDR 1							
BYTE 3	LOGICAL ADDR 0 (LSB)							
BYTE 4	# OF SECTORS							
BYTE 5	r	a	0	0	0	0	0	0

Write Data (HEX 0A)

Description

The Write Data command writes the number of sectors specified in byte 4 to the selected drive. The logical sector address in bytes 1 through 3 specifies the starting sector.

From 1 to 256 sectors can be written in a single command. If byte 4 is reset to 0, 256 sectors are transferred.

HEX 0A device command field

The following diagram illustrates the HEX 0A DCF format.

Note:

dd = drive LUN
r = retries

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	0	1	0	1	0
BYTE 1	0	d	d	X	LOGICAL ADDR 2 (MSB)			
BYTE 2				LOGICAL ADDR 1				
BYTE 3				LOGICAL ADDR 0 (LSB)				
BYTE 4				BLOCK COUNT				
BYTE 5	r	0	0	0	0	0	0	0

Seek (HEX 0B)

Description

The Seek command causes the device addressed by the LUN to be physically positioned to the cylinder and head as defined by the logical sector address in bytes 1 through 3.

- The cylinder address is automatically computed by the controller.
- Completion status is returned to the host immediately after receipt of the Seek command.
- The seek is not verified until the next read or write operation to the specified drive.

HEX 0B device command field

The following diagram illustrates the HEX 0B DCF format.

Note:

```
dd = drive LUN
 r = retries
```

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	0	1	0	1	1
BYTE 1	0	d	d	LOGICAL ADDR 2 (MSB)				
BYTE 2	LOGICAL ADDR 1							
BYTE 3	LOGICAL ADDR 0 (LSB)							
BYTE 4	NOT USED							
BYTE 5	r	0	0	0	0	0	0	0

Overlapping seeks

Seek commands to both drives can be overlapped. After the controller issues a seek to one drive, it returns with a completion status, not waiting for the drive to complete the seek. The following table describes the conditions indicated by the completion status.

IF THE RETURN STATUS...	THEN...
shows no error,	the seek was issued correctly.
shows an error,	the seek was not issued.

After transferring the status, another command can be issued to either drive. If a new command is received for a drive with an outstanding seek, the controller will wait with BUSY active for the seek to complete before executing any new command except Test Drive Ready. The controller waits a maximum of two seconds for the seek to complete on a drive command following a seek.

Testing for seeks

A series of Test Drive Ready commands can be used with overlapped seeks to determine when a drive is ready for the next command. If the drive is still seeking, the status byte returned after the Test Drive Ready command will contain error code 08.

Initialize Drive Characteristics (HEX 0C)

Description

The Initialize Drive Characteristics command allows you to configure the controller to work with drives that have different capacities. Drives 0 and 1 can be assigned different drive types.

Initialization

After power-up or a reset, the controller rejects all commands requiring disk or tape access until an Initialize Drive Characteristics command is received for the specified drive. This state returns error code 0A.

- Access to drive 0 is rejected until an Initialize Drive Characteristics for drive 0 is executed.
- Access to drive 1 is rejected until an Initialize Drive Characteristics for drive 1 is executed.
- Access to the tape drive is rejected until drive 0 is initialized.

If no disk drive 0 is present, drive 0 must be initialized to gain access to the tape drive. This sequence may be used to determine if a controller power-on reset has occurred since the last drive access.

HEX 0C device command field

The following diagram illustrates the HEX 0C DCF format.

Note:

dd = drive LUN

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	0	1	1	0	0
BYTE 1	0	d	d	NOT USED				
BYTE 2	NOT USED							
BYTE 3	NOT USED							
BYTE 4	NOT USED							
BYTE 5	NOT USED							

Drive parameters

After the host sends the command to the controller, it then sends an eight-byte data block that contains the drive parameters. Two-byte parameters are transferred with the most significant byte (MSB) first. Drive size parameters begin with 1.

Example:

If the drive has 306 cylinders and 6 heads, then the value for the maximum cylinder address is 306 and the value for the maximum number of heads is 6.

Cylinder values

For the reduced write current and increased write precomp parameters, the value specifies the cylinder number where the event occurs (relative to 0).

Example:

If reduced write current is set to 200, write current is reduced from cylinder 200 up.

If the maximum number of heads is eight or more, the reduced write current line on the drive interface is used for head select 2^3. If the maximum number of heads is 7 or less, reduced write current begins at the cylinder specified in bytes 3 and 4 of the parameter block.

Drive parameter data

The following diagram illustrates the drive parameter data returned by the Initialize Drive Characteristics command.

Note:

```
h = maximum number of heads
e = maximum ECC data burst length
v = 0 for nonembedded servo drives;
    1 for embedded servo drives
s = step rate
```

BIT	7	6	5	4	3	2	1	0
BYTE 0	MAXIMUM CYLINDER (MSB)							
BYTE 1	MAXIMUM CYLINDER (LSB)							
BYTE 2	v	0	0	0	h	h	h	h
BYTE 3	REDUCED WRITE CURRENT CYL (MSB)							
BYTE 4	REDUCED WRITE CURRENT CYL (MSB)							
BYTE 5	INCREASE WRITE PRECOMP CYL (MSB)							
BYTE 6	INCREASE WRITE PRECOMP CYL (LSB)							
BYTE 7	s	s	s	s	e	e	e	e

Drive parameter table

BYTE	BIT	DESCRIPTION
0	0-7	The maximum addressable cylinder (MSB).
11	0-7	The maximum addressable cylinder (LSB).
2	0-3	The number of disk read/write heads.
2	7	The embedded servo bit. When set to 1 during a format command, an embedded servo gap is left at the end of each track to accommodate embedded servo disk drives. In the embedded servo format, the controller times the disk rotation, subtracts a 300 ms cap, and formats the track. The controller works with drives having a servo area smaller than 300 ms preceding the index and 40 ms following the leading edge of the index pulse. For nonembedded servo drives, this bit is always reset to 0.
2	6	Always reset to 0.
3,4	0-7	The reduced write current parameter. This drive-dependent function is specified by the drive manufacturer. See the manufacturer's specification for cylinder number. If the maximum number of heads is eight or more, the reduced write current line is used for head select 2^3. If head select 2^3 is required, set reduced write current to maximum cylinder address plus 1.

Drive parameter table continued

<table>
<tr><th>BYTE</th><th>BIT</th><th>DESCRIPTION</th></tr>
<tr><td>5</td><td>0-7</td><td>Increase write precompensation (MSB) with bit 7, specifying the compensation type.
<table>
<tr><th>BIT 7...</th><th>INDICATES...</th></tr>
<tr><td>set to 1</td><td>• 0 ns compensation below the cylinder specified in bytes 5 and 6,
• 10 ns above the specified cylinder.</td></tr>
<tr><td>reset to 0</td><td>• 5 ns compensation below the cylinder specified in bytes 5 and 6,
• 10 ns above the specified cylinder.</td></tr>
</table>
</td></tr>
<tr><td>6</td><td>0-7</td><td>Increased write precompensation (LSB). This drive-dependent function is specified by the manufacturer. See the manufacturer's specification.</td></tr>
<tr><td>7</td><td>0-3</td><td>The maximum ECC data burst length. This defines the length of a burst error in the data field the controller can correct. The burst length is the number of bits from first error bit to last error bit. Recommended maximum ECC burst error length cannot exceed 4 bits. See Read ECC Burst Error Length.</td></tr>
<tr><td>7</td><td>4-7</td><td>Set the step rate for the disk drive.</td></tr>
</table>

Step rates table

The following table lists the supported step rates and their bit patterns.

BIT PATTERN				RATE
7	6	5	4	
0	0	0	0	3 ms
0	0	0	1	Reserved
0	0	1	0	Reserved
0	0	1	1	Reserved
0	1	0	0	200 usec buffered step
0	1	0	1	70 usec buffered step
0	1	1	0	30 usec buffered step
0	1	1	1	15 usec buffered step
1	0	0	0	12 usec buffered step
1 . . . 1	0 . . . 1	0 . . . 1	1 . . . 1	Not used

Read ECC Burst Error Length (HEX 0D)

Description

The Read ECC Burst Error Length command transfers one byte to the host indicating the length of the error corrected. The command is valid only following an ECC correctable data error code (HEX 18).

HEX 0D device command field

The following diagram illustrates the HEX 0D DCF format.

Note:

X = don't care

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	0	1	1	0	1
BYTE 1	X	0	0	NOT USED				
BYTE 2	NOT USED							
BYTE 3	NOT USED							
BYTE 4	NOT USED							
BYTE 5	NOT USED							

How error length is determined

The error length is determined by counting the number of bits between the first and the last bit in error, including the first and the last bits.

Example

Assume the drive is formatted with the default drive data pattern HEX 6C. If the data read back from the disk has an error, then:

					ERROR BURST LENGTH
CORRECT PATTERN	0110	1100	0110	1100	0
READ BACK PATTERN	0111	1100	0110	1100	1
READ BACK PATTERN	0111	1100	1110	1100	6
READ BACK PATTERN	0111	1100	0110	1110	12

Of the 3 error patterns above, the first and second patterns are correctable because the error bit span is less than 11 bits. The third pattern is uncorrectable since it exceeds the controller's correction capability, which is 11 bits.

Format Alternate Track (HEX 0E)

Description

The Format Alternate Track command is used to assign an alternate track address to a bad track so that on any future access to the bad track address, the controller automatically accesses the alternate track. This is accomplished in three steps.

STEP	ACTION
1	The controller issues the Format Alternate Track command with bytes 1 through 3 containing the logical address of the bad track.
2	The controller requests the 3-byte block containing the host-assigned alternate track address.
3	The controller formats the alternate track and the bad track.

Alternate tracks can be assigned to only one level; therefore, an alternate track cannot have another alternate track assigned to it.

HEX 0E device command field

The following diagram illustrates the HEX 0E DCF format.

Note:

```
dd = drive LUN
 r = retries
 p = 0 format data pattern default to 6C;
     1 format data pattern specified in disk
     buffer
```

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	0	1	1	1	0
BYTE 1	0	d	d	LOGICAL ADDR 2 (MSB)				
BYTE 2	LOGICAL ADDR 1							
BYTE 3	LOGICAL ADDR 0 (LSB)							
BYTE 4	0	0	0	INTERLEAVE				
BYTE 5	r	0	p	0	0	0	0	0

Logical address request

The host returns the alternate track address in the following 3-byte format. Sector address is ignored.

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	LOGICAL ADDR 2 (MSB)				
BYTE 1	LOGICAL ADDR 1							
BYTE 2	LOGICAL ADDR 0 (LSB)							

Alternate track formatting

After receiving the command and the assigned alternate address, the controller does the following:

<table>
<tr><th>STAGE</th><th>DESCRIPTION</th></tr>
<tr><td>1</td><td>The controller seeks to the alternate track and:<table>
<tr><th>IF...</th><th>THEN...</th></tr>
<tr><td>the track is an assigned alternate track,</td><td>error code HEX 1D is returned.</td></tr>
<tr><td>the track has the bad track flag set to 1,</td><td>error code HEX 19 is returned.</td></tr>
<tr><td>neither condition exists,</td><td>the controller proceeds to stage 2.</td></tr>
</table></td></tr>
<tr><td>2</td><td>The controller formats the track as an assigned alternate track using the interleave code in byte 4.</td></tr>
<tr><td>3</td><td>The controller seeks to the bad track, sets the alternate track flag, and writes the alternate track address in all sectors on the bad track.</td></tr>
</table>

Comment

Data fields on both the bad track and the alternate track are destroyed. The sector address is ignored, defaulting to Sector 0.

Write Sector Buffer (HEX 0F)

Description

The Write Sector Buffer command transfers a data sector of 256 bytes to the controller to be written into the controller sector buffer.

Purpose

This command can be used to:

- verify buffer operation by writing a test pattern to the controller buffer and reading it back, and
- specify a data pattern other than the HEX 6C default pattern during formatting. The data pattern is written into the buffer, and byte 5, bit 1 is set to 1.

HEX OF device command field

The following diagram illustrates the HEX OF DCF format.

Note:

X = don't care

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	0	1	1	1	1
BYTE 1	X	0	0	NOT USED				
BYTE 2	NOT USED							
BYTE 3	NOT USED							
BYTE 4	NOT USED							
BYTE 5	0	0	0	0	0	0	0	0

Read Sector Buffer (HEX 10)

Description

The Read Sector Buffer command transfers the contents of the controller disk buffer (256 bytes) to the host.

Purpose

This command can be used to:

- test the disk buffer by reading the data buffer after a Write Sector Buffer command and
- retrieve the data that caused an uncorrectable data error. When an uncorrectable data error occurs, the controller reports the data error and does not send the data to the host; the disk buffer contains the bad data.

HEX 10 device command field

The following diagram illustrates the HEX 10 DCF format.

Note:

X = don't care

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	1	0	0	0	0
BYTE 1	X	0	0	NOT USED				
BYTE 2	NOT USED							
BYTE 3	NOT USED							
BYTE 4	NOT USED							
BYTE 5	0	0	0	0	0	0	0	0

Request Controller Type (HEX 11)

Description

The Request Controller Type command returns six bytes of controller information to the host computer.

HEX 11 device command field

The following diagram illustrates the HEX 11 DCF format.

Note:

X = don't care

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	1	0	0	0	1
BYTE 1	X	0	0	NOT USED				
BYTE 2	NOT USED							
BYTE 3	NOT USED							
BYTE 4	NOT USED							
BYTE 5	NOT USED							

HEX 11 response data format

The information is returned to the host as data. The format of the data is shown below:

Note:

SP = spare configuration jumper
TDP = tape drive present configuration jumper

BIT	7	6	5	4	3	2	1	0
BYTE 0	CONTROLLER TYPE							
BYTE 1	VERSION NUMBER							
BYTE 2	CONFIGURATION BYTE							
BYTE 3	0	0	0	0	0	0	SP	TDP
BYTE 4	0	0	0	0	0	0	0	0
BYTE 5	0	0	0	0	0	0	0	0

Response byte table

BYTE	BIT	DESCRIPTION
0	0-7	Controller type is the supplier code and controller model. Vendor Model No. 1430DP = 08.
1	0-7	Version number is the firmware revision number.
2	0-7	Configuration byte contains the configuration jumper value that identifies subsystem devices. See *Controller Configuration Jumpers*.
3	0	Tape drive present is interpreted as: ● when set to 1, a tape drive is present and ● when reset to 0, no tape drive is present. See *Controller Configuration Jumpers*.
3	1	Indicates the spare jumper status in the configuration jumpers. See *Controller Configuration Jumpers*.

Read Identifier Physical (HEX 12)

Description

The Read Identifier Physical command causes the controller to transfer the ID field of the physical sector to the host in a six-byte block. The sector is identified by the physical sector address, bytes 1 through 3.

Note:

This command will not access an alternate track.

HEX 12 device command field

The following diagram illustrates the HEX 12 DCF format.

Note:

```
dd = drive LUN
 r = retries
```

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	1	0	0	1	0
BYTE 1	0	d	d	PHYSICAL ADDR 2 (MSB)				
BYTE 2	PHYSICAL ADDR 1							
BYTE 3	PHYSICAL ADDR 0 (LSB)							
BYTE 4	NOT USED							
BYTE 5	r	0	0	0	0	0	0	0

HEX 12 response data format

The data is returned to the host in the following format.

BIT	7	6	5	4	3	2	1	0
BYTE 0	COMPARE BYTE							
BYTE 1	CYLINDER HIGH							
BYTE 2	CYLINDER LOW							
BYTE 3	HEAD							
BYTE 4	LOGICAL SECTOR							
BYTE 5	FLAGS							

Note:

BYTE 0 always contains HEX C2.

Flag byte table

The flag byte, located in the ID field of each disk sector, provides additional information about the sector or the track on which the sector is located. The flag byte bits are described in the following table.

BIT...	IS...
0	set to 1 if the sector is on a defective track and an alternate track has been assigned.
1	set to 1 if the sector is on a defective track and an alternate track has not been assigned.
2	set to 1 if the sector is on an alternate track.
3	not used. It is always reset to 0.
4	set to 1 if the sector is the last physical sector on the track.
5	not used. It is always reset to 0.
6	always reset to 0.
7	not used. It is always set to 1.

Read Identifier Logical (HEX 13)

Description

The Read Identifier Logical command causes the controller to transfer the ID field of the addressed logical sector to the host in a 6-byte block. The processed sector is identified in the logical sector address, bytes 1 through 3.

Note:

This command does access an alternate track.

HEX 13 device command field

The following diagram illustrates the HEX 13 DCF format.

Note:

```
dd = drive LUN
 r = retries
```

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	1	0	0	1	1
BYTE 1	0	d	d	LOGICAL ADDR 2 (MSB)				
BYTE 2	LOGICAL ADDR 1							
BYTE 3	LOGICAL ADDR 0 (LSB)							
BYTE 4	NOT USED							
BYTE 5	r	0	0	0	0	0	0	0

Response data format

The data is returned to the host in the following format:

BIT	7	6	5	4	3	2	1	0
BYTE 0	COMPARE BYTE							
BYTE 1	CYLINDER HIGH							
BYTE 2	CYLINDER LOW							
BYTE 3	HEAD							
BYTE 4	LOGICAL SECTOR							
BYTE 5	FLAGS							

Note:

Byte 0 always contains HEX C2.

See *Flag byte table* in *Read Identifier Physical (HEX 12)* for a description of the flag byte bits.

Check Track ECC (HEX 14)

Description

The Check Track ECC command is similar to the Read Sectors No Data Transfer command except that the entire track is verified without regard to sector interleave. The ID and data fields are checked and any errors detected are reported.

Note:

If no errors are detected, this command must be complete in no more than two disk revolutions.

HEX 14 device command field

The following diagram illustrates the HEX 14 DCF format.

Note:

dd = drive LUN

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	1	0	1	0	0
BYTE 1	0	d	d	LOGICAL ADDR 2 (MSB)				
BYTE 2	LOGICAL ADDR 1							
BYTE 3	LOGICAL ADDR 0 (LSB)							
BYTE 4	NOT USED							
BYTE 5	NOT USED							

Comment

If an error is encountered during the execution of this command:

- execution is terminated,
- an uncorrectable error is reported, and
- the address is not updated as in commands that process sectors in a logical sequence. The logical address returned from a Request Sense Command following an error is the address passed by the command.

Chapter 6. DISK DRIVE ERROR CODES

Overview

Introduction

This chapter includes sections containing:

- a disk error code summary for quick reference,
- detailed descriptions of disk errors,
- firmware error recovery routines, and
- information on alternate track assignment and handling.

Disk Error Code Summary

Introduction

The following is a summary of the disk error codes.

<u>Note:</u>

The address valid bit (bit 7) may or may not be set and is not included here.

HEX CODE	MEANING
00	No error detected (command completed)
01	No index detected from disk drive
02	No seek complete from disk drive
03	Write fault from disk drive
04	Drive not ready after it was selected
05	Not used
06	Track 00 not found
07	Not used
08	Disk drive still seeking
09	Not used
0A	Drive not initialized
0B	A write was attempted to a write-protected drive

Error code summary continued

HEX CODE	MEANING
0C-0F	Not used
10	ID field read error
11	Uncorrectable data error
12	Sector mark not found
13	Not used
14	Target sector not found
15	Seek error
16	Not used
17	Not used
18	Correctable data error
19	Bad track flag detected
1A	Format error
1B	Data error did not reoccur on retry
1C	Not used
1D	Alternate track already assigned

Error code summary continued

HEX CODE	MEANING
1E	Assigned track pointed by the bad track is not formatted as an alternate track
1F	Alternate Track and Defective Track addresses are the same
20	Invalid command
21	Illegal disk address
22	Illegal parameter
23-2F	Not used
30	RAM diagnostic failure
31	Program memory checksum error
32	ECC diagnostic failure
33	Exceeded 20 errors on a single command
34-3F	Not used

Disk Error Codes

Introduction

This section describes disk error codes returned in the Disk Request Sense Status command or in the message byte on completion of a command. The cause of the error is given, followed by the most probable source of the error. The error code numbers are given in HEX notation.

HEX 00 No Error Occurred

ERROR HEX 00 IS RETURNED WHEN...
no error has occurred during command execution.

HEX 01 No Index Signal from the Drive

ERROR HEX 01 OCCURS WHEN...	THE PROBABLE SOURCE IS...
no drive index signal is detected within two disk revolutions during a data transfer or format command after a normal drive select.	• a faulty drive, • the disk control cable, or • the controller.

HEX 02 No Seek Complete Signal from the Drive

ERROR HEX 02 OCCURS WHEN...	THE PROBABLE SOURCE IS...
the controller does not receive a seek complete signal within two seconds following the start of the next drive command or on implied seeks generated by the controller.	• a faulty drive, • the disk control cable, or • the controller.

HEX 03 Write Fault Received from the Drive

ERROR HEX 03 OCCURS WHEN...	THE PROBABLE SOURCE IS...
the controller detects an active drive write fault signal at data transfer completion or after a successful drive seek.	• drive power supply voltage out of range, • a faulty drive, • the disk control cable, • the unit cable, or • the controller.

HEX 04 Disk Drive Not Ready

ERROR HEX 04 OCCURS WHEN...	THE PROBABLE SOURCE IS...
the controller fails to receive a select signal from the drive or the drive indicates not ready after selection.	• drive power supply voltage out of range, drive not up to operating speed after power on, • a faulty drive, • the disk control cable, or • the controller.

HEX 06 Track 00 Not Found

ERROR HEX 06 OCCURS WHEN...	THE PROBABLE SOURCE IS...
the track 00 signal is not received from the drive after stepping 200 steps more than the number of cylinders during a recalibrate.	• incorrect drive size initialization (too few cylinders), • a faulty drive, • the disk control cable, or • the controller.

HEX 08 Disk Drive is Seeking

ERROR HEX 08 OCCURS WHEN...
a Test Drive Ready command is issued when a previously issued Seek command is still executing.

HEX 0A Drive Not Initialized

ERROR HEX 0A OCCURS WHEN...
the drive specified has not been initialized using the Initialize Drive Characteristics command or the tape drive is accessed when drive 0 has not been initialized.

HEX 0B Write Protected Drive

ERROR HEX 0B OCCURS WHEN...
a write is attempted to a write-protected drive.

HEX 10 ID Field Read Error

ERROR HEX 10 OCCURS WHEN...	THE PROBABLE SOURCE IS...
address marks are detected, the target sector is not found, and an ECC error occurred on one or more ID fields during a Data Transfer or Format command.	• media defect on drive, • a faulty drive if errors are numerous or continuous, or • a faulty controller if errors are numerous and continuous.

Note:

Media defects may be overcome by deleting the defective sectors from system use or assigning an alternate track.

HEX 11 Uncorrectable Data Error in Data Field

ERROR HEX 11 OCCURS WHEN...	THE PROBABLE SOURCE IS...
the controller detects a data error that cannot be corrected using ECC. The error span is greater than the data burst length set in the Initialize Drive Characteristics command.	• media defect on the drive, • a faulty drive if errors are numerous or continuous, or • a faulty controller if errors are numerous and continuous.

Note:

Media defects may be overcome by deleting the defective sectors from system use or assigning an alternate track.

HEX 12 Address Mark Missing

ERROR HEX 12 OCCURS WHEN...	THE PROBABLE SOURCE IS...
the controller cannot detect an address mark from the drive within its timing window. The address mark tells the controller where a new sector starts.	• media defect on the drive, • a faulty drive (if errors are numerous), • disk data cable, or • the controller.

HEX 14 Target Sector Not Found

ERROR HEX 14 OCCURS WHEN...	THE PROBABLE SOURCE IS...
the target sector was not found within two revolutions of the disk.	• media defect on the drive, • faulty drive, or • faulty controller.

HEX 15 Seek Error

ERROR HEX 15 OCCURS WHEN...	THE PROBABLE SOURCE IS...
the target disk address did not match the ID address read from the disk after a seek. The cylinder or head bytes did not match.	• incorrect seek option in configuration, • faulty drive (seek to incorrect cylinder), • faulty control cable, or • faulty controller.

HEX 18 Correctable Data Error

ERROR HEX 18 OCCURS WHEN...	THE PROBABLE SOURCE IS...
the controller detects a media error during a read that was corrected by ECC.	<u>Note:</u> This code informs the host that correction has taken place.

HEX 19 Track is Flagged Bad

ERROR HEX 19 OCCURS WHEN...	THE PROBABLE SOURCE IS...
a data transfer command encountered a track flagged defective using the Format Bad Track command.	Note: Host software must ensure that deleted tracks are not accessed.

HEX 1A Format Error

ERROR HEX 1A OCCURS WHEN...	THE PROBABLE SOURCE IS...
the controller detects an unformatted track, incorrect inter-leave on disk, or an ID ECC error on at least one sector.	• drive not formatted, • incorrect interleave specified in command, or • media defect.

HEX 1B Data Error Did Not Occur on Retry

ERROR HEX 1B OCCURS WHEN...	THE PROBABLE SOURCE IS...
a read of a sector after a retry caused by a data error did not cause an error.	the original error was transient.

HEX 1D Alternate Track Already Assigned

ERROR HEX 1D OCCURS WHEN...	THE PROBABLE SOURCE IS...
a designated alternate track has been assigned to another defective track or formatted as a defective track.	host software has attempted to assign an unavailable alternate track.

Note:

If an alternate track is no longer needed, the host software must reformat the track using the Format Track Command before attempting to assign the track again.

HEX 1E Assigned Alternate Track Not Found

ERROR HEX 1E OCCURS WHEN...	THE PROBABLE SOURCE IS...
a defective track is assigned an alternate track that does not have the alternate track bit in the ID field set to 1.	reformatting an alternate track using the Format Track command without reprocessing the defective track.

HEX 1F Alternate and Defective Track Addresses Point to the Same Track

ERROR HEX 1F OCCURS WHEN...	THE PROBABLE SOURCE IS...
the alternate track and defective track addresses point to the same track.	Note: This alternate track scheme does not allow alternate and defective track addresses that point to the same track.

HEX 20 Invalid Command

ERROR HEX 20 OCCURS WHEN...
the controller receives an invalid command from the host.

HEX 21 Illegal Disk Address

ERROR HEX 21 OCCURS WHEN...
the controller receives an address beyond the disk's maximum range.

HEX 22 Illegal Parameter

ERROR HEX 22 OCCURS WHEN...
the controller receives an invalid command parameter or parameter combination.

HEX 30 RAM Error

ERROR HEX 30 OCCURS WHEN...	THE PROBABLE SOURCE IS...
the controller detects a data error during a RAM sector-buffer diagnostic.	faulty controller.

HEX 31 Program Memory Checksum Error

ERROR HEX 31 OCCURS WHEN...	THE PROBABLE SOURCE IS...
the controller detects a program-memory checksum error during internal diagnostics.	faulty controller.

HEX 32 ECC Polynomial Error

ERROR HEX 32 OCCURS WHEN...	THE PROBABLE SOURCE IS...
the hardware ECC generator failed its test during the controller's internal diagnostics.	faulty controller.

HEX 33 Maximum Single Command Error Count Exceeded

ERROR HEX 33 OCCURS WHEN...	THE RESULT IS...
error total (20, all types) is exceeded on a single command.	the command is aborted and an error message returned.

Firmware Error Recovery Routines

Introduction

This section lists the error recovery routines the controller performs when error recovery is enabled in the control field (byte 5) of each command. The 9330 subsystem error recovery routines are of three types. They are:

- drive or host related,
- data error related, and
- position error related.

Drive or host error recovery

The following table lists drive- or host-related error recovery codes. Byte 5, bit 7 must be set to 1 for error recovery.

HEX CODE	ROUTINE
00	No error returned
01	No index detected
02	No seek complete
03	Write fault
04	Drive not ready
06	Track 00 not found

Drive or host error recovery continued

HEX CODE	ROUTINE
08	Disk drive still seeking
19	Bad track flag detected
1A	Format error
1D	Alternate track already assigned
1E	Assigned track pointed by the faulty track is not formatted as an alternate
1F	Alternate track and defective track address are the same
20	Invalid command
21	Illegal disk address
30	RAM diagnostic failure
31	Program memory checksum error
32	ECC diagnostic failure

Data error recovery

The following error codes are data-related and the controller will perform:

- one reread and
- up to three recalibration, seek, and reread sequences.

HEX CODE	ROUTINE
10	ID field read error
11	Uncorrectable data error
12	Sector mask not found
14	Target sector not found
18	Correctable data error

Position error recovery

For position-related errors, the controller performs up to four recalibrations and reseeks.

HEX CODE	ROUTINE
15	Seek errors

Comment

A maximum of 20 retries is attempted for any combination of errors on a single command.

Alternate Track Assignment and Handling

Introduction

When alternate tracks are used, the entire disk is not available to the system. Generally, the disk space is fixed in the system software and leaves spare tracks at the inner cylinder of the disk that are assigned as alternates when needed.

Note:

An alternate track can be assigned to any track as long as the host maintains its location.

Using alternate tracking

To use alternate tracking, the controller must be initialized to include the alternate tracks cylinder and head ranges.

The procedure for assigning an alternate track is described in the following table.

STEP	ACTION
1	Format the entire disk, including spare tracks.
2	Verify the disk.
3	For each media defect, assign an alternate track.
4	Alternate tracks should also be assigned for the drive manufacturer's defect list.

Alternate track access

During system operation, alternate tracks are invisible to the host. The controller automatically seeks to the assigned alternate track when an access is made to a flagged defective track.

Alternate tracks can be accessed by:

- access to a defective track that has been flagged and assigned to the alternate or
- direct access to the alternate track.

Note:

Consecutive accesses to a flagged track will not result in reseeking to the alternate track. The controller maintains position on the alternate track.

Assigning Alternate Tracks

Introduction

Alternate track assignment and bad track lockout is done by the host computer. Bad areas on the disk are labeled defective by track using the Format Bad Track command (HEX 07). The host must maintain a table of all bad tracks found for which alternates must be assigned.

The following paragraphs describe a sample procedure for assignment and handling of alternate tracks.

Step 1

The entire disk drive is formatted using the Format Drive command (HEX 04) starting at logical track zero. The results are given in the following table.

IF...	THEN...
an error occurs during formatting,	a Disk Request Sense Status command should be issued.
a format error is indicated,	• bytes 1 through 3 in the returned status give the bad track address, • a Format Bad Track command (HEX 07) should be issued to that track,
AND...	
a new Format Drive command is issued to format the rest of the track beginning one track beyond the bad track.	
IF...	**THEN...**
any errors occur during subsequent formatting,	repeat these steps until the entire disk is formatted.

Step 2

A Recalibrate command (HEX 01) is issued to position the heads over track 00.

Step 3

All sectors on the disk are read to detect any uncorrectable ECC errors in the data as follows:

- The disk Format command places a HEX 6C pattern in the data fields of all sectors and the host program can verify this data pattern after the data is read into memory.
- Multiple sector reads can be issued to increase the verification process speed when the host has a large block of memory available.

Note:

Verifying the data byte for byte is not necessary since the ECC circuitry flags all uncorrectable errors.

Step 4

IF...	THEN...
an uncorrectable error is found,	the Format Bad Track command (HEX 07) is issued to the failed track.

Note:

When a track is flagged as defective, its data fields cannot be accessed.

Step 5

Continue verifying the disk surface.

IF...	THEN...
subsequent errors are detected,	repeat steps 1 through 5 until the entire disk is verified.

Step 6

Using the Format Alternate Track command (HEX 0E), assign alternates for all tracks logged by the host.

Caution

The disk cannot detect when an alternate track is being read; therefore, the host should not allow a user program to issue a read or write command to an alternate track.

An alternate track may be assigned to any track as long as the host maintains its location.

Chapter 7. TAPE COMMAND STRUCTURE

Overview

Introduction

This chapter provides detailed programming information for the DATAPOINT 9330 subsystem tape drive. Sections include:

- tape commands and their HEX equivalent codes,
- the Sense Status Block, and
- tape drive error codes.

Abbreviations

The following abbreviations are used in this chapter:

- LUN--Logical Unit Number, and
- DCF--Device Command Field.

Tape command summary

The following table lists the tape drive commands in ascending numerical order by their HEX codes.

HEX CODE	COMMAND
00	Test Unit Ready
01	Rewind
03	Request Sense Status
08	Read
0A	Write
0C	Initialize Drive Characteristics
10	Write File Mark
11	Space
15	Mode Select
18	Copy
19	Erase
1A	Mode Sense
1D	Send Diagnostics

Comment

If no disk drive 0 is present, drive 0 must be initialized to gain access to the tape drive. This sequence may be used to determine if a controller power-on reset has occurred since the last drive access. See *Initialize Drive Characteristics (HEX 0C)* in Chapter 5.

Test Unit Ready (HEX 00)

Description

The Test Unit Ready command examines the status bits at the tape drive interface. An error code is returned if the interrupt or device busy bits are set to 1 or if the ready bit is reset to 0. Use this command to:

- detect completion of the drive power-up diagnostics and retension cycle. The drive should be ready within four minutes after power up;
- detect completion of any command that returns completion immediately.

HEX 00 device command field

The following diagram illustrates the HEX 00 DCF format.

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	0	0	0	0	1
BYTE 1	0	1	0	NOT USED				
BYTE 2	NOT USED							
BYTE 3	NOT USED							
BYTE 4	NOT USED							
BYTE 5	NOT USED							

Rewind (HEX 01)

Description

The Rewind command positions the tape drive head to track 0 and rewinds the tape to the load point. Completion status is returned immediately after receipt of the Rewind command.

HEX 01 device command field

The following diagram illustrates the HEX 01 DCF format.

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	0	0	0	0	1
BYTE 1	0	1	0	NOT USED				
BYTE 2	NOT USED							
BYTE 3	NOT USED							
BYTE 4	NOT USED							
BYTE 5	NOT USED							

Request Sense Status (HEX 03)

Description

The host must send a Request Sense Status command immediately after it detects an error during a tape command. It causes the controller to:

- return the 22-byte sense status block containing tape drive and controller status or
- obtain the results of the tape drive power-up diagnostics immediately after power-up.

HEX 03 device command field

The following diagram illustrates the HEX 03 DCF format.

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	0	0	0	1	1
BYTE 1	0	1	0	NOT USED				
BYTE 2	NOT USED							
BYTE 3	NOT USED							
BYTE 4	NOT USED							
BYTE 5	NOT USED							

Sense status block

The Sense status block format is shown below.

Note:

```
  e = error code
VLD = valid bit
FIL = filemark bit
EOT = end of tape bit
ILI = incorrect length indicator
```

BIT	7	6	5	4	3	2	1	0
BYTE 0	VLD	e	e	e	e	e	e	e
BYTE 1	FIL	EOT	ILI	NOT USED				
BYTE 2	INFORMATION BYTE 1 (MSB)							
BYTE 3	INFORMATION BYTE 2							
BYTE 4	INFORMATION BYTE 3							
BYTE 5	INFORMATION BYTE 4 (LSB)							
BYTE 6								
.								
.								
.								
BYTE 21								

Bit designation summary

The Sense status block bit designations are listed in the following table.

BYTE...	BIT...	IS THE...
0	7	valid bit.
0	0-6	error code.
1	7	filemark bit.
1	6	end of tape bit.
1	5	incorrect length indicator bit.
2-5	0-7	information block.
6-21	0-7	sense status block.

Comment

For commands that involve both tape and disk drives, status is maintained for each device. If an error occurs during such a command, status should be requested from both drives.

Read (HEX 08)

Description

The Read command causes data to be transferred from the tape to the host. When a data error is encountered, the tape is stopped and backed up to the beginning of the current block. No data in a block containing an error is transferred to the host. Automatic data recovery is enabled by setting Retry (byte 5, bit 7) to 1.

HEX 08 device command field

The following diagram illustrates the HEX 08 DCF format.

Note:

```
FIX = fixed bit (specifies data transfer mode)
  r = retry
```

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	0	1	0	0	0
BYTE 1	0	1	0	NOT USED				FIX
BYTE 2		NUMBER OF BYTES/BLOCKS (MSB)						
BYTE 3		NUMBER OF BYTES/BLOCKS						
BYTE 4		NUMBER OF BYTES/BLOCKS (LSB)						
BYTE 5	r	NOT USED						

Fixed bit
Byte 1, bit 0

The fixed bit (FIX) specifies the data transfer mode as shown in the following table.

IF THE...	THEN...	AND THE...
FIX bit is reset to 0,	one data block is transferred to the controller buffer	number of bytes transferred is compared with the number of bytes specified by bytes 2 through 4.
block size is different from specified number,	an error message is sent to the host on command completion	information bytes in the sense status block are set to the difference between the requested length and the actual block size. ILI is set to 1.

IF THE...	THEN...
FIX bit is set to 1,	the controller transfers the number of data blocks specified in bytes 2 through 4 to the host.
block length differs from the block size specified in the previous Mode Select command,	information bytes in the sense status block contain the difference between the specified block and the actual block size. ILI is set to 1.
specified block size is larger than the block read,	00 bytes are appended to the tape data block to complete the block for the host.
specified block size is smaller than the block read,	only the specified number of bytes from the beginning of the tape block are sent to the host.

Comment

The number of data bytes transferred to the host always equals the requested number when the FIXED bit is reset to 0 or to the number specified by the MODE SELECT command if the FIXED bit is set to 1.

Number of bytes/blocks
Bytes 2 through 4

Bytes 2 through 4 specify the number of bytes/blocks to be transferred to the host. The number of bytes must be from 256 to 8,192 bytes.

If the specified size is outside this range, the command is not executed, and an error is reported to the host.

Retry bit
Byte 5, bit 7

The Retry bit enables automatic attempts to recover a data block when an error is detected. A data error stops the tape and returns it to the current block beginning. The following table describes the retry bit functions.

IF THE...	THEN THE...	AND THE...
Retry bit is reset to 0,	controller attempts to read the block again,	process is repeated until successful or 10 retries of the same block have been made.
error persists after 10 retries,	command is aborted and an error message is returned to the host,	tape is stopped at the end of the defective block. VLD is set to 1.
Retry bit is set to 1,	current block is aborted and an error message is returned to the host,	tape is stopped at the beginning of the defective block. VLD is set to 1.

Filemark
Byte 1, bit 7
Sense status block

IF THE...	THEN THE...	AND THE...
filemark is read prior to command completion,	tape is stopped after the filemark and before the next data block; no more data is read; the command is aborted when all data previously read has been sent to the host,	error message is sent to the host. VLD and FIL in the Sense Status block are set to 1.

Incorrect length indicator
Byte 1, bit 5
Sense status block

IF THE...	THEN THE...
ILI bit is reset to 0 and the VLD bit is set to 1,	information bytes contain the difference between the requested number of blocks and the actual number of blocks transferred.
FIX bit is reset to 0,	difference equals 1 since a request for a single data block is implied by the command.

Write (HEX 0A)

Description

The Write command causes data to be transferred from the host to the tape in blocks of 256 to 8,192 bytes. The tape subsystem performs a read-after-write to verify that all data is correctly written to the tape.

HEX 0A device command field

The following diagram illustrates the HEX 0A DCF format.

Note:

```
FIX = fixed bit (specifies data transfer mode)
  r = retry
```

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	0	1	0	1	0
BYTE 1	0	1	0	NOT USED				FIX
BYTE 2		NUMBER OF BYTES/BLOCKS (MSB)						
BYTE 3		NUMBER OF BYTES/BLOCKS						
BYTE 4		NUMBER OF BYTES/BLOCKS (LSB)						
BYTE 5	r	NOT USED						

The following pages describe the WRITE command in detail.

Fixed bit
Byte 1, bit 0

IF THE...	THEN...	AND THE...
FIXED bit is reset to 0,	one data block is transferred from host to the tape beginning at the current position,	number of bytes transferred is specified in bytes 2 through 4 and can differ from the block size in the previous mode select command.
block size is outside the 256 to 8,192 byte range,	the command is not executed, an error message is sent to the host,	tape sense status bytes of the sense status block are not updated.
FIXED bit is set to 1,	the controller sends the number of blocks specified in bytes 2 through 4 to the tape, beginning at the current tape position. The number of bytes in each block is specified by MODE SELECT.	

End-of-Tape (EOT) bit
Byte 1, bit 6
Sense status block

<table>
<tr><th>IF THE...</th><th>THEN...</th><th>AND THE...</th></tr>
<tr><td>early warning EOT is detected during a Write command,</td><td>the controller attempts to transfer any buffered data,</td><td>command terminates with an error to the host. The EOT bit is set to 1.</td></tr>
<tr><td rowspan="2">End-of-Tape (EOT) is encountered,</td><td>the VLD bit in the status block is set to 1,</td><td>information bytes are set to the difference between the number of blocks requested and the number written.</td></tr>
<tr><td colspan="2">Note:

The information bytes are also set to the difference following any error that causes a tape subsystem interrupt.</td></tr>
</table>

Retry bit
Byte 5, bit 7

The tape cartridge subsystem performs a read-after-write to verify that all data is written correctly to the tape. If a data error is detected, the tape is stopped, repositioned to the beginning of the current block, and a tape length equal to a 4096-byte block is erased. The following table describes the subsequent process.

IF THE...	THEN...
RETRY bit is set to 1,	the controller does not retry the write operation.
RETRY bit is reset to 0,	the controller retries writing the current block beginning at the tape position after the erased block.
data error occurs after seven attempts to write the same block,	the command is aborted and an error message is sent to the host. VLD is set to 1 and the information bytes are set to the difference between the requested number of blocks and the number of blocks written to the tape.

Write Filemark (HEX 10)

Description

The Write Filemark command causes a filemark pattern to be written to the tape after the appropriate gap has been generated.

HEX 10 device command field

The following diagram illustrates the HEX 10 DCF format.

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	1	0	0	0	0
BYTE 1	0	1	0	NOT USED				
BYTE 2	NOT USED							
BYTE 3	NOT USED							
BYTE 4	NOT USED							
BYTE 5	NOT USED							

Space (HEX 11)

Description

The Space command provides forward and reverse positioning. The functions are specified in the code and count fields of bytes 1 through 4. Forward positioning moves the tape toward the End-of-Tape (EOT), and reverse positioning moves the tape toward the Beginning-of-Tape (BOT).

HEX 11 device command field

The following diagram illustrates the HEX 11 DCF format.

<u>Note:</u>

c = code

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	1	0	0	0	1
BYTE 1	0	1	0	NOT USED			c	c
BYTE 2	COUNT (MSB)							
BYTE 3	COUNT							
BYTE 4	COUNT (LSB)							
BYTE 5	NOT USED							

The following pages describe the SPACE command in detail.

Code
Byte 1, bits 0-1

The code field specifies positioning type as shown in the following table.

CODE	POSITIONING TYPE
00	Variable block count
01	Variable filemark count
10	Fixed length space 1 block
11	Fixed length space 1 filemark

Count
Bytes 2-4

For variable length Space commands (code 00 or 01), the count field specifies the number of blocks or filemarks to be spaced over. The following sections describe the count field.

Variable length space commands

IF...	THEN THE...	ENDING ON THE...
count is positive,	tape moves forward over the specified number of blocks or filemarks,	EOT side of the last block or filemark.
count is negative using 2's complement notation,	tape moves backward over the specified number of blocks or filemarks,	BOT side of the last block or filemark.
count is 0,	tape does not move.	

Note:

Command completion status is returned to the host after tape repositioning.

Fixed length space commands

For fixed length Space commands (code 10 or 11), the count field specifies the direction of tape movement. The following table defines and describes the count field.

IF...	THEN THE...	ENDING ON THE...
count is positive,	tape moves forward one block to filemark,	EOT side of the last block or filemark.
count is negative using 2's complement notation,	tape moves backward one block or filemark,	BOT side of the last block or filemark.
count is 0,	tape does not move.	

Note:

Command completion status is returned to the host immediately after receipt of the Space command.

Filemark
Byte 1, bit 7
Sense status block

The following tables show the tape drive response if a filemark is encountered during command execution.

<table>
<tr><th colspan="3">IF A FILEMARK IS ENCOUNTERED WHILE SPACING OVER BLOCKS, THE RESPONSE IS...</th></tr>
<tr><th>WHEN THE TAPE IS MOVING...</th><th>THEN...</th><th>ON THE...</th></tr>
<tr><td>forward,</td><td>tape motion stops</td><td>EOT side of the filemark.</td></tr>
<tr><td>backward,</td><td>tape motion stops</td><td>BOT side of the filemark.</td></tr>
<tr><th>WHEN THIS OCCURS,...</th><th colspan="2">AND THE...</th></tr>
<tr><td>an error message is sent to the host at the end of the command,</td><td colspan="2">information bytes are set to the difference between the requested and actual number of blocks spaced over.
VLD and FIL are set to 1.

Note:

The number of blocks in the information bytes does not include the block containing the filemark.</td></tr>
</table>

Mode Select (HEX 15)

Description

The Mode Select command is used to specify the size of data blocks to be used on the tape. Bytes 3 and 4 contain the data block size that must be from 256 to 8,192 bytes.

Block size defaults to 8,192 bytes on power up.

HEX 15 device command field

The following diagram illustrates the HEX 15 DCF format.

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	1	0	1	0	1
BYTE 1	0	1	0	NOT USED				
BYTE 2	NOT USED							
BYTE 3	NUMBER OF BYTES PER BLOCK (MSB)							
BYTE 4	NUMBER OF BYTES PER BLOCK (LSB)							
BYTE 5	NOT USED							

Copy (HEX 18)

Description

The Copy command enables the transfer of data between a disk drive and the tape drive. The data block size is 8,192 bytes regardless of the size specified by the Mode Select command. The command parameters are sent to the controller by the host in a 16-byte parameter list following the 6-byte command transfer. Only one segment is specified for each command.

HEX 18 device command field

The following diagram illustrates the HEX 18 DCF format.

Note:

r = retries
a = ECC correction

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	1	1	0	0	0
BYTE 1	0	1	0	NOT USED				
BYTE 2	NOT USED							
BYTE 3	NOT USED							
BYTE 4	NOT USED							
BYTE 5	r	a	0	0	0	0	0	0

Copy command parameter list

The parameter list consists of 16 bytes transferred to the controller as data. A 4-byte preamble is reserved for future use; the remaining 12 bytes define the parameters of the single data segment to be transferred.

Parameter list format

Note:

s = source LUN
d = destination LUN

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	0	NOT USED			
BYTE 1	NOT USED							
BYTE 2	NOT USED							
BYTE 3	NOT USED							
BYTE 4	NOT USED					s	s	s
BYTE 5	NOT USED					d	d	d
BYTE 6	NOT USED							
BYTE 7	NOT USED							
BYTE 8	NUMBER OF BLOCKS TO BE TRANSFERRED (MSB)							
BYTE 9	NUMBER OF BLOCKS TO BE TRANSFERRED							
BYTE 10	NUMBER OF BLOCKS TO BE TRANSFERRED							
BYTE 11	NUMBER OF BLOCKS TO BE TRANSFERRED (LSB)							
BYTE 12	DISK BLOCK ADDRESS (MSB)							
BYTE 13	DISK BLOCK ADDRESS							
BYTE 14	DISK BLOCK ADDRESS							
BYTE 15	DISK BLOCK ADDRESS (LSB)							

Parameter definitions

The following table gives additional information on the copy command parameter list.

BYTE...	BITS...	SPECIFY THE...
4	0-2	device number (LUN) of the data source.
5	0-2	device number (LUN) of the data destination.
8-11	0-7	number of blocks to be transferred.
12-15	0-7	disk block address of the first block in the segment, in terms of block number. Note: The disk block address must be specified on a block boundary and is calculated as the integer value: $\frac{\text{LOGICAL SECTOR NUMBER (LSN)}}{\text{\# OF SECTORS PER BLOCK}}$

Comment

If an error is returned to the host following the Copy command, separate Request Sense Status commands should be issued for disk and tape devices to get more information about the nature of the error.

Retries and updating the sense status block are the same as described for the disk Read/Write commands and the tape Read/Write commands with the fixed bit set to one.

Caution

The LUN returned in the completion status byte identifies the tape drive selected--not the disk.

Erase (HEX 19)

Description

The Erase command causes part or all of the remaining tape to be erased, beginning from the current tape position. The distance to be erased is controlled by the long bit (byte 1, bit 0).

HEX 19 device command field

The following diagram illustrates the HEX 19 DCF format.

Note:

n = long

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	1	1	0	0	1
BYTE 1	0	1	0	NOT USED				n
BYTE 2	NOT USED							
BYTE 3	NOT USED							
BYTE 4	NOT USED							
BYTE 5	NOT USED							

Long bit
Byte 1, bit 0

The long bit determines the length of tape to be erased with the Erase command. It is set as follows:

IF THE LONG BIT IS...	THEN...
set to 1,	all the tape from the current position to EOT is erased.
reset to 0,	a 4,096-byte block of tape is erased.

Mode Sense (HEX 1A)

Description

The Mode Sense command provides a means for the host to read from the controller the block size as defined by a previous Mode Select command.

HEX 1A device command field

The following diagram illustrates the HEX 1A DCF format.

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	1	1	0	1	0
BYTE 1	0	1	0	NOT USED				
BYTE 2	NOT USED							
BYTE 3	NOT USED							
BYTE 4	NOT USED							
BYTE 5	NOT USED							

The controller responds to this command by sending two data bytes to the host which equal the number of bytes defining one block. The most significant byte (MSB) is sent first.

Send Diagnostic (HEX 1D)

Description

The Send Diagnostic command sends either the Device Health Check or the Cartridge Health Check command to the tape cartridge subsystem.

The Device Health Check determines that the device meets minimum operational conditions.

The Cartridge Health Check writes all 1's on tracks 0 and 5 while monitoring read-after-write data.

Completion status is returned to the host immediately after receipt of the Send Diagnostic command. After receipt of the Send Diagnostic command, errors are counted and the sense status block is updated.

HEX 1D device command field

The following diagram illustrates the HEX 1D DCF format.

Note:

```
 st = device health check
uol = cartridge health check
  X = don't care
```

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	1	1	1	0	1
BYTE 1	0	1	0	NOT USED		st	X	uol
BYTE 2	NOT USED							
BYTE 3	NOT USED							
BYTE 4	NOT USED							
BYTE 5	NOT USED							

Note:

Either st or uol may be set to 1, but not both. If both are 1, Illegal parameter error code 22 is returned.

Chapter 8. TAPE DRIVE ERROR CODES

Overview

Introduction

This chapter includes sections containing:

- a tape drive error code summary,
- descriptions of tape drive error codes, and
- a definition of the Sense status block.

Tape Drive Error Code Summary

The following table is a summary of the tape drive error codes returned in the Request Sense Status command.

HEX CODE	DEFINITION
00	no error occurred
04	drive not ready
08	drive busy
11	data error detected
20	invalid tape command
3A	tape drive interrupt
3B	health check fault
3C	filemark detected
3D	block size error
3E	block count too large
3F	disk data transfer error

Error Codes

The following table lists and defines the error codes that can be returned after a tape drive command.

HEX CODE	DEFINITION
00	No error occurred. This code is always returned if no error occurred during the previous tape command.
04	Tape drive is not ready.
08	Tape drive is busy. The tape drive sense status bytes are not valid.
11	A data error was detected during a read or read-after-write operation. If retries are enabled, then the error was uncorrectable.
20	Invalid tape command.
3A	Tape drive interrupt. An error occurred in the tape drive that caused the drive interface interrupt line to go active. Consult the sense status bytes for the cause of the interrupt.
3B	Health check fault. A drive fault was detected during execution of a health check command.

Error codes continued

HEX CODE	DEFINITION
3C	Filemark detected.
3D	Block size error. The block size specified by the command lies outside of the allowable range, or the size of the block read from the tape does not equal the size specified by the last command or the Mode Select command.
3E	The block count specified in the command block is too large.
3F	Disk data transfer error. An error occurred during the copy command in the transfer of data to or from the disk drive. The host should issue a Request Sense Status command to the disk drive for further details.

Sense Status Block

Introduction

The Sense Status Block consists of 22 bytes returned from the controller. Bytes 6 through 21 contain detailed tape information.

Sense Status Block format

The Sense Status Block format is shown in the following diagram. Bit descriptions are given in subsequent paragraphs.

<u>Note:</u>

e = error code

BIT	7	6	5	4	3	2	1	0
BYTE 0	VLD	e	e	e	e	e	e	e
BYTE 1	FIL	EOT	ILI	NOT USED				
BYTE 2	INFORMATION BYTE 1 (MSB)							
BYTE 3	INFORMATION BYTE 2							
BYTE 4	INFORMATION BYTE 3							
BYTE 5	INFORMATION BYTE 4 (LSB)							
BYTE 6 … BYTE 21	TAPE SENSE BYTES							

Bit designation summary

The Sense Status Block bit designations are listed in the following table.

BYTE...	BIT...	IS THE...
0	7	valid bit.
0	0-6	error code.
1	7	filemark bit.
1	6	end of tape bit.
1	5	incorrect length indicator bit.
2-5	0-7	information block.
6-21	0-7	sense status block.

Valid bit
Byte 0, bit 7

The valid bit (VLD) is defined in the following table.

WHEN BIT...	BYTE...	IS...	IT INDICATES...
7	0	set to 1,	a tape error has caused an interrupt and the information bytes contain valid information about the last command.
7	0	reset to 0,	the information bytes are not defined.

IF...	THEN..
the interrupt is caused by the end of tape,	the EOT bit is set to 1.
a filemark is read prior to command completion,	the VLD and FIL bits are set to 1.

Error code
Byte 0, bits 0-6

The error code indicates the type of error detected during a tape command.

Filemark bit
Byte 1, bit 7

When set to 1, the filemark bit (FIL) indicates that a filemark was read.

End-of-Tape (EOT) bit
Byte 1, bit 6

When set to 1, the EOT bit indicates that the end of the tape has been reached.

Incorrect length indicator
Byte 1, bit 5

The Incorrect Length Indicator (ILI), when set to 1, indicates the requested data transfer does not match the actual length of the data record.

Information bytes 1-4
Bytes 2-5

Bytes 2 through 5 of the sense status block are information bytes 1 through 4. They contain the difference between the data transfer length requested in the command and the actual length transferred. The difference is given in either bytes or blocks, as determined by the command.

Sense status bytes
Bytes 6-21

The tape sense bytes contain status information the controller reads directly from the tape drive. The following sections describe this 16-byte block.

Sense byte 6

BIT...	CALLED...	IS SET TO 1 IF...
0	Motion Fault	any of the conditions indicated by the flag bits in sense byte 14 occur.
1	Defective Cartridge	positioning holes (EPT/BOT/LP/EW) are not acceptable, servo track cannot be written, total read or write error count has been exceeded, or an attempt to write on a 450-foot cartridge occurred.
2	Health Check	a failure is detected by health check routines. The MPU will attempt to set other appropriate sense bits.
3	No Lamp Current	no sensor current is present.
4	Head Positioning Fault	a fault is detected while positioning the head.

Sense byte 6 continued

BIT...	CALLED...	IS SET TO 1 IF...
5	Data HER	• bit 3, 4, or 7 in byte 9 is set to 1. • bit 0 in byte 15 is set to 1. • bits 0, 1, 2, and 3, byte 10 are set to 1.
6	Preliminary EOT	an early warning EOT on track 10 occurs.
7	Illegal Command or Illegal Command Sequence	any of the following occur: • A command is received that is not in the command list. • A read is attempted immediately after writing. • A read is attempted where a servo stripe has not been written. • A command is attempted while sense is pending. • An attempt is made to write on a protected cartridge. • An attempt is made to perform a command other than rewind, restore, or sense with tape at EOT on the last track. • An attempt is made to perform a reverse command with tape at LP of track 0 (BOT). • A command other than Device Health Check or Read Sense is issued with health check fault or defective cartridge sense bit set or if INT is set to 1 on interface. • A write command is issued with a 450-foot cartridge installed.

Sense byte 7

BIT...	CALLED...	IS SET TO 1 IF...
0	Cartridge Present	a cartridge is in the operating position.
1	Write Protect	the cartridge is set for read only.
2	Load Point	the tape is positioned at the start of a track.
3	Early Warning	the tape is positioned at the end of a track.
4	Write Current Present	the cartridge is not write protected and write current is turned on.
5	Filemark Detected	a filemark is detected.
6	Retry in Process	a read or write retry is expected.
7	Stripe Found	a good servo stripe is found or written on the tape.

Sense byte 8

<table>
<tr><th>BIT...</th><th>CALLED...</th><th colspan="2">IS USED AS...</th></tr>
<tr><td>0</td><td>Track Counter</td><td colspan="2">Bit 0 (LSB).</td></tr>
<tr><td>1</td><td>Track Counter</td><td colspan="2">Bit 1.</td></tr>
<tr><td>2</td><td>Track Counter</td><td colspan="2">Bit 2.</td></tr>
<tr><td>3</td><td>Track Counter</td><td colspan="2">Bit 3 (MSB).</td></tr>
<tr><td>4</td><td>Model Code</td><td>Bit 0 (LSB).</td><td rowspan="4"><u>Code</u> <u>MODEL</u>
0000 92190
0010 92192
1000 92190
(92190 with optional firmware)</td></tr>
<tr><td>5</td><td>Model Code</td><td>Bit 1.</td></tr>
<tr><td>6</td><td>Model Code</td><td>Bit 2.</td></tr>
<tr><td>7</td><td>Model Code</td><td>Bit 3 (MSB).</td></tr>
</table>

Sense byte 9

BIT...	CALLED...	IS SET TO 1 IF...
0	Crease Detected	a crease is detected during a read. Note: A crease is defined as continuous data dropout of up to 1.8 ms (approximately 0.2 in.). Any data dropout over 1.8 ms is defined as an interblock gap.
1	RFU	Reserved for future use.
2	MK1 Not Found	the beginning of a record is not found.
3	*	noise is detected on read-after-write while erasing.
4	* No Data	no READ DATA is detected before a 16-foot timeout.
5	No RGATE	READ GATE did not go active during a Read command.
6	No Read-After Write Data	unable to detect data being written.
7	* Long Record	a record is more than 48 inches long.
*Sets data HER in sense byte 6 and Interrupt Status.		

Sense byte 10

BIT	CALLED...	IS SET TO 1 IF...
0	*	the maximum read error count is exceeded.
1	*	the maximum write error count is exceeded.
2	* Maximum Read Entry	the maximum read error recovery count is reached without reading the record correctly.
3	* Maximum Write Retry	the maximum write retry count is reached.
4	Retry Counter	bit 0 (LSB).
5	Retry Counter	bit 1.
6	Retry Counter	bit 2.
7	Retry Counter	bit 3 (MSB).
*Sets data HER in sense byte 6 and Interrupt Status.		

Sense byte 11

BITS...	ARE USED AS...	WITH...
0-7	read error counter	bit 7 MSB.

Sense byte 12

BITS...	ARE CALLED...	WITH...
0-7	write error counter	bit 7 MSB.

Sense byte 13

BIT..	CALLED...	IS...
0	last command	bit 0 LSB.
1	last command	bit 1.
2	last command	bit 2.
3	last command	bit 3 (MSB).
4	previous command	bit 0 (LSB).
5	previous command	bit 1.
6	previous command	bit 2.
7	previous command	bit 3 (MSB).

Note:

The Read Sense command (HEX 03) is not included in the last/previous command sense. This sense byte is set to HEX 01 when the device is powered up and the cartridge is removed.

Sense byte 14

BIT...	CALLED...	IS SET TO 1 IF...
0	No Tachs	the capstan motor does not start or tachs are not being generated or detected.
1	Under Speed	the velocity check shows low speed.
2	Over Speed	the velocity check shows high speed.
3	Timeout	the expected result is not returned in a specified time (used by MPU for some timing checks).
4	Tape Positioning Fault	the microprocessor loses tape reference. Tape should be rewound or restored.
5	Stop Fault	a motion fault is detected while stopping the tape. The present reference may not be valid.
6	Tape Break	the tape has broken or comes off the reel.
7	Unexpected EOT/BOT	either BOT or EOT is sensed when not expected.

Sense byte 15

BIT...	CALLED...	SETS...
0	Checksum Error	Data HER in sense byte 6 and Interrupt Status.
1-7	RFU	reserved for future use.

Sense byte 16

BITS...	ARE CALLED...	WITH...
0-7	Error Address (MSB)	bit 7 MSB.

Sense byte 17

BITS...	ARE CALLED...	WITH...
0-7	Error Address (LSB)	bit 7 MSB

Sense byte 18

BIT...	CALLED...	IS SET TO 1 IF...
0	Upper and Lower Tape Sense	both tape position sensors see light.
1	EW Write Error	EW is encountered during a write command.
2	RFU	Reserved for future use.
3	RFU	Reserved for future use.
4	END STATE	end of tape (EOT on last track).
5	EOT INT	end of tape is detected.
6	BOT INT	beginning of tape is detected.
7	LP/EW INT	LP or EW is detected.

Sense byte 19

BITS...	ARE CALLED...	WITH...
0-7	controlware revision level	bit 7 MSB.

Sense byte 20

BIT...	CALLED...	IS USED AS...
0	WFM	write filemark in processing flag.
1	TV	tach verification flag.
2	ERS	erase in process flag.
3	SFM	search filemark in process flag.
4	RDF	read file in process flag.
5	BKSP	backspace in process flag.
6	REPOS	repositioning flag.
7		write stripe in process flag.

Sense byte 21

BIT...	IS USED AS...
0	cartridge initialized flag.
1	write enable flag.
2	seek stripe in process flag.
3	health check in process flag.
4	600-ft cartridge installed.
5	Reserved for future use.
6	cartridge health check complete.
7	Reserved for future use.

Chapter 9. DIAGNOSTIC COMMAND STRUCTURE

Overview

Introduction

The DATAPOINT 9330 subsystem diagnostic command structure is discussed in the:

- Diagnostic execution sequence, and
- Diagnostic command formats.

A diagnostic command code summary is included for quick reference.

Diagnostic Execution Sequence

The internal diagnostics commands should be invoked by the host in the following order for the most thorough controller hardware verification.

STEP	HEX CODE...	CALLED...	IS USED TO...
1	04	Controller Internal Diagnostics	test the logical operation of the controller, the program memory checksum, and the ECC circuits. This ensures the controller and host can communicate.
2	E0	RAM Diagnostic	verify the data buffers by writing, reading, and verifying data patterns to and from all locations.
3	15	Check Disk Speed	allow the host to perform a spindle speed check on the specified disk drive.
4	E3	Drive Diagnostic	issue a Recalibrate to the disk drive and step through all tracks, verifying the ECC on ID fields of the first sector of each track. When successful, it implies the disk is formatted and the first ID field of each track is good.

Comment

Before drive diagnostics can be executed, the drive configuration must be sent to the controller using the Initialize Drive Characteristics command (HEX 0C). The host program should then issue a Test Drive Ready command (HEX 00) to ensure that the drive is on line and ready.

Diagnostic command code summary

The following table is a summary of the diagnostic command codes.

HEX CODE	COMMAND
15	Check Disk Speed
E0	RAM Diagnostic
E3	Drive Diagnostic
E4	Controller Internal Diagnostic
E5	Read Long
E6	Write Long

Check Disk Speed (HEX 15)

Description

The Check Disk Speed diagnostic allows the host to check disk drive spindle speed. The speed is returned as a counter value. The counter starts when index is detected, is incremented in 1-usec steps, and stops on the next index pulse. This gives the time of one disk revolution and provides speed check accuracy of ±5 usec.

HEX 15 Device Command Field

Note:

dd = drive LUN

BIT	7	6	5	4	3	2	1	0
BYTE 0	0	0	0	1	0	1	0	1
BYTE 1	0	d	d	NOT USED				
BYTE 2	NOT USED							
BYTE 3	NOT USED							
BYTE 4	NOT USED							
BYTE 5	NOT USED							

Returned data format

The returned data is in the following format:

BIT	7	6	5	4	3	2	1	0
BYTE 0	SPINDLE SPEED COUNT (MSB)							
BYTE 1	SPINDLE SPEED COUNT							
BYTE 2	SPINDLE SPEED COUNT (LSB)							
BYTE 3	0	0	0	0	0	0	0	0
BYTE 4	0	0	0	0	0	0	0	0
BYTE 5	0	0	0	0	0	0	0	0

RAM Diagnostic (HEX E0)

Description

The RAM Diagnostic tests controller RAM. Walking 1's and walking 0's test patterns are performed.

HEX E0 Device Command Field

Note:

X = don't care

BIT	7	6	5	4	3	2	1	0
BYTE 0	1	1	1	0	0	0	0	0
BYTE 1	X	0	0	NOT USED				
BYTE 2	NOT USED							
BYTE 3	NOT USED							
BYTE 4	NOT USED							
BYTE 5	NOT USED							

Drive Diagnostic (HEX E3)

Description

The Drive Diagnostic tests both the drive and the drive-to-controller interface. The controller issues Recalibrate and Seek commands to the selected drive and verifies sector 0 of all ID fields on the disk. The controller does not perform any write operations during this command; it is assumed that the disk has been previously formatted.

HEX E3 Device Command Field

Note:

dd = drive LUN
r = retries

BIT	7	6	5	4	3	2	1	0
BYTE 0	1	1	1	0	0	0	1	1
BYTE 1	0	d	d	NOT USED				
BYTE 2	NOT USED							
BYTE 3	NOT USED							
BYTE 4	NOT USED							
BYTE 5	r	0	0	0	0	0	0	0

Controller Internal Diagnostic (HEX E4)

Description

The Controller Internal Diagnostic tests EPROM and ECC circuitry for proper function. The controller does not access the disk during this diagnostic.

The EPROM is tested by adding the value of each memory location modulo 256 across the programmed area. The newly calculated checksum is compared to the checksum stored permanently in the EPROM. If the checksums are not equal, Error Code 31 is returned.

The ECC circuitry is tested by introducing an artificial error to the data and checking that the ECC circuitry detects the error. It also passes a good pattern and monitors the ECC circuitry to verify no ECC error.

HEX E4 Device Command Field

Note:

X = don't care

BIT	7	6	5	4	3	2	1	0
BYTE 0	1	1	1	0	0	1	0	0
BYTE 1	X	0	0	NOT USED				
BYTE 2	NOT USED							
BYTE 3	NOT USED							
BYTE 4	NOT USED							
BYTE 5	NOT USED							

Read Long (HEX E5)

Description

The Read Long diagnostic tests the ECC circuitry. During a normal READ command, the controller reads the 256 or 512 data bytes plus the four ECC bytes into the buffer. The controller sends only the 256 data bytes to the host. The four ECC bytes are used to determine if an ECC data error occurred.

Using the Read Long command, the controller appends the four ECC bytes to the data transfer, making the sector transfer 260 bytes long.

The ECC is not checked for errors during a Read Long command.

HEX E5 Device Command Field

Note:

```
dd = drive LUN
 r = retries
```

BIT	7	6	5	4	3	2	1	0
BYTE 0	1	1	1	0	0	1	0	1
BYTE 1	0	d	d	LOGICAL ADDR 2 (MSB)				
BYTE 2	LOGICAL ADDR 1							
BYTE 3	LOGICAL ADDR 0 (LSB)							
BYTE 4	NOT USED							
BYTE 5	r	0	0	0	0	0	0	0

Write Long (HEX E6)

Description

The Write Long diagnostic tests the ECC circuitry. During a normal Write command, the host supplies the 256 or 512 data bytes to the controller. The controller sends the 256 data bytes and appends the four bytes generated by the ECC hardware to the disk.

Using the Write Long command, the host must supply the four bytes of ECC information following the 256 bytes of data to the disk, making the sector transfer 260 bytes long.

HEX E6 Device Command Field

Note:

```
dd = drive LUN
 r = retries
```

BIT	7	6	5	4	3	2	1	0
BYTE 0	1	1	1	0	0	1	1	0
BYTE 1	0	d	d	LOGICAL ADDR 2 (MSB)				
BYTE 2	LOGICAL ADDR 1							
BYTE 3	LOGICAL ADDR 0 (LSB)							
BYTE 4	0	NOT USED						
BYTE 5	r	0	0	0	0	0	0	0

INDEX

Alphabetic Index

D

Error code, 8-7
Error code format, 5-22
Error code summary:
  disk drive, 6-2, 6-3, 6-4
  tape drive, 8-2