J. GERARD
Oct 84

XEBEC
432 LAKESIDE DRIVE
SUNNYVALE, CALIFORNIA 94086
(408) 733-4200

S1410A
5.25 INCH WINCHESTER DISK CONTROLLER

OWNER'S MANUAL
DOCUMENT #104478

REVISION B
FEBRUARY, 1984

PREFACE

This manual contains technical information on the S1410A 5.25 Inch Winchester Disk Controller. New features have been added to the S1410A over the original S1410. The following is a list of additions:

1. Write Precompensation is selectable at 0, 5, and 10 nanosecs. See Section 4.5.3.13 (Initialize Drive Characteristics).

2. Drives that have Imbedded Servo at index now operate on the S1410A. See Sections 4.5.1 (Control Byte) and 4.5.3.5 (Format Drive).

3. All non-buffered step options have been removed except the 3.0 millisec. See Section 4.5.1 (Control Byte).

4. Drives that have more than 8 heads and use the Reduced Write Current signal as head select 2^3 now operate on the S1410A. See Section 4.1.4.3 (Head Select Signals).

5. Two new commands have been added: Read Verify (Section 4.5.3.10) and Retry Statistics (Section 4.5.3.25).

6. During disk read operations, the S1410A operates different from the S1410 when reporting corectable errors (type 1, error code 8) to the host. See Sections 4.5.3.9 (Read) and 4.5.3.10 (Read Verify).

7. The sector format for the 256 byte sector size of the S1410A is not compatable with the S1410. See Section 4.6 (Sector Format).

8. The S1410A contains four new mounting holes, in addition to the four mounting holes on the original S1410. See Section 2.3 (Physical Requirements).

9. A Right Angle Power Connector (P1) is used on the S1410A. (The mating connectors and pins as used for the S1410 remain the same.)

10 Hardware has been added to provide for extra signals for the removeable cartridge Winchester Drives. (Firmware options will be available at a later date).

CONTENTS

INTRODUCTION

SPECIFICATIONS

BOARD SET-UP

THEORY OF OPERATION

CONTENTS (CONTINUED)

THEORY OF OPERATION (CONTINUED)

CONTENTS (CONTINUED)

THEORY OF OPERATION (CONTINUED)

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1
INTRODUCTION

1.1 GENERAL

The Xebec S1410A Disk Controller can control the operation of up to two 5¼-inch Winchester disk drives that have the industry standard Seagate interface, including those drives with index servo area. This means that the S1410A Controller can operate with a large and growing class of 5¼-inch Winchester disk drives.

1.2 DESCRIPTION

The S1410A Controller, shown in Figure 1-1, is packaged on a compact printed circuit board whose dimensions are 5-3/4 by 8 inches. The board with this popular form factor mounts easily on many 5¼-inch drives. If not mounted directly on the drive, the controller takes up very little space in a typical drive enclosure. Because the controller uses the Shugart Associates System Interface (SASI), it does not require special or complex design considerations in order to communicate with popular host buses. The following list highlights the operating and design features of the controller.

- Interlocked data transfer through the Shugart Associates System Interface (SASI).
- Microprocessor-based architecture (patent pending).
- Full-sector buffer, (256 or 512 bytes).
- Hardware 32-bit ECC polynomial with 11-bit burst correction.
- Field-proven data separator.
- Defacto Industry Standard (ST-506/412) disk interface.
- Automatic retries during disk access.
- Internal Diagnostics.
- Automatic burst error detection and correction.
- Separate sector format for ID and data fields with individual ECC fields for both the ID and data fields.
- High level command set.
- Variable Interleave.

1.3 FUNCTIONAL ORGANIZATION

The simplified block diagram in Figure 1-2 shows the functional organization of the Controller. Only the major areas are shown.

1.3.1 Host Interface

The host interface connects the internal data bus to the host adapter; the state machine controls the movement of data and commands through the host interface.

1.3.2 Processor

The eight-bit processor is the intelligence of the controller; it monitors and controls the operation of the controller.

1.3.3 State Machine

The state machine controls and synchronizes the operation of the host adapter, SERDES, and sector buffer.

1.3.4 SERDES

The serializer/deserializer (SERDES) converts parallel data from the internal data bus to serial data for transfer to a selected disk drive. It converts serial data from the selected disk drive to parallel data which it places on the internal data bus.

1.3.5 Data Separator

The data separator converts serial NRZ data to MFM for transfer to the selected disk drive. It converts MFM data coming from the selected disk drive to serial NRZ data for the SERDES.

1.3.6 Sector Buffer

The sector buffer stages data transfers between the disk and the host to prevent data overruns.

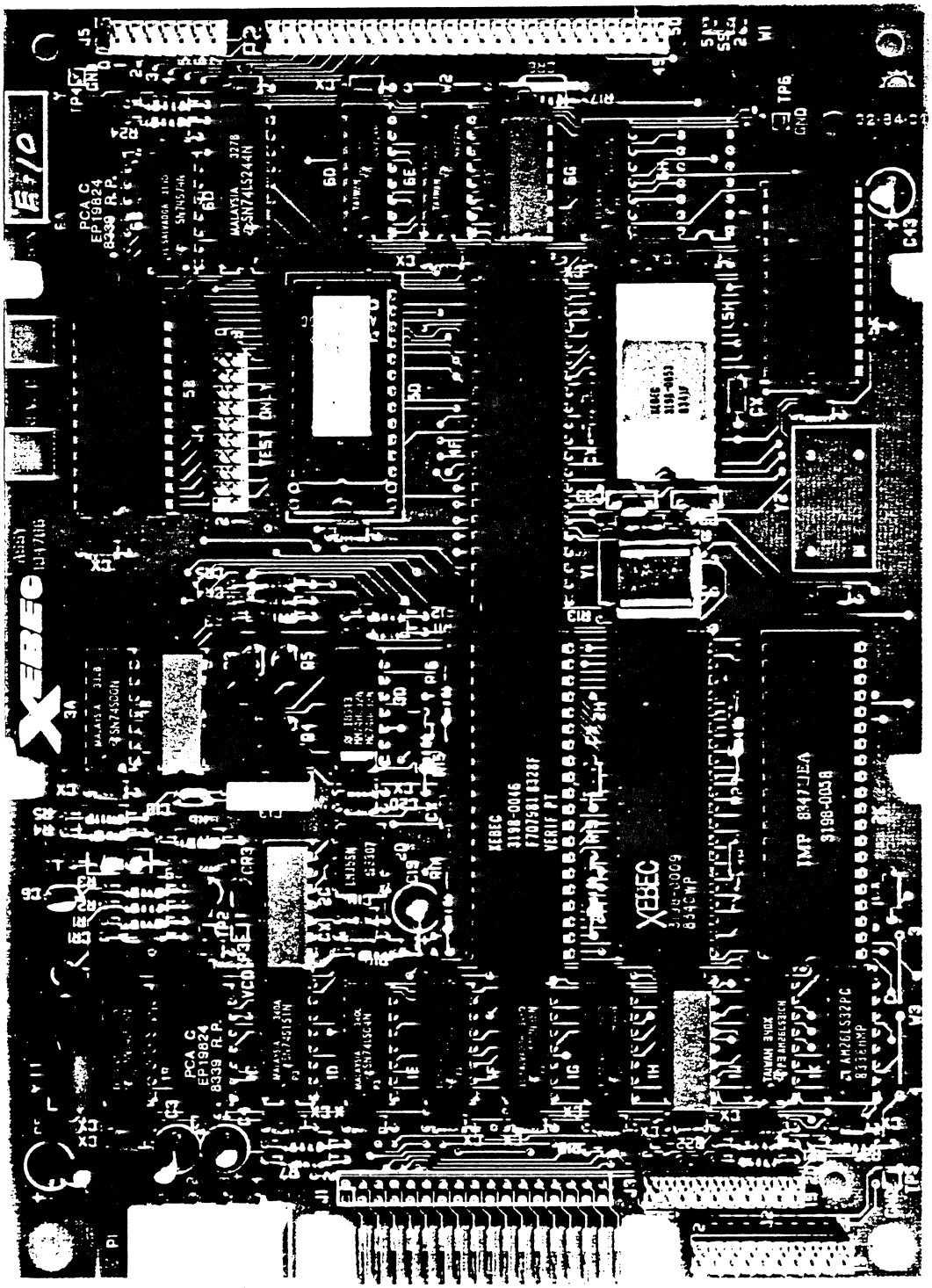


FIGURE 1-1 S1410A DISK CONTROLLER

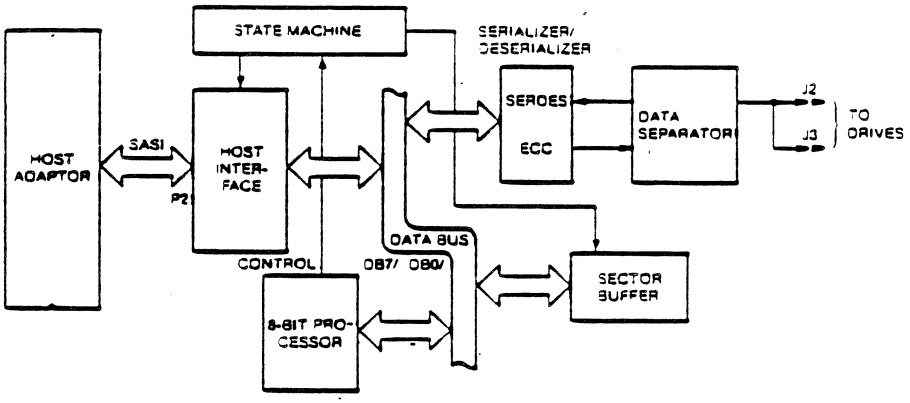


FIGURE 1-2 S1410A CONTROLLER, FUNCTIONAL ORGANIZATION

CHAPTER 2
SPECIFICATIONS

2.1 GENERAL

This chapter contains the overall specifications for the Controller. These specifications are meant to guide the user in placing the controller into operation. Some of the specifications indicate limits; the user must adhere to these in order to operate the controller successfully.

2.2 ELECTRICAL

Table 2-1 lists the electrical requirements of the controller.

TABLE 2-1 CONTROLLER ELECTRICAL REQUIREMENTS

NOTE: All measurements are made on the controller printed circuit board at the power connector P1.

Voltage	Range	Current
+5.0 Vdc	4.75 to 5.25 Vdc	1.3 Amp. Max.
		1.0 Amp. Typ.
+12.0 Vdc	10.8 to 13.2 Vdc	6.0 ma. Max.
		3.0 ma. Typ.

NOTE: The maximum conducted power supply ripple must not exceed 200 mv peak to peak from 0.1 to 25 mHz.

2.3 PHYSICAL SPECIFICATIONS

Table 2-2 lists the specifications of the controller board and Figure 2-1 illustrates the dimensions of the board.

TABLE 2-2 CONTROLLER BOARD SPECIFICATIONS

Item	Measurement
Width (W)	5.75 inches
Length (L)	8.00 inches
Height (H) (Board thickness, components and lead protrusion)	.75 inches
Weight	7.5 ounces

2.4 ENVIRONMENTAL REQUIREMENTS

The controller will operate under the environmental conditions listed in Table 2-3. The controller does not normally require fans in standard operating environments where airflow is not restricted.

TABLE 2-3 ENVIRONMENTAL LIMITS

Item	Measurement
Temperature	0 to 50 degrees Celsius
Relative Humidity	10 to 95 percent non-condensing
Altitude	Sea level to 10,000 feet

2.5 CONNECTORS

Table 2-4 lists the Controller mating connectors.

TABLE 2-4 CONTROLLER MATING CONNECTORS

Designation	Function	Type/Source (or equivalent)
J1	Drive control signals	3M 3463-0001
J2, J3	Drive data signals	3M 3421-7000
J4	Test connector	Not applicable
J5	Controller select number	Not applicable
P1	Power supply	AMP 1-480424-0 (housing) AMP 350078-4 (pins)
P2	Host interface	3M 3425-7000

NOTE: The user **must not** connect a cable to connector J4 or J5.

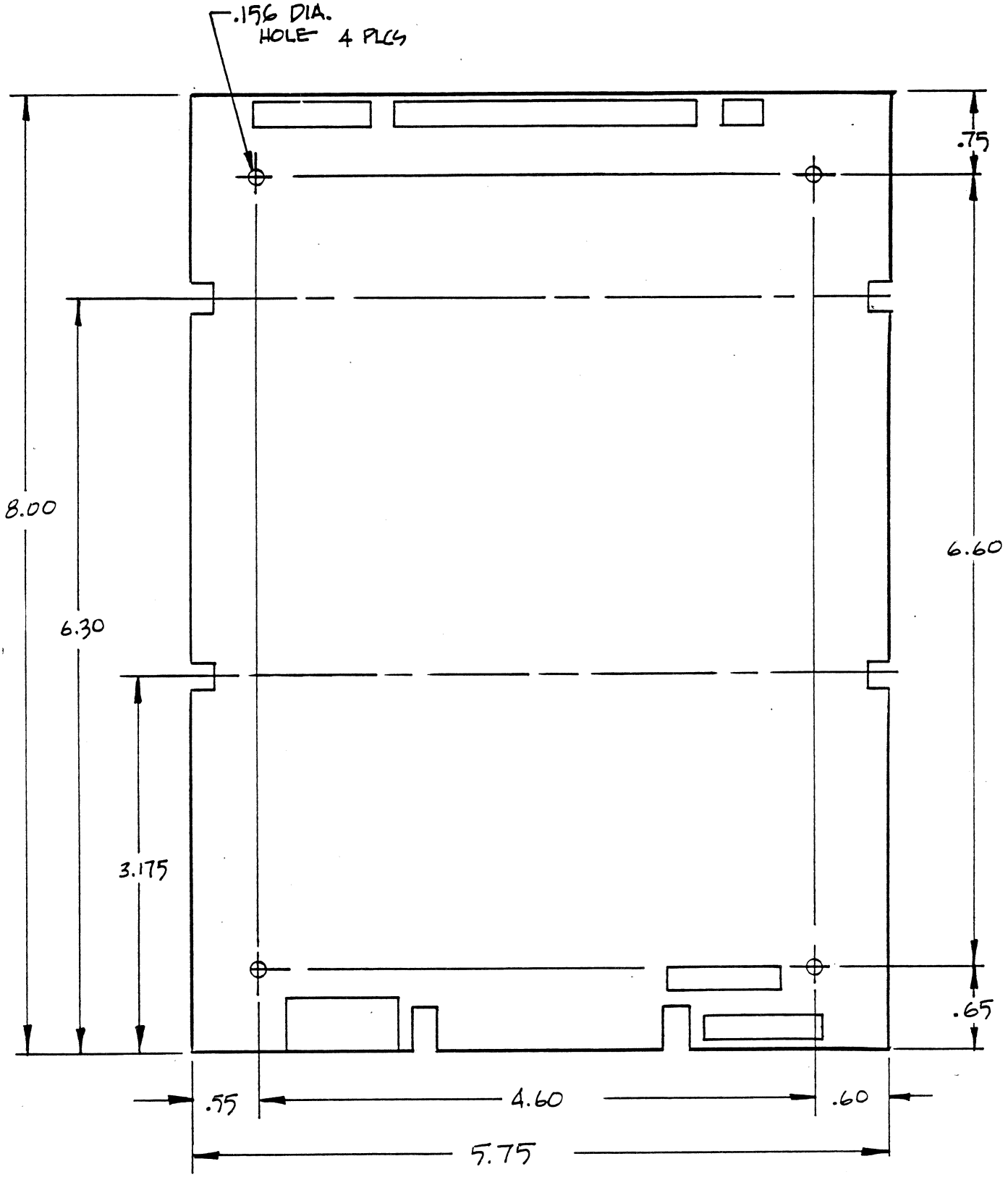


FIGURE 2-1 CONTROLLER BOARD DIMENSIONS

2.6 CONNECTOR PIN ASSIGNMENTS

Tables 2-5 through 2-8 list the pin assignments of the connectors on the controller board. The tables identify the signals on the pins. The connector P2 signals are defined in Chapter 4, Theory of Operation.

TABLE 2-5

CONNECTOR J1, CONTROL SIGNALS, PIN ASSIGNMENTS

Signal Pin	Ground Return	Signal Name
2	1	Reduced Write Current/ Head Select 2^3/-Change Cartridge
4	3	Head Select 2^2
6	5	Write Select Gate
8	7	Seek Complete
10	9	Track 00
12	11	Write Fault
14	13	Head Select 2^0
16	15	Spare / -Sector
18	17	Head Select 2^1
20	19	Index
22	21	Ready
24	23	Step
26	25	Drive Select 1
28	27	Drive Select 2
30	29	Reserved
32	31	Reserved
34	33	Direction In

TABLE 2-6

CONNECTORS J2 and J3, SIGNALS, PIN ASSIGNMENTS

Signal Pin	Ground Return	Signal Name
1	2	Drive Selected
3	4	Spare / -Recalibrate
5	6	Spare / -Write protected
7	8	Reserved
9	10	Spare / -Cartridge Changed
11	12	Ground (GND)
13		MFM Write Data+
14		MFM Write Data-
15	16	Ground (GND)
17		MFM Read Data+
18		MFM Read Data-
20	19	Ground (GND)

TABLE 2-7

CONNECTOR P2, HOST INTERFACE, PIN ASSIGNMENTS

Signal Pin	Ground Return	Signal Name
2	1	DATA0-
4	3	DATA1-
6	5	DATA2-
8	7	DATA3-
10	9	DATA4-
12	11	DATA5-
14	13	DATA6-
16	15	DATA7-
18	17	Spare
20	19	Spare
22	21	Spare
24	23	Spare
26	25	Reserved
28	27	Spare
30	29	Spare
32	31	Spare
34	33	Spare
36	35	BUSY-
38	37	ACK-
40	39	RST-
42	41	MSG-
44	43	SEL-
46	45	C-/D
48	47	REQ-
50	49	I-/0

TABLE 2-8

CONNECTOR P1, POWER SUPPLY, PIN ASSIGNMENTS

Pin Number	Voltage
1	+12 Vdc
2	Ground return
3	Ground return
4	+5 Vdc

CHAPTER 3
BOARD SETUP

3.1 GENERAL

This chapter contains the information for setting up and installing the controller before placing it in operation. These preparatory steps require the proper placement of jumpers, mounting the controller in its operating environment, and properly connecting the cables. In addition, the user has the option of using more than one controller with the host adapter in his system. Instructions for connecting multiple controllers appear later in the chapter.

3.2 BOARD SETUP

Setting up the Controller for operation requires checking that the factory-installed jumpers are according the listing in Table 3-1. Figure 3-1 shows the locations of the jumpers on the Controller.

TABLE 3-1 JUMPER LOCATIONS

Designation	Function	Connection and Result
W1	Selects sector size	SS to 2: 256-byte sector 32 Sectors/Track SS to 5: 512-byte sector 17 Sectors/Track

3.3 MOUNTING CONTROLLER

The controller board has eight mounting locations. The four mounting holes match those of the original S1410, while the four sloted mounting locations allow for easy mounting on the drive. The controller should be mounted to receive convection air flow.

3.4 CONNECTING CABLES

Before the controller can be placed in operation, the cables to the drive and host must be connected. These cables are listed below:

J1	Control Cable (controller to last drive): maximum 20 feet
J2	Data Cable: maximum 20 feet
J3	Data Cable: maximum 20 feet (optional second drive)
P1	Power Cable
P2	Host Interface Cable: maximum 15 feet

NOTE: Do not attempt to connect a cable to connector J4 or J5. Connector J4 is for factory test only. Connector J5 sets the controller SASI address.

Figure 3-1 shows the connector locations.

3.5 MULTIPLE CONTROLLERS

A separate Controller is required for each additional pair of drives. Figure 3-2 shows two operating setups; (A) using one Controller and (B) using two Controllers. Notice the terminator (resistor pack) in both drawings. The terminator is at position 6F on the board. When multiple Controllers are used, the terminator must be installed only in the last board in the daisy chain.

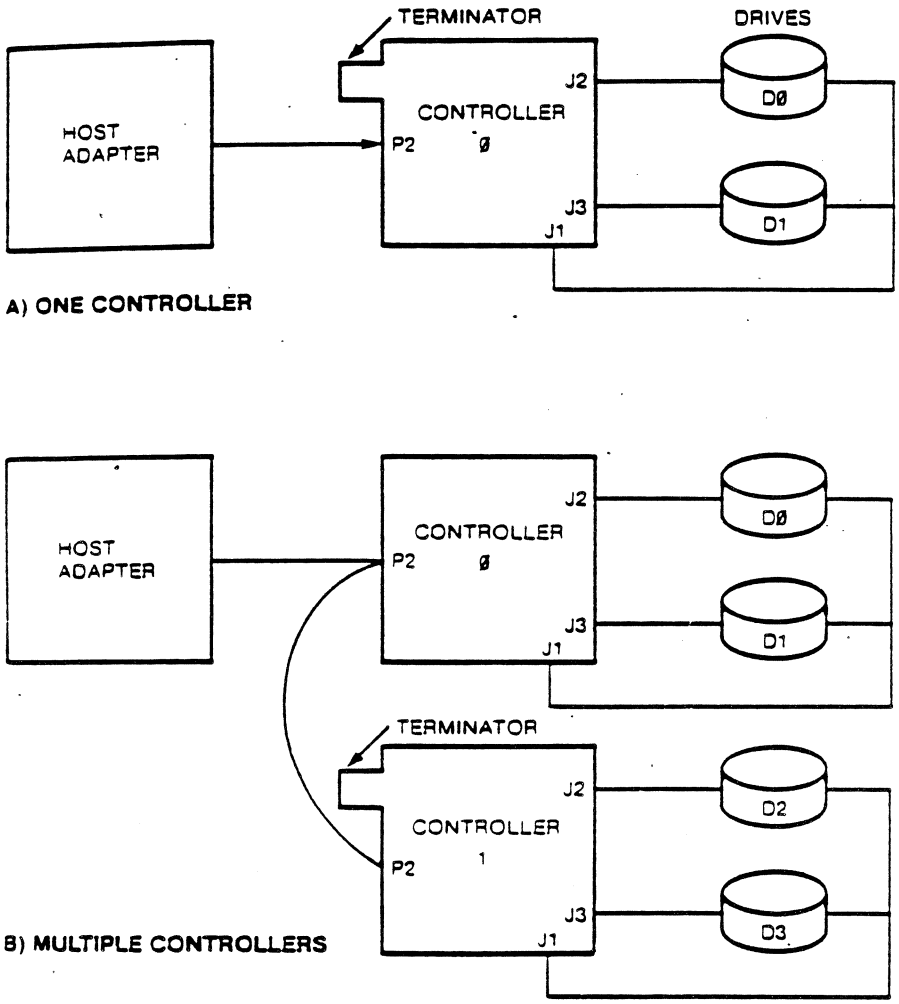


FIGURE 3-2 OPERATING SETUPS

3.6 ADDRESS JUMPER GROUP

The Controller supports one of eight unique device addresses. When more than one controller is used in a system, the address jumper on the controller must be changed. Figure 3-1 shows the address jumper group (J5) located next to the connector P2. It also shows that terminal (pad) 0 is connected. This is the factory-installed jumper, and it sets the controller's address to 0.

In order to change this address, the factory-installed jumper must be removed from postion zero and connected between terminals at a different selected address. The position of this jumper corresponds to the bit position on the SASI bus that must be set, when the select strobe is issued, to select this controller. Each controller attached to a SASI bus must use a dfferent SASI address.

CHAPTER 4
THEORY OF OPERATION

4.1 GENERAL

This chapter discusses the theory of operation of the S1410A Controller and lays down the guidelines that will enable the user to use the controller successfully.

4.1.1 SASI Conventions

Signals or lines can be active in either a high or low state. The terms signal, signal lines, and lines mean the same thing. A low state is equivalent to a voltage level of 0.8 volts or less, and a high state is equivalent to a voltage level of 2.4 volts or more. Some texts use the term 'asserted' to mean active. In this manual, only the term active is used; if the term 'asserted' appears, it is only for reference.

4.1.2 SASI Names and Abbreviations

A dash (-), or the lack of one, indicates the active state of a signal. The active state of a signal is that state which is required for a given operation. When a dash is appended to end of a signal name, the signal is active when it is low. When no dash appears at the end of a signal name, the signal is active when it is high. Some signal lines have two so-called active (or significant) states. When the level on the line is high, a particular operation takes place. When the level on the line is low, a different operation takes place. The following examples show the use of these conventions.

BUSY- The signal BUSY- is active when it is at a low level because it has the dash appended.

BUSY The signal BUSY is active at a high level because it does not have the dash appended.

C-/D The line C-/D (command-/data) has a dual purpose. The dash after the C indicates that when this line is at a low level, command mode is indicted and when it is at a high level, data mode is indicted.

Other designations used to define signal lines are listed below.

Drv	Driver
Rcvr	Receiver
OC	Open collector
Tri-State	Line has three states: high, low, high impedance
220/330	Line termination: 220 Ohms to source voltage/330 Ohms to ground.

4.1.3 SASI Signal Definitions

The following tables list and define the signals that appear on the SASI Bus lines between the host adapter and the controller.

TABLE 4-1 HOST BUS STATUS SIGNALS

NAME	DRV/RCVR	DEFINITION
I-/O	Drv OC	Input-/Output: The controller drives this line. A low level on this line indicates that the controller is driving the data in on the host bus. A high level on this line indicates that the host adapter is driving the data out on the host bus. The host adapter monitors this line and uses it to enable and disable its data bus drivers. This signal is qualified by signal REQ-.

TABLE 4-1 HOST BUS STATUS SIGNALS (Continued)

C-/D	Drv OC	Command-/Data: This signal line indicates whether the information on the data bus consists of command or data bytes. A low means command bytes; a high means data bytes. This signal is qualified by signal REQ-.
BUSY-	Drv OC	Busy: The controller generates this active low signal in response to the SEL- signal and the address bit (DB0- to DB7-) from the host adapter. The busy signal informs the host adapter the controller is present and ready to conduct transactions on the host bus.
MSG-	Drv OC	Message: The controller sends this active low signal to the host adapter to indicate that the current command has been completed. When MSG- is active, the I-/O signal line is always low so that the controller can drive the bus data lines. This signal is qualified by signal REQ-.

TABLE 4-2 SUMMARY OF HOST BUS STATUS SIGNALS

I-/O	C-/D	MSG-	DEFINITION
High	Low	High	The controller receives command bytes from the host adapter (All Commands).
High	High	High	The controller receives data bytes from the host adapter. Write Data Drive initialize Data Write Buffer Data Format Alternate Track Data Write Long Data
Low	High	High	The controller sends data bytes to the host adapter. Read Data Request Sense Data Read ECC Burst Error Data Read Sector Buffer Data Read Long Data Retry Statistics Data
Low	Low	High	The controller sends error status byte to the host adapter (All Commands).
Low	Low	Low	The controller sends message byte to host which informs the host adapter that it has completed the current command (All Commands).

TABLE 4-3 CONTROLLER-HOST ADAPTER HANDSHAKING

NAME	DRV/RCVR	DEFINITION
REQ-	Drv OC	Request: The controller sends this active low signal to the host adapter for each byte transferred across the interface. This signal qualifies signals I-/O, C-/D and MSG-.
ACK-	Rcvr, 220/330	Acknowledge: The host adapter generates this active low signal in response to the REQ- signal from the controller when the host is ready to receive or transmit a byte of data. In order to complete the handshake, the host adapter must send an acknowledge (ACK-) in response to each request (REQ-) from the controller.

TABLE 4-4 HOST BUS CONTROL SIGNALS

NAME	DRV/RCVR	DEFINITION
RST-	Rcvr, 220/330	Reset: The host adapter sends this active low signal to the controller to force the controller to the idle state. After RST- has become active, any controller status is cleared. RST- also causes the deactivation of all signals to the drives. Select should not be issued for 20 microseconds after a reset pulse. The time requirements for the RST- signal are as follows:

Minimum	Maximum
100 nsec.	None

SEL-	Rcvr, 220/330	Select: The host adapter sends this active low signal to the controller to initiate a command transaction. BUSY- must be in the inactive state at the beginning of the select process. Along with SEL-, the host adapter must also send an address bit to select the controller (DB0- for controller 0). The data bus must be stable 100 nsec before SEL- is active, and remain stable for 100 nsec after SEL- is deactivated. Minimum select pulse width is 100 nsec, there is no maximum, as long as the data bus remains stable during select. SEL- should not be issued for 25 microseconds after a reset.

TABLE 4-5 HOST BUS DATA SIGNAL

NAME	DRV/RCVR	DEFINITION
DB7- to DB0-	Tri-State, 220/330	These are the eight data bits (lines) of the host bus (DB0 = LSB). Each line is also used as address bits to select a controller in systems using multiple controllers (see Chapter 3). The normal connection is to DB0- which is the address of controller 0. Any other connection requires moving the jumper connection at J5. The following list shows the bit assignments. DB0- Controller 0 DB1- Controller 1 DB2- Controller 2 DB3- Controller 3 DB4- Controller 4 DB5- Controller 5 DB6- Controller 6 DB7- Controller 7

4.1.4 Drive Control Input Signals

The control input signals are of two kinds: those to be multiplexed in a multiple drive system and those that do the multiplexing. The control input signals to be multiplexed are: Reduced Write Current, Write Gate, Head Select Line 0, Head Select Line 1, Head Select Line 2, Step, and Direction. The signal to do the multiplexing is Drive Select 0 and Drive Select 1.

The input lines have the following electrical specifications as measured at the drive.

True: 0.0 volt D.C. to 0.4 volt D.C. @I=40 milliamperes, maximum

False: 2.5 volts D.C. to 5.25 volts D.C. @IL=250 microamperes, maximum

All input lines share a 220/330 ohm resistor pack for line termination. Only the last drive in the chain should have the resistor pack installed.

NOTE: Refer to drive specifications for detailed information of these line.

4.1.4.1 Reduced Write Current

When active, this line, together with Write Gate, causes the write circuitry to write on the disk with a lower write current. This line is set active at the cylinder specified in the initialize command.

4.1.4.2 Write Select Gate

The active state of this signal or logical zero level enables write data to be written on the disk. The inactive state of this signal enables the data to be transferred from the drive. In addition, the inactive state enables the step pulse to step the read/write actuator. MFM write data is sent to the drive within 400 nsec after write is active.

4.1.4.3 Head Select Signals $2^0, 2^1, 2^2, 2^3$

These four lines provide for the selection of each read/write head in a binary coded sequence. Head Select Line 2^0 is the least significant line. When all Head Select Lines are false, Head 0 is selected. If drive initialize is for greater than 8 heads, reduce Write Current line is used as Head Select 2^3.

4.1.4.4 Step

This interface line is a control signal that causes the read/write head to move with the direction of motion defined by the Direction In line.

The access motion is initiated at the logical true-to-logical false transition or the trailing edge of this signal pulse. Change in the Direction In line is made at least 5.0 microseconds before the leading edge of the step pulse.

The minimum pluse width is 5.0 microseconds.

4.1.4.5 Direction In

This signal defines the direction of motion of the read/write head when the Step line is pulsed. An open circuit or logical false defines the direction as "out". If a pulse is applied to the Step line, the read/write heads move away from the center of the disk. If this line is logical true, the direction is defined as "in", and the read/write heads move in toward the center of the disk. The direction line is set a minimum of 5 nsec before step pulse are issued.

Seek Complete is verified to be true prior to changing directions and the application of additional step pulses.

4.1.4.6 Drive Select 0, Drive Select 1

These control signals enable the selected drive's input receivers and output drivers. When logically false, the output drivers are open circuits or logically false and the input receivers do not acknowledge signals presented to them.

Drive addresses are assigned on the drive. Refer to Drive Manual for drive selection.

NOTE: Only one drive may be selected at a time.

4.1.5 Drive Output Signals

The output control signals are driven with an open collector output stage capable of sinking a maximum of 40 milliamperes in a logical true state with a maximum voltage of 0.4 volt measured at the driver. When the line driver is in the logical false state, the driver transistor is off, and the collector cutoff is a maximum of 250 microamperes.

All output lines are enabled by the respective Drive Select lines.

4.1.5.1 Seek Complete

This line goes true when the read/write heads have settled on the final track at the end of a seek. Reading or writing is not attempted when Seek Complete is false.

4.1.5.2 Track 000

This interface signal indicates a true state only when the drive's read/write heads are positioned at Track 000, the outermost data track.

4.1.5.3 Write Fault

This signal is used to indicate that a condition exists at the drive that could cause improper writing on the disk. When this line is true, further writing is inhibited. This line is edge detected, and cleared on the controller by deselection of the drive.

4.1.5.4 Index

This interface signal is provided by the drive once each revolution (16.7 milliseconds nominal) to indicate the beginning of the track. Normally, this signal is logical false and makes the transition to logical true to indicate Index. This line is edge detected on the logical false to logical true transition. The minimum pulse width is 200 nsecs, with the maximum pluse width less than one revolution of the disk.

4.1.5.5 Ready

When true, this interface signal, together with Seek Complete, indicates that the drive is ready to read, write, or seek, and that the I/O signals are valid. When this line is false, all controller-initiated functions are inhibited.

4.1.5.6 Select Status

A status line is provided at the J2,J3 connector to inform the controller of the selection status of the drive. This line is used to determine which port the drive data cable is connected.

4.1.6 Drive Data Transfer Signals

Data lines associated with the transfer of data between the drive and the S1410A are differential in nature and may not be multiplexed. These lines are provided at the J3,J2 connector. Signal levels are defined by the RS-422A specification.

Two pairs of balanced lines are used for the transfer of data: MFM Write Data and MFM Read Data.

4.1.6.1 MFM Write Data

This is a differential pair of lines that defines the flux transition to be written on the track. The transition of the +MFM Write Data line going more positive than the -MFM Write Data line causes a flux reversal on the track if the Write Gate is active. This signal is driven to an inactive state (+MFM Write Data more negative then -MFM Write Data) by the S1410A when in a read mode. Write Gate is inactive.

The delay from the leading edge of Write Gate to the Write Data pulse is 400 nanoseconds maximum.

4.1.6.2 MFM Read Data

The data recovered by reading a prerecorded track is transmitted to the S1410A via the differential pair of MFM Read Data lines. The transition of the +MFM Read Data lines going more positive then the -MFM Read Data line represents a flux reversal on the track of the selected head. This line is edge detected, minimum active pulse width is 50 nsec, maximum active pulse width is 150 nsec.

4.2 BASIC OPERATING CONFIGURATION

The basic operating configuration consists of a host adapter, S1410A Controller, and a Winchester disk drive with an industry standard Seagate interface. Figure 4-1 shows the basic setup. Also shown is an additional, optional drive; the controller can control a maximum of two drives.

The host can be one of a number of computer systems; the host adapter is an interface between the host's bus and the controller.

4.3 DETAILED DESCRIPTION (HANDSHAKING AND TIMING)

The following paragraphs describe the interaction between the controller and the host adapter.

4.3.1 Controller Selection

Before the host adapter can begin a transaction, it must select the controller. The host adapter selects the controller by activating the SEL- control signal and the address bit of the controller. Any bit, DB0- through DB7-, can be the address bit in a system with multiple controllers (all controllers leave the factory with DB0- connected to the controller's address logic). For this discussion, the controller's address is 0. Only one controller may be selected at a time. BUSY- must not be active at the start of the selection process.

The timing diagram in Figure 4-2 shows the basic timing requirements. Upon receiving both the SEL- signal and DB0-, the controller activates the BUSY- signal. As shown in the timing diagram, both SEL- and DBO- must be active (low) before the controller can activate the BUSY- signal. The data bus must be stable a minimum of 100 nanoseconds before SEL- is set, and held stable for 100 nanoseconds after SEL- is reset. Do not issue a select for at least 25 microseconds following a controller reset. During the selection process, the host has control of the data bus as signified by the deactivation of the I-/O line. Selection is complete when BUSY-becomes active. The controller then enters the command mode.

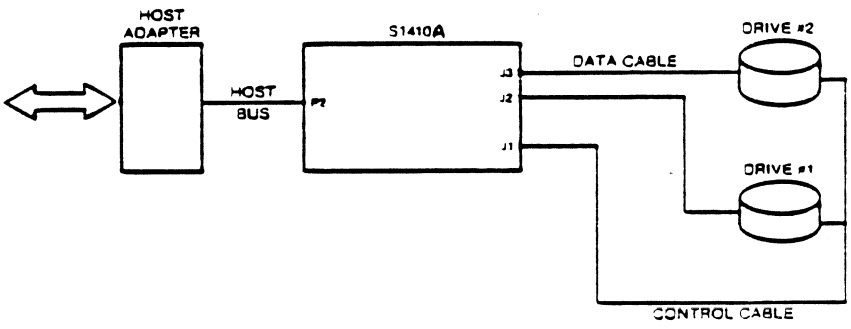


FIGURE 4-1 BASIC OPERATING CONFIGURATION

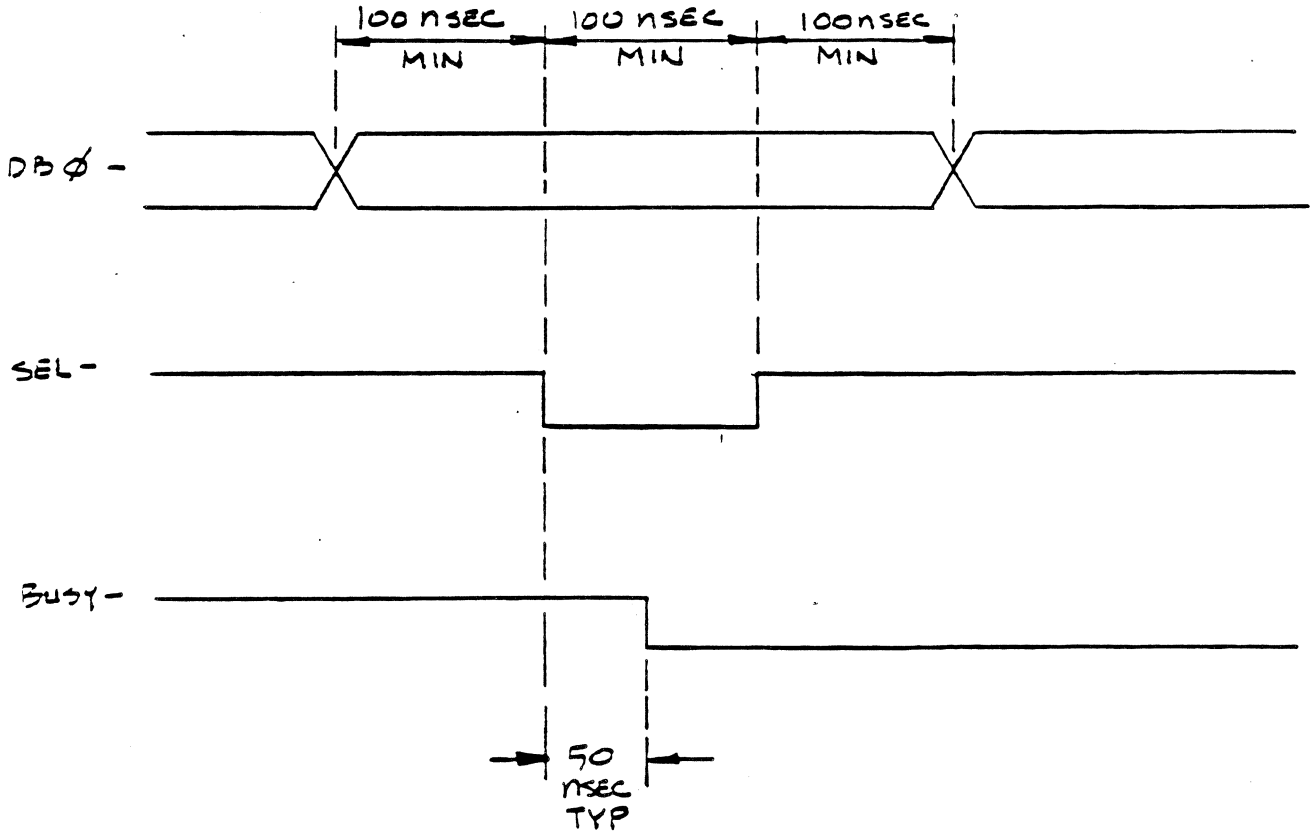


FIGURE 4-2 CONTROLLER SELECT TIMING

4.3.2 Command Mode

The controller receives commands from the host adapter using a handshaking sequence. The controller places a low level on the C-/D (command-/data) line to indicate that it wants a command from the host adapter and places a high level on the I-/O line to indicate that the movement of information is from the host adapter out to the controller. The MSG- line is high.

The controller activates the REQ- line within 20 microseconds after signals I-/O, C-/D and MSG- have been placed at high, low and high levels, respectively. The host adapter responds by activating the ACK- signal when a command byte is ready for the controller. The command byte placed on the data bus by the host must be stable within 250 nanoseconds after the ACK- signal is activated. The command byte must be held stable until REQ- is deactivated. The host deactivates ACK- after REQ- goes high. This completes the handshake for the first command byte. There are 6 command bytes, and each succeeding command byte from the host adapter requires the same complete handshake sequence. See Figure 4-4 and Section 4.3.3.2 for data bus, REQ, and ACK- timing. See Table 4-2 for I-/O, C-/D and MSG- definition.

4.3.3 Data Transfer

4.3.3.1 Data Transfer to Host

On the transfer of data from the controller to the host, the data is stable on the bus a minimum of 125 nanoseconds before REQ- becomes active. There is no time limit from REQ- active to ACK- active. After the controller receives ACK-, REQ- will become inactive within 25 to 75 nanoseconds. If the controller has more data to send, it will set REQ- active within 1.2 to 1.7 microseconds from REQ- going inactive, if ACK- was set inactive within 1.2 microseconds of REQ- going inactive. The controller can not set REQ- active if ACK- is active. If the controller is ready to set REQ- active, but ACK- is

active, the controller waits for ACK- to go inactive, then sets REQ- active between 25 and 50 nanoseconds later. There is no time limit for the controller waiting for ACK- to go inactive. Refer to Figure 4.3 for timing detail.

4.3.3.2 Data Transfer From Host

Each data byte transferred from host to the controller starts with the REQ- going active. For REQ- to go active, the host must ensure that ACK- is inactive at the beginning of each byte transfer sequence. After REQ- goes active, the host sets ACK- active while placing a byte of data on the bus. There is no time limit from the time REQ- goes active to ACK- going active. However, there is a time limit of 250 nanoseconds maximum for the host to place the data byte on the bus referenced from the ACK- edge. When ACK- goes active, the controller sets REQ- inactive within 1000 nanoseconds maximum. The host must not change data on the bus until REQ- goes inactive. There is no hold time for the data when REQ- goes inactive. There is no time limit for the host to set ACK- inactive. The controller will not set REQ- active again, for the next byte to transfer, until ACK- goes inactive. If ACK- goes inactive in response to REQ-going inactive, the controller will set REQ- active within 500 nanoseconds, ready to receive the next byte. If the controller is ready to set REQ- active, but ACK- is active, the controller waits for ACK- to go inactive, then sets REQ- between 25 and 50 nanoseconds later.

4.3.4 Status Bytes

Two bytes of status are passed to the host at the end of all commands. The first byte informs the host if any errors occurred during the execution of the command. The second is a zero byte. It signals to the host that the command is complete. Figure 4-3 shows the data bus, REQ- and ACK- timing. See Table 4-2 for I-/O, C-/D and MSG- definition. Figure 4-5a shows the format of these two bytes.

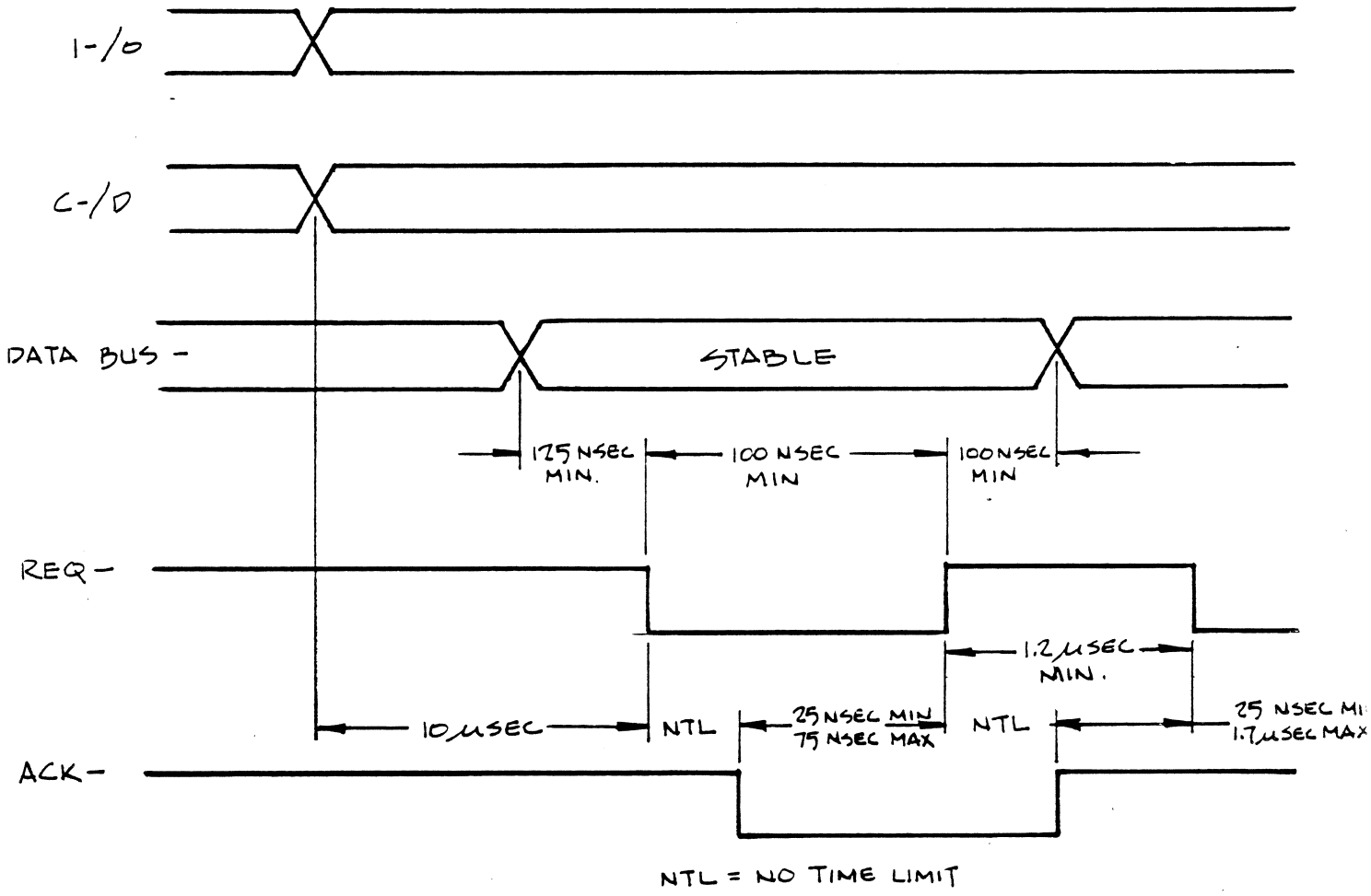


FIGURE 4-3 DATA TRANSFER TO HOST, TIMING

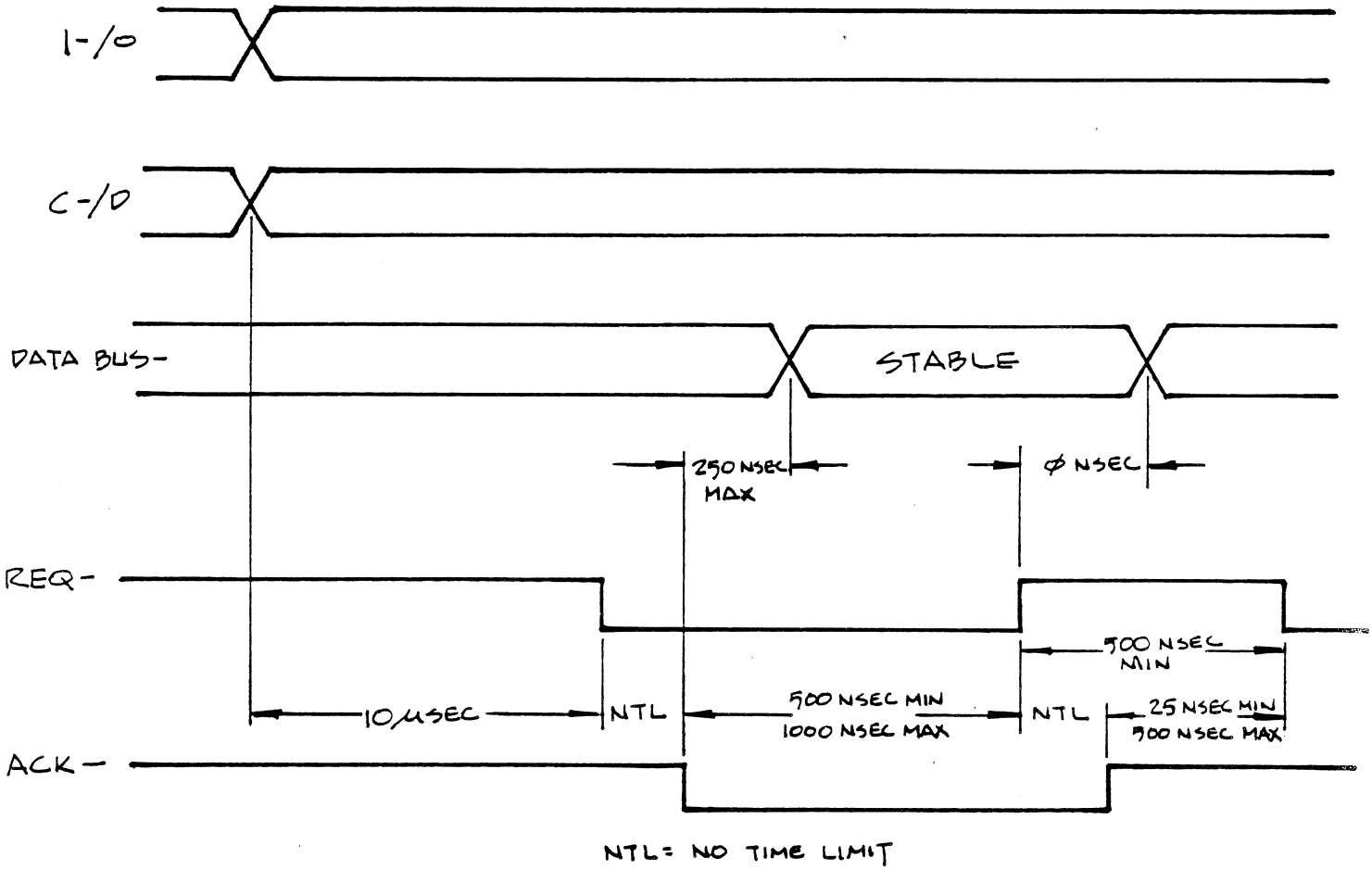


FIGURE 4-4 DATA TRANSFER FROM HOST TIMING

4.4 PROGRAMMING INFORMATION

The following paragraphs discuss communications between the controller and host from the point of view of the command codes that are passed. The host sends commands to the controller through the host adapter. The controller then performs the commands and reports status back to the host.

4.5 COMMANDS

The host sends a six-byte block to the controller to during command mode to specify the operation. This block is the Device Control Block (DCB). Figure 4-5 shows the composition of the DCB. The list that follows Figure 4-5 defines the byte that make up the DCB.

Bit	7	6	5	4	3	2	1	0
Byte 0	Command Class			Opcode				
Byte 1	LUN			High Address				
Byte 2	Middle Address							
Byte 3	Low Address							
Byte 4	Interleave or Block Count							
Byte 5	Control Field							

FIGURE 4-5 DEVICE CONTROL BLOCK (DCB), FORMAT

Byte 0 Bits 7, 6 and 5 identify the class of the command. Bits 4 through 0 contain the opcode of the command.

Byte 1 Bit 5 identifies the logical unit number (LUN). Bits 4 through 0 contain logical address 2.

Byte 2 Bits 7 through 0 contain logical address 1.

Byte 3 Bits 7 through 0 contain logical address 0 (LSB).

Byte 4 Bits 7 through 0 specify the interleave or sector count.

Byte 5 Bits 7 through 0 contain the contain the control field.

Next to Last Status Byte

Bit	7	6	5	4	3	2	1	0
	0	0	d	0	0	0	ERR	0

Bits 0, 2-4,6,7 — Set to zero.

Bit 1 — When set, error occurred during command execution.

Bit 5 — Logical unit number of drive, d=0 or 1.

Last Status Byte

Bit	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0

Bits 0-7 — Set to zero.

Figure 4-6 Completion Status Bytes

4.5.1 The Control Byte.

Bit 7 - If Bit 7 is set, the Controller will not retry the operation, but reports the error immediately. If Bit 7 is reset, the Controller will retry the operation 3 times, then recalibrate the drive, seek back to the current track, and retry once more. For example, if an ADDRESS MARK (AM) NOT FOUND error occurs, the Controller attempts to reread the same sector 3 times. If the error persists, the Controller recalibrates the drive to track 0, seeks to the target track and rereads for the last time. If the error still persists the Controller aborts the command and reports the error to the host. Set this bit only during the evaluation of a disk drive.

Bit 6 - If Bit 6 is set, the Controller will not retry a read of a sector that contains a data error before attempting error correction. If Bit

6 is reset, the Controller will reread the sector before attempting error correction. It is faster and more reliable to read the data again than apply error correction immediately. If the data error is transient in nature, such as noise or electrical interference in the disk heads or read amplifier, noise in cable, or a power supply transient, then another read of the sector will be successful. If the error occurs twice in succession, the error is usually caused by a media defect, so ECC correction is used to recover the data.

Bit 5 - Not used.

Bit 4 - If set to one, indicates the disk drive has disk servo information prior to index on each track. The "Imbedded Servo" drives use this servo information to control the rotational speed of the drive and to control position of the read/write head over the track.

Bit 0-3 - Bits 0-3 will set the step rate for the disc drive. The following describes the step rates provided.

DESCRIPTION	BITS				
	3	2	1	0	
Default 3 msec step rate	0	0	0	0	
Reserved	0	0	0	1	
Reserved	0	0	1	0	
Reserved	0	0	1	1	
200 usec buffered step	0	1	0	0	
70 usec buffered step	0	1	0	1	
30 usec buffered step	0	1	1	0	
15 usec buffered step	0	1	1	1	
12 usec buffered step	1	0	0	0	
Spare (for future use)	1	0	0	1	thru
	1	1	1	1	

4.5.2 Logical Addressing

The logical address of the drive is computed by using the following equation:

Logical Address = (CYADR * HDCYL + HDADR) * SETRK + SEADR

where:

CYADR	Cylinder Address
HDADR	Head Address
SEADR	Sector Address
HDCYL	Number of Heads per Cylinder
SETRK	Number of sectors per track

4.5.3 Command Set

The commands fall into eight classes, 0 through 7; only classes 0 and 7 are used. Class 0 command are data, non-data transfer, and status commands. Classes 1 through 6 are reserved. Class 7 are diagnostic commands.

Each command is described below. The description includes its class, opcode, and format. When a slash (/) represents a bit position, the slash means that the value of that bit is not important (a don't-care bit).

4.5.3.1 Test Drive Ready (Code 00)

This command selects the drive specified by the Byte 1 of DCB and reads back the status from that drive. If all status bits are in the correct state, the command will not return an error code. If the drive status is not OK, the command will return an error code.

This command is usually used in 2 occasions:

A) When initially powered on, the host should issue this command continuously with appropriate time out loop to insure the drive spins up to speed and comes ready.

B) When implementing overlapped seeks. First, issue a seek command to the first drive, then issue another seek command to the second drive. Now keep issuing a TEST DRIVE READY command to each drive until either drive finishes its seek operation. Then continue with the normal READ/WRITE operation on that drive.

The following diagram shows the format of the device control block for this command.

d = drive, 0 or 1

Bit	7	6	5	4	4	2	1	0
Byte 0	0	0	0	0	0	0	0	0
Byte 1	0	0	d	/	/	/	/	/
Byte 2	/	/	/	/	/	/	/	/
Byte 3	/	/	/	/	/	/	/	/
Byte 4	/	/	/	/	/	/	/	/
Byte 5	/	/	/	/	/	/	/	/

At the end of the TEST DRIVE READY command if no error is reported (bit 1 of status byte) then the drive is ready for the next operation. If an error is reported, then a Request Sense Status command must be issued to determine the status of the drive.

4.5.3.2 Recalibrate (Code 01)

This command will move the drive arm to the track 00 position. This command should only be used to attempt to correct a drive position error, since it is slower than a direct seek to track 0. Also, if retries are enabled, the Controller will recalibrate automatically in case of error. The difference between this command and a direct seek to track 0 is this command steps the drive one cylinder at a time looking for the signal TRACK 00 from the drive to become active. A direct seek to track 0 is faster because the Controller steps the drive at the programmed step rate.

d = drive, 0 or 1

r = retries

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	0	0	0	1
Byte 1	0	0	d	/	/	/	/	/
Byte 2	/	/	/	/	/	/	/	/
Byte 3	/	/	/	/	/	/	/	/
Byte 4	/	/	/	/	/	/	/	/
Byte 5	r	0	0	0	0	0	0	0

4.5.3.3 Reserved (Class 0, Opcode 02)

This opcode is not used.

4.5.3.4 Request Sense Status (Code 03)

The host must send this command immediately after it detects an error. The command causes the Controller to return four bytes of drive and Controller status; the formats of these four bytes are shown after the DCB. When an error occurs on a multiple sector data transfer, (read or write), the Request Sense Status command

returns the logical address of the failing sector in bytes 1, 2 and 3. If the Request Sense Status command is issued after any of the Format commands or the Check Track Format command, then the logical address returned by the controller points to one sector beyond the last track formatted or checked if there was no error. If there was an error, then the logical address returned points to the track in error.

d = drive, 0 or 1

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	0	0	1	1
Byte 1	0	0	d	/	/	/	/	/
Byte 2	/	/	/	/	/	/	/	/
Byte 3	/	/	/	/	/	/	/	/
Byte 4	/	/	/	/	/	/	/	/
Byte 5	/	/	/	/	/	/	/	/

SENSE BYTES

Bit	7	6	5	4	3	2	1	0
Byte 0	SEE BELOW							

Bits 0-3 Error Code

Bits 4-5 Error Type

Bits 6 Spare, set to zero

Bits 7 Address valid, when set

The address valid bit in the error code byte (bit 7) is relevant only when the previous command required a logical block address; in which case it is always returned as a one, otherwise it is set to zero. For instance, if a Recalibrate command is followed immediately by a Request Sense Status command, the address valid bit would be returned as zero since this command does not require a logical block address to be passed in its DCB.

Bit	7	6	5	4	3	2	1	0
Byte 1	0	0	d	High Address				
Byte 2	Middle Address							
Byte 3	Low Address							

d = drive, 0 or 1

This section details the S1410A Error Codes returned in the REQUEST SENSE STATUS 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.

TABLE 4-6 TYPE 0 ERROR CODES, DISK DRIVE

HEX CODE	DEFINITION
00	No Error Occurred. This code is always returned if no error had occurred during the previous command.
01	No Index Signal from the Drive. This error occurs during any data transfer or format command if a normal drive select occurs, the drive is ready, but no index signal is detected from the drive within two revolutions of the disk. Possible error causes are: - Bad Drive - Control Cable (J1) - Controller
02	No Seek Complete Signal from the Drive. This error occurs on non-buffered seek processing if the controller does not receive the Seek Complete Signal from the Drive within one second following the last step pulse. Possible error causes are: - Bad Drive - Control Cable (J1) - Controller
03	Write Fault Signal Received from the Drive. This error occurs if the controller detects an active write fault signal from the disk drive either at the completion of a sector data transfer or initially after a successful drive select and the drive indicates ready. Possible error causes: - Drive power supply voltage out of range - Bad Drive - Control Cable (J1) - Unit Cable (J2, J3) - Controller

TABLE 4-6 TYPE 0 ERROR CODES, DISK DRIVE (CONTINUED)

HEX CODE	DEFINITION
04	Disk Drive Not Ready. This error occurs if the controller fails to the select the drive, or the drive indicates not ready after selection. Possible error causes: - Drive power supply voltages out of range - Drive not yet up to operating speed following power on - Bad drive - Control Cable (J1) - Controller
05	Not used
06	Track 00 Not Found. After stepping the drive 200 more steps than the number of configured cylinders during a recalibrate command, the Track 00 Signal was not received fron the drive. Possible causes are: - Incorrect drive size initialization (too few cylinders) - Bad Drive - Control Cable (J1) - Bad controller
07	Not used
08	Disk Drive is Seeking. This code is returned in response to a test drive ready command if the drive had previously been issued a buffered seek command that has not yet completed.
09 to 0F	Not used

TABLE 4-7 TYPE 1 ERROR CODES, CONTROLLER

10 ID Field Read Error. During a data transfer or format command, address marks were detected, but the target sector was not found and an ECC error occurred on one or more ID fields. Possible causes are:

- Media Defect on Drive
- Bad Drive (if errors are excessive or continuous)
- Bad controller (if errors are excessive or continuous)

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

11 Uncorrectable Data Error in the Data Field. The controller detected a data error that could not be corrected using ECC. Possible causes are:

- Media defect on drive
- Bad drive (if errors are excessive or continuous)
- Bad controller (if errors are excessive or continuous) Media defects may be overcome by deleting the defective sectors from system use or assigning an alternate track.

12 Sector Address Mark Not Found. The controller did not detect an address mark (AM) from the drive within its timing window. An address mark is a special recording pattern preceeding the ID field of a sector. The AM is only written at format time. The AM tells the controller where new sectors start. The error may occur during any data transfer or format commands. The error may mean that no address marks were detected on the track, or the target sector address mark was not detected. Possible causes are:

- Media defect on drive
- Drive has not been formatted
- Bad drive
- Bad unit cable (J2, J3)
- Bad controller

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

TABLE 4-7 CONTROLLER ERROR CODES (CONTINUED)

HEX CODE	DEFINITION
14	Target Sector Not Found. The target sector was not located within two revolutions of the disk. Possible causes are: - Media defect on drive. - Invalid format (changing sector size from 512 to 256 bytes and not reformatting the drive) - Bad drive - Bad controller
15	Seek Error. After a seek, the target disk address did not match the ID address read from the disk. Either the cylinder or head bytes did not match. Possible causes are: - Incorrect seek option specified in the command - Bad drive (Seek to incorrect cylinder) - Bad control cable (J1) - Bad controller
16 to 17	Not used
18	Correctable Data Error. The controller detected a media error while reading that was corrected by ECC. This error code informs the host software that error correction has taken place.
19	Track is Flagged Bad. The last data transfer command encountered a track that had been flagged defective using the Format Bad Track Command. Host software is responsible for insuring that deleted tracks are never accessed.
1A	Format Error. During execution of a check track command, the controller detected an unformatted track, the wrong interleave on disk, or an ID ECC error on at least one sector.
1B	Not used

TABLE 4-7 CONTROLLER ERROR CODES (CONTINUED)

HEX CODE	DEFINITION
1C	Controller Detected a Direct Access to an Alternate Track. A track that has the alternate track flag set in the ID has been directly accessed by the host, instead of coming from the defective track that that is assigned to this alternate track. Host software is responsible for insuring that the alternate track area is not directly accessed.
1D	The Designated Alternate Track is already assigned to another Defective Track, or is flagged as a Bad Track. Host software has attempted to assign an alternate track to replace a defective track, but the alternate had previously been assigned to a defective track, or has been previously formatted as a bad track. If an alternate track is no longer needed, the host software must reformat the track using the FORMAT TRACK command before attempting to reassign the track again.
1E	Assigned Alternate Track Not Found. A defective track has been assigned an alternate track, but the alternate track does not have the alternate track bit set in the ID field. This may be caused by reformatting the alternate track with the format track command without reprocessing the defective track.
1F	The Alternate and Defective Track Addresses point to the same track. Host software has attempted to assign a defective track to itself. That is not allowed in this alternate track scheme.

TABLE 4-8 TYPES 2 AND 3 ERROR CODES, COMMAND AND MISCELLANEOUS

HEX CODE	DEFINITION
20	Invalid Command: The controller has received an invalid command from the host.
21	Illegal Disk Address: The controller detected an address that is beyond the maximum range.
22	Invalid Parameter: The controller detected an invalid parameter value or an invalid combination of parameters. (Drive Initialize Command)
23 to 2F	Not Used.
30	RAM Error: The controller detected a data error during the RAM sector-buffer diagnostic. Replace controller.
31	Program Memory Checksum Error: During its internal diagnostic, the controller detected a program-memory checksum error. This is caused by a defect in the program memory chip of the controller.
32	ECC Polynomial Error: During the controller's internal diagnostic, the hardware ECC generator failed its test. Replace controller.
33 to 3F	Not used.

The following is a summary of the error codes returned as the result of the Request Sense Status command.

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

Error Code (hex)	Meaning.
00	No error detected (command completed ok).
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.
09-0F	Not used.
10	ID field read error.
11	Uncorrectable data error.
12	Address 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	Not used.
1C	Illegal (direct) access to an alternate track.
1D	Alternate track already assigned.
1E	Assigned track pointed by the bad track is not formatted as alternate.
1F	On format alternate track command, alternate and defective track addresses point to the same track.
20	Invalid command.
21	Illegal disk address.
22	Invalid Parameter.
23-2F	Not used.
30	Ram diagnostic failure.
31	Program memory checksum error.
32	ECC diagnostic failure.
33-3F	Not used.

4.5.3.5 Format Drive (Class 0, Opcode 04)

This command recalibrates the drive, then seeks to the starting address specified by the byte 1, 2 and 3 of the DCB. It times the spindle speed, divides the track into equal size sectors, and writes out address mark (AM) and header field for all sectors. The logical sector layout is specified by the interleave value contained in byte 4 of the DCB. Then it reads back the AM and header field and writes the DATA field for all sectors. The data pattern is defaulted to 6C Hex or the host can initialize the pattern by using the WRITE SECTOR BUFFER command and then set bit 5 of the Control Byte of the DCB to 1 in the FORMAT DRIVE command to tell the Controller not to change the buffer content. Note that if the format command gets a hard error while formatting a track, the format operation stops immediately and the error is reported. To continue, the host software must provide the data fields for all logical sectors following the sector in error, then continue with the format command at the beginning of the next track. Also note that the format operation always start at the first sector of a track, even though the address specified in the DCB did not point to a track boundary.

Bit 4 of the control byte (v bit) determines the format type. If the v bit is set to one, the controller leaves an "Imbedded Servo" gap at the end of the track. This gap accommodates the servo area required by higher capacity drives. This "Imbedded Servo" allows the drive to more accurately position the heads and control the spindle speed. If the disk is not an "Imbedded Servo" drive, set the v bit to zero. If the imbedded servo format is selected, the firmware times the disk rotation, subtracts off a 300 microsecond gap for the servo area, and formats the track. This firmware will work properly with drives that have a servo area that is smaller than 300 microseconds preceeding index and 40 microseconds

following the leading edge of the index pulse; and the worst-case spindle speed variation falls within 3536 +/-3.0% RPM for an imbedded servo drive, 3600 +/-3.0% RPM for a conventional drive. All drives must maintain speed tolerance of ± 1% after format.

Refer to Drive specifications to determine if the drive has the imbedded servo at index.

d = drive, 0 or 1

r = retries

p = data field pattern option

v = drive type option

s = seek option

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	0	1	0	0
Byte 1	0	0	d	High Address				
Byte 2	Middle Address							
Byte 3	Low Address							
Byte 4	0	0	0	Interleave				
Byte 5	r	0	p	v	s	s	s	s

Interleave (See Section 4.11)

1 to 31 for 256 byte sectors

1 to 16 for 512 byte sectors

Control Byte:

Bit 5 = 0 Format data pattern is defaulted to 6C.

Bit 5 = 1 Use format pattern previously written to the controller using the write buffer command (Opcode 0FH).

Bit 4 = 0 Normal disk drive.

Bit 4 = 1 "Imbedded Servo" drive.

Bit 3 - 0 Step rate.

4.5.3.6 Check Track Format (Class 0, Opcode 05)

This command will generate an interleave table from the byte 4 of the DCB, seek to the target track (without recalibrating), and read each sector from sector 0 to the last sector. While reading each sector, the Controller checks for a bad ID field and compares the current sector in the ID field with the value generated in the interleave table. If a discrepancy occurs between the sector number read and the value in the interleave table, the controller will report the error code 1A hex, format error. The Controller does not read the DATA fields in this command.

d = drive, 0 or 1

r = retries

s = seek option

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	0	1	0	1
Byte 1	0	0	d	High Address				
Byte 2	Middle Address							
Byte 3	Low Address							
Byte 4	0	0	0	Interleave				
Byte 5	r	0	0	0	s	s	s	s

Interleave: (Ref: Section 4.11)

1 to 31 for 256 byte sectors

1 to 16 for 512 byte sectors

4.5.3.7 Format Track (Class 0, Opcode 06)

This command recalibrates the drive, seeks to the target track specified in byte 1, 2 and 3 of the DCB, and writes the ID and DATA fields with the interleave value specified in byte 4 of the DCB. This command can be used to clear the defective or alternate track bits, or to reformat one track that lacks data integrity on a drive.

d = drive, 0 or 1

r = retries

p = format pattern (Ref: Bit 5 of the control byte of the FORMAT DRIVE command)

v = drive type (Ref: Bit 4 control byte of the control byte of the FORMAT DRIVE command)

s = step option

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	0	1	1	0
Byte 1	0	0	d	High Address				
Byte 2	Middle Address							
Byte 3	Low Address							
Byte 4	0	0	0	Interleave				
Byte 5	r	0	p	v	s	s	s	s

Interleave: (Ref Section 4.11)

1 to 31 for 256 byte sectors

1 to 16 for 512 byte sectors

4.5.3.8 Format Bad Track Class (Class 0, Opcode 07)

This command is the same as FORMAT TRACK command except the BAD TRACK flag is set in the ID field. DATA fields are not written. This command is used to prevent system access to defective tracks. There is an alternate way to process bad tracks. See the ASSIGN ALTERNATE TRACK Command for details.

d = drive, 0 or 1

r = retries

v = drive type

s = seek option

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	0	1	1	1
Byte 1	0	0	d	High Address				
Byte 2	Middle Address							
Byte 3	Low Address							
Byte 4	0	0	0	Interleave				
Byte 5	r	0	0	v	s	s	s	s

Interleave (Ref: Section 4.11)

1 to 31 for 256 byte sectors

1 to 16 for 512 byte sectors

4.5.3.9 Read (Class 0, Opcode 08)

This command will read 1 to 256 sectors as specified by the byte 4 of the DCB. The starting address is specified by byte 1, 2 and 3 of the DCB. The address specified by the address field is the linear sector number from the beginning of the disk. The first sector of the disk is sector zero. The Controller converts this linear address to the physical cylinder, head, and sector address for the drive. If an unrecoverable error occurs during a multiple sector transfer, the transfer will terminate at the sector where the error occurs. For example, assume the user wants to read 10 sectors starting at logical address 1000. If a correctable data error occurs at logical address 1005 and the "a" bit is set (no retries), the Controller completes the transfer of 6 sectors, including the sixth one because the data was corrected. It terminates the read operation and sets REQUEST SENSE command data to inform the host what error has occurred. To continue the operation, the host calculates the difference between sectors desired and sectors completed. In this case, 6 out of 10 are completed, therefore, the host should issue a second read command of 4 remaining sectors at starting logical address 1006. If any other error code occurred, the data is not returned to the host, so the retry logical address is one sector less, and the retry sector count one sector more than the continuation after a correctable data error. In the previous example, the restart logical address is 1005, and the transfer length is 5 sectors for any error other than a correctable data error. If the "a" option bit is set to zero, the Controller will not report correctable data errors. If the "a" option bit is set to one, correctable data errors are reported.

d = drive, 0 or 1

r = retries

a = retry option on data ECC error

s = seek option

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	1	0	0	0
Byte 1	0	0	d	High Address				
Byte 2	Middle Address							
Byte 3	Low Address							
Byte 4	Block Count							
Byte 5	r	a	0	0	s	s	s	s

4.5.3.10 Read Verify (Class 0, Opcode 09)

This command functions the same as a READ command, except that no data is passed to the host. This command may be used to insure data integrity following a write or format command. If the ECC retry bit (a) is set active (0) then one retry is attempted on data errors. If no error occurs on the retry, an error is not reported. If the error occurs on the retry, then an ECC error is reported to the host to flag as a media defect. If the retry bit (a) is set inactive (1), then all ECC errors are reported and no retrys are attempted.

d = drive, 0 or 1

r = retries

a = inhibit retry option on data ECC error if 1

s = seek option

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	1	0	0	1
Byte 1	0	0	d	High Address				
Byte 2	Middle Address							
Byte 3	Low Address							
Byte 4	Block Count							
Byte 5	r	a	0	0	s	s	s	s

4.5.3.11 Write (Class 0, Opcode 0A)

This command will write from 1 to 256 sectors as specified by the byte 4 of the DCB, starting at the address specified by the bytes 1, 2 and 3 of the DCB. The multiple sector transfer scheme works the same as the READ command. There is no read verify of data on a write command, (therefore, no ECC errors).

d = drive, 0 or 1

r = retries

s = seek option

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	1	0	1	0
Byte 1	0	d	d	High Address				
Byte 2	Middle Address							
Byte 3	Low Address							
Byte 4	Block Count							
Byte 5	r	0	0	0	s	s	s	s

4.5.3.12 Seek (Class 0, Opcode 0B)

This command initiates a seek to the track specified in the DCB. The drive must be formatted. If buffered step option is selected then the Controller will issue the seek step pulse and complete the command before the drive completes the seek. This allows overlap seek operation.

d = drive, 0 or 1

r = retries

s = seek option

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	1	0	1	1
Byte 1	0	0	d	High Address				
Byte 2	Middle Address							
Byte 3	Low Address							
Byte 4	/	/	/	/	/	/	/	/
Byte 5	r	0	0	0	s	s	s	s

4.5.3.13 Initialize Drive Characteristics (Class 0, Opcode 0C)

This command enables the user to configure the Controller to work with drives that have different capacities and characteristics. However, the characteristics of both drives are intialized at the same time. If drives of different characteristics are used, the command must be issued when switching between drives.

After the host sends the command (DCB) to the controller, it then sends an eight-byte block of data that contains the drive parameters. Some of the parameters occupy two bytes; all two-byte parameters are transferred with the most significant byte (MSB) first. The eight bytes are listed below.

C= Maximum number of cylinders (2 bytes)
H= Maximum number of heads (1 byte)
W= Reduce write current cylinder (2 bytes)
P= Increase write precompensation cylinder (2 bytes)
E= Maximum ECC data burst length (1 byte)

When the controller is powered up or reset, the default values listed below are set.

Maximum number of cylinders = 153
Maximum number of heads = 4
Starting reduced write current cylinder = 128
Starting write precompensation cylinder = 64
Maximum ECC data burst length = 11 bits

The drive size parameters passed are relative to one. For example, if the drive has 306 cylinders and 6 heads, then the value of C is 306 and the value of H is 6. For the W and P parameters, the value specifies the cylinder number at which the event occurs. For example, if the value of W is set to 200, write current will be reduced starting at cylinder 200 and higher.

The controller offers precompensation values of 0, 5, and 10 nanoseconds. The most significant bit of the most significant byte of the "p" parameter determines the compensation type. If this bit is set to one, zero nanoseconds compensation is used below the P cylinder, and 10 nanoseconds is used at and above the P cylinder. If the most significant bit is set to zero, five nanoseconds compensation is used below the P cylinder, and 10 nanoseconds is used at or above the P cylinder. Usually, oxide media drives require the 5-10 nanosecond compensation at some intermediate cylinder value. Plated media drive require no compensation over the entire cylinder range. In this case, set the most significant bit of the P parameter to one, and set the cylinder value of P to the same value as the C parameter. These are only suggested methods, consult the disk drive technical manual for precompensation details.

The parameter for the maximum ECC burst length defines the length of a burst error in the data field that the controller will correct. The burst length is defined as the number of bits from first error bit to the last error bit. For example, the controller detected a 5-bit ECC error and the erroneous data appeared as C5 (1100 0101), before correction and could appear as D4 (1101 0100) after the correction.

However, if the host has set the maximum ECC burst length at 4 bits, the controller would flag this data as uncorrectable. This is a type 1, code 1 error.

COMMAND BYTES

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	1	1	0	0
Byte 1	/	/	/	/	/	/	/	/
Byte 2	/	/	/	/	/	/	/	/
Byte 3	/	/	/	/	/	/	/	/
Byte 4	/	/	/	/	/	/	/	/
Byte 5	/	/	/	/	/	/	/	/

Parameter Bytes

Bit	7	6	5	4	3	2	1	0
Byte 0	C	C	C	C	C	C	C	C
Byte 1	C	C	C	C	C	C	C	C
Byte 2	0	0	0	0	H	H	H	H
Byte 3	W	W	W	W	W	W	W	W
Byte 4	W	W	W	W	W	W	W	W
Byte 5	P	P	P	P	P	P	P	P
Byte 6	P	P	P	P	P	P	P	P
Byte 7	0	0	0	0	E	E	E	E

NOTE: If the controller is not powered on the same DC supply as the host system, it is recommended that the drive intialize command be sent before each command accessing the disk. This will prevent the controller from operating incorrectly if the intialize data is lost due to power resets.

4.5.3.14 Read ECC Burst Error Length (Class 0, Opcode 0D)

This command is only valid following a Correctable Data Error (Error Code 18 Hex). It will transfer one byte to the host indicating the length of the error corrected. 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. For example:

Assume the drive is formatted with the default drive data pattern 6C Hex. The first 2 bytes expanded to the binary level has the pattern -- 0110 1100 0110 1100. This is the 2 byte pattern stored on the disk. Now, 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

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

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	1	1	0	1
Byte 1	/	/	/	/	/	/	/	/
Byte 2	/	/	/	/	/	/	/	/
Byte 3	/	/	/	/	/	/	/	/
Byte 4	/	/	/	/	/	/	/	/
Byte 5	/	/	/	/	/	/	/	/

4.5.3.15 Format Alternate Track (Class 0, Opcode OE)

Format Alternate Track will format the header and data fields of the "Bad Track" with the alternate track information (assigned by the host). The alternate track is formatted to identify it as an alternate. The command bytes for Format Alternate Track are:

d = drive, 0 or 1
r = retries
s = step option
v = drive type
p = data field pattern option

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	1	1	1	0
1	0	d	d	High Address				
2	Middle Address							
3	Low Address							
4	0	0	0	Interleave				
5	r	0	p	v	s	s	s	s

The logical address in the command bytes point to the "Bad Track". Sector address is ignored, defaulting to sector 0.

The interleave byte (4) is programmed the same as in the format command, and is used on the alternate track.

If Bit 4 of the Control Byte (5) is set, the format is set for an "Imbedded Servo" drive.

If Bit 5 of Control Byte (5) is set, the data in the existing sector buffer is used to fill the data field; if not set, the data field is written with hex 6C.

After issuing the command the Controller will ask for 3 data bytes. These bytes point to the host assigned alternate logical address. Again sector address is ignored.

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	High Address				
Byte 1	Middle Address							
Byte 2	Low Address							

After receiving the command and the assigned alternate, the Controller does the following:

A) Seeks to "Alternate Assigned Track" and verifies it is not already an assigned alternate track, or flagged bad track.
   - If the track has already been assigned as an alternate or is flagged "BAD", then error code ID Hex is given, and the command is aborted. This usually implies that the host is attempting to assign two bad tracks to the same alternate track.

B) Formats the track as an assigned alternate track.

C) Seeks to the "Bad Track" and formats the header as a defective track that has an alternate track assigned.

Note: Data fields on both the bad track and alternate track are rewritten.

4.5.3.16 Write Sector Buffer (Class 0, Opcode 0F)

This command is usually used in 2 ways:

A) Write a test pattern to the Controller and read it back to verify the Controller's buffer memory is functioning.

B) If the user wants another data field pattern for format such as E5 Hex for CP/M compatibility, the user can initialize the data pattern by writing the data pattern to the Controller. Then, before issuing the FORMAT DRIVE or FORMAT TRACK command, set bit 5 of the byte 5 of the DCB high to tell the Controller to use the data pattern in the buffer instead of the default format data pattern.

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	1	1	1	1
1	/	/	/	/	/	/	/	/
2	/	/	/	/	/	/	/	/
3	/	/	/	/	/	/	/	/
4	/	/	/	/	/	/	/	/
5	/	/	/	/	/	/	/	/

4.5.3.17 Read Sector Buffer (Class 0, Opcode 10)

This command is used in 2 ways:

A) Read the contents of the data buffer after writing it using WRITE SECTOR BUFFER command to test the RAM.

B) If an uncorrectable data error occurs, the Controller will not send data to the host, but reports the error immediately. If the host wants the corrupt data, the host issues this command to retrieve it.

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	1	0	0	0	0
1	/	/	/	/	/	/	/	/
2	/	/	/	/	/	/	/	/
3	/	/	/	/	/	/	/	/
4	/	/	/	/	/	/	/	/
5	/	/	/	/	/	/	/	/

4.5.3.18 RAM Diagnostic (Class 7, Opcode 00)

This command does a walking 1 and walking 0 pattern test of its internal RAM buffer.

Bit	7	6	5	4	3	2	1	0
Byte 0	1	1	1	0	0	0	0	0
1	/	/	/	/	/	/	/	/
2	/	/	/	/	/	/	/	/
3	/	/	/	/	/	/	/	/
4	/	/	/	/	/	/	/	/
5	/	/	/	/	/	/	/	/

4.5.3.19 Reserved (Class 7, Opcode 01)

This opcode is not used.

4.5.3.20 Reserved (Class 7, Opcode 02)

This opcode is not used.

4.5.3.21 Drive Diagnostic (Class 7, Opcode 03)

This command tests both the drive and the drive-to-controller interface. The controller sends recalibrate and seek commands to the selected drive and verifies the ID field of sector 0 on all tracks on the disk. The controller does not perform any write operations during this command; it is assumed that the disk has been previously formatted.

d = drive, 0 or 1

r = retries

s = step option

Bit	7	6	5	4	3	2	1	0
Byte 0	1	1	1	0	0	0	1	1
1	0	0	d	/	/	/	/	/
2	/	/	/	/	/	/	/	/
3	/	/	/	/	/	/	/	/
4	/	/	/	/	/	/	/	/
5	r	0	0	0	s	s	s	s

4.5.3.22 Controller Internal Diagnostics (Class 7, Opcode 04)

This command checksums the EPROM 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 do not compare, then a CHECKSUM ERROR (Error Code 31 Hex) is returned. The ECC circuitry is tested by introducing an artificial error to the data and check that the ECC circuitry detects the error. It also passes a good pattern and sees if the ECC circuitry detects no ECC error. The Controller does not access the disk.

Bit	7	6	5	4	3	2	1	0
Byte 0	1	1	1	0	0	1	0	0
1	/	/	/	/	/	/	/	/
2	/	/	/	/	/	/	/	/
3	/	/	/	/	/	/	/	/
4	/	/	/	/	/	/	/	/
5	/	/	/	/	/	/	/	/

4.5.3.23 Read Long (Class 7, Opcode 05)

This command is used to test the ECC circuitry. When the host issues a write command to the Controller (assume 256 bytes/sector), the Controller writes to the disk the 256 bytes sent by the host and appends the four bytes generated by the ECC hardware. During a normal read command, the Controller reads the 256 data bytes plus the 4 ECC bytes into the buffer. But the Controller sends only the 256 data bytes to the host. The 4 ECC bytes are used to determine if an ECC data error occurred. The only difference between READ LONG and READ is the Controller appends the 4 ECC bytes to the data transfer, making the sector transfer 260 bytes long. The method to test the ECC circuitry is as follows:

A) Use the normal READ command to find a sector that does not have any data errors.

B) Use READ LONG to read that sector plus ECC into the host.

C) Modify the data pattern in a known way.

D) Use WRITE LONG to write the pattern to the same sector.

E) Use the READ command to read the same sector again.

F) If the pattern change is less than or equal to 11 bits in length, the Controller flags it as a correctable data error. If the change is greater than 11 bits in length, the Controller will flag it as an uncorrectable data error.

G) Use a WRITE command to restore the sector for system use.

d = drive, 0 or 1

r = retries

s = step option

Bit	7	6	5	4	3	2	1	0
Byte 0	1	1	1	0	0	1	0	1
1	0	0	d	HIGH ADDRESS				
2	MIDDLE ADDRESS							
3	LOW ADDRESS							
4	0	BLOCK COUNT						
5	r	0	0	0	s	s	s	s

4.5.3.24 Write Long (Class 7, Opcode 06)

When this command is used, the host supplies the 4 bytes of ECC information following the data bytes. This command is used to test the Controller's ECC circuitry only. For detailed description of the test, see the READ LONG command above.

d = drive, 0 or 1

r = retries

s = step option

Bit	7	6	5	4	3	2	1	0
Byte 0	1	1	1	0	0	1	1	0
1	0	0	d	HIGH ADDRESS				
2	MIDDLE ADDRESS							
3	LOW ADDRESS							
4	BLOCK COUNT							
5	r	0	0	0	s	s	s	s

4.5.3.25 Retry Statistics (Class 7, Opcode 07)

This command allows the user to gather retry and error statistics on a disk drive over a period of time. This data can be used to indicate the overall reliability of the drive and may provide an early warning of trouble before a catastrophic failure occurs. After the command is issued, the controller returns eight bytes of data, followed by status.

d = drive, 0 or 1

Bit	7	6	5	4	3	2	1	0
Byte 0	1	1	1	0	0	1	1	1
1	0	0	d	/	/	/	/	/
2	/	/	/	/	/	/	/	/
3	/	/	/	/	/	/	/	/
4	/	/	/	/	/	/	/	/
5	/	/	/	/	/	/	/	/

The controller returns four 2 byte parameters, most significant byte first.

Bit	7	6	5	4	3	2	1	0
Byte 0	N	N	N	N	N	N	N	N
1	N	N	N	N	N	N	N	N
2	R	R	R	R	R	R	R	R
3	R	R	R	R	R	R	R	R
4	S	S	S	S	S	S	S	S
5	S	S	S	S	S	S	S	S
6	C	C	C	C	C	C	C	C
7	C	C	C	C	C	C	C	C

The first two bytes (N) are the number of non-recoverable media or drive errors that occurred. A non-recoverable error is an error that cannot be overcome even after all retries are exhausted, except correctable data errors. The second two bytes (R) are the number of recoverable media or drive errors that occurred. A recoverable error is an error that occurred at least once, but did not occur again after one or more retries, except correctable data errors. The third parameter is the number of "soft" ECC errors that occurred. A "soft" ECC error is an ECC error that occurs on the first read attempt, but no error

occurs on the retry read. The error is not due to a media defect. The last parameter is the number of errors that were correctable using error correction (ECC). On a normal read, the controller will reread the failing sector once before attempting error correction. If no error occurs on the retry attempt, the "S" count is incremented. If the second read still fails, but the error is corrected using ECC, the "C" parameter is incremented. If the ECC error is uncorrectable, further retries are done if the "r" bit in the control byte of the read command is reset. If the error cannot be recovered, the "N" parameter is incremented. If the error is recoverable, the "R" parameter is incremented.

All parameters are set to zero after this command is issued and when the controller is reset. If a parameter reaches the maximum count prior to executing this command, the count remains at the maximum value even though more events occur. The maximum count value is all bits set to one (65, 535).

4.6 SECTOR FORMAT

Figure 4-6 and 4-7 shows the format of the sector and the names of the fields of the information traveling over the Controller-drive interface. Table 4-9 and 4-10 lists the fields and a description of each field.

TABLE 4-9 SECTOR FIELD DESCRIPTION FOR 256 BYTES PER SECTOR

FIELD	BYTES	FIELD DESCRIPTION
GAP1	9	Zero Byte Gap
AM	4	Address Mark
SYNC1	1	ID Sync Byte
GAP2	2	ID Zero Byte Gap
COM	1	ID Compare Byte
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 Bytes
GAP3	16	Zero Byte Gap
SYNC2	1	Data Field Sync Byte
GAP4	2	Data Field Zero Byte Gap
DATA	256	Data Field
ECC2	4	Data Field ECC Bytes
GAP5	14	Inter-record Zero Gap

TABLE 4-10 SECTOR FIELD DESCRIPTION FOR 512 BYTES PER SECTOR

FIELD	BYTES	FIELD DESCRIPTION
GAP1	9	Zero Byte Gap
AM	4	Address Mark
GAP1	9	Zero Byte Gap
SYNC1	1	ID Sync Byte
GAP2	2	ID Zero Byte Gap
COM	1	ID Compare Byte
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 Bytes
GAP3	16	Zero Byte Gap
SYNC2	1	Data Field Sync Byte
GAP4	2	Data Field Zero Byte Gap
DATA	512	Data Field
ECC2	4	Data Field ECC Bytes
GAP5	34	Inter-record Zero Gap

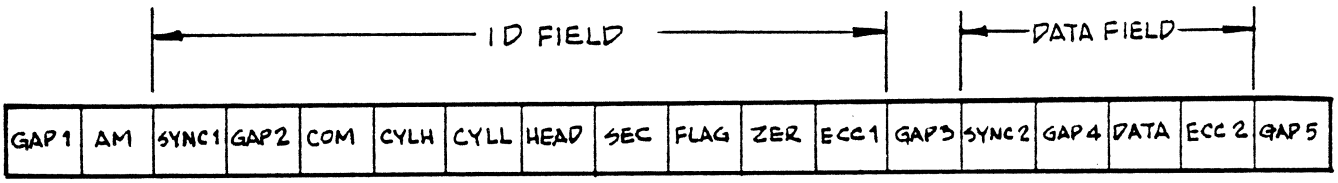


FIGURE 4-7 SECTOR FORMAT
FOR 256 BYTE SECTOR

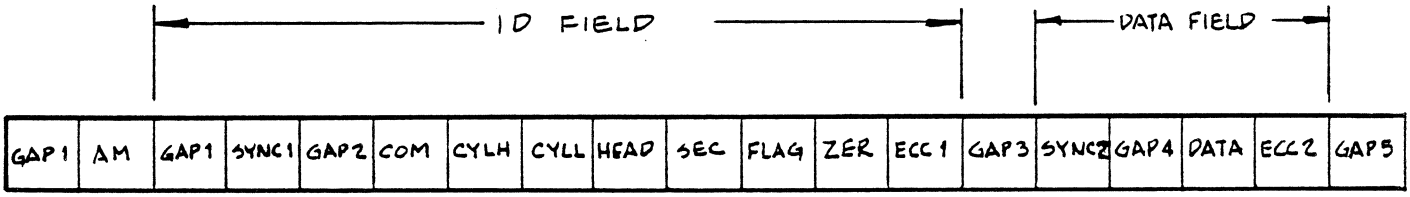


FIGURE 4-8 SECTOR FORMAT
FOR 512 BYTE SECTOR

4.7 EXECUTION OF DIAGNOSTICS

Since all of the diagnostics are not executed by the Controller on power-up, it is suggested that they be envoked by the host in the following order:

1) Controller internal diagnostics (Command code E4). This diagnostic tests all the logical and decision making capability of the controller as well as the program memory checksum and the error detection and correction circuits (ECC). Execution of this diagnostic ensures that the controller can communicate with the host.

2) Ram Diagnostic (Command code E0) should be executed next. This command verifies that the sector buffer is operational by writing, reading and verifying various data patterns to and from all locations.

3) If the parameters of the connected drives are different than the default parameters, see Section 4.5.15, the new configuration must be sent to the controller using the Initialize Drive Characteristics command (Command code 0C) before the Drive Diagnostic is executed.

4) Before the Drive Diagnostic is executed, the host program should continuously issue a Test Drive Ready command to the controller (Command code 00) with the appropriate time-out until the drive becomes ready.

5) Drive Diagnostic (Command code E3). This diagnostic issues a Recalibrate to the disk drive and then steps through all tracks verifying the ECC on the identifier fields of the first sector of each track. If this diagnostic passes, it implies that the disk has been formatted and that the first ID field of each track is good.

4.8 ERROR CORRECTION PHILOSOPHY

Since the typical error correction time of the S1410A controller is approximately 50 milliseconds and therefore greater than the time for one revolution of the disk, the sector in error is optionally re-read (if bit 6 is not set in byte 5 of the read command DCB) on the next revolution during a Read command. In most cases, the error will be soft and will not reappear on the re-read. This initial re-read of the failing sector is over and above the retry count specified in the DCB (bit 7, byte 5).

The retry count on errors is preset to 4 by the controller each time a sector has been read successfully. On a multiple sector transfer if an uncorrectable error was detected but subsequently found to be correctable on a retry, the retry count is reset to 4 before the next sector is read from the disk.

4.9 ALTERNATE TRACK ASSIGNMENT AND HANDLING

The assignment of alternate tracks or lockout of defective tracks must be done by the host computer. One procedure for initializing a disk and assigning alternate tracks is as follows.

1) The entire disk drive is formatted by issuing a Format Disk command (Command Code 04) starting at logical address zero.

   If an error occurs during formatting, a Request Sense command is issued to retrieve the error code and disk address. The defective track address is determined from the last three request sense data bytes. Another Format Disk command is issued starting at the track following the defective track. If other errors occur, the process is repeated until the disk is formatted. The host must maintain the list of defective tracks. All defective tracks should be formatted using the format bad track command to make verifying easier.

2) Now, the entire disk is read to check for uncorrectable data errors. Any track that contains an uncorrectable error is added to the defective track list by the host. If all defective tracks have been formatted bad, the read command will terminate with an "Access to Defective Track" Error (code 19) when a read is attempted from that track. Software then knows to advance the disk address to the next track and continue verifying. It is a good idea to reduce the ECC correction length during verify. This ensures that a correctable data error does not become uncorrectable later because the number of defective bits increased.

3) Steps 1 and 2 should be repeated using different data patterns for each pass. This is done easily by issuing a Write Buffer command with the data pattern followed by a Format command with bit 5 of the control byte (byte 5) of the DCB set. Three to five passes should be sufficient.

4) Next, pick an alternate track area on the disk. The alternate track area usually resides at the highest numbered disk tracks. The area should be of sufficient size to allow one alternate track per 50 to 100 tracks on the drive. Each defective track in the list created in steps 2 and 3 is reassigned to the alternate track area by using the Assign Alternate Track command. If any tracks in the alternate area are defective, skip over them and use the next available track.

4.10 OVERLAPPING SEEKS WITH BUFFER STEP DRIVES

For drives employing buffered seeks, seek commands can be overlapped. After the controller issues a seek to the drive, it returns with a completion status, not waiting for the drive to complete the seek. If the return status shows no error, then the seek was issued correctly. If there is an error, then 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, then the controller will wait, with Busy active, for the seek to complete before executing the new command (Except Test Drive Ready Command).

The Test Drive Ready command can be used with overlapped seeks to determine when a drive has completed seeking before issuing the next command. If the drive is still seeking, the status byte at the end of the command will indicate an error, and the sense status will indicate "drive still seeking" (type 0 error, code 8). A sequence of Test Drive Ready commands can thus be used to determine when the drive is ready for the next command.

4.11 SECTOR INTERLEAVING

Variable sector interleaving is supported by the S1410A. When any format command is issued, any interleave value up to the number of sectors-per-track minus one may be passed in the Device Control Block (DCB byte 4). The interleave factor may be adjusted for maximum system performance. Interleaving allows logical contiguous sectors of data on a given track to be mapped onto non-adjacent physical sectors. An interleave factor of five, for instance, means that every fifth physical sector is transferred as the next contiguous logical sector of data. If the operation is a read and the interleave factor is five then a sector of data is read into the sector buffer first and during the time that the heads are passing over the next three 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 three sector times available, then the S1410A has to 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 speed is noticed.

In order to take full advantage of the interleaving feature of the S1410A, 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.

The interleave value can be set to improve system throughput based on overhead time of the host software, overhead time in the disk driver, and overhead time for the S1410A to process a command. If the host system is capable of multisector transfers, system throughput can be optimized by setting the interleave value such that the next logical sector comes under the heads just as the S1410A completes the data transfer of the previous sector. If the host is capable of passing a sector of data at DMA speed (one millisecond for a 256 byte sector), then the interleave value should be set to four to optimize multisector transfers. This is the minimum value for continuous sector transfers. If a sector data transfer takes between one and two milliseconds, set the interleave value to five. The best method is to experimentally determine the best interleave value for your system using a representative benchmark.

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

PHYSICAL	LOGICAL	LOGICAL	PHYSICAL
00	00	00	00
01	13	01	05
02	26	02	10
03	07	03	15
04	20	04	20
05	01	05	25
06	14	06	30
07	27	07	03
08	08	08	08
09	21	09	13
10	02	10	18
11	15	11	23
12	28	12	28
13	09	13	01
14	22	14	06
15	03	15	11
16	16	16	16
17	29	17	21
18	10	18	26
19	23	19	31
20	04	20	04
21	17	21	09
22	30	22	14
23	11	23	19
24	24	24	24
25	05	25	29
26	18	26	02
27	31	27	07
28	12	28	12
29	25	29	17
30	06	30	22
31	19	31	27

FIGURE 4-9 TRACK FORMAT EXAMPLE OF 32
SECTORS-PER-TRACK WITH AN INTERLEAVE FACTOR OF 5

APPENDIX A

SAMPLE HARDWARE INTERFACE AND PROGRAMMING EXAMPLES FOR A Z-80 BASED MICROPROCESSOR TO INTERFACE TO THE XEBEC S1410A CONTROLLER

SASI read port Function.

0	Read data (from controller to CPU)
1	Read hardware status lines (from controller to CPU)
2	Not used
3	Not used

SASI write port Function.

0	Write data (from CPU to controller)
1	Controller reset (and interface reset)
2	Generate controller select pulse
3	Not used

WRITE TRANSPARENT LATCH (DATA FROM HOST).

The data to be sent to the controller is passed through this register when the enable line is high and is latched up when the enable goes low. The latch enable is generated by a programmed I/O write to SASI write port 0 from the host. The positive going (trailing) edge of the negative true I/O write signal from the host also sets the handshake acknowledge flip flop which asserts the acknowledge signal (ACK-) on the SASI bus indicating to the controller that data is available from the host.

WRITE DATA BUS DRIVER (TO THE CONTROLLER).

This is a tri-state bus driver which is enabled to drive the SASI bus under command of the controller when it brings the Input-/Ouput signal on the SASI bus high.

READ DATA BUS RECEIVER (FROM THE CONTROLLER).

Data or Status from the controller is driven onto the host data bus by a programmed I/O read (SASI read port 0) from the host.

SASI HARDWARE STATUS LINE RECEIVER (FROM THE CONTROLLER).

The hardware status is driven onto the host data bus when an I/O read is issued by the host program to SASI read port 1. It should be noted that the status signals from the SASI bus are inverted before they are sent to the host.

The bits in the hardware status as seen by the host program are defined as follows:*

Bit Usage

7 Not used (MSB).

6 Not used.

5 Not used.

4 Input/Output- signal from the controller.
1 = input from controller to host, 0 = output from host to controller.

3 Command/Data- signal from the controller.
1 = Command info on SASI bus, 0 = data on SASI bus.

2 Message (MSG-) signal from the controller.
1 = second completion status byte.

1 BUSY- signal from the controller.
1 = controller busy.

0 Request signal from the controller (REQ-).
1 = transfer request.

***NOTE:** The polarities of these signals do not match those of the S1410A because in this sample circuit they are inverted by the SASI status interface chip.

HANDSHAKE ACKNOWLEDGE FLIP FLOP.

This flip flop is used for all data, command or status handshakes between the host and the controller under programmed I/O control. As long as the controller is not selected, this flip flop is kept directly cleared by the inactive REQ- line which in turn keeps the acknowledge line (ACK-) inactive to the controller.

After the controller is selected and ready to accept a command from the host, it asserts the REQ- line signaling the host to write a command byte into the transparent latch. When the host writes into the latch, the handshake acknowledge flip flop is set signaling the controller that the data is available. When the controller has read the command byte from the SASI bus, it deactivated the REQ- line clearing the handshake acknowledge flip flop, which in turn deactivated the acknowledge line (ACK-).

When the ACK- signal is deasserted, the REQ- signal is again activated by the controller starting the handshake sequence for the next byte.

Z-80 PROCESSOR INTERFACE TO THE XEBEC S1410 CONTROLLER

Z-80 SIGNAL DEFINITIONS

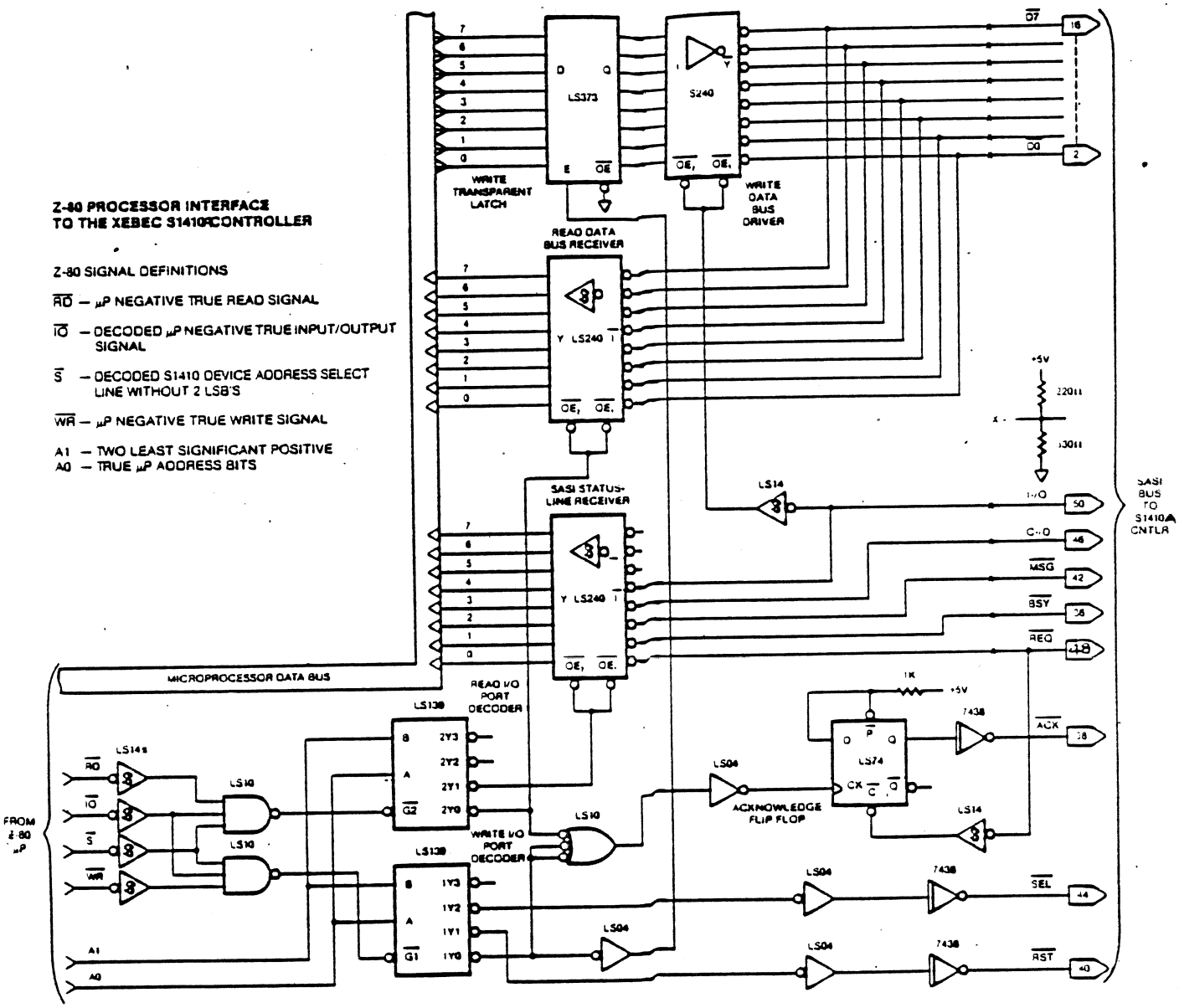
$\overline{RD}$ — μP NEGATIVE TRUE READ SIGNAL

$\overline{IO}$ — DECODED μP NEGATIVE TRUE INPUT/OUTPUT SIGNAL

$\overline{S}$ — DECODED S1410 DEVICE ADDRESS SELECT LINE WITHOUT 2 LSB'S

$\overline{WR}$ — μP NEGATIVE TRUE WRITE SIGNAL

A1 — TWO LEAST SIGNIFICANT POSITIVE
A0 — TRUE μP ADDRESS BITS



Z80 PROGRAMMING EXAMPLE
TO INTERFACE TO THE XEBEC S1410A

THIS IS A STAND ALONE PROGRAM WRITTEN IN Z80 ASSEMBLY LANGUAGE WHICH SENDS COMMANDS TO THE XEBEC S1410A CONTROLLER AND 5¼" HARD DISK IN THE FOLLOWING ORDER:

1) RESET THE CONTROLLER
2) RECALIBRATE THE DRIVE
3) FORMAT THE DRIVE
4) WRITE ONE SECTOR FROM WRITEBUF
5) READ THE SAME SECTOR INTO READBUF

IF THE PROGRAM COMPLETES PROPERLY, THE READ BUFFER (READBUF) AND THE WRITE BUFFER (WRITEBUF) SHOULD BE THE SAME. NORMAL COMPLETION IS AT LABEL TEST6.

```
LABEL       OPCD    OPERAND     COMMENT

OUTPUT AND INPUT PORTS

WPORT0      EQU     000H        SASI WRITE PORT 0 - WRITE DATA
WPORT1      EQU     001H        SASI WRITE PORT 1 - SOFTWARE RESET
WPORT2      EQU     002H        SASI WRITE PORT 2 - CNTLR SELECT
WPORT3      EQU     003H        SASI WRITE PORT 3 - NOT USED
RPORT0      EQU     000H        SASI READ PORT 0 - READ DATA
RPORT1      EQU     001H        SASI READ PORT 1 - READ STATUS
RPORT2      EQU     002H        SASI READ PORT 2 - NOT USED
RPORT3      EQU     003H        SASI READ PORT 3 - NOT USED

VARIOUS EQUATES

REQBIT      EQU     000H        REQUEST LINE BIT POSITION
REQMASK     EQU     001H        REQUEST MASK FOR BIT TEST
BUSYBIT     EQU     001H        BUSY LINE BIT POSTION
BUSYMASK    EQU     002H        BUSY MASK FOR BIT TEST
MSGBIT      EQU     002H        MESSAGE LINE BIT POSITION
MSGMASK     EQU     004H        MESSAGE MASK FOR BIT TEST
CDBIT       EQU     003H        COMMAND/DATA BIT POSITION
CDMASK      EQU     008H        COMMAND/DATA BIT POSITION TEST
IOBIT       EQU     004H        INPUT/OUTPUT BIT POSITION
IOMASK      EQU     010H        INPUT/OUTPUT BIT MASK
```

```
LABEL       OPCD    OPERAND         COMMENT
```

CONTROLLER COMMAND EQUATES

```
DRVREADY    EQU     000H            TEST DRIVE READY COMMAND
FORMAT      EQU     004H            FORMAT COMMAND CODE
READ        EQU     008H            READ COMMAND CODE
WRITE       EQU     00AH            WRITE COMMAND CODE
SENSE       EQU     003H            STATUS SENSE COMMAND CODE
INITL       EQU     00CH            INITIALIZE DISK SIZE COMMAND
SEEK        EQU     00BH            SEEK COMMAND CODE
RECAL       EQU     001H            RECALIBRATE COMMAND CODE
RAMDIAG     EQU     0E0H            RAM DIAGNOSTIC COMMAND CODE
ERROR       EQU     002H            TEST FOR AN ERROR
WRITEBUF    DS      256             WRITE BUFFER
READBUF     DS      256             READ BUFFER
STACK       DS      020H            CALL STACK
STACKTOP    EQU     $               TOP OF STACK
TASK        DB      0,0,0,0,1,80,F,0,0,0,0
```

RESET

FIRST RESET THE CONTROLLER

```
            OUT     (WPORT1),A      SEND OUT A RESET PULSE
```

TEST FOR DRIVE READY

TEST TO SEE THAT THE DRIVE IS UP AND READY AFTER CONTROLLER HAS BEEN SELECTED.

```
TEST1       EQU     $
            LD      SP, STACKTOP    SET THE STACK POINTER
            CALL    SELCNTLR        SELECT THE CONTROLLER
            LD      A,DRVREADY      DRIVE READY COMMAND
            CALL    TASKOUT         SEND OUT THE COMMAND
            CALL    GETSTAT         GET THE COMPLETION STATUS
            JP      Z,TEST2         TEST FOR PROPER COMPLETION
            RST     038H            ERROR COMPLETION
```

RECALIBRATE

SEND OUT A RECALIBRATE TO THE CONTROLLER

```
TEST2       EQU     $
            CALL    SELCNTLR        SELECT THE CONTROLLER
            LD      A,RECAL         RECALIBRATE COMMAND CODE
            CALL    TASKOUT         SEND COMMAND TO CNTLR
            CALL    GETSTAT         GET COMPLETION STATUS
            JP      Z,TEST3         TEST FOR ERROR COMPLETION
            RST     038H            ERROR ON COMPLETION
```

LABEL	OPCD	OPERAND	COMMENT

FORMAT

SEND A FORMAT COMMAND TO THE CONTROLLER

```
TEST3   EQU     $
        CALL    SELCNTLR        SELECT THE CONTROLLER
        LD      A,FORMAT        FORMAT COMMAND CODE
        CALL    TASKOUT         SEND COMMAND TO CONTROLLER
        CALL    GETSTAT         GET COMPLETION STATUS
        JP      Z,TEST4         TEST COMPLETION STATUS
        RST     038H            NON ZERO IS AN ERROR
```

WRITE

WRITE OUT A SECTOR TO THE DISK

```
TEST4   EQU     $
        CALL    SELCNTLR        SELECT THE CONTROLLER
        LD      A,WRITE         WRITE COMMAND CODE
        CALL    TASKOUT         SEND TASK TO CXNTROLLER
        LD      HL, WRITEBUF    POINT TO THE WRITE BUFFER
WRITE1  EQU     $
        CALL    REQWAIT         WAIT FOR CNTLR REQUEST

        IN      A,(RPORT1)      GET CNTLR STATUS LINES
        AND     CDMASK          TEST FORM COMMAND MODE
        JP      NZ,WRITE2       IF CMND, TRANSFER IS DONE
        LD      A,(HL)          GET A DATA BYTE
        OUT     (WPORT0),A      SEND DATA TO CONTROLLER
        INC     HL              BUMP THE BUFFER POINTER
        JP      WRITE1          MORE TO GO, LOOP

WRITE2  EQU     $
        CALL    GETSTAT         GET TRANSFER STATUS
        JP      Z,TEST5         TEST COMPLETION STATUS
        RST     038H            IF STATUS # 0, ERROR
```

READ

READ A SECTOR FROM THE DISK

```
TEST5   EQU     $
        CALL    SELCNTLR        SELECT THE CONTROLLER
        LD      A,READ          READ COMMAND CODE
        CALL    TASKOUT         SEND COMMAND TO CONTROLLER
        LD      HL,READBUF      POINT TO READ BUFFER
READ1   EQU     $
        CALL    REQWAIT         WAIT FOR REQUEST FROM CNTLR
```

```
LABEL     OPCD    OPERAND       COMMENT

          IN      A,(RPORT1)    GET CNTLR STATUS LINES
          AND     CDMASK        TEST FOR CMND MODE FROM CNTLR
          JP      NZ,READ2      IF ON, END OF TRANSFER
          IN      A,(RPORT0)    READ IN THE DISK DATA
          LD      (HL),A        SAVE IT IN THE BUFFER
          INC     HL            BUMP THE BUFFER POINTER
          JP      READ1         LOOP UNTIL 256 TRANSFERRED

READ2     EQU     $
          CALL    GETSTAT       GET COMPLETION STATUS
          JP      Z,TEST6       CONTINUE IF NO ERROR
          RST     038H          ERROR, STOP
TEST6     EQU     $
          RST     038H
```

SELCNTLR

THIS SUBROUTINE SELECTS THE DEFAULT CONTROLLER

```
SELCNTLR  EQU     $
          IN      A,(RPORT1)    READ STATUS PORT
          AND     BUSYMASK      MASK BUSY BIT
          JP      NZ,SELCNTLR   JUMP, IF BUSY
          LD      A,1           CNTLR DEFAULT SELECT CODE
          OUT     (WPORT0),A    SEND IT TO TRANSPARENT LATCH
          OUT     (WPORT2),A    GENERATE A SELECT STROBE
SEL1      EQU     $
          IN      A,(RPORT1)    GET CNTLR RESPONSE
          AND     BUSYMASK      ISOLATE THE BUSY MASK
          JP      Z,SEL1        WAIT FOR CNTLR BUSY
          RET                   BUSY HAS ARRIVED, EXIT
```

TASKOUT

THIS SUBROUTINE SENDS OUT THE COMMAND CONTAINED IN A REGISTER TO THE DISK CONTROLLER.

```
TASKOUT   EQU     $
          LD      HL, TASK      POINT TO TASK CONTROL BLOCK
          LD      (HL),A        SAVE THE COMMAND
          LD      B,6           SET UP A BYTE COUNTER
          CALL    REQWAIT       WAIT FOR CONTROLLER REQUEST
```

```
LABEL     OPCD     OPERAND      COMMENT

TASK1     EQU      $
          LD       A,(HL)       GET A COMMAND BYTE
          OUT      (WPORT0),A   SEND IT TO THE CONTROLLER
          INC      HL           BUMP THE TASK POINTER
          DEC      B            DECREMENT THE BYTE COUNT
          JP       NZ,TASK1     WAIT UNTIL ALL ARE OUTPUT
          RET
```

GETSTAT

THIS SUBROUNTINE RETRIEVES THE STATUS BYTE AND THE NULL BYTE FROM THE DISK CONTROLLER AT THE END OF A COMMAND. THE A REGISTER IS RETURNED WITH A NON-ZERO VALUE IF ERROR HAS BEEN DETECTED.

```
GETSTAT   EQU      $
          CALL     REQWAIT      WAIT FOR REQUEST
          IN       A,(RPORT0)   READ IN THE STATUS BYTE
          LD       D,A          SAVE STATUS TEMPORAIRILY
          CALL     REQWAIT      WAIT FOR SECOND BYTE
          IN       A,(RPORT0)   GET THE NULL BYTE
          LD       A,D          RESTORE STATUS TO A
          AND      ERROR        ISOLATE THE ERROR BIT
          RET
```

REQWAIT

THIS SUBROUTINE WAITS FOR THE REQUEST LINE TO BECOME ACTIVE FROM THE DISK CONTROLLER.

```
REQWAIT   EQU      $
          IN       A,(RPORT1)   GET CNTLR STATUS BITS
          AND      REQMASK      ISOLATE THE CONTROLLER
          JP       Z,REQWAIT    REQUEST AND WAIT FOR IT
          RET
          END
```

APPENDIX B
FIRMWARE OPTIONS

The standard firmware set (P/N 104788) was previously described in this manual. In addition, two other firmware sets have been developed. In the descriptions below, only the differences between the standard set and the firmware being described are discussed. All the information in the main manual applies to these firmware sets except as noted below.

B.1 PERFORMANCE OPTION

The performance option firmware (P/N 104792) is desinged for higher performance systems that allow the controller to operate at a sector interleave of three. To achieve this performance, the sector format on the disk was altered. This means that the S1410A with the standard firmware and the S1410 controllers cannot read a disk that has been formatted with this firmware, nor can this firmware read a disk formtted with the S1410 or S1410A standard firmware. The number of sectors per track has been reduced from 32 to 30 when using the 256 byte sector size. The number of sectors per track remains at 17 when using the 512 byte sector size.

B.2 ENHANCED OPTION

The enhanced firmware option (P/N 104793) gives the user a simpler and more flexable disk drive configuration, including support of some cartridge drives.

Cylinder zero of the disk drive is reserved for disk drive configuration parameter storage and enhanced diagnostic capability. The controller bumps the cylinder address by one after the linear disk address passed in the control DCB is converted to the physical disk address. This address change causes the user to skip over cylinder zero, reserving it for the controller's internal use. This change is transparent to the user, except that the disk drive appears one cylinder smaller than the number of cylinders specified in the initialization data.

The command set has been altered to make the controller easier to use. The Initialize Drive Characteristics command (command code 0C hex) has been disabled. The controller will still accept the command and the eight bytes of configuration data, but the configuration data is ignored. This command has been replaced in function by the Initialize Format command. The Format Track command has been

changed to a Format Tracks command. Other commands have been added as described below.

The drive options specified in the control byte of the DCB have been moved to the Initialize Format command. This includes the drive step option and the imbedded servo bit used in the format commands. These fields in the control byte should be set to zero when using the enhanced firmware to allow for future compatibility. In addition, the sector size is set using the Initialize Format command. The sector size jumper on the S1410A controller board is not used.

The 512 byte per sector format is the same format used in the Performance Option above. This format allows the controller to operate at an interleave of three, but is not format compatible with the S1410A standard firmware or the S1410. The 256 byte per sector format is the same format as the S1410A standard firmware. This format is not compatible with the S1410 256 byte per sector format.

B.2.1 Format Tracks (Class 0, Opcode 06)

This command formats the number of tracks specified in the two data bytes passed following the DCB. The format operation starts the first sector of the track specified by the disk address bytes in the DCB. The controller recalibrates the drive, seeks to the starting track, and begins the format operation. Formatting continues on a track by track basis until the track count is exhausted or the end of the disk is reached. If the track count exceeds the disk capacity, the controller returns an illegal address error after formatting the last track on the drive. The data field is filled with the default data pattern 6C Hex on the hard disk. If bit 5 of byte 5 is set to one, the data field will be taken as is from the buffer for the hard disk. If the T bit in the initialize format command is set to one, the hard disk format will leave a 300 microsecond gap before index to allow for the imbedded servo area if the disk requires it. If the T bit is zero, the entire track is used.

If the two track count data bytes contain zero, the initialization data will be written on the reserved cylinder (Cylinder 0). No tracks are formatted. This allows changing the initialization parameters for the drive without

reformatting the drive. Issue the Initialize Format Command, followed by the Format Tracks Command with track count set to zero to permanently update the disk drive parameters.

d = drive, 0 or 1

r = retries

b = use data already in the buffer

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	0	1	1	0
1	0	0	d	HIGH ADDRESS				
2	MIDDLE ADDRESS							
3	LOW ADDRESS							
4	BLOCK COUNT							
5	r	0	b	0	0	0	0	0

The two byte track count is transferred with the most significant byte first.

B.2.2 Initialize Format (Class 0, Opcode 11)

This command allows the user to set up the Controller for various disk drives of different configuration. The host specifies all the necessary parameters to enable the Controller to control the drive. This command must be issued prior to the drive format command. After the hard disk drive is formatted, the Controller will store these parameters on the maintenance cylinder (cylinder 0) of the drive. Following subsequent Controller resets, the Controller will fetch initialization data from the maintenance cylinder of the drive. This means that the host software does not issue this command during normal use of the system, but only under a special format utility program.

The Controller does not have any default initialization parameters at reset time. If the Controller cannot read the initialization parameters from the drive, this command must be issued before any command that accesses the drive. Failure to adhere to this rule will result in a "Controller not initialized" error (Error Code 0A Hex).

d = drive, 0 or 1

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	1	0	0	0	1
1	0	0	d	/	/	/	/	/
2	/	/	/	/	/	/	/	/
3	/	/	/	/	/	/	/	/
4	/	/	/	/	/	/	/	/
5	/	/	/	/	/	/	/	/

After sending the command (DCB) to the Controller, the host must follow with a ten byte data block. This block provides all the initialization parameters to the Controller. The hard disk configuration block is:

Bit	7	6	5	4	3	2	1	0
Byte 0	C	C	C	C	C	C	C	C
1	C	C	C	C	C	C	C	C
2	0	0	0	0	H	H	H	H
3	S	S	S	S	R	R	R	T
4	0	0	0	0	0	0	D	D
5	W	W	W	W	W	W	W	W
6	W	W	W	W	W	W	W	W
7	P	P	P	P	P	P	P	P
8	P	P	P	P	P	P	P	P
9	0	0	0	0	E	E	E	E

Where:

C = Number of Cylinders (Most Significant byte first)

H = Number of Heads

S = Step Option

DESCRIPTION	BITS				
	7	6	5	4	
Default 3 msec step rate	0	0	0	0	
Reserved	0	0	0	1	
Reserved	0	0	1	0	
Reserved	0	0	1	1	
200 usec buffered step	0	1	0	0	
70 usec buffered step	0	1	0	1	
30 usec buffered step	0	1	1	0	
15 usec buffered step	0	1	1	1	
12 usec buffered step	1	0	0	0	
Spare (for future use)	1 1	0 1	0 1	1 1	thru

R = Disk Drive Type

DESCRIPTION	BITS			
	3	2	1	
Fixed Disk	0	0	0	
Syquest 306R	0	0	1	
Spare (for future use)	0 1	1 1	0 1	thru

T = Encoded Disk Drive Type

0 = Standrad drive

1 = Imbedded servo field drive

D = Data Field Size

01 = 256 data bytes per sector (32 sectors per track)

10 = 512 data bytes per sector (17 sectors per track)

W = Reduced Write Current Cylinder

P = Write Precompensation Cylinder

E = Maximum burst error correction length, up to 11 bits

The number of cylinders, number of heads, drive type, step option, reduced write and write precompensation is obtained from the disk drive manufacturer's specification. The data field size and maximum burst error length are selected by the user. The ECC hardware in the Controller can correct a single burst error of 11 bits or less. If E is set to less than 11 bits, the Controller will flag as uncorrectable any error whose length exceeds E bits, even though the hardware has the capability to correct 11 bits. The host software may use this feature to provide an early warning indicator of growing defects on the drive. The Controller will terminate with an "Illegal Parameter" error (Error 22 Hex) if it detects an invalid parameter or invalid combination of parameters.

The Controller will determine the number of sectors per track from the sector data field size. A data field size of 256 bytes will have 32 sectors per track, while a data field size of 512 bytes will have 17 sectors per track.

B.2.3 Read Initialize Data (Class 0, Opcode 12)

This command permits the host software to determine the initialization parameters set previously by the Initialize Format command and recorded on the maintenance cylinder of the drive. The data format is identical to that of the Initialize Format command.

d = drive, 0 or 1

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	1	0	0	1	0
1	0	0	d	/	/	/	/	/
2	/	/	/	/	/	/	/	/
3	/	/	/	/	/	/	/	/
4	/	/	/	/	/	/	/	/
5	/	/	/	/	/	/	/	/

After sending the command (DCB) to the Controller, the host must read a ten byte data block. This block provides all the intialization parameters to the Controller in the same format passed in the intialize format command.

B.2.4 Stop Drive (Class 0, Opcode 13)

This command causes a removable cartrdige disk drive to spin down and stop so the operator can change the cartridge. After the disk cartridge is changed, the drive must be started by external means, usually closing the cartridge access door or pressing a start switch. Software can check for completion of the cartridge change by using the Test Drive Ready command.

d = drive, 0 or 1

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	1	0	0	1	1
1	0	0	d	/	/	/	/	/
2	/	/	/	/	/	/	/	/
3	/	/	/	/	/	/	/	/
4	/	/	/	/	/	/	/	/
5	/	/	/	/	/	/	/	/