IBM

Personal Computer
Hardware Reference
Library

IBM Fixed Disk Adapter

6361503

IBM Fixed Disk Adapter

Contents

Description

The Fixed Disk Adapter attaches to one or two fixed disk drive units through an internal, daisy-chained, flat cable (data/control cable). Each system supports a maximum of one Fixed Disk Adapter and two fixed disk drives.

The adapter is buffered on the I/O bus and uses the system board's direct memory access (DMA) for record data transfers. An interrupt level also is used to indicate operation completion and status conditions that require microprocessor attention.

The Fixed Disk Adapter provides automatic 11-bit burst error detection and correction in the form of 32-bit error checking and correction (ECC).

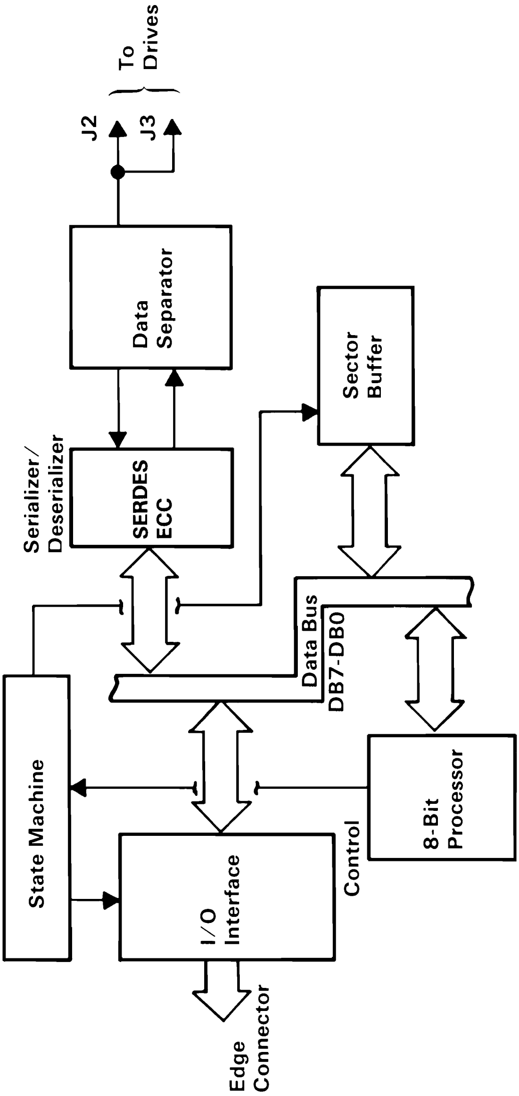
The device level control for the Fixed Disk Adapter is contained on a ROM module on the adapter. A listing of this device level control can be found in "BIOS Listing" of this section.

> **Warning:** The last cylinder on the fixed disk drive is reserved for diagnostic use. The diagnostic write test will destroy any data on this cylinder.

Fixed Disk Controller

The disk controller has two registers that may be accessed by the system unit's microprocessor: a status register and a data register. The 8-bit status register contains the status information of the disk controller, and can be accessed at any time. The 8-bit data register (actually consisting of several registers in a stack with only one register presented to the data bus) stores data, commands, and parameters, and provides the disk controller's status information. Data bytes are read from, or written to the data register in order to program or obtain the results after a particular command. The status register is a read-only register that is used to help the transfer of data between the system unit's microprocessor and the disk controller. The controller-select pulse is generated by writing to port address hex 322.

The following is a block diagram of the IBM Fixed Disk Adapter.



Fixed Disk Adapter Block Diagram

Programming Considerations

Status Register

At the end of all commands from the system board, the disk controller sends a completion status byte to the system board. This byte informs the system unit's microprocessor if an error occurred during the execution of the command. The following shows the format of this byte.

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

Bits 0, 1, 2, 3, 4, 6, 7	These bits are set to zero.
Bit 1	When set, this bit shows an error has occurred during command execution.
Bit 5	This bit shows the logical unit number of the drive.

If the interrupts are enabled, the controller sends an interrupt when it is ready to transfer the status byte. Busy from the disk controller is unasserted when the byte is transferred to complete the command.

Sense Bytes

If the status register receives an error (bit 1 set), the disk controller requests four bytes of sense data. The format for the four bytes is as follows:

Bits	7	6	5	4	3	2	1	0
Byte 0	Address Valid	0	Error Type		Error Code			
Byte 1	0	0	d	Head Number				
Byte 2	Cylinder High			Sector Number				
Byte 3	Cylinder Low							

Remarks
d = drive

Byte 0	**Bits 0, 1, 2, 3**	Error code.
Byte 0	**Bits 4, 5**	Error type.
Byte 0	**Bit 6**	Set to 0 (spare)
Byte 0	**Bit 7**	The address-valid bit. Set only when the previous command required a disk address, in which case it is returned as a 1; otherwise, it is 0.

Disk Controller Error Tables

The following disk controller error tables list the error types and error codes found in byte 0:

	Error Type	Error Code	
Bits	**5 4**	**3 2 1 0**	**Description**
	0 0	0 0 0 0	The controller did not detect any error during the execution of the previous operation.
	0 0	0 0 0 1	The controller did not detect an index signal from the drive.
	0 0	0 0 1 0	The controller did not get a seek-complete signal from the drive after a seek operation (for all non-buffered step seeks).
	0 0	0 0 1 1	The controller detected a write fault from the drive during the last operation.
	0 0	0 1 0 0	After the controller selected the drive, the drive did not respond with a ready signal.
	0 0	0 1 0 1	Not used.
	0 0	0 1 1 0	After stepping the maximum number of cylinders, the controller did not receive the track 00 signal from the drive.
	0 0	0 1 1 1	Not used.
	0 0	1 0 0 0	The drive is still seeking. This status is reported by the Test Drive Ready command for an overlap seek condition when the drive has not completed the seek. No time-out is measured by the controller for the seek to complete.

	Error Type		Error Code				Description
Bits	**5**	**4**	**3**	**2**	**1**	**0**	
	0	1	0	0	0	0	ID Read Error: The controller detected an ECC error in the target ID field on the disk.
	0	1	0	0	0	1	Data Error: The controller detected an uncorrectable ECC error in the target sector during a read operation.
	0	1	0	0	1	0	Address Mark: The controller did not detect the target address mark (AM) on the disk.
	0	1	0	0	1	1	Not used.
	0	1	0	1	0	0	Sector Not Found: The controller found the correct cylinder and head, but not the target sector.
	0	1	0	1	0	1	Seek Error: The cylinder or head address (either or both) did not compare with the expected target address as a result of a seek.
	0	1	0	1	1	0	Not used.
	0	1	0	1	1	1	Not used.
	0	1	1	0	0	0	Correctable Data Error: The controller detected a correctable ECC error in the target field.
	0	1	1	0	0	1	Bad Track: The controller detected a bad track flag during the last operation. No retries are attempted on this error.

	Error Type		Error Code				Description
Bits	**5**	**4**	**3**	**2**	**1**	**0**	
	1	0	0	0	0	0	Invalid Command: The controller has received an invalid command from the system unit.
	1	0	0	0	0	1	Illegal Disk Address. The controller detected an address that is beyond the maximum range.

	Error Type		Error Code				
Bits	**5**	**4**	**3**	**2**	**1**	**0**	**Description**
	1	1	0	0	0	0	RAM Error: The controller detected a data error during the RAM sector-buffer diagnostic test.
	1	1	0	0	0	1	Program Memory Checksum Error: During this internal diagnostic test, the controller detected a program-memory checksum error.
	1	1	0	0	1	0	ECC Polynominal Error: During the controller's internal diagnostic tests, the hardware ECC generator failed its test.

Data Register

The system unit's microprocessor specifies the operation by sending the 6-byte device control block (DCB) to the controller. The figure below shows the composition of the DCB, and defines the bytes that make up the DCB.

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Byte 0	Command Class			Opcode				
Byte 1	0	0	d	Head Number				
Byte 2	Cylinder High		Sector Number					
Byte 3	Cylinder Low							
Byte 4	Interleave or Block Count							
Byte 5	Control Field							

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

Byte 1 Bit 5 identifies the drive number. Bits 4 through 0 contain the disk head number to be selected. Bits 6 and 7 are not used.

Byte 2 Bits 6 and 7 contain the two most significant bits of the cylinder number. Bits 0 through 5 contain the sector number.

Byte 3 Bits 0 through 7 are the eight least-significant bits of the cylinder number.

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

Byte 5 Bits 0 through 7 contain the control field.

Control Byte

Byte 5 is the control field of the DCB and allows the user to select options for several types of disk drives. The format of this byte is as follows:

Bits	7	6	5	4	3	2	1	0
	r	a	0	0	0	s	s	s

Remarks

r = retries
s = step option
a = retry option on data ECC error

Bit 7 Disables the four retries by the controller on all disk-access commands. Set this bit only during the evaluation of the performance of a disk drive.

Bit 6 If set to 0 during read commands, a reread is attempted when an ECC error occurs. If no error occurs during reread, the command will finish without an error status. If this bit is set to 1, no reread is attempted.

Bits 5, 4, 3 Set to 0.

Bits 2, 1, 0 These bits define the type of drive and select the step option. See the following figure.

Bits 2, 1, 0	
0 0 0	This drive is not specified and defaults to 3 milliseconds per step
0 0 1	N/A
0 1 0	N/A
0 1 1	N/A
1 0 0	200 microseconds per step.
1 0 1	70 microseconds per step (specified by BIOS).
1 1 0	3 milliseconds per step.
1 1 1	3 milliseconds per step.

Command Summary

<table>
<tr><th>Command</th><th>Data Control Block</th><th>Remarks</th></tr>
<tr><td>Test Drive Ready (Class 0, Opcode 00)</td><td><table>
<tr><td>Bit</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>Byte 0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>Byte 1</td><td>0</td><td>0</td><td>d</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td></tr>
</table></td><td>d = drive (0 or 1)
x = don't care
Bytes 2, 3, 4, 5 = don't care</td></tr>
<tr><td>Recalibrate (Class 0, Opcode 01)</td><td><table>
<tr><td>Bit</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>Byte 0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>Byte 1</td><td>0</td><td>0</td><td>d</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td></tr>
<tr><td>Byte 5</td><td>r</td><td>0</td><td>0</td><td>0</td><td>0</td><td>s</td><td>s</td><td>s</td></tr>
</table></td><td>d = drive (0 or 1)
x = don't care
r = retries
s = Step Option
Bytes 2, 3, 4 = don't care
ch = cylinder high</td></tr>
<tr><td>Reserved (Class 0, Opcode 02)</td><td></td><td>This Opcode is not used.</td></tr>
<tr><td>Request Sense Status (Class 0, Opcode 03)</td><td><table>
<tr><td>Bit</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>Byte 0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>Byte 1</td><td>0</td><td>0</td><td>d</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td></tr>
</table></td><td>d = drive (0 or 1)
x = don't care
Bytes 2, 3, 4, 5 = don't care</td></tr>
<tr><td>Format Drive (Class 0, Opcode 04)</td><td><table>
<tr><td>Bit</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>Byte 0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>Byte 1</td><td>0</td><td>0</td><td>d</td><td colspan="5">Head Number</td></tr>
<tr><td>Byte 2</td><td colspan="2">ch</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>Byte 3</td><td colspan="8">Cylinder Low</td></tr>
<tr><td>Byte 4</td><td>0</td><td>0</td><td>0</td><td colspan="5">Interleave</td></tr>
<tr><td>Byte 5</td><td>r</td><td>0</td><td>0</td><td>0</td><td>0</td><td>s</td><td>s</td><td>s</td></tr>
</table></td><td>d = drive (0 or 1)
r = retries
s = step option
ch = cylinder high
Interleave 1 to 16 for 512-byte sectors.</td></tr>
<tr><td>Ready Verify (Class 0, Opcode 05)</td><td><table>
<tr><td>Bit</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>Byte 0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>Byte 1</td><td>0</td><td>0</td><td>d</td><td colspan="5">Head Number</td></tr>
<tr><td>Byte 2</td><td colspan="2">ch</td><td colspan="6">Sector Number</td></tr>
<tr><td>Byte 3</td><td colspan="8">Cylinder Low</td></tr>
<tr><td>Byte 4</td><td colspan="8">Block Count</td></tr>
<tr><td>Byte 5</td><td>r</td><td>a</td><td>0</td><td>0</td><td>0</td><td>s</td><td>s</td><td>s</td></tr>
</table></td><td>d = drive (0 or 1)
r = retries
s = step option
a = retry option on data ECC
ch = cylinder high</td></tr>
</table>

Command	Data Control Block									Remarks
Format Track	Bit	7	6	5	4	3	2	1	0	d = drive (0 or 1)
(Class 0,	Byte 0	0	0	0	0	0	1	1	0	r = retries
Opcode 06)	Byte 1	0	0	d	Head Number					s = step option
	Byte 2	ch		0	0	0	0	0	0	ch = cylinder high
	Byte 3	Cylinder Low								
	Byte 4	0	0	0	Interleave					Interleave 1 to 16
	Byte 5	r	0	0	0	0	s	s	s	for 512-byte sectors.
Format Bad Track	Bit	7	6	5	4	3	2	1	0	d = drive (0 or 1)
(Class 0,	Byte 0	0	0	0	0	0	1	1	1	r = retries
Opcode 07)	Byte 1	0	0	d	Head Number					s = step option
	Byte 2	ch		0	0	0	0	0	0	ch = cylinder high
	Byte 3	Cylinder Low								
	Byte 4	0	0	0	Interleave					Interleave 1 to 16
	Byte 5	r	0	0	0	0	s	s	s	for 512-byte sectors.
Read	Bit	7	6	5	4	3	2	1	0	d = drive (0 or 1)
(Class 0,	Byte 0	0	0	0	0	1	0	0	0	r = retries
Opcode 08)	Byte 1	0	0	d	Head Number					a = retry option on data ECC error
	Byte 2	ch		Sector Number						s = step option
	Byte 3	Cylinder Low								ch = cylinder high
	Byte 5	r	a	0	0	0	s	s	s	
Reserved (Class 0, Opcode 09)										This Opcode is not used.
Write	Bit	7	6	5	4	3	2	1	0	d = drive (0 or 1)
(Class 0,	Byte 0	0	0	0	0	1	0	1	0	r = retries
Opcode 0A)	Byte 1	0	0	d	Head Number					s = step option
	Byte 2	ch		Sector Number						ch = cylinder high
	Byte 3	Cylinder Low								
	Byte 4	Block Count								
	Byte 5	r	0	0	0	0	s	s	s	
Seek	Bit	7	6	5	4	3	2	1	0	d = drive (0 or 1)
(Class 0,	Byte 0	0	0	0	0	1	0	1	1	r = retries
Opcode 0B)	Byte 1	0	0	d	Head Number					s = step option
	Byte 2	ch		0	0	0	0	0	0	x = don't care
	Byte 3	Cylinder Low								ch = cylinder high
	Byte 4	x	x	x	x	x	x	x	x	
	Byte 5	r	0	0	0	0	s	s	s	

Command	Data Control Block									Remarks
Initialize Drive Characteristics* (Class 0, Opcode 0C)	Bit	7	6	5	4	3	2	1	0	Bytes 1, 2, 3, 4, 5, = don't care
	Byte 0	0	0	0	0	1	1	0	0	
Read ECC Burst Error Length (Class 0, Opcode 0D)	Bit	7	6	5	4	3	2	1	0	Bytes 1, 2, 3, 4, 5, = don't care
	Byte 0	0	0	0	0	1	1	0	1	
Read Data from Sector Buffer (Class 0, Opcode 0E)	Bit	7	6	5	4	3	2	1	0	Bytes 1, 2, 3, 4, 5, = don't care
	Byte 0	0	0	0	0	1	1	1	0	
Write Data to Sector Buffer (Class 0, Opcode 0F)	Bit	7	6	5	4	3	2	1	0	Bytes 1, 2, 3, 4, 5, = don't care
	Byte 0	0	0	0	0	1	1	1	1	
RAM Diagnostic (Class 7, Opcode 00)	Bit	7	6	5	4	3	2	1	0	Bytes 1, 2, 3, 4, 5, = don't care
	Byte 0	1	1	1	0	0	0	0	0	
Reserved (Class 7, Opcode 01)										This Opcode is not used.
Reserved (Class 7, Opcode 02)										This Opcode is not used.

*Initialize Drive Characteristics: The DBC must be followed by eight additional bytes.

Maximum number of cylinders	(2 bytes)
Maximum number of heads	(1 byte)
Start reduced write current cylinder	(2 bytes)
Start write precompensation cylinder	(2 bytes)
Maximum ECC data burst length	(1 byte)

Command	Data Control Block									Remarks
Drive Diagnostic (Class 7, Opcode 03)	Bit	7	6	5	4	3	2	1	0	d = drive (0 or 1)
	Byte 0	1	1	1	0	0	0	1	1	s = step option
	Byte 1	0	0	d	x	x	x	x	x	r = retries
	Byte 2	x	x	x	x	x	x	x	x	x = don't care
	Byte 3	x	x	x	x	x	x	x	x	
	Byte 4	x	x	x	x	x	x	x	x	
	Byte 5	r	0	0	0	0	s	s	s	
Controller Internal Diagnostics (Class 7, Opcode 04)	Bit	7	6	5	4	3	2	1	0	Bytes 1, 2, 3, 4, 5, = don't care
	Byte 0	1	1	1	0	0	1	0	0	
Read Long* (Class 7, Opcode 05)	Bit	7	6	5	4	3	2	1	0	d = drive (0 or 1)
	Byte 0	1	1	1	0	0	1	0	1	s = step option
	Byte 1	0	0	d	Head Number					r = retries
	Byte 2	ch		Sector Number						ch = cylinder high
	Byte 3	Cylinder Low								
	Byte 4	Block Count								
	Byte 5	r	0	0	0	0	s	s	s	
Write Long** (Class 7, Opcode 06)	Bit	7	6	5	4	3	2	1	0	d = drive (0 or 1)
	Byte 0	1	1	1	0	0	1	1	0	s = step option
	Byte 1	0	0	d	Head Number					r = retries
	Byte 2	ch		Sector Number						ch = cylinder high
	Byte 3	Cylinder Low								
	Byte 4	Block Count								
	Byte 5	r	0	0	0	0	s	s	s	

*Returns 512 bytes plus 4 bytes of ECC data per sector.

**Requires 512 bytes plus 4 bytes of ECC data per sector.

Programming Summary

The two least-significant bits of the address bus are sent to the system board's I/O port decoder, which has two sections. One section is enabled by the I/O read signal (-IOR) and the other by the I/O write signal (-IOW). The result is a total of four read/write ports assigned to the disk controller board.

The address enable signal (AEN) is asserted by the system board when DMA is controlling data transfer. When AEN is asserted, the I/O port decoder is disabled.

The following figure is a table of the read/write ports.

R/W	Port Address	Function
Read	320	Read data (from controller to system unit).
Write	320	Write data (from system unit to controller).
Read	321	Read controller hardware status.
Write	321	Controller reset.
Read	322	Reserved.
Write	322	Generate controller-select pulse.
Read	323	Not used.
Write	323	Write pattern to DMA and interrupt mask register.

Interface

The following lines are used by the disk controller:

A0-A19 Positive true 20-bit address. The least-significant 10 bits contain the I/O address within the range of hex 320 to hex 323 when an I/O read or write is executed by the system unit. The full 20 bits are decoded to address the read-only memory (ROM) between the addresses of hex C8000 and C9FFF.

DO-D7 Positive 8-bit data bus over which data and status information is passed between the system board and the controller.

-IOR Negative true signal that is asserted when the system board reads status or data from the controller under either programmed I/O or DMA control.

-IOW Negative true signal that is asserted when the system board sends a command or data to the controller under either programmed I/O or DMA control.

AEN Positive true signal that is asserted when the DMA in the system board is generating the I/O Read (-IOR) or I/O Write (-IOW) signals and has control of the address and data buses.

RESET Positive true signal that forces the disk controller to its initial power-up condition.

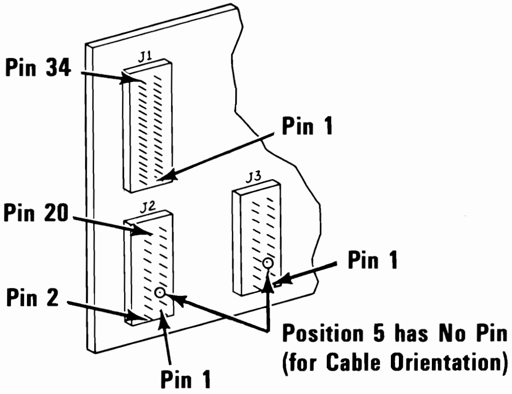
IRQ 5 Positive true interrupt-request signal that is asserted by the controller when enabled to interrupt the system board on the return ending status byte from the controller.

DRQ 3 Positive true DMA-request signal that is asserted by the controller when data is available for transfer to or from the controller under DMA control. This signal remains active until the system board's DMA channel activates the DMA-acknowledge signal (-DACK 3) in response.

-DACK 3 This signal is true when negative, and is generated by the system board DMA channel in response to a DMA request (DRQ 3).

Specifications

The Fixed Disk Adapter connector and interface specifications follow.



Disk Drive Connector J1

Signal	Pin Number	
Ground-Odd Numbers	1-33	
Reserved	4,16,30,32	
– Reduced Write Current	2	←
– Write Gate	6	←
– Seek Complete	8	→
– Track 00	10	→
– Write Fault	12	→
– Head Select 2^0	14	←
– Head Select 2^1	18	←
– Index	20	→
– Ready	22	→
– Step	24	←
– Drive Select 1	26	←
– Drive Select 2	28	←
– Direction In	34	←

Disk Adapter Connector J1

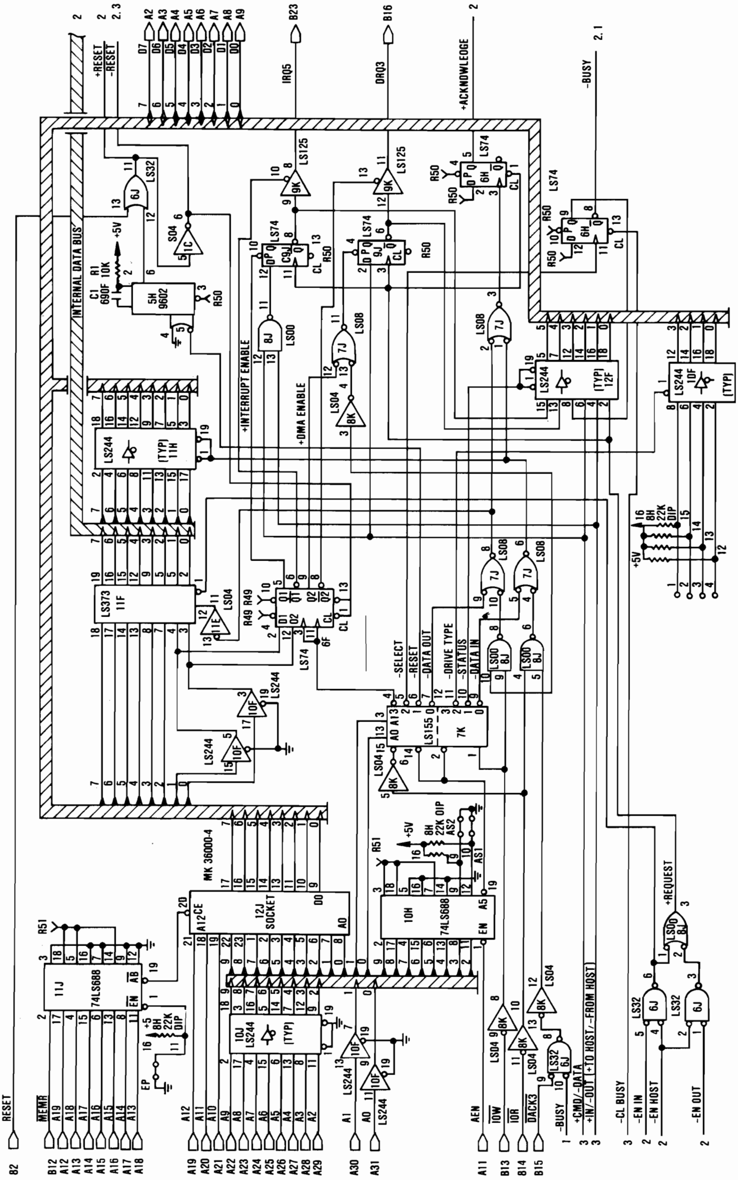
Disk Drive Connector J2 or J3

Signal	Pin Number	
Ground	2,4,6,8,12,16,20	
Drive Select	1	→
Reserved	3,7	
Spare	9,10,5 (No Pin)	
Ground	11	
MFM Wire Data	13	←
– MTM Write Data	14	←
Ground	15	
MFM Read Data	17	→
– MFM Read Data	18	→
Ground	19	

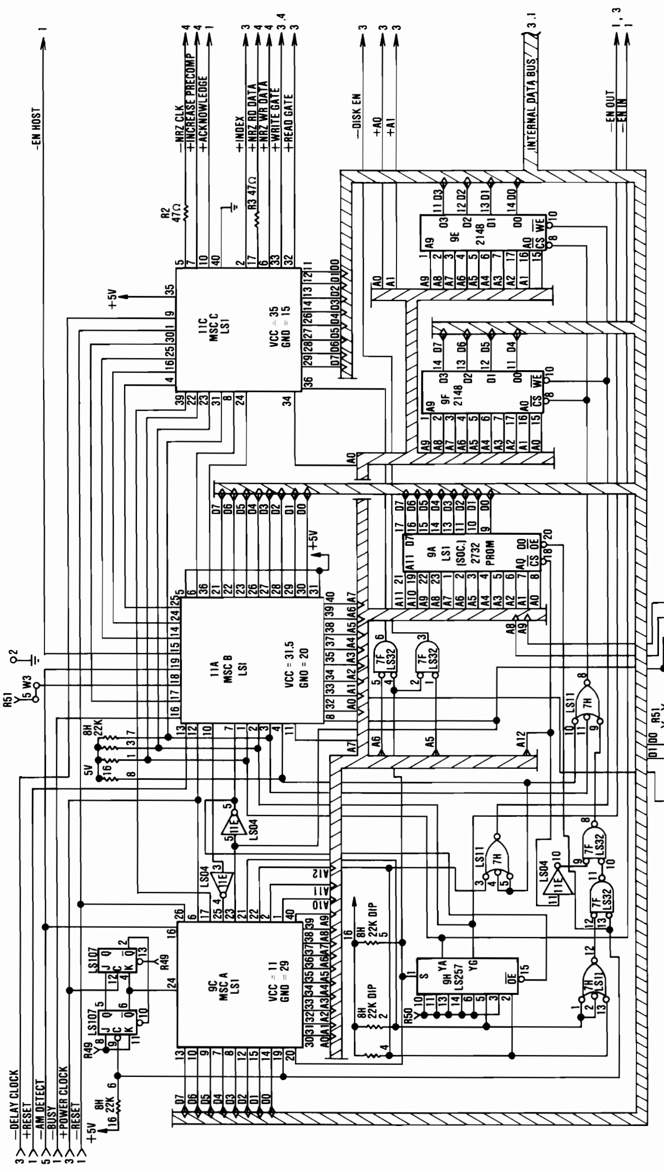
Disk Adapter Connector J2 or J3

Fixed Disk Adapter Interface Specifications

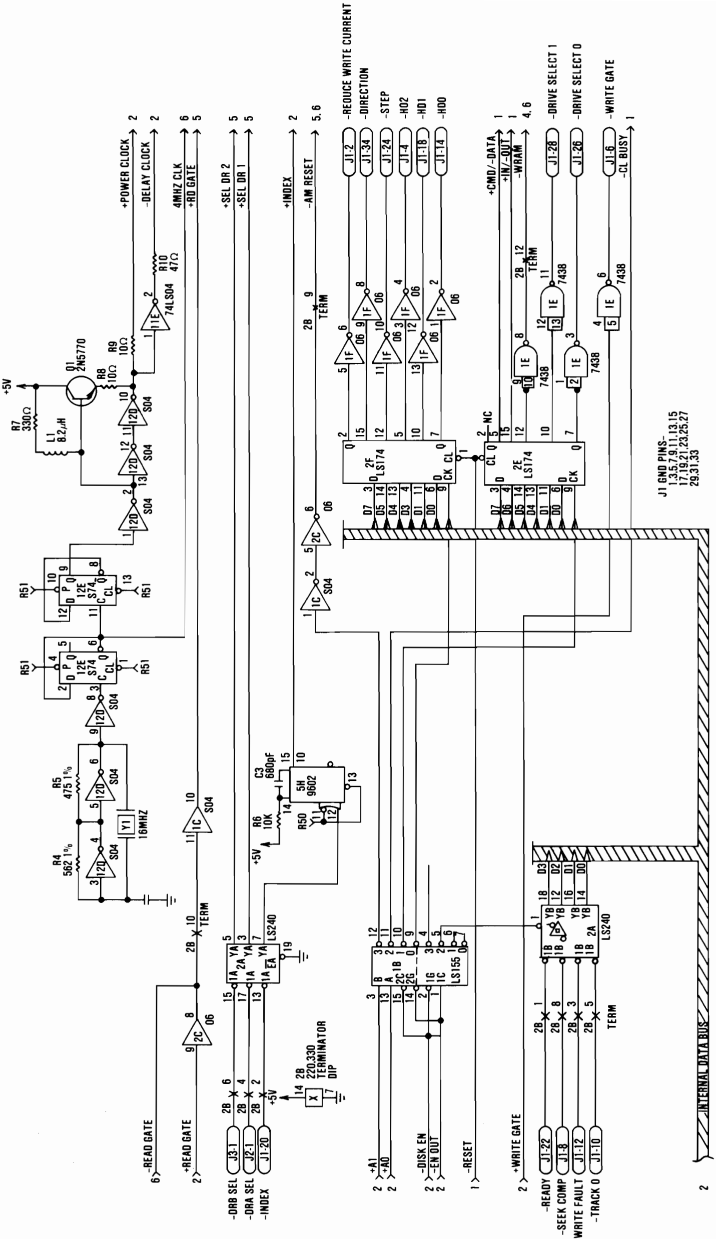
Logic Diagrams



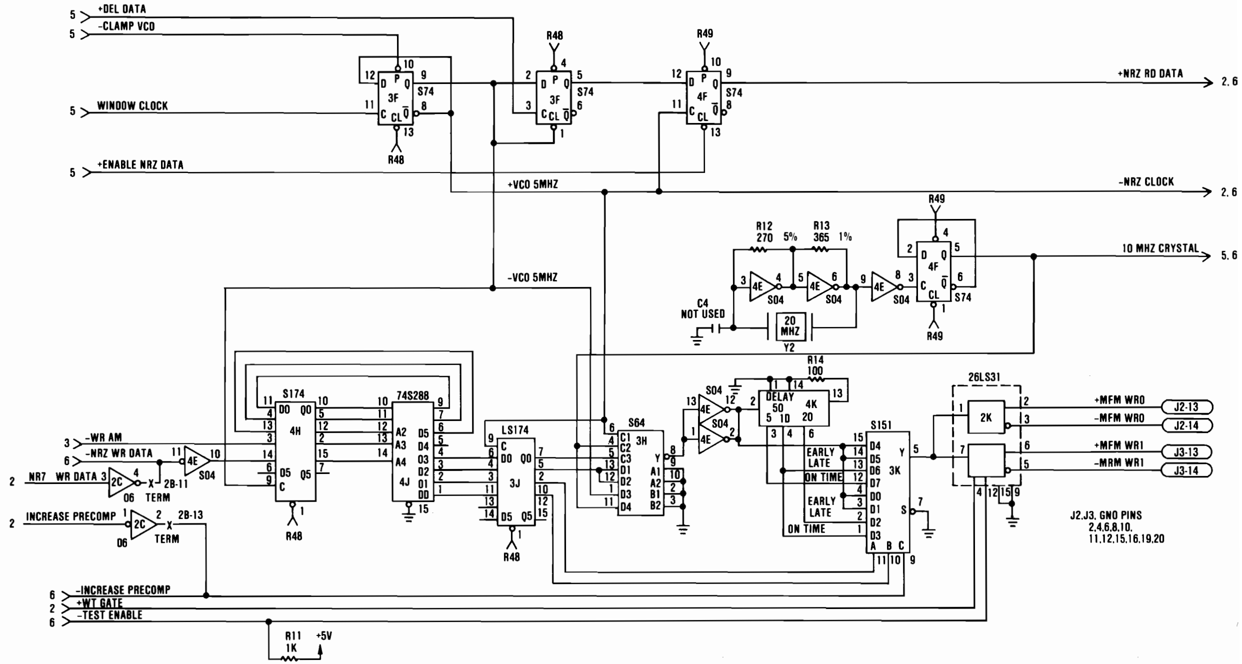
Fixed Disk Adapter (Sheet 1 of 6)



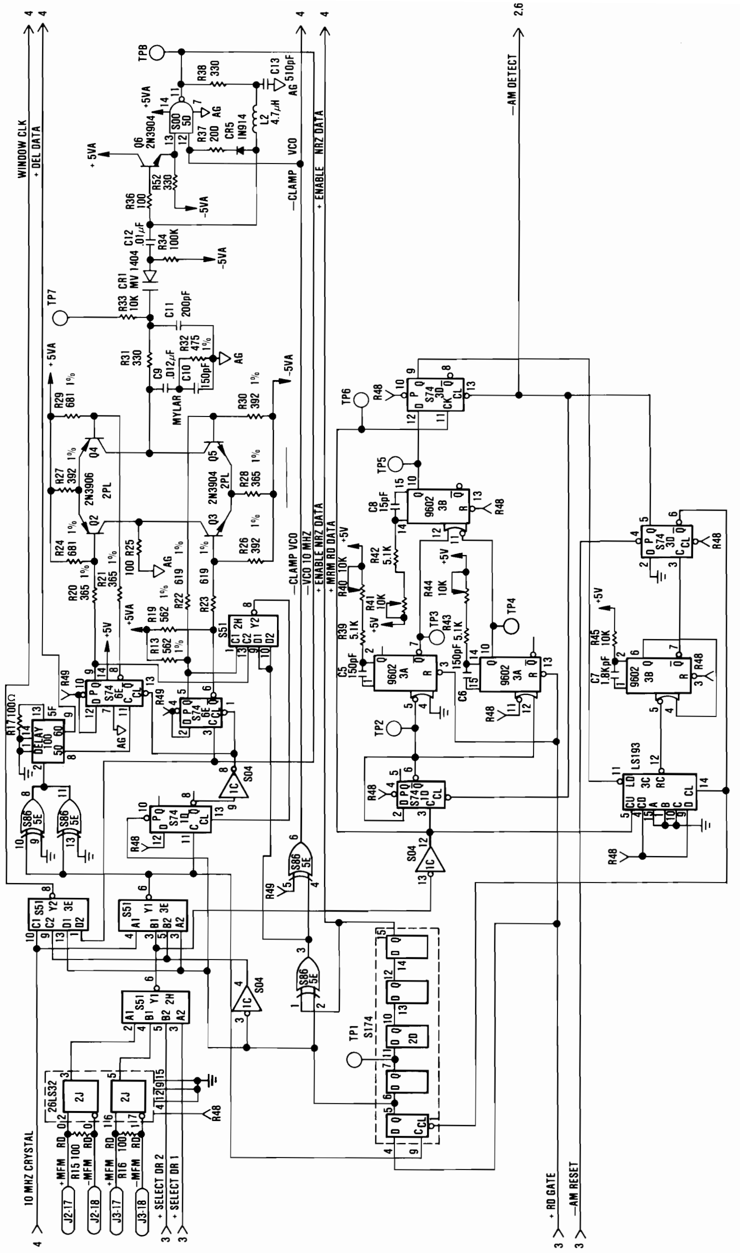
Fixed Disk Adapter (Sheet 2 of 6)



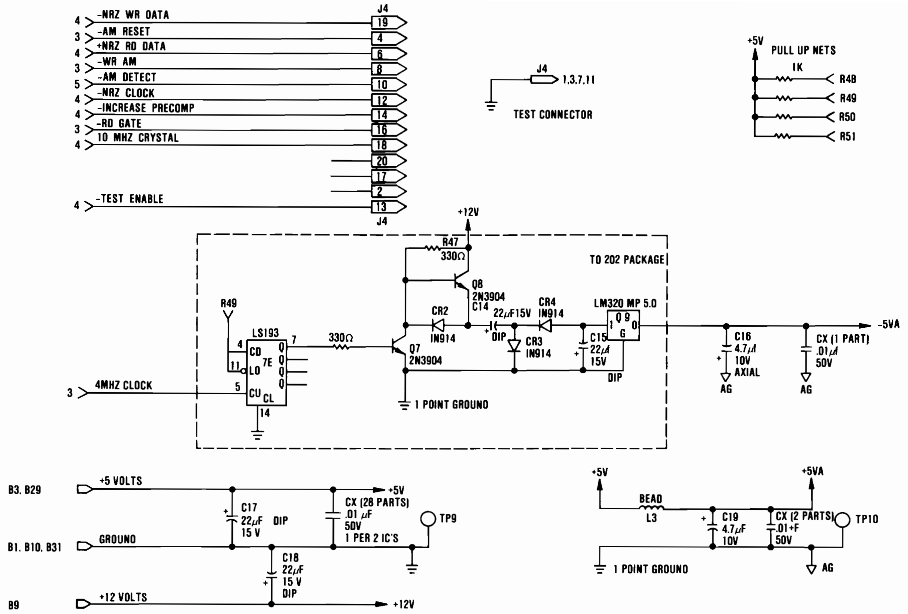
Fixed Disk Adapter (Sheet 3 of 6)



Fixed Disk Adapter (Sheet 4 of 6)



Fixed Disk Adapter (Sheet 5 of 6)



NOTES:
UNLESS OTHERWISE SPECIFIED:
1. ALL RESISTORS 1/4 W, 5%, CARBON FILTER.
2. ALL CAPS +10V OR GREATER +10%.
3. NO MORE THAN 15 LOADS PER PULLUP NET.

Fixed Disk Adapter (Sheet 6 of 6)

BIOS Listing

The BIOS Listing for the IBM Fixed Disk Adapter follows.

```
$TITLE(FIXED DISK BIOS FOR IBM DISK CONTROLLER)

;-- INT 13 ----------------------------------------------------
;                                                             :
; FIXED DISK I/O INTERFACE                                    :
;                                                             :
;       THIS INTERFACE PROVIDES ACCESS TO 5 1/4" FIXED DISKS  :
;       THROUGH THE IBM FIXED DISK CONTROLLER.                :
;                                                             :
;--------------------------------------------------------------

;--------------------------------------------------------------
;       THE  BIOS ROUTINES ARE MEANT TO BE ACCESSED THROUGH   :
;       SOFTWARE INTERRUPTS ONLY.  ANY ADDRESSES PRESENT IN   :
;       THE LISTINGS  ARE INCLUDED  ONLY FOR  COMPLETENESS,   :
;       NOT FOR  REFERENCE.   APPLICATIONS WHICH  REFERENCE   :
;       ABSOLUTE   ADDRESSES   WITHIN    THE    CODE  SEGMENT :
;       VIOLATE THE STRUCTURE AND DESIGN OF BIOS.             :
;--------------------------------------------------------------
;
; INPUT    (AH = HEX VALUE)
;
;       (AH)=00 RESET DISK (DL = 80H,81H) / DISKETTE
;       (AH)=01 READ THE STATUS OF THE LAST DISK OPERATION INTO (AL)
;               NOTE: DL < 80H - DISKETTE
;                     DL > 80H - DISK
;       (AH)=02 READ THE DESIRED SECTORS INTO MEMORY
;       (AH)=03 WRITE THE DESIRED SECTORS FROM MEMORY
;       (AH)=04 VERIFY THE DESIRED SECTORS
;       (AH)=05 FORMAT THE DESIRED TRACK
;       (AH)=06 FORMAT THE DESIRED TRACK AND SET BAD SECTOR FLAGS
;       (AH)=07 FORMAT THE DRIVE STARTING AT THE DESIRED TRACK
;       (AH)=08 RETURN THE CURRENT DRIVE PARAMETERS
;
;       (AH)=09 INITIALIZE DRIVE PAIR CHARACTERISTICS
;               INTERRUPT 41 POINTS TO DATA BLOCK
;       (AH)=0A READ LONG
;       (AH)=0B WRITE LONG
;       NOTE: READ AND WRITE LONG ENCOMPASS 512 + 4 BYTES ECC
;       (AH)=0C SEEK
;       (AH)=0D ALTERNATE DISK RESET (SEE DL)
;       (AH)=0E READ SECTOR BUFFER
;       (AH)=0F WRITE SECTOR BUFFER,
;               (RECOMMENDED PRACTICE BEFORE FORMATTING)
;       (AH)=10 TEST DRIVE READY
;       (AH)=11 RECALIBRATE
;       (AH)=12 CONTROLLER RAM DIAGNOSTIC
;       (AH)=13 DRIVE DIAGNOSTIC
;       (AH)=14 CONTROLLER INTERNAL DIAGNOSTIC
;
;               REGISTERS USED FOR FIXED DISK OPERATIONS
;
;               (DL)    -  DRIVE NUMBER     (80H-87H FOR DISK, VALUE CHECKED)
;               (DH)    -  HEAD NUMBER      (0-7 ALLOWED, NOT VALUE CHECKED)
;               (CH)    -  CYLINDER NUMBER  (0-1023, NOT VALUE CHECKED)(SEE CL)
;               (CL)    -  SECTOR NUMBER    (1-17, NOT VALUE CHECKED)
;
;                          NOTE: HIGH 2 BITS OF CYLINDER NUMBER ARE PLACED
;                                IN THE HIGH 2 BITS OF THE CL REGISTER
;                                (10 BITS TOTAL)
;               (AL)    -  NUMBER OF SECTORS (MAXIMUM POSSIBLE RANGE 1-80H,
;                                             FOR READ/WRITE LONG 1-79H)
;                          (INTERLEAVE VALUE FOR FORMAT 1-16D)
;               (ES:BX) -  ADDRESS OF BUFFER FOR READS AND WRITES,
;                          (NOT REQUIRED FOR VERIFY)
;
; OUTPUT
;       AH = STATUS OF CURRENT OPERATION
;            STATUS BITS ARE DEFINED IN THE EQUATES BELOW
;       CY = 0  SUCCESSFUL OPERATION (AH=0 ON RETURN)
;       CY = 1  FAILED OPERATION (AH HAS ERROR REASON)
;
;       NOTE:   ERROR 11H  INDICATES THAT THE DATA READ HAD A RECOVERABLE
;               ERROR WHICH WAS CORRECTED BY THE ECC ALGORITHM.  THE DATA
;               IS PROBABLY GOOD,   HOWEVER THE BIOS ROUTINE INDICATES AN
;               ERROR TO ALLOW THE CONTROLLING PROGRAM A CHANCE TO DECIDE
;               FOR  ITSELF.  THE  ERROR  MAY  NOT  RECUR  IF THE DATA IS
```

```
                  ;               REWRITTEN. (AL) CONTAINS THE BURST LENGTH.
                  ;
                  ;       IF DRIVE PARAMETERS WERE REQUESTED,
                  ;
                  ;       DL = NUMBER OF CONSECUTIVE ACKNOWLEDGING DRIVES ATTACHED (0-2)
                  ;               (CONTROLLER CARD ZERO TALLY ONLY)
                  ;       DH = MAXIMUM USEABLE VALUE FOR HEAD NUMBER
                  ;       CH = MAXIMUM USEABLE VALUE FOR CYLINDER NUMBER
                  ;       CL = MAXIMUM USEABLE VALUE FOR SECTOR NUMBER
                  ;            AND CYLINDER NUMBER HIGH BITS
                  ;
                  ;       REGISTERS WILL BE PRESERVED EXCEPT WHEN THEY ARE USED TO RETURN
                  ;       INFORMATION.
                  ;
                  ;       NOTE: IF AN ERROR IS REPORTED BY THE DISK CODE, THE APPROPRIATE
                  ;               ACTION IS TO RESET THE DISK, THEN RETRY THE OPERATION.
                  ;
                  ;----------------------------------------------------------------------

 00FF             SENSE_FAIL      EQU     0FFH            ; SENSE OPERATION FAILED
 00BB             UNDEF_ERR       EQU     0BBH            ; UNDEFINED ERROR OCCURRED
 0080             TIME_OUT        EQU     80H             ; ATTACHMENT FAILED TO RESPOND
 0040             BAD_SEEK        EQU     40H             ; SEEK OPERATION FAILED
 0020             BAD_CNTLR       EQU     20H             ; CONTROLLER HAS FAILED
 0011             DATA_CORRECTED  EQU     11H             ; ECC CORRECTED DATA ERROR
 0010             BAD_ECC         EQU     10H             ; BAD ECC ON DISK READ
 000B             BAD_TRACK       EQU     0BH             ; BAD TRACK FLAG DETECTED
 0009             DMA_BOUNDARY    EQU     09H             ; ATTEMPT TO DMA ACROSS 64K BOUNDARY
 0007             INIT_FAIL       EQU     07H             ; DRIVE PARAMETER ACTIVITY FAILED
 0005             BAD_RESET       EQU     05H             ; RESET FAILED
 0004             RECORD_NOT_FND  EQU     04H             ; REQUESTED SECTOR NOT FOUND
 0002             BAD_ADDR_MARK   EQU     02H             ; ADDRESS MARK NOT FOUND
 0001             BAD_CMD         EQU     01H             ; BAD COMMAND PASSED TO DISK I/O

                  ;----------------------------------------
                  ;       INTERRUPT AND STATUS AREAS      :
                  ;----------------------------------------

----              DUMMY   SEGMENT AT 0
0034                      ORG     0DH*4                   ; FIXED DISK INTERRUPT VECTOR
0034              HDISK_INT       LABEL   DWORD
004C                      ORG     13H*4                   ; DISK INTERRUPT VECTOR
004C              ORG_VECTOR      LABEL   DWORD
0064                      ORG     19H*4                   ; BOOTSTRAP INTERRUPT VECTOR
0064              BOOT_VEC        LABEL   DWORD
0078                      ORG     1EH*4                   ; DISKETTE PARAMETERS
0078              DISKETTE_PARM   LABEL   DWORD
0100                      ORG     040H*4                  ; NEW DISKETTE INTERRUPT VECTOR
0100              DISK_VECTOR     LABEL   DWORD
0104                      ORG     041H*4                  ; FIXED DISK PARAMETER VECTOR
0104              HF_TBL_VEC      LABEL   DWORD
7C00                      ORG     7C00H                   ; BOOTSTRAP LOADER VECTOR
7C00              BOOT_LOCN       LABEL   FAR
----              DUMMY   ENDS

----              DATA    SEGMENT AT 40H
0042                      ORG     42H
0042              CMD_BLOCK       LABEL   BYTE
0042 (7 ??)       HD_ERROR        DB      7 DUP(?)        ; OVERLAYS DISKETTE STATUS
006C                      ORG     06CH
006C ????         TIMER_LOW       DW      ?               ; TIMER LOW WORD
0072                      ORG     72H
0072 ????         RESET_FLAG      DW      ?               ; 1234H IF KEYBOARD RESET UNDERWAY
0074                      ORG     74H
0074 ??           DISK_STATUS     DB      ?               ; FIXED DISK STATUS BYTE
0075 ??           HF_NUM          DB      ?               ; COUNT OF FIXED DISK DRIVES
0076 ??           CONTROL_BYTE    DB      ?               ; CONTROL BYTE DRIVE OPTIONS
0077 ??           PORT_OFF        DB      ?               ; PORT OFFSET
----              DATA    ENDS

----              CODE    SEGMENT

                  ;-------------------------------------------------------
                  ; HARDWARE SPECIFIC VALUES                              :
                  ;                                                       :
                  ;  -  CONTROLLER I/O PORT                               :
                  ;       > WHEN READ FROM:                               :
```

```
LOC OBJ          LINE    SOURCE

                         ;        HF_PORT+0 - READ DATA (FROM CONTROLLER TO CPU)  :
                         ;        HF_PORT+1 - READ CONTROLLER HARDWARE STATUS     :
                         ;                    (CONTROLLER TO CPU)                 :
                         ;        HF_PORT+2 - READ CONFIGURATION SWITCHES         :
                         ;        HF_PORT+3 - NOT USED                            :
                         ;      > WHEN WRITTEN TO:                                :
                         ;        HF_PORT+0 - WRITE DATA (FROM CPU TO CONTROLLER) :
                         ;        HF_PORT+1 - CONTROLLER RESET                    :
                         ;        HF_PORT+2 - GENERATE CONTROLLER SELECT PULSE    :
                         ;        HF_PORT+3 - WRITE PATTERN TO DMA AND INTERRUPT  :
                         ;                    MASK REGISTER                       :
                         ;                                                        :
                         ;---------------------------------------------------------

 0320                    HF_PORT         EQU     0320H           ; DISK PORT
 0008                    R1_BUSY         EQU     00001000B       ; DISK PORT 1 BUSY BIT
 0004                    R1_BUS          EQU     00000100B       ;               COMMAND/DATA BIT
 0002                    R1_IOMODE       EQU     00000010B       ;               MODE BIT
 0001                    R1_REQ          EQU     00000001B       ;               REQUEST BIT

 0047                    DMA_READ        EQU     01000111B       ; CHANNEL 3 (047H)
 004B                    DMA_WRITE       EQU     01001011B       ; CHANNEL 3 (04BH)
 0000                    DMA             EQU     0               ; DMA ADDRESS
 0082                    DMA_HIGH        EQU     082H            ; PORT FOR HIGH 4 BITS OF DMA

 0000                    TST_RDY_CMD     EQU     00000000B       ; CNTLR READY (00H)
 0001                    RECAL_CMD       EQU     00000001B       ;       RECAL (01H)
 0003                    SENSE_CMD       EQU     00000011B       ;       SENSE (03H)
 0004                    FMTDRV_CMD      EQU     00000100B       ;       DRIVE (04H)
 0005                    CHK_TRK_CMD     EQU     00000101B       ;       T CHK (05H)
 0006                    FMTTRK_CMD      EQU     00000110B       ;       TRACK (06H)
 0007                    FMTBAD_CMD      EQU     00000111B       ;       BAD   (07H)
 0008                    READ_CMD        EQU     00001000B       ;       READ  (08H)
 000A                    WRITE_CMD       EQU     00001010B       ;       WRITE (0AH)
 000B                    SEEK_CMD        EQU     00001011B       ;       SEEK  (0BH)
 000C                    INIT_DRV_CMD    EQU     00001100B       ;       INIT  (0CH)
 000D                    RD_ECC_CMD      EQU     00001101B       ;       BURST (0DH)
 000E                    RD_BUFF_CMD     EQU     00001110B       ;       BUFFR (0EH)
 000F                    WR_BUFF_CMD     EQU     00001111B       ;       BUFFR (0FH)
 00E0                    RAM_DIAG_CMD    EQU     11100000B       ;       RAM   (E0H)
 00E3                    CHK_DRV_CMD     EQU     11100011B       ;       DRV   (E3H)
 00E4                    CNTLR_DIAG_CMD  EQU     11100100B       ;       CNTLR (E4H)
 00E5                    RD_LONG_CMD     EQU     11100101B       ;       RLONG (E5H)
 00E6                    WR_LONG_CMD     EQU     11100110B       ;       WLONG (E6H)

 0020                    INT_CTL_PORT    EQU     20H             ; 8259 CONTROL PORT
 0020                    EOI             EQU     20H             ; END OF INTERRUPT COMMAND

 0008                    MAX_FILE        EQU     8
 0002                    S_MAX_FILE      EQU     2

                                 ASSUME  CS:CODE
0000                             ORG     0H
0000 55                          DB      055H                    ; GENERIC BIOS HEADER
0001 AA                          DB      0AAH
0002 10                          DB      16D

                         ;-------------------------------------------------------------
                         ; FIXED DISK I/O SETUP                                        :
                         ;                                                             :
                         ;  -  ESTABLISH TRANSFER VECTORS FOR THE FIXED DISK           :
                         ;  -  PERFORM POWER ON DIAGNOSTICS                            :
                         ;     SHOULD AN ERROR OCCUR A "1701" MESSAGE IS DISPLAYED     :
                         ;                                                             :
                         ;-------------------------------------------------------------

0003                     DISK_SETUP      PROC    FAR
0003 EB1E                        JMP     SHORT   L3
0005 35303030303539              DB      '5000059 (C)COPYRIGHT  IBM 1982'       ; COPYRIGHT NOTICE
     20284329434F50
     59524947485420
     2049424D203139
     3832
0023                     L3:
                                 ASSUME  DS:DUMMY
0023 2BC0                        SUB     AX,AX                                  ; ZERO
0025 8ED8                        MOV     DS,AX
```

```
LOC OBJ              LINE    SOURCE

0027 FA                             CLI
0028 A14C00                         MOV     AX,WORD PTR ORG_VECTOR                  ; GET DISKETTE VECTOR
002B A30001                         MOV     WORD PTR DISK_VECTOR,AX                 ;  INTO INT 40H
002E A14E00                         MOV     AX,WORD PTR ORG_VECTOR+2
0031 A30201                         MOV     WORD PTR DISK_VECTOR+2,AX
0034 C7064C005602                   MOV     WORD PTR ORG_VECTOR, OFFSET DISK_IO     ; HDISK HANDLER
003A 8C0E4E00                       MOV     WORD PTR ORG_VECTOR+2,CS
003E B86007                         MOV     AX, OFFSET HD_INT                       ; HDISK INTERRUPT
0041 A33400                         MOV     WORD PTR HDISK_INT,AX
0044 8C0E3600                       MOV     WORD PTR HDISK_INT+2,CS
0048 C70664008601                   MOV     WORD PTR BOOT_VEC,OFFSET BOOT_STRAP     ; BOOTSTRAP
004E 8C0E6600                       MOV     WORD PTR BOOT_VEC+2,CS
0052 C7060401E703                   MOV     WORD PTR HF_TBL_VEC,OFFSET FD_TBL       ; PARAMETER TBL
0058 8C0E0601                       MOV     WORD PTR HF_TBL_VEC+2,CS
005C FB                             STI

                                    ASSUME  DS:DATA
005D B84000                         MOV     AX,DATA                 ; ESTABLISH SEGMENT
0060 8ED8                           MOV     DS,AX
0062 C606740000                     MOV     DISK_STATUS,0           ; RESET THE STATUS INDICATOR
0067 C606750000                     MOV     HF_NUM,0                ; ZERO COUNT OF DRIVES
006C C606430000                     MOV     CMD_BLOCK+1,0           ; DRIVE ZERO, SET VALUE IN BLOCK
0071 C606770000                     MOV     PORT_OFF,0              ; ZERO CARD OFFSET

0076 B92500                         MOV     CX,25H                  ; RETRY COUNT
0079                        L4:
0079 E8F200                         CALL    HD_RESET_1              ; RESET CONTROLLER
007C 7305                           JNC     L7
007E E2F9                           LOOP    L4                      ; TRY RESET AGAIN
0080 E9BF00                         JMP     ERROR_EX
0083                        L7:
0083 B90100                         MOV     CX,1
0086 BA8000                         MOV     DX,80H

0089 B80012                         MOV     AX,1200H                ; CONTROLLER DIAGNOSTICS
008C CD13                           INT     13H
008E 7303                           JNC     P7
0090 E9AF00                         JMP     ERROR_EX
0093                        P7:
0093 B80014                         MOV     AX,1400H                ; CONTROLLER DIAGNOSTICS
0096 CD13                           INT     13H
0098 7303                           JNC     P9
009A E9A500                         JMP     ERROR_EX
009D                        P9:
009D C7066C000000                   MOV     TIMER_LOW,0             ; ZERO TIMER
00A3 A17200                         MOV     AX,RESET_FLAG
00A6 3D3412                         CMP     AX,1234H                ; KEYBOARD RESET
00A9 7506                           JNE     P8
00AB C7066C009A01                   MOV     TIMER_LOW,410D          ; SKIP WAIT ON RESET
00B1                        P8:
00B1 E421                           IN      AL,021H                 ; TIMER
00B3 24FE                           AND     AL,0FEH                 ; ENABLE TIMER
00B5 E621                           OUT     021H,AL                 ; START TIMER
00B7                        P4:
00B7 E8B400                         CALL    HD_RESET_1              ; RESET CONTROLLER
00BA 7207                           JC      P10
00BC B80010                         MOV     AX,1000H                ; READY
00BF CD13                           INT     13H
00C1 730B                           JNC     P2
00C3                        P10:
00C3 A16C00                         MOV     AX,TIMER_LOW
00C6 3DBE01                         CMP     AX,446D                 ; 25 SECONDS
00C9 72EC                           JB      P4
00CB EB7590                         JMP     ERROR_EX
00CE                        P2:
00CE B90100                         MOV     CX,1
00D1 BA8000                         MOV     DX,80H

00D4 B80011                         MOV     AX,1100H                ; RECALIBRATE
00D7 CD13                           INT     13H
00D9 7267                           JC      ERROR_EX

00DB B80009                         MOV     AX,0900H                ; SET DRIVE PARAMETERS
00DE CD13                           INT     13H
00E0 7260                           JC      ERROR_EX

00E2 B800C8                         MOV     AX,0C800H               ; DMA TO BUFFER
```

```
LOC OBJ                 SOURCE

00E5 8EC0                       MOV     ES,AX                   ; SET SEGMENT
00E7 2BDB                       SUB     BX,BX
00E9 B8000F                     MOV     AX,0F00H                ; WRITE SECTOR BUFFER
00EC CD13                       INT     13H
00EE 7252                       JC      ERROR_EX

00F0 FE067500                   INC     HF_NUM                  ; DRIVE ZERO RESPONDED

00F4 BA1302                     MOV     DX,213H                 ; EXPANSION BOX
00F7 B000                       MOV     AL,0
00F9 EE                         OUT     DX,AL                   ; TURN BOX OFF
00FA BA2103                     MOV     DX,321H                 ; TEST IF CONTROLLER
00FD EC                         IN      AL,DX                   ; ... IS IN THE SYSTEM UNIT
00FE 240F                       AND     AL,0FH
0100 3C0F                       CMP     AL,0FH
0102 7406                       JE      BOX_ON
0104 C7066C00A401               MOV     TIMER_LOW,420D          ; CONTROLLER IS IN SYSTEM UNIT
010A                    BOX_ON:
010A BA1302                     MOV     DX,213H                 ; EXPANSION BOX
010D B0FF                       MOV     AL,0FFH
010F EE                         OUT     DX,AL                   ; TURN BOX ON

0110 B90100                     MOV     CX,1                    ; ATTEMPT NEXT DRIVES
0113 BA8100                     MOV     DX,081H
0116                    P3:
0116 2BC0                       SUB     AX,AX                   ; RESET
0118 CD13                       INT     13H
011A 7240                       JC      POD_DONE
011C B80011                     MOV     AX,01100H               ; RECAL
011F CD13                       INT     13H
0121 730B                       JNC     P5
0123 A16C00                     MOV     AX,TIMER_LOW
0126 3DBE01                     CMP     AX,446D                 ; 25 SECONDS
0129 72EB                       JB      P3
012B EB2F90                     JMP     POD_DONE
012E                    P5:
012E B80009                     MOV     AX,0900H                ; INITIALIZE CHARACTERISTICS
0131 CD13                       INT     13H
0133 7227                       JC      POD_DONE
0135 FE067500                   INC     HF_NUM                  ; TALLY ANOTHER DRIVE
0139 81FA8100                   CMP     DX,(80H + S_MAX_FILE - 1)
013D 731D                       JAE     POD_DONE
013F 42                         INC     DX
0140 EBD4                       JMP     P3

                        ;----- POD ERROR

0142                    ERROR_EX:
0142 BD0F00                     MOV     BP,0FH                  ; POD ERROR FLAG
0145 2BC0                       SUB     AX,AX
0147 8BF0                       MOV     SI,AX
0149 B9060090                   MOV     CX,F17L                 ; MESSAGE CHARACTER COUNT
014D B700                       MOV     BH,0                    ; PAGE ZERO
014F                    OUT_CH:
014F 2E8A846801                 MOV     AL,CS:F17[SI]           ; GET BYTE
0154 B40E                       MOV     AH,14D                  ; VIDEO OUT
0156 CD10                       INT     10H                     ; DISPLAY CHARACTER
0158 46                         INC     SI                      ; NEXT CHAR
0159 E2F4                       LOOP    OUT_CH                  ; DO MORE
015B F9                         STC
015C                    POD_DONE:
015C FA                         CLI
015D E421                       IN      AL,021H                 ; BE SURE TIMER IS DISABLED
015F 0C01                       OR      AL,01H
0161 E621                       OUT     021H,AL
0163 FB                         STI
0164 E8A500                     CALL    DSBL
0167 CB                         RET

0168 31373031           F17     DB      '1701',0DH,0AH
016C 0D
016D 0A
  0006                  F17L    EQU     $-F17

016E                    HD_RESET_1      PROC    NEAR
016E 51                         PUSH    CX                      ; SAVE REGISTER
016F 52                         PUSH    DX
```

```
LOC OBJ                 SOURCE

0170 F8                         CLC                             ; CLEAR CARRY
0171 B90001                     MOV     CX,0100H                ; RETRY COUNT
0174                    L6:
0174 E80706                     CALL    PORT_1
0177 EE                         OUT     DX,AL                   ; RESET CARD
0178 E80306                     CALL    PORT_1
017B EC                         IN      AL,DX                   ; CHECK STATUS
017C 2402                       AND     AL,2                    ; ERROR BIT
017E 7403                       JZ      R3
0180 E2F2                       LOOP    L6
0182 F9                         STC
0183                    R3:
0183 5A                         POP     DX                      ; RESTORE REGISTER
0184 59                         POP     CX
0185 C3                         RET
                        HD_RESET_1      ENDP

                        DISK_SETUP      ENDP

                        ;----- INT 19 ---------------------------------------------------
                        ;                                                              :
                        ; INTERRUPT 19 BOOT STRAP LOADER                               :
                        ;                                                              :
                        ;  -  THE FIXED DISK BIOS REPLACES THE INTERRUPT 19            :
                        ;     BOOT STRAP VECTOR WITH A POINTER TO THIS BOOT ROUTINE    :
                        ;  -  RESET THE DEFAULT DISK AND DISKETTE PARAMETER VECTORS    :
                        ;  -  THE BOOT BLOCK TO BE READ IN WILL BE ATTEMPTED FROM      :
                        ;     CYLINDER 0 SECTOR 1 OF THE DEVICE.                       :
                        ;  -  THE BOOTSTRAP SEQUENCE IS:                               :
                        ;      > ATTEMPT TO LOAD FROM THE DISKETTE INTO THE BOOT       :
                        ;        LOCATION (0000:7C00) AND TRANSFER CONTROL THERE       :
                        ;      > IF THE DISKETTE FAILS THE FIXED DISK IS TRIED FOR A   :
                        ;        VALID BOOTSTRAP BLOCK. A VALID BOOT BLOCK ON THE      :
                        ;        FIXED DISK CONSISTS OF THE BYTES  055H 0AAH  AS THE   :
                        ;        LAST TWO BYTES OF THE BLOCK                           :
                        ;      > IF THE ABOVE FAILS CONTROL IS PASSED TO RESIDENT BASIC :
                        ;                                                              :
                        ;---------------------------------------------------------------

0186                    BOOT_STRAP:
                                ASSUME  DS:DUMMY,ES:DUMMY
0186 2BC0                       SUB     AX,AX
0188 8ED8                       MOV     DS,AX                   ; ESTABLISH SEGMENT

                        ;----- RESET PARAMETER VECTORS

018A FA                         CLI
018B C7060401E703               MOV     WORD PTR HF_TBL_VEC, OFFSET FD_TBL
0191 8C0E0601                   MOV     WORD PTR HF_TBL_VEC+2, CS
0195 C70678000102               MOV     WORD PTR DISKETTE_PARM, OFFSET DISKETTE_TBL
019B 8C0E7A00                   MOV     WORD PTR DISKETTE_PARM+2, CS
019F FB                         STI

                        ;----- ATTEMPT BOOTSTRAP FROM DISKETTE

01A0 B90300                     MOV     CX,3                    ; SET RETRY COUNT
01A3                    H1:                                     ; IPL_SYSTEM
01A3 51                         PUSH    CX                      ; SAVE RETRY COUNT
01A4 2BD2                       SUB     DX,DX                   ; DRIVE ZERO
01A6 2BC0                       SUB     AX,AX                   ; RESET THE DISKETTE
01A8 CD13                       INT     13H                     ; FILE IO CALL
01AA 720F                       JC      H2                      ; IF ERROR, TRY AGAIN
01AC B80102                     MOV     AX,0201H                ; READ IN THE SINGLE SECTOR

01AF 2BD2                       SUB     DX,DX
01B1 8EC2                       MOV     ES,DX                   ; ESTABLISH SEGMENT
01B3 BB007C                     MOV     BX,OFFSET BOOT_LOCN

01B6 B90100                     MOV     CX,1                    ; SECTOR 1, TRACK 0
01B9 CD13                       INT     13H                     ; FILE IO CALL
01BB 59                 H2:     POP     CX                      ; RECOVER RETRY COUNT
01BC 730A                       JNC     H4                      ; CF SET BY UNSUCCESSFUL READ
01BE 80FC80                     CMP     AH,80H                  ; IF TIME OUT, NO RETRY
01C1 740A                       JZ      H5                      ; TRY FIXED DISK
01C3 E2DE                       LOOP    H1                      ; DO IT FOR RETRY TIMES
01C5 EB0690                     JMP     H5                      ; UNABLE TO IPL FROM THE DISKETTE
01C8                    H4:                                     ; IPL WAS SUCCESSFUL
```

```
LOC OBJ                 LINE    SOURCE

01C8 EA007C0000                         JMP     BOOT_LOCN

                                ;----- ATTEMPT BOOTSTRAP FROM FIXED DISK

01CD                            H5:
01CD 2BC0                               SUB     AX,AX                   ; RESET DISKETTE
01CF 2BD2                               SUB     DX,DX
01D1 CD13                               INT     13H
01D3 B90300                             MOV     CX,3                    ; SET RETRY COUNT
01D6                            H6:                                     ; IPL_SYSTEM
01D6 51                                 PUSH    CX                      ; SAVE RETRY COUNT
01D7 BA8000                             MOV     DX,0080H                ; FIXED DISK ZERO
01DA 2BC0                               SUB     AX,AX                   ; RESET THE FIXED DISK
01DC CD13                               INT     13H                     ; FILE IO CALL
01DE 7212                               JC      H7                      ; IF ERROR, TRY AGAIN
01E0 B80102                             MOV     AX,0201H                ; READ IN THE SINGLE SECTOR
01E3 2BDB                               SUB     BX,BX
01E5 8EC3                               MOV     ES,BX
01E7 BB007C                             MOV     BX,OFFSET BOOT_LOCN     ; TO THE BOOT LOCATION
01EA BA8000                             MOV     DX,80H                  ; DRIVE NUMBER
01ED B90100                             MOV     CX,1                    ; SECTOR 1, TRACK 0
01F0 CD13                               INT     13H                     ; FILE IO CALL
01F2 59                         H7:     POP     CX                      ; RECOVER RETRY COUNT
01F3 7208                               JC      H8
01F5 A1FE7D                             MOV     AX,WORD PTR BOOT_LOCN+510D
01F8 3D55AA                             CMP     AX,0AA55H               ; TEST FOR GENERIC BOOT BLOCK
01FB 74CB                               JZ      H4
01FD                            H8:
01FD E2D7                               LOOP    H6                      ; DO IT FOR RETRY TIMES

                                ;------ UNABLE TO IPL FROM THE DISKETTE OR FIXED DISK

01FF CD18                               INT     18H                     ; RESIDENT BASIC

0201                            DISKETTE_TBL:

0201 CF                                 DB      11001111B               ; SRT=C, HD UNLOAD=OF - 1ST SPEC BYTE
0202 02                                 DB      2                       ; HD LOAD=1, MODE=DMA - 2ND SPEC BYTE
0203 25                                 DB      25H                     ; WAIT AFTER OPN TIL MOTOR OFF
0204 02                                 DB      2                       ; 512 BYTES PER SECTOR
0205 08                                 DB      8                       ; EOT (LAST SECTOR ON TRACK)
0206 2A                                 DB      02AH                    ; GAP LENGTH
0207 FF                                 DB      0FFH                    ; DTL
0208 50                                 DB      050H                    ; GAP LENGTH FOR FORMAT
0209 F6                                 DB      0F6H                    ; FILL BYTE FOR FORMAT
020A 19                                 DB      25                      ; HEAD SETTLE TIME (MILLISECONDS)
020B 04                                 DB      4                       ; MOTOR START TIME (1/8 SECOND)

                                ;----- MAKE SURE THAT ALL HOUSEKEEPING IS DONE BEFORE EXIT

020C                            DSBL    PROC    NEAR
                                        ASSUME  DS:DATA
020C 1E                                 PUSH    DS                      ; SAVE SEGMENT
020D B84000                             MOV     AX,DATA
0210 8ED8                               MOV     DS,AX

0212 8A267700                           MOV     AH,PORT_OFF
0216 50                                 PUSH    AX                      ; SAVE OFFSET

0217 C606770000                         MOV     PORT_OFF,0H
021C E86905                             CALL    PORT_3
021F 2AC0                               SUB     AL,AL
0221 EE                                 OUT     DX,AL                   ; RESET INT/DMA MASK
0222 C606770004                         MOV     PORT_OFF,4H
0227 E85E05                             CALL    PORT_3
022A 2AC0                               SUB     AL,AL
022C EE                                 OUT     DX,AL                   ; RESET INT/DMA MASK
022D C606770008                         MOV     PORT_OFF,8H
0232 E85305                             CALL    PORT_3
0235 2AC0                               SUB     AL,AL
0237 EE                                 OUT     DX,AL                   ; RESET INT/DMA MASK
0238 C60677000C                         MOV     PORT_OFF,0CH
023D E84805                             CALL    PORT_3
0240 2AC0                               SUB     AL,AL
0242 EE                                 OUT     DX,AL                   ; RESET INT/DMA MASK
0243 B007                               MOV     AL,07H
0245 E60A                               OUT     DMA+10,AL               ; SET DMA MODE TO DISABLE
```

```
LOC OBJ                     SOURCE

0247 FA                             CLI                             ; DISABLE INTERRUPTS
0248 E421                           IN      AL,021H
024A 0C20                           OR      AL,020H
024C E621                           OUT     021H,AL                 ; DISABLE INTERRUPT 5
024E FB                             STI                             ; ENABLE INTERRUPTS
024F 58                             POP     AX                      ; RESTORE OFFSET
0250 88267700                       MOV     PORT_OFF,AH
0254 1F                             POP     DS                      ; RESTORE SEGMENT
0255 C3                             RET
                            DSBL    ENDP

                            ;----------------------------------------
                            ;       FIXED DISK BIOS ENTRY POINT     :
                            ;----------------------------------------

0256                        DISK_IO PROC    FAR
                                    ASSUME  DS:NOTHING,ES:NOTHING
0256 80FA80                         CMP     DL,80H                  ; TEST FOR FIXED DISK DRIVE
0259 7305                           JAE     HARD_DISK               ; YES, HANDLE HERE
025B CD40                           INT     40H                     ; DISKETTE HANDLER
025D                        RET_2:
025D CA0200                         RET     2                       ; BACK TO CALLER
0260                        HARD_DISK:
                                    ASSUME  DS:DATA
0260 FB                             STI                             ; ENABLE INTERRUPTS
0261 0AE4                           OR      AH,AH
0263 7509                           JNZ     A3
0265 CD40                           INT     40H                     ; RESET NEC WHEN AH=0
0267 2AE4                           SUB     AH,AH
0269 80FA81                         CMP     DL,(80H + S_MAX_FILE - 1)
026C 77EF                           JA      RET_2
026E                        A3:
026E 80FC08                         CMP     AH,08                   ; GET PARAMETERS IS A SPECIAL CASE
0271 7503                           JNZ     A2
0273 E91A01                         JMP     GET_PARM_N
0276                        A2:
0276 53                             PUSH    BX                      ; SAVE REGISTERS DURING OPERATION
0277 51                             PUSH    CX
0278 52                             PUSH    DX
0279 1E                             PUSH    DS
027A 06                             PUSH    ES
027B 56                             PUSH    SI
027C 57                             PUSH    DI

027D E86A00                         CALL    DISK_IO_CONT            ; PERFORM THE OPERATION

0280 50                             PUSH    AX
0281 E888FF                         CALL    DSBL                    ; BE SURE DISABLES OCCURRED
0284 B84000                         MOV     AX,DATA
0287 8ED8                           MOV     DS,AX                   ; ESTABLISH SEGMENT
0289 58                             POP     AX
028A 8A267400                       MOV     AH,DISK_STATUS          ; GET STATUS FROM OPERATION
028E 80FC01                         CMP     AH,1                    ; SET THE CARRY FLAG TO INDICATE
0291 F5                             CMC                             ;  SUCCESS OR FAILURE
0292 5F                             POP     DI                      ; RESTORE REGISTERS
0293 5E                             POP     SI
0294 07                             POP     ES
0295 1F                             POP     DS
0296 5A                             POP     DX
0297 59                             POP     CX
0298 5B                             POP     BX
0299 CA0200                         RET     2                       ; THROW AWAY SAVED FLAGS
                            DISK_IO ENDP

029C                        M1      LABEL   WORD                    ; FUNCTION TRANSFER TABLE
029C 3803                           DW      DISK_RESET              ; 000H
029E 4D03                           DW      RETURN_STATUS           ; 001H
02A0 5603                           DW      DISK_READ               ; 002H
02A2 6003                           DW      DISK_WRITE              ; 003H
02A4 6A03                           DW      DISK_VERF               ; 004H
02A6 7203                           DW      FMT_TRK                 ; 005H
02A8 7903                           DW      FMT_BAD                 ; 006H
02AA 8003                           DW      FMT_DRV                 ; 007H
02AC 3003                           DW      BAD_COMMAND             ; 008H
02AE 2704                           DW      INIT_DRV                ; 009H
02B0 CF04                           DW      RD_LONG                 ; 00AH
02B2 DD04                           DW      WR_LONG                 ; 00BH
```

```
LOC OBJ            SOURCE

02B4 F204                  DW      DISK_SEEK              ; 00CH
02B6 3803                  DW      DISK_RESET             ; 00DH
02B8 F904                  DW      RD_BUFF                ; 00EH
02BA 0705                  DW      WR_BUFF                ; 00FH
02BC 1505                  DW      TST_RDY                ; 010H
02BE 1C05                  DW      HDISK_RECAL            ; 011H
02C0 2305                  DW      RAM_DIAG               ; 012H
02C2 2A05                  DW      CHK_DRV                ; 013H
02C4 3105                  DW      CNTLR_DIAG             ; 014H
  002A             M1L     EQU     $-M1

02C6               SETUP_A PROC    NEAR

02C6 C606740000            MOV     DISK_STATUS,0          ; RESET THE STATUS INDICATOR
02CB 51                    PUSH    CX                     ; SAVE CX

                   ;----- CALCULATE THE PORT OFFSET

02CC 8AEA                  MOV     CH,DL                  ; SAVE DL
02CE 80CA01                OR      DL,1
02D1 FECA                  DEC     DL
02D3 D0E2                  SHL     DL,1                   ; GENERATE OFFSET
02D5 88167700              MOV     PORT_OFF,DL            ; STORE OFFSET
02D9 8AD5                  MOV     DL,CH                  ; RESTORE DL
02DB 80E201                AND     DL,1

02DE B105                  MOV     CL,5                   ; SHIFT COUNT
02E0 D2E2                  SHL     DL,CL                  ; DRIVE NUMBER (0,1)
02E2 0AD6                  OR      DL,DH                  ; HEAD NUMBER
02E4 88164300              MOV     CMD_BLOCK+1,DL
02E8 59                    POP     CX
02E9 C3                    RET
                   SETUP_A ENDP

02EA               DISK_IO_CONT    PROC    NEAR
02EA 50                    PUSH    AX
02EB B84000                MOV     AX,DATA
02EE 8ED8                  MOV     DS,AX                  ; ESTABLISH SEGMENT
02F0 58                    POP     AX
02F1 80FC01                CMP     AH,01H                 ; RETURN STATUS
02F4 7503                  JNZ     A4
02F6 EB5590                JMP     RETURN_STATUS
02F9               A4:
02F9 80EA80                SUB     DL,80H                 ; CONVERT DRIVE NUMBER TO 0 BASED RANGE
02FC 80FA08                CMP     DL,MAX_FILE            ; LEGAL DRIVE TEST
02FF 732F                  JAE     BAD_COMMAND

0301 E8C2FF                CALL    SETUP_A

                   ;----- SET UP COMMAND BLOCK

0304 FEC9                  DEC     CL                     ; SECTORS 0-16 FOR CONTROLLER
0306 C606420000            MOV     CMD_BLOCK+0,0
030B 880E4400              MOV     CMD_BLOCK+2,CL         ; SECTOR AND HIGH 2 BITS CYLINDER
030F 882E4500              MOV     CMD_BLOCK+3,CH         ; CYLINDER
0313 A24600                MOV     CMD_BLOCK+4,AL         ; INTERLEAVE / BLOCK COUNT
0316 A07600                MOV     AL,CONTROL_BYTE        ; CONTROL BYTE (STEP OPTION)
0319 A24700                MOV     CMD_BLOCK+5,AL
031C 50                    PUSH    AX                     ; SAVE AX
031D 8AC4                  MOV     AL,AH                  ; GET INTO LOW BYTE
031F 32E4                  XOR     AH,AH                  ; ZERO HIGH BYTE
0321 D1E0                  SAL     AX,1                   ; *2 FOR TABLE LOOKUP
0323 8BF0                  MOV     SI,AX                  ; PUT INTO SI FOR BRANCH
0325 3D2A00                CMP     AX,M1L                 ; TEST WITHIN RANGE
0328 58                    POP     AX                     ; RESTORE AX
0329 7305                  JNB     BAD_COMMAND
032B 2EFFA49C02            JMP     WORD PTR CS:[SI + OFFSET M1]
0330               BAD_COMMAND:
0330 C606740001            MOV     DISK_STATUS,BAD_CMD    ; COMMAND ERROR
0335 B000                  MOV     AL,0
0337 C3                    RET
                   DISK_IO_CONT    ENDP

                   ;----------------------------------------------
                   ;       RESET THE DISK SYSTEM  (AH = 000H)    :
                   ;----------------------------------------------

```

```
LOC OBJ              SOURCE

0338                 DISK_RESET      PROC    NEAR
0338 E84304                  CALL    PORT_1                  ; RESET PORT
033B EE                      OUT     DX,AL                   ; ISSUE RESET
033C E83F04                  CALL    PORT_1                  ; CONTROLLER HARDWARE STATUS
033F EC                      IN      AL,DX                   ; GET STATUS
0340 2402                    AND     AL,2                    ; ERROR BIT
0342 7406                    JZ      DR1
0344 C606740005              MOV     DISK_STATUS,BAD_RESET
0349 C3                      RET
034A                 DR1:
034A E9DA00                  JMP     INIT_DRV                ; SET THE DRIVE PARAMETERS
                     DISK_RESET      ENDP

                     ;-----------------------------------------------
                     ;       DISK STATUS ROUTINE   (AH = 001H)       :
                     ;-----------------------------------------------

034D                 RETURN_STATUS   PROC    NEAR
034D A07400                  MOV     AL,DISK_STATUS          ; OBTAIN PREVIOUS STATUS
0350 C606740000              MOV     DISK_STATUS,0           ; RESET STATUS
0355 C3                      RET
                     RETURN_STATUS   ENDP

                     ;-----------------------------------------------
                     ;       DISK READ ROUTINE   (AH = 002H)         :
                     ;-----------------------------------------------

0356                 DISK_READ       PROC    NEAR
0356 B047                    MOV     AL,DMA_READ             ; MODE BYTE FOR DMA READ
0358 C606420008              MOV     CMD_BLOCK+0,READ_CMD
035D E9E501                  JMP     DMA_OPN
                     DISK_READ       ENDP

                     ;-----------------------------------------------
                     ;       DISK WRITE ROUTINE   (AH = 003H)        :
                     ;-----------------------------------------------

0360                 DISK_WRITE      PROC    NEAR
0360 B04B                    MOV     AL,DMA_WRITE            ; MODE BYTE FOR DMA WRITE
0362 C60642000A              MOV     CMD_BLOCK+0,WRITE_CMD
0367 E9DB01                  JMP     DMA_OPN
                     DISK_WRITE      ENDP

                     ;-----------------------------------------------
                     ;       DISK VERIFY   (AH = 004H)               :
                     ;-----------------------------------------------

036A                 DISK_VERF       PROC    NEAR
036A C606420005              MOV     CMD_BLOCK+0,CHK_TRK_CMD
036F E9C401                  JMP     NDMA_OPN
                     DISK_VERF       ENDP

                     ;-----------------------------------------------
                     ;       FORMATTING   (AH = 005H 006H 007H)      :
                     ;-----------------------------------------------

0372                 FMT_TRK PROC    NEAR                    ; FORMAT TRACK  (AH = 005H)
0372 C606420006              MOV     CMD_BLOCK,FMTTRK_CMD
0377 EB0C                    JMP     SHORT   FMT_CONT
                     FMT_TRK ENDP

0379                 FMT_BAD PROC    NEAR                    ; FORMAT BAD TRACK  (AH = 006H)
0379 C606420007              MOV     CMD_BLOCK,FMTBAD_CMD
037E EB05                    JMP     SHORT   FMT_CONT
                     FMT_BAD ENDP

0380                 FMT_DRV PROC    NEAR                    ; FORMAT DRIVE  (AH = 007H)
0380 C606420004              MOV     CMD_BLOCK,FMTDRV_CMD
                     FMT_DRV ENDP

0385                 FMT_CONT:
0385 A04400                  MOV     AL,CMD_BLOCK+2          ; ZERO OUT SECTOR FIELD
0388 24C0                    AND     AL,11000000B
038A A24400                  MOV     CMD_BLOCK+2,AL
038D E9A601                  JMP     NDMA_OPN

```

```
                        ;-----------------------------------------------
                        ;       GET PARAMETERS  (AH = 8)               :
                        ;-----------------------------------------------

0390                    GET_PARM_N      LABEL   NEAR
0390                    GET_PARM        PROC    FAR             ; GET DRIVE PARAMETERS
0390 1E                         PUSH    DS                      ; SAVE REGISTERS
0391 06                         PUSH    ES
0392 53                         PUSH    BX

                                ASSUME  DS:DUMMY
0393 2BC0                       SUB     AX,AX                   ; ESTABLISH ADDRESSING
0395 8ED8                       MOV     DS,AX
0397 C41E0401                   LES     BX,HF_TBL_VEC
                                ASSUME  DS:DATA
039B B84000                     MOV     AX,DATA
039E 8ED8                       MOV     DS,AX                   ; ESTABLISH SEGMENT

03A0 80EA80                     SUB     DL,80H
03A3 80FA08                     CMP     DL,MAX_FILE             ; TEST WITHIN RANGE
03A6 732F                       JAE     G4

03A8 E81BFF                     CALL    SETUP_A

03AB E8DF03                     CALL    SW2_OFFS
03AE 7227                       JC      G4
03B0 03D8                       ADD     BX,AX

03B2 268B07                     MOV     AX,ES:[BX]              ; MAX NUMBER OF CYLINDERS
03B5 2D0200                     SUB     AX,2                    ; ADJUST FOR 0-N
                                                                ; AND RESERVE LAST TRACK
03B8 8AE8                       MOV     CH,AL
03BA 250003                     AND     AX,0300H                ; HIGH TWO BITS OF CYL
03BD D1E8                       SHR     AX,1
03BF D1E8                       SHR     AX,1
03C1 0C11                       OR      AL,011H                 ; SECTORS
03C3 8AC8                       MOV     CL,AL

03C5 268A7702                   MOV     DH,ES:[BX][2]           ; HEADS
03C9 FECE                       DEC     DH                      ; 0-N RANGE
03CB 8A167500                   MOV     DL,HF_NUM               ; DRIVE COUNT
03CF 2BC0                       SUB     AX,AX
03D1                    G5:
03D1 5B                         POP     BX                      ; RESTORE REGISTERS
03D2 07                         POP     ES
03D3 1F                         POP     DS
03D4 CA0200                     RET     2
03D7                    G4:
03D7 C606740007                 MOV     DISK_STATUS,INIT_FAIL   ; OPERATION FAILED
03DC B407                       MOV     AH,INIT_FAIL
03DE 2AC0                       SUB     AL,AL
03E0 2BD2                       SUB     DX,DX
03E2 2BC9                       SUB     CX,CX
03E4 F9                         STC                             ; SET ERROR FLAG
03E5 EBEA                       JMP     G5
                        GET_PARM        ENDP

                        ;----------------------------------------------------
                        ; INITIALIZE DRIVE CHARACTERISTICS                   :
                        ;                                                    :
                        ; FIXED DISK PARAMETER TABLE                         :
                        ;                                                    :
                        ;  -  THE TABLE IS COMPOSED OF A BLOCK DEFINED AS:   :
                        ;                                                    :
                        ;       (1 WORD) - MAXIMUM NUMBER OF CYLINDERS       :
                        ;       (1 BYTE) - MAXIMUM NUMBER OF HEADS           :
                        ;       (1 WORD) - STARTING REDUCED WRITE CURRENT CYL  :
                        ;       (1 WORD) - STARTING WRITE PRECOMPENSATION CYL  :
                        ;       (1 BYTE) - MAXIMUM ECC DATA BURST LENGTH     :
                        ;       (1 BYTE) - CONTROL BYTE (DRIVE STEP OPTION)  :
                        ;                  BIT    7 DISABLE DISK-ACCESS RETRIES :
                        ;                  BIT    6 DISABLE ECC RETRIES      :
                        ;                  BITS 5-3 ZERO                     :
                        ;                  BITS 2-0 DRIVE OPTION             :
                        ;       (1 BYTE) - STANDARD TIME OUT VALUE (SEE BELOW)  :
                        ;       (1 BYTE) - TIME OUT VALUE FOR FORMAT DRIVE   :
                        ;       (1 BYTE) - TIME OUT VALUE FOR CHECK DRIVE    :
                        ;       (4 BYTES)                                    :
```

```
                    ;                 - RESERVED FOR FUTURE USE                :
                    ;                                                          :
                    ;          - TO DYNAMICALLY DEFINE A SET OF PARAMETERS     :
                    ;            BUILD A TABLE OF VALUES AND PLACE THE         :
                    ;            CORRESPONDING VECTOR INTO INTERRUPT 41.       :
                    ;                                                          :
                    ;       NOTE:                                              :
                    ;               THE DEFAULT TABLE IS VECTORED IN FOR       :
                    ;               AN INTERRUPT 19H (BOOTSTRAP)               :
                    ;                                                          :
                    ;                                                          :
                    ; ON THE CARD SWITCH SETTINGS                              :
                    ;                                                          :
                    ;                 DRIVE 0    DRIVE 1                       :
                    ;               ----------------------                     :
                    ;         ON    :          /         :                     :
                    ;               : -1-   -2- / -3-   -4- :                  :
                    ;         OFF   :          /         :                     :
                    ;               ----------------------                     :
                    ;                                                          :
                    ;                                                          :
                    ;       TRANSLATION TABLE                                  :
                    ;                                                          :
                    ;       1/3  :  2/4  :  TABLE ENTRY                        :
                    ;       --------------------------                         :
                    ;        ON  :   ON  :      0                              :
                    ;        ON  :  OFF  :      1                              :
                    ;       OFF  :   ON  :      2                              :
                    ;       OFF  :  OFF  :      3                              :
                    ;                                                          :
                    ;-----------------------------------------------------------

03E7                FD_TBL:

                    ;----- DRIVE TYPE 00

03E7 3201                   DW      0306D
03E9 02                     DB      02D
03EA 3201                   DW      0306D
03EC 0000                   DW      0000D
03EE 0B                     DB      0BH
03EF 00                     DB      00H
03F0 0C                     DB      0CH                     ; STANDARD
03F1 B4                     DB      0B4H                    ; FORMAT DRIVE
03F2 28                     DB      028H                    ; CHECK DRIVE
03F3 00000000               DB      0,0,0,0

                    ;----- DRIVE TYPE 01

03F7 7701                   DW      0375D
03F9 08                     DB      08D
03FA 7701                   DW      0375D
03FC 0000                   DW      0000D
03FE 0B                     DB      0BH
03FF 05                     DB      05H
0400 0C                     DB      0CH                     ; STANDARD
0401 B4                     DB      0B4H                    ; FORMAT DRIVE
0402 28                     DB      028H                    ; CHECK DRIVE
0403 00000000               DB      0,0,0,0

                    ;----- DRIVE TYPE 02

0407 3201                   DW      0306D
0409 06                     DB      06D
040A 8000                   DW      0128D
040C 0001                   DW      0256D
040E 0B                     DB      0BH
040F 05                     DB      05H
0410 0C                     DB      0CH                     ; STANDARD
0411 B4                     DB      0B4H                    ; FORMAT DRIVE
0412 28                     DB      028H                    ; CHECK DRIVE
0413 00000000               DB      0,0,0,0

                    ;----- DRIVE TYPE 03

0417 3201                   DW      0306D
0419 04                     DB      04D
```

```
LOC OBJ                 LINE    SOURCE

041A 3201               919             DW      0306D
041C 0000               920             DW      0000D
041E 0B                 921             DB      0BH
041F 05                 922             DB      05H
0420 0C                 923             DB      0CH                     ; STANDARD
0421 B4                 924             DB      0B4H                    ; FORMAT DRIVE
0422 28                 925             DB      028H                    ; CHECK DRIVE
0423 00000000           926             DB      0,0,0,0
                        927
0427                    928     INIT_DRV        PROC    NEAR
                        929
                        930     ;----- DO DRIVE ZERO
                        931
0427 C60642000C         932             MOV     CMD_BLOCK+0,INIT_DRV_CMD
042C C606430000         933             MOV     CMD_BLOCK+1,0
0431 E81000             934             CALL    INIT_DRV_R
0434 720D               935             JC      INIT_DRV_OUT
                        936
                        937     ;----- DO DRIVE ONE
                        938
0436 C60642000C         939             MOV     CMD_BLOCK+0,INIT_DRV_CMD
043B C606430020         940             MOV     CMD_BLOCK+1,00100000B
0440 E80100             941             CALL    INIT_DRV_R
0443                    942     INIT_DRV_OUT:
0443 C3                 943             RET
                        944     INIT_DRV        ENDP
                        945
0444                    946     INIT_DRV_R      PROC    NEAR
                        947             ASSUME  ES:CODE
0444 2AC0               948             SUB     AL,AL
0446 E81901             949             CALL    COMMAND                 ; ISSUE THE COMMAND
0449 7301               950             JNC     B1
044B C3                 951             RET
044C                    952     B1:
044C 1E                 953             PUSH    DS                      ; SAVE SEGMENT
                        954             ASSUME  DS:DUMMY
044D 2BC0               955             SUB     AX,AX
044F 8ED8               956             MOV     DS,AX                   ; ESTABLISH SEGMENT
0451 C41E0401           957             LES     BX,HF_TBL_VEC
0455 1F                 958             POP     DS                      ; RESTORE SEGMENT
                        959             ASSUME  DS:DATA
0456 E83403             960             CALL    SW2_OFFS
0459 7257               961             JC      B3
045B 03D8               962             ADD     BX,AX
                        963
                        964     ;----- SEND DRIVE PARAMETERS MOST SIGNIFICANT BYTE FIRST
                        965
045D BF0100             966             MOV     DI,1
0460 E85F00             967             CALL    INIT_DRV_S
0463 724D               968             JC      B3
                        969
0465 BF0000             970             MOV     DI,0
0468 E85700             971             CALL    INIT_DRV_S
046B 7245               972             JC      B3
                        973
046D BF0200             974             MOV     DI,2
0470 E84F00             975             CALL    INIT_DRV_S
0473 723D               976             JC      B3
                        977
0475 BF0400             978             MOV     DI,4
0478 E84700             979             CALL    INIT_DRV_S
047B 7235               980             JC      B3
                        981
047D BF0300             982             MOV     DI,3
0480 E83F00             983             CALL    INIT_DRV_S
0483 722D               984             JC      B3
                        985
0485 BF0600             986             MOV     DI,6
0488 E83700             987             CALL    INIT_DRV_S
048B 7225               988             JC      B3
                        989
048D BF0500             990             MOV     DI,5
0490 E82F00             991             CALL    INIT_DRV_S
0493 721D               992             JC      B3
                        993
0495 BF0700             994             MOV     DI,7
0498 E82700             995             CALL    INIT_DRV_S
```

```
LOC OBJ                 SOURCE

049B 7215                       JC      B3

049D BF0800                     MOV     DI,8                    ; DRIVE STEP OPTION
04A0 268A01                     MOV     AL,ES:[BX + DI]
04A3 A27600                     MOV     CONTROL_BYTE,AL

04A6 2BC9                       SUB     CX,CX
04A8                    B5:
04A8 E8D302                     CALL    PORT_1
04AB EC                         IN      AL,DX
04AC A802                       TEST    AL,R1_IOMODE            ; STATUS INPUT MODE
04AE 7509                       JNZ     B6
04B0 E2F6                       LOOP    B5
04B2                    B3:
04B2 C606740007                 MOV     DISK_STATUS,INIT_FAIL   ; OPERATION FAILED
04B7 F9                         STC
04B8 C3                         RET

04B9                    B6:
04B9 E8B502                     CALL    PORT_0
04BC EC                         IN      AL,DX
04BD 2402                       AND     AL,2                    ; MASK ERROR BIT
04BF 75F1                       JNZ     B3
04C1 C3                         RET
                                ASSUME  ES:NOTHING
                        INIT_DRV_R      ENDP

                        ;----- SEND THE BYTE OUT TO THE CONTROLLER

04C2                    INIT_DRV_S      PROC    NEAR
04C2 E8C501                     CALL    HD_WAIT_REQ
04C5 7207                       JC      D1
04C7 E8A702                     CALL    PORT_0
04CA 268A01                     MOV     AL,ES:[BX + DI]
04CD EE                         OUT     DX,AL
04CE                    D1:
04CE C3                         RET
                        INIT_DRV_S      ENDP

                        ;---------------------------------------
                        ;       READ LONG  (AH = 0AH)           :
                        ;---------------------------------------

04CF                    RD_LONG         PROC    NEAR
04CF E81900                     CALL    CHK_LONG
04D2 726B                       JC      G8
04D4 C6064200E5                 MOV     CMD_BLOCK+0,RD_LONG_CMD
04D9 B047                       MOV     AL,DMA_READ
04DB EB68                       JMP     SHORT   DMA_OPN
                        RD_LONG         ENDP

                        ;---------------------------------------
                        ;       WRITE LONG  (AH = 0BH)          :
                        ;---------------------------------------

04DD                    WR_LONG         PROC    NEAR
04DD E80B00                     CALL    CHK_LONG
04E0 725D                       JC      G8
04E2 C6064200E6                 MOV     CMD_BLOCK+0,WR_LONG_CMD
04E7 B04B                       MOV     AL,DMA_WRITE
04E9 EB5A                       JMP     SHORT   DMA_OPN
                        WR_LONG         ENDP

04EB                    CHK_LONG        PROC    NEAR
04EB A04600                     MOV     AL,CMD_BLOCK+4
04EE 3C80                       CMP     AL,080H
04F0 F5                         CMC
04F1 C3                         RET
                        CHK_LONG        ENDP

                        ;---------------------------------------
                        ;       SEEK   (AH = 0CH)               :
                        ;---------------------------------------

04F2                    DISK_SEEK       PROC    NEAR
04F2 C60642000B                 MOV     CMD_BLOCK,SEEK_CMD
04F7 EB3D                       JMP     SHORT   NDMA_OPN
```

```
LOC OBJ              SOURCE

                     DISK_SEEK       ENDP

                     ;----------------------------------------------
                     ;       READ SECTOR BUFFER    (AH = 0EH)        :
                     ;----------------------------------------------

04F9                 RD_BUFF PROC    NEAR
04F9 C60642000E              MOV     CMD_BLOCK+0,RD_BUFF_CMD
04FE C606460001              MOV     CMD_BLOCK+4,1           ; ONLY ONE BLOCK
0503 B047                    MOV     AL,DMA_READ
0505 EB3E                    JMP     SHORT   DMA_OPN
                     RD_BUFF ENDP

                     ;----------------------------------------------
                     ;       WRITE SECTOR BUFFER    (AH = 0FH)       :
                     ;----------------------------------------------

0507                 WR_BUFF PROC    NEAR
0507 C60642000F              MOV     CMD_BLOCK+0,WR_BUFF_CMD
050C C606460001              MOV     CMD_BLOCK+4,1           ; ONLY ONE BLOCK
0511 B04B                    MOV     AL,DMA_WRITE
0513 EB30                    JMP     SHORT   DMA_OPN
                     WR_BUFF ENDP

                     ;----------------------------------------------
                     ;       TEST DISK READY    (AH = 010H)          :
                     ;----------------------------------------------

0515                 TST_RDY PROC    NEAR
0515 C606420000              MOV     CMD_BLOCK+0,TST_RDY_CMD
051A EB1A                    JMP     SHORT   NDMA_OPN
                     TST_RDY ENDP

                     ;----------------------------------------------
                     ;       RECALIBRATE    (AH = 011H)              :
                     ;----------------------------------------------

051C                 HDISK_RECAL     PROC    NEAR
051C C606420001              MOV     CMD_BLOCK,RECAL_CMD
0521 EB13                    JMP     SHORT   NDMA_OPN
                     HDISK_RECAL     ENDP

                     ;------------------------------------------------------
                     ;       CONTROLLER RAM DIAGNOSTICS    (AH = 012H)       :
                     ;------------------------------------------------------

0523                 RAM_DIAG        PROC    NEAR
0523 C6064200E0              MOV     CMD_BLOCK+0,RAM_DIAG_CMD
0528 EB0C                    JMP     SHORT   NDMA_OPN
                     RAM_DIAG        ENDP

                     ;----------------------------------------------
                     ;       DRIVE DIAGNOSTICS    (AH = 013H)        :
                     ;----------------------------------------------

052A                 CHK_DRV PROC    NEAR
052A C6064200E3              MOV     CMD_BLOCK+0,CHK_DRV_CMD
052F EB05                    JMP     SHORT   NDMA_OPN
                     CHK_DRV ENDP

                     ;--------------------------------------------------------
                     ;      CONTROLLER INTERNAL DIAGNOSTICS    (AH = 014H)     :
                     ;--------------------------------------------------------

0531                 CNTLR_DIAG      PROC    NEAR
0531 C6064200E4              MOV     CMD_BLOCK+0,CNTLR_DIAG_CMD
                     CNTLR_DIAG      ENDP

                     ;------------------------------------------------------
                     ;                    SUPPORT ROUTINES                   :
                     ;------------------------------------------------------

0536                 NDMA_OPN:
0536 B002                    MOV     AL,02H
0538 E82700                  CALL    COMMAND                 ; ISSUE THE COMMAND
053B 7221                    JC      G11
053D EB16                    JMP     SHORT   G3
```

```
053F                  G8:
053F C606740009               MOV     DISK_STATUS,DMA_BOUNDARY
0544 C3                       RET
0545                  DMA_OPN:
0545 E85701                   CALL    DMA_SETUP               ; SET UP FOR DMA OPERATION
0548 72F5                     JC      G8
054A B003                     MOV     AL,03H
054C E81300                   CALL    COMMAND                 ; ISSUE THE COMMAND
054F 720D                     JC      G11
0551 B003                     MOV     AL,03H
0553 E60A                     OUT     DMA+10,AL               ; INITIALIZE THE DISK CHANNEL
0555                  G3:
0555 E421                     IN      AL,021H
0557 24DF                     AND     AL,0DFH
0559 E621                     OUT     021H,AL
055B E8AA01                   CALL    WAIT_INT
055E                  G11:
055E E83B00                   CALL    ERROR_CHK
0561 C3                       RET

                      ;-------------------------------------------------------
                      ; COMMAND                                               :
                      ;       THIS ROUTINE OUTPUTS THE COMMAND BLOCK          :
                      ; INPUT                                                 :
                      ;       AL = CONTROLLER DMA/INTERRUPT REGISTER MASK     :
                      ;                                                       :
                      ;-------------------------------------------------------

0562                  COMMAND PROC    NEAR
0562 BE4200                   MOV     SI,OFFSET CMD_BLOCK
0565 E81B02                   CALL    PORT_2
0568 EE                       OUT     DX,AL                   ; CONTROLLER SELECT PULSE
0569 E81C02                   CALL    PORT_3
056C EE                       OUT     DX,AL
056D 2BC9                     SUB     CX,CX                   ; WAIT COUNT
056F E80C02                   CALL    PORT_1
0572                  WAIT_BUSY:
0572 EC                       IN      AL,DX                   ; GET STATUS
0573 240F                     AND     AL,0FH
0575 3C0D                     CMP     AL,R1_BUSY OR R1_BUS OR R1_REQ
0577 7409                     JE      C1
0579 E2F7                     LOOP    WAIT_BUSY
057B C606740080               MOV     DISK_STATUS,TIME_OUT
0580 F9                       STC
0581 C3                       RET                             ; ERROR RETURN
0582                  C1:
0582 FC                       CLD
0583 B90600                   MOV     CX,6                    ; BYTE COUNT
0586                  CM3:
0586 E8E801                   CALL    PORT_0
0589 AC                       LODSB                           ; GET THE NEXT COMMAND BYTE
058A EE                       OUT     DX,AL                   ; OUT IT GOES
058B E2F9                     LOOP    CM3                     ; DO MORE

058D E8EE01                   CALL    PORT_1                  ; STATUS
0590 EC                       IN      AL,DX
0591 A801                     TEST    AL,R1_REQ
0593 7406                     JZ      CM7
0595 C606740020               MOV     DISK_STATUS,BAD_CNTLR
059A F9                       STC
059B                  CM7:
059B C3                       RET
                      COMMAND ENDP

                      ;---------------------------------------------
                      ;                   SENSE STATUS BYTES        :
                      ;                                             :
                      ; BYTE 0                                      :
                      ;       BIT     7   ADDRESS VALID, WHEN SET   :
                      ;       BIT     6   SPARE, SET TO ZERO        :
                      ;       BITS 5-4    ERROR TYPE                :
                      ;       BITS 3-0    ERROR CODE                :
                      ;                                             :
                      ; BYTE 1                                      :
                      ;       BITS 7-6    ZERO                      :
                      ;       BIT     5   DRIVE (0-1)               :
                      ;       BITS 4-0    HEAD NUMBER               :
```

```
                      ;                                                  :
                      ; BYTE 2                                           :
                      ;         BITS 7-5   CYLINDER HIGH                 :
                      ;         BITS 4-0   SECTOR NUMBER                 :
                      ;                                                  :
                      ; BYTE 3                                           :
                      ;         BITS 7-0   CYLINDER LOW                  :
                      ;                                                  :
                      ;--------------------------------------------------

059C                  ERROR_CHK         PROC    NEAR
                              ASSUME  ES:DATA
059C A07400                   MOV     AL,DISK_STATUS          ; CHECK IF THERE WAS AN ERROR
059F 0AC0                     OR      AL,AL
05A1 7501                     JNZ     G21
05A3 C3                       RET

                      ;----- PERFORM SENSE STATUS

05A4                  G21:
05A4 B84000                   MOV     AX,DATA
05A7 8EC0                     MOV     ES,AX                   ; ESTABLISH SEGMENT
05A9 2BC0                     SUB     AX,AX
05AB 8BF8                     MOV     DI,AX
05AD C606420003               MOV     CMD_BLOCK+0,SENSE_CMD
05B2 2AC0                     SUB     AL,AL
05B4 E8ABFF                   CALL    COMMAND                 ; ISSUE SENSE STATUS COMMAND
05B7 7223                     JC      SENSE_ABORT             ; CANNOT RECOVER
05B9 B90400                   MOV     CX,4
05BC                  G22:
05BC E8CB00                   CALL    HD_WAIT_REQ
05BF 7220                     JC      G24
05C1 E8AD01                   CALL    PORT_0
05C4 EC                       IN      AL,DX
05C5 26884542                 MOV     ES:HD_ERROR[DI],AL      ; STORE AWAY SENSE BYTES
05C9 47                       INC     DI
05CA E8B101                   CALL    PORT_1
05CD E2ED                     LOOP    G22
05CF E8B800                   CALL    HD_WAIT_REQ
05D2 720D                     JC      G24
05D4 E89A01                   CALL    PORT_0
05D7 EC                       IN      AL,DX
05D8 A802                     TEST    AL,2
05DA 740F                     JZ      STAT_ERR
05DC                  SENSE_ABORT:
05DC C6067400FF               MOV     DISK_STATUS,SENSE_FAIL
05E1                  G24:
05E1 F9                       STC
05E2 C3                       RET
                      ERROR_CHK         ENDP

05E3 1A06             T_0     DW        TYPE_0
05E5 2706             T_1     DW        TYPE_1
05E7 6A06             T_2     DW        TYPE_2
05E9 7706             T_3     DW        TYPE_3

05EB                  STAT_ERR:
05EB 268A1E4200               MOV     BL,ES:HD_ERROR          ; GET ERROR BYTE
05F0 8AC3                     MOV     AL,BL
05F2 240F                     AND     AL,0FH
05F4 80E330                   AND     BL,00110000B            ; ISOLATE TYPE
05F7 2AFF                     SUB     BH,BH
05F9 B103                     MOV     CL,3
05FB D3EB                     SHR     BX,CL                   ; ADJUST
05FD 2EFFA7E305               JMP     WORD PTR CS:[BX + OFFSET T_0]
                              ASSUME ES:NOTHING

0602                  TYPE0_TABLE       LABEL   BYTE
0602 00204020800020           DB        0,BAD_CNTLR,BAD_SEEK,BAD_CNTLR,TIME_OUT,0,BAD_CNTLR
0609 0040                     DB        0,BAD_SEEK
  0009                TYPE0_LEN         EQU     $-TYPE0_TABLE
060B                  TYPE1_TABLE       LABEL   BYTE
060B 1010020004               DB        BAD_ECC,BAD_ECC,BAD_ADDR_MARK,0,RECORD_NOT_FND
0610 400000110B               DB        BAD_SEEK,0,0,DATA_CORRECTED,BAD_TRACK
  000A                TYPE1_LEN         EQU     $-TYPE1_TABLE
0615                  TYPE2_TABLE       LABEL   BYTE
0615 0102                     DB        BAD_CMD,BAD_ADDR_MARK
```

```
  0002              TYPE2_LEN       EQU     $-TYPE2_TABLE
0617                TYPE3_TABLE     LABEL   BYTE
0617 202010                 DB      BAD_CNTLR,BAD_CNTLR,BAD_ECC
  0003              TYPE3_LEN       EQU     $-TYPE3_TABLE

                    ;----- TYPE 0 ERROR

061A                TYPE_0:
061A BB0206                 MOV     BX,OFFSET TYPE0_TABLE
061D 3C09                   CMP     AL,TYPE0_LEN            ; CHECK IF ERROR IS DEFINED
061F 7363                   JAE     UNDEF_ERR_L
0621 2ED7                   XLAT    CS:TYPE0_TABLE          ; TABLE LOOKUP
0623 A27400                 MOV     DISK_STATUS,AL          ; SET ERROR CODE
0626 C3                     RET

                    ;------ TYPE 1 ERROR

0627                TYPE_1:
0627 BB0B06                 MOV     BX,OFFSET TYPE1_TABLE
062A 8BC8                   MOV     CX,AX
062C 3C0A                   CMP     AL,TYPE1_LEN            ; CHECK IF ERROR IS DEFINED
062E 7354                   JAE     UNDEF_ERR_L
0630 2ED7                   XLAT    CS:TYPE1_TABLE          ; TABLE LOOKUP
0632 A27400                 MOV     DISK_STATUS,AL          ; SET ERROR CODE
0635 80E108                 AND     CL,08H                  ; CORRECTED ECC
0638 80F908                 CMP     CL,08H
063B 752A                   JNZ     G30

                    ;----- OBTAIN ECC ERROR BURST LENGTH

063D C60642000D             MOV     CMD_BLOCK+0,RD_ECC_CMD
0642 2AC0                   SUB     AL,AL
0644 E81BFF                 CALL    COMMAND
0647 721E                   JC      G30
0649 E83E00                 CALL    HD_WAIT_REQ
064C 7219                   JC      G30
064E E82001                 CALL    PORT 0
0651 EC                     IN      AL,DX
0652 8AC8                   MOV     CL,AL
0654 E83300                 CALL    HD_WAIT_REQ
0657 720E                   JC      G30
0659 E81501                 CALL    PORT_0
065C EC                     IN      AL,DX
065D A801                   TEST    AL,01H
065F 7406                   JZ      G30
0661 C606740020             MOV     DISK_STATUS,BAD_CNTLR
0666 F9                     STC
0667                G30:
0667 8AC1                   MOV     AL,CL
0669 C3                     RET

                    ;----- TYPE 2 ERROR

066A                TYPE_2:
066A BB1506                 MOV     BX,OFFSET TYPE2_TABLE
066D 3C02                   CMP     AL,TYPE2_LEN            ; CHECK IF ERROR IS DEFINED
066F 7313                   JAE     UNDEF_ERR_L
0671 2ED7                   XLAT    CS:TYPE1_TABLE          ; TABLE LOOKUP
0673 A27400                 MOV     DISK_STATUS,AL          ; SET ERROR CODE
0676 C3                     RET

                    ;----- TYPE 3 ERROR

0677                TYPE_3:
0677 BB1706                 MOV     BX,OFFSET TYPE3_TABLE
067A 3C03                   CMP     AL,TYPE3_LEN
067C 7306                   JAE     UNDEF_ERR_L
067E 2ED7                   XLAT    CS:TYPE3_TABLE
0680 A27400                 MOV     DISK_STATUS,AL
0683 C3                     RET

0684                UNDEF_ERR_L:
0684 C6067400BB             MOV     DISK_STATUS,UNDEF_ERR
0689 C3                     RET

068A                HD_WAIT_REQ     PROC    NEAR
068A 51                     PUSH    CX
```

```
068B 2BC9                    SUB     CX,CX
068D E8EE00                  CALL    PORT_1
0690                 L1:
0690 EC                      IN      AL,DX
0691 A801                    TEST    AL,R1_REQ
0693 7508                    JNZ     L2
0695 E2F9                    LOOP    L1
0697 C606740080              MOV     DISK_STATUS,TIME_OUT
069C F9                      STC
069D                 L2:
069D 59                      POP     CX
069E C3                      RET
                     HD_WAIT_REQ     ENDP

                     ;--------------------------------------------------------
                     ; DMA_SETUP                                              :
                     ;       THIS ROUTINE SETS UP FOR DMA OPERATIONS.         :
                     ; INPUT                                                  :
                     ;       (AL) = MODE BYTE FOR THE DMA                     :
                     ;       (ES:BX) = ADDRESS TO READ/WRITE THE DATA         :
                     ; OUTPUT                                                 :
                     ;       (AX) DESTROYED                                   :
                     ;--------------------------------------------------------
069F                 DMA_SETUP       PROC    NEAR
069F 50                      PUSH    AX
06A0 A04600                  MOV     AL,CMD_BLOCK+4
06A3 3C81                    CMP     AL,81H                  ; BLOCK COUNT OUT OF RANGE
06A5 58                      POP     AX
06A6 7202                    JB      J1
06A8 F9                      STC
06A9 C3                      RET
06AA                 J1:
06AA 51                      PUSH    CX                      ; SAVE THE REGISTER
06AB FA                      CLI                             ; NO MORE INTERRUPTS
06AC E60C                    OUT     DMA+12,AL               ; SET THE FIRST/LAST F/F
06AE 50                      PUSH    AX
06AF 58                      POP     AX
06B0 E60B                    OUT     DMA+11,AL               ; OUTPUT THE MODE BYTE
06B2 8CC0                    MOV     AX,ES                   ; GET THE ES VALUE
06B4 B104                    MOV     CL,4                    ; SHIFT COUNT
06B6 D3C0                    ROL     AX,CL                   ; ROTATE LEFT
06B8 8AE8                    MOV     CH,AL                   ; GET HIGHEST NYBBLE OF ES TO CH
06BA 24F0                    AND     AL,0F0H                 ; ZERO THE LOW NYBBLE FROM SEGMENT
06BC 03C3                    ADD     AX,BX                   ; TEST FOR CARRY FROM ADDITION
06BE 7302                    JNC     J33
06C0 FEC5                    INC     CH                      ; CARRY MEANS HIGH 4 BITS MUST BE INC
06C2                 J33:
06C2 50                      PUSH    AX                      ; SAVE START ADDRESS
06C3 E606                    OUT     DMA+6,AL                ; OUTPUT LOW ADDRESS
06C5 8AC4                    MOV     AL,AH
06C7 E606                    OUT     DMA+6,AL                ; OUTPUT HIGH ADDRESS
06C9 8AC5                    MOV     AL,CH                   ; GET HIGH 4 BITS
06CB 240F                    AND     AL,0FH
06CD E682                    OUT     DMA_HIGH,AL             ; OUTPUT THE HIGH 4 BITS TO PAGE REG

                     ;------ DETERMINE COUNT

06CF A04600                  MOV     AL,CMD_BLOCK+4          ; RECOVER BLOCK COUNT
06D2 D0E0                    SHL     AL,1                    ; MULTIPLY BY 512 BYTES PER SECTOR
06D4 FEC8                    DEC     AL                      ;   AND DECREMENT VALUE BY ONE
06D6 8AE0                    MOV     AH,AL
06D8 B0FF                    MOV     AL,0FFH

                     ;----- HANDLE READ AND WRITE LONG (516D BYTE BLOCKS)

06DA 50                      PUSH    AX                      ; SAVE REGISTER
06DB A04200                  MOV     AL,CMD_BLOCK+0          ; GET COMMAND
06DE 3CE5                    CMP     AL,RD_LONG_CMD
06E0 7407                    JE      ADD4
06E2 3CE6                    CMP     AL,WR_LONG_CMD
06E4 7403                    JE      ADD4
06E6 58                      POP     AX                      ; RESTORE REGISTER
06E7 EB11                    JMP     SHORT   J20
06E9                 ADD4:
06E9 58                      POP     AX                      ; RESTORE REGISTER
06EA B80402                  MOV     AX,516D                 ; ONE BLOCK (512) PLUS 4 BYTES ECC
06ED 53                      PUSH    BX
```

```
06EE 2AFF                     SUB     BH,BH
06F0 8A1E4600                 MOV     BL,CMD_BLOCK+4
06F4 52                       PUSH    DX
06F5 F7E3                     MUL     BX                      ; BLOCK COUNT TIMES 516
06F7 5A                       POP     DX
06F8 5B                       POP     BX
06F9 48                       DEC     AX                      ; ADJUST
06FA                  J20:

06FA 50                       PUSH    AX                      ; SAVE COUNT VALUE
06FB E607                     OUT     DMA+7,AL                ; LOW BYTE OF COUNT
06FD 8AC4                     MOV     AL,AH
06FF E607                     OUT     DMA+7,AL                ; HIGH BYTE OF COUNT
0701 FB                       STI                             ; INTERRUPTS BACK ON
0702 59                       POP     CX                      ; RECOVER COUNT VALUE
0703 58                       POP     AX                      ; RECOVER ADDRESS VALUE
0704 03C1                     ADD     AX,CX                   ; ADD, TEST FOR 64K OVERFLOW
0706 59                       POP     CX                      ; RECOVER REGISTER
0707 C3                       RET                     ; RETURN TO CALLER, CFL SET BY ABOVE IF ERROR
                      DMA_SETUP       ENDP

                      ;------------------------------------------------
                      ; WAIT_INT                                       :
                      ;       THIS ROUTINE WAITS FOR THE FIXED DISK    :
                      ;       CONTROLLER TO SIGNAL THAT AN INTERRUPT   :
                      ;       HAS OCCURRED.                            :
                      ;------------------------------------------------
0708                  WAIT_INT        PROC    NEAR
0708 FB                       STI                             ; TURN ON INTERRUPTS
0709 53                       PUSH    BX                      ; PRESERVE REGISTERS
070A 51                       PUSH    CX
070B 06                       PUSH    ES
070C 56                       PUSH    SI
070D 1E                       PUSH    DS
                              ASSUME  DS:DUMMY
070E 2BC0                     SUB     AX,AX
0710 8ED8                     MOV     DS,AX                   ; ESTABLISH SEGMENT
0712 C4360401                 LES     SI,HF_TBL_VEC
                              ASSUME  DS:DATA
0716 1F                       POP     DS

                      ;----- SET TIMEOUT VALUES

0717 2AFF                     SUB     BH,BH
0719 268A5C09                 MOV     BL,BYTE PTR ES:[SI][9]          ; STANDARD TIME OUT
071D 8A264200                 MOV     AH,CMD_BLOCK
0721 80FC04                   CMP     AH,FMTDRV_CMD
0724 7506                     JNZ     W5
0726 268A5C0A                 MOV     BL,BYTE PTR ES:[SI][0AH]        ; FORMAT DRIVE
072A EB09                     JMP     SHORT   W4
072C 80FCE3           W5:     CMP     AH,CHK_DRV_CMD
072F 7504                     JNZ     W4
0731 268A5C0B                 MOV     BL,BYTE PTR ES:[SI][0BH]        ; CHECK DRIVE
0735                  W4:
0735 2BC9                     SUB     CX,CX

                      ;----- WAIT FOR INTERRUPT

0737                  W1:
0737 E84400                   CALL    PORT_1
073A EC                       IN      AL,DX
073B 2420                     AND     AL,020H
073D 3C20                     CMP     AL,020H                 ; DID INTERRUPT OCCUR
073F 740A                     JZ      W2
0741 E2F4                     LOOP    W1                      ; INNER LOOP
0743 4B                       DEC     BX
0744 75F1                     JNZ     W1                      ; OUTER LOOP
0746 C606740080               MOV     DISK_STATUS,TIME_OUT
074B                  W2:
074B E82300                   CALL    PORT_0
074E EC                       IN      AL,DX
074F 2402                     AND     AL,2                    ; ERROR BIT
0751 08067400                 OR      DISK_STATUS,AL          ; SAVE
0755 E83000                   CALL    PORT_3                  ; INTERRUPT MASK REGISTER
0758 32C0                     XOR     AL,AL                   ; ZERO
075A EE                       OUT     DX,AL                   ; RESET MASK
075B 5E                       POP     SI                      ; RESTORE REGISTERS
```

```
LOC OBJ          LINE    SOURCE

075C 07                          POP     ES
075D 59                          POP     CX
075E 5B                          POP     BX
075F C3                          RET
                         WAIT_INT        ENDP

0760                     HD_INT  PROC    NEAR
0760 50                          PUSH    AX
0761 B020                        MOV     AL,EOI                          ; END OF INTERRUPT
0763 E620                        OUT     INT_CTL_PORT,AL
0765 B007                        MOV     AL,07H                          ; SET DMA MODE TO DISABLE
0767 E60A                        OUT     DMA+10,AL
0769 E421                        IN      AL,021H
076B 0C20                        OR      AL,020H
076D E621                        OUT     021H,AL
076F 58                          POP     AX
0770 CF                          IRET
                         HD_INT  ENDP

                         ;---------------------------------------
                         ; PORTS                                 :
                         ;       GENERATE PROPER PORT VALUE      :
                         ;       BASED ON THE PORT OFFSET        :
                         ;---------------------------------------

0771                     PORT_0  PROC    NEAR
0771 BA2003                      MOV     DX,HF_PORT                      ; BASE VALUE
0774 50                          PUSH    AX
0775 2AE4                        SUB     AH,AH
0777 A07700                      MOV     AL,PORT_OFF                     ; ADD IN THE OFFSET
077A 03D0                        ADD     DX,AX
077C 58                          POP     AX
077D C3                          RET
                         PORT_0  ENDP

077E                     PORT_1  PROC    NEAR
077E E8F0FF                      CALL    PORT_0
0781 42                          INC     DX                              ; INCREMENT TO PORT ONE
0782 C3                          RET
                         PORT_1  ENDP

0783                     PORT_2  PROC    NEAR
0783 E8F8FF                      CALL    PORT_1
0786 42                          INC     DX                              ; INCREMENT TO PORT TWO
0787 C3                          RET
                         PORT_2  ENDP

0788                     PORT_3  PROC    NEAR
0788 E8F8FF                      CALL    PORT_2
078B 42                          INC     DX                              ; INCREMENT TO PORT THREE
078C C3                          RET
                         PORT_3  ENDP

                         ;---------------------------------------------
                         ; SW2_OFFS                                    :
                         ;       DETERMINE PARAMETER TABLE OFFSET      :
                         ;       USING CONTROLLER PORT TWO AND         :
                         ;       DRIVE NUMBER SPECIFIER (0-1)          :
                         ;---------------------------------------------

078D                     SW2_OFFS        PROC    NEAR
078D E8F3FF                      CALL    PORT_2
0790 EC                          IN      AL,DX                           ; READ PORT 2
0791 50                          PUSH    AX
0792 E8E9FF                      CALL    PORT_1
0795 EC                          IN      AL,DX
0796 2402                        AND     AL,2                            ; CHECK FOR ERROR
0798 58                          POP     AX
0799 7516                        JNZ     SW2_OFFS_ERR
079B 8A264300                    MOV     AH,CMD_BLOCK+1
079F 80E420                      AND     AH,00100000B                    ; DRIVE 0 OR 1
07A2 7504                        JNZ     SW2_AND
07A4 D0E8                        SHR     AL,1                            ; ADJUST
07A6 D0E8                        SHR     AL,1
07A8                     SW2_AND:
07A8 2403                        AND     AL,011B                         ; ISOLATE
07AA B104                        MOV     CL,4
```

```
LOC OBJ                 SOURCE

07AC D2E0                       SHL     AL,CL                   ; ADJUST
07AE 2AE4                       SUB     AH,AH
07B0 C3                         RET
07B1                    SW2_OFFS_ERR:
07B1 F9                         STC
07B2 C3                         RET
                        SW2_OFFS        ENDP

07B3 30382F31362F38             DB      '08/16/82'              ; RELEASE MARKER
     32

07BB                    END_ADDRESS     LABEL   BYTE
----                    CODE    ENDS
                                END
```

Notes: