Personal Computer
Hardware Reference
Library

IBM Personal Computer 20MB Fixed Disk Drive Adapter

6139790

March 17, 1986

Notes:

Contents

Notes:

Description

The 20MB Fixed Disk Drive Adapter attaches to one or two fixed disk drive units through an internal, daisy-chained, flat cable (data/control cable).

The adapter is buffered on the I/O bus and uses the system board's direct memory access (DMA) for fixed-disk-drive data transfers. When the adapter is enabled, an interrupt request occurs on the IRQ-5 line to the 8259A Interrupt Controller. The 8259A then causes an interrupt hex 0D.

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

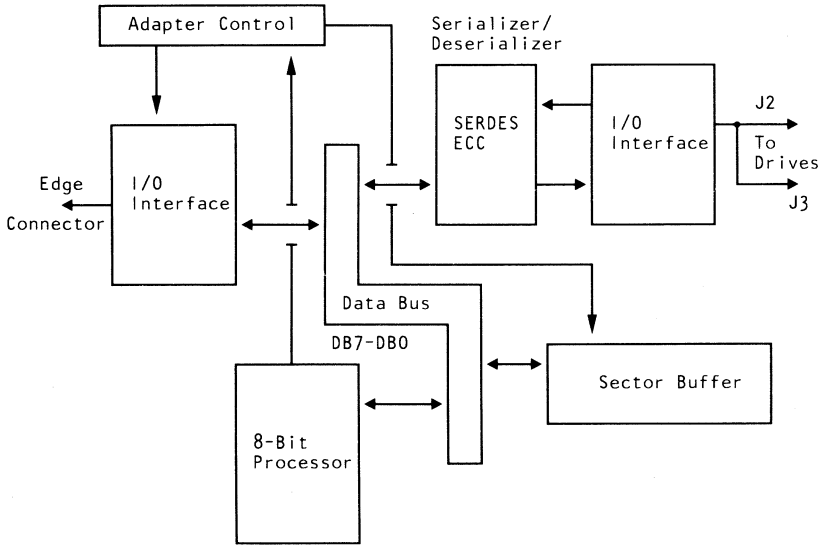
The device level control for the Fixed Disk Adapter is contained on a ROM module on the adapter. A listing of this device level control can be found in "BIOS Listing" of this section.

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 three registers that may be accessed by the system unit's microprocessor: a status register, a data register, and a read-option-jumpers register. The 8-bit status register contains the status information of the disk controller, and can be accessed at any time. This register is read-only and is used to help the transfer of data between the system unit's microprocessor and the disk controller. 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 controller-select pulse is generated by writing to port address hex 322.

The following is a block diagram of the IBM 20MB Fixed Disk Drive Adapter.



Programming Considerations

Types of Drives

The fixed disk drive adapter will accommodate any two of four different types of drives. The figure below shows the configuration of the different type drives.

Type	Cylinders	Heads	Start of Write Pre-Comp	Landing Zone
1	306	4	0	306
2	615	4	300	615
13	306	8	128	336
16	612	4	0	663

Fixed Disk Types

The figure below shows the switch settings for the above mentioned drive types. Switches 1 and 2 set the parameters of Drive 0, and switches 3 and 4 set Drive 1.

	Drive 0		Drive 1	
	Switch		Switch	
	1	2	3	4
Type 1	On	On	On	On
Type 2	Off	On	Off	On
Type 13	Off	Off	Off	Off
Type 16	On	Off	On	Off

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

Bit 5 This bit shows the logical unit number of the drive.

Bit 1 When set, this bit shows an error has occurred during command execution.

Bits 7, 6, 4, 3, 2, 0 These bits are set to zero.

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

Disk Controller Error Tables

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

The address-valid bit (bit 7) is only set when the previous command required a disk address. Bit 6 is set to 0 (spare).

	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 had not completed the seek. No time-out is measured by the controller for the seek to complete.

	Error Type		Error Code				
Bits	5	4	3	2	1	0	Description
	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				
Bits	5	4	3	2	1	0	Description
	1	0	0	0	0	0	Invalid Command: The controller had 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 Polynomial 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 format of the DCB, and defines the bytes that make up the DCB.

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

Byte 5 Bits 7 through 0 contain the control field.

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

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

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

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 0 Bits 7, 6, and 5 identify the class of the command. Bits 4 through 0 contain the Opcode (see command byte on page 10

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:

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

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 Byte

Test Drive / Ready / (Class 0, Opcode 00)

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

d = drive (0 or 1)
x = don't care
Bytes 2, 3, 4, 5, = don't care.

Recalibrate / (Class 0, Opcode 00)

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

d = drive (0 or 1)
x = don't care
r = retries
s = Step Option
Bytes 2, 3, 4, = don't care
ch = cylinder high

Reserved (Class 0, Opcode 02)

This Opcode is not used.

Request Sense / Status (Class 0, Opcode 03)

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

d = drive (0 or 1)
x = don't care
Bytes 2, 3, 4, 5, = don't care.

Format Drive / (Class 0, Opcode 04)

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	0	1	0	0
Byte 1	0	0	d	Head No.				
Byte 2	ch	0	0	0	0	0	0	
Byte 3	Cylinder Low							
Byte 4	0	0	0	Interleave				
Byte 5	r	0	0	0	0	s	s	s

d = drive (0 or 1)
r = retries
s = Step Option
ch = cylinder high
Interleave 1 to 16
for 512-byte sectors.

Ready Verify / (Class 0, Opcode 05)

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	0	1	0	1
Byte 1	0	0	d	Head No.				
Byte 2	ch	Sector No.						
Byte 3	Cylinder Low							
Byte 4	Block Count							
Byte 5	r	a	0	0	0	s	s	s

d = drive (0 or 1)
r = retries
s = Step Option
a = retry option on data ECC
ch = cylinder high
for 512-byte sectors.

Format Track (Class 0, Opcode 06)

Bit	7	6	5	4	3	2	1	0	Remarks
Byte 0	0	0	0	0	0	1	1	0	r = retries
Byte 1	0	0	d	Head No.					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 (Class 0, Opcode 07)

Bit	7	6	5	4	3	2	1	0	Remarks
Byte 0	0	0	0	0	0	1	1	1	d = drive (0 or 1)
Byte 1	0	0	d	Head No.					x = don't care
Byte 2	ch	0	0	0	0	0	0		s = Step Option
Byte 3	Cylinder Low								ch = cylinder high
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 (Class 0, Opcode 08)

Bit	7	6	5	4	3	2	1	0	Remarks
Byte 0	0	0	0	0	1	0	0	0	d = drive (0 or 1)
Byte 1	0	0	d	Head No.					r = retries
Byte 2	ch	Sector No.							a = retry option on data ECC error
Byte 3	Cylinder Low								s = step option
Byte 5	r	a	0	0	0	s	s	s	ch = cylinder high

Reserved (Class 0, Opcode 09)

This Opcode is not used.

Write (Class 0, Opcode 0A)

Bit	7	6	5	4	3	2	1	0	Remarks
Byte 0	0	0	0	0	1	0	1	0	d = drive (0 or 1)
Byte 1	0	0	d	Head No.					r = retries
Byte 2	ch	Sector No.							s = step option
Byte 3	Cylinder Low								ch = cylinder high
Byte 4	Block Count								
Byte 5	r	0	0	0	0	s	s	s	

Seek (Class 0, Opcode 0B)

Bit	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	1	0	1	1
Byte 1	0	0	d	Head No.				
Byte 2	ch	0	0	0	0	0	0	
Byte 3	Cylinder Low							
Byte 4	x	x	x	x	x	x	x	x
Byte 5	r	0	0	0	0	s	s	s

d = drive (0 or 1)
r = retries
s = Step Option
x = don't care

Initialize Drive Character-istics* (Class 0, Opcode 0C)

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

Bytes 1, 2, 3, 4, 5, = don't care.

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

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

Bytes 1, 2, 3, 4, 5, = don't care.

Read Data from Sector Buffer (Class 0, Opcode 0E)

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

Bytes 1, 2, 3, 4, 5, = don't care.

Write Data to Sector Buffer (Class 0, Opcode 0F)

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

Bytes 1, 2, 3, 4, 5, = don't care.

RAM Diagnostic (Class 7, Opcode 00)

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

Bytes 1, 2, 3, 4, 5, = don't care.

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)

Drive Diagnostic (Class 7, Opcode 03)

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

d = drive (0 or 1)
r = retries
s = step option
x = don't care

Controller Internal Diagnostics (Class 7, Opcode 04)

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

Bytes 1, 2, 3, 4, 5, = don't care.

Read Long * Track (Class 7, Opcode 05)

Bit	7	6	5	4	3	2	1	0
Byte 0	1	1	1	0	0	1	0	1
Byte 1	0	0	d	Head No.				
Byte 2	ch	Sector No.						
Byte 3	Cylinder Low							
Byte 4	Block Count							
Byte 5	r	0	0	0	0	s	s	s

d = drive (0 or 1)
r = retries
s = step option
ch = cylinder high

Write Long ** (Class 7, Opcode 06)

Bit	7	6	5	4	3	2	1	0
Byte 0	1	1	1	0	0	1	1	0
Byte 1	0	0	d	Head No.				
Byte 2	ch	Sector No.						
Byte 3	Cylinder Low							
Byte 4	Block Count							
Byte 5	r	0	0	0	0	s	s	s

d = drive (0 or 1)
s = step option
s = step option
ch = cylinder high
s = step option

* 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 active, the I/O port decoder is disabled.

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

R/W	Port Address	Function
Read Write	320 320	Read data (from controller to system unit) Write data (from system unit to controller)
Read Write	321 321	Read controller hardware status. Controller reset.
Read Write	322 322	Read option jumpers Generate controller-select-pulse
Read Write	323 323	Not used. 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 hex C9FFF.

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

-IOR This signal is active when the system board reads status or data from the controller under either programmed I/O or DMA control.

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

AEN This signal is active 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 This signal forces the disk controller to its initial power-up condition.

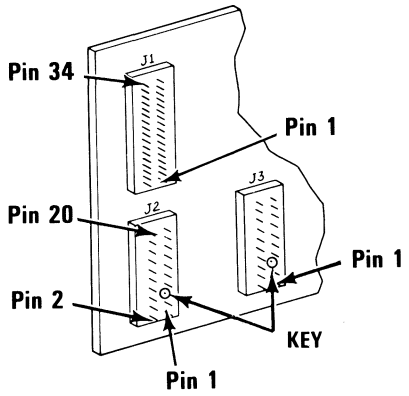
IRQ 5 This signal is active by the controller when enabled to interrupt the system board on the return ending status byte from the controller.

DRQ 3 This signal is activated 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.

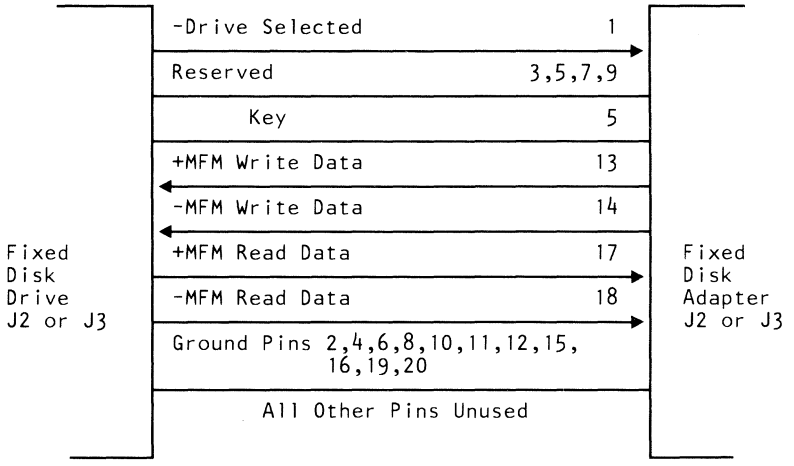
-DACK 3 This signal is active when negative, and is generated by the system board DMA channel in response to a DMA request (DRQ 3).

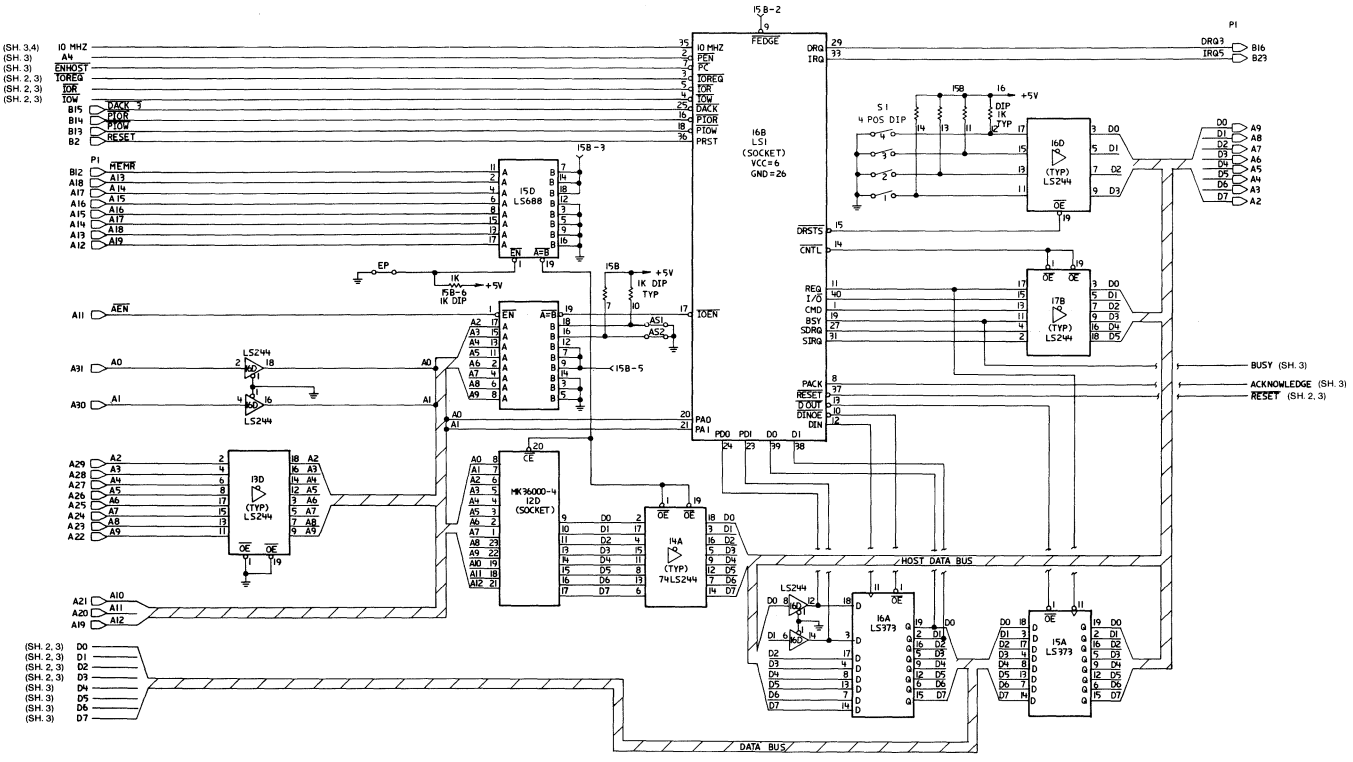
Connectors

The 20MB Fixed Disk Drive Adapter connector and interface specifications follow.

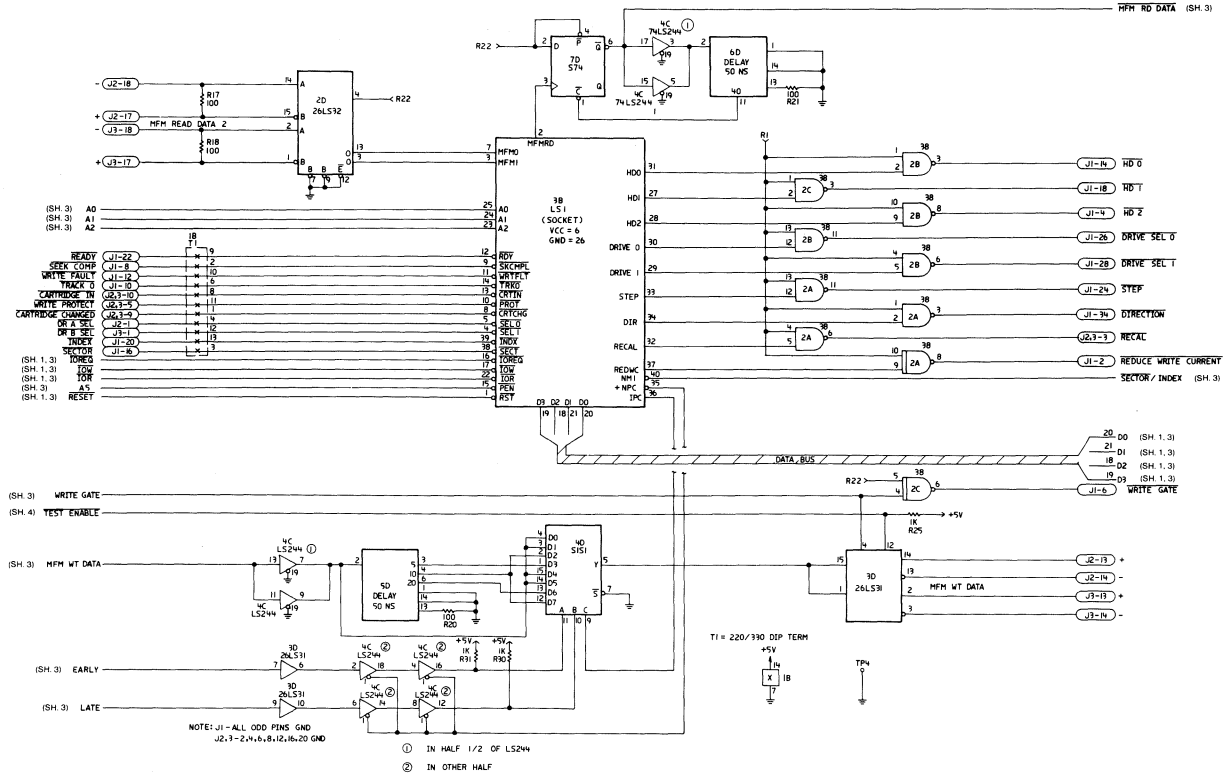


Disk Drive J1	At Standard TTL Levels	Land Number	Disk Adapter J1
	Ground–Odd Numbers	1–33	
	-Reserved	2, 16, 30, 32	
←	-Head Select 2	4	
←	-Write Gate	6	
→	-Seek Complete	8	
→	-Track 000	10	
→	-Write Fault	12	
←	-Head Select 0	14	
←	-Head Select 1	18	
→	-Index	20	
→	-Ready	22	
←	-Step	24	
←	-Drive Select 1	26	
←	-Drive Select 2	28	
←	-Drive Select 3	30	
←	-Drive Select 4	32	
←	-Direction In	34	

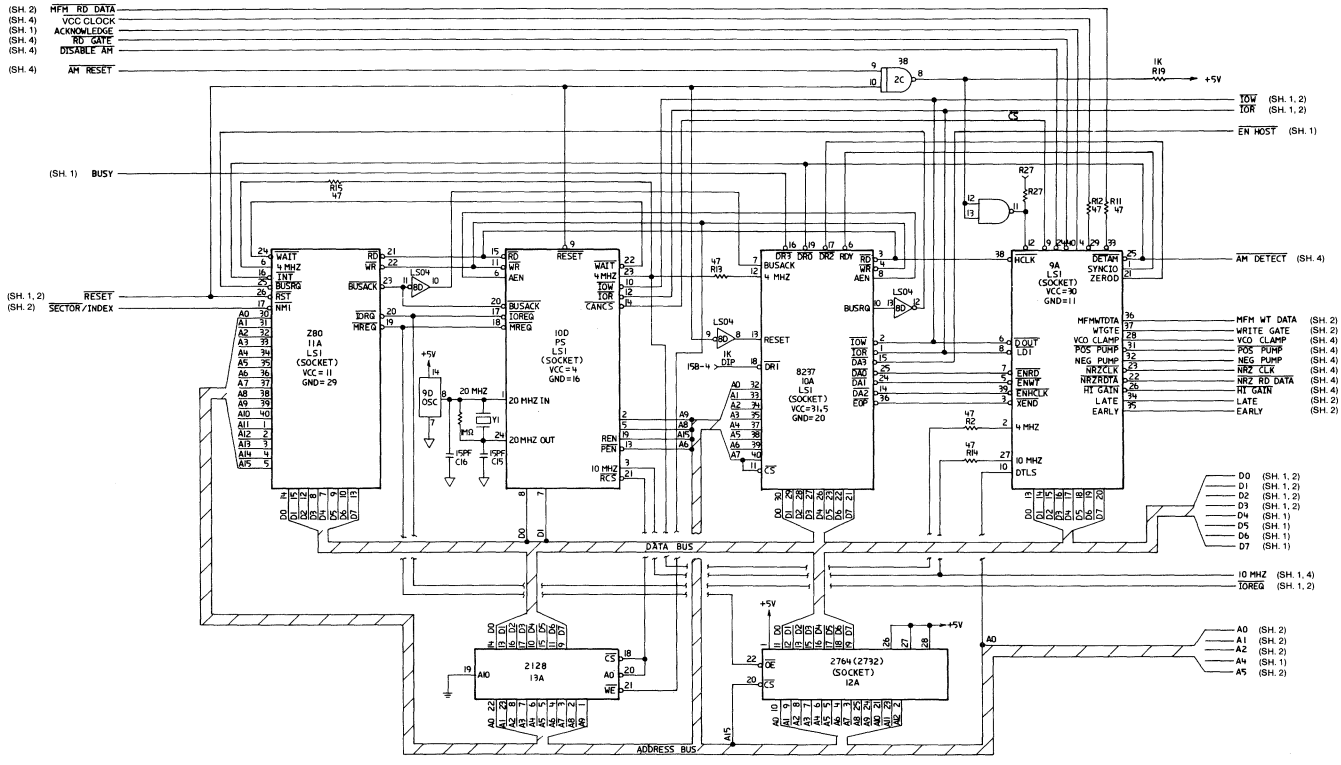

-Drive Selected 1
Reserved 3,5,7,9
Key 5
+MFM Write Data 13
-MFM Write Data 14
+MFM Read Data 17
-MFM Read Data 18
Ground Pins 2,4,6,8,10,11,12,15, 16,19,20
All Other Pins Unused
Fixed Disk Drive J2 or J3
Fixed Disk Adapter J2 or J3



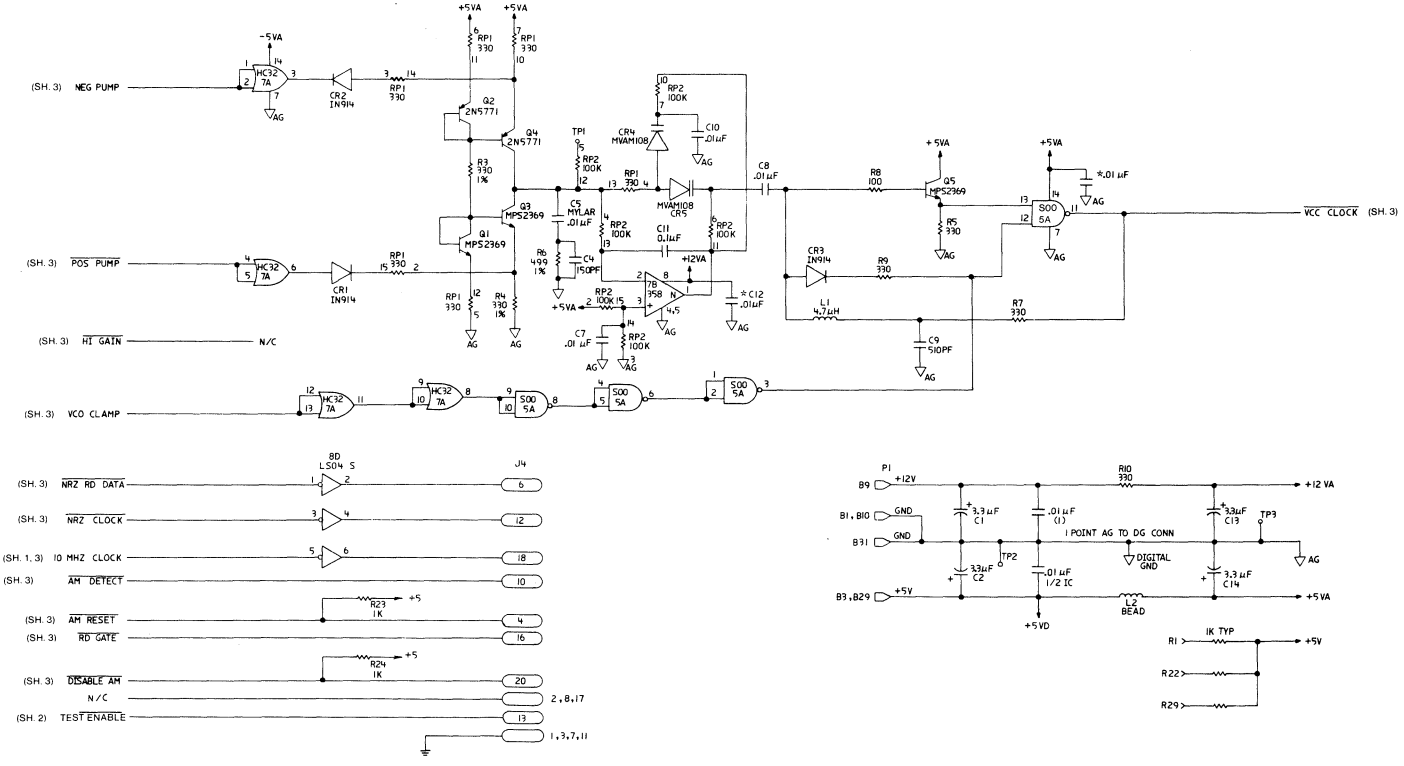
20MB Fixed Disk Drive Adapter (Sheet 1 of 4)



20MB Fixed Disk Drive Adapter (Sheet 2 of 4)



20MB Fixed Disk Drive Adapter (Sheet 3 of 4)



20MB Fixed Disk Drive Adapter (Sheet 4 of 4)

BIOS Listing

The BIOS Listing for the IBM 20MB Fixed Disk Drive Adapter follows.

```
                        PAGE    118,121
                        TITLE DISK2 ---- 10/28/85  FIXED DISK BIOS

                        ;--  INT  13H  ------------------------------------------------
                        ;                                                             :
                        ; FIXED DISK I/O INTERFACE                                    :
                        ;                                                             :
                        ;       THIS INTERFACE PROVIDES ACCESS TO 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)= 00H  RESET DISK (DL = 80H,81H) / DISKETTE
                        ;       (AH)= 01H  READ THE STATUS OF THE LAST DISK OPERATION INTO (AL)
                        ;                  NOTE: DL < 80H - DISKETTE
                        ;                        DL > 80H - DISK
                        ;       (AH)= 02H  READ THE DESIRED SECTORS INTO MEMORY
                        ;       (AH)= 03H  WRITE THE DESIRED SECTORS FROM MEMORY
                        ;       (AH)= 04H  VERIFY THE DESIRED SECTORS
                        ;       (AH)= 05H  FORMAT THE DESIRED TRACK
                        ;       (AH)= 06H  FORMAT THE DESIRED TRACK AND SET BAD SECTOR FLAGS
                        ;       (AH)= 07H  FORMAT THE DRIVE STARTING AT THE DESIRED TRACK
                        ;       (AH)= 08H  RETURN THE CURRENT DRIVE PARAMETERS
                        ;
                        ;       (AH)= 09H  INITIALIZE DRIVE PAIR CHARACTERISTICS
                        ;                  INTERRUPT 41H POINTS TO DATA BLOCK
                        ;       (AH)= 0AH  READ LONG
                        ;       (AH)= 0BH  WRITE LONG
                        ;                  NOTE: READ AND WRITE LONG ENCOMPASS
                        ;                  512 BYTES + 4 BYTES OF ECC
                        ;       (AH)= 0CH  SEEK
                        ;       (AH)= 0DH  ALTERNATE DISK RESET (SEE DL)
                        ;       (AH)= 0EH  READ SECTOR BUFFER
                        ;       (AH)= 0FH  WRITE SECTOR BUFFER,
                        ;                  (RECOMMENDED PRACTICE BEFORE FORMATTING)
                        ;       (AH)= 10H  TEST DRIVE READY
                        ;       (AH)= 11H  RECALIBRATE
                        ;       (AH)= 12H  CONTROLLER RAM DIAGNOSTIC
                        ;       (AH)= 13H  DRIVE DIAGNOSTIC
                        ;       (AH)= 14H  CONTROLLER INTERNAL DIAGNOSTIC
                        ;
                        ;               REGISTERS USED FOR FIXED DISK OPERATIONS
                        ;
                        ;               (DL)    -  DRIVE NUMBER      (80H-87H FOR DISK, VALUE CHECKED)
                        ;               (DH)    -  HEAD NUMBER       (0-7D ALLOWED, NOT VALUE CHECKED)
                        ;               (CH)    -  CYLINDER NUMBER   (0-1023D, NOT VALUE CHECKED)(SEE CL)
                        ;               (CL)    -  SECTOR NUMBER     (1-17D, 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= 00H 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
                        ;
                        ;       IF AN ERROR OCCURS ON READ DRIVE PARAMETERS,
                        ;
                        ;               AH = ERROR CODE (INIT_FAIL)
                        ;               AL = CX = DX = 0
                        ;
                        ;       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.
                        ;
                        ;--------------------------------------------------------------------------
```

```
                        PAGE
                        ;--------------------------------------------------------
                        ;       ERROR RETURN STATUS  (AH)= ??H  WHEN  CY= 1      :
                        ;--------------------------------------------------------
= 00FF                  SENSE_FAIL      EQU     0FFH            ; SENSE OPERATION FAILED
= 00CC                  WRITE_FAULT     EQU     0CCH            ; WRITE FAULT ON SELECTED DRIVE
= 00BB                  UNDEF_ERR       EQU     0BBH            ; UNDEFINED ERROR OCCURRED
= 0080                  TIME_OUT        EQU     080H            ; ATTACHMENT FAILED TO RESPOND
= 0040                  BAD_SEEK        EQU     040H            ; SEEK OPERATION FAILED
= 0020                  BAD_CNTLR       EQU     020H            ; CONTROLLER HAS FAILED
= 0011                  DATA_CORRECTED  EQU     011H            ; ECC CORRECTED DATA ERROR
= 0010                  BAD_ECC         EQU     010H            ; BAD ECC ON DISK READ
= 000B                  BAD_TRACK       EQU     00BH            ; BAD TRACK FLAG DETECTED
= 0009                  DMA_BOUNDARY    EQU     009H            ; ATTEMPT TO DMA ACROSS 64K BOUNDARY
= 0007                  INIT_FAIL       EQU     007H            ; DRIVE PARAMETER ACTIVITY FAILED
= 0005                  BAD_RESET       EQU     005H            ; RESET FAILED
= 0004                  RECORD_NOT_FND  EQU     004H            ; REQUESTED SECTOR NOT FOUND
= 0002                  BAD_ADDR_MARK   EQU     002H            ; ADDRESS MARK NOT FOUND
= 0001                  BAD_CMD         EQU     001H            ; BAD COMMAND PASSED TO DISK I/O

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

0000                    ABS0    SEGMENT AT 0H
0034                            ORG     00DH*4          ; FIXED DISK INTERRUPT VECTOR
0034                    HDISK_INT       LABEL   DWORD
004C                            ORG     013H*4          ; DISK INTERRUPT VECTOR
004C                    ORG_VECTOR      LABEL   DWORD
0064                            ORG     019H*4          ; BOOTSTRAP INTERRUPT VECTOR
0064                    BOOT_VEC        LABEL   DWORD
0078                            ORG     01EH*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
7C00                    ABS0    ENDS

0000                    DATA    SEGMENT AT 40H
006C                            ORG     06CH
006C ????               TIMER_LOW       DW      ?               ; TIMER LOW WORD
0072                            ORG     072H
0072 ????               RESET_FLAG      DW      ?               ; 1234H IF KEYBOARD RESET UNDERWAY
0074                            ORG     074H
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
0078                    DATA    ENDS

0000                    CODE    SEGMENT

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

=                       CMD_BLOCK       EQU     BYTE PTR [BP]-8 ; CMD_BLOCK HEAD
= 0320                  HF_PORT         EQU     0320H           ; DISK PORT
= 0020                  INTA00          EQU     020H            ; 8259 PORT
= 0021                  INTA01          EQU     021H            ; 8259 PORT
= 0020                  EOI             EQU     020H            ; END OF INTERRUPT COMMAND
= 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     000H            ; 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)

= 0008                  MAX_FILE        EQU     8
= 0002                  S_MAX_FILE      EQU     2
```

```
                        PAGE
                        ASSUME  CS:CODE,DS:ABS0
0000                    ORG     0H
0000 55                 DB      055H                    ; GENERIC BIOS HEADER
0001 AA                 DB      0AAH
0002 08                 DB      08D                     ; 4K MODULE

                        ;--------------------------------------------------------
                        ; 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 EB 35                      JMP     SHORT   L3
0005 35 39 58 37 32 39          DB      '59X7291 (C) COPYRIGHT IBM  CORP.'      ; COPYRIGHT NOTICE
       31 20 28 43 29 20
       43 4F 50 59 52 49
       47 48 54 20 49 42
       4D 20 20 43 4F 52
       50 2E
0025 2C 31 39 38 32 20          DB      ',1982 ,1985.'
       2C 31 39 38 35 2E
0031 20 31 30 2F 32 38          DB      ' 10/28/85'                             ; RELEASE MARKER
       2F 38 35
003A                    L3:
003A 2B C0                      SUB     AX,AX                                   ; ADDRESS LOW RAM
003C 8E D8                      MOV     DS,AX
003E FA                         CLI
003F A1 004C R                  MOV     AX,WORD PTR ORG_VECTOR                  ; LOAD DISKETTE IP
0042 A3 0100 R                  MOV     WORD PTR DISK_VECTOR,AX                 ; STORE AT INT 40H
0045 A1 004E R                  MOV     AX,WORD PTR ORG_VECTOR+2                ; LOAD DISKETTE CS
0048 A3 0102 R                  MOV     WORD PTR DISK_VECTOR+2,AX               ; STORE AT INT 40H
004B C7 06 004C R 0251 R        MOV     WORD PTR ORG_VECTOR, OFFSET DISK_IO     ; FIXED DISK HANDLER
0051 8C 0E 004E R               MOV     WORD PTR ORG_VECTOR+2,CS                ;  AT INT 13H
0055 B8 0755 R                  MOV     AX, OFFSET HD_INT                       ; FIXED DISK INTERRUPT
0058 A3 0034 R                  MOV     WORD PTR HDISK_INT,AX                   ;  HANDLER AT INT 0DH
005B 8C 0E 0036 R               MOV     WORD PTR HDISK_INT+2,CS
005F C7 06 0064 R 0192 R        MOV     WORD PTR BOOT_VEC,OFFSET BOOT_STRAP     ; BOOTSTRAP ROUTINE AT
0065 8C 0E 0066 R               MOV     WORD PTR BOOT_VEC+2,CS                  ;  INT 19H
0069 C7 06 0104 R 03FF R        MOV     WORD PTR HF_TBL_VEC,OFFSET FD_TBL       ; PARAMETER TABLE AT
006F 8C 0E 0106 R               MOV     WORD PTR HF_TBL_VEC+2,CS                ;  INT 41H
0073 FB                         STI

                                ASSUME  DS:DATA
0074 B8 ---- R                  MOV     AX,DATA                 ; ESTABLISH SEGMENT
0077 8E D8                      MOV     DS,AX
0079 C6 06 0074 R 00            MOV     DISK_STATUS,0           ; RESET THE STATUS INDICATOR
007E C6 06 0075 R 00            MOV     HF_NUM,0                ; ZERO COUNT OF DRIVES
0083 C6 06 0077 R 00            MOV     PORT_OFF,0              ; ZERO CARD OFFSET
0088 B9 0025                    MOV     CX,025H                 ; RETRY COUNT
008B                    L4:
008B E8 0177 R                  CALL    HD_RESET_1              ; RESET CONTROLLER
008E 73 05                      JNC     L7
0090 E2 F9                      LOOP    L4                      ; TRY RESET AGAIN
0092 E9 0154 R                  JMP     ERROR_EX
0095                    L7:
0095 B9 0001                    MOV     CX,1
0098 BA 0080                    MOV     DX,80H
009B B8 1200                    MOV     AX,1200H                ; CONTROLLER DIAGNOSTICS
009E CD 13                      INT     13H                     ;  CHECK THE INTERNAL RAM
00A0 73 03                      JNC     P7                      ;  BUFFERS
00A2 E9 0154 R                  JMP     ERROR_EX
00A5                    P7:
00A5 B8 1400                    MOV     AX,1400H                ; CONTROLLER DIAGNOSTICS
00A8 CD 13                      INT     13H                     ;  INTERNAL CHECKSUM AND
00AA 73 03                      JNC     P9                      ;  ECC CIRCUITRY TEST.
00AC E9 0154 R                  JMP     ERROR_EX
00AF                    P9:
00AF C7 06 006C R 0000          MOV     TIMER_LOW,0             ; ZERO TIMER
00B5 81 3E 0072 R 1234          CMP     RESET_FLAG,1234H        ; KEYBOARD RESET
00BB 75 06                      JNE     P8
00BD C7 06 006C R 019A          MOV     TIMER_LOW,410D          ; SKIP WAIT ON RESET
00C3                    P8:
00C3 FA                         CLI                             ; DISABLE INTERRUPTS
00C4 E4 21                      IN      AL,INTA01               ; TIMER
00C6 24 FE                      AND     AL,0FEH                 ; ENABLE TIMER
00C8 E6 21                      OUT     INTA01,AL               ; START TIMER
00CA FB                         STI                             ; INTERRUPTS ON
00CB                    P4:
00CB E8 0177 R-                 CALL    HD_RESET_1              ; RESET CONTROLLER
00CE 72 07                      JC      P10
00D0 B8 1000                    MOV     AX,1000H                ; TEST TO SEE IF THE DRIVE
00D3 CD 13                      INT     13H                     ;  IS READY
00D5 73 0A                      JNC     P2
00D7                    P10:
00D7 A1 006C R                  MOV     AX,TIMER_LOW
00DA 3D 01BE                    CMP     AX,446D                 ; 25 SECONDS
00DD 72 EC                      JB      P4
00DF EB 73                      JMP     SHORT ERROR_EX
00E1                    P2:
00E1 B8 1100                    MOV     AX,1100H                ; RECALIBRATE THE DRIVE 0
00E4 CD 13                      INT     13H
00E6 72 6C                      JC      ERROR_EX

00E8 B8 0900                    MOV     AX,0900H                ; SET DRIVE PARAMETERS
00EB CD 13                      INT     13H                     ;  FOR DRIVE 0
00ED 72 65                      JC      ERROR_EX

00EF B8 C800                    MOV     AX,0C800H               ; DMA TO BUFFER
00F2 8E C0                      MOV     ES,AX                   ; SET SEGMENT
00F4 2B DB                      SUB     BX,BX
00F6 B8 0F00                    MOV     AX,0F00H                ; WRITE SECTOR BUFFER
00F9 CD 13                      INT     13H
00FB 72 57                      JC      ERROR_EX

00FD FE 06 0075 R               INC     HF_NUM                  ; DRIVE ZERO RESPONDED
0101 BA 0213                    MOV     DX,213H                 ; EXPANSION BOX
0104 B0 00                      MOV     AL,0
0106 EE                         OUT     DX,AL                   ; TURN BOX OFF
0107 BA 0321                    MOV     DX,321H                 ; TEST IF CONTROLLER
```

```
010A EC                         IN      AL,DX                   ; ... IS IN THE SYSTEM UNIT
010B 24 0F                      AND     AL,0FH
010D 3C 0F                      CMP     AL,0FH
010F 74 06                      JE      BOX_ON
0111 C7 06 006C R 01A4          MOV     TIMER_LOW,420D          ; CONTROLLER IS IN SYSTEM UNIT
0117                    BOX_ON:
0117 BA 0213                    MOV     DX,213H                 ; EXPANSION BOX
011A B0 FF                      MOV     AL,0FFH
011C EE                         OUT     DX,AL                   ; TURN BOX ON
011D B9 0001                    MOV     CX,1                    ; ATTEMPT NEXT DRIVES
0120 BA 0081                    MOV     DX,081H
0123                    P3:
0123 2B C0                      SUB     AX,AX                   ; RESET THE CONTROLLER
0125 CD 13                      INT     13H
0127 72 42                      JC      POD_DONE
0129 B8 1100                    MOV     AX,01100H               ; RECALIBRATE THE DRIVE 1
012C CD 13                      INT     13H
012E 73 0A                      JNC     P5
0130 A1 006C R                  MOV     AX,TIMER_LOW
0133 3D 01BE                    CMP     AX,446D                 ; 25 SECONDS
0136 72 EB                      JB      P3
0138 EB 31                      JMP     SHORT POD_DONE
013A                    P5:
013A B8 0900                    MOV     AX,0900H                ; INITIALIZE DRIVE CHARACTERISTICS
013D CD 13                      INT     13H                     ;  FOR DRIVE 1
013F 72 2A                      JC      POD_DONE
0141 FE 06 0075 R               INC     HF_NUM                  ; TALLY ANOTHER DRIVE
0145 81 FA 0081                 CMP     DX,(80H + S_MAX_FILE - 1)
0149 73 20                      JAE     POD_DONE
014B 42                         INC     DX
014C EB D5                      JMP     P3

014E 31 37 30 31 0D 0A  F17     DB      '1701',0DH,0AH          ; POST MESSAGE
= 0006                  F17L    EQU     $-F17

                        ;-----  POD ERROR

0154                    ERROR_EX:
0154 BD 000F                    MOV     BP,0FH                  ; POD ERROR FLAG
0157 2B F6                      SUB     SI,SI
0159 B9 0006                    MOV     CX,F17L                 ; MESSAGE CHARACTER COUNT
015C B7 00                      MOV     BH,0                    ; PAGE ZERO
015E                    OUT_CH:
015E 2E: 8A 84 014E R           MOV     AL,CS:F17[SI]           ; GET BYTE
0163 B4 0E                      MOV     AH,14D                  ; VIDEO OUT
0165 CD 10                      INT     10H                     ; DISPLAY CHARACTER
0167 46                         INC     SI                      ; NEXT CHAR
0168 E2 F4                      LOOP    OUT_CH                  ; DO MORE
016A F9                         STC
016B                    POD_DONE:
016B FA                         CLI                             ; NO INTERRUPTS
016C E4 21                      IN      AL,INTA01               ; READ THE INTERRUPT MASK
016E 0C 01                      OR      AL,01H                  ; DISABLE THE TIMER
0170 E6 21                      OUT     INTA01,AL
0172 FB                         STI                             ; ENABLE INTERRUPTS
0173 E8 0232 R                  CALL    DSBL                    ; DISABLE THE CARD MASKS
0176 CB                         RET

0177                    HD_RESET_1      PROC    NEAR
0177 51                         PUSH    CX                      ; SAVE REGISTER
0178 52                         PUSH    DX
0179 B9 0100                    MOV     CX,0100H                ; RETRY COUNT
017C                    L6:
017C E8 076D R                  CALL    PORT_0
017F 42                         INC     DX                      ; ADDRESS PORT_1
0180 EE                         OUT     DX,AL                   ; RESET CARD
0181 EB 00                      JMP     $+2                     ; I/O DELAY AT LEAST +5us
0183 EB 00                      JMP     $+2                     ; ALLOW TIME TO CLEAR THE
0185 EB 00                      JMP     $+2                     ;  HARDWARE STATUS REGISTER
0187 EC                         IN      AL,DX                   ; READ THE HARDWARE STATUS
0188 24 3F                      AND     AL,00111111B            ; MASK OFF UPPER 2 BITS AND CLEAR CY
018A 74 03                      JZ      R3                      ; EXIT IF REGISTER IS CLEARED WITH CY=0
018C E2 EE                      LOOP    L6                      ; TRY AGAIN
018E F9                         STC                             ; SET ERROR CONDITION CY=1
018F                    R3:
018F 5A                         POP     DX                      ; RESTORE REGISTER
0190 59                         POP     CX
0191 C3                         RET
0192                    HD_RESET_1      ENDP
0192                    DISK_SETUP      ENDP
```

```
                                PAGE
                                ;---  INT  19 H ----------------------------------------------
                                ;                                                            :
                                ; INTERRUPT 19 BOOT STRAP LOADER                             :
                                ;                                                            :
                                ;  -  THE FIXED DISK BIOS REPLACES THE INTERRUPT 19H BOOT    :
                                ;       STRAP VECTOR WITH A POINTER TO THIS BOOT ROUTINE AND  :
                                ;       RESETS 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 (0000:7C00H) WHERE CONTROL IS TRANSFERRED.  :
                                ;       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:
                                ;                                                            :
                                ;------------------------------------------------------------

 0192                           BOOT_STRAP:
                                        ASSUME  DS:ABS0,ES:ABS0
 0192 2B C0                             SUB     AX,AX
 0194 8E D8                             MOV     DS,AX                   ; ESTABLISH SEGMENT
 0196 B4 C0                             MOV     AH,0C0H
 0198 CD 15                             INT     15H                     ; READ CONFIGURATION PARAMETERS
                                                                        ; IF XT OR PC, INTERRUPTS ARE DISABLED
                                ;-----  RESET PARAMETER VECTORS          ;  AT THIS POINT.

 019A FA                                CLI
 019B C7 06 0104 R 03FF R               MOV     WORD PTR HF_TBL_VEC,OFFSET FD_TBL
 01A1 8C 0E 0106 R                      MOV     WORD PTR HF_TBL_VEC+2,CS
 01A5 73 0A                             JNC     H0                      ; JMP IF INT 15 FUNCTION IMPLEMENTED

 01A7 C7 06 0078 R 0227 R               MOV     WORD PTR DISKETTE_PARM,OFFSET DISKETTE_TBL
 01AD 8C 0E 007A R                      MOV     WORD PTR DISKETTE_PARM+2,CS
 01B1                           H0:
 01B1 FB                                STI

                                ;-----  ATTEMPT BOOTSTRAP FROM DISKETTE

 01B2 2B D2                             SUB     DX,DX                   ; DRIVE ZERO

                                ;-----  ESTABLISH ES:BX POINTER

 01B4 8E C2                             MOV     ES,DX                   ; ESTABLISH SEGMENT
 01B6 BB 7C00 R                         MOV     BX,OFFSET BOOT_LOCN     ; SET BX TO 7C00H

                                ;-----  CLEAR  BOOT_LOCN

 01B9 FC                                CLD                             ; DIRECTION FORWARD
 01BA 33 C0                             XOR     AX,AX
 01BC B9 0100                           MOV     CX,256                  ; CLEAR 256 WORDS
 01BF 8B FB                             MOV     DI,BX                   ; POINT TO BOOT LOCATION BUFFER
 01C1 F3/ AB                            REP     STOSW                   ; ZERO THE BOOT LOCATION BUFFER

 01C3 B9 0004                           MOV     CX,4                    ; SET RETRY COUNT
 01C6                           H1:                                     ; IPL_SYSTEM
 01C6 51                                PUSH    CX                      ; SAVE RETRY COUNT
 01C7 2B C0                             SUB     AX,AX                   ; RESET THE DISKETTE
 01C9 CD 13                             INT     13H                     ; FILE IO CALL
 01CB 72 08                             JC      H2                      ; IF ERROR, TRY AGAIN

 01CD B8 0201                           MOV     AX,0201H                ; READ IN THE SINGLE SECTOR
 01D0 B9 0001                           MOV     CX,1                    ; SECTOR 1, TRACK 0
 01D3 CD 13                             INT     13H                     ; FILE IO CALL
 01D5 59                        H2:     POP     CX                      ; RECOVER RETRY COUNT
 01D6 73 09                             JNC     H3                      ; CARRY FLAG SET BY UNSUCCESSFUL READ

 01D8 80 FC 80                          CMP     AH,80H                  ; IF TIME OUT, NO RETRY
 01DB 74 22                             JZ      H6                      ; TRY FIXED DISK

 01DD E2 E7                             LOOP    H1                      ; DO IT FOR RETRY TIMES
 01DF EB 1E                             JMP     SHORT H6                ; UNABLE TO IPL FROM THE DISKETTE

 01E1 80 3E 7C00 R 06           H3:     CMP     BYTE PTR BOOT_LOCN,06H  ; CHECK FOR FIRST INSTRUCTION INVALID
 01E6 72 3D                             JB      H10                     ; IF BOOT NOT VALID, GO TO BASIC

                                ;-----  INSURE DATA PATTERN FIRST 8 WORDS NOT ALL EQUAL

 01E8 BF 7C00 R                         MOV     DI,OFFSET BOOT_LOCN     ; CHECK DATA PATTERN
 01EB B9 0008                           MOV     CX,8                    ; CHECK THE NEXT 8 WORDS
 01EE A1 7C00 R                         MOV     AX,WORD PTR BOOT_LOCN   ; LOAD THE FIRST WORD

 01F1 83 C7 02                  H4:     ADD     DI,2                    ; POINT TO NEXT WORD
 01F4 3B 05                             CMP     AX,[DI]                 ; CHECK DATA PATTERN FOR A FILL PATTERN
 01F6 E1 F9                             LOOPZ   H4                      ;
 01F8 74 2B                             JZ      H10                     ; BOOT NOT VALID, GO TO BASIC
 01FA                           H5:
 01FA EA 7C00 ---- R                    JMP     BOOT_LOCN

                                ;-----  ATTEMPT BOOTSTRAP FROM FIXED DISK

 01FF                           H6:
 01FF 2B C0                             SUB     AX,AX                   ; RESET DISKETTE
 0201 CD 13                             INT     13H
 0203 B9 0003                           MOV     CX,3                    ; SET RETRY COUNT
 0206 BA 0080                           MOV     DX,0080H                ; FIXED DISK ZERO
 0209                           H7:                                     ; IPL_SYSTEM
 0209 51                                PUSH    CX                      ; SAVE RETRY COUNT
 020A 2B C0                             SUB     AX,AX                   ; RESET THE FIXED DISK
 020C CD 13                             INT     13H                     ; FILE IO CALL
 020E 72 08                             JC      H8                      ; IF ERROR, TRY AGAIN

                                ;-----  ES AND BX ALREADY ESTABLISHED

 0210 B8 0201                           MOV     AX,0201H                ; READ IN THE SINGLE SECTOR
 0213 B9 0001                           MOV     CX,1                    ; SECTOR 1, TRACK 0
 0216 CD 13                             INT     13H                     ; FILE IO CALL
 0218 59                        H8:     POP     CX                      ; RECOVER RETRY COUNT
 0219 72 08                             JC      H9
```

```
0218 A1 7DFE R                  MOV     AX,WORD PTR BOOT_LOCN+510D
021E 3D AA55                    CMP     AX,0AA55H               ; TEST FOR GENERIC BOOT BLOCK
0221 74 D7                      JZ      H5                      ; GO TO BOOT LOCATION
0223                    H9:
0223 E2 E4                      LOOP    H7                      ; DO IT FOR RETRY TIMES

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

0225                    H10:
0225 CD 18                      INT     18H                     ; RESIDENT BASIC

0227                    DISKETTE_TBL:

0227 CF                         DB      11001111B               ; SRT=D, HD UNLOAD=0F - 1ST SPEC BYTE
0228 02                         DB      2                       ; HD LOAD=1, MODE=DMA - 2ND SPEC BYTE
0229 25                         DB      25H                     ; MOTOR TIMEOUT AFTER OPERATION
022A 02                         DB      2                       ; 512 BYTES PER SECTOR
022B 08                         DB      8                       ; EOT (LAST SECTOR ON TRACK)
022C 2A                         DB      02AH                    ; GAP LENGTH
022D FF                         DB      0FFH                    ; DTL
022E 50                         DB      050H                    ; GAP LENGTH FOR FORMAT
022F F6                         DB      0F6H                    ; FILL BYTE FOR FORMAT
0230 19                         DB      25                      ; HEAD SETTLE TIME (MILLISECONDS)
0231 04                         DB      4                       ; MOTOR START TIME (1/8 SECOND)

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

0232                    DSBL    PROC    NEAR
0232 2A C0                      SUB     AL,AL                   ; RESET INT/DMA MASK
0234 BA 0323                    MOV     DX,HF_PORT+3            ; LOAD FOR PORT_ADDRESS 3
0237 FA                         CLI                             ; DISABLE INTERRUPTS
0238 EE                         OUT     DX,AL                   ; RESET INT/DMA MASK CARD 0
0239 83 C2 04                   ADD     DX,4
023C EE                         OUT     DX,AL                   ; RESET INT/DMA MASK CARD 1
023D 83 C2 04                   ADD     DX,4
0240 EE                         OUT     DX,AL                   ; RESET INT/DMA MASK CARD 2
0241 83 C2 04                   ADD     DX,4
0244 EE                         OUT     DX,AL                   ; RESET INT/DMA MASK CARD 3

0245 B0 07                      MOV     AL,07H
0247 E6 0A                      OUT     DMA+10,AL               ; SET DMA MODE TO DISABLE
0249 E4 21                      IN      AL,INTA01
024B 0C 20                      OR      AL,020H
024D E6 21                      OUT     INTA01,AL               ; DISABLE IREQ 5
024F FB                         STI                             ; ENABLE INTERRUPTS
0250 C3                         RET
0251                    DSBL    ENDP

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

0251                    DISK_IO PROC    FAR
                                ASSUME  DS:DATA,ES:NOTHING
0251 80 FA 80                   CMP     DL,080H                 ; TEST FOR FIXED DISK DRIVE
0254 73 05                      JAE     HARD_DISK               ; YES, HANDLE HERE
0256 CD 40                      INT     40H                     ; DISKETTE HANDLER
0258                    RET_2:
0258 CA 0002                    RET     2                       ; BACK TO CALLER

025B                    HARD_DISK:
025B FB                         STI                             ; ENABLE INTERRUPTS
025C 0A E4                      OR      AH,AH
025E 75 09                      JNZ     A3
0260 CD 40                      INT     40H                     ; RESET NEC WHEN AH=0
0262 2A E4                      SUB     AH,AH
0264 80 FA 81                   CMP     DL,(80H+S_MAX_FILE-1)   ; DL IN LIMITS?
0267 77 EF                      JA      RET_2
0269                    A3:
0269 80 FC 08                   CMP     AH,8                    ; GET PARAMETERS IS A SPECIAL CASE
026C 75 03                      JNZ     A2
026E E9 0380 R                  JMP     GET_PARM_N
0271                    A2:
0271 55                         PUSH    BP                      ; SAVE THE BASE POINTER
0272 8B EC                      MOV     BP,SP                   ; LOAD THE CMD_BLOCK POINTER
0274 83 EC 08                   SUB     SP,8                    ; ALLOCATE SPACE FOR THE COMMAND BLOCK
                                                                ;  ON THE STACK.
0277 53                         PUSH    BX                      ; SAVE REGISTERS DURING OPERATION
0278 51                         PUSH    CX
0279 52                         PUSH    DX
027A 1E                         PUSH    DS
027B 06                         PUSH    ES
027C 56                         PUSH    SI
027D 57                         PUSH    DI
027E BE ---- R                  MOV     SI,DATA
0281 8E DE                      MOV     DS,SI                   ; ESTABLISH DATA SEGMENT

0283 E8 02D0 R                  CALL    DISK_IO_CONT            ; PERFORM THE OPERATION

0286 50                         PUSH    AX
0287 E8 0232 R                  CALL    DSBL                    ; BE SURE DISABLES OCCURRED
028A B8 ---- R                  MOV     AX,DATA
028D 8E D8                      MOV     DS,AX                   ; ESTABLISH SEGMENT
028F 58                         POP     AX                      ; RESTORE THE REGISTERS
0290 8A 26 0074 R               MOV     AH,DISK_STATUS          ; GET STATUS FROM OPERATION
0294 5F                         POP     DI
0295 5E                         POP     SI
0296 07                         POP     ES
0297 1F                         POP     DS
0298 5A                         POP     DX
0299 59                         POP     CX
029A 5B                         POP     BX

029B 83 C4 08                   ADD     SP,8                    ; ADJUST FOR THE COMMAND BLOCK.
029E 5D                         POP     BP                      ; RESTORE BASE POINTER
029F 80 FC 01                   CMP     AH,1                    ; SET THE CARRY FLAG TO INDICATE
02A2 F5                         CMC                             ;  SUCCESS OR FAILURE
02A3 CA 0002                    RET     2                       ; THROW AWAY SAVED FLAGS
02A6                    DISK_IO ENDP
```

```
                        PAGE
02A6                    M1      LABEL   WORD                    ; FUNCTION TRANSFER TABLE
02A6 032E R                     DW      DISK_RESET              ; 000H
02A8 0347 R                     DW      RETURN_STATUS           ; 001H
02AA 0350 R                     DW      DISK_READ               ; 002H
02AC 0359 R                     DW      DISK_WRITE              ; 003H
02AE 0362 R                     DW      DISK_VERF               ; 004H
02B0 0369 R                     DW      FMT_TRK                 ; 005H
02B2 036F R                     DW      FMT_BAD                 ; 006H
02B4 0375 R                     DW      FMT_DRV                 ; 007H
02B6 0326 R                     DW      BAD_COMMAND             ; 008H
02B8 043F R                     DW      INIT_DRV                ; 009H
02BA 04F4 R                     DW      RD_LONG                 ; 00AH
02BC 0501 R                     DW      WR_LONG                 ; 00BH
02BE 0515 R                     DW      DISK_SEEK               ; 00CH
02C0 032E R                     DW      DISK_RESET              ; 00DH
02C2 051B R                     DW      RD_BUFF                 ; 00EH
02C4 0527 R                     DW      WR_BUFF                 ; 00FH
02C6 0533 R                     DW      TST_RDY                 ; 010H
02C8 0539 R                     DW      HDISK_RECAL             ; 011H
02CA 053F R                     DW      RAM_DIAG                ; 012H
02CC 0545 R                     DW      CHK_DRV                 ; 013H
02CE 054B R                     DW      CNTLR_DIAG              ; 014H
= 002A                  M1L     EQU     $-M1

02D0                    DISK_IO_CONT    PROC    NEAR
02D0 80 FC 01                   CMP     AH,01H                  ; RETURN STATUS
02D3 74 72                      JE      RETURN_STATUS

02D5 80 EA 80                   SUB     DL,080H                 ; CONVERT DRIVE NUMBER TO 0 BASED RANGE
02D8 80 FA 08                   CMP     DL,MAX_FILE             ; LEGAL DRIVE TEST
02DB 73 49                      JAE     BAD_COMMAND

02DD C6 06 0074 R 00            MOV     DISK_STATUS,0           ; RESET THE STATUS INDICATOR

                        ;-----  SET UP COMMAND BLOCK

02E2 FE C9                      DEC     CL                      ; SECTORS 0-16 FOR CONTROLLER
02E4 C6 46 F8 00                MOV     CMD_BLOCK+0,0           ; SET TO ZERO THE OP CODE
02E8 88 4E FA                   MOV     CMD_BLOCK+2,CL          ; SECTOR AND HIGH 2 BITS CYLINDER
02EB 88 6E FB                   MOV     CMD_BLOCK+3,CH          ; CYLINDER LOW
02EE 88 46 FC                   MOV     CMD_BLOCK+4,AL          ; INTERLEAVE / BLOCK COUNT
02F1 A0 0076 R                  MOV     AL,CONTROL_BYTE         ; CONTROL BYTE (STEP OPTION)
02F4 88 46 FD                   MOV     CMD_BLOCK+5,AL          ; SET THE CONTROL FIELD

                        ;-----  CALCULATE THE PORT OFFSET

02F7 8A EA                      MOV     CH,DL                   ; SAVE DL
02F9 80 CA 01                   OR      DL,1
02FC FE CA                      DEC     DL
02FE D0 E2                      SHL     DL,1                    ; GENERATE OFFSET
0300 88 16 0077 R               MOV     PORT_OFF,DL             ; STORE OFFSET
0304 8A D5                      MOV     DL,CH                   ; RESTORE DL
0306 80 E2 01                   AND     DL,1                    ; MAKE DRIVE 0 OR 1
0309 B1 05                      MOV     CL,5                    ; SHIFT COUNT
030B D2 E2                      SHL     DL,CL                   ; DRIVE NUMBER (0,1)
030D 0A D6                      OR      DL,DH                   ; HEAD NUMBER
030F 88 56 F9                   MOV     CMD_BLOCK+1,DL          ; SET THE DRIVE AND HEAD

0312 8B C8                      MOV     CX,AX                   ; CALCULATE JUMP ADDRESS
0314 8A CD                      MOV     CL,CH                   ; GET INTO LOW BYTE
0316 32 ED                      XOR     CH,CH                   ; ZERO HIGH BYTE
0318 D1 E1                      SAL     CX,1                    ; *2 FOR TABLE LOOKUP
031A 8B F1                      MOV     SI,CX                   ; PUT INTO SI FOR BRANCH
031C 83 F9 2A                   CMP     CX,MIL                  ; TEST WITHIN RANGE
031F 73 05                      JNB     BAD_COMMAND
0321 2E: FF A4 02A6 R           JMP     WORD PTR CS:[SI+OFFSET M1]      ; GO DO THE COMMAND
0326                    BAD_COMMAND:
0326 C6 06 0074 R 01            MOV     DISK_STATUS,BAD_CMD     ; SET BAD COMMAND ERROR
032B B0 00                      MOV     AL,0
032D C3                         RET                             ; EXIT
032E                    DISK_IO_CONT    ENDP

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

032E                    DISK_RESET      PROC    NEAR
032E E8 076D R                  CALL    PORT_0                  ; RESET PORT
0331 42                         INC     DX                      ; PORT 1 ADDRESS
0332 EE                         OUT     DX,AL                   ; RESET CARD
0333 EB 00                      JMP     $+2                     ; I/O DELAY AT LEAST +5us
0335 EB 00                      JMP     $+2                     ; ALLOW TIME TO CLEAR THE
0337 EB 00                      JMP     $+2                     ;  HARDWARE STATUS REGISTER
0339 EC                         IN      AL,DX                   ; READ THE HARDWARE STATUS
033A 24 3F                      AND     AL,00111111B            ; MASK OFF UPPER 2 BITS AND CLEAR CY
033C 74 06                      JZ      DR1                     ; EXIT IF REGISTER IS CLEARED WITH CY=0
033E C6 06 0074 R 05            MOV     DISK_STATUS,BAD_RESET   ; SET THE ERROR CONDITION
0343 C3                         RET                             ; EXIT
0344                    DR1:
0344 E9 043F R                  JMP     INIT_DRV                ; SET THE DRIVE PARAMETERS

0347                    DISK_RESET      ENDP

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

0347                    RETURN_STATUS   PROC    NEAR
0347 A0 0074 R                  MOV     AL,DISK_STATUS          ; OBTAIN PREVIOUS STATUS
034A C6 06 0074 R 00            MOV     DISK_STATUS,0           ; RESET STATUS
034F C3                         RET
0350                    RETURN_STATUS   ENDP

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

0350                    DISK_READ       PROC    NEAR
0350 B0 47                      MOV     AL,DMA_READ             ; MODE BYTE FOR DMA READ
0352 C6 46 F8 08                MOV     CMD_BLOCK+0,READ_CMD
0356 E9 055E R                  JMP     DMA_OPN
0359                    DISK_READ       ENDP
```

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

0359                    DISK_WRITE      PROC    NEAR
0359 B0 4B                      MOV     AL,DMA_WRITE            ; MODE BYTE FOR DMA WRITE
035B C6 46 F8 0A                MOV     CMD_BLOCK+0,WRITE_CMD
035F E9 055E R                  JMP     DMA_OPN
0362                    DISK_WRITE      ENDP

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

0362                    DISK_VERF       PROC    NEAR
0362 C6 46 F8 05                MOV     CMD_BLOCK+0,CHK_TRK_CMD
0366 E9 054F R                  JMP     NDMA_OPN
0369                    DISK_VERF       ENDP

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

0369                    FMT_TRK PROC    NEAR                    ; FORMAT TRACK  (AH = 005H)
0369 C6 46 F8 06                MOV     CMD_BLOCK+0,FMTTRK_CMD
036D EB 0A                      JMP     SHORT   FMT_CONT
036F                    FMT_TRK ENDP

036F                    FMT_BAD PROC    NEAR                    ; FORMAT BAD TRACK  (AH = 006H)
036F C6 46 F8 07                MOV     CMD_BLOCK+0,FMTBAD_CMD
0373 EB 04                      JMP     SHORT   FMT_CONT
0375                    FMT_BAD ENDP

0375                    FMT_DRV PROC    NEAR                    ; FORMAT DRIVE  (AH = 007H)
0375 C6 46 F8 04                MOV     CMD_BLOCK+0,FMTDRV_CMD
0379                    FMT_DRV ENDP

0379                    FMT_CONT:
0379 80 66 FA C0                AND     CMD_BLOCK+2,11000000B   ; ZERO OUT SECTOR FIELD
037D E9 054F R                  JMP     NDMA_OPN

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

0380                    GET_PARM_N      LABEL   NEAR
0380                    GET_PARM        PROC    FAR             ; GET DRIVE PARAMETERS
0380 1E                         PUSH    DS                      ; SAVE REGISTERS
0381 06                         PUSH    ES
0382 53                         PUSH    BX

                                ASSUME  DS:ABS0
0383 2B C0                      SUB     AX,AX                   ; ESTABLISH ADDRESSING
0385 8E D8                      MOV     DS,AX
0387 C4 1E 0104 R               LES     BX,HF_TBL_VEC

                                ASSUME  DS:DATA
038B B8 ---- R                  MOV     AX,DATA
038E 8E D8                      MOV     DS,AX                   ; ESTABLISH SEGMENT
0390 80 EA 80                   SUB     DL,80H
0393 80 FA 08                   CMP     DL,MAX_FILE             ; TEST WITHIN RANGE
0396 73 57                      JAE     G4
0398 C6 06 0074 R 00            MOV     DISK_STATUS,0           ; RESET THE STATUS INDICATOR
039D 8A EA                      MOV     CH,DL                   ; SAVE THE DRIVE
039F 80 CA 01                   OR      DL,1
03A2 FE CA                      DEC     DL
03A4 D0 E2                      SHL     DL,1                    ; GENERATE OFFSET
03A6 88 16 0077 R               MOV     PORT_OFF,DL             ; STORE OFFSET
03AA 8A D5                      MOV     DL,CH                   ; RESTORE DL
03AC 80 E2 01                   AND     DL,00000001B            ; DRIVE 0 OR DRIVE 1
03AF 8A E2                      MOV     AH,DL
03B1 E8 076D R                  CALL    PORT_0
03B4 42                         INC     DX                      ; PORT_2 ADDRESS
03B5 42                         INC     DX
03B6 EC                         IN      AL,DX                   ; READ SWITCH SETTINGS
03B7 80 FC 00                   CMP     AH,0                    ; DRIVE 0 OR 1
03BA 75 04                      JNZ     G0
03BC D0 E8                      SHR     AL,1                    ; RIGHT JUSTIFY THE SWITCH BITS
03BE D0 E8                      SHR     AL,1
03C0                    G0:
03C0 24 03                      AND     AL,00000011B            ; ISOLATE THE TABLE BITS
03C2 B1 04                      MOV     CL,4                    ; TABLE LENGTH IS 16 BYTES
03C4 D2 E0                      SHL     AL,CL                   ; ADJUST
03C6 2A E4                      SUB     AH,AH
03C8 03 D8                      ADD     BX,AX
03CA 26: 8B 07                  MOV     AX,ES:[BX]              ; MAX NUMBER OF CYLINDERS
03CD 2D 0002                    SUB     AX,2                    ; ADJUST FOR 0-N
                                                                ; AND RESERVE LAST TRACK
03D0 8A E8                      MOV     CH,AL
03D2 25 0300                    AND     AX,0300H                ; HIGH TWO BITS OF CYLINDER
03D5 D1 E8                      SHR     AX,1
03D7 D1 E8                      SHR     AX,1
03D9 0C 11                      OR      AL,011H                 ; SECTORS
03DB 8A C8                      MOV     CL,AL
03DD 26: 8A 77 02               MOV     DH,ES:[BX][2]           ; HEADS
03E1 FE CE                      DEC     DH                      ; 0-N RANGE
03E3 8A 16 0075 R               MOV     DL,HF_NUM               ; DRIVE COUNT
03E7 2B C0                      SUB     AX,AX
03E9                    G5:
03E9 5B                         POP     BX                      ; RESTORE REGISTERS
03EA 07                         POP     ES
03EB 1F                         POP     DS
03EC CA 0002                    RET     2                       ; EXIT
03EF                    G4:
03EF C6 06 0074 R 07            MOV     DISK_STATUS,INIT_FAIL   ; OPERATION FAILED
03F4 B4 07                      MOV     AH,INIT_FAIL
03F6 2A C0                      SUB     AL,AL
03F8 2B D2                      SUB     DX,DX
03FA 2B C9                      SUB     CX,CX
03FC F9                         STC                             ; SET ERROR FLAG
03FD EB EA                      JMP     G5                      ; EXIT
03FF                    GET_PARM        ENDP
```

```
                        PAGE
                        ;--------------------------------------------------
                        ; 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  :
                        ;       (1 WORD) - LANDING ZONE                    :
                        ;       (1 BYTE) - SECTORS/TRACK                   :
                        ;       (1 BYTE) - 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                           :
                        ;                                                   :
                        ;       DRIVE 0  :  DRIVE 1  :  TABLE ENTRY         :
                        ;         1/2    :    3/4     :                     :
                        ;       ------------------------------------        :
                        ;        ON ON   :   ON ON    :       0             :
                        ;        ON OFF  :   ON OFF   :       1             :
                        ;        OFF ON  :   OFF ON   :       2             :
                        ;        OFF OFF :   OFF OFF  :       3             :
                        ;                                                   :
                        ;---------------------------------------------------
                        
 03FF                   FD_TBL:
                        
                        ;-----  DRIVE TABLE 0
                        
 03FF 0132                      DW      0306D                   ; MAX CYLINDERS
 0401 04                        DB      04D                     ; MAX HEADS
 0402 0132                      DW      0306D                   ; START REDUCED WRITE CURRENT CYL
 0404 0000                      DW      0                       ; START WRITE PRECOMPENSATION CYL
 0406 0B                        DB      0BH                     ; MAX ECC BURST DATA LENGTH
 0407 05                        DB      00000101B               ; CONTROL BYTE
 0408 10                        DB      010H                    ; STANDARD TIME OUT
 0409 C0                        DB      0C0H                    ; TIME OUT FOR FORMAT DRIVE
 040A 28                        DB      028H                    ; TIME FOR CHECK DRIVE
 040B 0132                      DW      0306D                   ; LANDING ZONE
 040D 11                        DB      017D                    ; SECTORS/TRACK
 040E 00                        DB      0                       ; RESERVED
                        
                        ;-----  DRIVE TABLE 1
                        
 040F 0264                      DW      0612D                   ; MAX CYLINDERS
 0411 04                        DB      04D                     ; MAX HEADS
 0412 0264                      DW      0612D                   ; START REDUCED WRITE CURRENT CYL
 0414 0000                      DW      0                       ; START WRITE PRECOMPENSATION CYL
 0416 0B                        DB      0BH                     ; MAX ECC BURST DATA LENGTH
 0417 05                        DB      00000101B               ; CONTROL BYTE
 0418 28                        DB      028H                    ; STANDARD TIME OUT
 0419 E0                        DB      0E0H                    ; TIME OUT FOR FORMAT DRIVE
 041A 42                        DB      042H                    ; TIME FOR CHECK DRIVE
 041B 0297                      DW      0663D                   ; LANDING ZONE
 041D 11                        DB      017D                    ; SECTORS/TRACK
 041E 00                        DB      0                       ; RESERVED
                        
                        ;-----  DRIVE TABLE 2
                        
 041F 0267                      DW      0615D                   ; MAX CYLINDERS
 0421 04                        DB      04D                     ; MAX HEADS
 0422 0267                      DW      0615D                   ; START REDUCED WRITE CURRENT CYL
 0424 012C                      DW      0300D                   ; START WRITE PRECOMPENSATION CYL
 0426 0B                        DB      0BH                     ; MAX ECC BURST DATA LENGTH
 0427 05                        DB      00000101B               ; CONTROL BYTE
 0428 28                        DB      028H                    ; STANDARD TIME OUT
 0429 E0                        DB      0E0H                    ; TIME OUT FOR FORMAT DRIVE
 042A 42                        DB      042H                    ; TIME FOR CHECK DRIVE
 042B 0267                      DW      0615D                   ; LANDING ZONE
 042D 11                        DB      017D                    ; SECTORS/TRACK
 042E 00                        DB      0                       ; RESERVED
                        
                        ;-----  DRIVE TABLE 3
                        
 042F 0132                      DW      0306D                   ; MAX CYLINDERS
 0431 08                        DB      08D                     ; MAX HEADS
 0432 0132                      DW      0306D                   ; START REDUCED WRITE CURRENT CYL
 0434 0080                      DW      0128D                   ; START WRITE PRECOMPENSATION CYL
 0436 0B                        DB      0BH                     ; MAX ECC BURST DATA LENGTH
 0437 05                        DB      00000101B               ; CONTROL BYTE
 0438 28                        DB      028H                    ; STANDARD TIME OUT
 0439 E0                        DB      0E0H                    ; TIME OUT FOR FORMAT DRIVE
 043A 42                        DB      042H                    ; TIME FOR CHECK DRIVE
 043B 0150                      DW      0336D                   ; LANDING ZONE
 043D 11                        DB      017D                    ; SECTORS/TRACK
 043E 00                        DB      0                       ; RESERVED
```

```
                        ;--------------------------------------------------------
                        ;       INITIALIZE DRIVE                        (AH = 09H) :
                        ;--------------------------------------------------------

043F                    INIT_DRV        PROC    NEAR

                        ;-----  DO DRIVE ZERO

043F C6 46 F8 0C                MOV     CMD_BLOCK+0,INIT_DRV_CMD
0443 C6 46 F9 00                MOV     CMD_BLOCK+1,0           ; SET FOR DRIVE 0
0447 E8 0458 R                  CALL    INIT_DRV_R              ; SEND THE PARAMETERS
044A 72 0B                      JC      INIT_DRV_OUT            ; ERROR?

                        ;-----  DO DRIVE ONE

044C C6 46 F8 0C                MOV     CMD_BLOCK+0,INIT_DRV_CMD
0450 C6 46 F9 20                MOV     CMD_BLOCK+1,00100000B   ; SET TO DRIVE 1
0454 E8 0458 R                  CALL    INIT_DRV_R              ; SEND THE PARAMETERS
0457                    INIT_DRV_OUT:
0457 C3                         RET                             ; EXIT
0458                    INIT_DRV        ENDP

0458                    INIT_DRV_R      PROC    NEAR
0458 2A C0                      SUB     AL,AL
045A E8 057C R                  CALL    COMMAND                 ; ISSUE THE COMMAND
045D 73 01                      JNC     B1                      ;  DX = PORT 0 AFTER CALL
045F C3                         RET
0460                    B1:
0460 8C D9                      MOV     CX,DS                   ; SAVE SEGMENT

                                ASSUME  DS:ABS0
0462 2B C0                      SUB     AX,AX
0464 8E D8                      MOV     DS,AX                   ; ESTABLISH SEGMENT
0466 C4 1E 0104 R               LES     BX,HF_TBL_VEC           ; LOAD THE TABLE VECTOR
046A 8E D9                      MOV     DS,CX                   ; RESTORE SEGMENT

                                ASSUME  DS:DATA
                        ;-----------------------------------------
                        ;       DETERMINE PARAMETER TABLE OFFSET  :
                        ;       USING CONTROLLER PORT TWO AND     :
                        ;       DRIVE NUMBER SPECIFIER (0-1)      :
                        ;-----------------------------------------
046C 42                         INC     DX
046D 42                         INC     DX                      ; ADDRESS PORT 2
046E EC                         IN      AL,DX                   ; READ THE SWITCH SETTINGS
046F 8A 66 F9                   MOV     AH,CMD_BLOCK+1
0472 80 E4 20                   AND     AH,00100000B            ; DRIVE 0 OR 1
0475 75 04                      JNZ     B2
0477 D0 E8                      SHR     AL,1                    ; ADJUST
0479 D0 E8                      SHR     AL,1
047B                    B2:
047B 24 03                      AND     AL,011B                 ; ISOLATE
047D B1 04                      MOV     CL,4
047F D2 E0                      SHL     AL,CL                   ; ADJUST
0481 2A E4                      SUB     AH,AH
0483 03 D8                      ADD     BX,AX
0485 B4 09                      MOV     AH,00001001B            ; SET MASK FOR DATA MODE CPU TO CARD

                        ;-----  SEND DRIVE PARAMETERS MOST SIGNIFICANT BYTE FIRST

0487 BF 0001                    MOV     DI,1                    ; SEND MSB OF MAX CYLINDER
048A E8 04E9 R                  CALL    INIT_DRV_S
048D 72 4C                      JC      B3

048F BF 0000                    MOV     DI,0                    ; SEND LSB OF MAX CYLINDER
0492 E8 04E9 R                  CALL    INIT_DRV_S
0495 72 44                      JC      B3

0497 BF 0002                    MOV     DI,2                    ; SEND THE MAXIMUM HEADS
049A E8 04E9 R                  CALL    INIT_DRV_S
049D 72 3C                      JC      B3

049F BF 0004                    MOV     DI,4                    ; SEND MSB OF REDUCE WRITE CURRENT
04A2 E8 04E9 R                  CALL    INIT_DRV_S              ;  CYLINDER
04A5 72 34                      JC      B3

04A7 BF 0003                    MOV     DI,3                    ; SEND LSB OF REDUCE WRITE CURRENT
04AA E8 04E9 R                  CALL    INIT_DRV_S              ;  CYLINDER
04AD 72 2C                      JC      B3

04AF BF 0006                    MOV     DI,6                    ; SEND MSB OF WRITE PRECOMP CYLINDER
04B2 E8 04E9 R                  CALL    INIT_DRV_S
04B5 72 24                      JC      B3

04B7 BF 0005                    MOV     DI,5                    ; SEND LSB OF WRITE PRECOMP CYLINDER
04BA E8 04E9 R                  CALL    INIT_DRV_S
04BD 72 1C                      JC      B3

04BF BF 0007                    MOV     DI,7                    ; SEND ECC BURST LENGTH
04C2 E8 04E9 R                  CALL    INIT_DRV_S
04C5 72 14                      JC      B3

04C7 BF 0008                    MOV     DI,8                    ; LOAD THE CONTROL BYTE AND PLACE IN
04CA 26: 8A 01                  MOV     AL,ES:[BX+DI]           ;  MEMORY AT 40:76H
04CD A2 0076 R                  MOV     CONTROL_BYTE,AL

04D0 2B C9                      SUB     CX,CX
04D2 B4 0F                      MOV     AH,00001111B            ; SET THE MASK FOR STATUS MODE
04D4                    B5:
04D4 E8 068D R                  CALL    HD_WAIT                 ; GO WAIT FOR THE STATE TO HAPPEN
04D7 73 09                      JNC     B6                      ; JMP TO READ THE STATUS BYTE
04D9 E2 F9                      LOOP    B5                      ; TRY AGAIN
04DB                    B3:
04DB C6 06 0074 R 07            MOV     DISK_STATUS,INIT_FAIL   ; OPERATION FAILED
04E0 F9                         STC                             ; SET THE ERROR CONDITION
04E1 C3                         RET
04E2                    B6:
04E2 4A                         DEC     DX                      ; ADDRESS PORT 0
04E3 EC                         IN      AL,DX                   ; READ STATUS BYTE OF THE OPERATION
04E4 24 02                      AND     AL,2                    ; MASK ERROR BIT
04E6 75 F3                      JNZ     B3                      ; ERROR BIT SET?
04E8 C3                         RET
04E9                    INIT_DRV_R      ENDP
```

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

04E9                    INIT_DRV_S      PROC    NEAR
04E9 E8 068D R                  CALL    HD_WAIT                 ; GO WAIT FOR REQUEST
04EC 72 05                      JC      DI                      ;  AFTER CALL DX = PORT 1
04EE 4A                         DEC     DX                      ; ADDRESS PORT 0
04EF 26: 8A 01                  MOV     AL,ES:[BX+DI]
04F2 EE                         OUT     DX,AL                   ; WRITE THE DATA TO THE CARD
04F3                    DI:
04F3 C3                         RET
04F4                    INIT_DRV_S      ENDP

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

04F4                    RD_LONG         PROC    NEAR
04F4 E8 050E R                  CALL    CHK_LONG                ; CHECK LIMITS
04F7 72 5F                      JC      G8
04F9 C6 46 F8 E5                MOV     CMD_BLOCK+0,RD_LONG_CMD
04FD B0 47                      MOV     AL,DMA_READ
04FF EB 5D                      JMP     SHORT   DMA_OPN
0501                    RD_LONG         ENDP

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

0501                    WR_LONG         PROC    NEAR
0501 E8 050E R                  CALL    CHK_LONG                ; CHECK LIMITS
0504 72 52                      JC      G8
0506 C6 46 F8 E6                MOV     CMD_BLOCK+0,WR_LONG_CMD
050A B0 4B                      MOV     AL,DMA_WRITE
050C EB 50                      JMP     SHORT   DMA_OPN
050E                    WR_LONG         ENDP

050E                    CHK_LONG        PROC    NEAR
050E 8A 46 FC                   MOV     AL,CMD_BLOCK+4          ; LOAD THE NUMBER OF SECTORS
0511 3C 80                      CMP     AL,080H                 ; COMPARE WITH LIMITS
0513 F5                         CMC                             ; SET THE CONDITION
0514 C3                         RET
0515                    CHK_LONG        ENDP

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

0515                    DISK_SEEK       PROC    NEAR
0515 C6 46 F8 0B                MOV     CMD_BLOCK+0,SEEK_CMD
0519 EB 34                      JMP     SHORT   NDMA_OPN
051B                    DISK_SEEK       ENDP

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

051B                    RD_BUFF PROC    NEAR
051B C6 46 F8 0E                MOV     CMD_BLOCK+0,RD_BUFF_CMD
051F C6 46 FC 01                MOV     CMD_BLOCK+4,1           ; ONLY ONE BLOCK
0523 B0 47                      MOV     AL,DMA_READ
0525 EB 37                      JMP     SHORT   DMA_OPN
0527                    RD_BUFF ENDP

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

0527                    WR_BUFF PROC    NEAR
0527 C6 46 F8 0F                MOV     CMD_BLOCK+0,WR_BUFF_CMD
052B C6 46 FC 01                MOV     CMD_BLOCK+4,1           ; ONLY ONE BLOCK
052F B0 4B                      MOV     AL,DMA_WRITE
0531 EB 2B                      JMP     SHORT   DMA_OPN
0533                    WR_BUFF ENDP

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

0533                    TST_RDY PROC    NEAR
0533 C6 46 F8 00                MOV     CMD_BLOCK+0,TST_RDY_CMD
0537 EB 16                      JMP     SHORT   NDMA_OPN
0539                    TST_RDY ENDP

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

0539                    HDISK_RECAL     PROC    NEAR
0539 C6 46 F8 01                MOV     CMD_BLOCK+0,RECAL_CMD
053D EB 10                      JMP     SHORT   NDMA_OPN
053F                    HDISK_RECAL     ENDP
```

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

053F                    RAM_DIAG        PROC    NEAR
053F C6 46 F8 E0                MOV     CMD_BLOCK+0,RAM_DIAG_CMD
0543 EB 0A                      JMP     SHORT   NDMA_OPN
0545                    RAM_DIAG        ENDP

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

0545                    CHK_DRV PROC    NEAR
0545 C6 46 F8 E3                MOV     CMD_BLOCK+0,CHK_DRV_CMD
0549 EB 04                      JMP     SHORT   NDMA_OPN
054B                    CHK_DRV ENDP

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

054B                    CNTLR_DIAG      PROC    NEAR
054B C6 46 F8 E4                MOV     CMD_BLOCK+0,CNTLR_DIAG_CMD
054F                    CNTLR_DIAG      ENDP

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

054F                    NDMA_OPN:
054F B0 02                      MOV     AL,02H
0551 E8 057C R                  CALL    COMMAND                 ; ISSUE THE COMMAND
0554 72 22                      JC      G11
0556 EB 16                      JMP     SHORT   G3
0558                    G8:
0558 C6 06 0074 R 09            MOV     DISK_STATUS,DMA_BOUNDARY
055D C3                         RET
055E                    DMA_OPN:
055E E8 06A5 R                  CALL    DMA_SETUP               ; SET UP FOR DMA OPERATION
0561 72 F5                      JC      G8
0563 B0 03                      MOV     AL,03H
0565 E8 057C R                  CALL    COMMAND                 ; ISSUE THE COMMAND
0568 72 0E                      JC      G11
056A B0 03                      MOV     AL,03H
056C E6 0A                      OUT     DMA+10,AL               ; INITIALIZE THE DISK CHANNEL
056E                    G3:
056E FA                         CLI                             ; NO INTERRUPTS
056F E4 21                      IN      AL,INTA01               ; READ THE MASK
0571 24 0F                      AND     AL,0DFH                 ; ENABLE IRQ-5
0573 E6 21                      OUT     INTA01,AL               ; WRITE THE MASK OUT
0575 E8 0700 R                  CALL    WAIT_INT                ; PROCEDURE DOES STI
0578                    G11:
0578 E8 05AD R                  CALL    ERROR_CHK               ; SEE IF THERE IS AN ERROR
057B C3                         RET                             ; EXIT

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

057C                    COMMAND PROC    NEAR
057C E8 076D R                  CALL    PORT_0                  ; GET THE BASE ADDRESS
057F 42                         INC     DX
0580 42                         INC     DX                      ; ADDRESS PORT_2
0581 EE                         OUT     DX,AL                   ; ISSUE CONTROLLER SELECT PULSE
0582 42                         INC     DX                      ; ADDRESS PORT 3
0583 2B C9                      SUB     CX,CX                   ; WAIT COUNT
0585 EE                         OUT     DX,AL                   ; WRITE DMA MASK REGISTER
0586 4A                         DEC     DX
0587 4A                         DEC     DX                      ; ADDRESS PORT 1
0588                    WAIT_BUSY:
0588 EC                         IN      AL,DX                   ; READ THE HARDWARE STATUS
0589 24 0F                      AND     AL,0FH
058B 3C 0D                      CMP     AL,R1_BUSY OR R1_BUS OR R1_REQ  ; CHECK FOR BUSY,COMMAND
058D 74 09                      JE      C1                              ;  AND REQUEST BITS
058F E2 F7                      LOOP    WAIT_BUSY               ; KEEP TRYING
0591 C6 06 0074 R 80            MOV     DISK_STATUS,TIME_OUT
0596 F9                         STC                             ; SET THE ERROR CONDITION
0597 C3                         RET                             ; ERROR RETURN
0598                    C1:
0598 B9 0006                    MOV     CX,6                    ; SET FOR 6 BYTES OF COMMAND
059B 4A                         DEC     DX                      ; ADDRESS PORT 0
059C 8B F5                      MOV     SI,BP                   ; SAVE THE BASE POINTER
059E 83 ED 08                   SUB     BP,8                    ; SET FIRST BYTE OF COMMAND BLOCK
05A1 FA                         CLI                             ; NO INTERRUPTS IN COMMAND SEQUENCE
05A2                    CM3:
05A2 8A 46 00                   MOV     AL,[BP]                 ; GET A COMMAND BYTE
05A5 EE                         OUT     DX,AL                   ; ALLOW AT LEAST 2us BETWEEN EACH BYTE
05A6 45                         INC     BP                      ;  ON SENDING THE COMMAND SEQUENCE.
05A7 E2 F9                      LOOP    CM3                     ; DO MORE
05A9 8B EE                      MOV     BP,SI                   ; RESTORE THE BASE POINTER
05AB FB                         STI                             ; INTERRUPTS BACK ON
05AC C3                         RET
05AD                    COMMAND ENDP
```

```
                        PAGE
                        ;--------------------------------------------------------
                        ;                                                      :
                        ;                    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                   :
                        ;                                                      :
                        ;--------------------------------------------------------

05AD                    ERROR_CHK       PROC    NEAR
05AD A0 0074 R                  MOV     AL,DISK_STATUS          ; CHECK IF THERE WAS AN ERROR
05B0 0A C0                      OR      AL,AL                   ; ANYTHING IN AL?
05B2 75 01                      JNZ     G21
05B4 C3                         RET

                        ;-------------------------------
                        ;     PERFORM SENSE STATUS      :
                        ;-------------------------------

                                                                ; SENSE STATUS CAN BE ISSUED MULTIPLE
                                                                ;  TIMES
05B5                    G21:
05B5 C6 46 F8 03                MOV     CMD_BLOCK+0,SENSE_CMD
05B9 2A C0                      SUB     AL,AL                   ; WRITE ZERO IN INT/DMA MASK
05BB E8 057C R                  CALL    COMMAND                 ; ISSUE SENSE STATUS COMMAND
05BE 72 26                      JC      G24                     ; CANNOT RECOVER-EXIT WITH COMMAND
                                                                ;  ERROR
05C0 2B FF                      SUB     DI,DI                   ; SET INDEX POINTER TO ZERO
05C2 B9 0004                    MOV     CX,4                    ; READ FOUR BYTES
05C5 B4 0B                      MOV     AH,00001011B            ; SET MASK FOR DATA MODE CARD TO CPU
05C7                    G22:
05C7 E8 068D R                  CALL    HD_WAIT                 ; GO WAIT FOR DATA INPUT STATE
05CA 72 1A                      JC      G24
05CC 4A                         DEC     DX                      ; ADDRESS PORT 0
05CD EC                         IN      AL,DX                   ; READ THE DATA BYTE
05CE 88 43 F8                   MOV     [DI+CMD_BLOCK],AL       ; STORE AWAY SENSE BYTES
05D1 47                         INC     DI                      ; NEXT DATA LOCATION
05D2 E2 F3                      LOOP    G22                     ; LOOP TILL ALL FOUR READ.
05D4 B4 0F                      MOV     AH,00001111B            ; SET THE MASK FOR STATUS MODE
05D6 E8 068D R                  CALL    HD_WAIT                 ; GO WAIT FOR STATUS STATE
05D9 72 0B                      JC      G24
05DB 4A                         DEC     DX                      ; ADDRESS PORT 0
05DC EC                         IN      AL,DX                   ; READ THE STATUS BYTE
05DD A8 02                      TEST    AL,2                    ; SENSE OPERATION FAIL?
05DF 74 0F                      JZ      STAT_ERR                ; GO GET THE ERROR.

05E1 C6 06 0074 R FF            MOV     DISK_STATUS,SENSE_FAIL  ; SET SENSE OPERATION FAIL
05E6                    G24:
05E6 F9                         STC
05E7 C3                         RET
05E8                    ERROR_CHK       ENDP

05E8 061E R             T_0     DW      TYPE_0                  ; ERROR TYPE JUMP TABLE
05EA 062B R                     DW      TYPE_1
05EC 066D R                     DW      TYPE_2
05EE 067A R                     DW      TYPE_3

05F0                    STAT_ERR:
05F0 8A 5E F8                   MOV     BL,CMD_BLOCK+0          ; GET ERROR BYTE
05F3 8A C3                      MOV     AL,BL
05F5 24 0F                      AND     AL,0FH
05F7 80 E3 30                   AND     BL,00110000B            ; ISOLATE THE TYPE OF ERROR
05FA 2A FF                      SUB     BH,BH
05FC B1 03                      MOV     CL,3
05FE D3 EB                      SHR     BX,CL                   ; ADJUST
0600 2E: FF A7 05E8 R           JMP     WORD PTR CS:[BX + OFFSET T_0]

0605                    TYPE0_TABLE     LABEL   BYTE
0605 00 20 40 CC 80 00          DB      0,BAD_CNTLR,BAD_SEEK,WRITE_FAULT,TIME_OUT,0,BAD_CNTLR
     20
060C 00 40                      DB      0,BAD_SEEK
= 0009                  TYPE0_LEN       EQU     $-TYPE0_TABLE

060E                    TYPE1_TABLE     LABEL   BYTE
060E 04 10 02 00 04             DB      RECORD_NOT_FND,BAD_ECC,BAD_ADDR_MARK,0,RECORD_NOT_FND
0613 40 00 00 11 0B             DB      BAD_SEEK,0,0,DATA_CORRECTED,BAD_TRACK
= 000A                  TYPE1_LEN       EQU     $-TYPE1_TABLE

0618                    TYPE2_TABLE     LABEL   BYTE
0618 01 02 01                   DB      BAD_CMD,BAD_ADDR_MARK,BAD_CMD
= 0003                  TYPE2_LEN       EQU     $-TYPE2_TABLE

061B                    TYPE3_TABLE     LABEL   BYTE
061B 20 20 10                   DB      BAD_CNTLR,BAD_CNTLR,BAD_ECC
= 0003                  TYPE3_LEN       EQU     $-TYPE3_TABLE
```

```
                        PAGE
                        ;-----  TYPE 0 ERROR
                        TYPE_0:
                                MOV     BX,OFFSET TYPE0_TABLE
                                CMP     AL,TYPE0_LEN            ; CHECK IF ERROR IS DEFINED
                                JAE     UNDEF_ERR_L
                                XLAT    CS:TYPE0_TABLE          ; TABLE LOOKUP
                                MOV     DISK_STATUS,AL          ; SET ERROR CODE
                                RET

                        ;------ TYPE 1 ERROR

                        TYPE_1:
                                MOV     BX,OFFSET TYPE1_TABLE
                                MOV     CX,AX
                                CMP     AL,TYPE1_LEN            ; CHECK IF ERROR IS DEFINED
                                JAE     UNDEF_ERR_L
                                XLAT    CS:TYPE1_TABLE          ; TABLE LOOKUP
                                MOV     DISK_STATUS,AL          ; SET ERROR CODE
                                AND     CL,08H                  ; CORRECTED ECC
                                CMP     CL,08H
                                JNZ     G30

                        ;-----  OBTAIN ECC ERROR BURST LENGTH

                                MOV     CMD_BLOCK+0,RD_ECC_CMD
                                SUB     AL,AL
                                CALL    COMMAND                 ; ISSUE THE COMMAND
                                JC      G30
                                MOV     AH,00001011B            ; SET MASK FOR DATA INPUT CARD TO CPU
                                CALL    HD_WAIT                 ; GO WAIT FOR THE INPUT STATE
                                JC      G30
                                DEC     DX                      ; ADDRESS PORT 0
                                IN      AL,DX                   ; READ THE LENGTH OF THE ERROR
                                MOV     CL,AL                   ;  CORRECTED AND SAVE IN CL
                                MOV     AH,00001111B            ; SET MASK FOR STATUS STATE
                                CALL    HD_WAIT                 ; GO WAIT FOR STATUS STATE
                                JC      G30
                                DEC     DX                      ; ADDRESS PORT 0
                                IN      AL,DX                   ; READ THE STATUS BYTE
                                TEST    AL,2                    ; ERROR BIT SET?
                                JZ      G30
                                MOV     DISK_STATUS,BAD_CNTLR
                                STC
                        G30:
                                MOV     AL,CL
                                RET

                        ;-----  TYPE 2 ERROR

                        TYPE_2:
                                MOV     BX,OFFSET TYPE2_TABLE
                                CMP     AL,TYPE2_LEN            ; CHECK IF ERROR IS DEFINED
                                JAE     UNDEF_ERR_L
                                XLAT    CS:TYPE2_TABLE          ; TABLE LOOKUP
                                MOV     DISK_STATUS,AL          ; SET ERROR CODE
                                RET

                        ;-----  TYPE 3 ERROR

                        TYPE_3:
                                MOV     BX,OFFSET TYPE3_TABLE
                                CMP     AL,TYPE3_LEN            ; CHECK IF ERRGR IS DEFINED
                                JAE     UNDEF_ERR_L
                                XLAT    CS:TYPE3_TABLE          ; TABLE LOOKUP
                                MOV     DISK_STATUS,AL          ; SET ERROR CODE
                                RET

                        UNDEF_ERR_L:
                                MOV     DISK_STATUS,UNDEF_ERR
                                RET

                        ;--------------------------------------------------------
                        ; ON ENTRY AH CONTAINS THE CONTROLLER BUS STATUS DECODE :
                        ; MASK USED TO CHECK THE HARDWARE STATUS.               :
                        ;--------------------------------------------------------
                        HD_WAIT         PROC    NEAR
                                PUSH    CX                      ; SAVE CX
                                SUB     CX,CX                   ; SET THE LOOP COUNT
                        L1:
                                CALL    PORT_0
                                INC     DX                      ; PORT_1 ADDRESS
                                IN      AL,DX                   ; READ THE HARDWARE STATUS
                                AND     AL,00001111B            ; CLEAR UPPER NIBBLE OF HARDWARE STATUS
                                CMP     AL,AH                   ; CHECK THE STATE WITH THE MASK
                                JZ      L2                      ; JMP IF O.K WITH CARRY CLEARED
                                LOOP    L1                      ; TRY AGAIN
                                MOV     DISK_STATUS,TIME_OUT
                                STC                             ; SET ERROR CONDITION
                        L2:
                                POP     CX                      ; RESTORE CX
                                RET
                        HD_WAIT         ENDP
```

```
                        PAGE
                        ;--------------------------------------------------------
                        ; 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                                  :
                        ;--------------------------------------------------------

06A5                    DMA_SETUP       PROC    NEAR
06A5 80 7E FC 81                CMP     CMD_BLOCK+4,81H         ; BLOCK COUNT OUT OF RANGE
06A9 72 02                      JB      J1
                        
06AB F9                         STC                             ; SET THE ERROR CONDITION
06AC C3                         RET
                        
06AD                    J1:
06AD FA                         CLI                             ; NO MORE INTERRUPTS
06AE E6 0C                      OUT     DMA+12,AL               ; SET THE FIRST/LAST F/F
06B0 B1 04                      MOV     CL,4                    ; SHIFT COUNT
06B2 E6 0B                      OUT     DMA+11,AL               ; OUTPUT THE MODE BYTE
06B4 8C C0                      MOV     AX,ES                   ; GET THE ES VALUE
06B6 D3 C0                      ROL     AX,CL                   ; ROTATE LEFT
06B8 8A E8                      MOV     CH,AL                   ; GET HIGHEST NIBBLE OF ES TO CH
06BA 24 F0                      AND     AL,0F0H                 ; ZERO THE LOW NIBBLE FROM SEGMENT
06BC 03 C3                      ADD     AX,BX                   ; TEST FOR CARRY FROM ADDITION
06BE 80 D5 00                   ADC     CH,0                    ; CARRY MEANS HIGH 4 BITS MUST BE INC
                        
06C1 8B F0                      MOV     SI,AX                   ; SAVE START ADDRESS
06C3 E6 06                      OUT     DMA+6,AL                ; OUTPUT LOW ADDRESS
06C5 8A C4                      MOV     AL,AH
06C7 E6 06                      OUT     DMA+6,AL                ; OUTPUT HIGH ADDRESS
06C9 8A C5                      MOV     AL,CH                   ; GET HIGH 4 BITS
06CB 24 0F                      AND     AL,0FH
06CD E6 82                      OUT     DMA_HIGH,AL             ; OUTPUT THE HIGH 4 BITS TO PAGE REG
                        
                        ;------ DETERMINE COUNT
                        
06CF 8A 66 FC                   MOV     AH,CMD_BLOCK+4          ; RECOVER BLOCK COUNT
06D2 D0 E4                      SHL     AH,1                    ; MULTIPLY BY 512 BYTES PER SECTOR
06D4 32 C0                      XOR     AL,AL                   ; CLEAR LOW BYTE
06D6 48                         DEC     AX                      ;   AND DECREMENT VALUE BY ONE
                        
                        ;-----  HANDLE READ AND WRITE LONG (516D BYTE BLOCKS)
                        
06D7 80 7E F8 E5                CMP     CMD_BLOCK+0,RD_LONG_CMD
06DB 74 06                      JE      ADD4
                        
06DD 80 7E F8 E6                CMP     CMD_BLOCK+0,WR_LONG_CMD
06E1 75 0F                      JNE     J20
06E3                    ADD4:
06E3 B8 0204                    MOV     AX,516D                 ; ONE BLOCK (512) PLUS 4 BYTES ECC
06E6 53                         PUSH    BX
06E7 2A FF                      SUB     BH,BH
06E9 8A 5E FC                   MOV     BL,CMD_BLOCK+4
06EC 52                         PUSH    DX
06ED F7 E3                      MUL     BX                      ; BLOCK COUNT TIMES 516
06EF 5A                         POP     DX
06F0 5B                         POP     BX
06F1 48                         DEC     AX                      ; ADJUST
06F2                    J20:
06F2 8B C8                      MOV     CX,AX                   ; SAVE COUNT VALUE
06F4 E6 07                      OUT     DMA+7,AL                ; LOW BYTE OF COUNT
06F6 8A C4                      MOV     AL,AH
06F8 E6 07                      OUT     DMA+7,AL                ; HIGH BYTE OF COUNT
06FA FB                         STI                             ; INTERRUPTS BACK ON
06FB 8B C6                      MOV     AX,SI                   ; RECOVER ADDRESS VALUE
06FD 03 C1                      ADD     AX,CX                   ; ADD, TEST FOR 64K OVERFLOW
06FF C3                         RET                             ; RETURN TO CALLER,
                                                                ;   CY SET BY ABOVE IF ERROR
0700                    DMA_SETUP       ENDP
```

```
                        PAGE
                        ;--------------------------------------------
                        ; WAIT_INT                                   :
                        ;       THIS ROUTINE WAITS FOR THE FIXED DISK  :
                        ;       CONTROLLER TO SIGNAL THAT AN INTERRUPT  :
                        ;       HAS OCCURRED.                          :
                        ;--------------------------------------------

0700                    WAIT_INT        PROC    NEAR
                                ASSUME  DS:ABS0
FB                              STI                             ; TURN ON INTERRUPTS
8C DB                           MOV     BX,DS                   ; SAVE DS
2B C0                           SUB     AX,AX
8E D8                           MOV     DS,AX                   ; ESTABLISH SEGMENT
C4 36 0104 R                    LES     SI,HF_TBL_VEC           ; LOAD THE TABLE VECTOR

                                ASSUME  DS:DATA,ES:NOTHING
8E DB                           MOV     DS,BX                   ; RESTORE DS

                        ;-----  SET TIMEOUT VALUES

2A FF                           SUB     BH,BH
26: 8A 5C 09                    MOV     BL,BYTE PTR ES:[SI][9]  ; LOAD THE STANDARD TIME OUT
8A 66 F8                        MOV     AH,CMD_BLOCK+0
80 FC 04                        CMP     AH,FMTDRV_CMD
75 06                           JNZ     W5

26: 8A 5C 0A                    MOV     BL,BYTE PTR ES:[SI][0AH]        ; LOAD THE FORMAT DRIVE
EB 09                           JMP     SHORT   W4                      ;  TIME OUT VALUE
80 FC E3                W5:     CMP     AH,CHK_DRV_CMD
75 04                           JNZ     W4

26: 8A 5C 0B                    MOV     BL,BYTE PTR ES:[SI][0BH]        ; LOAD THE CHECK DRIVE
                        W4:                                             ;  TIME OUT VALUE
F8                              CLC                             ; CLEAR CY
B8 9000                         MOV     AX,9000H                ; DEVICE WAIT INTERRUPT
CD 15                           INT     15H
FB                              STI                             ; ENABLE INTERRUPTS FOR PC AND
                                                                ;  XT MACHINES.
2B C9                           SUB     CX,CX                   ; SET THE LOOP COUNT

                        ;-----  WAIT FOR INTERRUPT

                        W1:
E8 076D R                       CALL    PORT_0
42                              INC     DX                      ; PORT_1 ADDRESS
EC                              IN      AL,DX                   ; READ THE HARDWARE STATUS
A8 20                           TEST    AL,020H                 ; DID INTERRUPT OCCUR
75 0A                           JNZ     W2                      ; JUMP IF YES

E2 F5                           LOOP    W1                      ; INNER LOOP
4B                              DEC     BX
75 F2                           JNZ     W1                      ; OUTER LOOP

C6 06 0074 R 80                 MOV     DISK_STATUS,TIME_OUT
                        W2:
4A                              DEC     DX                      ; ADDRESS PORT 0
EC                              IN      AL,DX                   ; READ THE STATUS BYTE
24 02                           AND     AL,2                    ; ISOLATE THE ERROR BIT
08 06 0074 R                    OR      DISK_STATUS,AL          ; SAVE IN THE STATUS
83 C2 03                        ADD     DX,3                    ; PORT_3 ADDRESS
32 C0                           XOR     AL,AL                   ; ZERO
EE                              OUT     DX,AL                   ; RESET INTERRUPT MASK
C3                              RET

0755                    WAIT_INT        ENDP

                        ;---  HD_INT  ------------------------------------------
                        ;                                                       :
                        ;     FIXED DISK INTERRUPT 0DH ROUTINE IRQ-5            :
                        ;                                                       :
                        ;-------------------------------------------------------

0755                    HD_INT  PROC    NEAR
50                              PUSH    AX                      ; SAVE WORK REGISTER
B0 07                           MOV     AL,07H                  ; SET DMA MODE TO DISABLE
E6 0A                           OUT     DMA+10,AL
FA                              CLI                             ; NO INTERRUPTS
E4 21                           IN      AL,INTA01               ; LOAD THE INTERRUPT ENABLE MASK
0C 20                           OR      AL,020H                 ; TURN OFF FIXED DISK IRQ-5
E6 21                           OUT     INTA01,AL               ; REPLACE THE MASK
B0 20                           MOV     AL,EOI                  ; LOAD THE END OF INTERRUPT MASK
E6 20                           OUT     INTA00,AL               ; CLEAR THE ACTIVE INTERRUPT LEVEL
FB                              STI                             ; INTERRUPTS BACK ON
B8 9100                         MOV     AX,9100H                ; DEVICE POST
CD 15                           INT     15H                     ;  INTERRUPT
58                              POP     AX                      ; RESTORE AX
CF                              IRET
076D                    HD_INT  ENDP

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

076D                    PORT_0  PROC    NEAR
BA 0320                         MOV     DX,HF_PORT              ; BASE VALUE
02 16 0077 R                    ADD     DL,PORT_OFF             ; ADD IN OFFSET VALUE (00,04,08,0C)
C3                              RET
0775                    PORT_0  ENDP

0775                    END_ADDRESS     LABEL   BYTE
0775                    CODE    ENDS
                                END
```

Notes:

Index