Preliminary

Application Note

WD1007A – WAH/2

February 17, 1988

Dated Material/Subject to Change

WESTERN DIGITAL

TABLE OF CONTENTS

1.0 INTRODUCTION

1.1 Document Scope

This document provides the user with the necessary information to integrate the WD1007A-WAH/WA2 into the IBM PC/AT or compatible environment. This document, accompanied by a WD1007A support diskette and a data sheet forms a complete 1007A support package.

1.2 Product Description

The WD1007A-WAH and the WD1007A-WA2 are single board Winchester Disk Controllers designed to interface two ESDI compatible disk drives to the IBM PC/AT or compatible host computer. The WD1007A-WA2 supports two floppy disk drives in the AT environment. The WD1007A-WAH is a hard disk only version of the controller. The two boards use the same artwork, with the WD1007A-WAH being depopulated with the floppy controller device and support circuitry. The controller is based on the WD50C12 disk controller device and the WD12C00 host interface and buffer manager device. Overall controller operation is handled by an Intel 8053 microcontroller (WD1018). Floppy support is handled by the WD37C65 integrated floppy controller device.

1.3 Features

- Controls two ESDI hard disk drives;
- Controls two floppy disk drives (WD1007A-WA2 only);
- Supports drives with up to 2048 cylinders and 16 heads;
- Device interface transfer rates to 10MHz;
- Supports 1:1 interleave;
- 16KB (8K word) ring buffer;
- Default cacheing operation;
- Expanded command support;
- Totally AT compatible.

2.0 DESCRIPTION

2.1 Overview

This section will present some information on two areas of concern when integrating the WD1007A controllers into an IBM PC/AT environment. First, some basic information on the ESDI interface and options at the device level that affect operation of the controller. Second, information about the system parameter information required to bring the hard disk on-line.

2.1.1 ESDI Interface

ESDI is an interface standard that allows the support of storage devices of different types on the same physical interface. These devices are hard disk, tape, and optical storage. The interface is intelligent, having command and status structures for each device, and a communications protocol for controller to device information transfer. Configuration data is available from the device containing information about the physical makeup of the drive and its current configuration. The attached controller has the option to read this data for use in configuring itself to the drive. See appendix for more ESDI information.

2.1.2 Sector Modes

The ESDI specification allows for two sectoring modes to be supported; hard or soft. This section will cover only the hard sector mode, as it is the only mode supported by the WD1007A controllers. In the hard sector mode, the drive will generate a sector pulse for every sector on the drive. The number of sectors per track can be specified in one of two ways:

A. Set Unformatted Bytes/Sector Mode

This mode allows the controller to send the ESDI command SET UNFORMATTED BYTES PER SECTOR to the ESDI device. Normal operation of the WD1007-WAH/WA2 (when the W8 is not jumpered). Not allowed using the WD1005-WAH board. Note that the WD1007-WAH assumes the drive will override any jumper settings on the drive (that might dictate other values) when this command is issued. If this assumption is not valid (as in the NEC 8652 ESDI drive), the drive jumper settings will have to be set to make it accept this command.

The byte per sector value is the total number of unformatted bytes per sector, including ID fields, PLO fields, and DATA fields. The device uses this value to divide a track into a given number of sectors and generate a pulse for each. This is the mode the WD1007A controllers default to when no jumper is installed on W8. When in this mode, the WD1007A will configure the drive for 35 sectors per track unless the physical drive parameter differs from 35 SPT. See section on the 1007A BIOS for details.

B. Read Configuration Switches Mode

Normal operation of the WD1005-WAH can be invoked on the WD1007A by putting a jumper on W8. The controller reads a jumper array from the drive representing the equivalent of the bytes/sector to be used. In this mode, the WD1007A controller is completely equivalent to the WD1005-WAH.

2.1.3 Parameter Information

When integrating a hard disk into the AT environment, it is essential to be able to select the proper drive type (conforming to the physical drive parameters) from the system BIOS parameter tables. The drive type information contains the number of cylinders, heads, and sectors/track. This parameter information is used to properly communicate with the drive. Matching the physical drive parameters is essential because it maximizes the use of the storage space on the drive. If, for example, a drive is specified by the drive manufacturer to have a capacity of 300MB at 35 sectors/track, but the system BIOS is only capable of recognizing 17 sectors/track, only half of the drive's storage capacity can be utilized. There are three possible ways of assuring compatibility:

1) Use the drive in a system that has a BIOS parameter table matching the drive's;

2) Use the optional WD1007A BIOS (see section 2.2 on the WD1007A-BIOS);

3) Use third party software of vendors that have developed products allowing customization of parameter information.

Of these options, the WD1007A BIOS option is the easiest to implement in terms of flexibility, availability and adaptibility to different drive types.

2.1.4 Translation

When integrating an ESDI drive into the AT environment, certain physical parameters may exceed limitations of the system BIOS or operating system. The sectors per track parameter and the number of total cylinders are the most likely to exceed these limitations.

Some older versions of MSDOS and DOS versions distributed by OEMs, as well as Xenix and Novell software, assumed that the drive would be divided into 17 sectors on each track. This was also reflected in the parameter tables available in the system BIOS.

MS-DOS assumes that a hard disk has 17 sectors per track (SPT). This reflects the ST-506 type drives. However, the ESDI drives support 34 to 36 sectors per track with 512 bytes per track. In order to utilize the maximum storage capacity of the ESDI drives, the WD1007A-WA2 controller provides two methods of translation:

1. **Translation of 17 sectors per track mode** - For those operating systems that recognize only 17 sectors per track, it is necessary to translate physical sectors per track into logical sectors per track. This is accomplished by setting the number of logical heads requested in the SDH register to twice the number of physical heads (for a

maximum of 16 heads). This mode is invoked when the host issues a Set Parameters command with 17 in the Sector Count Register. The WD1007A-WA2 utilized the physical track in the following manner:

```
------------------------------------------------------------------------
  0  /  sectors 1 - 17       /  sectors 18 - 34                /   35
------------------------------------------------------------------------
```

Sector 0 and 35 are optional alternate sectors which are not accessed by DOS. Sectors 1 through 17 represent a logical track, i.e. logical head 0, and sectors 18 - 34 represent another logical track, i.e. logical head 1.

PHYSICAL PARAMETERS	LOGICAL PARAMETERS
1024 Cylinders	1024 Cylinders
8 Heads	16 Heads
34 Sectors/Track	17 Sectors/Track

Low-level formatting of ESDI drives must be accomplished with physical parameters of 34 to 35 sectors per track. If a format for 17 sectors per track is attempted, only the data fields will be initialized.

2. General translation mode - This mode is invoked when the physical number of cylinders exceeds 1024. By increasing the logical sectors per track to 63 and increasing the logical heads to 16, the number of logical cylinders will decrease accordingly.

For this type of drive, the following algorithm for disk address translation is used:

ABS_SEC = ((LOG_CYL * LOG_HDS) + LOG_HD) * LOG_SPT) + LOG_SEC -1

ABS_HEAD = ABS_SEC DIV PHY_SPT

PHY_SEC = ABS_SEC MOD PHY_SPT + 1

PHY_HEAD = ABS_HEAD MOD PHY_HEADS

PHY_CYL = ABS_SEC DIV PHY_HEADS

Where :

ABS_SEC is the absolute logical sector number with range 1 through 1,032,192;
LOG_HDS is the maximum number of logical heads 1 - 16;
LOG_SPT is the logical sector/track 1 - 63;
LOG_SEC is the logical sector address 1 - 63;
LOG_HD is the logical head address 0 - 15;
LOG_CYL is the logical cylinder address 0 - 1023;
PHY_HEADS is the maximum number of physical heads;

```
PHY_SPT is the disk sector/track = 34
PHY_SEC is the disk physical sector 1 - 34;
PHY_HEAD is the disk physical head 0 - 15;
ABS_HEAD is the calculated absolute head address 0 - 19114;
PHY_CYL is the disk physical cylinder address.
```

This algorithm is based on the fact that a physical cylinder contrains 510 (34 * 15) physical user sectors. Using half the number of physical cylinders as the logical cylinder gives a sector count of 1020 per logical cylinder. Dividing by 16 logical heads yields 63.75 sectors per track.

Physical to Logical Formulas

```
ABS_SEC = (((PHY-CYL * PHY HEADS) + PHY_HEAD)*PHY_SPT)+PHY_SEC
LOG_SEC = ABS_SEC MOD LOG_SPT
ABS_HEAD = ABS_SEC DIV LOG_SPT
LOG_HEAD = ABS_HEAD MOD LOG_HEADS
LOG_CYL = ABS_HEAD DIV LOG_HEADS
```

Translation Algorithm Example

The following translation example uses an HP 9753XEA ESDI:

HP 9753XEA physical parameters:

```
PHY_CYLS = 1600 Cylinders          1224
PHY_HEADS = 12 heads               7
PHY_SPT = 32 sectors per track (actually 64,256 byte sectors) 34
```

By transposing these values into the following formula, this drive can offer 614,400 sectors:

```
ABS_SECS = PHY_HEADS * PHY_SPT
ABS_SECS = 1600 * 12 * 32 = 614,400     291,312
```

The following formula determines the logical parameters:

```
ABS_SECS = LOG_CYLS * LOG_HEADS * LOG_SPT
```

If the logical sectors per track (LOG_SPT) is 63 and the logical number of heads (LOG_HEADS) is 16, then to determine the logical number of cylinders (LOG_CYLS):

```
LOG_CYLS = ABS_SECS / (LOG_HEADS * LOG_SPT)
LOG_CYLS = 614,400 / (16 * 63) = 609          291,312/(16*63) = 289
```

The resulting logical parameter table for the HP 9753XEA would be:

LOG_CYCLS = 609 cylinders
LOG_HEADS = 16 heads
LOG_SPT = 63 sectors per track (512 byte sector)

This yields a 314.3 Mbyte drive. The drive would actually have a capacity of 314.5 Mbytes if it could be accessed physically. Only read and write data operations use these logical parameters. Low level initialization and alternate sector utilization must be accomplished physically. When using diagnostics, such as IBM ADVANCED DIAGNOSTICS, and a format track command is issued with a sector count not equal to the physical sectors per track, determine the starting physical sector and write a pattern of zeros in the data field of all sectors on that logical track.

Translation can be disabled by using the translation override jumper, W14. This requires that the parameters passed from the system must be the same as the parameters of the drive.

2.1.5 Floppy Support

Floppy disk drive support is provided by the WD37C65 integrated floppy disk controller chip. The device is fully compatible with the NEC 765 floppy controller chip, thus maintaining full AT BIOS compatibility. The controller will support combinations of 5.25" and 3.5" drives within the following key guidelines:

a. Only two floppies can be installed at any one time.

b. If using 3.5" drives it must be verified that the system BIOS supports the given version (either 720 Kb or 1.44 Mb) of the drive. Appropriate steps, including 'Setup' and the use of 'Driver.Sys' type device drivers, might have to be taken to assure that the system BIOS recognizes the drive.

c. When using 1.44 Mb drives one MUST make sure that the drive is of the type referred to hereon as the 'ANSI' type. This means that drive must be AT compatible and have a 'no care' condition on pin 2 at the floppy interface.

2.1.6 Track Cacheing

The WD1007A incorporates a track cacheing function to improve disk subsystem performance. This feature can be more clearly defined as a look-ahead read. When the host issues a read command to the controller, single or multi-sector, the controller will perform the read of the sector or sectors. While the host is offloading this data, the controller firmware will initiate the look-ahead read function. The controller will continue to read sectors from the track where the read command was started. The controller will continue to read sectors until the buffer is full or the end of the track is reached. This cacheing routine is done in anticipation of a read of another sector or sectors on that track by the operating system. The cacheing read will be halted if a command is issued to the controller that is not a read command of the track that initiated the cacheing read. The cacheing feature can be disabled for those applications that may not gain performance from this feature. The WD1007A control firmware supports a non-standard command called "Cache Control". This command is used to enable or disable the cache feature. By writing a value to the Write Precomp Register and issuing the command to the Command Register, cacheing can be controlled by the host. To enable cacheing, an AAH is written to the precomp register, and a 55H will disable the feature. Writing any other value to this register aborts the command and the state of cacheing remains the same. The WD1007A default is cacheing enabled.

2.2 BIOS Notes

There are two different BIOS' that play a role in the use of the WD1007A with ESDI drive: the System BIOS and the 1007A BIOS.

A. The System BIOS.

1. The system BIOS refers to the BIOS that is controlling the computer (e.g. Phoenix, IBM-AT, Faraday). If the appropriate drive parameter tables are present, the system BIOS can be used, along with the operating system (MSDOS, SCO Xenix) to partition and high level format a given drive.

The easiest way to determine if a system BIOS has the physical drive parameters required for the ESDI drive is to use the appropriate set-up diskette and see if the appropriate heads, number of cylinders and number of sectors/track are listed. If they are not listed, the BIOS probably does not have the required parameter tables and one needs to use other means (such as WDFMT or the WD1007A BIOS) to low level format.

Another way of examining the system parameter tables to determine the drive type number for the given ESDI drive, is through the DOS Debug Utility.

Although some systems locate drive parameters in other locations, the standard location in the AT BIOS is F000:E401. The proper command at the DEBUG prompt (-) is as follows:

```
df000:e401 <CR>
```

This corresponds to drive type 1 and will display 8 lines of data from the BIOS ROM. By entering another "d" at the DEBUG prompt, another 8 lines of data will be displayed, corresponding to drive type 2, etc. This procedure is continued until all drive parameter information is displayed. Each parameter table has 16 bytes of data. Data from the BIOS ROM is displayed in hex, least signifcant byte first. This data breaks down as follows:

Bytes 1&2	Number of cylinders
Byte 3	Number of heads
Bytes 4&5	Not Used
Bytes 6&7	Write Pre-Compensation cylinder
Byte 8	Not Used
Byte 9	Control Byte (= 08H for > 8 heads)
Byte 10-12	Not Used
Bytes 13&14	Landing Zone3
Byte 15	Sectors Per Track
Byte 16	Not Used

Another feature of the DOS DEBUG utility is in the user's ability to confirm the parameters for the current drive selected. This is done through the INT 41 (located at d0:104) and the INT 46 (located at d0:118) handlers. INT 41 corresponds to the address of the current selected parameters for drive) and INT 46 corresponds to the address of the current selected parameters for drive 1. For example, if one desires to see the current selected parameters for drive 0, the proper command at the DEBUG prompt (-) is:

d0:104

The first four bytes displayed correspond to the offset (2 bytes) and segment 6 (2 bytes) of the location where the selected parameters are, e.g., if after the above d0:104 the result is:

```
30   IF     00   C8    AB   73  . . . <=> Location DC800:1F30
[      ]    [      ]
Offset      Segment
```

The current selected parameters are located at address C800:1F30. By typing this at the DEBUG prompt, 16 bytes will be displayed (as explained above) describing the selected drive parameters.

B. 1007A BIOS

1.0 General Notes

The WD1007A controllers have an optional BIOS rom that provides functions to help the user integrate the controller with an ESDI drive into an IBM PC/AT or compatible. The BIOS provides drive parameter tables, low-level formatting routines, and surface analysis routines. The parameter information is needed to support the drive and controller in systems that do not support the parameters needed for this combination in their system BIOS. Low-level formatting and verification routines are used to prepare the drive for use by the operating system. Part of the optional BIOS is a "Shadow Ram". This is a static memory device that resides in the upper 256 bytes of the BIOS address space. This memory is used to store the parameter information generated by the BIOS in an area outside system memory. The BIOS resides in the external I/O BIOS address space. There are four address ranges available, selected by configuring the jumpers at W1-W2. See Figures 1-5 for a complete flowchart description of the 1007A BIOS and Figure 6 for the addressing scheme implemented for the shadow RAM.

2. Parameter Tables

When integrating a hard disk into the AT environment, a drive type table must be selected that represents the physical characteristics of the drive being used. This information consist of the number of cylinders, heads, and sectors per track. It is often difficult to match the drive and the table exactly. When integrating an ESDI hard disk, the number of sectors per track becomes the most important parameter to match. Most system BIOS roms support only the older MFM/ST506 drives that used 17 sectors per track. The ESDI disk drive will have typically 34 sectors per track when operating in the hard sector mode (required by the 1007A controllers). The BIOS generates the needed information by using the ability of the ESDI drive to present the actual drive characteristics to the controller. The BIOS reads the ESDI information and generates the appropriate logical parameter tables. Since all low level formatting is done at 35 SPT (in order to accomodate the extra 'Alternate' sector feature) the parameter tables generated will make all their calculations based on this 35 SPT value. Parameter tables are constructed for all the translation features of the controller.

3. Options.

The 1007A BIOS offers the following features to the user:

A. Drive type routine

The drive types can be changed by using the "+" and "-" keys. Assuming that all the ESDI drives in question are specified at 34 SPT, there are four possible choices that can be implemented for a drive:

1. No drive present - the BIOS will automatically select a drive type 0 if there is no drive present. The user will get the message "* NONE SELECTED OR NO DRIVE PRESENT ! ***" next to the drive number.**

2. A selection with 17 SPT. This feature should be used when the drive is being used in a system that does not recognize drives with SPT values other than 17 SPT. Although the low level format will still be at 35 SPT, logical parameter tables will be created reflecting 17 SPT (with translation enabled). See section on Translations for more details.

3. A selection with 34 SPT. This feature should be used if the system being used recognizes the standard 34 SPT drive.

4. A selection with 63 SPT. This feature should be implemented only when the drive in question is specified as having greater than 1024 cylinders. Translation should always be enabled when such a condition exists. What this feature does is to allow full use of the all the cylinders of the drive (even though most AT BIOS' only recognize 1024 cylinders as a maximum) through a translation scheme that uses 63 SPT. See section on Translations for more details.

B. Low level Format Routine

Formatting routines are present to do the low-level initialization of the disk surface. The drive is formatted at 35 sectors/track (SPT). Transparent to the user, the format routine formats with a sector skew and also formats a spare sector on each track. This sector is used by the surface analysis routines to provide the ability to reallocate a bad sector on a track. It is also used to store the parameter information generated by the BIOS. This information is written to the spare sector on cylinder 0, head 0. The sector skew, which is fixed at two, allows the controller to maintain a one to one interleave accross all head boundaries. Sector Skewing is a method of formatting in which the sector numbers are rotated in the interleave table for each track (see Appendix 7.0).

C. Enter Defect List Routine

This routine allows the user to enter the list of 'bad tracks' as listed by the drive manufacterer on the drive.

D. Surface Analysis Routine

A surface analysis routine is available that identifies bad tracks on the drive and in the event that there is only one sector bad on the track (and it is NOT sector zero), assigns the alternate sector (sector 35) in the place of the bad one. This 'saves' the track from being marked bad by the controller.

If one had entered a list of the 'bad tracks' at the begining of this routine, after the surface analysis all the tracks marked bad will now be error free (within the constrictions of the above paragraph).

E. Verify Drive Routine.

The verify routine will identify all the 'bad tracks' on the drive and list them by head and cylinder number.

4. User Interface

The user gains access to the BIOS from the DOS DEBUG utility by giving the command "G=C800:5". This causes instructions in the BIOS rom to be executed starting at location 5 of the segment address. Note that the segment address may change depending on the BIOS address jumpers W1 and W2. When execution is started the user is presented with the following simple menu screen:

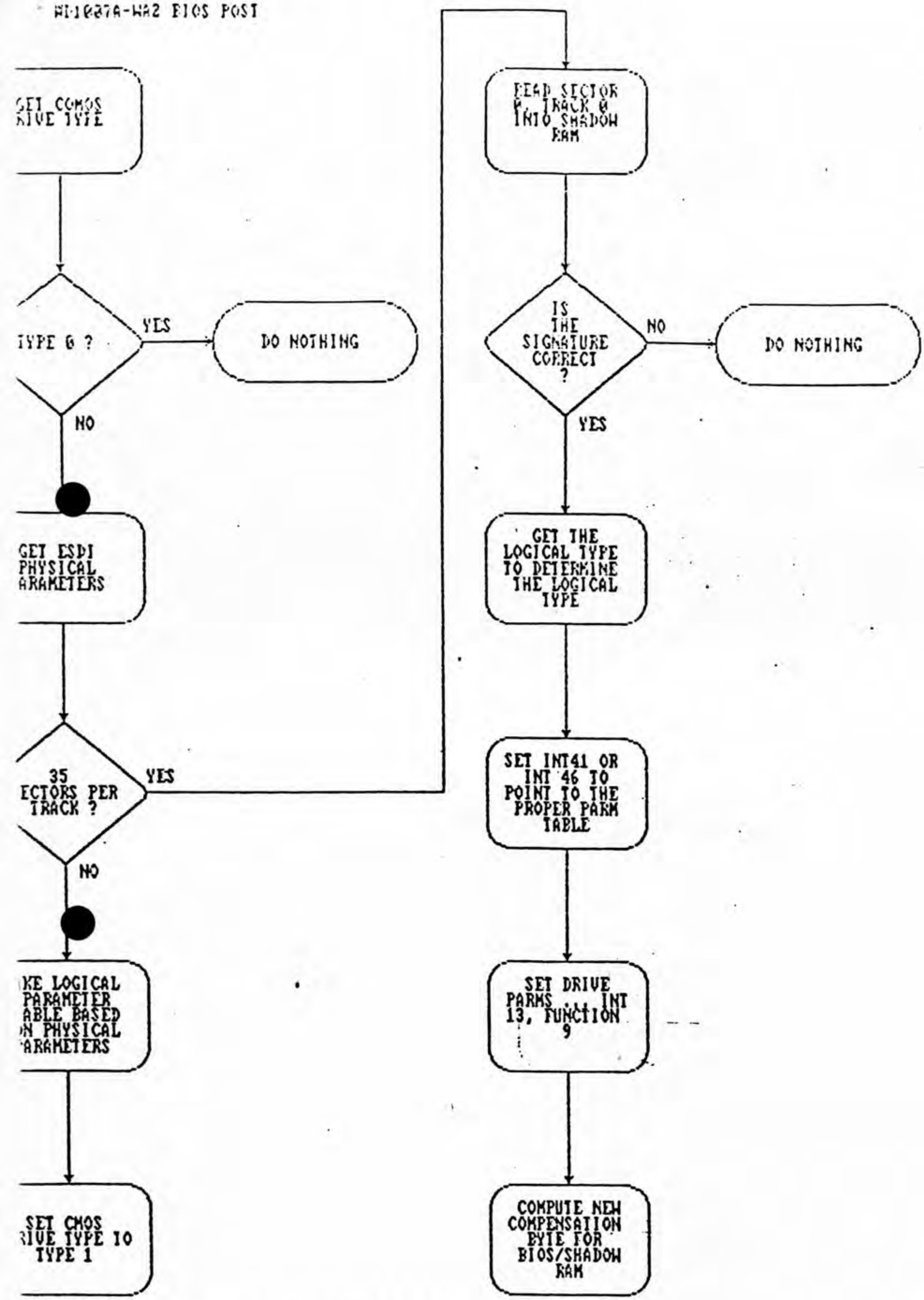
```
*** Western Digital 1007A-WA2 Initialization Utilities, Rev. 1.0***

PRESENT DRIVE SETUP ... + or - to change, <ENTER> for selection

DRIVE 0 CYLINDERS XXX HEADS XX PRECOMP CYLINDER XXX SPT XX
DRIVE 1 CYLINDERS XXX HEADS XX PRECOMP CYLINDER XXX SPT XX


Change Drive Types     ---> 1
Low Level Format       ---> 2
Surface Analysis       ---> 3
Verify Drive           ---> 4
Enter Defect List      ---> 5
Exit and Reboot        ---> 6


Enter Choice (1-6) -->
```

All functions can be executed by simply entering the number for the desired routine. The BIOS will execute that routine and then prompt the user to press a key to return to the main menu. Note that all changes made while in the BIOS will only be finalized after a proper exit through function 6 - otherwise some features might not be valid including drive type selected.

If the drive physical parameters are specified at a value other than 34 SPT, the 1007A BIOS will allow the board to function. The BIOS will read the drive parameters off the drive and if it recognizes the SPT value as being different from 34 it will allow a low level format at the read values with no skew and no alternate sector and at 1:1 interleave.

5. 1005 (1005-WAH) Mode As Opposed To The 1007 (1007A) Mode.

If a drive has been formatted with the WD1005-WAH it has been formatted at 34, 35 or 36 SPT. In the 1005 (1005-WAH) mode, the number of SPT were read from the drive (as opposed to the 1007 mode where the controller specifies the SPT through the 'set the unformatted bytes/sector' command). If one desires to use the 1007A in the 1005 mode, the following two facts should be true:



NOTE: This procedure is performed for both drives 0 and 1.

FIG. 1

WI1007A-WA2 BIOS SETUP

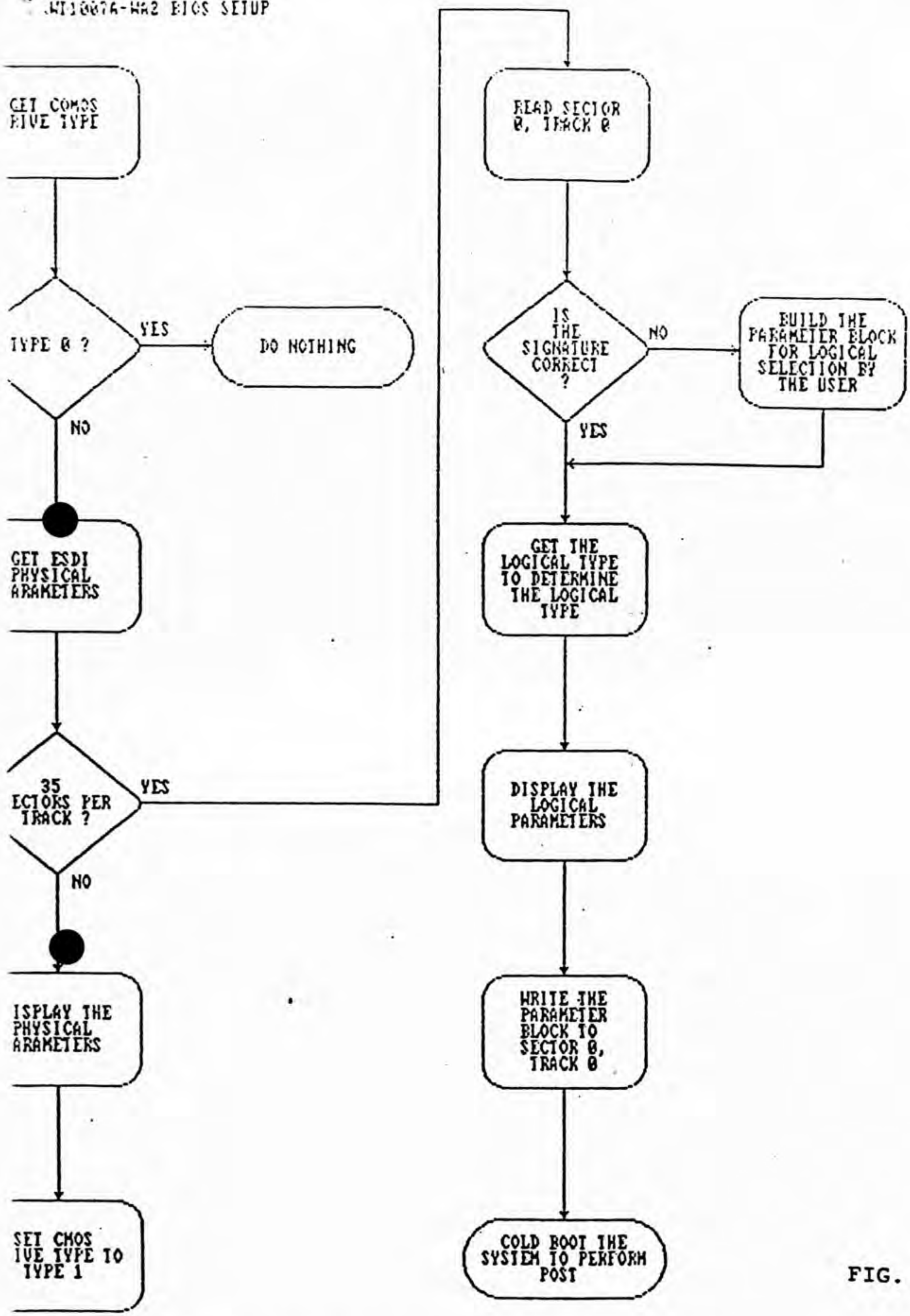


FIG. 2

NOTE: Except for the cold boot, the same procedure is performed for bot drives 0 and 1..

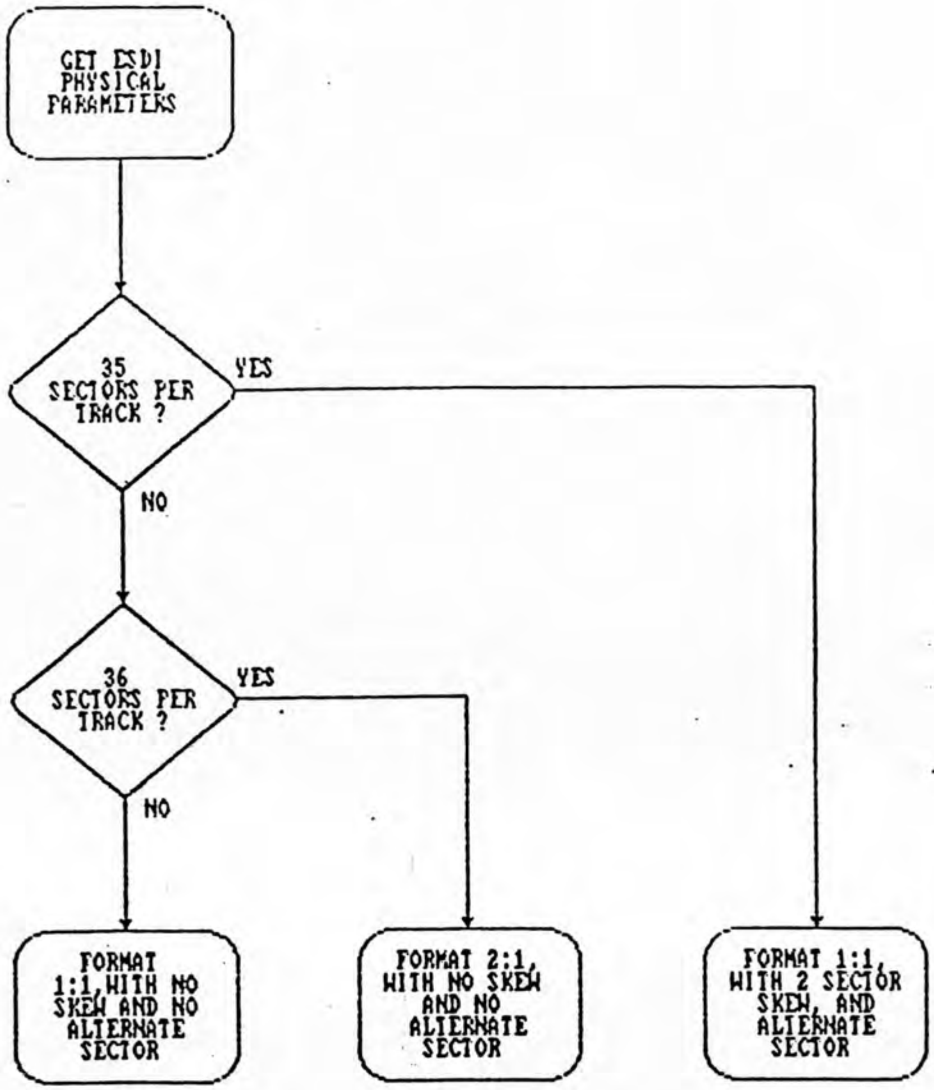


FIG. 3

NOTE: The skew factor of 2 allows for head switch time ... there is no compensation for a cylinder change. The alternate sector is represented with an ID of 0, thereby providing sectors 1-34 for system use.

WD1007A-WA2 BIOS SURFACE TEST

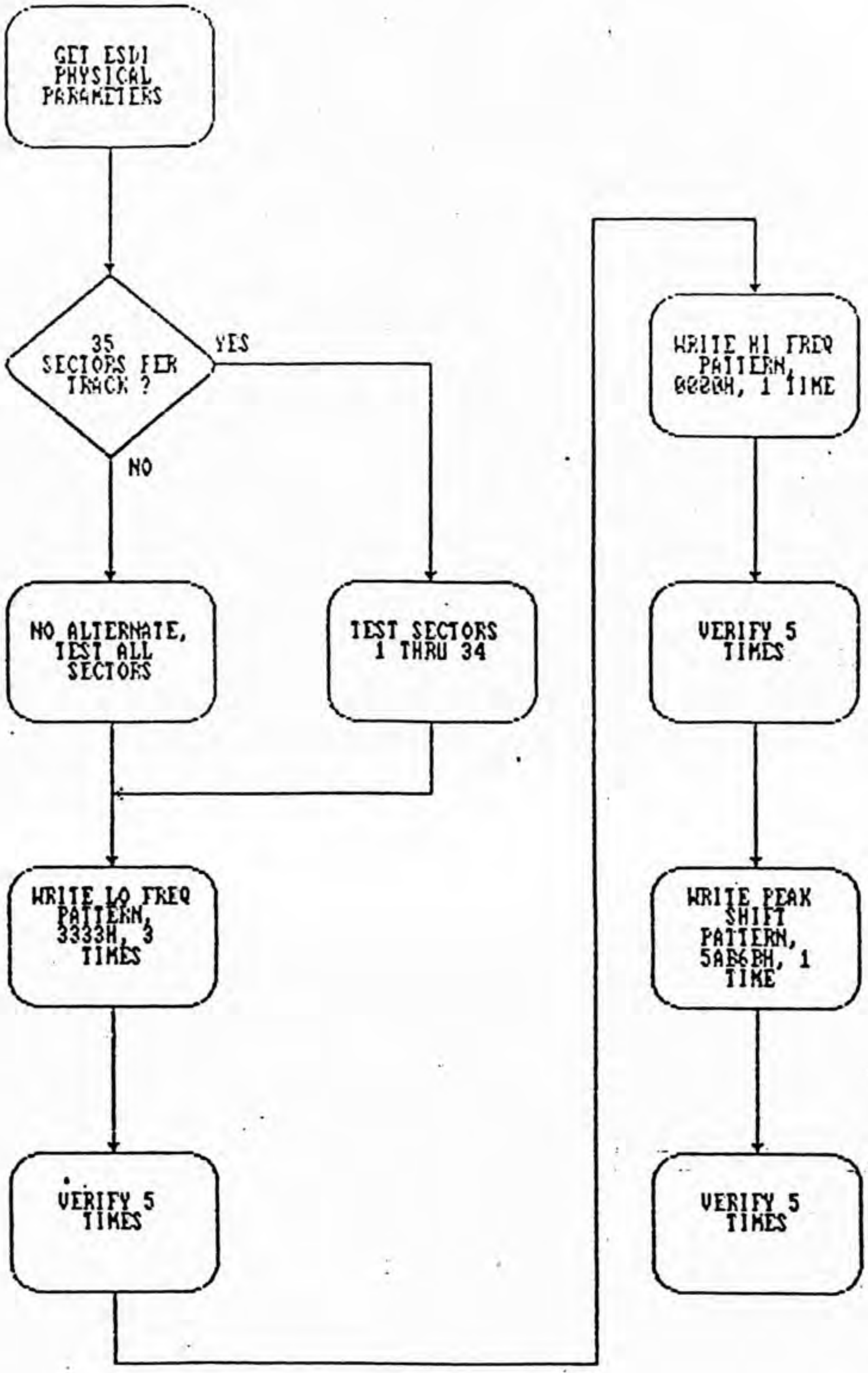
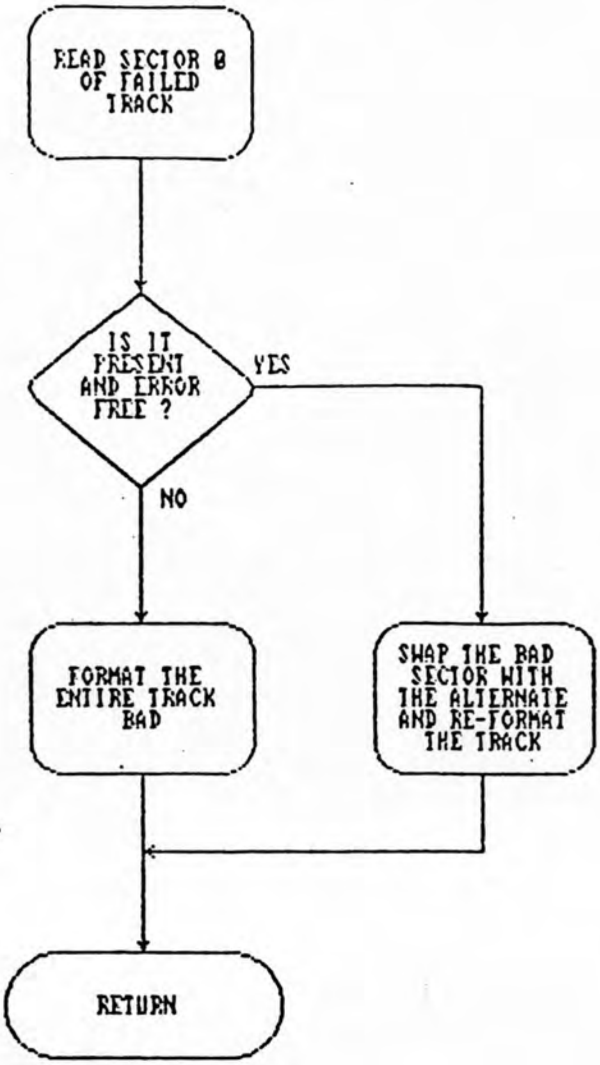


FIG. 4

NOTE: All operations are performed on a track ... if, at any time, there is an error, an alternate is assigned, or format bad track, and the test is resumed from the start of the failed test.

WD1007A-WA2 BIOS ALTERNATE ASSIGNMENT



NOTE: When the alternate is swapped with the bad sector, it is re-assigned the ID of 6FFH. The skew factor is accounted for during the re-assignment.

FIG. 5

RESS	BIOS ROM ... 2764 EPROM	SHADOW RAM ... 256 BYTES
	55h, 0AAh - BIOS signature 10h - length, # of 512 byte blocks (8k) JMP INIT - entry point to POST routine JMP SETUP - entry point to SETUP routine	
0		DRIVE 0 SIGNATURE
0		DRIVE 0 TYPE (INDEX TO LOGICAL)
0		DRIVE 0 PHYSICAL TABLE
0		DRIVE 0 LOGICAL TABLE ... 34 SP
0		DRIVE 0 LOGICAL TABLE ... 17 SP
0		DRIVE 0 LOGICAL TABLE ... 63 SP
0		DRIVE 1 SIGNATURE
0		DRIVE 1 TYPE (INDEX TO LOGICAL)
0		DRIVE 1 PHYSICAL TABLE
0		DRIVE 1 LOGICAL TABLE ... 34 SP
0		DRIVE 1 LOGICAL TABLE ... 17 SP
0		DRIVE 1 LOGICAL TABLE ... 63 SP
F	COMP BYTE - required to insure that modulo hex 100 checksum equals 0	COMP BYTE

E:
t system RESET, the shadow ram is disabled, and the entire 8k of ram is enabl
dummy write to the shadow ram enables it and disables that portion of the ro
f course, the compensation byte will be different, therfore, must be calculat

FIG. 6

a. The configuration switches on the drive should be set to the appropriate SPT value.

b. Jumper W8 should be present.

This allows one to make the 1007A fully equivalant to the 1005-WAH for all operations. As such all drives that have been formatted using the 1005-WAH can be read/written to using the 1007A.

6. Drives With Physical Parameters That Are Greater Than 34 SPT.

There are some ESDI drives in the market that are specified at greater than 34 SPT. This usually implies that the drive functions at 15 MB/s data rate. The 50C12, which is the 1007A hard disk controller, is specified to function at up to 10 MB/s. As such it is recommended that the 1007A NOT be used with drives that require greater than 10 MB/s data rates.

If the above drive (with the greater than 36 SPT specification) functions at 10 MB/s, the 1007A can successfully be used, if the system the drive is being used in recognizes the physical drive parameters explicitly.

2.3 Operation

The WD1007A controller can be installed into any 16 bit slot in the IBM PC/AT or compatible host computer. By referring to the jumper option, the controller can be configured for the desired mode of operation. Be sure that the drive configuration is set properly to support the hard sector mode. Connecting cables for the ESDI hard disk are the same as those used on the ST506 interface disks. Also, device ID jumpers are implemented on the ESDI device just as with ST506 drives. In order to use an ESDI drive in a given system, a three step process has to be followed:

1) Low level format;
2) Active drive partitioning through operating system;
3) High level format.

See the 1007A Users Guide for a more detailed explanation on the mechanics of installing and running and ESDI disk controller using the 1007A board.

A. Low Level Format

After the proper parameter information has been selected (see 2.1.3), the hard drive must be initialized through a low level format.

a) If the WD1007A controller does not have a BIOS, it is recommended that the user use routines available in third party software or use WDFMT (a formatting utility available on diskette from Western Digital as part of the support package for the 1007A).

b) If the WD1007A has a BIOS, low level formatting can be done through the BIOS (see section 2.2).

B. Active Drive Partitioning Through the Operating System

When the disk drive has been low-level formatted, it is now ready to be prepared for use by the operating system. Some operating systems handle disk usage differently so their methods of preparing the disk are different. We will cover the two main operating systems used today, MSDOS and SCO Xenix.

1. MSDOS (FDISK and FORMAT)
These two utilities are used to prepare the disk for use by the MSDOS operating system. In order to use the typical large capacity ESDI drives under MSDOS, the user must use version 3.3 or a compatible version that supports the division of a large physical drive into smaller logical volumes. It must be noted that the MSDOS operating system has a limit of 33.3 Megabytes per volume. If the version of MSDOS does not support the creation of multiple logical drives, disk enhancement software drivers must be used to obtain full usage of the high capacity drives.

2. SCO Xenix
The Xenix operating system does not limit the size of the disk volume. During the installation of the Xenix operating system, the user is prompted for information on how the disk is to be used. The entire disk may be used or a partition for Xenix and one for DOS may be created. The Xenix installation will perform all tasks comparable to the DOS FDISK and FORMAT utilities. A unique feature of SCO Xenix version 2.1 is the ability to create a parameter table for the drive outside of the system BIOS. The user is prompted at installation time as to what parameters are to be used and allows them to be customized for application requirements. Previous versions of the operating system assumed 17 sectors per track on the drive regardless of the parameter table information presented. These versions require the 17 sector translation operation mentioned above.

C. High Level Format

The drive can now be high level formatted with the given operating system through the 'Format C: (or D:)/S' command.

APPENDIX

1. Commands

A. Standard AT Commands

Command	Op Code
Read sector	20H
Write Sector	30H
Restore (Recal)	1XH
Set Parameters	91H
Format Track	50H
Diagnostics	90H
Seek	7XH
Read Verify	40H

Read Sector - A number of sectors (1-256) are read from the selected disk. If the drive is not positioned at the specified cylinder, an implied 'seek' will occur. Drive furnished ECC check bits will be used if the Read Long mode is specified. Single burst data errors (up to 11 bits) will be corrected if retries are enabled and the long mode is not selected. Uncorrectable errors do not inhibit the (error sector) data transfer, however multi-sector transfers will terminate. The WFDC interrupt occurs as each sector is ready for system input. The WFDC also caches the remaining sectors until the buffer RAMs are full upon completion of a successful Read Command. When the next Read Command occurs, and if the desired sectors are the same as the cached sectors, the data transfer occurs immediately. cacheing therefore improves the data throughput by reducing the disk access time. IRQ14, BSY and DRQ bits of HDSTT register operate in the same fashion as in non-cache operations.

Write Sector - A number of sectors (1-256) are written to the selected disk with an implied ~~seek~~ occurring, if required. Multiple sector write (and read) operations may cross track and cylinder boundaries. The Write Long mode appends the ECC bytes to the data supplied by the system processor. The data request bit (On) along with the command cause the system processor to output the contents of the first data buffer. An interrupt occurs as the data for each subsequent sector is required.

Read Verify - This command verifies that a previous write command is correct by checking ECC bytes. The host processor does not input read data. The command may be used with multi-sector operations. An error condition will abort a multi-sector verify operation. The retry command may be used with this command.

Seek - The Seek command positions the drive heads over the cylinder specified in the Task File registers (HDCLH/L) and clears any track offsets. The command aborts under the conditions noted for the Restore command above. Bit SKC of the HDSTT register sets true upon the completion of a Seek command. The fixed disk priority interrupt (IRQ14) issues after a successful ESDI Seek command transfer. The host can check for completion of the seek operation by checking bit SKC of HDSTT register.

Restore - The selected ESDI drive received a Seek-to-Cylinder 0 command via the serial command interface (WD1018 port 1.5). The drive heads seek to cylinder 0 and any track offsets are clear. The command aborts when the ERR bit sets in the status register. The Aborted Command (ACD) bit sets in the error register if the WD1018 receives an Attention interrupt from the drive indicating a transfer protocol or transfer parity error.

Diagnostics - The diagnostics command causes the WD1018 to execute an on-board diagnostic program and to report the test results to the WD50C12 Error Register. See Appendix for WFDC self tests.

Set (Drive) Parameters - This command communicates drive parameters to the controller. It selects the had, cylinder, and sector for each drive. The WD1018 uses the drive parameters in the execution of multi-sector commands and in evaluating legal controller commands.

Format Track - The Task File specifies the track to be formatted with identification, data, and check fields in accordance with the interleave table transferred to the sector buffer. The interleave table is composed of two bytes per sector, with the first byte set to "00", for a good sector or "80h" for a bad sector. The second byte designates the logical sector number. The Task File (HDSCT and HDSDH) specifies sectors per track and sector size. Command completion initializes the data field to 'zeros' and appends four ECC bytes after the data field. The Completion Interrupt occurs as each track is formatted. The WD1007A-WA2 controller forces 512 bytes/sector, 35 sectors/track, and the hard sectored drive format.

B. Non-Standard AT Commands

Command	Op Code
Write Data Stack	E8H
Cache Control	EFH
Read Parameters	ECH
Read Data Stack	E4H
Initiate ESDI	E0H

Read Data Stack - This diagnostic command allows the host to read the sector buffer without executing a disk read operation. This command does not generate an interrupt upon completion.

Write Data Stack - This diagnostic command allows the host to write data to the sector buffer without executing an actual disk write command. This command does not generate an interrupt upon completion.

Initiate ESDI - This command allows the system processor to send instructions directly to the selected drive by loading the AMAC's cylinder register and executing the Initiate command. The host must load AMAC's cylinder registers (high and low data bytes) with the command it wants the drive to execute prior to issuing the Initiate ESDI command. The controller serializes the data, adds the required parity bit and transmits the instruction to the drive. The drive completes the instruction and transmits completion status to the controller. The drive's completion status data is then stored back into AMAC's cylinder registers for host access.

The following examples show how the Initiate ESDI command is used:
Using ARM (Western Digital Test Software) when * appears, type 'MAKE' and 'ENTER'.
Monitor will display:

> Enter Task File Parameters: All entries are in hex
> Enter Command =
> Enter Cylinder # =
> Enter Head # =
> Enter Sector =
> Enter 'EO' for the command.
> Enter '20' (Request Standard Status) for the Cylinder number.

The high byte (bits 15 - 8) of the ESDI command will go to Cylinder High REgister, and the low byte (bits 7 - 0), if applicable, will go to Cylinder Low Register. Refer to ANSI ESDI Specification, Document No. X3T9.3/8X, for all ESDI commands.

The host can now access the drive's status by reading the cylinder register's addresses (AF4 - AF5). The high order status byte is stored in the Cylinder High Register: the low order status byte is stored in Cylinder Low Register.

Using DEBUG enter:

```
01F4 00
01F5 20    ;Output Read Status Command
01F7 E0    ;Initiate ESDI Command
11F4
11F5    ;Read Drive Status
```

Cache Control - This command (EF) allows the user to enable or disable cacheing. By writing AA (enable) or 55 (disable) into the Write Precomp Register (1F1), then issuing the Cache Control command, cacheing will be turned on or off accordingly. The command will abort if any other code is written into 1F1, and cacheing stays unchanged. cacheing is enabled in default.

Read Parameters - This command causes the MC8753 to store 49 words of drive and controller parameters into the sector buffer for host access.

A. ESDI Config Information

General configuration

fixed cylinders
removable cylinders
heads
unformatted bytes/track
unformatted bytes/sector
sectors/track
minimum bytes in ISG
minimum bytes in PLO
words of vendor unique status

B. Controller Information

serial # (20 ASCII characters...0 = not specified)
controller type...0 = not specified
1 = single ported, multi sector buffer
2 = dual ported, multi sector buffer
3 = dual ported, multi sector buffer, cache

controller buffer size in 512 byte increments
# of ECC bytes xferred on long operations
controller firmware revision (8 ASCII characters)
# of sectors xferred per interrupt on read commands
double word capability (0 = not capable, 1 = capable)

2.0 ESDI Commands

CMD Function Bit 15 14 13 12	CMD Function Definition	CMD Modifier Applicable Bits 11-8	CMD Parameter Applicable Bits 11-0	Status/conf returned to controller
0 0 0 0	Seek	No	Yes	No
0 0 0 1	Recalibrate	No	No	No
0 0 1 0	Request Status	Yes	No	Yes
0 0 1 1	Request Configuration	Yes	No	Yes
0 1 0 0	Select Head Group*	No	Yes	No
0 1 0 1	Control	Yes	No	No
0 1 1 0	Data Strobe Offset*	Yes	No	No
0 1 1 1	Track Offset*	Yes	No	No
1 0 0 0	Initiate Diagnostics*	No	Yes	No
1 0 0 1	Set Bytes per Sector*	No	Yes	No
1 0 1 0	Reserved	-	-	-
1 0 1 1	Reserved	-	-	-
1 1 0 0	Reserved	-	-	-
1 1 0 1	Reserved	-	-	-
1 1 1 0	Set Configuration*	No	Yes	No
1 1 1 1	Reserved	-	-	-

* - Optional commands

3.0 ESDI Request Configuration Data

Command Modifier Bits 11	10	9	8	Function
0	0	0	0	General Configuration Response bits
0	0	0	1	Number of Cylinders-Fixed
0	0	1	0	Number of Cylinders-Removable
0	0	1	1	Number of Heads
0	1	0	0	Unformatted Bytes per Track
0	1	0	1	Unformatted Bytes per Sector(Hard Sector mode)
0	1	1	0	Number of Sectors per Track(Hard Sector mode)
0	1	1	1	Bytes in ISG Fieldc
1	0	0	0	Bytes per PLO Sync Field
1	0	0	1	Number of Vendor Unique Status Words Availble
1	0	1	0	Reserved
1	0	1	1	Reserved
1	1	0	0	Reserved
1	1	0	1	Reserved
1	1	1	0	Reserved
1	1	1	1	Vendor Identification

4. ESDI General Configuration Response Word

Bit Position	Function
15	0=Magnetic Disk Drive 1=Not Magnetic Disk Drive
14	1=Format Speed Tolerance Gap required
13	1-Track Offset Option available
12	1=Data Strobe Offset Option available
11	1=Rotational Speed Tolerance is >0.5%
10	1=Transfer Rate > 10Mhz
9	1=Transfer Rate > 5Mhz 10Mhz
8	1=Transfer Rate 5Mhz
7	1=Removable Cartridge
6	1=Fixed Drive
5	1=Spindle Motor Control Option Implemented
4	1=Head Switch Time > 15 sec
3	1=Not MFM
2	1=Controller Soft Sectored(Address Mark)
1	1=Hard Sectored(Sector Pulses)
0	Reserved

5.0 ESDI Status Response Bits

Bit	
15	Reserved
14	1=Removable Media Not presetn 0=If Not Removable
13	1=Write Protected-Removable Media 0=If Not Removable
12	1=Write Protected-Fixed Media
11	Reserved
10	Reserved
9	1=Spindle Motor Stopped by Stop command 1=Spindle Motor Stopped for Other(e.g. reset)
8	1=Power On Reset Conditions Exist(reconfiguration or Spindle Motor command may be required)
7	1=Command Data Parity Fault
6	1=Interface Fault
5	1=Invalid or Unimplemented command Fault
4	1=Seek Fault
3	1=Write Gate with Track Offset Fault
2	Vendor Unique Status Available
1	1=Write Fault
0	1=Removable Media Changed since last request

6.0 Executing ESDI Commands

As mentioned above, the host can send any of the ESDI commands to the device and receive data and status upon completion.

Below are some sample routines to send command and receive data from the drive.

EXAMPLE 1
REQUEST CONFIGURATION BYTE
(NUMBER OF HEADS)

```
mov   ah,cyllow          ;cylinder low address-1F4
mov   al,00              ;low byte of ESDI command
out   dx,al              ;output to port
mov   dx,cylhigh         ;cylinder low address-1F5
mov   al,33              ;request configuration command
                         ;with modifier for head byte
out   dx,al              ;output to port
mov   dx,cmdreg          ;command register address-1F7
mov   al,initesdi        ;initiate ESDI command-E0
out   dx,al              ;output to port
```

This routine loads the 16 bit ESDI command into the cylinder high and low registers, 1F5 and 1F4, and then loads the initiate ESDI command into the command register, 1F7. The Host should check for the controller to go not busy, then read the data returned by the drive, from the cylinder high and low registers. The controller Status register will reflect any error condition and the Host can request Status from the drive to determine if any device level error occurred.

A 1007A support diskette is available from Western Digital that encorporates these extended command features into three demo programs. These programs are explained in Appendix F.

The above command information is intended to give the user a general overview. Specific information can be obtained from the 1007A engineering specification available from Western Digital, and the ANSI Specification on the ESDI interface, available from the ANSI committee in Washington D.C.

7.0 WDFMT

Western Digital provides a low-level format utility that will allow the user to prepare the drive for use by the system. The program includes routines for low-level formatting, disk verify, surface analysis and bad track entry. When using the 1007A board one should use an interleave of 1, a skew on 2 and format at 35 SPT with an alternate sector. See the explanations for these features provided below.

A. Sector Skewing

Sector Skewing is a method of formatting a drive in which sector numbers are rotated in the interleave table each time a new head on a cylinder is formatted. For example, using a 2 sector skew, the first sector after index on head 0 will be identified as sector 1. The sector identified as sector 1 on head 1 will be the third physical sector from index.

Sector Skew formatting is available with version 2.10 of WDFMT.

Example: Ten sectors per track with a skew of 2 and interleave of 1

Head #	Sector #'s									
0	1	2	3	4	5	6	7	8	9	10
1	9	10	1	2	3	4	5	6	7	8
2	7	8	9	10	1	2	3	4	5	6

This formatting procedure allows the controller to maintain the 1:1 interleave when reading across the head boundary. This becomes critical when the number of sectors per track increases and the time allowed for overhead functions to be completed decreases as with ESDI applications. Because of controller firmware overhead, the controller will not be able to read the ID Field of the first sector on the next head. By changing the sector numbers, the controller can do the needed tasks and be ready to read the sector marked as number 1. The minimum sector skew factor for proper performance of the WD1007A has been determined to be 2. Different skew factors may be needed to optimize performance for different applications.

B. Sector Spare

Another option available in WDFMT Version 2.10 is the ability to format a spare sector on the track. This spare sector is given the ID of zero, making it invisible to the AT compatible System BIOS, which expects sector numbers starting at 1. This sector is always formatted as the last physical sector on the track.

The surface analysis portion of WDFMT will use that spare sector if an error is encountered with any sector on the given track. The program will reformat the track, numbering the bad sector as zero and shifting the following sectors one to the right. If more than one sector is found to be bad on the track, the entire track is marked as bad when reformatted.

This feature is useful since many system BIOS ROMS that support ESDI drives have a sector per track parameter of 34. By using the spare sector option, the drive will look like it has only 34 sectors. The spare sector can be used for the above mentioned bad sector reassignment, or it can be used to store custom data by providing software drivers to use the hidden sector.

D1007A-WA2 JUMPERS CONFIGURATION TABLES

WD1007A-WA2 DEFAULT SETTING		WD1007A-WAH DEFAULT SETTING	
W1	2 - 3	W1	2 - 3
W2	2 - 3	W2	2 - 3
W3	JUMPED	W3	JUMPED
W4	NON JUMPED		
W5	NON JUMPED		
W6	2 - 3		
W7	1 - 2		
W8	NON JUMPED	W8	NON JUMPED
W9	NON JUMPED	W9	NON JUMPED
W10	NON JUMPED	W10	NON JUMPED
W11	JUMPED	W11	NON JUMPED
W12	NON JUMPED	W12	NON JUMPED
W13	UNCUT		
W14	NON JUMPED	W14	NON JUMPED
W15	NON JUMPED	W15	NON JUMPED

TABLE I

BIOS ADDRESS RANGES	JUMPER SETTINGS		
	W1	W2	W3
C8000 - C9FFF	2 - 3	2 - 3	JUMPED
CA000 - CBFFF	2 - 3	1 - 2	JUMPED
CC000 - CDFFF	1 - 2	2 - 3	JUMPED
CE000 - CFFFF	1 - 2	1 - 2	JUMPED
DISABLE			NON JUMPED

TABLE II

FLOPPY CONTROLLER	W4	W13 IN ETCH
ENABLE	NON JUMPED	UNCUT
DISABLE	JUMPED	CUT

TABLE III

DRIVE TYPE INPUT	W5
2 SPEEDS SPINDLE MOTOR	JUMPED
SINGLE SPEED	NON JUMPED

TABLE IV

FLOPPY ADDRESS RANGES	W6
37X	1 - 2
3FX	2 - 3

TABLE V

HIGH DENSITY SELECTION	W7
5.25", 1.2M	1 - 2
3.5", 1.44M	1 - 2

TABLE VI

WD1005 MODE	W8	JUMPED
WD1007 MODE		NON JUMPED

TABLE VII

CHASSIS GROUND & DIGITAL GROUND	W9
CONNECTED	2 - 3
UNCONNECTED.	1 - 2

TABLE VIII

DIGITAL INPUT REG MODE	W10
LATCHED	JUMPED
NON LATCHED	NON JUMPED

TABLE IX

DISKCHANGE INPUT	W11
WITH FDC OPTION	JUMPED
WITHOUT FDC OPTION	NON JUMPED

TABLE X

PRIMARY/SECONDARY HARD DISK ADDRESSES	W12
1FX	NON JUMPED
17X	JUMPED

TABLE XI

SECTOR TRANSLATION	W14
ENABLE	NON JUMPED
DISABLE	JUMPED

TABLE XII

ECC SELECTION	W15
7 BYTES	JUMPED
4 BYTES	NON JUMPED

WD1005 Mode - The controller reads 'Unformatted bytes/sector' from drive
WD1007 Mode - The controller reads 'Set Unformatted bytes/sector' command to the drive.

WD1007A TRACK FORMAT AND TIMING